

NORTHWEST NAZARENE UNIVERSITY

Android Application for Completing Medical Forms

THESIS

Submitted to the Department of Mathematics and Computer Science  
in partial fulfillment of the requirements  
for the degree of  
BACHELOR OF ARTS

Matthew Rankin  
2013

THESIS

Submitted to the Department of Mathematics and Computer Science  
in partial fulfillment of the requirements  
for the degree of  
BACHELOR OF ARTS

Matthew Rankin  
2013

Android Application for Completing Medical Forms

Author: Matthew W. Rankin  
Matthew Rankin

Approved: Barry Myers  
Dr. Barry Myers, PhD, Associate Professor of Computer Science,  
Department of Mathematics and Computer Science, Faculty Advisor

Approved: Ken Becker  
Ken Becker, Head Athletic Trainer of Northwest Nazarene University  
Second Reader

Approved: Barry Myers  
Dr. Barry Myers, PhD, Chair,  
Department of Mathematics & Computer Science

## ABSTRACT

Android Application for Medical Forms.

RANKIN, MATTHEW (Department of Computer Science), MYERS, DR. BARRY  
(Department of Computer Science).

An application for providing a simplified and more efficient way for filling out medical forms on Android-based mobile devices was created. The application consists of a main menu with six medical forms that are filled out by athletes. Logic and simplicity are used to minimize the amount of time and effort it takes to fill out the forms. It uses edit text fields, check boxes, toggles, spinners, and a canvas for capturing the user's signature. Some features that are included in the application to minimize time and effort include date selectors that are preset to the phone's system time and toggles preset to the most common selection. After each form is completed and the signature is written, the user is given the option to click a finish button. When the finish button is selected, the form and signature are written to the phone's local store as a text file and PNG file, respectively. The text and PNG files are then attached to Android's built in Gmail application and sent to a specified recipient.

## **ACKNOWLEDGEMENTS**

I would like to first acknowledge my friends and family who helped me with the testing of my Android application. I would also like to thank Ken Becker for giving me the opportunity to develop an Android application to try to help make the process of filling out sports medical forms easier and for answering questions that I had. I also want to thank Dr. Barry Myers for being such a good advisor over the 4 years I spent at Northwest Nazarene University.

**TABLE OF CONTENTS**

**Title Page ..... i**

**Signature Page..... ii**

**Abstract..... iii**

**Acknowledgements ..... iv**

**Table of Contents .....v**

**Background .....1**

**Overview of the Project .....3**

**Target Audience .....3**

**Decisions.....4**

**Problems .....6**

**Unfinished Portions and Additions .....9**

**Marketing Strategy .....10**

**Conclusion .....11**

**References .....13**

**Appendix A .....13**

**CompHealthHistoryPart1.java .....13**

**layout\_comp\_health\_history\_part1.xml .....26**

**PrePartMedQuestPart2.java.....41**

**layout\_pre\_part\_med\_quest\_part2.xml.....79**

<b>PhysicalExamPart3.java</b> .....	<b>114</b>
<b>physical_exam_part3.xml</b> .....	<b>117</b>
<b>AthletesInsurInfoPart4.java</b> .....	<b>123</b>
<b>athletes_insur_info_part4.xml</b> .....	<b>128</b>
<b>ProvMedTreatPart5.java</b> .....	<b>138</b>
<b>prov_med_treat_part5.xml</b> .....	<b>140</b>
<b>InterimhealthformPart6.java</b> .....	<b>144</b>
<b>iterim_health_form_part6.xml</b> .....	<b>164</b>
<b>MainActivity.java</b> .....	<b>174</b>
<b>menu.xml</b> .....	<b>176</b>
<b>CaptureSignature.java</b> .....	<b>178</b>
<b>signature.xml</b> .....	<b>183</b>
<b>arrays.xml</b> .....	<b>185</b>
<b>strings.xml</b> .....	<b>186</b>
<b>styles.xml</b> .....	<b>189</b>
<b>AndroidManifest.xml</b> .....	<b>189</b>

## BACKGROUND

With the growing popularity of mobile devices and the Android operating system, there are many opportunities to simplify life tasks. Since the beginning of the Android operating system in October 2003, Android allows “smarter mobile devices that are more aware of its owner's location and preferences” (Fauguy, 2011). Applications like Google Maps, Hipmunk, and Dropbox have been providing people with GPS navigation, a way to book plane flights and hotel rooms, and even a place to store files with a cloud-based storage system that nearly eliminates the need for a flash drive. With the capabilities of the Android operating system, it seemed like a good idea to develop an Android application that speeds up the process of filling out medical forms for athletes.

The application needed the following functions:

- select checkboxes
- select various options (like *year in school*)
- fill in text fields, get the current date
- provide a digital signature with a timestamp
- write data to file, send files using the stock Android application called Gmail
- import data into the windows program called Injury Management System that was developed by NExTT Solutions

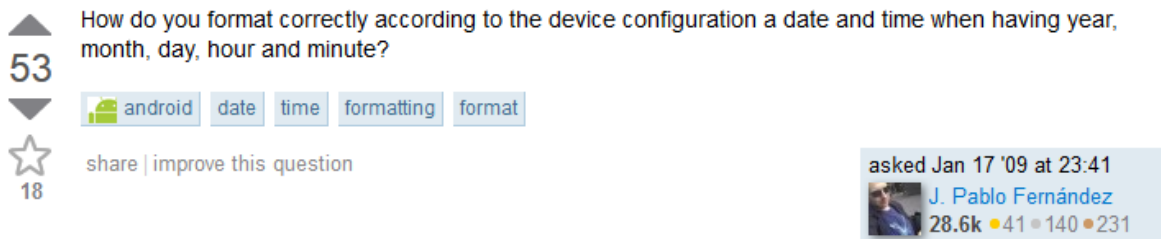
The Injury Management System (IMS) is a program that is used by athletic organizations to help keep accurate records of athletes and injuries that may occur. The primary purpose of the Android application that was developed is to help ease the process of filling out medical forms for sports and provide a tool that will import data into the IMS program to simplify the athletic trainer, Ken Becker’s job.

After analyzing the six medical forms that NNU's athletic department uses, research was then done to find similar solutions that may have already been developed. Some of the solutions that were found were the Android applications called Patient Records Doctor ON GO-T, Onpatient Medical Record PHR, and My Medical Records PHR. Although these applications have many features that go beyond filling out medical forms, they provided examples of functionality that the application would need.

After brainstorming and reviewing some of the basic concepts of Android, an Android application from a Networking II class was reviewed. The application was a Sudoku game that used features that would be incorporated into the project. The Sudoku game used edit text fields, check boxes, and a canvas for capturing user selections on the Sudoku board. Although it did not consist of any spinners, toggles, or uses for writing data to a file, research was carried out on websites that showed how to implement those types of functions. The number one website that was very detailed and provided great examples was [developer.android.com/develop](http://developer.android.com/develop). Another great site that usually had solutions to problems was the Stackoverflow forums. When a specific problem was encountered, some other developer likely has also run into the same problem in the past and has asked for a solution online. In most cases, several solutions would be posted by other developers.

Example of Stackoverflow forum:

Question:



▲ How do you format correctly according to the device configuration a date and time when having year, month, day, hour and minute?


53

▼

android date time formatting format

☆ share | improve this question

asked Jan 17 '09 at 23:41

 J. Pablo Fernández  
28.6k ● 41 ● 140 ● 231



Answer:

9 Answers

active

oldest

votes



Use the standard Java DateFormat class.

63

For example to display the current date and time do the following:



```
Date date = new Date(location.getTime());  
DateFormat dateFormat = android.text.format.DateFormat.getDateFormat(getApplicationContext());  
mTimeText.setText("Time: " + dateFormat.format(date));
```

## OVERVIEW OF THE PROJECT

The application that was developed consists of eight Android activities. Six of the eight activities are medical forms that consist of text, edit text fields, check boxes, toggles, spinners, and date selectors that are used to get medical related information from a user/athlete. The other two activities are the menu activity and the capture signature. The menu activity consists of six medical forms that the user selects and fills out. Upon completion of each form, the user can click a “sign and finish” button, which writes the data obtained from the user to a text file and starts the capture signature activity. In the capture signature activity, the user puts in his/her name and signs on a canvas. When the user selects the “Finished” button, the signature is saved as a PNG file with a timestamp, the text file and PNG file are attached to a Gmail activity, and the user fills in the athletic trainer’s email address before hitting send.

## TARGET AUDIENCE

The target audience that was taken into consideration during the development was all the different types of athletic programs that require medical forms to be filled out. Some of these athletic programs include high school and collegiate athletic programs, NBA, NFL, and the NHL. Medical forms are required to be filled out before participating in athletics at all levels.

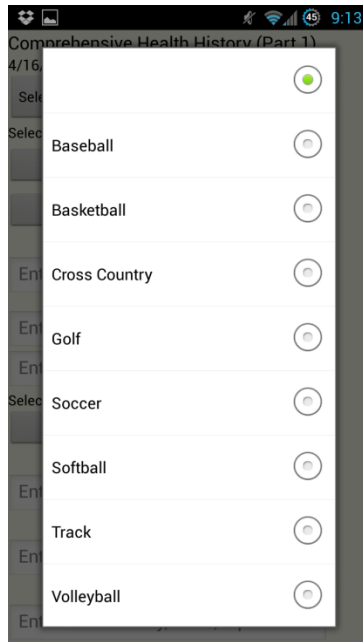
High school, college, and professional sports programs require medical records to be filled out by athletes for accurate records, insurance purposes, injury purposes, and emergency purposes. In many cases, this is a time consuming event for both the athletic trainer and the athlete; however, it has to be done. So at the beginning of each season or career, the process begins with the handing out of the packet of medical forms. Athletes need a paper copy of the forms, pen/pencil, a large chunk of time, and surface to write on. This project aims to eliminate or reduce these constraints.

Since the target audiences were sports related programs, an application name that would be appealing to athletes was created. Some of the original names were Athlete Record Doc, Sport Forms to Go, and SportDoc. The name that was simple and really stuck was SportDoc. The word “sport” in the name relates it to athletics, and the word doc implies that it consists of documents that are to be filled out.

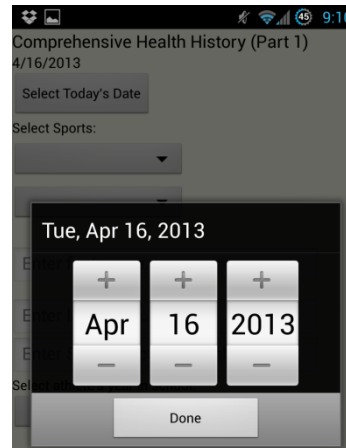
## **DECISIONS**

The process of filling out medical forms can be simplified. By turning it into a digital format many constraints like pen/pencil, a large chunk of time, and a surface to write on can be eliminated. Having medical forms in digital format on the Android operating system makes it possible for users to fill it out with nothing more than a few fingers. The time to fill out medical forms can be reduced by providing functionality that has preset options for the user to select, preset date selectors, and fewer redundant questions.

Preset Option Spinner:



Preset Date Selector:



The primary reasons the Android Operating system was chosen was because of previous experience and the growing popularity it has been gaining in the mobile device market. Although the Apple IOS operating system is also very popular, I do not have as much experience with the objective-C programming language as I do with the java IDE for Android. Another objective that was planned for was porting the Android app over to the IOS operating system if the essential features were first created on Android. However, due to a few problems trying to implement a camera feature for taking pictures of insurance cards and importing data into an Injury Management System program the porting over to IOS was not accomplished.

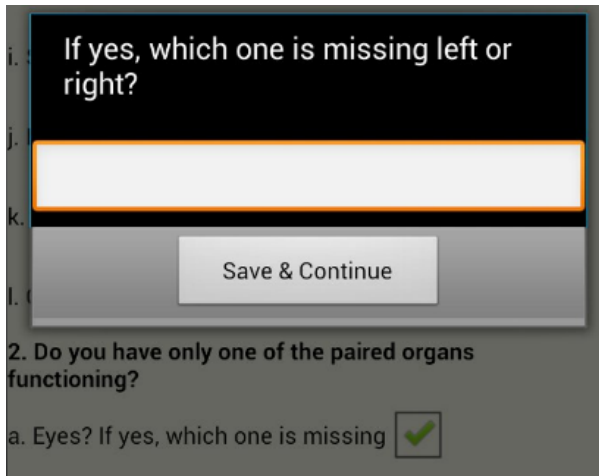
Another decision made at the beginning of development was whether to make the Android app a web-based, native app, or a hybrid of both. The advantage of web-based is the ability to use HTML, CSS and JS for most mobile platforms. It also makes it simpler to fix bugs

in real time. The disadvantages of web-based are that the user cannot access all hardware functions and cannot sell the app. The advantage of a native app is the ability to take advantage of all hardware features, the ability to sell the app, and to use the application when internet access is not available. The disadvantage of native apps is the different platforms, SDK's, and programming languages that have to be used to develop it. There is also a developer fee to submit an app to the iPhone or Android marketplace along with a tedious process of getting the app accepted.

SportDoc is designed for filling out medical forms in any place a user might be. So a native app was developed. Since mobile devices' internet access varies, it is important for users not to need good internet access to fill out medical forms. A web-based application would require a good internet connection for desirable use and would be less secure given that confidential information like Social Security Numbers would be sent wirelessly. It is in the best interest of the users to store this type of information in their phone's local storage before sending it all at once via the designated Gmail app using SSL (Secure Socket Layer).

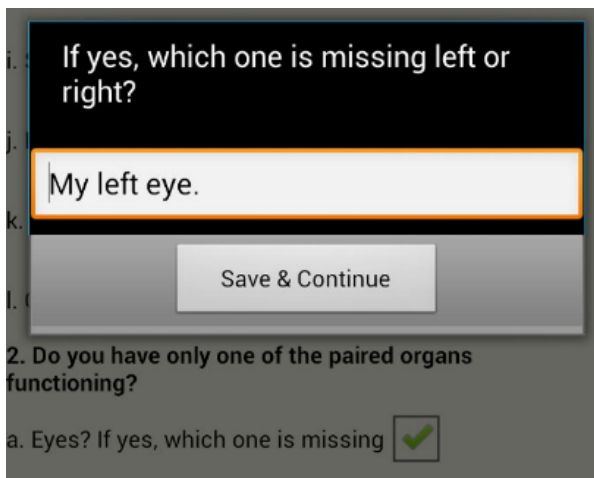
## **PROBLEMS**

A problem encountered in the development of the app was getting a checkbox dialog to save text that the user put in so that if it is reopened, the text is still there to edit. Before a solution to the problem was found, the following image is what the user would see after they typed "My left eye" and tried to go back to make a change to perhaps right eye:



As shown above, it was simply blank. The data was being stored, but the user couldn't go back and edit it.

The solution was to embed an on click listener function that will detect if the user selects the box again and then restore the previous text input to the edit text field so it can be modified.



The highlighted portion of the following code is the embedded on click listener that converts the user's previous input to a string and restores it to the edit text field:

```

CheckBox PairedFunctioningEye = (CheckBox) findViewById(R.id.PairedFunctioningEye);
PairedFunctioningEye.setOnClickListener(new View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update

        alertDialog.setMessage("If yes, which one is missing left or
right?");

        input1 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input

        input1.setText (inputrestore1);
        alertDialog.setView(input1);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog) {
                    PairedFunctioningEyeResponse = input1; //
PairedFunctioningEyeResponse is written to file later

                    inputrestore1 = input1.getText ().toString();
                }
            });
        alertDialog.show();
    }
}

```

The second problem that occurred involves the previously described dialog text box. Whenever the “Save & Continue” button is pressed, the activity is reloaded and the user is taken back to the top of the form. This is an undesirable event that could cause users to be unhappy. Extensive research was done and the cause of the bug was not identified. A theory is that the alert dialog reloads the activity by default, thus causing the screen to move to the top of the form.

## UNFINISHED PORTIONS AND ADDITIONS

A portion of the project that was not finished was the camera feature for taking pictures of insurance cards. Although insurance information is filled out in one of the forms, an insurance card is also needed along with the document. So some research on how to implement a camera feature into an Android application was carried out. Research revealed that the stock camera application can be launched from within the application, but the problem was identifying the image's file name that changes with each picture taken and its location so it could be attached to an email to be sent via Gmail. The Android operating system stores images in the /DCIM/Camera directory; however, some of the new ROM's for the latest Jelly Bean operating system gives the user the ability to save camera images in the DCIM/Camera directory in the extSdCard storage and the sdcard storage. This creates another problem because extSdCard storage is an external microSD card that can be inserted into the phone for added storage space, and sdcard storage is internal storage that a lot of new phones have built in.

The camera feature for taking pictures of insurance cards would be a nice feature to have. However, its implementation took place at the end of the project timeline. Time ran out to work on the project, but a good basis of knowledge was obtained so it can be completed at a later time. It may require not using the stock camera application and developing a custom camera application that will allow images to be stored with a desired file name in the SportDoc folder located in local storage. This way, all the text files with form data, digital signature PNG file, and insurance pictures can be stored and attached to an email from the same directory location.

An unfinished portion of the project that proved to be very time consuming is the ability to import information into the Injury Management System program. Several hours were spent on the phone talking with NExTT solution technicians. They were informed that an Android

application that fills out medical forms and writes data to a file was developed. They were then told that it would be desirable to be able to import data into their program to help speed up the process of filling out athlete information. Not much help was given from the technicians and they said that importing data from any type of file that is not associated with the Injury Management System program is probably not possible. However, the import feature in their new version will likely be able to. It seems that data could be imported in xml formats, but several attempts were unsuccessful.

The last portion of the project was porting the Android application over to the Apple IOS platform. The first step was to conduct a search for some tools to help with the conversion process and one tool was found. The main reason there is a lack of tools for this conversion process is because Android and IOS have vastly different user interfaces. First off, Android is programmed in Java on top of a custom Virtual Machine and IOS is programmed in Objective-C. However, the tool that was discovered to aid in the conversion process is called XMLVM. It can be used to cross compile Android application to the iPhone. It is an open source project and is in its early development stages, so it does have some bugs depending on what API features are being ported. Due to the two mobile operating systems being so different, XMLVM offers a compatibility library to link the two OS's.

## **MARKETING STRATEGY**

Since the application was developed on the Android operating system, it must be registered for a Google Play application and the creator must accept the policies and agreements before submitting. The Google Play store is the marketplace where Android users download apps. To get the best advertising of the application after it is accepted to Google Play will require



a few characteristics. The first characteristic that is essential to make the app aesthetically pleasing is to have a high resolution Android Market icon. The icon for the Android app meets the requirements of Google and is a 512 x 512 pixel PNG image. The icon that was created uses a font called glowing steel from [cooltext.com](http://cooltext.com). The center of the text has a shimmery look and is surrounded by a blue glow like so:



The second characteristic that will need to be focused on for marketing is to construct a concise, detailed Android Market description. The first 167 characters are crucial because they are used as keywords when a user searches for an app on Google Play, but it is important to be careful when using a lot of keywords. If the app description is loaded with too many keywords in an attempt to improve search results instead of describing functionality, the app can be denied. However, the description is the second attention-getter behind the Market icon and is very important to grab a user's attention.

The third and final marketing strategy that will be used is the optional promotional material that is allowed on Google Play. This includes providing a feature graphic that can be used on an app page, uploading up to eight screenshots of the app, and creating a short video to highlight the features so the users know exactly what they are buying.

## **CONCLUSION**

Many valuable skills were learned in the development process of this Android application. Android development skills were increased through researching problems encountered, coding a lot in java, and manually writing Android xml layouts. Although the xml layouts can be dynamically created with the Android SDK, it has limited features, so manually

writing the xml layouts was done. Another valuable skill obtained from doing this project is the importance of time management skills. It is important to have a calendar and to write down tasks that are to be accomplished. However, when it comes to developing an application over a long time span, other responsibilities and distractions cut into development time. It is important to make that time up the next opportunity free time is available.

Overall, I am proud of what was accomplished in my project. Six medical forms were created, a digital signature feature, a method to write data to files in an Android phone's local storage, and a method to send the files using the stock Android email application called Gmail was completed. This project has the potential to help athletic organizations of all types streamline the process of filling out medical forms. With the growing mobile market, this project has the potential to become a very useful tool that athletic trainers can use.

## REFERENCES

Fauguy. (2011, August 17). *Google's Android OS: Past, Present, and Future*. Retrieved March 18, 2013, from phonearena: [http://www.phonearena.com/news/Googles-Android-OS-Past-Present-and-Future\\_id21273](http://www.phonearena.com/news/Googles-Android-OS-Past-Present-and-Future_id21273)

## APPENDIX A

### CompHealthHistoryPart1.java

```
package com.matrankin.sportdoc;
import java.io.*;
import android.annotation.SuppressLint;
import android.app.Activity;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.os.Environment;
import android.view.View.OnClickListener;
import android.view.View;
import java.util.Calendar;
import android.app.AlertDialog;
import android.app.DatePickerDialog;
import android.app.Dialog;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

public class CompHealthHistoryPart1 extends Activity{
    public static final int SIGNATURE_ACTIVITY = 1;
    String CurrentDocDirectory =
Environment.getExternalStorageDirectory().getPath() +
"/SportDoc/ComprehensiveHealthHistory(Part1).txt"; // set DocDirectory to
desired path

    /* Private members of the class */
    private TextView pDisplayDate;
    private Button pPickDate;
    private int Year;
    private int Month;
    private int Day;
    private Button btnWriteSDFFile;
    private EditText txtSSN, ParentsName, ParentsAddress,
ParentsCityStateZip, ParentsHomeNumber, ParentsWorkNumber,
OtherChronicOrRecurrentInjury, txtFirstName, txtLastName, AnyAllergyResponse,
RequiredSurgery, DailyBasisMedications, HistoryOfKnockOutOrSeverConcussions,
NeckInjuryHistory, NeckShoulderArmsHandsWeaknessResponse,
BackInjurySportHistory, CorrectiveLensesInSportsResponse;
    private EditText input1, input2, input3, input4;
```

```

    private String inputrestore1, inputrestore2, inputrestore3,
inputrestore4;
    private EditText PairedFunctioningEyeResponse,
PairedFunctioningOvariesResponse, PairedFunctioningTesticlesResponse,
PairedFunctioningKidneysResponse;

    /* Static members of the class */
    static final int DATE_DIALOG_ID = 0; // This integer will uniquely define
the dialog to be used for displaying the date picker.

    /* Callback received when the user "picks" a date in the dialog */
    private DatePickerDialog.OnDateSetListener pDateSetListener = new
DatePickerDialog.OnDateSetListener() {

        public void onDateSet(DatePicker view, int year, int monthOfYear, int
dayOfMonth) {
            Year = year;
            Month = monthOfYear;
            Day = dayOfMonth;
            updateDisplay();
            displayToast();
        }
    };

    /* Updates the date in the TextView */
    private void updateDisplay() {
        pDisplayDate.setText(new StringBuilder().append(Month +
1).append("/").append(Day).append("/").append(Year).append(" ")); // Month
is 0 based so add 1
    }

    /* Displays a notification when the date is updated */
    private void displayToast() {
        Toast.makeText(this, new StringBuilder().append("Date chosen is
").append(pDisplayDate.getText()), Toast.LENGTH_SHORT).show();
    }

    /* Called when the activity is first created. */
    @SuppressWarnings("SdCardPath")
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.layout_comp_health_history_part1);

        /* Capture our View elements */
        pDisplayDate = (TextView) findViewById(R.id.displayDate);
        pPickDate = (Button) findViewById(R.id.pickDate);

        /* Listener for click event of the date button */
        pPickDate.setOnClickListener(new View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View v) {
                showDialog(DATE_DIALOG_ID);
            }
        });

        /* Get the current date */

```

```

final Calendar cal = Calendar.getInstance();
Year = cal.get(Calendar.YEAR);
Month = cal.get(Calendar.MONTH);
Day = cal.get(Calendar.DAY_OF_MONTH);

/* Display the current date in the TextView */
updateDisplay();

// setup text fields and hints for filling out text fields
txtFirstName = (EditText) findViewById(R.id.txtFirstName);
txtFirstName.setHint("Enter first name...");
txtLastName = (EditText) findViewById(R.id.txtLastName);
txtLastName.setHint("Enter last name...");
txtSSN = (EditText) findViewById(R.id.txtSSN);
txtSSN.setHint("Enter Social Security Number...");
ParentsName = (EditText) findViewById(R.id.ParentsName);
ParentsName.setHint("Enter Parents' Name...");
ParentsAddress = (EditText) findViewById(R.id.ParentsAddress);
ParentsAddress.setHint("Enter Parents' Address...");
ParentsCityStateZip = (EditText)
findViewById(R.id.ParentsCityStateZip);
ParentsCityStateZip.setHint("Enter Parents' City, State, Zip...");
ParentsHomeNumber = (EditText) findViewById(R.id.ParentsHomeNumber);
ParentsHomeNumber.setHint("Enter Parents' Home Phone #...");
ParentsWorkNumber = (EditText) findViewById(R.id.ParentsWorkNumber);
ParentsWorkNumber.setHint("Enter Parents' Work Phone #...");
OtherChronicOrRecurrenInjury = (EditText)
findViewById(R.id.OtherChronicOrRecurrenInjury);
OtherChronicOrRecurrenInjury.setHint("");

CheckBox PairedFunctioningEye = (CheckBox)
findViewById(R.id.PairedFunctioningEye);
PairedFunctioningEye.setOnClickListener(new View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
        alertDialog.setMessage("If yes, which one is missing left or
right?");

        input1 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input
        input1.setText(inputrestore1);
        alertDialog.setView(input1);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int which) {
                    PairedFunctioningEyeResponse = input1;
                    inputrestore1 = input1.getText().toString();
                }
            });
        alertDialog.show();
    }
});

CheckBox PairedFunctioningOvaries = (CheckBox)
findViewById(R.id.PairedFunctioningOvaries);

```

```

        PairedFunctioningOvaries.setOnClickListener(new
View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
        alertDialog.setMessage("If yes, which one is missing left or
right?");
        input2 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input
        input2.setText(inputrestore2);
        alertDialog.setView(input2);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    PairedFunctioningOvariesResponse = input2;
                    inputrestore2 = input2.getText().toString();
                }
            });
        alertDialog.show();
    }
});

        CheckBox PairedFunctioningTesticles = (CheckBox)
findViewById(R.id.PairedFunctioningTesticles);
        PairedFunctioningTesticles.setOnClickListener(new
View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
        alertDialog.setMessage("If yes, which one is missing left or
right?");
        input3 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input
        input3.setText(inputrestore3);
        alertDialog.setView(input3);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    PairedFunctioningTesticlesResponse = input3;
                    inputrestore3 = input3.getText().toString();
                }
            });
        alertDialog.show();
    }
});

        CheckBox PairedFunctioningKidneys = (CheckBox)
findViewById(R.id.PairedFunctioningKidneys);
        PairedFunctioningKidneys.setOnClickListener(new
View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update

```

```

        alertDialog.setMessage("If yes, which one is missing left or
right?");
        input4 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input
        input4.setText(inputrestore4);
        alertDialog.setView(input4);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    PairedFunctioningKidneysResponse = input4;
                    inputrestore4 = input4.getText().toString();
                }
            });
        alertDialog.show();
    }
});

CheckBox AnyAllergy = (CheckBox) findViewById(R.id.ChecKBox13);
AnyAllergy.setOnClickListener(new View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
        alertDialog.setMessage("If yes, list the foods, insects, and
medications you have allergies to:");
        input1 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input1
        input1.setText(inputrestore1);
        alertDialog.setView(input1);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    AnyAllergyResponse = input1;
                    inputrestore1 = input1.getText().toString();
                }
            });
        alertDialog.show();
    }
});

CheckBox RequiredSurgeryQuestion = (CheckBox)
findViewById(R.id.ChecKBox14);
RequiredSurgeryQuestion.setOnClickListener(new View.OnClickListener()
{
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
        alertDialog.setMessage("If yes, list the required surgery you
had for a medical illness or injury:");
        input1 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input1
        input1.setText(inputrestore1);
        alertDialog.setView(input1);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {

```

```

        RequiredSurgery = input1;
        inputrestore1 = input1.getText().toString();
    }
    });
    alertDialog.show();
}
});

CheckBox DailyUsedMedications = (CheckBox)
findViewById(R.id.CheckBox15);
DailyUsedMedications.setOnClickListener(new View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
        alertDialog.setMessage("If yes, list the medications you take
on a daily basis:");
        input1 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input1
        input1.setText(inputrestore1);
        alertDialog.setView(input1);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    DailyBasisMedications = input1;
                    inputrestore1 = input1.getText().toString();
                }
            });
        alertDialog.show();
    }
});

CheckBox HistoryKnockedOutOrSevereConcussions = (CheckBox)
findViewById(R.id.CheckBox16);
HistoryKnockedOutOrSevereConcussions.setOnClickListener(new
View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
        alertDialog.setMessage("If yes, list whether you have been
knocked out and/or concussions:");
        input1 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input1
        input1.setText(inputrestore1);
        alertDialog.setView(input1);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    HistoryOfKnockOutOrSeverConcussions = input1;
                    inputrestore1 = input1.getText().toString();
                }
            });
        alertDialog.show();
    }
});
});

```



```

        CheckBox NeckInjuries = (CheckBox) findViewById(R.id.ChecKBox17);
        NeckInjuries.setOnClickListener(new View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View view) {
                AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
                alertDialog.setMessage("If yes, list whether you currently
have a neck injury or have had one in the past:");
                input1 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input1
                input1.setText(inputrestore1);
                alertDialog.setView(input1);
                alertDialog.setButton("Save & Continue",
                    new DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialog,int which) {
                            NeckInjuryHistory = input1;
                            inputrestore1 = input1.getText().toString();
                        }
                    });
                alertDialog.show();
            }
        });

        CheckBox NeckShoulderArmsHandsWeakness = (CheckBox)
findViewById(R.id.ChecKBox18);
        NeckShoulderArmsHandsWeakness.setOnClickListener(new
View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View view) {
                AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
                alertDialog.setMessage("If yes, list where the weakness
was:");
                input1 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input1
                input1.setText(inputrestore1);
                alertDialog.setView(input1);
                alertDialog.setButton("Save & Continue",
                    new DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialog,int which) {
                            NeckShoulderArmsHandsWeaknessResponse = input1;
                            inputrestore1 = input1.getText().toString();
                        }
                    });
                alertDialog.show();
            }
        });

        CheckBox BackInjurySportHist = (CheckBox)
findViewById(R.id.ChecKBox19);
        BackInjurySportHist.setOnClickListener(new View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View view) {
                AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
                alertDialog.setMessage("If yes, list what sport you had a
back injury in:");

```

```

        input1 = new EditText(CompHealthHistoryPart1.this); // Set an
EditText view to get user input1
        input1.setText(inputrestore1);
        alertDialog.setView(input1);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    BackInjurySportHistory = input1;
                    inputrestore1 = input1.getText().toString();
                }
            });
        alertDialog.show();
    }
});

    CheckBox CorrectiveLensesInSports = (CheckBox)
findViewById(R.id.CheckBox20);
    CorrectiveLensesInSports.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {
            AlertDialog alertDialog = new
AlertDialog.Builder(CompHealthHistoryPart1.this).create(); // Read Update
            alertDialog.setMessage("If yes, list what type. (E.g.
            glasses, contacts (hard/soft/extended wear));");
            input1 = new EditText(CompHealthHistoryPart1.this); // Set an
            EditText view to get user input1
            input1.setText(inputrestore1);
            alertDialog.setView(input1);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        CorrectiveLensesInSportsResponse = input1;
                        inputrestore1 = input1.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

    btnWriteSDFFile = (Button) findViewById(R.id.signature);
    btnWriteSDFFile.setOnClickListener(new OnClickListener() {

        @SuppressWarnings("SdCardPath")
        public void onClick(View v) {

            // bind GUI elements with local controls
            Spinner YearInSchoolSpinner = (Spinner)
findViewById(R.id.YearInSchoolSpinner);
            String feedbackYearInSchoolSpinner =
YearInSchoolSpinner.getSelectedItem().toString();
            Spinner feedbackSpinner1 = (Spinner)
findViewById(R.id.Spinner1);
            String feedbackType1 =
feedbackSpinner1.getSelectedItem().toString();
            Spinner feedbackSpinner2 = (Spinner)
findViewById(R.id.Spinner2);

```

```

        String feedbackType2 =
feedbackSpinner2.getSelectedItem().toString();
        CheckBox responseCheckbox1 = (CheckBox)
findViewById(R.id.ChecKBox1);
        boolean CheckBoxResponse1 = responseCheckbox1.isChecked();
        String stringCheckBox1 = String.valueOf(CheckBoxResponse1);
        CheckBox responseCheckbox2 = (CheckBox)
findViewById(R.id.ChecKBox2);
        boolean CheckBoxResponse2 = responseCheckbox2.isChecked();
        String stringCheckBox2 = String.valueOf(CheckBoxResponse2);
        CheckBox responseCheckbox3 = (CheckBox)
findViewById(R.id.ChecKBox3);
        boolean CheckBoxResponse3 = responseCheckbox3.isChecked();
        String stringCheckBox3 = String.valueOf(CheckBoxResponse3);
        CheckBox responseCheckbox4 = (CheckBox)
findViewById(R.id.ChecKBox4);
        boolean CheckBoxResponse4 = responseCheckbox4.isChecked();
        String stringCheckBox4 = String.valueOf(CheckBoxResponse4);
        CheckBox responseCheckbox5 = (CheckBox)
findViewById(R.id.ChecKBox5);
        boolean CheckBoxResponse5 = responseCheckbox5.isChecked();
        String stringCheckBox5 = String.valueOf(CheckBoxResponse5);
        CheckBox responseCheckbox6 = (CheckBox)
findViewById(R.id.ChecKBox6);
        boolean CheckBoxResponse6 = responseCheckbox6.isChecked();
        String stringCheckBox6 = String.valueOf(CheckBoxResponse6);
        CheckBox responseCheckbox7 = (CheckBox)
findViewById(R.id.ChecKBox7);
        boolean CheckBoxResponse7 = responseCheckbox7.isChecked();
        String stringCheckBox7 = String.valueOf(CheckBoxResponse7);
        CheckBox responseCheckbox8 = (CheckBox)
findViewById(R.id.ChecKBox8);
        boolean CheckBoxResponse8 = responseCheckbox8.isChecked();
        String stringCheckBox8 = String.valueOf(CheckBoxResponse8);
        CheckBox responseCheckbox9 = (CheckBox)
findViewById(R.id.ChecKBox9);
        boolean CheckBoxResponse9 = responseCheckbox9.isChecked();
        String stringCheckBox9 = String.valueOf(CheckBoxResponse9);
        CheckBox responseCheckbox10 = (CheckBox)
findViewById(R.id.ChecKBox10);
        boolean CheckBoxResponse10 = responseCheckbox10.isChecked();
        String stringCheckBox10 = String.valueOf(CheckBoxResponse10);
        CheckBox responseCheckbox11 = (CheckBox)
findViewById(R.id.ChecKBox11);
        boolean CheckBoxResponse11 = responseCheckbox11.isChecked();
        String stringCheckBox11 = String.valueOf(CheckBoxResponse11);
        CheckBox responseCheckbox12 = (CheckBox)
findViewById(R.id.ChecKBox12);
        boolean CheckBoxResponse12 = responseCheckbox12.isChecked();
        String stringCheckBox12 = String.valueOf(CheckBoxResponse12);
        CheckBox responseCheckbox23 = (CheckBox)
findViewById(R.id.ChecKBox23);
        boolean CheckBoxResponse23 = responseCheckbox23.isChecked();
        String stringCheckBox23 = String.valueOf(CheckBoxResponse23);
        CheckBox responseCheckbox24 = (CheckBox)
findViewById(R.id.ChecKBox24);
        boolean CheckBoxResponse24 = responseCheckbox24.isChecked();

```

```

        String stringCheckBox24 = String.valueOf(CheckBoxResponse24);
        CheckBox responseCheckbox25 = (CheckBox)
findViewById(R.id.ChecKBox25);
        boolean CheckBoxResponse25 = responseCheckbox25.isChecked();
        String stringCheckBox25 = String.valueOf(CheckBoxResponse25);
        CheckBox responseCheckbox26 = (CheckBox)
findViewById(R.id.ChecKBox26);
        boolean CheckBoxResponse26 = responseCheckbox26.isChecked();
        String stringCheckBox26 = String.valueOf(CheckBoxResponse26);

        /* write data to file */
        try {
            File testDir = new
File(Environment.getExternalStorageDirectory().getPath() + "/SportDoc/");
            testDir.mkdirs();// Have the object build the directory
structure SD/SportDoc/
            CaptureSignature.PicBindDir =
"ComprehensiveHealthHistoryPart1"; // Set static string PicBindDir to
Document name for signature validation.
            CaptureSignature.DocDir = CurrentDocDirectory; // Set
static DocDir used in CaptureSignature class to this directory.
(CaptureSignatures uses DocDir to attach txt to email app)
            File myFile = new File(CurrentDocDirectory);
            myFile.createNewFile();

            FileOutputStream fOut = new FileOutputStream(myFile); //
Write info to file
            OutputStreamWriter myOutWriter = new
OutputStreamWriter(fOut);
            myOutWriter.append("date:");
            myOutWriter.append(String.valueOf(Month + 1 + "/")); //
Add one because month is 0 based.
            myOutWriter.append(String.valueOf(Day + "/"));
            myOutWriter.append(String.valueOf(Year));
            myOutWriter.append("\r\n");
            myOutWriter.append("sport1:");
            myOutWriter.append(feedbackType1);
            myOutWriter.append("\r\n");
            myOutWriter.append("sport2:");
            myOutWriter.append(feedbackType2);
            myOutWriter.append("\r\n");
            myOutWriter.append("firstname:");
            myOutWriter.append(txtFirstName.getText());
            myOutWriter.append(" ");
            myOutWriter.append(txtLastName.getText());
            myOutWriter.append("\r\n");
            myOutWriter.append("SSN:");
            myOutWriter.append(txtSSN.getText());
            myOutWriter.append("\r\n");
            myOutWriter.append("Year in school:");
            myOutWriter.append(feedbackYearInSchoolSpinner);
            myOutWriter.append("\r\n");
            myOutWriter.append("Parents name:");
            myOutWriter.append(ParentsName.getText());
            myOutWriter.append("\r\n");
            myOutWriter.append("Parents address:");
            myOutWriter.append(ParentsAddress.getText());

```

```

myOutWriter.append("\r\n");
myOutWriter.append("Parents city, state, and zip:");
myOutWriter.append(ParentsCityStateZip.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Parents home number:");
myOutWriter.append(ParentsHomeNumber.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Parents work number:");
myOutWriter.append(ParentsWorkNumber.getText());
myOutWriter.append("\r\n");
if(stringCheckBox1 == "true"){
    myOutWriter.append("Diabetes response:");
    myOutWriter.append(stringCheckBox1);
    myOutWriter.append("\r\n");
}
if(stringCheckBox2 == "true"){
    myOutWriter.append("Kidney disease response:");
    myOutWriter.append(stringCheckBox2);
    myOutWriter.append("\r\n");
}
if(stringCheckBox3 == "true"){
    myOutWriter.append("Mono (last 12 months)
response:");

    myOutWriter.append(stringCheckBox3);
    myOutWriter.append("\r\n");
}
if(stringCheckBox4 == "true"){
    myOutWriter.append("Epilepsy response:");
    myOutWriter.append(stringCheckBox4);
    myOutWriter.append("\r\n");
}
if(stringCheckBox5 == "true"){
    myOutWriter.append("Asthma (environmental/exercise)
response:");

    myOutWriter.append(stringCheckBox5);
    myOutWriter.append("\r\n");
}
if(stringCheckBox6 == "true"){
    myOutWriter.append("Marfans Syndrome response:");
    myOutWriter.append(stringCheckBox6);
    myOutWriter.append("\r\n");
}
if(stringCheckBox7 == "true"){
    myOutWriter.append("Heart murmurs response:");
    myOutWriter.append(stringCheckBox7);
    myOutWriter.append("\r\n");
}
if(stringCheckBox8 == "true"){
    myOutWriter.append("Anemia (last 12 months)
response:");

    myOutWriter.append(stringCheckBox8);
    myOutWriter.append("\r\n");
}
if(stringCheckBox9 == "true"){
    myOutWriter.append("Sickle cell trait response:");
    myOutWriter.append(stringCheckBox9);
    myOutWriter.append("\r\n");
}

```

```

    }
    if(stringCheckBox10 == "true"){
        myOutWriter.append("Hernia response:");
        myOutWriter.append(stringCheckBox10);
        myOutWriter.append("\r\n");
    }
    if(stringCheckBox11 == "true"){
        myOutWriter.append("Menstrual problems or
irregularities");
        myOutWriter.append(stringCheckBox11);
        myOutWriter.append("\r\n");
    }
    if(stringCheckBox12 == "true"){
        myOutWriter.append("Gall bladder dysfunction or
irregularities response:");
        myOutWriter.append(stringCheckBox12);
        myOutWriter.append("\r\n");
    }
    if(PairedFunctioningEyeResponse != null){
        myOutWriter.append("Paired organs, eyes left or right:");
myOutWriter.append(PairedFunctioningEyeResponse.getText());
        myOutWriter.append("\r\n");
    }
    if(PairedFunctioningOvariesResponse != null){
        myOutWriter.append("Paired organs, ovaries left or
right:");
myOutWriter.append(PairedFunctioningOvariesResponse.getText());
        myOutWriter.append("\r\n");
    }
    if(PairedFunctioningTesticlesResponse != null){
        myOutWriter.append("Paired organs, testicles left or
right:");
myOutWriter.append(PairedFunctioningTesticlesResponse.getText());
        myOutWriter.append("\r\n");
    }
    if(PairedFunctioningKidneysResponse != null){
        myOutWriter.append("Paired organs, kidneys left or
right:");
myOutWriter.append(PairedFunctioningKidneysResponse.getText());
        myOutWriter.append("\r\n");
    }
    if(AnyAllergyResponse != null){
        myOutWriter.append("3. Are you or do you have an
allergy to any kind of food, stinging insects, medications response:");
        myOutWriter.append(AnyAllergyResponse.getText());
        myOutWriter.append("\r\n");
    }
    if(RequiredSurgery != null){
        myOutWriter.append("4. Have you ever had or required
surgery for any medical illness or injury response:");
        myOutWriter.append(RequiredSurgery.getText());
        myOutWriter.append("\r\n");
    }
}

```

```

        if(DailyBasisMedications != null){
            myOutWriter.append("5. Do you use on a daily basis
medication for an illness or condition response:");
            myOutWriter.append(DailyBasisMedications.getText());
            myOutWriter.append("\r\n");
        }
        if(HistoryOfKnockOutOrSeverConcussions != null){
            myOutWriter.append("6. Do you have a history of being
knocked out (unconscious) or had severe concussions while participating in
sports or other activities response:");
            myOutWriter.append(DailyBasisMedications.getText());
            myOutWriter.append("\r\n");
        }
        if(NeckInjuryHistory != null){
            myOutWriter.append("7. Have you had a history of or
currently suffering from neck injuries response:");
            myOutWriter.append(NeckInjuryHistory.getText());
            myOutWriter.append("\r\n");
        }
        if(NeckShoulderArmsHandsWeaknessResponse != null){
            myOutWriter.append("8. Have you had problems with
"burners", "stingers", weakness or numbness within your neck, shoulders, arms
or hands response:");
myOutWriter.append(NeckShoulderArmsHandsWeaknessResponse.getText());
            myOutWriter.append("\r\n");
        }
        if(BackInjurySportHistory != null){
            myOutWriter.append("9. Is there a history of back
injury while participating in sports response:");
            myOutWriter.append(BackInjurySportHistory.getText());
            myOutWriter.append("\r\n");
        }
        if(CorrectiveLensesInSportsResponse != null){
            myOutWriter.append("10. Do you wear corrective lenses
while participating in sports response:");
myOutWriter.append(CorrectiveLensesInSportsResponse.getText());
            myOutWriter.append("\r\n");
        }
        if(OtherChronicOrRecurrenInjury != null){
            myOutWriter.append("11. Other Medical Info
response:");
myOutWriter.append(OtherChronicOrRecurrenInjury.getText());
            myOutWriter.append("\r\n");
        }
        if(stringCheckBox23 == "true"){
            myOutWriter.append("Ever had a lower leg injury
response:");
            myOutWriter.append(stringCheckBox23);
            myOutWriter.append("\r\n");
        }
        if(stringCheckBox24 == "true"){
            myOutWriter.append("Ever suffered a knee injury
response:");
            myOutWriter.append(stringCheckBox24);

```

```

        myOutWriter.append("\r\n");
    }
    if(stringCheckBox25 == "true"){
        myOutWriter.append("Ever had a shoulder injury
response:");

        myOutWriter.append(stringCheckBox25);
        myOutWriter.append("\r\n");
    }
    if(stringCheckBox26 == "true"){
        myOutWriter.append("Ever had any type of hip or
pelvic injury response:");
        myOutWriter.append(stringCheckBox26);
        myOutWriter.append("\r\n");
    }
    myOutWriter.close();
    fOut.close();
} catch (Exception e) {
    Toast.makeText(getBaseContext(), e.getMessage(),
        Toast.LENGTH_SHORT).show();
}

Intent intent = new Intent(CompHealthHistoryPart1.this,
CaptureSignature.class);
startActivityForResult(intent, SIGNATURE_ACTIVITY);
} // onClick
}); // btnWriteSDFFile
} // onCreate
/** Create a new dialog for date picker */
@Override
protected Dialog onCreateDialog(int id) {
    switch (id) {
        case DATE_DIALOG_ID:
            return new DatePickerDialog(this, pDateSetListener, Year, Month,
Day);
    }
    return null;
}
} // AndSDcard

```

## layout\_comp\_health\_history\_part1.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#bbeeff" >

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:background="#FFFFFF1"
        android:isScrollContainer="true"
        android:orientation="vertical" >

```



```

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Comprehensive Health History (Part 1)"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="@color/textcolor" />

<TextView
    android:id="@+id/displayDate"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Date will appear here after being selected"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<Button
    android:id="@+id/pickDate"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select Today's Date"
    android:textColor="@color/textcolor" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select Sports:"
    android:textColor="@color/textcolor" />

<Spinner
    android:id="@+id/Spinner1"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/txtLastName"
    android:layout_marginTop="5dp"
    android:entries="@array/sports" />

<Spinner
    android:id="@+id/Spinner2"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/txtLastName"
    android:layout_marginTop="5dp"
    android:entries="@array/sports" />

<EditText
    android:id="@+id/txtFirstName"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:layout_marginTop="25dp"
    android:ems="10"

```

```

        android:textSize="18sp" />

<EditText
    android:id="@+id/txtLastName"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/TextView01"
    android:layout_marginTop="14dp"
    android:ems="10"
    android:textSize="18sp" />

<EditText
    android:id="@+id/txtSSN"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select athlete's year in school:"
    android:textColor="@color/textcolor" />

<Spinner
    android:id="@+id/YearInSchoolSpinner"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:entries="@array/YearInSchool" />

<EditText
    android:id="@+id/ParentsName"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/LinearLayout1"
    android:layout_marginTop="25dp"
    android:ems="10"
    android:textSize="18sp" />

<EditText
    android:id="@+id/ParentsAddress"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/LinearLayout1"
    android:layout_marginTop="25dp"
    android:ems="10"
    android:textSize="18sp" />

<EditText
    android:id="@+id/ParentsCityStateZip"
    android:layout_width="320dp"

```

```

        android:layout_height="40dp"
        android:layout_alignParentLeft="true"
        android:layout_below="@+id/linearLayout1"
        android:layout_marginTop="25dp"
        android:ems="10"
        android:textSize="18sp" />

<EditText
    android:id="@+id/ParentsHomeNumber"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:layout_marginTop="25dp"
    android:ems="10"
    android:textSize="18sp" />

<EditText
    android:id="@+id/ParentsWorkNumber"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:layout_marginTop="25dp"
    android:ems="10"
    android:textSize="18sp" />

<View
    android:layout_width="fill_parent"
    android:layout_height="1dip"
    android:background="#000000" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:paddingBottom="10dp"
    android:text="Please read carefully and answer each question
completely! General Health Survey"
    android:textColor="#ff0000"
    android:textSize="12sp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="1. Have you ever had problems with or been
diagnosed with any of the following:"
    android:textColor="@color/textcolor"
    android:textStyle="bold" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"

```

```

        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="a. Diabetes"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckBox1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="b. Kidney disease"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckBox2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="c. Mono (last 12 months)"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckBox3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"

```

```

        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="d. Epilepsy"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="e. Asthma (environmental/exercise)"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox5"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="f. Marfans Syndrome"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox6"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"

```

```

        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="g. Heart murmurs"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox7"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="h. Anemia (last 12 months)"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox8"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="i. Sickle cell trait"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox9"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"

```

```

        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="j. Hernia"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckBox10"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="k. Menstrual problems or irregularities"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckBox11"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="l. Gall bladder dysfunction or irregularities"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckBox12"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"

```

```

        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="2. Do you have only one of the paired organs
functioning?"
            android:textColor="@color/textcolor"
            android:textStyle="bold" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="a. Eyes? If yes, which one is missing"
            android:textColor="@color/textcolor" />

            <CheckBox
                android:id="@+id/PairedFunctioningEye"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="b. Ovaries? If yes, which one is missing"
            android:textColor="@color/textcolor" />

            <CheckBox
                android:id="@+id/PairedFunctioningOvaries"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView

```



```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="c. Testicles? If yes, which one is missing"
        android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/PairedFunctioningTesticles"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="d. Kidneys? If yes, which one is missing"
            android:textColor="@color/textcolor" />

            <CheckBox
                android:id="@+id/PairedFunctioningKidneys"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="3. Are you or do you have an allergy to any
kind of food, stinging insects, medications?"
            android:textColor="@color/textcolor"
            android:textStyle="bold" />
    </LinearLayout>

    <CheckBox
        android:id="@+id/CheckBox13"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

```

```

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="4. Have you ever had or required surgery for
any medical illness or injury?"
            android:textColor="@color/textcolor"
            android:textStyle="bold" />
    </LinearLayout>

    <CheckBox
        android:id="@+id/CheckKBox14"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="5. Do you use on a daily basis medication for
an illness or condition?"
            android:textColor="@color/textcolor"
            android:textStyle="bold" />
    </LinearLayout>

    <CheckBox
        android:id="@+id/CheckKBox15"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="6. Do you have a history of being knocked out
(unconscious) or had severe concussions while participating in sports or
other activities?"
            android:textColor="@color/textcolor"
            android:textStyle="bold" />
    </LinearLayout>

    <CheckBox
        android:id="@+id/CheckKBox16"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

```

```

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="7. Have you had a history of or currently
suffering from neck injuries?"
        android:textColor="@color/textcolor"
        android:textStyle="bold" />
</LinearLayout>

<CheckBox
    android:id="@+id/CheckBox17"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="8. Have you had problems with "burners",
"stingers", weakness or numbness within your neck, shoulders, arms or hands?"
        android:textColor="@color/textcolor"
        android:textStyle="bold" />
</LinearLayout>

<CheckBox
    android:id="@+id/CheckBox18"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="9. Is there a history of back injury while
participating in sports?"
        android:textColor="@color/textcolor"
        android:textStyle="bold" />

```

```

</LinearLayout>

<CheckBox
    android:id="@+id/CheckKBox19"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="10. Do you wear corrective lenses while
participating in sports?"
        android:textColor="@color/textcolor"
        android:textStyle="bold" />
</LinearLayout>

<CheckBox
    android:id="@+id/CheckKBox20"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="11. Are there or have there been any other
medical illness or injury, past or present that we should know about for your
own protection?"
        android:textColor="@color/textcolor"
        android:textStyle="bold" />
</LinearLayout>

<EditText
    android:id="@+id/OtherPastOrPresentIllnessOrInjury"
    android:layout_width="320dp"
    android:layout_height="120dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:layout_marginTop="25dp"
    android:ems="10"
    android:textSize="14sp" />

<View
    android:layout_width="fill_parent"

```

```

        android:layout_height="1dip"
        android:background="#000000" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Please read carefully and answer each question
completely! Orthopedic Injury Survey"
        android:textColor="#ff0000"
        android:textStyle="bold" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="1. Have you ever had a lower leg injury?"
        android:textColor="@color/textcolor"
        android:textStyle="bold" />
</LinearLayout>

<CheckBox
    android:id="@+id/CheckBox23"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="2. Have you ever suffered a knee injury?"
        android:textColor="@color/textcolor"
        android:textStyle="bold" />
</LinearLayout>

<CheckBox
    android:id="@+id/CheckBox24"
    android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="3. Have you ever had a shoulder injury?"
        android:textColor="@color/textcolor"
        android:textStyle="bold" />
</LinearLayout>

<CheckBox
    android:id="@+id/CheckKBox25"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="5. Have you had any type of hip or pelvic injury?"
        android:textColor="@color/textcolor"
        android:textStyle="bold" />
</LinearLayout>

<CheckBox
    android:id="@+id/CheckKBox26"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<View
    android:layout_width="fill_parent"
    android:layout_height="1dip"
    android:background="#000000" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"

```

```
        android:text="If there is any other chronic or recurrent injury  
that you have experienced, or would like to expand on injuries listed above,  
please use the space below."
```

```
        android:textColor="@color/textcolor"  
        android:textStyle="bold" />
```

```
</LinearLayout>
```

```
<EditText
```

```
    android:id="@+id/OtherChronicOrRecurrenInjury"  
    android:layout_width="320dp"  
    android:layout_height="120dp"  
    android:layout_alignParentLeft="true"  
    android:layout_below="@+id/linearLayout1"  
    android:layout_marginTop="25dp"  
    android:ems="10"  
    android:textSize="14sp" />
```

```
<TextView
```

```
    android:layout_width="fill_parent"  
    android:layout_height="wrap_content"  
    android:layout_above="@+id/signature"  
    android:layout_alignParentLeft="true"  
    android:paddingBottom="20dp"  
    android:paddingTop="20dp"  
    android:text="@string/Validate"  
    android:textColor="#ff0000"  
    android:textSize="12sp"  
    android:textStyle="bold" />
```

```
<Button
```

```
    android:id="@+id/signature"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_above="@+id/btnWriteSDFile"  
    android:layout_alignParentLeft="true"  
    android:layout_marginBottom="19dp"  
    android:text="Click here to sign and finish" />
```

```
</LinearLayout>
```

```
</ScrollView>
```

## PrePartMedQuestPart2.java

```
package com.mattrankin.sportdoc;  
import java.io.*;  
import java.util.Calendar;  
import android.annotation.SuppressLint;  
import android.app.Activity;  
import android.app.AlertDialog;  
import android.app.DatePickerDialog;  
import android.app.Dialog;  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.os.Bundle;  
import android.os.Environment;
```

```

import android.view.*;
import android.view.View.OnClickListener;
import android.widget.*;

public class PrePartMedQuestPart2 extends Activity {

    public static final int SIGNATURE_ACTIVITY = 1;
    String CurrentDocDirectory =
Environment.getExternalStorageDirectory().getPath() +
"/SportDoc/PreParticipationMedicalQuestionnaire(Part2).txt"; // Initialize
current document directory

    /* Private members of the class */
    private EditText txtName;
    private TextView pDisplayDate;
    private Button pPickDate;
    private int Year;
    private int Month;
    private int Day;
    private Button btnWriteSDFFile;
    private EditText txtSSN, txtDateLastTetanusShot,
txtDateLastHepatitisBShot, txtDateLastAllergyShot, txtDateLastMeaslesShot,
txtDateLastFluShot, txtDateLastInsulinShot;
    private EditText BirthControlsResponse, OtherMedicationsTakingResponse,
FoodsThatAthleteIsAllergicToResponse, OtherAllergiesNotListedResponse,
HaveYouEverHadSurgeryResponse, AreYouCurrentlyUsingAnInhaler,
DiabeticResponse, DiabeticLenghOfDiagnose, DoYouWantToWayMoreOrLess,
presentOrPastTreatmentForDepression, FirstMenstrualPeriod,
LastMenstrualPeriod, LongestTimeBetweenPeriodsLastYearInDays,
FemaleStressFracture, SoftOrCaffinatedDrinksEachWeek,
AlcoholicBeveragesAthleteDrinks, DoYouSmoke;
    private EditText input, input2, input3, input4, input5, input6, input7,
input8, input9, input10, input11;
    private String inputrestore, inputrestore2, inputrestore3, inputrestore4,
inputrestore5, inputrestore6, inputrestore7, inputrestore8, inputrestore9,
inputrestore10, inputrestore11;
    private String MaleOrFemale;
    boolean Accutane, Advil, Albuterol, Aleve, Allegra, Amoxicillin, Aspirin,
Benadryll, Claritin, Ibuprofen, Midol, Minocycline, Insulin, Proventil,
Ritalin, Sudafed, SulfaDrugs, Tavist, Tetracycline, Tylenol, Ventolin,
Zantac, Aspirin2, BeeInsectStings, Codeine, Ibuprofen2, Penicillin,
PollenPlants, SeasonalAllergies, SulfaDrugs2,
MedicalIllnessOrInjurySinceLastCheckupOrSportPhysical,
DoYouHaveAnOngoingOrChronicIllness, HospitalizationOvernight,
AreYouTakingAnyPrescriptionOrNonPrescriptionMedications,
SupplementsOrVitaminsToGainOrLoseWeightORImproveAthleticPerformance,
RashOrHivesDuringOrAfterExercise, PassedOutDuringOrAfterExercise,
DizzyDuringOrAfterExercise,
ChestPainDiscomfortOrUnexplainedShortnessOfBreathWithExercise,
MoreTiredThanFriendsDuringExercise,RacingHeartOrSkippedHeartbeats,HighBloodPr
essureOrHighCholesterol,ToldYouHaveHeartMurmor,FamilyMemberOrRelativeHeartPro
blemsOrDeathBefore50,FamilyMemberOrRelativeSignificantDisabilityForCardioDise
aseOrEventUnderAge50,KnowledgeOfHeartConditionsOrSuddenCardiacEventInFamilyOr
Relative,SevereViralInfectionInLastMonth,PhysicanDeniedOrRestrictedYourPartic
ipationInSportsForHeartProblems,AnySkinProblems,EverHadHeadInjuryOrConcussion
,EverKnockedOutBecomeUnconsciousOrLostMemory,EverHadASeizure,FrequentOrSeverH
eadaches,NumbnessOrTinglingInArmsHandLegsOrFeet,IllFromExercisinginHeat,Cough

```



```

WheezeOrHaveTroubleBreathingDuringOrAfterActivity,DoYouHaveAsthma,SeasonalAll
ergiesThatRequireMedicalTreatment,ProblemsOrInjuryToEyesOrVision,WearGlassesC
ontactsOrProtectiveEyeWear,SprainStrainOrSwellingAfterAnInjury,DoYouFeelStres
sedOut,BeenToldYouHaveMarfansSyndrome,DentalCheckupInLast6Months,FemaleEatReg
ularMealsEachDay,RegularlyEatingLeafyGreenVegetables,AreMilkAndMilkProductsAR
egularPartOfYourDiet,EatMeatEggYolksWholeGrainsPeasOrNutsRegularly,DoYouMissM
enstrualPeriodsFrequentlyDuringYourSportingSeason,DiagnosedWithAnyPsychologic
alDisorder,CurrentlyTakingMedicationsForADHD,
AreThereQuestionsYouDoNotCompletelyUnderstand,BrokenOrFracturedAnyBonesOrDisl
ocatedAnyJoints, CheckBox37, CheckBox38, CheckBox39, CheckBox40, CheckBox41,
CheckBox42, CheckBox43, CheckBox44, CheckBox45,
CheckBox46,CurrentlyTakingMedicinesForTreatmentOfSeizures,TakenInhaledCortico
steroidInhalerMedicineForMoreThan6Months,CurrentlyTakingMedicinesForGynecolog
icalDisorderOrDiseases,CurrentlyTakingBirthControlMedication,HistoryInFamilyO
fCusingsSyndrome,EverBeenDiagnosedWithAnyTypeOfThyroidDisorder,IsThereAHistor
yOfOsteoporosisInYourFamily;

```

```

    /* Static members of the class */
    static final int DATE_DIALOG_ID = 0; // This integer will uniquely define
the dialog to be used for displaying the date picker.

```

```

    /* Callback received when the user "picks" a date in the dialog */
    private DatePickerDialog.OnDateSetListener pDateSetListener = new
DatePickerDialog.OnDateSetListener() {

```

```

        public void onDateSet(DatePicker view, int year, int monthOfYear, int
dayOfMonth) {
            Year = year;
            Month = monthOfYear;
            Day = dayOfMonth;
            updateDisplay();
            displayToast();
        }
    };

```

```

    /* Updates the date in the TextView */
    private void updateDisplay() {
        pDisplayDate.setText(
            new StringBuilder().append(Month +
1).append("/").append(Day).append("/").append(Year).append(" "); // Month
is 0 based so add 1
        }

```

```

    /* Displays a notification when the date is updated */
    private void displayToast() {
        Toast.makeText(this, new StringBuilder().append("Date chosen is
").append(pDisplayDate.getText()), Toast.LENGTH_SHORT).show();
    }

```

```

@SuppressLint("SdCardPath")
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.layout_pre_part_med_quest_part2);

```

```

    /* Capture our View elements */

```

```

pDisplayDate = (TextView) findViewById(R.id.displayDate);
pPickDate = (Button) findViewById(R.id.pickDate);

/* Listener for click event of the date button */
pPickDate.setOnClickListener(new View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View v) {
        showDialog(DATE_DIALOG_ID);
    }
});

/* Get the current date */
final Calendar cal = Calendar.getInstance();
Year = cal.get(Calendar.YEAR);
Month = cal.get(Calendar.MONTH);
Day = cal.get(Calendar.DAY_OF_MONTH);

/* Display the current date in the TextView */
updateDisplay();

// setup text fields and hints for filling out text fields
txtName = (EditText) findViewById(R.id.txtName);
txtName.setHint("Enter First, MI, and Last name...");
txtSSN = (EditText) findViewById(R.id.txtSSN);
txtSSN.setHint("Enter Social Security Number...");
txtDateLastTetanusShot = (EditText)
findViewById(R.id.txtDateLastTetanusShot);
txtDateLastTetanusShot.setHint("Enter date of last tetanus shot...");
txtDateLastHepatitisBShot = (EditText)
findViewById(R.id.txtDateLastHepatitisBShot);
txtDateLastHepatitisBShot.setHint("Enter date of last Hepatitis B
shot..");
txtDateLastAllergyShot = (EditText)
findViewById(R.id.txtDateLastAllergyShot);
txtDateLastAllergyShot.setHint("Enter date of last allergy shot..");
txtDateLastMeaslesShot = (EditText)
findViewById(R.id.txtDateLastMeaslesShot);
txtDateLastMeaslesShot.setHint("Enter date of last measles shot..");
txtDateLastFluShot = (EditText)
findViewById(R.id.txtDateLastFluShot);
txtDateLastFluShot.setHint("Enter date of last flu shot..");
txtDateLastInsulinShot = (EditText)
findViewById(R.id.txtDateLastInsulinShot);
txtDateLastInsulinShot.setHint("Enter date of last insulin shot..");
DoYouWantToWayMoreOrLess = (EditText)
findViewById(R.id.DoYouWantToWayMoreOrLess);
DoYouWantToWayMoreOrLess.setHint("Enter date of last insulin
shot..");
FirstMenstrualPeriod = (EditText)
findViewById(R.id.FirstMenstrualPeriod);
FirstMenstrualPeriod.setHint("Month and Year..");
LastMenstrualPeriod = (EditText)
findViewById(R.id.LastMenstrualPeriod);
LastMenstrualPeriod.setHint("Month and Year..");
SoftOrCaffinatedDrinksEachWeek = (EditText)
findViewById(R.id.SoftOrCaffinatedDrinksEachWeek);

```

```

        SoftOrCaffinatedDrinksEachWeek.setHint("Enter how many soft drink /
caffinated drinks you consume each day and week..");
        AlcoholicBeveragesAthleteDrinks = (EditText)
findViewById(R.id.AlcoholicBeveragesAthleteDrinks);
        AlcoholicBeveragesAthleteDrinks.setHint("Enter how many alcoholic
beverages you consume each day, week, and month..");

        ToggleButton MaleOrFemaleToggle = (ToggleButton)
findViewById(R.id.toggleButtonMaleorFemale);
        MaleOrFemale = "Male"; //Initialize MaleOrFemale to default male
        MaleOrFemaleToggle.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (((ToggleButton) v).isChecked()){
                    MaleOrFemale = "Female";
                }
            }
        });

        CheckBox BirthControls = (CheckBox) findViewById(R.id.CheKBox23);
        BirthControls.setOnClickListener(new View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View view) {
                AlertDialog alertDialog = new
AlertDialog.Builder(PartMedQuestPart2.this).create(); // Read Update
                alertDialog.setMessage("If yes, list birth controls:");
                input = new EditText(PartMedQuestPart2.this); // Set an
EditText view to get user input
                input.setText(inputrestore);
                alertDialog.setView(input);
                alertDialog.setButton("Save & Continue",
                    new DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialog,int which) {
                            BirthControlsResponse = input;
                            inputrestore = input.getText().toString();
                        }
                    });
                alertDialog.show();
            }
        });

        CheckBox OtherMedicationsTaking = (CheckBox)
findViewById(R.id.CheKBox24);
        OtherMedicationsTaking.setOnClickListener(new View.OnClickListener())
{
            @SuppressWarnings("deprecation")
            public void onClick(View view) {
                AlertDialog alertDialog = new
AlertDialog.Builder(PartMedQuestPart2.this).create(); // Read Update
                alertDialog.setMessage("If yes, list other medications you
are taking:");
                input2 = new EditText(PartMedQuestPart2.this); // Set an
EditText view to get user input
                input2.setText(inputrestore2);
                alertDialog.setView(input2);
                alertDialog.setButton("Save & Continue",
                    new DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialog,int which) {

```

```

        OtherMedicationsTakingResponse = input2;
        inputrestore2 = input2.getText().toString();
    }
    });
    alertDialog.show();
}
});

    CheckBox FoodsThatAthleteIsAllergicTo = (CheckBox)
findViewById(R.id.CheckBox28);
    FoodsThatAthleteIsAllergicTo.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {
            AlertDialog alertDialog = new
AlertDialog.Builder(PrePartMedQuestPart2.this).create(); // Read Update
            alertDialog.setMessage("If yes, list foods you are allergic
to:");

            input3 = new EditText(PrePartMedQuestPart2.this); // Set an
EditText view to get user input
            input3.setText(inputrestore3);
            alertDialog.setView(input3);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        FoodsThatAthleteIsAllergicToResponse = input3;
                        inputrestore3 = input3.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });
});

    CheckBox OtherAllergiesNotListed = (CheckBox)
findViewById(R.id.CheckBox34);
    OtherAllergiesNotListed.setOnClickListener(new View.OnClickListener()
{
        @SuppressWarnings("deprecation")
        public void onClick(View view) {
            AlertDialog alertDialog = new
AlertDialog.Builder(PrePartMedQuestPart2.this).create(); // Read Update
            alertDialog.setMessage("If yes, list other types of
allergies:");

            input4 = new EditText(PrePartMedQuestPart2.this); // Set an
EditText view to get user input
            input4.setText(inputrestore4);
            alertDialog.setView(input4);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        OtherAllergiesNotListedResponse = input4;
                        inputrestore4 = input4.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });
});

```

```

        CheckBox Diabetic = (CheckBox) findViewById(R.id.ChecKBox35);
        Diabetic.setOnClickListener(new View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View view) {
                AlertDialog alertDialog = new
AlertDialog.Builder(PrePartMedQuestPart2.this).create(); // Read Update
                alertDialog.setMessage("If yes, state what type you have?
Insulin Dependent (type 1) or Non-Insulin Dependent (type 2). How long have
you been diagnosed with this condition? If currently under the care of a
physician please type his name and number below:");
                input5 = new EditText(PrePartMedQuestPart2.this); // Set an
EditText view to get user input
                input5.setText(inputrestore5);
                alertDialog.setView(input5);
                alertDialog.setButton("Save & Continue",
                    new DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialog,int which) {
                            DiabeticResponse = input5;
                            inputrestore5 = input5.getText().toString();
                        }
                    });
                alertDialog.show();
            }
        });

        CheckBox HaveYouEverHadSurgery = (CheckBox)
findViewById(R.id.HaveYouEverHadSurgery);
        HaveYouEverHadSurgery.setOnClickListener(new View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View view) {
                AlertDialog alertDialog = new
AlertDialog.Builder(PrePartMedQuestPart2.this).create(); // Read Update
                alertDialog.setMessage("If yes, what type of surgery?");
                input6 = new EditText(PrePartMedQuestPart2.this); // Set an
EditText view to get user input
                input6.setText(inputrestore6);
                alertDialog.setView(input6);
                alertDialog.setButton("Save & Continue",
                    new DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialog,int which) {
                            HaveYouEverHadSurgeryResponse = input6;
                            inputrestore6 = input6.getText().toString();
                        }
                    });
                alertDialog.show();
            }
        });

        CheckBox CurrentlyUsingAnInhaler = (CheckBox)
findViewById(R.id.AreYouCurrentlyUsingAnInhaler);
        CurrentlyUsingAnInhaler.setOnClickListener(new View.OnClickListener()
{
            @SuppressWarnings("deprecation")
            public void onClick(View view) {
                AlertDialog alertDialog = new
AlertDialog.Builder(PrePartMedQuestPart2.this).create(); // Read Update

```

```

        alertDialog.setMessage("If yes, for what condition?");
        input7 = new EditText(PrePartMedQuestPart2.this); // Set an
EditText view to get user input
        input7.setText(inputrestore7);
        alertDialog.setView(input7);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    AreYouCurrentlyUsingAnInhaler = input7;
                    inputrestore7 = input7.getText().toString();
                }
            });
        alertDialog.show();
    }
});

    CheckBox PresentOrPastTreatmentForDepression = (CheckBox)
findViewById(R.id.PresentOrPastTreatmentForDepression);
    PresentOrPastTreatmentForDepression.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {
            AlertDialog alertDialog = new
AlertDialog.Builder(PrePartMedQuestPart2.this).create(); // Read Update
            alertDialog.setMessage("If so, what medication(s) are you
taking");
            input8 = new EditText(PrePartMedQuestPart2.this); // Set an
EditText view to get user input
            input8.setText(inputrestore8);
            alertDialog.setView(input8);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        presentOrPastTreatmentForDepression = input8;
                        inputrestore8 = input8.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

    CheckBox longestTimeBetweenPeriodsLastYearInDays = (CheckBox)
findViewById(R.id.LongestTimeBetweenPeriodsLastYearInDays);
    longestTimeBetweenPeriodsLastYearInDays.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {
            AlertDialog alertDialog = new
AlertDialog.Builder(PrePartMedQuestPart2.this).create(); // Read Update
            alertDialog.setMessage("If yes, how many days?");
            input9 = new EditText(PrePartMedQuestPart2.this); // Set an
EditText view to get user input
            input9.setText(inputrestore9);
            alertDialog.setView(input9);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {

```

```

        LongestTimeBetweenPeriodsLastYearInDays = input9;
        inputrestore9 = input9.getText().toString();
    }
    });
    alertDialog.show();
}
});

CheckBox femaleStressFracture = (CheckBox)
findViewById(R.id.FemaleStressFracture);
femaleStressFracture.setOnClickListener(new View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view){
        AlertDialog alertDialog = new
AlertDialog.Builder(PrePartMedQuestPart2.this).create(); // Read Update
        alertDialog.setMessage("If so, where?");
        input10 = new EditText(PrePartMedQuestPart2.this); // Set an
EditText view to get user input
        input10.setText(inputrestore10);
        alertDialog.setView(input10);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    FemaleStressFracture = input10;
                    inputrestore10 = input10.getText().toString();
                }
            });
        alertDialog.show();
    }
});

CheckBox doYouSmoke = (CheckBox) findViewById(R.id.DoYouSmoke);
doYouSmoke.setOnClickListener(new View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view){
        AlertDialog alertDialog = new
AlertDialog.Builder(PrePartMedQuestPart2.this).create(); // Read Update
        alertDialog.setMessage("If so, how many per day, per week, or
just occasionally.");
        input11 = new EditText(PrePartMedQuestPart2.this); // Set an
EditText view to get user input11
        input11.setText(inputrestore11);
        alertDialog.setView(input11);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    DoYouSmoke = input11;
                    inputrestore11 = input11.getText().toString();
                }
            });
        alertDialog.show();
    }
});

CheckBox responseCheckbox1 = (CheckBox) findViewById(R.id.ChecKBox1);

```

```

responseCheckbox1.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Accutane = true;
        }
    }
});

CheckBox responseCheckbox2 = (CheckBox) findViewById(R.id.ChecKBox2);
responseCheckbox2.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Advil = true;
        }
    }
});

CheckBox responseCheckbox3 = (CheckBox) findViewById(R.id.ChecKBox3);
responseCheckbox3.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Albuterol = true;
        }
    }
});

CheckBox responseCheckbox4 = (CheckBox) findViewById(R.id.ChecKBox4);
responseCheckbox4.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Aleve = true;
        }
    }
});

CheckBox responseCheckbox5 = (CheckBox) findViewById(R.id.ChecKBox5);
responseCheckbox5.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Allegra = true;
        }
    }
});

CheckBox responseCheckbox6 = (CheckBox) findViewById(R.id.ChecKBox6);
responseCheckbox6.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Amoxicillin = true;
        }
    }
});

CheckBox responseCheckbox7 = (CheckBox) findViewById(R.id.ChecKBox7);
responseCheckbox7.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){

```



```

        Aspirin = true;
    }
}
});

CheckBox responseCheckbox8 = (CheckBox) findViewById(R.id.ChecKBox8);
responseCheckbox8.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Benadryll = true;
        }
    }
});

CheckBox responseCheckbox9 = (CheckBox) findViewById(R.id.ChecKBox9);
responseCheckbox9.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Claritin = true;
        }
    }
});

CheckBox responseCheckbox10 = (CheckBox)
findViewById(R.id.ChecKBox10);
responseCheckbox10.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Ibuprofen = true;
        }
    }
});

CheckBox responseCheckbox11 = (CheckBox)
findViewById(R.id.ChecKBox11);
responseCheckbox11.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Midol = true;
        }
    }
});

CheckBox responseCheckbox12 = (CheckBox)
findViewById(R.id.ChecKBox12);
responseCheckbox12.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Minocycline = true;
        }
    }
});

CheckBox responseCheckbox13 = (CheckBox)
findViewById(R.id.ChecKBox13);
responseCheckbox13.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {

```

```

        if (((CheckBox) v).isChecked()){
            Insulin = true;
        }
    }
});

CheckBox responseCheckbox14 = (CheckBox)
findViewById(R.id.ChecKBox14);
responseCheckbox14.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Proventil = true;
        }
    }
});

CheckBox responseCheckbox15 = (CheckBox)
findViewById(R.id.ChecKBox15);
responseCheckbox15.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Ritalin = true;
        }
    }
});

CheckBox responseCheckbox16 = (CheckBox)
findViewById(R.id.ChecKBox16);
responseCheckbox16.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Sudafed = true;
        }
    }
});

CheckBox responseCheckbox17 = (CheckBox)
findViewById(R.id.ChecKBox17);
responseCheckbox17.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            SulfaDrugs = true;
        }
    }
});

CheckBox responseCheckbox18 = (CheckBox)
findViewById(R.id.ChecKBox18);
responseCheckbox18.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Tavist = true;
        }
    }
});

```

```

        CheckBox responseCheckbox19 = (CheckBox)
findViewById(R.id.ChecKBox19);
        responseCheckbox19.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){
                    Tetracycline = true;
                }
            }
        });

        CheckBox responseCheckbox20 = (CheckBox)
findViewById(R.id.ChecKBox20);
        responseCheckbox20.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){
                    Tylenol = true;
                }
            }
        });

        CheckBox responseCheckbox21 = (CheckBox)
findViewById(R.id.ChecKBox21);
        responseCheckbox21.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){
                    Ventolin = true;
                }
            }
        });

        CheckBox responseCheckbox22 = (CheckBox)
findViewById(R.id.ChecKBox22);
        responseCheckbox22.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){
                    Zantac = true;
                }
            }
        });

        CheckBox responseCheckbox25 = (CheckBox)
findViewById(R.id.ChecKBox25);
        responseCheckbox25.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){
                    Aspirin2 = true;
                }
            }
        });

        CheckBox responseCheckbox26 = (CheckBox)
findViewById(R.id.ChecKBox26);
        responseCheckbox26.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){
                    BeeInsectStings = true;
                }
            }
        });

```

```

    }
});

CheckBox responseCheckbox27 = (CheckBox)
findViewById(R.id.ChecKBox27);
responseCheckbox27.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Codeine = true;
        }
    }
});

CheckBox responseCheckbox29 = (CheckBox)
findViewById(R.id.ChecKBox29);
responseCheckbox29.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Ibuprofen2 = true;
        }
    }
});

CheckBox responseCheckbox30 = (CheckBox)
findViewById(R.id.ChecKBox30);
responseCheckbox30.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            Penicillin = true;
        }
    }
});

CheckBox responseCheckbox31 = (CheckBox)
findViewById(R.id.ChecKBox31);
responseCheckbox31.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            PollenPlants = true;
        }
    }
});

CheckBox responseCheckbox32 = (CheckBox)
findViewById(R.id.ChecKBox32);
responseCheckbox32.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            SeasonalAllergies = true;
        }
    }
});

CheckBox responseCheckbox33 = (CheckBox)
findViewById(R.id.ChecKBox33);
responseCheckbox33.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {

```

```

        if (((CheckBox) v).isChecked()){
            SulfaDrugs2 = true;
        }
    }
});

CheckBox MedicalIllness = (CheckBox)
findViewById(R.id.MedicalIllnessOrInjurySinceLastCheckupOrSportPhysical);
MedicalIllness.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            MedicalIllnessOrInjurySinceLastCheckupOrSportPhysical =
true;
        }
    }
});

CheckBox OngoingOrChronicIllness = (CheckBox)
findViewById(R.id.DoYouHaveAnOngoingOrChronicIllness);
OngoingOrChronicIllness.setOnClickListener(new View.OnClickListener()
{
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            DoYouHaveAnOngoingOrChronicIllness = true;
        }
    }
});

CheckBox HospitalizationOverNight = (CheckBox)
findViewById(R.id.HospitalizationOvernight);
HospitalizationOverNight.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            HospitalizationOvernight = true;
        }
    }
});

CheckBox TakingAnyPrescriptionOrNonPrescriptionMedications =
(CheckBox)
findViewById(R.id.AreYouTakingAnyPrescriptionOrNonPrescriptionMedications);
TakingAnyPrescriptionOrNonPrescriptionMedications.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            AreYouTakingAnyPrescriptionOrNonPrescriptionMedications =
true;
        }
    }
});

CheckBox
SupplementsOrVitaminsToGainOrLoseWeightOrImproveAthleticPerformance =
(CheckBox)

```

```

findViewById(R.id.SupplementsOrVitaminsToGainOrLoseWeightORImproveAthleticPer
formance);

SupplementsOrVitaminsToGainOrLoseWeightOrImproveAthleticPerformance.setOnClic
kListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){

SupplementsOrVitaminsToGainOrLoseWeightORImproveAthleticPerformance = true;
        }
    }
});

CheckBox RashOrHivesExercise = (CheckBox)
findViewById(R.id.SupplementsOrVitaminsToGainOrLoseWeightORImproveAthleticPer
formance);
RashOrHivesExercise.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            RashOrHivesDuringOrAfterExercise = true;
        }
    }
});

CheckBox PassedOutExercise = (CheckBox)
findViewById(R.id.SupplementsOrVitaminsToGainOrLoseWeightORImproveAthleticPer
formance);
PassedOutExercise.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            PassedOutDuringOrAfterExercise = true;
        }
    }
});

CheckBox
chestPainDiscomfortOrUnexplainedSHortnessOfBreathWithExercise = (CheckBox)
findViewById(R.id.ChestPainDiscomfortOrUnexplainedSHortnessOfBreathWithExerci
se);

chestPainDiscomfortOrUnexplainedSHortnessOfBreathWithExercise.setOnClickListener
(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){

ChestPainDiscomfortOrUnexplainedSHortnessOfBreathWithExercise = true;
        }
    }
});

CheckBox moreTiredThanFriendsDuringExercise = (CheckBox)
findViewById(R.id.MoreTiredThanFriendsDuringExercise);
moreTiredThanFriendsDuringExercise.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            MoreTiredThanFriendsDuringExercise = true;

```

```

        }
    }
});

    CheckBox racingHeartOrSkippedHeartbeats = (CheckBox)
findViewById(R.id.RacingHeartOrSkippedHeartbeats);
    racingHeartOrSkippedHeartbeats.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            RacingHeartOrSkippedHeartbeats = true;
        }
    }
});

    CheckBox highBloodPressureOrHighCholesterol = (CheckBox)
findViewById(R.id.HighBloodPressureOrHighCholesterol);
    highBloodPressureOrHighCholesterol.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            HighBloodPressureOrHighCholesterol = true;
        }
    }
});

    CheckBox toldYouHaveHeartMurmor = (CheckBox)
findViewById(R.id.ToldYouHaveHeartMurmor);
    toldYouHaveHeartMurmor.setOnClickListener(new View.OnClickListener()
{
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            ToldYouHaveHeartMurmor = true;
        }
    }
});

    CheckBox familyMemberOrRelativeHeartProblemsOrDeathBefore50 =
(CheckBox)
findViewById(R.id.FamilyMemberOrRelativeHeartProblemsOrDeathBefore50);

familyMemberOrRelativeHeartProblemsOrDeathBefore50.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            FamilyMemberOrRelativeHeartProblemsOrDeathBefore50 =
true;
        }
    }
});

    CheckBox
familyMemberOrRelativeSignificantDisabilityForCardioDiseaseOrEventUnderAge50
= (CheckBox)
findViewById(R.id.FamilyMemberOrRelativeSignificantDisabilityForCardioDisease
OrEventUnderAge50);

```

```

familyMemberOrRelativeSignificantDisabilityForCardioDiseaseOrEventUnderAge50.
setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){

FamilyMemberOrRelativeSignificantDisabilityForCardioDiseaseOrEventUnderAge50
= true;
        }
    }
});

    CheckBox
knowledgeOfHeartConditionsOrSuddenCardiacEventInFamilyOrRelative = (CheckBox)
findViewById(R.id.KnowledgeOfHeartConditionsOrSuddenCardiacEventInFamilyOrRel
ative);

knowledgeOfHeartConditionsOrSuddenCardiacEventInFamilyOrRelative.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){

KnowledgeOfHeartConditionsOrSuddenCardiacEventInFamilyOrRelative = true;
        }
    }
});

    CheckBox severeViralInfectionInLastMonth = (CheckBox)
findViewById(R.id.SevereViralInfectionInLastMonth);
    severeViralInfectionInLastMonth.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            SevereViralInfectionInLastMonth = true;
        }
    }
});

    CheckBox
physicanDeniedOrRestrictedYourParticipationInSportsForHeartProblems =
(CheckBox)
findViewById(R.id.PhysicanDeniedOrRestrictedYourParticipationInSportsForHeart
Problems);

physicanDeniedOrRestrictedYourParticipationInSportsForHeartProblems.setOnclic
kListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){

PhysicanDeniedOrRestrictedYourParticipationInSportsForHeartProblems = true;
        }
    }
});

    CheckBox anySkinProblems = (CheckBox)
findViewById(R.id.AnySkinProblems);
    anySkinProblems.setOnClickListener(new View.OnClickListener() {

```



```

        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                AnySkinProblems = true;
            }
        }
    });

    CheckBox everHadHeadInjuryOrConcussion = (CheckBox)
    findViewById(R.id.EverHadHeadInjuryOrConcussion);
    everHadHeadInjuryOrConcussion.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                EverHadHeadInjuryOrConcussion = true;
            }
        }
    });

    CheckBox everKnockedOutBecomeUnconsciousOrLostMemory = (CheckBox)
    findViewById(R.id.EverKnockedOutBecomeUnconsciousOrLostMemory);
    everKnockedOutBecomeUnconsciousOrLostMemory.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                EverKnockedOutBecomeUnconsciousOrLostMemory = true;
            }
        }
    });

    CheckBox everHadASeizure = (CheckBox)
    findViewById(R.id.EverHadASeizure);
    everHadASeizure.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                EverHadASeizure = true;
            }
        }
    });

    CheckBox frequentOrSeverHeadaches = (CheckBox)
    findViewById(R.id.FrequentOrSeverHeadaches);
    frequentOrSeverHeadaches.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                FrequentOrSeverHeadaches = true;
            }
        }
    });

    CheckBox numbnessOrTinglingInArmsHandLegsOrFeet = (CheckBox)
    findViewById(R.id.NumbnessOrTinglingInArmsHandLegsOrFeet);
    numbnessOrTinglingInArmsHandLegsOrFeet.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                NumbnessOrTinglingInArmsHandLegsOrFeet = true;
            }
        }
    });

```

```

        }
    }
});

CheckBox illFromExercisinginHeat = (CheckBox)
findViewById(R.id.IllFromExercisinginHeat);
illFromExercisinginHeat.setOnClickListener(new View.OnClickListener()
{
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            IllFromExercisinginHeat = true;
        }
    }
});

CheckBox coughWheezeOrHaveTroubleBreathingDuringOrAfterActivity =
(CheckBox)
findViewById(R.id.CoughWheezeOrHaveTroubleBreathingDuringOrAfterActivity);
coughWheezeOrHaveTroubleBreathingDuringOrAfterActivity.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            CoughWheezeOrHaveTroubleBreathingDuringOrAfterActivity =
true;
        }
    }
});

CheckBox doYouHaveAsthma = (CheckBox)
findViewById(R.id.DoYouHaveAsthma);
doYouHaveAsthma.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            DoYouHaveAsthma = true;
        }
    }
});

CheckBox seasonalAllergiesThatRequireMedicalTreatment = (CheckBox)
findViewById(R.id.SeasonalAllergiesThatRequireMedicalTreatment);
seasonalAllergiesThatRequireMedicalTreatment.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            SeasonalAllergiesThatRequireMedicalTreatment = true;
        }
    }
});

CheckBox problemsOrInjuryToEyesOrVision = (CheckBox)
findViewById(R.id.ProblemsOrInjuryToEyesOrVision);
problemsOrInjuryToEyesOrVision.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            ProblemsOrInjuryToEyesOrVision = true;
        }
    }
});

```

```

    }
}
});

CheckBox wearGlassesContactsOrProtectiveEyeWear = (CheckBox)
findViewById(R.id.WearGlassesContactsOrProtectiveEyeWear);
wearGlassesContactsOrProtectiveEyeWear.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            WearGlassesContactsOrProtectiveEyeWear = true;
        }
    }
});

CheckBox sprainStrainOrSwellingAfterAnInjury = (CheckBox)
findViewById(R.id.SprainStrainOrSwellingAfterAnInjury);
sprainStrainOrSwellingAfterAnInjury.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            SprainStrainOrSwellingAfterAnInjury = true;
        }
    }
});

CheckBox brokenOrFracturedAnyBonesOrDislocatedAnyJoints = (CheckBox)
findViewById(R.id.BrokenOrFracturedAnyBonesOrDislocatedAnyJoints);
brokenOrFracturedAnyBonesOrDislocatedAnyJoints.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            BrokenOrFracturedAnyBonesOrDislocatedAnyJoints = true;
        }
    }
});

CheckBox checkBox37 = (CheckBox) findViewById(R.id.CheckBox37);
checkBox37.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            checkBox37 = true;
        }
    }
});

CheckBox checkBox38 = (CheckBox) findViewById(R.id.CheckBox38);
checkBox38.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            checkBox38 = true;
        }
    }
});

CheckBox checkBox39 = (CheckBox) findViewById(R.id.CheckBox39);
checkBox39.setOnClickListener(new View.OnClickListener() {

```

```

        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                ChecKBox39 = true;
            }
        }
    });

    CheckBox checKBox40 = (CheckBox) findViewById(R.id.ChecKBox40);
    checKBox40.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                ChecKBox40 = true;
            }
        }
    });

    CheckBox checKBox41 = (CheckBox) findViewById(R.id.ChecKBox41);
    checKBox41.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                ChecKBox41 = true;
            }
        }
    });

    CheckBox checKBox42 = (CheckBox) findViewById(R.id.ChecKBox42);
    checKBox42.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                ChecKBox42 = true;
            }
        }
    });

    CheckBox checKBox43 = (CheckBox) findViewById(R.id.ChecKBox43);
    checKBox43.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                ChecKBox43 = true;
            }
        }
    });

    CheckBox checKBox44 = (CheckBox) findViewById(R.id.ChecKBox44);
    checKBox44.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                ChecKBox44 = true;
            }
        }
    });

    CheckBox checKBox45 = (CheckBox) findViewById(R.id.ChecKBox45);
    checKBox45.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                ChecKBox45 = true;
            }
        }
    });

```

```

    }
}
});

CheckBox checKBox46 = (CheckBox) findViewById(R.id.ChecKBox46);
checKBox46.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            ChecKBox46 = true;
        }
    }
});

CheckBox doYouFeelStressedOut = (CheckBox)
findViewById(R.id.DoYouFeelStressedOut);
doYouFeelStressedOut.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            DoYouFeelStressedOut = true;
        }
    }
});

CheckBox beenToldYouHaveMarfansSyndrome = (CheckBox)
findViewById(R.id.BeenToldYouHaveMarfansSyndrome);
beenToldYouHaveMarfansSyndrome.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            BeenToldYouHaveMarfansSyndrome = true;
        }
    }
});

CheckBox dentalCheckupInLast6Months = (CheckBox)
findViewById(R.id.DentalCheckupInLast6Months);
dentalCheckupInLast6Months.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            DentalCheckupInLast6Months = true;
        }
    }
});

CheckBox diagnosedWithAnyPsychologicalDisorder = (CheckBox)
findViewById(R.id.DiagnosedWithAnyPsychologicalDisorder);
diagnosedWithAnyPsychologicalDisorder.setOnClickListener(new
View.OnClickListener() {
    public void onClick(View v) {
        if (((CheckBox) v).isChecked()){
            DiagnosedWithAnyPsychologicalDisorder = true;
        }
    }
});

```

```

        CheckBox currentlyTakingMedicationsForADHD = (CheckBox)
findViewById(R.id.CurrentlyTakingMedicationsForADHD);
        currentlyTakingMedicationsForADHD.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){
                    CurrentlyTakingMedicationsForADHD = true;
                }
            }
        });

        CheckBox areThereQuestionsYouDoNotCompletelyUnderstand = (CheckBox)
findViewById(R.id.AreThereQuestionsYouDoNotCompletelyUnderstand);
        areThereQuestionsYouDoNotCompletelyUnderstand.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){
                    AreThereQuestionsYouDoNotCompletelyUnderstand = true;
                }
            }
        });

        CheckBox doYouMissMenstrualPeriodsFrequentlyDuringYourSportingSeason
= (CheckBox)
findViewById(R.id.DoYouMissMenstrualPeriodsFrequentlyDuringYourSportingSeason
);

doYouMissMenstrualPeriodsFrequentlyDuringYourSportingSeason.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){

DoYouMissMenstrualPeriodsFrequentlyDuringYourSportingSeason = true;
                }
            }
        });

        CheckBox femaleEatRegularMealsEachDay = (CheckBox)
findViewById(R.id.FemaleEatRegularMealsEachDay);
        femaleEatRegularMealsEachDay.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){
                    FemaleEatRegularMealsEachDay = true;
                }
            }
        });

        CheckBox regularlyEatingLeafyGreenVegetables = (CheckBox)
findViewById(R.id.RegularlyEatingLeafyGreenVegetables);
        regularlyEatingLeafyGreenVegetables.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View v) {
                if (((CheckBox) v).isChecked()){
                    RegularlyEatingLeafyGreenVegetables = true;
                }
            }
        });

```

```

    });

    CheckBox areMilkAndMilkProductsARegularPartOfYourDiet = (CheckBox)
    findViewById(R.id.AreMilkAndMilkProductsARegularPartOfYourDiet);
    areMilkAndMilkProductsARegularPartOfYourDiet.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                AreMilkAndMilkProductsARegularPartOfYourDiet = true;
            }
        }
    });

    CheckBox eatMeatEggYolksWholeGrainsPeasOrNutsRegularly = (CheckBox)
    findViewById(R.id.EatMeatEggYolksWholeGrainsPeasOrNutsRegularly);
    eatMeatEggYolksWholeGrainsPeasOrNutsRegularly.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                EatMeatEggYolksWholeGrainsPeasOrNutsRegularly = true;
            }
        }
    });

    CheckBox currentltyTakingMedicinesForTreatmentOfSeizures = (CheckBox)
    findViewById(R.id.CurrentltyTakingMedicinesForTreatmentOfSeizures);
    currentltyTakingMedicinesForTreatmentOfSeizures.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                CurrentltyTakingMedicinesForTreatmentOfSeizures = true;
            }
        }
    });

    CheckBox takenInhaledCorticosteroidInhalerMedicineForMoreThan6Months
    = (CheckBox)
    findViewById(R.id.TakenInhaledCorticosteroidInhalerMedicineForMoreThan6Months
    );

    takenInhaledCorticosteroidInhalerMedicineForMoreThan6Months.setOnClickListener
    (new View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                TakenInhaledCorticosteroidInhalerMedicineForMoreThan6Months = true;
            }
        }
    });

    CheckBox currentlyTakingMedicinesForGynecologicalDisorderOrDiseases =
    (CheckBox)
    findViewById(R.id.CurrentlyTakingMedicinesForGynecologicalDisorderOrDiseases)
    ;

    currentlyTakingMedicinesForGynecologicalDisorderOrDiseases.setOnClickListener
    (new View.OnClickListener() {

```

```

        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                CurrentlyTakingMedicinesForGynecologicalDisorderOrDiseases = true;
            }
        }
    });

    CheckBox currentlyTakingBirthControlMedication = (CheckBox)
    findViewById(R.id.CurrentlyTakingBirthControlMedication);
    currentlyTakingBirthControlMedication.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                CurrentlyTakingBirthControlMedication = true;
            }
        }
    });

    CheckBox historyInFamilyOfCusingsSyndrome = (CheckBox)
    findViewById(R.id.HistoryInFamilyOfCusingsSyndrome);
    historyInFamilyOfCusingsSyndrome.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                HistoryInFamilyOfCusingsSyndrome = true;
            }
        }
    });

    CheckBox everBeenDiagnosedWithAnyTypeOfThyroidDisorder = (CheckBox)
    findViewById(R.id.EverBeenDiagnosedWithAnyTypeOfThyroidDisorder);
    everBeenDiagnosedWithAnyTypeOfThyroidDisorder.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                EverBeenDiagnosedWithAnyTypeOfThyroidDisorder = true;
            }
        }
    });

    CheckBox isThereAHistoryOfOsteoporosisInYourFamily = (CheckBox)
    findViewById(R.id.IsThereAHistoryOfOsteoporosisInYourFamily);
    isThereAHistoryOfOsteoporosisInYourFamily.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View v) {
            if (((CheckBox) v).isChecked()){
                IsThereAHistoryOfOsteoporosisInYourFamily = true;
            }
        }
    });

    btnWriteSDFFile = (Button) findViewById(R.id.signature);
    btnWriteSDFFile.setOnClickListener(new OnClickListener() {

        @SuppressWarnings("SdCardPath")
        public void onClick(View v) {

```



```

        // bind GUI elements with local controls
        Spinner feedbackSpinner1 = (Spinner)
findViewById(R.id.Spinner1);
        String feedbackType1 =
feedbackSpinner1.getSelectedItem().toString();
        Spinner feedbackSpinner2 = (Spinner)
findViewById(R.id.Spinner2);
        String feedbackType2 =
feedbackSpinner2.getSelectedItem().toString();
        Spinner AgeSpinner = (Spinner) findViewById(R.id.AgeSpinner);
        String feedbackAgeSpinner =
AgeSpinner.getSelectedItem().toString();
        Spinner YearInSchoolSpinner = (Spinner)
findViewById(R.id.YearInSchoolSpinner);
        String feedbackYearInSchoolSpinner =
YearInSchoolSpinner.getSelectedItem().toString();

        // write on SD card file data in the text box
        try {

            File testDir = new
File(Environment.getExternalStorageDirectory().getPath() +
"/sdcard/SportDoc/");
            testDir.mkdirs(); // Have the object build the directory
structure SD/SportDoc/
            CaptureSignature.PicBindDir =
"PreParticipationMedicalQuestionnairePart2"; // Set static string PicBindDir
to Document name for signature validation.
            CaptureSignature.DocDir = CurrentDocDirectory; //Set
static DocDir used in CaptureSignature class to this directory.
(CaptureSignatures uses DocDir to attach txt to email app)
            File myFile = new File(CurrentDocDirectory);
            myFile.createNewFile();
            FileOutputStream fOut = new FileOutputStream(myFile); //
Write info to file
            OutputStreamWriter myOutWriter = new
OutputStreamWriter(fOut);
            myOutWriter.append("fullname:");
            myOutWriter.append(txtName.getText());
            myOutWriter.append("\r\n");
            myOutWriter.append("BirthDate:");
            myOutWriter.append(String.valueOf(Month + 1 + "/")); //
Add one because month is 0 based.
            myOutWriter.append(String.valueOf(Day + "/"));
            myOutWriter.append(String.valueOf(Year));
            myOutWriter.append("\r\n");
            myOutWriter.append("SSN:");
            myOutWriter.append(txtSSN.getText());
            myOutWriter.append("\r\n");
            myOutWriter.append("MaleOrFemale:");
            myOutWriter.append(MaleOrFemale);
            myOutWriter.append("\r\n");
            myOutWriter.append("AthletesAge:");
            myOutWriter.append(feedbackAgeSpinner);
            myOutWriter.append("\r\n");
            myOutWriter.append("YearInSchool:");

```

```

myOutWriter.append(feedbackYearInSchoolSpinner);
myOutWriter.append("\r\n");
myOutWriter.append("sport1:");
myOutWriter.append(feedbackType1);
myOutWriter.append("\r\n");
myOutWriter.append("sport2:");
myOutWriter.append(feedbackType2);
myOutWriter.append("\r\n");

myOutWriter.append("CurrentMedications:");
if(Accutane == true){
    myOutWriter.append("Accutane, ");
}
if(Advil == true){
    myOutWriter.append("Advil, ");
}
if(Albuterol == true){
    myOutWriter.append("Albuterol, ");
}
if(Aleve == true){
    myOutWriter.append("Aleve, ");
}
if(Allegra == true){
    myOutWriter.append("Allegra, ");
}
if(Amoxicillin == true){
    myOutWriter.append("Amoxicillin, ");
}
if(Aspirin == true){
    myOutWriter.append("Aspirin, ");
}
if(Benadryll == true){
    myOutWriter.append("Benadryll, ");
}
if(Claritin == true){
    myOutWriter.append("Claritin, ");
}
if(Ibuprofen == true){
    myOutWriter.append("Ibuprofen, ");
}
if(Midol == true){
    myOutWriter.append("Midol, ");
}
if(Minocycline == true){
    myOutWriter.append("Minocycline, ");
}
if(Insulin == true){
    myOutWriter.append("Insulin, ");
}
if(Proventil == true){
    myOutWriter.append("Proventil, ");
}
if(Ritalin == true){
    myOutWriter.append("Ritalin, ");
}
if(Sudafed == true){
    myOutWriter.append("Sudafed, ");
}

```

```

    }
    if(SulfaDrugs == true){
        myOutWriter.append("SulfaDrugs, ");
    }
    if(Tavist == true){
        myOutWriter.append("Tavist, ");
    }
    if(Tetracycline == true){
        myOutWriter.append("Tetracycline, ");
    }
    if(Tylenol == true){
        myOutWriter.append("Tylenol, ");
    }
    if(Ventolin == true){
        myOutWriter.append("Ventolin, ");
    }
    if(Zantac == true){
        myOutWriter.append("Zantac, ");
    }
    if(BirthControlsResponse != null){
        myOutWriter.append(BirthControlsResponse.getText());
        myOutWriter.append(", ");
    }
    if(OtherMedicationsTakingResponse != null){
myOutWriter.append(OtherMedicationsTakingResponse.getText());
        myOutWriter.append(", ");
    }
    myOutWriter.append(".");
    myOutWriter.append("\r\n");

    myOutWriter.append("Allergic To:");
    if(Aspirin2 == true){
        myOutWriter.append("Aspirin, ");
    }
    if(BeeInsectStings == true){
        myOutWriter.append("Bee/Insect stings, ");
    }
    if(Codeine == true){
        myOutWriter.append("Codeine, ");
    }
    if(FoodsThatAthleteIsAllergicToResponse != null){
myOutWriter.append(FoodsThatAthleteIsAllergicToResponse.getText());
        myOutWriter.append(", ");
    }
    if(Ibuprofen2 == true){
        myOutWriter.append("Ibuprofen, ");
    }
    if(Penicillin == true){
        myOutWriter.append("Penicillin, ");
    }
    if(PollenPlants == true){
        myOutWriter.append("PollenPlants, ");
    }
    if(SeasonalAllergies == true){
        myOutWriter.append("SeasonalAllergies, ");
    }

```

```

    }
    if(SulfaDrugs == true){
        myOutWriter.append("SulfaDrugs, ");
    }
    if(OtherAllergiesNotListedResponse != null){
myOutWriter.append(OtherAllergiesNotListedResponse.getText());
        myOutWriter.append(", ");
    }
    myOutWriter.append(".");
    myOutWriter.append("\r\n");
    if(DiabeticResponse != null){
        myOutWriter.append("Diabetic response:");
        myOutWriter.append(DiabeticResponse.getText());
        myOutWriter.append("\r\n");
        myOutWriter.append("Length of diagnose:");

myOutWriter.append(DiabeticLenghOfDiagnose.getText());
        myOutWriter.append("\r\n");
    }
    myOutWriter.append("Last tetanus shot date:");
    myOutWriter.append(txtDateLastTetanusShot.getText());
    myOutWriter.append("\r\n");
    myOutWriter.append("Last hepatitis B shot date:");
    myOutWriter.append(txtDateLastHepatitisBShot.getText());
    myOutWriter.append("\r\n");
    myOutWriter.append("Last allergy shot date:");
    myOutWriter.append(txtDateLastAllergyShot.getText());
    myOutWriter.append("\r\n");
    myOutWriter.append("Last Measles shot date:");
    myOutWriter.append(txtDateLastMeaslesShot.getText());
    myOutWriter.append("\r\n");
    myOutWriter.append("Last flu shot date:");
    myOutWriter.append(txtDateLastFluShot.getText());
    myOutWriter.append("\r\n");
    myOutWriter.append("Last insulin shot date:");
    myOutWriter.append(txtDateLastInsulinShot.getText());
    myOutWriter.append("\r\n");

    if(MedicalIllnessOrInjurySinceLastCheckupOrSportPhysical
== true){
        myOutWriter.append("Medical illness or injury since
your last check-up or sport physical response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    if(DoYouHaveAnOngoingOrChronicIllness == true){
myOutWriter.append("Do you have an ongoing or chronic
illness response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    if(HospitalizationOvernight == true){
myOutWriter.append("Have you ever been hospitalized
overnight response:");

```

```

        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if(HaveYouEverHadSurgeryResponse != null){
        myOutWriter.append("Type of surgery athlete has
had:");
myOutWriter.append(HaveYouEverHadSurgeryResponse.getText());
        myOutWriter.append("\r\n");
    }

    if(AreYouTakingAnyPrescriptionOrNonPrescriptionMedications == true){
        myOutWriter.append("Are you currently taking any
prescription or non-prescription medications response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if(AreYouCurrentlyUsingAnInhaler != null){
        myOutWriter.append("Are you currently using an
inhaler response:");
myOutWriter.append(AreYouCurrentlyUsingAnInhaler.getText());
        myOutWriter.append("\r\n");
    }

    if(SupplementsOrVitaminsToGainOrLoseWeightORImproveAthleticPerformance ==
true){
        myOutWriter.append("Have you ever taken any
supplements or vitamins to help you gain or lose weight or Improve your
athletic performance response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if(RashOrHivesDuringOrAfterExercise == true){
        myOutWriter.append("Have you ever had a rash or hives
develop during or after exercise response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if(PassedOutDuringOrAfterExercise == true){
        myOutWriter.append("Have you ever passed out during
or after exercise response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if(DizzyDuringOrAfterExercise == true){
        myOutWriter.append("Have you ever become dizzy during
or after exercise response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    if(ChestPainDiscomfortOrUnexplainedShortnessOfBreathWithExercise == true){
        myOutWriter.append("Have you ever had chest pain,
discomfort or unexplained shortness of breath with exercise response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

```

```

    }
    if(MoreTiredThanFriendsDuringExercise == true){
        myOutWriter.append("Do you get tired more quickly
than your friends do during exercise response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if(RacingHeartOrSkippedHeartbeats == true){
        myOutWriter.append("Have you ever had racing of your
heart or skipped heartbeats response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if(HighBloodPressureOrHighCholesterol == true){
        myOutWriter.append("Have you ever had high blood
pressure or high cholesterol response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if(ToldYouHaveHeartMurmor == true){
        myOutWriter.append("Have you ever been told you have
a heart murmur response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if(FamilyMemberOrRelativeHeartProblemsOrDeathBefore50 ==
true){
        myOutWriter.append("Has any family member or relative
died of heart problems or of sudden death before age 50 response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    if(FamilyMemberOrRelativeSignificantDisabilityForCardioDiseaseOrEventUnderAge
50 == true){
        myOutWriter.append("Has any family member or relative
had significant disability from any cardiovascular disease or event under the
age of 50 response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    if(KnowledgeOfHeartConditionsOrSuddenCardiacEventInFamilyOrRelative == true){
        myOutWriter.append("Do you have knowledge of any
heart conditions or sudden cardiac events in family members or relatives
response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if(SevereViralInfectionInLastMonth == true){
        myOutWriter.append("Have you ever had a severe viral
infection(myocarditis, mononucleosis) within the last month response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
}

```

```

if(PhysicanDeniedOrRestrictedYourParticipationInSportsForHeartProblems ==
true){
    myOutWriter.append("Has a physician ever denied or
restricted your participation in sports for any heart problems response:");
    myOutWriter.append("Yes");
    myOutWriter.append("\r\n");
}
if(AnySkinProblems == true){
    myOutWriter.append("Do you have any skin problems
(psoriasis, itching, rashes, acne) response:");
    myOutWriter.append("Yes");
    myOutWriter.append("\r\n");
}
if(EverHadHeadInjuryOrConcussion == true){
    myOutWriter.append("Have you ever had a head injury
or concussion response:");
    myOutWriter.append("Yes");
    myOutWriter.append("\r\n");
}
if(EverKnockedOutBecomeUnconsciousOrLostMemory == true){
    myOutWriter.append("Have you ever been knocked out,
become unconscious or lost your memory response:");
    myOutWriter.append("Yes");
    myOutWriter.append("\r\n");
}
if(EverHadASeizure == true){
    myOutWriter.append("Have you ever had a seizure
response:");
    myOutWriter.append("Yes");
    myOutWriter.append("\r\n");
}
if(FrequentOrSeverHeadaches == true){
    myOutWriter.append("Do you have frequent or severe
headaches response:");
    myOutWriter.append("Yes");
    myOutWriter.append("\r\n");
}
if(NumbnessOrTinglingInArmsHandLegsOrFeet == true){
    myOutWriter.append("Have you ever had numbness or
tingling in your arms, hands, legs or feet response:");
    myOutWriter.append("Yes");
    myOutWriter.append("\r\n");
}
if(IllFromExercisinginHeat == true){
    myOutWriter.append("Have you ever become ill from
exercising in the heat response:");
    myOutWriter.append("Yes");
    myOutWriter.append("\r\n");
}
if(CoughWheezeOrHaveTroubleBreathingDuringOrAfterActivity
== true){
    myOutWriter.append("Do you cough, wheeze or have
trouble breathing during or after activity response:");
    myOutWriter.append("Yes");
    myOutWriter.append("\r\n");
}
}

```

```

        if(DoYouHaveAsthma == true){
            myOutWriter.append("Do you have asthma response:");
            myOutWriter.append("Yes");
            myOutWriter.append("\r\n");
        }
        if(SeasonalAllergiesThatRequireMedicalTreatment == true){
            myOutWriter.append("Do you have seasonal allergies
that require medical treatment response:");
            myOutWriter.append("Yes");
            myOutWriter.append("\r\n");
        }
        if(ProblemsOrInjuryToEyesOrVision == true){
            myOutWriter.append("Have you had any problems /
injury to your eyes or vision response:");
            myOutWriter.append("Yes");
            myOutWriter.append("\r\n");
        }
        if(WearGlassesContactsOrProtectiveEyeWear == true){
            myOutWriter.append("Do you wear glasses, contacts or
protective eye wear response:");
            myOutWriter.append("Yes");
            myOutWriter.append("\r\n");
        }
        if(SprainStrainOrSwellingAfterAnInjury == true){
            myOutWriter.append("Have you ever had a sprain,
strain or swelling after an injury response:");
            myOutWriter.append("Yes");
            myOutWriter.append("\r\n");
        }
        if(BrokenOrFracturedAnyBonesOrDislocatedAnyJoints ==
true){
            myOutWriter.append("Have you ever broken or fractured
any bones or dislocated any joints response:");
            myOutWriter.append("Yes");
            myOutWriter.append("\r\n");
        }

        myOutWriter.append("Athlete has Injured or had problems
with the following areas:");
        if(CheckBox37 == true){
            myOutWriter.append("Head, ");
        }
        if(CheckBox38 == true){
            myOutWriter.append("Neck, ");
        }
        if(CheckBox39 == true){
            myOutWriter.append(" (Back or chest, ");
        }
        if(CheckBox40 == true){
            myOutWriter.append("Shoulder, ");
        }
        if(CheckBox41 == true){
            myOutWriter.append("Upper arm or elbow or forearm,
");
        }
        if(CheckBox42 == true){
            myOutWriter.append("Wrist or hand or finger, ");

```



```

    }
    if(CheckKB43 == true){
        myOutWriter.append("Hip or thigh, ");
    }
    if(CheckKB44 == true){
        myOutWriter.append("Knee, ");
    }
    if(CheckKB45 == true){
        myOutWriter.append("Shine/calf, ");
    }
    if(CheckKB46 == true){
        myOutWriter.append("Ankle or foot.");
    }
    myOutWriter.append("\r\n");
    myOutWriter.append("Do you want to weigh more or less
response:");

    myOutWriter.append(DoYouWantToWayMoreOrLess.getText());
    myOutWriter.append("\r\n");

    if(DoYouFeelStressedOut == true){
        myOutWriter.append("Do you feel stressed out
response:");

        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    if(BeenToldYouHaveMarfansSyndrome == true){
        myOutWriter.append("Have you ever been told that you
have Marfan's Syndrome response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    if(DentalCheckupInLast6Months == true){
        myOutWriter.append("Have you had a dental checkup in
the last 6 months response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    if(DiagnosedWithAnyPsychologicalDisorder == true){
        myOutWriter.append("Have you been diagnosed with any
type of psychological disorder response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    if( presentOrPastTreatmentForDepression != null){
        myOutWriter.append("Medications taking for depression
response:");

        myOutWriter.append(presentOrPastTreatmentForDepression.getText());
        myOutWriter.append("\r\n");
    }

    if(CurrentlyTakingMedicationsForADHD == true){

```

```

        myOutWriter.append("Are you currently taking any
medications for ADHD:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    if(AreThereQuestionsYouDoNotCompletelyUnderstand ==
true){
        myOutWriter.append("Are there any questions above
that you DO NOT completely understand? response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }

    myOutWriter.append("\r\n");
    myOutWriter.append("\r\n");
    myOutWriter.append("\r\n");
    myOutWriter.append("Month and YearFemale Info below -----
-----");
    myOutWriter.append("\r\n");
    myOutWriter.append("\r\n");
    myOutWriter.append("\r\n");
    myOutWriter.append("Month and Year of first menstrual
period response:");
    myOutWriter.append(FirstMenstrualPeriod.getText());
    myOutWriter.append("\r\n");
    myOutWriter.append("Month and Year of last menstrual
period response:");
    myOutWriter.append>LastMenstrualPeriod.getText());
    myOutWriter.append("\r\n");

    if(DoYouMissMenstrualPeriodsFrequentlyDuringYourSportingSeason == true){
        myOutWriter.append("Do you miss menstrual periods
frequently during your sporting season response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
    if( LongestTimeBetweenPeriodsLastYearInDays != null){
        myOutWriter.append("What was the longest time between
your periods last year in days response:");

myOutWriter.append(LongestTimeBetweenPeriodsLastYearInDays.getText());
        myOutWriter.append("\r\n");
    }
    if( FemaleStressFracture != null){
        myOutWriter.append("Location of stress fracture
response (female):");
        myOutWriter.append(FemaleStressFracture.getText());
        myOutWriter.append("\r\n");
    }
    if(FemaleEatRegularMealsEachDay == true){
        myOutWriter.append("Do you eat regular meals each day
response:");
        myOutWriter.append("Yes");
        myOutWriter.append("\r\n");
    }
}

```

```

        if(RegularlyEatingLeafyGreenVegetables == true){
            myOutWriter.append("Regularly eating leafy green
vegetables response:");
            myOutWriter.append("Yes");
            myOutWriter.append("\r\n");
        }
        if(AreMilkAndMilkProductsARegularPartOfYourDiet == true){
            myOutWriter.append("Are milk and milk products a
regular part of your diet response:");
            myOutWriter.append("Yes");
            myOutWriter.append("\r\n");
        }
        if(EatMeatEggYolksWholeGrainsPeasOrNutsRegularly ==
true){
            myOutWriter.append("Do you eat meat, egg yolks, whole
grains peas or nuts regularly response:");
            myOutWriter.append("Yes");
            myOutWriter.append("\r\n");
        }
        myOutWriter.append("How many soft drink / caffinated
drinks do you drink each week response: (in days and weeks)");
myOutWriter.append(SoftOrCaffinatedDrinksEachWeek.getText());
        myOutWriter.append("\r\n");
        myOutWriter.append("How many alcoholic beverages do you
drink response: (in days, weeks and months)");
myOutWriter.append(AlcoholicBeveragesAthleteDrinks.getText());
        myOutWriter.append("\r\n");

        if( DoYouSmoke != null){
            myOutWriter.append("Do you smoke response:");
            myOutWriter.append(DoYouSmoke.getText());
            myOutWriter.append("\r\n");
        }
        if(CurrenltlyTakingMedicinesForTreatmentOfSeizures ==
true){
            myOutWriter.append("Are you currently taking
medicines for the treatment of seizures response:");
            myOutWriter.append("yes");
            myOutWriter.append("\r\n");
        }

        if(TakenInhaledCorticosteroidInhalerMedicineForMoreThan6Months == true){
            myOutWriter.append("Have you taken inhaled
corticosteroid (inhalers) medicine for more than 6 months response:");
            myOutWriter.append("yes");
            myOutWriter.append("\r\n");
        }

        if(CurrentlyTakingMedicinesForGynecologicalDisorderOrDiseases == true){
            myOutWriter.append("Are you currently taking any
medicines for gynecological disorders / diseases response:");
            myOutWriter.append("yes");
            myOutWriter.append("\r\n");
        }

```

```

        if(CurrentlyTakingBirthControlMedication == true){
            myOutWriter.append("Are you currently taking birth
control medication response:");
            myOutWriter.append("yes");
            myOutWriter.append("\r\n");
        }
        if(HistoryInFamilyOfCusingsSyndrome == true){
            myOutWriter.append("Is there a history in your family
of Cushing's Syndrome response:");
            myOutWriter.append("yes");
            myOutWriter.append("\r\n");
        }
        if(EverBeenDiagnosedWIThAnyTypeOfThyroidDisorder ==
true){
            myOutWriter.append("Have you ever been diagnosed with
any type of thyroid disorder response:");
            myOutWriter.append("yes");
            myOutWriter.append("\r\n");
        }
        if(IsThereAHistoryOfOsteoporosisInYourFamily == true){
            myOutWriter.append("Is there are history of
Osteoporosis in your family response:");
            myOutWriter.append("yes");
            myOutWriter.append("\r\n");
        }

        myOutWriter.close();
        fOut.close();
    } catch (Exception e) {
        Toast.makeText(getApplicationContext(), e.getMessage(),
            Toast.LENGTH_SHORT).show();
    }

    Intent intent = new Intent(PrePartMedQuestPart2.this,
        CaptureSignature.class);
    startActivityForResult(intent, SIGNATURE_ACTIVITY);
} // onClick
}); // btnWriteSDFile

} // onCreate

/** Create a new dialog for date picker */
@Override
protected Dialog onCreateDialog(int id) {
    switch (id) {
        case DATE_DIALOG_ID:
            return new DatePickerDialog(this, pDateSetListener, Year, Month,
Day);
    }
    return null;
}

} // AndSDcard

```

## layout\_pre\_part\_med\_quest\_part2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#bbeeff" >

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:background="#FFFFFF1"
        android:isScrollContainer="true"
        android:orientation="vertical" >

        <TextView
            android:id="@+id/textView2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Pre Participation Medical Questionnaire (Part 2)"
            android:textAppearance="?android:attr/textAppearanceMedium"
            android:textColor="@color/textcolor" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Enter First, MI, and Last Name:"
            android:textColor="@color/textcolor" />

        <EditText
            android:id="@+id/txtName"
            android:layout_width="320dp"
            android:layout_height="40dp"
            android:layout_alignParentLeft="true"
            android:ems="10"
            android:textSize="18sp" />

        <TextView
            android:id="@+id/displayDate"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:text="Your Birth Date will appear here after being
selected"
            android:textColor="@color/textcolor"
            android:textSize="15sp" />

        <Button
            android:id="@+id/pickDate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Select your Birth Date"
            android:textColor="@color/textcolor" />

        <TextView
            android:layout_width="wrap_content"
```

```

        android:layout_height="wrap_content"
        android:text="Enter Your SSN:"
        android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/txtSSN"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Male/Female?"
        android:textColor="@color/textcolor" />

    <ToggleButton
        android:id="@+id/toggleButtonMaleorFemale"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checked="true"
        android:textOff="@string/toggle_female"
        android:textOn="@string/toggle_male" />
</LinearLayout>

<TextView
    android:id="@+id/textView8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select athlete's age:"
    android:textColor="@color/textcolor" />

<Spinner
    android:id="@+id/AgeSpinner"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:entries="@array/age" />

<TextView
    android:id="@+id/textView8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select athlete's year in school:"
    android:textColor="@color/textcolor" />

<Spinner
    android:id="@+id/YearInSchoolSpinner"

```

```

        android:layout_width="180dp"
        android:layout_height="40dp"
        android:layout_alignParentLeft="true"
        android:entries="@array/YearInSchool" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select Sports:"
    android:textColor="@color/textcolor" />

<Spinner
    android:id="@+id/Spinner1"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/txtLastName"
    android:layout_marginTop="5dp"
    android:entries="@array/sports" />

<Spinner
    android:id="@+id/Spinner2"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/txtLastName"
    android:layout_marginTop="5dp"
    android:entries="@array/sports" />

<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="All of the following fields are required
information. Be complete and accurate with the information. Have the Medical
Doctor review and sign this form at the time of your physical exam. Please
provide a copy of your inoculation record with this form. All of the
information provided is confidential and will not be shared with anyone
outside the NNU Trauma Center staff and Physicians. A copy of HIPAA
regulation can be provided if you do not already understand your rights."
    android:textColor="@color/textcolor" />

<View
    android:layout_width="fill_parent"
    android:layout_height="10dip"
    android:background="#000000" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Current Medication(s) :"
    android:textColor="@color/textcolor" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"

```

```

        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Accutane"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Advil"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Albuterol"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"

```



```

        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Aleve"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Allegra"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox5"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Amoxicillin"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox6"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"

```

```

        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Aspirin"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox7"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Benadryll"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox8"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Claritin"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox9"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"

```

```

        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Ibuprofen"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox10"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Midol"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox11"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Minocycline"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox12"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"

```

```

        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Insulin"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox13"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Proventil"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox14"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Ritalin"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox15"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"

```

```

        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Sudafed"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox16"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Sulfa Drugs"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox17"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Tavist"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox18"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"

```

```

        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Tetracycline"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox19"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Tylenol"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox20"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Ventolin"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox21"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"

```

```

        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Zantac"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox22"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Birth Control"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox23"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Other"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox24"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <View
        android:layout_width="fill_parent"
        android:layout_height="10dip"

```

```

        android:background="#000000" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Identify if you are allergic to:"
    android:textColor="@color/textcolor"
    android:textSize="16sp"
    android:textStyle="bold" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Aspirin"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckBox25"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Bee/Insect stings"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckBox26"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"

```



```

        android:layout_height="wrap_content"
        android:text="Codeine"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox27"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Foods"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox28"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Ibuprofen"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox29"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:text="Penicillin"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckBox30"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Pollen/Plants"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckBox31"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Seasonal Allergies"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckBox32"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:text="Sulfa Drugs"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox33"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Others"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox34"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<View
    android:layout_width="fill_parent"
    android:layout_height="10dip"
    android:background="#000000" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Are you a diabetic?"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox35"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"

```

```

        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Are you currently under the care of a
physician?"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckBox36"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Date of last Tetanus Shot:"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="@color/textcolor" />

    <EditText
        android:id="@+id/txtDateLastTetanusShot"
        android:layout_width="320dp"
        android:layout_height="40dp"
        android:layout_alignParentLeft="true"
        android:ems="10"
        android:textSize="18sp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Date of last Hepatitis B shot:"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="@color/textcolor" />

    <EditText
        android:id="@+id/txtDateLastHepatitisBShot"
        android:layout_width="320dp"
        android:layout_height="40dp"
        android:layout_alignParentLeft="true"
        android:ems="10"
        android:textSize="18sp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Date of last Allergy shot:"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="@color/textcolor" />

    <EditText
        android:id="@+id/txtDateLastAllergyShot"
        android:layout_width="320dp"
        android:layout_height="40dp"

```

```

        android:layout_alignParentLeft="true"
        android:ems="10"
        android:textSize="18sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Date of last Measles shot:"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/txtDateLastMeaslesShot"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Date of last Flu shot:"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/txtDateLastFluShot"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Date of last Insulin shot:"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/txtDateLastInsulinShot"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<View
    android:layout_width="fill_parent"
    android:layout_height="10dip"
    android:background="#000000" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"

```

```

        android:text="Have you had a medical illness or injury since your
last check-up or sport physical?"
        android:textColor="@color/textcolor" />

<CheckBox

android:id="@+id/MedicalIllnessOrInjurySinceLastCheckupOrSportPhysical"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Do you have an ongoing or chronic illness?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/DoYouHaveAnOngoingOrChronicIllness"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever been hospitalized overnight?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/HospitalizationOvernight"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever had surgery? If yes, what type:"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/HaveYouEverHadSurgery"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Are you currently taking any prescription or non-
prescription medications?      "
    android:textColor="@color/textcolor" />

<CheckBox

android:id="@+id/AreYouTakingAnyPrescriptionOrNonPrescriptionMedications"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Are you currently using an inhaler?"
        android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/AreYouCurrentlyUsingAnInhaler"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever taken any supplements or vitamins to
help you gain or lose weight or Improve your athletic performance?"
    android:textColor="@color/textcolor" />

<CheckBox

android:id="@+id/SupplementsOrVitaminsToGainOrLoseWeightORImproveAthleticPerf
ormance"

    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever had a rash or hives develop during or
after exercise?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/RashOrHivesDuringOrAfterExercise"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever passed out during or after exercise?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/PassedOutDuringOrAfterExercise"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever become dizzy during or after
exercise?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/DizzyDuringOrAfterExercise"
    android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever had chest pain, discomfort or
unexplained shortness of breath with exercise?"
    android:textColor="@color/textcolor" />

<CheckBox

android:id="@+id/ChestPainDiscomfortOrUnexplainedSHortnessOfBreathWithExercis
e"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Do you get tired more quickly than your friends do
during exercise?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/MoreTiredThanFriendsDuringExercise"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever had racing of your heart or skipped
heartbeats?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/RacingHeartOrSkippedHeartbeats"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever had high blood pressure or high
cholesterol?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/HighBloodPressureOrHighCholesterol"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever been told you have a heart murmur?"
    android:textColor="@color/textcolor" />

```



```

<CheckBox
    android:id="@+id/ToldYouHaveHeartMurmor"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Has any family member or relative died of heart
problems or of sudden death before age 50?"
    android:textColor="@color/textcolor" />

<CheckBox

android:id="@+id/FamilyMemberOrRelativeHeartProblemsOrDeathBefore50"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Has any family member or relative had significant
disability from any cardiovascular disease or event under the age of 50?"
    android:textColor="@color/textcolor" />

<CheckBox

android:id="@+id/FamilyMemberOrRelativeSignificantDisabilityForCardioDiseaseO
rEventUnderAge50"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Do you have knowledge of any heart conditions or
sudden cardiac events in family members or relatives?"
    android:textColor="@color/textcolor" />

<CheckBox

android:id="@+id/KnowledgeOfHeartConditionsOrSuddenCardiacEventInFamilyOrRela
tive"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever had a severe viral
infection(myocarditis, mononucleosis) within the last month?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/SevereViralInfectionInLastMonth"
    android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Has a physician ever denied or restricted your
participation in sports for any heart problems?"
    android:textColor="@color/textcolor" />

<CheckBox

android:id="@+id/PhysicanDeniedOrRestrictedYourParticipationInSportsForHeartP
roblems"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Do you have any skin problems (psoriasis, itching,
rashes, acne)? "
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/AnySkinProblems"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever had a head injury or concussion?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/EverHadHeadInjuryOrConcussion"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever been knocked out, become unconscious
or lost your memory?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/EverKnockedOutBecomeUnconsciousOrLostMemory"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever had a seizure?"
    android:textColor="@color/textcolor" />

```

```

<CheckBox
    android:id="@+id/EverHadASeizure"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Do you have frequent or severe headaches?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/FrequentOrSeverHeadaches"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever had numbness or tingling in your
arms, hands, legs or feet?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/NumbnessOrTinglingInArmsHandLegsOrFeet"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever become ill from exercising in the
heat?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/IllFromExercisinginHeat"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Do you cough, wheeze or have trouble breathing
during or after activity?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/CoughWheezeOrHaveTroubleBreathingDuringOrAfterActivity"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Do you have asthma? "

```

```

        android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/DoYouHaveAsthma"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Do you have seasonal allergies that require medical
treatment?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/SeasonalAllergiesThatRequireMedicalTreatment"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you had any problems / injury to your eyes or
vision?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/ProblemsOrInjuryToEyesOrVision"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Do you wear glasses, contacts or protective eye
wear?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/WearGlassesContactsOrProtectiveEyeWear"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you ever had a sprain, strain or swelling
after an injury?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/SprainStrainOrSwellingAfterAnInjury"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:text="Have you ever broken or fractured any bones or
dislocated any joints?"
        android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/BrokenOrFracturedAnyBonesOrDislocatedAnyJoints"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you injured or had problems with any of the
following areas:"
    android:textColor="@color/textcolor" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Head"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox37"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Neck"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox38"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"

```

```

        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Back, chest"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox39"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Shoulder"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox40"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Upper arm, elbow, forearm"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox41"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"

```

```

        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Wrist, hand, finger"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox42"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Hip, thigh"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox43"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Knee"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox44"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"

```

```

        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Shin/calf"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox45"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Ankle, foot"
            android:textColor="@color/textcolor" />

        <CheckBox
            android:id="@+id/CheckKBox46"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <View
        android:layout_width="fill_parent"
        android:layout_height="10dip"
        android:background="#000000" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Do you want to weigh more or less than you do now?"
        android:textColor="@color/textcolor" />

        <EditText
            android:id="@+id/DoYouWantToWayMoreOrLess"
            android:layout_width="320dp"
            android:layout_height="40dp"
            android:layout_alignParentLeft="true"
            android:ems="10"
            android:textSize="18sp" />

    <TextView

```



```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Do you lose weight regularly to meet weight
requirements for your sport?"
        android:textColor="@color/textcolor" />

<CheckBox

android:id="@+id/CheckBoxDoYouLoseWeightRegularlyToMeetWeigthRequirementsForY
ourSport"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

<TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Do you feel stressed out?"
        android:textColor="@color/textcolor" />

<CheckBox
        android:id="@+id/DoYouFeelStressedOut"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

<TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Have you ever been told that you have Marfan's
Syndrome?"
        android:textColor="@color/textcolor" />

<CheckBox
        android:id="@+id/BeenToldYouHaveMarfansSyndrome"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

<TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Have you had a dental checkup in the last 6
months?"
        android:textColor="@color/textcolor" />

<CheckBox
        android:id="@+id/DentalCheckupInLast6Months"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

<TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Have you been diagnosed with any type of
psychological disorder?"
        android:textColor="@color/textcolor" />

<CheckBox
        android:id="@+id/DiagnosedWithAnyPsychologicalDisorder"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you or are you currently being treated for
depression?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/PresentOrPastTreatmentForDepression"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Are you currently taking any medications for ADHD?
(If so, have your physician complete the Clinical Assessment form)"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/CurrentlyTakingMedicationsForADHD"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Are the any questions above that you DO NOT
completely understand?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/AreThereQuestionsYouDoNotCompletelyUnderstand"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<View
    android:layout_width="fill_parent"
    android:layout_height="10dip"
    android:background="#000000" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="FOR FEMALES ONLY"
    android:textColor="@color/textcolor"
    android:textStyle="bold" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="** All of the information below is confidential and
for medical personnel only**"
    android:textColor="@color/textcolor" />

```

```

<TextView
    android:id="@+id/displayDate"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="When was your first menstrual period?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<EditText
    android:id="@+id/FirstMenstrualPeriod"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="When was your last menstrual period?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<EditText
    android:id="@+id/LastMenstrualPeriod"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Do you miss menstrual periods frequently during
your sporting season? "
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/DoYouMissMenstrualPeriodsFrequentlyDuringYourSportingSeason"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="What was the longest time between your periods last
year in days?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox
    android:id="@+id/LongestTimeBetweenPeriodsLastYearInDays"
    android:layout_width="320dp"
    android:layout_height="40dp"

```

```

        android:layout_alignParentLeft="true"
        android:ems="10"
        android:textSize="18sp" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Have you ever had a stress fracture? "
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox
    android:id="@+id/FemaleStressFracture"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Do you eat regular meals each day?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox
    android:id="@+id/FemaleEatRegularMealsEachDay"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Are you regularly eating leafy green vegetables
(broccoli, spinach, kale)?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox
    android:id="@+id/RegularlyEatingLeafyGreenVegetables"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Are milk and milk products a regular part of your
diet?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox
    android:id="@+id/AreMilkAndMilkProductsARegularPartOfYourDiet"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

```

```

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Do you eat meat, egg yolks, whole grains, peas or
nuts regularly?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox
    android:id="@+id/EatMeatEggYolksWholeGrainsPeasOrNutsRegularly"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="How many soft drink / caffinated drinks do you
drink each week?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<EditText
    android:id="@+id/SoftOrCaffinatedDrinksEachWeek"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="How many alcoholic beverages do you drink?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<EditText
    android:id="@+id/AlcoholicBeveragesAthleteDrinks"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Do you smoke?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox
    android:id="@+id/DoYouSmoke"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView

```

```

        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Are you currently taking medicines for the
treatment of seizures?"
        android:textColor="@color/textcolor"
        android:textSize="15sp" />

<CheckBox
    android:id="@+id/CurrentltyTakingMedicinesForTreatmentOfSeizures"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Have you taken inhaled corticosteroid (inhalers)
medicine for more than 6 months?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox

android:id="@+id/TakenInhaledCorticosteroidInhalerMedicineForMoreThan6Months"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Are you currently taking any medicines for
gynecological disorders / diseases?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox

android:id="@+id/CurrentlyTakingMedicinesForGynecologicalDisorderOrDiseases"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Are you currently taking birth control medication?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox
    android:id="@+id/CurrentlyTakingBirthControlMedication"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Is there a history in your family of Cushing's
Syndrome?"

```

```

        android:textColor="@color/textcolor"
        android:textSize="15sp" />

<CheckBox
    android:id="@+id/HistoryInFamilyOfCusingsSyndrome"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Have you ever been diagnosed with any type of
thyroid disorder?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox
    android:id="@+id/EverBeenDiagnosedWithAnyTypeOfThyroidDisorder"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Is there a history of Osteoporosis in your family?"
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<CheckBox
    android:id="@+id/IsThereAHistoryOfOsteoporosisInYourFamily"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<View
    android:layout_width="fill_parent"
    android:layout_height="10dip"
    android:background="#000000" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Use this space to expand or explain any conditions
from above."
    android:textColor="@color/textcolor"
    android:textSize="15sp" />

<EditText
    android:id="@+id/ExpandOrExplainConditionsFromAbove"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"

```

```

        android:layout_above="@+id/signature"
        android:layout_alignParentLeft="true"
        android:paddingBottom="20dp"
        android:text="@string/Validate"
        android:textColor="#ff0000"
        android:textSize="12sp"
        android:textStyle="bold" />

<Button
    android:id="@+id/signature"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/btnWriteSDFile"
    android:layout_alignParentLeft="true"
    android:layout_marginBottom="19dp"
    android:text="Click here to sign and finish" />
</LinearLayout>

</ScrollView>

```

### PhysicalExamPart3.java

```

package com.mattrankin.sportdoc;
import java.io.*;
import android.annotation.SuppressLint;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Environment;
import android.view.*;
import android.view.View.OnClickListener;
import android.widget.*;

public class PhysicalExamPart3 extends Activity {

    public static final int SIGNATURE_ACTIVITY = 1;
    String CurrentDocDirectory =
Environment.getExternalStorageDirectory().getPath() +
"/SportDoc/PhysicalExam(Part3).txt"; // Initialize current document directory

    // GUI controls
    static EditText txtFirstName; //Shared with CaoptureSignature Activity
    static EditText txtLastName; //Also shared with CaoptureSignature
Activity
    private Button btnWriteSDFile;

    @SuppressWarnings("SdCardPath")
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.physical_exam_part3);

        // setup text fields and hints for filling out text fields
        txtFirstName = (EditText) findViewById(R.id.txtFirstName);
        txtFirstName.setHint("Enter first name...");

```



```

txtLastName = (EditText) findViewById(R.id.txtLastName);
txtLastName.setHint("Enter last name...");
btnWriteSDFFile = (Button) findViewById(R.id.signature);
btnWriteSDFFile.setOnClickListener(new OnClickListener() {

    @SuppressWarnings("SdCardPath")
    public void onClick(View v) {

        // bind GUI elements with local controls
        Spinner feedbackSpinner1 = (Spinner)
findViewById(R.id.Spinner1);
        String feedbackType1 =
feedbackSpinner1.getSelectedItem().toString();
        Spinner feedbackSpinner2 = (Spinner)
findViewById(R.id.Spinner2);
        String feedbackType2 =
feedbackSpinner2.getSelectedItem().toString();
        CheckBox responseCheckbox1 = (CheckBox)
findViewById(R.id.CheckBox1);
        boolean CheckBoxResponse1 = responseCheckbox1.isChecked();
        String stringCheckBox1 = String.valueOf(CheckBoxResponse1);
        CheckBox responseCheckbox2 = (CheckBox)
findViewById(R.id.CheckBox2);
        boolean CheckBoxResponse2 = responseCheckbox2.isChecked();
        String stringCheckBox2 = String.valueOf(CheckBoxResponse2);
        CheckBox responseCheckbox3 = (CheckBox)
findViewById(R.id.CheckBox3);
        boolean CheckBoxResponse3 = responseCheckbox3.isChecked();
        String stringCheckBox3 = String.valueOf(CheckBoxResponse3);
        CheckBox responseCheckbox4 = (CheckBox)
findViewById(R.id.CheckBox4);
        boolean CheckBoxResponse4 = responseCheckbox4.isChecked();
        String stringCheckBox4 = String.valueOf(CheckBoxResponse4);
        CheckBox responseCheckbox5 = (CheckBox)
findViewById(R.id.CheckBox5);
        boolean CheckBoxResponse5 = responseCheckbox5.isChecked();
        String stringCheckBox5 = String.valueOf(CheckBoxResponse5);
        CheckBox responseCheckbox6 = (CheckBox)
findViewById(R.id.CheckBox6);
        boolean CheckBoxResponse6 = responseCheckbox6.isChecked();
        String stringCheckBox6 = String.valueOf(CheckBoxResponse6);
        CheckBox responseCheckbox7 = (CheckBox)
findViewById(R.id.CheckBox7);
        boolean CheckBoxResponse7 = responseCheckbox7.isChecked();
        String stringCheckBox7 = String.valueOf(CheckBoxResponse7);
        CheckBox responseCheckbox8 = (CheckBox)
findViewById(R.id.CheckBox8);
        boolean CheckBoxResponse8 = responseCheckbox8.isChecked();
        String stringCheckBox8 = String.valueOf(CheckBoxResponse8);
        CheckBox responseCheckbox9 = (CheckBox)
findViewById(R.id.CheckBox9);
        boolean CheckBoxResponse9 = responseCheckbox9.isChecked();
        String stringCheckBox9 = String.valueOf(CheckBoxResponse9);
        CheckBox responseCheckbox10 = (CheckBox)
findViewById(R.id.CheckBox10);
        boolean CheckBoxResponse10 = responseCheckbox10.isChecked();
        String stringCheckBox10 = String.valueOf(CheckBoxResponse10);

```

```

        CheckBox responseCheckbox11 = (CheckBox)
findViewById(R.id.ChecKBox11);
        boolean CheckBoxResponse11 = responseCheckbox11.isChecked();
        String stringCheckBox11 = String.valueOf(CheckBoxResponse11);
        CheckBox responseCheckbox12 = (CheckBox)
findViewById(R.id.ChecKBox12);
        boolean CheckBoxResponse12 = responseCheckbox12.isChecked();
        String stringCheckBox12 = String.valueOf(CheckBoxResponse12);

        // write on SD card file data in the text box
        try {

                File testDir= new
File(Environment.getExternalStorageDirectory().getPath() +
"/sdcard/SportDoc/");
                testDir.mkdirs(); // Have the object build the
directory structure SD/SportDoc/
                CaptureSignature.PicBindDir =
"SportDoc/PhysicalExamPart3"; // Set static string PicBindDir to Document
name for signature validation.
                CaptureSignature.DocDir = CurrentDocDirectory; // Set
static DocDir used in CaptureSignature class to this directory.
                (CaptureSignatures uses DocDir to attach txt to email app)
                File myFile = new File(CurrentDocDirectory);
                myFile.createNewFile();

                //destinationFile.createNewFile();
                FileOutputStream fOut = new FileOutputStream(myFile); //
Write info to file
                OutputStreamWriter myOutWriter = new
OutputStreamWriter(fOut);
                myOutWriter.append("<sport1>");
                myOutWriter.append(feedbackType1);
                myOutWriter.append("</sport1>");
                myOutWriter.append("<sport2>");
                myOutWriter.append(feedbackType2);
                myOutWriter.append("</sport2>");
                myOutWriter.append("<firstname>");
                myOutWriter.append(txtFirstName.getText());
                myOutWriter.append("</firstname>");
                myOutWriter.append("<lastname>");
                myOutWriter.append(txtLastName.getText());
                myOutWriter.append("</lastname>");
                myOutWriter.append("<diabetes>");
                myOutWriter.append(stringCheckBox1);
                myOutWriter.append("</diabetes>");
                myOutWriter.append("<kidneydisease>");
                myOutWriter.append(stringCheckBox2);
                myOutWriter.append("</kidneydisease>");
                myOutWriter.append("<mono>");
                myOutWriter.append(stringCheckBox3);
                myOutWriter.append("</mono>");
                myOutWriter.append("<epilepsy>");
                myOutWriter.append(stringCheckBox4);
                myOutWriter.append("</epilepsy>");
                myOutWriter.append("<asthma>");
                myOutWriter.append(stringCheckBox5);

```

```

        myOutWriter.append("</asthma>");
        myOutWriter.append("<marfans>");
        myOutWriter.append(stringCheckBox6);
        myOutWriter.append("</marfans>");
        myOutWriter.append("<heartmurmurs>");
        myOutWriter.append(stringCheckBox7);
        myOutWriter.append("</heartmurmurs>");
        myOutWriter.append("<anemia>");
        myOutWriter.append(stringCheckBox8);
        myOutWriter.append("</anemia>");
        myOutWriter.append("<sicklecell>");
        myOutWriter.append(stringCheckBox9);
        myOutWriter.append("</sicklecell>");
        myOutWriter.append("<hernia>");
        myOutWriter.append(stringCheckBox10);
        myOutWriter.append("</hernia>");
        myOutWriter.append("<menstrual>");
        myOutWriter.append(stringCheckBox11);
        myOutWriter.append("</menstrual>");
        myOutWriter.append("<gallbladder>");
        myOutWriter.append(stringCheckBox12);
        myOutWriter.append("</gallbladder>");

        myOutWriter.close();
        fOut.close();
    } catch (Exception e) {
        Toast.makeText(getApplicationContext(), e.getMessage(),
Toast.LENGTH_SHORT).show();
    }

    Intent intent = new Intent(PhysicalExamPart3.this,
CaptureSignature.class);
    startActivityForResult(intent, SIGNATURE_ACTIVITY);
    } // onClick
}); // btnWriteSDFile

} // onCreate

} // AndSDcard

```

### physical\_exam\_part3.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#bbeeff" >

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:background="#FFFFFF1"
        android:isScrollContainer="true"
        android:orientation="vertical" >

```

```

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Physical Exam (Part 3)"
    android:textColor="@color/textcolor"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select Sports:"
    android:textColor="@color/textcolor" />

<Spinner
    android:id="@+id/Spinner1"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/txtLastName"
    android:layout_marginTop="5dp"
    android:entries="@array/sports" />

<Spinner
    android:id="@+id/Spinner2"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/txtLastName"
    android:layout_marginTop="5dp"
    android:entries="@array/sports" />

<EditText
    android:id="@+id/txtFirstName"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:layout_marginTop="25dp"
    android:ems="10"
    android:textSize="18sp" />

<EditText
    android:id="@+id/txtLastName"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/TextView01"
    android:layout_marginTop="14dp"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:text="1. Have you ever had problems with or been
diagnosed with any of the following:"
        android:textColor="@color/textcolor"/>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="a. Diabetes"
        android:textColor="@color/textcolor"/>

    <CheckBox
        android:id="@+id/CheckBox1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="b. Kidney disease"
        android:textColor="@color/textcolor"/>

    <CheckBox
        android:id="@+id/CheckBox2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="c. Mono (last 12 months)"
        android:textColor="@color/textcolor" />

    <CheckBox

```

```

        android:id="@+id/CheckBox3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="d. Epilepsy"
        android:textColor="@color/textcolor"/>

    <CheckBox
        android:id="@+id/CheckBox4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="e. Asthma (environmental/exercise)"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckBox5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="f. Marfans Syndrome"
        android:textColor="@color/textcolor"/>

    <CheckBox

```

```

        android:id="@+id/CheckBox6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="g. Heart murmurs"
        android:textColor="@color/textcolor"/>

    <CheckBox
        android:id="@+id/CheckBox7"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="h. Anemia (last 12 months)"
        android:textColor="@color/textcolor"/>

    <CheckBox
        android:id="@+id/CheckBox8"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="i. Sickle cell trait"
        android:textColor="@color/textcolor"/>

    <CheckBox

```

```

        android:id="@+id/CheckBox9"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="j. Hernia"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckBox10"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="left"
        android:orientation="horizontal" >

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="k. Menstrual problems or irregularities"
            android:textColor="@color/textcolor"/>

        <CheckBox
            android:id="@+id/CheckBox11"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
</LinearLayout>

        <LinearLayout
            xmlns:android="http://schemas.android.com/apk/res/android"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:gravity="left"
            android:orientation="horizontal" >

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="l. Gall bladder dysfunction or irregularities"
                android:textColor="@color/textcolor" />

            <CheckBox

```



```

        android:id="@+id/CheckBox12"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_above="@+id/signature"
    android:layout_alignParentLeft="true"
    android:paddingBottom="20dp"
    android:text="@string/Validate"
    android:textColor="#ff0000"
    android:textSize="12sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/signature"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/btnWriteSDFFile"
    android:layout_alignParentLeft="true"
    android:layout_marginBottom="19dp"
    android:text="Click here to sign and finish" />

</LinearLayout>

</ScrollView>

```

## AthletesInsurInfoPart4.java

```

package com.mattrankin.sportdoc;
import java.io.*;
import android.annotation.SuppressLint;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Environment;
import android.view.*;
import android.view.View.OnClickListener;
import android.widget.*;

public class AthletesInsurInfoPart4 extends Activity {

    public static final int SIGNATURE_ACTIVITY = 1;
    String CurrentDocDirectory =
Environment.getExternalStorageDirectory().getPath() +
"/SportDoc/AthletesInsurInfo(Part4).txt"; // Initialize current document
directory

    // GUI controls
    private EditText txtFullName, dateofbirth, InsuredName, InsuredSSN,
InsuredHomeTelephone, InsuredWorkTelephone, InsuredCellPhoneNumber,
InsuranceCompanyName, InsuranceCompanyGroupNumber,

```

```

InsuranceCompanyPolicyNumber, MailingAddressForClaims, ClaimCityStateZip,
InsuranceCompanyTelephoneNumber, InsuranceCompanyFaxNumber;
    private Button btnWriteSDFFile;

    @SuppressWarnings("SdCardPath")
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.athletes_insur_info_part4);

        // setup text fields and hints for filling out text fields
        txtFullName = (EditText) findViewById(R.id.txtFullName);
        txtFullName.setHint("Enter last name, first name, and MI...");
        dateofbirth = (EditText) findViewById(R.id.dateofbirth);
        dateofbirth.setHint("Enter athletes date of birth...");
        InsuredName = (EditText) findViewById(R.id.InsuredName);
        InsuredName.setHint("Enter Insured Name...");
        InsuredSSN = (EditText) findViewById(R.id.InsuredSSN);
        InsuredSSN.setHint("Enter Insured's SSN...");
        InsuredHomeTelephone = (EditText)
findViewById(R.id.InsuredHomeTelephone);
        InsuredHomeTelephone.setHint("Enter Insured's Home Telephone
Number...");
        InsuredWorkTelephone = (EditText)
findViewById(R.id.InsuredWorkTelephone);
        InsuredWorkTelephone.setHint("Enter Insured's Work Telephone
Number...");
        InsuredCellPhoneNumber = (EditText)
findViewById(R.id.InsuredCellPhoneNumber);
        InsuredCellPhoneNumber.setHint("Enter Insured's Cell Phone
Number...");
        InsuranceCompanyName = (EditText)
findViewById(R.id.InsuranceCompanyName);
        InsuranceCompanyName.setHint("Enter insurance company name...");
        InsuranceCompanyGroupNumber = (EditText) findViewById(R.id.
InsuranceCompanyGroupNumber);
        InsuranceCompanyGroupNumber.setHint("Enter insurance company group
#...");
        InsuranceCompanyPolicyNumber = (EditText)
findViewById(R.id.InsuranceCompanyPolicyNumber);
        InsuranceCompanyPolicyNumber.setHint("Enter insurance company policy
#...");
        MailingAddressForClaims = (EditText)
findViewById(R.id.MailingAddressForClaims);
        MailingAddressForClaims.setHint("Enter address for claims...");
        ClaimCityStateZip = (EditText) findViewById(R.id.ClaimCityStateZip);
        ClaimCityStateZip.setHint("Enter address City, State and
Zipcode...");
        InsuranceCompanyTelephoneNumber = (EditText)
findViewById(R.id.InsuranceCompanyTelephoneNumber);
        InsuranceCompanyTelephoneNumber.setHint("Enter insurance company
telephone number...");
        InsuranceCompanyFaxNumber = (EditText)
findViewById(R.id.InsuranceCompanyFaxNumber);
        InsuranceCompanyFaxNumber.setHint("Enter insurance company fax
number...");

```

```

btnWriteSDFFile = (Button) findViewById(R.id.signature);
btnWriteSDFFile.setOnClickListener(new OnClickListener() {

    @SuppressWarnings("SdCardPath")
    public void onClick(View v) {

        // bind GUI elements with local controls
        Spinner feedbackSpinner1 = (Spinner)
findViewById(R.id.Spinner1);
        String feedbackType1 =
feedbackSpinner1.getSelectedItem().toString();
        Spinner feedbackSpinner2 = (Spinner)
findViewById(R.id.Spinner2);
        String feedbackType2 =
feedbackSpinner2.getSelectedItem().toString();
        Spinner YearInSchoolSpinner = (Spinner)
findViewById(R.id.YearInSchoolSpinner);
        String feedbackYearInSchoolSpinner =
YearInSchoolSpinner.getSelectedItem().toString();
        CheckBox responseCheckbox1 = (CheckBox)
findViewById(R.id.ChecKBox1);
        boolean CheckBoxResponse1 = responseCheckbox1.isChecked();
        String InsuredFatherCheckBox1 =
String.valueOf(CheckBoxResponse1);
        CheckBox responseCheckbox2 = (CheckBox)
findViewById(R.id.ChecKBox2);
        boolean CheckBoxResponse2 = responseCheckbox2.isChecked();
        String InsuredMotherCheckBox2 =
String.valueOf(CheckBoxResponse2);
        CheckBox responseCheckbox3 = (CheckBox)
findViewById(R.id.ChecKBox3);
        boolean CheckBoxResponse3 = responseCheckbox3.isChecked();
        String InsuredGuardianCheckBox3 =
String.valueOf(CheckBoxResponse3);
        CheckBox responseCheckbox4 = (CheckBox)
findViewById(R.id.ChecKBox4);
        boolean CheckBoxResponse4 = responseCheckbox4.isChecked();
        String InsuredSpouseCheckBox4 =
String.valueOf(CheckBoxResponse4);
        CheckBox responseCheckbox5 = (CheckBox)
findViewById(R.id.ChecKBox5);
        boolean CheckBoxResponse5 = responseCheckbox5.isChecked();
        String InsuredSelfCheckBox5 =
String.valueOf(CheckBoxResponse5);
        CheckBox responseCheckbox6 = (CheckBox)
findViewById(R.id.ChecKBox6);
        boolean CheckBoxResponse6 = responseCheckbox6.isChecked();
        String ARE_YOU_COVERED_UNDER_THE_ABOVE_POLICY =
String.valueOf(CheckBoxResponse6);
        CheckBox responseCheckbox7 = (CheckBox)
findViewById(R.id.ChecKBox7);
        boolean CheckBoxResponse7 = responseCheckbox7.isChecked();
        String A_second_opinion_for_surgery =
String.valueOf(CheckBoxResponse7);
        CheckBox responseCheckbox8 = (CheckBox)
findViewById(R.id.ChecKBox8);
        boolean CheckBoxResponse8 = responseCheckbox8.isChecked();

```

```

        String Pre_authorization_for_services =
String.valueOf(CheckBoxResponse8);
        CheckBox responseCheckbox9 = (CheckBox)
findViewById(R.id.ChecKBox9);
        boolean CheckBoxResponse9 = responseCheckbox9.isChecked();
        String Is_your_primary_insurance_an_HMO =
String.valueOf(CheckBoxResponse9);
        CheckBox responseCheckbox10 = (CheckBox)
findViewById(R.id.ChecKBox10);
        boolean CheckBoxResponse10 = responseCheckbox10.isChecked();
        String Is_your_primary_insurance_an_PPO =
String.valueOf(CheckBoxResponse10);

        // write on SD card file data in the text box
        try {

                File testDir= new
File(Environment.getExternalStorageDirectory().getPath() +
"/sdcard/SportDoc/");
                testDir.mkdirs(); // Have the object build the
directory structure SD/SportDoc/
                CaptureSignature.PicBindDir = "AthletesInsurInfoPart4";
// Set static string PicBindDir to Document name for signature validation.
                CaptureSignature.DocDir = CurrentDocDirectory; // Set
static DocDir used in CaptureSignature class to this directory.
(CaptureSignatures uses DocDir to attach txt to email app)
                File myFile = new File(CurrentDocDirectory);
                myFile.createNewFile();

                //destinationFile.createNewFile();
                FileOutputStream fOut = new FileOutputStream(myFile); //
Write info to file
                OutputStreamWriter myOutWriter = new
OutputStreamWriter(fOut);
                myOutWriter.append("Athletes Fulltname:");
                myOutWriter.append(txtFullName.getText());
                myOutWriter.append("\r\n");
                myOutWriter.append("Athletes date of birth:");
                myOutWriter.append(dateofbirth.getText());
                myOutWriter.append("\r\n");
                myOutWriter.append("sport1:");
                myOutWriter.append(feedbackType1);
                myOutWriter.append("\r\n");
                myOutWriter.append("sport2:");
                myOutWriter.append(feedbackType2);
                myOutWriter.append("\r\n");
                myOutWriter.append("Athletes year in school:");
                myOutWriter.append(feedbackYearInSchoolSpinner);
                myOutWriter.append("\r\n");
                myOutWriter.append("Insured Father:");
                myOutWriter.append(InsuredFatherCheckBox1);
                myOutWriter.append("\r\n");
                myOutWriter.append("Insured Mother:");
                myOutWriter.append(InsuredMotherCheckBox2);
                myOutWriter.append("\r\n");
                myOutWriter.append("Insured guardian:");
                myOutWriter.append(InsuredGuardianCheckBox3);

```

```

myOutWriter.append("\r\n");
myOutWriter.append("Insured spouse:");
myOutWriter.append(InsuredSpouseCheckBox4);
myOutWriter.append("\r\n");
myOutWriter.append("Insured self:");
myOutWriter.append(InsuredSelfCheckBox5);
myOutWriter.append("\r\n");
myOutWriter.append("Name of insured person:");
myOutWriter.append(InsuredName.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Insured persons SSN:");
myOutWriter.append(InsuredSSN.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Insured persons home phone number:");
myOutWriter.append(InsuredHomeTelephone.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Insured persons work phone number:");
myOutWriter.append(InsuredWorkTelephone.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Insured persons cell phone number:");
myOutWriter.append(InsuredCellPhoneNumber.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Insurance company name:");
myOutWriter.append(InsuranceCompanyName.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Insurance company group number:");

myOutWriter.append(InsuranceCompanyGroupNumber.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Insurance company policy number:");

myOutWriter.append(InsuranceCompanyPolicyNumber.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Mailing address for claims:");
myOutWriter.append(MailingAddressForClaims.getText());
myOutWriter.append("\r\n");
myOutWriter.append(ClaimCityStateZip.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Insurance company telephone
number:");

myOutWriter.append(InsuranceCompanyTelephoneNumber.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Insurance company fax number:");
myOutWriter.append(InsuranceCompanyFaxNumber.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Is the athlete covered under the
above policy response:");

myOutWriter.append(ARE_YOU_COVERED_UNDER_THE_ABOVE_POLICY);
myOutWriter.append("\r\n");
myOutWriter.append("Is a second opinion needed for
surgery response:");
myOutWriter.append(A_second_opinion_for_surgery);
myOutWriter.append("\r\n");
myOutWriter.append("Is a pre-authorization for services
needed response:");

```

```

        myOutWriter.append(Pre_authorization_for_services);
        myOutWriter.append("\r\n");
        myOutWriter.append("Is your primary insurance an HMO
response:");

        myOutWriter.append(Is_your_primary_insurance_an_HMO);
        myOutWriter.append("\r\n");
        myOutWriter.append("Is your primary insurance an PPO
response:");

        myOutWriter.append(Is_your_primary_insurance_an_PPO);
        myOutWriter.append("\r\n");
        myOutWriter.close();
        fOut.close();
    } catch (Exception e) {
        Toast.makeText(getApplicationContext(), e.getMessage(),
Toast.LENGTH_SHORT).show();
    }
    Intent intent = new Intent(AthletesInsurInfoPart4.this,
CaptureSignature.class);
    startActivityForResult(intent, SIGNATURE_ACTIVITY);
    } // onClick
}); // btnWriteSDFFile
} // onCreate
} // AndSDcard

```

### athletes\_insur\_info\_part4.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#bbeeff" >

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:background="#FFFFFF1"
        android:isScrollContainer="true"
        android:orientation="vertical" >

        <TextView
            android:id="@+id/textView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Athlete's Insurance Information (Part 4)"
            android:textAppearance="?android:attr/textAppearanceMedium"
            android:textColor="@color/textcolor" />

        <TextView
            android:id="@+id/textView1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Enter full name:"
            android:textColor="@color/textcolor" />

```

```

<EditText
    android:id="@+id/txtFullName"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter date of birth:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/dateofbirth"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select Sports:"
    android:textColor="@color/textcolor" />

<Spinner
    android:id="@+id/Spinner1"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/txtLastName"
    android:layout_marginTop="5dp"
    android:entries="@array/sports" />

<Spinner
    android:id="@+id/Spinner2"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/txtLastName"
    android:layout_marginTop="5dp"
    android:entries="@array/sports" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select athlete's year in school:"
    android:textColor="@color/textcolor" />

<Spinner

```

```

        android:id="@+id/YearInSchoolSpinner"
        android:layout_width="180dp"
        android:layout_height="40dp"
        android:layout_alignParentLeft="true"
        android:entries="@array/YearInSchool" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Please include pictures for front and back of your
insurance card."
    android:textColor="@color/textcolor"
    android:textStyle="bold" />

<Button
    android:id="@+id/InsuranceCardPicture"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/btnWriteSDFFile"
    android:layout_alignParentLeft="true"
    android:layout_marginBottom="19dp"
    android:text="Click here to take pictures of insurance card front
and back!"
    android:textColor="#ff0000" />

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select who is insured:"
    android:textColor="@color/textcolor" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:id="@+id/textView6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Father"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"

```



```

    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:id="@+id/textView7"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Mother"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckBox2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:id="@+id/textView8"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Guardian"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckBox3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:id="@+id/textView9"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Spouse"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckBox4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout

```

```

xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:gravity="left"
android:orientation="horizontal" >

<TextView
    android:id="@+id/textView10"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Self"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/CheckBox5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
</LinearLayout>

<TextView
    android:id="@+id/textView11"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter insured's name:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InsuredName"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView12"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Insured SSN:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InsuredSSN"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView13"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Insured Home telephone:"
    android:textColor="@color/textcolor" />

```

```

<EditText
    android:id="@+id/InsuredHomeTelephone"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/LinearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView14"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Insured Work telephone:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InsuredWorkTelephone"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/LinearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView15"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Insured Cell Phone Number:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InsuredCellPhoneNumber"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/LinearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<View
    android:layout_width="fill_parent"
    android:layout_height="10dip"
    android:background="#000000" />

<TextView
    android:id="@+id/textView16"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Name of Insurance Company"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InsuranceCompanyName"
    android:layout_width="320dp"

```

```

        android:layout_height="40dp"
        android:layout_alignParentLeft="true"
        android:layout_below="@+id/linearLayout1"
        android:ems="10"
        android:textSize="18sp" />

<TextView
    android:id="@+id/textView17"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Group #:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InsuranceCompanyGroupNumber"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView18"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Policy # :"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InsuranceCompanyPolicyNumber"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView19"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter mailing address for claims:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/MailingAddressForClaims"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView20"
    android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:text="Enter City/State/Zip:"
        android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/ClaimCityStateZip"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView21"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Your Insurance Companies Telephone #:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InsuranceCompanyTelephoneNumber"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView22"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Your Insurance Companies Fax #:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InsuranceCompanyFaxNumber"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:id="@+id/textView10"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="ARE YOU COVERED UNDER THE ABOVE POLICY?"

```

```

        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<TextView
    android:id="@+id/textView10"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="DOES YOUR INSURANCE REQUIRE:"
    android:textColor="@color/textcolor" />

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:id="@+id/textView10"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="A second opinion for surgery?"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox7"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

    <TextView
        android:id="@+id/textView10"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Pre-authorization for services?"
        android:textColor="@color/textcolor" />

    <CheckBox
        android:id="@+id/CheckKBox8"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>

```

```

<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="fill_parent"
  android:layout_height="wrap_content"
  android:gravity="left"
  android:orientation="horizontal" >

  <TextView
    android:id="@+id/textView10"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Is your primary insurance an HMO?"
    android:textColor="@color/textcolor" />

  <CheckBox
    android:id="@+id/CheckBox9"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
</LinearLayout>

<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="fill_parent"
  android:layout_height="wrap_content"
  android:gravity="left"
  android:orientation="horizontal" >

  <TextView
    android:id="@+id/textView10"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Is your primary insurance an PPO?"
    android:textColor="@color/textcolor" />

  <CheckBox
    android:id="@+id/CheckBox10"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
</LinearLayout>

<TextView
  android:id="@+id/textView10"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
  android:text="This form is for information purposes only. This
is not authorization to bill or use this insurance for injury. Information
contained within this form is strictly for the use of Northwest Nazarene
University and one of its representatives. "
  android:textSize="12sp"
  android:textColor="@color/textcolor" />

<TextView
  android:id="@+id/textView10"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
  android:text="Please submit a copy of your current insurance card
with the above information (front and back)."
```

```

        android:textColor="@color/textcolor"
        android:textSize="16sp"
        android:textStyle="bold" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_above="@+id/signature"
    android:layout_alignParentLeft="true"
    android:paddingBottom="20dp"
    android:text="@string/Validate"
    android:textColor="#ff0000"
    android:textSize="12sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/signature"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/btnWriteSDFFile"
    android:layout_alignParentLeft="true"
    android:layout_marginBottom="19dp"
    android:text="Click here to sign and finish" />
</LinearLayout>
</ScrollView>

```

## ProvMedTreatPart5.java

```

package com.mattrankin.sportdoc;
import java.io.*;
import android.annotation.SuppressLint;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Environment;
import android.view.*;
import android.view.View.OnClickListener;
import android.widget.*;

public class ProvMedTreatPart5 extends Activity {

    public static final int SIGNATURE_ACTIVITY = 1;
    String CurrentDocDirectory =
Environment.getExternalStorageDirectory().getPath() +
"/SportDoc/ProcMedTreat(Part5).txt"; // Initialize current document directory

    // GUI controls
    private EditText txtparagraphName;
    private EditText dateofbirth;
    private EditText txtSSN;
    private EditText ParentGuardianPrintedName;
    private Button btnWriteSDFFile;

    @SuppressLint("SdCardPath")

```



```

@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.prov_med_treat_part5);

    // setup text fields and hints for filling out text fields
    txtparagraphName = (EditText) findViewById(R.id.txtparagraphName);
    txtparagraphName.setHint("Enter full name...");
    dateofbirth = (EditText) findViewById(R.id.dateofbirth);
    dateofbirth.setHint("Enter athletes date of birth...");
    txtSSN = (EditText) findViewById(R.id.txtSSN);
    txtSSN.setHint("Enter Social Security Number...");
    ParentGuardianPrintedName = (EditText)
findViewById(R.id.ParentGuardianPrintedName);
    ParentGuardianPrintedName.setHint("Enter Parent/Guardian full
name...");

    btnWriteSDFFile = (Button) findViewById(R.id.signature);
    btnWriteSDFFile.setOnClickListener(new OnClickListener(){

        @SuppressWarnings("SdCardPath")
        public void onClick(View v) {

            // bind GUI elements with local controls
            Spinner feedbackSpinner1 = (Spinner)
findViewById(R.id.Spinner1);
            String feedbackType1 =
feedbackSpinner1.getSelectedItem().toString();
            Spinner feedbackSpinner2 = (Spinner)
findViewById(R.id.Spinner2);
            String feedbackType2 =
feedbackSpinner2.getSelectedItem().toString();
            Spinner AgeSpinner = (Spinner) findViewById(R.id.AgeSpinner);
            String feedbackAgeSpinner =
AgeSpinner.getSelectedItem().toString();

            // write on SD card file data in the text box
            try {

                File testDir= new
File(Environment.getExternalStorageDirectory().getPath() +
"/sdcard/SportDoc/");
                testDir.mkdirs(); // Have the object build the
directory structure SD/SportDoc/
                CaptureSignature.PicBindDir = "ProcMedTreatPart5"; //
Set static string PicBindDir to Document name for signature validation.
                CaptureSignature.DocDir = CurrentDocDirectory; // Set
static DocDir used in CaptureSignature class to this directory.
                (CaptureSignatures uses DocDir to attach txt to email app)
                File myFile = new File(CurrentDocDirectory);
                myFile.createNewFile();

                //destinationFile.createNewFile();
                FileOutputStream fOut = new FileOutputStream(myFile); //
Write info to file
                OutputStreamWriter myOutWriter = new
OutputStreamWriter(fOut);

```

```

        myOutWriter.append("Name filled in permission
paragraph:");
        myOutWriter.append(txtparagraphName.getText());
        myOutWriter.append("\r\n");
        myOutWriter.append("sport1:");
        myOutWriter.append(feedbackType1);
        myOutWriter.append("\r\n");
        myOutWriter.append("sport2:");
        myOutWriter.append(feedbackType2);
        myOutWriter.append("\r\n");
        myOutWriter.append("Athletes Age:");
        myOutWriter.append(feedbackAgeSpinner);
        myOutWriter.append("\r\n");
        myOutWriter.append("Athletes Date of Birth:");
        myOutWriter.append(dateofbirth.getText());
        myOutWriter.append("\r\n");
        myOutWriter.append("Athletes SSN:");
        myOutWriter.append(txtSSN.getText());
        myOutWriter.append("\r\n");
        myOutWriter.append("Parent or guardians printed name:");
        myOutWriter.append(ParentGuardianPrintedName.getText());
        myOutWriter.append("\r\n");
        myOutWriter.close();
        fOut.close();
    } catch (Exception e) {
        Toast.makeText(getApplicationContext(), e.getMessage(),
Toast.LENGTH_SHORT).show();
    }
    Intent intent = new Intent( ProvMedTreatPart5.this,
CaptureSignature.class);
    startActivityForResult(intent, SIGNATURE_ACTIVITY);
    } // onClick
}); // btnWriteSDFile
} // onCreate
} // AndSDcard

```

### prov\_med\_treat\_part5.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#bbeeff" >

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:background="#FFFFFF1"
        android:isScrollContainer="true"
        android:orientation="vertical" >

        <TextView
            android:id="@+id/textView1"
            android:layout_width="wrap_content"

```

```

    android:layout_height="wrap_content"
    android:text="Permission to Provide Medical Treatment (Part 5)"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="@color/textcolor" />

```

```

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:gravity="left"
    android:orientation="horizontal" >

```

```

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="I,"
        android:textColor="@color/textcolor" />

```

```

    <EditText
        android:id="@+id/txtparagraphName"
        android:layout_width="320dp"
        android:layout_height="40dp"
        android:layout_alignParentLeft="true"
        android:ems="10"
        android:textSize="18sp" />

```

```

</LinearLayout>

```

```

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text=", hereby give my permission to undergo medical
treatment for any injury or illness that may be sustained or acquired while
engaged in intercollegiate athletics at Northwest Nazarene University (NNU)
or at any College or University in which NNU will compete against by Medical
Personnel representing said College or University, clinic, Hospital/Emergency
department. I understand that the medical personnel will perform only those
procedures that are within their training, credentials, and scope of
professional practice to prevent, care for, and rehabilitate athletic
injuries. In the event that more serious medical procedures are required
(i.e. surgery) I understand that every attempt will be made to contact my
parent(s) and or guardian(s) for consent (if minor). I understand that if I
suffer a potentially life threatening injury or illness, and in the event
that my parent(s)/ guardian(s) cannot be reached within a reasonable period
of time, that I authorize any duly licensed medical practitioner to perform
such procedures as may be medically necessary to alleviate the problem."
    android:textColor="@color/textcolor" />

```

```

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="" />

```

```

<TextView
    android:id="@+id/textView4"

```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="I have had time to ask questions regarding this
release and all of my questions have been answered to my satisfaction."
        android:textColor="@color/textcolor" />
```

```
<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="" />
```

```
<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Having understood the above agreement, I freely
sign this Permission to Provide Medical Treatment Agreement."
    android:textColor="@color/textcolor" />
```

```
<TextView
    android:id="@+id/textView7"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="" />
```

```
<TextView
    android:id="@+id/textView8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Sports:"
    android:textColor="@color/textcolor" />
```

```
<Spinner
    android:id="@+id/Spinner1"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:entries="@array/sports" />
```

```
<Spinner
    android:id="@+id/Spinner2"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:entries="@array/sports" />
```

```
<TextView
    android:id="@+id/textView8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Athlete's Age:"
    android:textColor="@color/textcolor" />
```

```
<Spinner
    android:id="@+id/AgeSpinner"
    android:layout_width="180dp"
```

```

        android:layout_height="40dp"
        android:layout_alignParentLeft="true"
        android:entries="@array/age" />

<TextView
    android:id="@+id/textView8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Athlete's Date of Birth:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/dateofbirth"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView9"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Social Security Number:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/txtSSN"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:ems="10"
    android:textSize="18sp" />

<View
    android:layout_width="fill_parent"
    android:layout_height="1dip"
    android:background="#000000" />

<TextView
    android:id="@+id/textView9"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="***In the event that the athlete is under the age
of 18, the parent and or guardian must sign this form below."
    android:textColor="@color/textcolor" />

<TextView
    android:id="@+id/textView9"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Parent/Guardian printed name:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/ParentGuardianPrintedName"
    android:layout_width="320dp"

```

```

        android:layout_height="40dp"
        android:layout_alignParentLeft="true"
        android:ems="10"
        android:textSize="18sp" />

<TextView
    android:id="@+id/textView9"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Also have parent/guardian sign after clicking the
&apos;Click Here to sign and finish&apos; button bellow."
    android:textColor="@color/textcolor" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_above="@+id/signature"
    android:layout_alignParentLeft="true"
    android:text="@string/Validate"
    android:textColor="#ff0000"
    android:textSize="12sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/signature"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/btnWriteSDFFile"
    android:layout_alignParentLeft="true"
    android:text="Click here to sign and finish" />
</LinearLayout>

</ScrollView>

```

## InterimhealthformPart6.java

```

package com.matrankin.sportdoc;
import java.io.*;
import java.util.Calendar;
import android.annotation.SuppressLint;
import android.app.Activity;
import android.app.AlertDialog;
import android.app.DatePickerDialog;
import android.app.Dialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.os.Environment;
import android.view.*;
import android.view.View.OnClickListener;
import android.widget.*;

public class InterimhealthformPart6 extends Activity {

    public static final int SIGNATURE_ACTIVITY = 1;

```

```

    private String CurrentDocDirectory =
Environment.getExternalStorageDirectory().getPath() +
"/SportDoc/InterimHealthForm(Part6).txt"; // Initialize current document
directory

    // Private Variables
    private TextView pDisplayDate;
    private Button pPickDate;
    private int Year;
    private int Month;
    private int Day;
    private Button btnWriteSDFFile;
    private EditText txtFullName, ProvidersName, ProvidersPhoneNumber,
ProvidersAddress, ProvidersCity, ProvidersState, ProvidersDatesTreated,
InitialForAccuracy, InitialForPermmision;
    private EditText CellPhoneNumber, OtherPhoneNumber,
SuddenDeathBefore50Response, SignificantInjuryResponse,
HospitalizedForAnyReasonResponse, OtherInjuryResponse,
PassedOutDuringAfterExerciseResponse, SurgeryOfAnyKindResponse,
RecurrentBackPainResponse, HighBloodPressureHighCholesterolResponse,
SignificantHeadInjuryKnockedUnconsciousResponse, SeizuresOfAnyKindResponse,
ShortnessOfBreatWithExerciseResponse,
ChronicRecurrentCoughWithExerciseResponse,
LossDecreasedFunctionOfAnyganResponse,
AllergicReactionToMedicationStringingInsectFoodResponse,
IrregularSkippingHeartbeatsResponse, MenstrualProblemsIrregularitiesResponse,
AsthmaSeasonalAllergiesRequiringMedicationResponse,
UnusualOnsetOfHeadachesResponse,
HaveYouBeenTreatedForAnEatingDisorderResponse,
MonoOtherChronicIllnessesLastingLongerThanAWeekResponse,
BloodDisderEtcResponse, TreatedEncouragedFTreatmentFSubstanceAbuseResponse,
WouldYouLikeInfoAboutSpeakToSomeoneAboutIssuesResponse,
StartedNewMedicationsResponse, AnyOtherProblemsIllnessesResponse;
    private EditText input, input2, input3, input4, input5, input6, input7,
input8, input9, input10, input11, input12, input13, input14, input15,
input16, input17, input18, input19, input20, input21, input22, input23,
input24, input25;
    private String inputrestore, input2restore, input3restore, input4restore,
input5restore, input6restore, input7restore, input8restore, input9restore,
input10restore, input11restore, input12restore, input13restore,
input14restore, input15restore, input16restore, input17restore,
input18restore, input19restore, input20restore, input21restore,
input22restore, input23restore, input24restore, input25restore;
    private String SportFeedback1, SportFeedback2;

    /* Static members of the class */
    static final int DATE_DIALOG_ID = 0; // This integer will uniquely define
the dialog to be used for displaying the date picker.

    /* Callback received when the user "picks" a date in the dialog */
    private DatePickerDialog.OnDateSetListener pDateSetListener = new
DatePickerDialog.OnDateSetListener() {

        public void onDateSet(DatePicker view, int year, int monthOfYear, int
dayOfMonth) {
            Year = year;
            Month = monthOfYear;

```

```

        Day = dayOfMonth;
        updateDisplay();
        displayToast();
    }
};

/* Updates the date in the TextView */
private void updateDisplay() {
    pDisplayDate.setText(
        new StringBuilder().append(Month +
1).append("/").append(Day).append("/").append(Year).append(" ");    // Month
is 0 based so add 1
    }

/* Displays a notification when the date is updated */
private void displayToast() {
    Toast.makeText(this, new StringBuilder().append("Date chosen is
").append(pDisplayDate.getText()), Toast.LENGTH_SHORT).show();
}

@SuppressLint("SdCardPath")
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.iterim_health_form_part6);

    /* Capture our View elements */
    pDisplayDate = (TextView) findViewById(R.id.displayDate);
    pPickDate = (Button) findViewById(R.id.pickDate);

    /* Listener for click event of the date button */
    pPickDate.setOnClickListener(new View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View v) {
            showDialog(DATE_DIALOG_ID);
        }
    });

    /* Get the current date */
    final Calendar cal = Calendar.getInstance();
    Year = cal.get(Calendar.YEAR);
    Month = cal.get(Calendar.MONTH);
    Day = cal.get(Calendar.DAY_OF_MONTH);

    /* Display the current date in the TextView */
    updateDisplay();

    // setup text fields and hints for filling out text fields
    txtFullName = (EditText) findViewById(R.id.txtFullName);
    txtFullName.setHint("Enter last name, first name, and MI...");
    CellPhoneNumber = (EditText) findViewById(R.id.CellPhoneNumber);
    CellPhoneNumber.setHint("Enter Your Cell Phone Number...");
    OtherPhoneNumber = (EditText) findViewById(R.id.OtherPhoneNumber);
    OtherPhoneNumber.setHint("Enter Other Phone Number...");
    ProvidersName = (EditText) findViewById(R.id.ProvidersName);
    ProvidersName.setHint("Enter providers name...");

```



```

        ProvidersPhoneNumber = (EditText)
findViewById(R.id.ProvidersPhoneNumber);
        ProvidersPhoneNumber.setHint("Enter providers phone number...");
        ProvidersAddress = (EditText) findViewById(R.id.ProvidersAddress);
        ProvidersAddress.setHint("Enter providers address...");
        ProvidersCity = (EditText) findViewById(R.id.ProvidersCity);
        ProvidersCity.setHint("Enter providers city...");
        ProvidersState = (EditText) findViewById(R.id.ProvidersState);
        ProvidersState.setHint("Enter providers state...");
        ProvidersDatesTreated = (EditText)
findViewById(R.id.ProvidersDatesTreated);
        ProvidersDatesTreated.setHint("Enter date(s) treated...");
        InitialForAccuracy = (EditText)
findViewById(R.id.InitialForAccuracy);
        InitialForAccuracy.setHint("Enter Intials for above...");
        InitialForPermmision = (EditText)
findViewById(R.id.InitialForPermmision);
        InitialForPermmision.setHint("Enter Intials for above...");

        CheckBox SuddenDeathBefore50 = (CheckBox)
findViewById(R.id.SuddenDeathBefore50);
        SuddenDeathBefore50.setOnClickListener(new View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View view) {
                AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
                alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
                input = new EditText(InterimhealthformPart6.this); // Set an
EditText view to get user input
                input.setText(inputrestore);
                alertDialog.setView(input);
                alertDialog.setButton("Save & Continue", new
DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        SuddenDeathBefore50Response = input;
                        inputrestore = input.getText().toString();
                    }
                });
                alertDialog.show();
            }
        });

        CheckBox SignificantInjury = (CheckBox)
findViewById(R.id.SignificantInjury);
        SignificantInjury.setOnClickListener(new View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View view) {
                AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
                alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
                input2= new EditText(InterimhealthformPart6.this); // Set an
EditText view to get user input
                input2.setText(input2restore);
                alertDialog.setView(input2);
            }
        });

```

```

        alertDialog.setButton("Save & Continue", new
DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog,int which) {
                SignificantInjuryResponse = input2;
                input2restore = input2.getText().toString();
            }
        });
        alertDialog.show();
    }
});

    CheckBox HospitalizedForAnyReason = (CheckBox)
findViewById(R.id.HospitalizedForAnyReason);
    HospitalizedForAnyReason.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {
            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input3 = new EditText(InterimhealthformPart6.this); // Set an
EditText view to get user input
            input3.setText(input3restore);
            alertDialog.setView(input3);
            HospitalizedForAnyReasonResponse = input3;
            alertDialog.setButton("Save & Continue", new
DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    HospitalizedForAnyReasonResponse = input3;
                    input3restore = input3.getText().toString();
                }
            });
            alertDialog.show();
        }
    });

    CheckBox OtherInjury = (CheckBox) findViewById(R.id.OtherInjury);
    OtherInjury.setOnClickListener(new View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {
            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input4 = new EditText(InterimhealthformPart6.this); // Set an
EditText view to get user input4
            input4.setText(input4restore);
            alertDialog.setView(input4);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        OtherInjuryResponse = input4;
                        input4restore = input4.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

```

```

    }
});

    CheckBox PassedOutDuringAfterExercise = (CheckBox)
findViewById(R.id.PassedOutDuringAfterExercise);
    PassedOutDuringAfterExercise.setOnClickListener(new
View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
        input5 = new EditText(InterimhealthformPart6.this); // Set an
EditText view to get user input5
        input5.setText(input5restore);
        alertDialog.setView(input5);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    PassedOutDuringAfterExerciseResponse = input5;
                    input5restore = input5.getText().toString();
                }
            });
        alertDialog.show();
    }
});

    CheckBox SurgeryOfAnyKind = (CheckBox)
findViewById(R.id.SurgeryOfAnyKind);
    SurgeryOfAnyKind.setOnClickListener(new View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {
        AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
        input6 = new EditText(InterimhealthformPart6.this); // Set an
EditText view to get user input6
        input6.setText(input6restore);
        alertDialog.setView(input6);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    SurgeryOfAnyKindResponse = input6;
                    input6restore = input6.getText().toString();
                }
            });
        alertDialog.show();
    }
});

    CheckBox RecurrentBackPain = (CheckBox)
findViewById(R.id.RecurrentBackPain);
    RecurrentBackPain.setOnClickListener(new View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {

```

```

        AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
        alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
        input7 = new EditText(InterimhealthformPart6.this); // Set an
EditText view to get user input7
        input7.setText(input7restore);
        alertDialog.setView(input7);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    RecurrentBackPainResponse = input7;
                    input7restore = input7.getText().toString();
                }
            });
        alertDialog.show();
    }
});

    CheckBox HighBloodPressureHighCholesterol = (CheckBox)
findViewById(R.id.HighBloodPressureHighCholesterol);
    HighBloodPressureHighCholesterol.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {
            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input8 = new EditText(InterimhealthformPart6.this); // Set an
EditText view to get user input8
            input8.setText(input8restore);
            alertDialog.setView(input8);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        HighBloodPressureHighCholesterolResponse = input8;
                        input8restore = input8.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

    CheckBox SignificantHeadInjuryKnockedUnconscious = (CheckBox)
findViewById(R.id.SignificantHeadInjuryKnockedUnconscious);
    SignificantHeadInjuryKnockedUnconscious.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {
            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input9 = new EditText(InterimhealthformPart6.this); // Set an
EditText view to get user input9

```

```

        input9.setText(input9restore);
        alertDialog.setView(input9);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    SignificantHeadInjuryKnockedUnconsciousResponse =
input9;
                    input9restore = input9.getText().toString();
                }
            });
        alertDialog.show();
    }
});

    CheckBox SeizuresOfAnyKind = (CheckBox)
findViewById(R.id.SeizuresOfAnyKind);
    SeizuresOfAnyKind.setOnClickListener(new View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {

            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input10 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input10
            input10.setText(input10restore);
            alertDialog.setView(input10);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        SeizuresOfAnyKindResponse = input10;
                        input10restore = input10.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

    CheckBox ShortnessOfBreatWithExercise = (CheckBox)
findViewById(R.id.ShortnessOfBreathWithExercise);
    ShortnessOfBreatWithExercise.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {

            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input11 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input11
            input11.setText(input11restore);
            alertDialog.setView(input11);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {

```

```

        ShortnessOfBreatWithExerciseResponse = input11;
        input11restore = input11.getText().toString();
    }
    });
    alertDialog.show();
}
});

    CheckBox ChronicRecurrentCoughWithExercise = (CheckBox)
findViewById(R.id.ChronicRecurrentCoughWithExercise);
    ChronicRecurrentCoughWithExercise.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {

            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input12 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input12
            input12.setText(input12restore);
            alertDialog.setView(input12);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        ChronicRecurrentCoughWithExerciseResponse = input12;
                        input12restore = input12.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

    CheckBox LossDecreasedFunctionOfAnygan = (CheckBox)
findViewById(R.id.LossDecreasedFunctionOfAnygan);
    LossDecreasedFunctionOfAnygan.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {

            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input13 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input13
            input13.setText(input13restore);
            alertDialog.setView(input13);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        LossDecreasedFunctionOfAnyganResponse = input13;
                        input13restore = input13.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

```

```

    }
});

    CheckBox AllergicReactionToMedicationStringingInsectFood = (CheckBox)
findViewById(R.id.AllergicReactionToMedicationStringingInsectFood);

AllergicReactionToMedicationStringingInsectFood.setOnClickListener(new
View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {

        AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
        alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
        input14 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input14
        input14.setText(input14restore);
        alertDialog.setView(input14);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {

AllergicReactionToMedicationStringingInsectFoodResponse = input14;
                    input14restore = input14.getText().toString();
                }
            });
        alertDialog.show();
    }
});

    CheckBox IrregularSkippingHeartbeats = (CheckBox)
findViewById(R.id.IrregularSkippingHeartbeats);
    IrregularSkippingHeartbeats.setOnClickListener(new
View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {

        AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
        alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
        input15 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input15
        input15.setText(input15restore);
        alertDialog.setView(input15);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    IrregularSkippingHeartbeatsResponse = input15;
                    input15restore = input15.getText().toString();
                }
            });
        alertDialog.show();
    }
});
});

```

```

        CheckBox MenstrualProblemsIrregularities = (CheckBox)
findViewById(R.id.MenstrualProblemsIrregularities);
        MenstrualProblemsIrregularities.setOnClickListener(new
View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View view) {

                AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
                alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
                input16 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input16
                input16.setText(input16restore);
                alertDialog.setView(input16);
                alertDialog.setButton("Save & Continue",
                    new DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialog,int which) {
                            MenstrualProblemsIrregularitiesResponse = input16;
                            input16restore = input16.getText().toString();
                        }
                    });
                alertDialog.show();
            }
        });

        CheckBox AsthmaSeasonalAllergiesRequiringMedication = (CheckBox)
findViewById(R.id.AsthmaSeasonalAllergiesRequiringMedication);
        AsthmaSeasonalAllergiesRequiringMedication.setOnClickListener(new
View.OnClickListener() {
            @SuppressWarnings("deprecation")
            public void onClick(View view) {

                AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
                alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
                input17 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input17
                input17.setText(input17restore);
                alertDialog.setView(input17);
                alertDialog.setButton("Save & Continue",
                    new DialogInterface.OnClickListener() {
                        public void onClick(DialogInterface dialog,int which) {
                            AsthmaSeasonalAllergiesRequiringMedicationResponse =
input17;
                            input17restore = input17.getText().toString();
                        }
                    });
                alertDialog.show();
            }
        });

        CheckBox UnusualOnsetOfHeadaches = (CheckBox)
findViewById(R.id.UnusualOnsetOfHeadaches);
        UnusualOnsetOfHeadaches.setOnClickListener(new View.OnClickListener()
{

```



```

        @SuppressWarnings("deprecation")
        public void onClick(View view) {

            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input18 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input18
            input18.setText(input18restore);
            alertDialog.setView(input18);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        UnusualOnsetOfHeadachesResponse = input18;
                        input18restore = input18.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

    CheckBox HaveYouBeenTreatedForAnEatingDisorder = (CheckBox)
findViewById(R.id.HaveYouBeenTreatedForAnEatingDisorder);
    HaveYouBeenTreatedForAnEatingDisorder.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {

            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input19 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input19
            input19.setText(input19restore);
            alertDialog.setView(input19);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        HaveYouBeenTreatedForAnEatingDisorderResponse =
input19;
                        input19restore = input19.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

    CheckBox MonoOtherChronicIllnessesLastingLongerThanAWeek = (CheckBox)
findViewById(R.id.MonoOtherChronicIllnessesLastingLongerThanAWeek);
    MonoOtherChronicIllnessesLastingLongerThanAWeek.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {

```

```

        AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
        alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
        input20 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input20
        input20.setText(input20restore);
        alertDialog.setView(input20);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
MonoOtherChronicIllnessesLastingLongerThanAWeekResponse = input20;
                    input20restore = input20.getText().toString();
                }
            });
        alertDialog.show();
    }
});

    CheckBox BloodDisderEtc = (CheckBox)
findViewById(R.id.BloodDisderEtc);
    BloodDisderEtc.setOnClickListener(new View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {

            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input21 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input21
            input21.setText(input21restore);
            alertDialog.setView(input21);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        BloodDisderEtcResponse = input21;
                        input21restore = input21.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

    CheckBox TreatedEncouragedFTreatmentFSubstanceAbuse = (CheckBox)
findViewById(R.id.TreatedEncouragedFTreatmentFSubstanceAbuse);
    TreatedEncouragedFTreatmentFSubstanceAbuse.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {

            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");

```

```

        input22 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input22
        input22.setText(input22restore);
        alertDialog.setView(input22);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    TreatedEncouragedFTreatmentFSubstanceAbuseResponse =
input22;
                    input22restore = input22.getText().toString();
                }
            });
        alertDialog.show();
    });
}

CheckBox WouldYouLikeInfoAboutSpeakToSomeoneAboutIssues = (CheckBox)
findViewById(R.id.WouldYouLikeInfoAboutSpeakToSomeoneAboutIssues);
WouldYouLikeInfoAboutSpeakToSomeoneAboutIssues.setOnClickListener(new
View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {

        AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
        alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
        input23 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input23
        input23.setText(input23restore);
        alertDialog.setView(input23);
        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {

WouldYouLikeInfoAboutSpeakToSomeoneAboutIssuesResponse = input23;
                    input23restore = input23.getText().toString();
                }
            });
        alertDialog.show();
    }
});

CheckBox StartedNewMedications = (CheckBox)
findViewById(R.id.StartedNewMedications);
StartedNewMedications.setOnClickListener(new View.OnClickListener() {
    @SuppressWarnings("deprecation")
    public void onClick(View view) {

        AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
        alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
        input24 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input24
        input24.setText(input24restore);
        alertDialog.setView(input24);
    }
});

```

```

        alertDialog.setButton("Save & Continue",
            new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog,int which) {
                    StartedNewMedicationsResponse = input24;
                    input24restore = input24.getText().toString();
                }
            });
        alertDialog.show();
    }
});

    CheckBox AnyOtherProblemsIllnesses = (CheckBox)
findViewById(R.id.AnyOtherProblemsIllnesses);
    AnyOtherProblemsIllnesses.setOnClickListener(new
View.OnClickListener() {
        @SuppressWarnings("deprecation")
        public void onClick(View view) {

            AlertDialog alertDialog = new
AlertDialog.Builder(InterimhealthformPart6.this).create(); // Read Update
            alertDialog.setMessage("If yes, explain and state if it's
still a problem:");
            input25 = new EditText(InterimhealthformPart6.this); // Set
an EditText view to get user input25
            input25.setText(input25restore);
            alertDialog.setView(input25);
            alertDialog.setButton("Save & Continue",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface dialog,int which) {
                        AnyOtherProblemsIllnessesResponse = input25;
                        input25restore = input25.getText().toString();
                    }
                });
            alertDialog.show();
        }
    });

    btnWriteSDFile = (Button) findViewById(R.id.signature);
    btnWriteSDFile.setOnClickListener(new OnClickListener() {
        @SuppressWarnings("SdCardPath")
        public void onClick(View v) {

            // bind GUI elements with local controls
            Spinner feedbackSpinner1 = (Spinner)
findViewById(R.id.Spinner1);
            SportFeedback1 =
feedbackSpinner1.getSelectedItem().toString();
            Spinner feedbackSpinner2 = (Spinner)
findViewById(R.id.Spinner2);
            SportFeedback2 =
feedbackSpinner2.getSelectedItem().toString();

            // write on SD card file data in the text box
            try {

```

```

        File testDir = new
File(Environment.getExternalStorageDirectory().getPath() +
"/sdcard/SportDoc/");
        testDir.mkdirs(); // Have the object build the directory
structure SD/SportDoc/
        CaptureSignature.PicBindDir = "InterimHealthFormPart6";
// Set static string PicBindDir to Document name for signature validation.
        CaptureSignature.DocDir = CurrentDocDirectory; // Set
static DocDir used in CaptureSignature class to this directory.
(CaptureSignatures uses DocDir to attach txt to email app)
        File myFile = new File(CurrentDocDirectory);
        myFile.createNewFile();

        // destinationFile.createNewFile();
        FileOutputStream fOut = new FileOutputStream(myFile); //
Write info to file
        OutputStreamWriter myOutWriter = new
OutputStreamWriter(fOut);
        myOutWriter.append("Date:");
        myOutWriter.append(String.valueOf(Month + 1 + "/")); //
Add one because month is 0 based.
        myOutWriter.append(String.valueOf(Day + "/"));
        myOutWriter.append(String.valueOf(Year));
        myOutWriter.append("\r\n");
        myOutWriter.append("FullName:");
        myOutWriter.append(txtFullName.getText());
        myOutWriter.append("\r\n");
        myOutWriter.append("Sport1:");
        myOutWriter.append(SportFeedback1);
        myOutWriter.append("\r\n");
        myOutWriter.append("Sport2:");
        myOutWriter.append(SportFeedback2);
        myOutWriter.append("\r\n");
        myOutWriter.append("Cell Phone Number:");
        myOutWriter.append(CellPhoneNumber.getText());
        myOutWriter.append("\r\n");
        myOutWriter.append("Other Phone Number:");
        myOutWriter.append(OtherPhoneNumber.getText());
        myOutWriter.append("\r\n");

        if(SuddenDeathBefore50Response != null){
            myOutWriter.append("Family sudden death before age of
50 response:");
myOutWriter.append(SuddenDeathBefore50Response.getText());
            myOutWriter.append("\r\n");
        }

        if(SignificantInjuryResponse != null){
            myOutWriter.append("Significant injury response:");
myOutWriter.append(SignificantInjuryResponse.getText());
            myOutWriter.append("\r\n");
        }

        if(HospitalizedForAnyReasonResponse != null){

```

```

        myOutWriter.append("Hospitalized for any reason
response:");
myOutWriter.append(HospitalizedForAnyReasonResponse.getText());
        myOutWriter.append("\r\n");
    }

    if(OtherInjuryResponse != null){
        myOutWriter.append("Other injury (sprain, strain,
contusion) response:");
        myOutWriter.append(OtherInjuryResponse.getText());
        myOutWriter.append("\r\n");
    }

    if(PassedOutDuringAfterExerciseResponse != null){
        myOutWriter.append("Passed out during or after
exercise response:");
myOutWriter.append(PassedOutDuringAfterExerciseResponse.getText());
        myOutWriter.append("\r\n");
    }

    if(SurgeryOfAnyKindResponse != null){
        myOutWriter.append("Surgery of any kind response:");
myOutWriter.append(SurgeryOfAnyKindResponse.getText());
        myOutWriter.append("\r\n");
    }

    if(RecurrentBackPainResponse != null){
        myOutWriter.append("Recurrent back pain response:");
myOutWriter.append(RecurrentBackPainResponse.getText());
        myOutWriter.append("\r\n");
    }

    if(HighBloodPressureHighCholesterolResponse != null){
        myOutWriter.append("High blood pressure / high
cholesterol response:");
myOutWriter.append(HighBloodPressureHighCholesterolResponse.getText());
        myOutWriter.append("\r\n");
    }

    if(SignificantHeadInjuryKnockedUnconsciousResponse !=
null){
        myOutWriter.append("Significant head injury/knocked
unconscious response:");
myOutWriter.append(SignificantHeadInjuryKnockedUnconsciousResponse.getText());
;
        myOutWriter.append("\r\n");
    }

    if(SeizuresOfAnyKindResponse != null){
        myOutWriter.append("Seizures of any kind response:");

```

```

myOutWriter.append(SeizuresOfAnyKindResponse.getText());
    myOutWriter.append("\r\n");
}

    if(ShortnessOfBreatWithExerciseResponse != null){
        myOutWriter.append("Shortness of breath with
exercise? Wheezing response:");
myOutWriter.append(ShortnessOfBreatWithExerciseResponse.getText());
    myOutWriter.append("\r\n");
}

    if(ChronicRecurrentCoughWithExerciseResponse != null){
        myOutWriter.append("Chronic or recurrent cough with
exercise response:");
myOutWriter.append(ChronicRecurrentCoughWithExerciseResponse.getText());
    myOutWriter.append("\r\n");
}

    if(LossDecreasedFunctionOfAnyganResponse != null){
        myOutWriter.append("Loss or decreased function of any
organ response:");
myOutWriter.append(LossDecreasedFunctionOfAnyganResponse.getText());
    myOutWriter.append("\r\n");
}

}

if(AllergicReactionToMedicationStringingInsectFoodResponse != null){
    myOutWriter.append("Allergic reaction to medication,
stinging insect or food response:");
myOutWriter.append(AllergicReactionToMedicationStringingInsectFoodResponse.ge
tText());
    myOutWriter.append("\r\n");
}

    if(IrregularSkippingHeartbeatsResponse != null){
        myOutWriter.append("Irregular or skipping heartbeats
response:");
myOutWriter.append(IrregularSkippingHeartbeatsResponse.getText());
    myOutWriter.append("\r\n");
}

    if(MenstrualProblemsIrregularitiesResponse != null){
        myOutWriter.append("Menstrual problems /
irregularities response:");
myOutWriter.append(MenstrualProblemsIrregularitiesResponse.getText());
    myOutWriter.append("\r\n");
}

    if(AsthmaSeasonalAllergiesRequiringMedicationResponse !=
null){

```

```

        myOutWriter.append("Asthma / Seasonal allergies
requiring medication response:");

myOutWriter.append(AsthmaSeasonalAllergiesRequiringMedicationResponse.getText
());
        myOutWriter.append("\r\n");
    }

    if(UnusualOnsetOfHeadachesResponse != null){
        myOutWriter.append("Unusual onset of headaches
response:");

myOutWriter.append(UnusualOnsetOfHeadachesResponse.getText());
        myOutWriter.append("\r\n");
    }

    if(HaveYouBeenTreatedForAnEatingDisorderResponse !=
null){
        myOutWriter.append("Have you been treated for an
eating disorder response:");

myOutWriter.append(HaveYouBeenTreatedForAnEatingDisorderResponse.getText());
        myOutWriter.append("\r\n");
    }

    if(MonoOtherChronicIllnessesLastingLongerThanAWeekResponse != null){
        myOutWriter.append("Have you had mono or other
chronic illnesses that recur or last longer than a week response:");

myOutWriter.append(MonoOtherChronicIllnessesLastingLongerThanAWeekResponse.ge
tText());
        myOutWriter.append("\r\n");
    }

    if(BloodDisderEtcResponse != null){
        myOutWriter.append("Have you had any blood disorders
including anemia, sickle cell or unusual bleeding, etc response:");
        myOutWriter.append(BloodDisderEtcResponse.getText());
        myOutWriter.append("\r\n");
    }

    if(TreatedEncouragedFTreatmentFSubstanceAbuseResponse !=
null){
        myOutWriter.append("Have you been treated for or
encouraged to seek treatment for substance abuse problems from alcohol,
illicit drugs, prescription drugs or any other addictive substance
response:");

myOutWriter.append(TreatedEncouragedFTreatmentFSubstanceAbuseResponse.getText
());
        myOutWriter.append("\r\n");
    }

```



```

        if (WouldYouLikeInfoAboutSpeakToSomeoneAboutIssuesResponse
!= null) {
            myOutWriter.append("Would you like information about
or speak to someone confidentially about these issues response:");

myOutWriter.append(WouldYouLikeInfoAboutSpeakToSomeoneAboutIssuesResponse.getText());

            myOutWriter.append("\r\n");
        }

        if (StartedNewMedicationsResponse != null) {
            myOutWriter.append("Have you started any new
medications response:");

myOutWriter.append(StartedNewMedicationsResponse.getText());
            myOutWriter.append("\r\n");
        }

        if (AnyOtherProblemsIllnessesResponse != null) {
            myOutWriter.append("Any other problems or illnesses
response:");

myOutWriter.append(AnyOtherProblemsIllnessesResponse.getText());
            myOutWriter.append("\r\n");
        }

myOutWriter.append("Providers name:");
myOutWriter.append(ProvidersName.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Providers address:");
myOutWriter.append(ProvidersAddress.getText());
myOutWriter.append("\r\n");
myOutWriter.append(ProvidersCity.getText());
myOutWriter.append(", ");
myOutWriter.append(ProvidersState.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Providers phone number:");
myOutWriter.append(ProvidersPhoneNumber.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Dates treated by Provider:");
myOutWriter.append(ProvidersDatesTreated.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Initial for accurate info:");
myOutWriter.append(InitialForAccuracy.getText());
myOutWriter.append("\r\n");
myOutWriter.append("Initial for permmission to exchange
info:");

myOutWriter.append(InitialForPermmision.getText());
myOutWriter.append("\r\n");
myOutWriter.close();
fOut.close();
    } catch (Exception e) {
        Toast.makeText(getApplicationContext(), e.getMessage(),
Toast.LENGTH_SHORT).show();
    }
    Intent intent = new Intent(InterimhealthformPart6.this,
CaptureSignature.class);

```

```

        startActivityForResult(intent, SIGNATURE_ACTIVITY);
    } // onClick
}); // btnWriteSDFile
} // onCreate
/** Create a new dialog for date picker */
@Override
protected Dialog onCreateDialog(int id) {
    switch (id) {
        case DATE_DIALOG_ID:
            return new DatePickerDialog(this, pDateSetListener, Year, Month,
Day);
    }
    return null;
}
} // AndSDcard

```

### iterim\_health\_form\_part6.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#bbeeff" >

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:background="#FFFFFF1"
        android:isScrollContainer="true"
        android:orientation="vertical" >

        <TextView
            android:id="@+id/textView2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Interim Health Histry (Part 6)"
            android:textAppearance="?android:attr/textAppearanceMedium"
            android:textColor="@color/textcolor" />

        <TextView
            android:id="@+id/displayDate"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:text="Date will appear here after being selected"
            android:textColor="@color/textcolor"
            android:textSize="15sp" />

        <Button
            android:id="@+id/pickDate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Select Today's Date"
            android:textColor="@color/textcolor" />

```

```

<EditText
    android:id="@+id/txtFullName"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select athlete's year in school:"
    android:textColor="@color/textcolor" />

<Spinner
    android:id="@+id/YearInSchoolSpinner"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:entries="@array/YearInSchool" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select Sports:"
    android:textColor="@color/textcolor" />

<Spinner
    android:id="@+id/Spinner1"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/txtLastName"
    android:layout_marginTop="5dp"
    android:entries="@array/sports" />

<Spinner
    android:id="@+id/Spinner2"
    android:layout_width="180dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/txtLastName"
    android:layout_marginTop="5dp"
    android:entries="@array/sports" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Cell Phone Number:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/CellPhoneNumber"

```

```

    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

```

```

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Enter Other Phone You May Have:"
    android:textColor="@color/textcolor" />

```

```

<EditText
    android:id="@+id/OtherPhoneNumber"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

```

```

<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="" />

```

```

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="The purpose of this form is to update each Athletes
medical chart with any infmation regarding injuries, medical conditions,
medications or symptoms of illness since the initial medical exam upon
entering intercollegiate athletics. This information is vital to screen and
medically clear all athletes for continued participation in athletics at
NNU."

```

```

    android:textColor="@color/textcolor" />

```

```

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="*Has anyone in your family developed heart
problems/experienced sudden death before age 50?"
    android:textColor="@color/textcolor"
    android:textStyle="bold" />

```

```

<CheckBox
    android:id="@+id/SuddenDeathBefore50"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

```

```

<TextView
    android:id="@+id/textView4"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Answer all questions completely and to the best of
your ability."
        android:textColor="@color/textcolor"
        android:textStyle="bold" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="In the last six months, have any of the following
happened to you?"
    android:textColor="@color/textcolor" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Significant injury (fracture, dislocation etc.) "
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/SignificantInjury"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hospitalized for any reason?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/HospitalizedForAnyReason"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Other injury (sprain, strain, contusion)"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/OtherInjury"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Passed out during or after exercise?"
    android:textColor="@color/textcolor" />

```

```

<CheckBox
    android:id="@+id/PassedOutDuringAfterExercise"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Surgery of any kind?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/SurgeryOfAnyKind"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Recurrent back pain?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/RecurrentBackPain"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="High blood pressure / high cholesterol?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/HighBloodPressureHighCholesterol"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Significant head injury/knocked unconscious? "
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/SignificantHeadInjuryKnockedUnconscious"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:text="Seizures of any kind?"
        android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/SeizuresOfAnyKind"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Shortness of breath with exercise? Wheezing?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/ShortnessOfBreathWithExercise"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Chronic or recurrent cough with exercise?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/ChronicRecurrentCoughWithExercise"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Loss or decreased function of any gan?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/LossDecreasedFunctionOfAnygan"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Allergic reaction to medication, stinging insect
food?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/AllergicReactionToMedicationStringingInsectFood"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

```

```

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Irregular skipping heartbeats?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/IrregularSkippingHeartbeats"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Menstrual problems / irregularities?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/MenstrualProblemsIrregularities"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Asthma / Seasonal allergies requiring medication"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/AsthmaSeasonalAllergiesRequiringMedication"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Unusual onset of headaches?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/UnusualOnsetOfHeadaches"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you been treated for an eating disorder?"
    android:textColor="@color/textcolor" />

<CheckBox

```



```

        android:id="@+id/HaveYouBeenTreatedForAnEatingDisorder"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you had mono or other chronic illnesses that
recur or last longer than a week?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/MonoOtherChronicIllnessesLastingLongerThanAWeek"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you had any blood disorders including anemia,
sickle cell or unusual bleeding, etc.?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/BloodDisorderEtc"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you been treated for encouraged to seek
treatment for substance abuse problems from alcohol, illicit drugs,
prescription drugs or any other addictive substance?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/TreatedEncouragedFTreatmentFSubstanceAbuse"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Would you like information about or speak to
someone confidentially about these issues?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/WouldYouLikeInfoAboutSpeakToSomeoneAboutIssues"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

```

```

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Have you started any new medications?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/StartedNewMedications"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Any other problems or illness?"
    android:textColor="@color/textcolor" />

<CheckBox
    android:id="@+id/AnyOtherProblemsIllnesses"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="If you have seen a physician for any of the above
mentioned problems or injuries, please provide the name, address, phone
number and dates treated so we can obtain medical records if deemed necessary
by the team physicians:"
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/ProvidersName"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<EditText
    android:id="@+id/ProvidersPhoneNumber"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<EditText
    android:id="@+id/ProvidersAddress"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"

```

```

        android:layout_below="@+id/linearLayout1"
        android:ems="10"
        android:textSize="18sp" />

<EditText
    android:id="@+id/ProvidersCity"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<EditText
    android:id="@+id/ProvidersState"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<EditText
    android:id="@+id/ProvidersDatesTreated"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Please read the following and initial below where
indicated:"
    android:textColor="@color/textcolor"
    android:textStyle="bold" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="1. I certify that the infmation given above is
accurate and true to the best of my knowledge. I understand that
falsification of any infmation given above may jeopardize my ability to
participate in intercollegiate athletics."
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InitialForAccuracy"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"

```

```

        android:textSize="18sp" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="2. I give my permission f the Certified Athletic
Trainers, Student Center Health Clinic, and all consulting physicians'
permission to exchange written ally, any infmation concerning any injuries
illness which affects my ability to participate in physical activities during
the time in which I am an official student athlete at Northwest Nazarene
University. Any change is this status must be made in writing by the student
athlete and given to all concerned parties."
    android:textColor="@color/textcolor" />

<EditText
    android:id="@+id/InitialForPermmision"
    android:layout_width="320dp"
    android:layout_height="40dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/linearLayout1"
    android:ems="10"
    android:textSize="18sp" />

<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_above="@+id/signature"
    android:layout_alignParentLeft="true"
    android:paddingBottom="20dp"
    android:paddingTop="20dp"
    android:text="@string/Validate"
    android:textColor="#ff0000"
    android:textSize="12sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/signature"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/btnWriteSDFile"
    android:layout_alignParentLeft="true"
    android:layout_marginBottom="19dp"
    android:text="Click here to sign and finish" />
</LinearLayout>

</ScrollView>

```

## MainActivity.java

```

package com.mattrankin.sportdoc;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;

```

```

import android.view.View.OnClickListener;
import android.widget.Button;

public class MainActivity extends Activity {

    public static final int SIGNATURE_ACTIVITY = 1;

    // GUI controls
    private Button CompHealthHistoryPart1;
    private Button PrePartMedQuestPart2;
    private Button PhysicalExamPart3;
    private Button AthletesInsurInfoPart4;
    private Button ProvMedTreatPart5;
    private Button InterimhealthformPart6;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.menu);
        // bind GUI elements with local controls

        CompHealthHistoryPart1 = (Button)
findViewById(R.id.CompHealthHistoryPart1);
        CompHealthHistoryPart1.setOnClickListener(new OnClickListener() {

            public void onClick(View v) {

                Intent intent = new Intent(MainActivity.this,
CompHealthHistoryPart1.class);
                startActivityForResult(intent, SIGNATURE_ACTIVITY);
            }
        }); // btnClose

        PrePartMedQuestPart2 = (Button)
findViewById(R.id.PrePartMedQuestPart2);
        PrePartMedQuestPart2.setOnClickListener(new OnClickListener() {

            public void onClick(View v) {

                Intent intent = new Intent(MainActivity.this,
PrePartMedQuestPart2.class);
                startActivityForResult(intent, SIGNATURE_ACTIVITY);
            }
        }); // btnClose

        PhysicalExamPart3 = (Button) findViewById(R.id.PhysicalExamPart3);
        PhysicalExamPart3.setOnClickListener(new OnClickListener() {

            public void onClick(View v) {

                Intent intent = new Intent(MainActivity.this,
PhysicalExamPart3.class);
                startActivityForResult(intent, SIGNATURE_ACTIVITY);
            }
        }); // btnClose

        AthletesInsurInfoPart4 = (Button)
findViewById(R.id.AthletesInsurInfoPart4);

```

```

AthletesInsurInfoPart4.setOnClickListener(new OnClickListener() {

    public void onClick(View v) {

        Intent intent = new Intent(MainActivity.this,
AthletesInsurInfoPart4.class);
        startActivityForResult(intent,SIGNATURE_ACTIVITY);
    }
}); // btnClose

ProvMedTreatPart5 = (Button) findViewById(R.id.ProvMedTreatPart5);
ProvMedTreatPart5.setOnClickListener(new OnClickListener() {

    public void onClick(View v) {

        Intent intent = new Intent(MainActivity.this,
ProvMedTreatPart5.class);
        startActivityForResult(intent,SIGNATURE_ACTIVITY);
    }
}); // btnClose

InterimhealthformPart6 = (Button)
findViewById(R.id.InterimHealthFormPart6);
InterimhealthformPart6.setOnClickListener(new OnClickListener() {

    public void onClick(View v) {

        Intent intent = new Intent(MainActivity.this,
InterimhealthformPart6.class);
        startActivityForResult(intent,SIGNATURE_ACTIVITY);
    }
}); // btnClose

} // onCreate
}

```

## menu.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#FFFFFFD" >

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:textSize="35sp"
        android:background="#FFFFFFF"
        android:isScrollContainer="true"
        android:orientation="vertical" >

```

```

<Gallery
    android:id="@+id/gallery1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/title" />

<Button
    android:id="@+id/CompHealthHistoryPart1"
    android:layout_width="match_parent"
    android:layout_height="55dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/LinearLayout1"
    android:layout_marginTop="20dp"
    android:layout_marginRight="10dp"
    android:layout_marginLeft="10dp"
    android:textSize="15sp"
    android:background="@layout/button_theme"
    android:text="Comprehensive Health History (Part 1)" />

<Button
    android:id="@+id/PrePartMedQuestPart2"
    android:layout_width="match_parent"
    android:layout_height="55dp"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/CompHealthHistoryPart1"
    android:layout_marginTop="5dp"
    android:layout_marginRight="10dp"
    android:layout_marginLeft="10dp"
    android:textSize="15sp"
    android:background="@layout/button_theme"
    android:text="Pre Participation Medical Questionnaire (Part 2)" />

<Button
    android:id="@+id/PhysicalExamPart3"
    android:layout_width="match_parent"
    android:layout_height="55dp"
    android:layout_marginTop="5dp"
    android:layout_marginRight="10dp"
    android:layout_marginLeft="10dp"
    android:background="@layout/button_theme"
    android:text="Physical Exam (Part 3)"
    android:textSize="15sp" />

<Button
    android:id="@+id/AthletesInsurInfoPart4"
    android:layout_width="match_parent"
    android:layout_height="55dp"
    android:layout_marginTop="5dp"
    android:layout_marginRight="10dp"
    android:layout_marginLeft="10dp"
    android:background="@layout/button_theme"
    android:text="Athlete's Insurance Information (Part 4)"
    android:textSize="15sp" />

<Button
    android:id="@+id/ProvMedTreatPart5"
    android:layout_width="match_parent"

```

```

        android:layout_height="55dp"
        android:layout_marginTop="5dp"
        android:layout_marginRight="10dp"
        android:layout_marginLeft="10dp"
        android:background="@layout/button_theme"
        android:text="Permission to Provide Medical Treatment (Part 5)"
        android:textSize="15sp" />

<Button
    android:id="@+id/InterimHealthFormPart6"
    android:layout_width="match_parent"
    android:layout_height="55dp"
    android:layout_marginTop="5dp"
    android:layout_marginRight="10dp"
    android:layout_marginLeft="10dp"
    android:background="@layout/button_theme"
    android:text="Interim Health History (Part 6)"
    android:textSize="15sp" />

</LinearLayout>

</ScrollView>

```

## CaptureSignature.java

```

package com.matrankin.sportdoc;

import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.util.ArrayList;
import java.util.Calendar;
import android.annotation.SuppressLint;
import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.graphics.Path;
import android.graphics.RectF;
import android.net.Uri;
import android.os.Bundle;
import android.os.Environment;
import android.util.AttributeSet;
import android.view.Gravity;
import android.view.MotionEvent;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.ViewGroup.LayoutParams;
import android.view.Window;
import android.widget.Button;
import android.widget.EditText;
import android.widget.LinearLayout;

```



```

import android.widget.Toast;

public class CaptureSignature extends Activity {
    private LinearLayout Content;
    private signature Signature;
    private Button Clear, GetSignature, Cancel;
    private int count = 1;
    private String getPicDirectory = null; // Initialization of the directory
for the signature picture.
    private String getDocDirectory = null; // Initialization of the directory
for the document/txt.
    private String PicDir; // Local directory variable
    public static final int SIGNATURE_ACTIVITY = 1;
    private Bitmap mBitmap;
    private View mView;
    private File mypath;
    private String uniqueId;
    private EditText yourName;
    private String yourNameSet;
    static String DocDir; // Static string variable for all classes to use
including this one for attaching to email app.
    static String PicBindDir; // Used to bind signature to document

    @SuppressWarnings("deprecation")
    @Override
    public void onCreate(Bundle savedInstanceState){
        super.onCreate(savedInstanceState);
        this.requestWindowFeature(Window.FEATURE_NO_TITLE);
        setContentView(R.layout.signature);

        yourName = (EditText) findViewById(R.id.yourName); //Setup edit text
view and hints
        yourName.setHint("Enter your first and last name...");
        yourNameSet = yourName.getText().toString();

        //prepareDirectory();
        uniqueId = "BindedDocument:" + PicBindDir + "_" + getTodaysDate() +
"_" + getCurrentTime() + "_" + Math.random() + "_" + yourNameSet; //Get date,
time, and random generated value for signature picture authenticity
        PicDir = uniqueId + ".png"; // Add image png extension
        File directory = new File(Environment.getExternalStorageDirectory() +
"/SportDoc/");
        mypath = new File(directory, PicDir);
        Content = (LinearLayout) findViewById(R.id.linearLayout);
        Signature = new signature(this, null);
        Signature.setBackgroundColor(Color.WHITE);
        Content.addView(Signature, LayoutParams.FILL_PARENT,
LayoutParams.FILL_PARENT);
        Clear = (Button) findViewById(R.id.clear);
        GetSignature = (Button) findViewById(R.id.finish);
        GetSignature.setEnabled(false);
        Cancel = (Button) findViewById(R.id.cancel);
        mView = Content;

        Clear.setOnClickListener(new OnClickListener(){
            public void onClick(View v){
                Signature.clear();
            }
        });
    }
}

```

```

        GetSignature.setEnabled(false);
    }
});

GetSignature.setOnClickListener(new OnClickListener(){
    @SuppressWarnings("SdCardPath")
    public void onClick(View v){
        boolean error = captureSignature();
        if(!error){
            mView.setDrawingCacheEnabled(true);
            try {
                Signature.save(mView);
            }
            catch (IOException e1) {
                e1.printStackTrace();
            }
            Intent intent = new Intent();
            setResult(RESULT_OK,intent);

            try{
                final Intent emailIntent = new
Intent(android.content.Intent.ACTION_SEND_MULTIPLE);

emailIntent.setClassName("com.google.android.gm","com.google.android.gm.Compo
seActivityGmail"); // Prefer Gmail but most other email clients will work.
                emailIntent.setType("plain/text");

emailIntent.putExtra(android.content.Intent.EXTRA_EMAIL,new String[] {""});

emailIntent.putExtra(android.content.Intent.EXTRA_SUBJECT, "SportDoc
Received!");

                getPicDirectory =
Environment.getExternalStorageDirectory().getPath()+ "/SportDoc/" + PicDir;
// Signature directory in storage
                getDocDirectory = DocDir; // getDocDirectory set by
the java classes that use it through the static variable DocDir and add to
txt.

                String filePaths[] = {getPicDirectory,
getDocDirectory}; // File Directories for .jpg and .txt

                ArrayList<Uri> uris = new ArrayList<Uri>(); //
multiple filenames passed in as an ArrayList of parsable Uris
                for (String file : filePaths) {
                    File fileIn = new File(file);
                    Uri u = Uri.fromFile(fileIn);
                    uris.add(u);
                }

emailIntent.putParcelableArrayListExtra(Intent.EXTRA_STREAM, uris);
                try {
                    startActivity(Intent.createChooser(emailIntent,
"Sending mail...")); // Send Email
                }
                catch (android.content.ActivityNotFoundException ex)
{ // No email client available
                    Toast.makeText(CaptureSignature.this, "There are
no email clients installed.", Toast.LENGTH_SHORT).show();
                }
            }
        }
    }
});

```

```

        }
    }
    catch (Exception e) {
        Toast.makeText(getApplicationContext(), e.getMessage(),
Toast.LENGTH_SHORT).show();
    }
    finish();
}
});

Cancel.setOnClickListener(new OnClickListener() { // Cancel and return
to previous activity
    public void onClick(View v)
    {
        finish();
    }
});
}

@Override
protected void onDestroy() {
    super.onDestroy();
}

private boolean captureSignature() {
    boolean error = false;
    String errorMessage = "";

    if(yourName.getText().toString().equalsIgnoreCase("")){
        errorMessage = errorMessage + "Please enter your Name\n";
        error = true;
    }

    if(error){
        Toast toast = Toast.makeText(this, errorMessage,
Toast.LENGTH_SHORT);
        toast.setGravity(Gravity.TOP, 105, 50);
        toast.show();
    }
    return error;
}

private String getTodaysDate() { // Get system date for signature
validation
    final Calendar c = Calendar.getInstance();
    int todaysDate = (c.get(Calendar.YEAR) * 10000) +
((c.get(Calendar.MONTH) + 1) * 100) + (c.get(Calendar.DAY_OF_MONTH));
    return(String.valueOf(todaysDate));
}

private String getCurrentTime() { // Get system time for signature
validation
    final Calendar c = Calendar.getInstance();
    int currentTime = (c.get(Calendar.HOUR_OF_DAY) * 10000) +
(c.get(Calendar.MINUTE) * 100) + (c.get(Calendar.SECOND));

```

```

        return(String.valueOf(currentTime));
    }

    public class signature extends View{ // Get signature using a canvas
        private static final float STROKE_WIDTH = 5f;
        private static final float HALF_STROKE_WIDTH = STROKE_WIDTH / 2;
        private Paint paint = new Paint();
        private Path path = new Path();
        private float LastTouchXcoordinate;
        private float LastTouchYcoordinate;
        private final RectF dirtyRect = new RectF();

        public signature(Context context, AttributeSet attrs){
            super(context, attrs);
            paint.setAntiAlias(true);
            paint.setColor(Color.BLACK);
            paint.setStyle(Paint.Style.STROKE);
            paint.setStrokeJoin(Paint.Join.ROUND);
            paint.setStrokeWidth(STROKE_WIDTH);
        }

        public void save(View v) throws IOException{
            if(mBitmap == null)
            {
                mBitmap = Bitmap.createBitmap (Content.getWidth(),
                Content.getHeight(), Bitmap.Config.RGB_565);
            }
            Canvas canvas = new Canvas(mBitmap);
            FileOutputStream mFileOutStream = new FileOutputStream(mypath);
            v.draw(canvas);
            mBitmap.compress(Bitmap.CompressFormat.PNG, 90, mFileOutStream);
            mFileOutStream.flush();
            mFileOutStream.close();
        }

        public void clear(){ //Erase signature markings
            path.reset();
            invalidate(); // Repaint the view
        }

        @Override
        protected void onDraw(Canvas canvas){
            canvas.drawPath(path, paint);
        }

        @Override
        public boolean onTouchEvent(MotionEvent event){
            float eventX = event.getX();
            float eventY = event.getY();
            GetSignature.setEnabled(true);

            switch (event.getAction()){
                case MotionEvent.ACTION_DOWN:
                    path.moveTo(eventX, eventY);
                    LastTouchXcoordinate = eventX;
                    LastTouchYcoordinate = eventY;
                    return true;
            }
        }
    }
}

```



```

        android:layout_height="50dp"
        android:layout_weight=".30"
        android:text="Cancel" />

<Button
    android:id="@+id/clear"
    android:layout_width="0dp"
    android:layout_height="50dp"
    android:layout_weight=".35"
    android:text="Clear" />

<Button
    android:id="@+id/finish"
    android:layout_width="0dp"
    android:layout_height="50dp"
    android:layout_weight=".35"
    android:text="Finish" />
</LinearLayout>

<TableLayout
    android:id="@+id/tableLayout1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" >

    <TableRow
        android:id="@+id/tableRow1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" >

        <TextView
            android:id="@+id/textView2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="right"
            android:paddingLeft="10sp"
            android:text="Your Name"
            android:textAppearance="?android:attr/textAppearanceMedium"
/>

        <EditText
            android:id="@+id/yourName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:maxLength="30" >

            <requestFocus />
        </EditText>
    </TableRow>

    <TableRow
        android:id="@+id/tableRow3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" >

        <TextView
            android:id="@+id/textView2"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:maxLength="30"
        android:text=""
        android:textAppearance="?android:attr/textAppearanceMedium"
    />
</TableRow>

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:maxLength="60"
    android:text="Please Sign below and select gmail after clicking
Finish."
    android:textAppearance="?android:attr/textAppearanceMedium" />
</TableLayout>

<LinearLayout
    android:id="@+id/linearLayout"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />
</LinearLayout>

```

## arrays.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>

    <string-array name="sports">
        <item>@string/nosport</item>
        <item>@string/sport2</item>
        <item>@string/sport3</item>
        <item>@string/sport4</item>
        <item>@string/sport5</item>
        <item>@string/sport6</item>
        <item>@string/sport7</item>
        <item>@string/sport8</item>
        <item>@string/sport9</item>
        <item>@string/sport10</item>
    </string-array>
    <string-array name="age">
        <item>@string/noage</item>
        <item>@string/age1</item>
        <item>@string/age2</item>
        <item>@string/age3</item>
        <item>@string/age4</item>
        <item>@string/age5</item>
        <item>@string/age6</item>
        <item>@string/age7</item>
        <item>@string/age8</item>
        <item>@string/age9</item>
        <item>@string/age10</item>
        <item>@string/age11</item>
    </string-array>

```

```

    <item>@string/age12</item>
    <item>@string/age13</item>
    <item>@string/age14</item>
    <item>@string/age15</item>
    <item>@string/age16</item>
    <item>@string/age17</item>
    <item>@string/age18</item>
    <item>@string/age19</item>
    <item>@string/age20</item>
    <item>@string/age21</item>
    <item>@string/age22</item>
    <item>@string/age23</item>
    <item>@string/age24</item>
    <item>@string/age25</item>
    <item>@string/age26</item>
    <item>@string/age27</item>
    <item>@string/age28</item>
    <item>@string/age29</item>
    <item>@string/age30</item>
    <item>@string/age31</item>
    <item>@string/age32</item>
    <item>@string/age33</item>
    <item>@string/age34</item>
    <item>@string/age35</item>
    <item>@string/age36</item>
    <item>@string/age37</item>
    <item>@string/age38</item>
    <item>@string/age39</item>
    <item>@string/age40</item>
    <item>@string/age41</item>
    <item>@string/age42</item>
    <item>@string/age43</item>
    <item>@string/age44</item>
    <item>@string/age45</item>
    <item>@string/age46</item>
    <item>@string/age47</item>
    <item>@string/age48</item>
    <item>@string/age49</item>
    <item>@string/age50</item>
</string-array>
<string-array name="YearInSchool">
    <item>@string/none</item>
    <item>@string/Freshman</item>
    <item>@string/Sophomore</item>
    <item>@string/Junior</item>
    <item>@string/Senior</item>
    <item>@string/Transfer</item>
    <item>@string/RSSenior</item>
</string-array>
</resources>

```

## strings.xml

```
<resources>
```



```

    <string name="Validate">Please validate this document with your signature
(must be 18 or older) or a legal guardian signature:</string>
    <string name="external_dir">GetSignature</string>
    <string name="app_name">SportDoc</string>
    <string name="close_app">Close App</string>
    <string name="menu_settings">Settings</string>
    <string name="title_activity_main">SportDoc</string>
    <string
name="title_activity_CompHealthHistoryPart1">CompHealthHistoryPart1</string>
    <string
name="title_activity_PrePartMedQuestPart2">PreParticipationMedicalQuestionnai
rePart2)</string>
    <string
name="title_activity_PhysicalExamPart3">PhysicalExamPart3</string>
    <string
name="title_activity_AthletesInsurInfoPart4">AthletesInsurInfoPart4</string>
    <string
name="title_activity_ProvMedTreatPart5">Permission to Provide Medical Treatment Pa
rt5</string>
    <string
name="title_activity_InterimhealthformPart6">InterimHealthHistoryPart6</strin
g>
    <string
name="title_activity_signatures_save">Signatures_saveActivity</string>
    <string
name="title_activity_capture_signature">CaptureSignatureActivity</string>

    <string name="nosport"></string>
    <string name="sport2">Baseball</string>
    <string name="sport3">Basketball</string>
    <string name="sport4">Cross Country</string>
    <string name="sport5">Golf</string>
    <string name="sport6">Soccer</string>
    <string name="sport7">Softball</string>
    <string name="sport8">Track</string>
    <string name="sport9">Volleyball</string>
    <string name="sport10">Tennis</string>

    <string name="toggle_right">Right</string>
    <string name="toggle_left">Left</string>

    <string name="toggle_male">Male</string>
    <string name="toggle_female">Female</string>

    <string name="toggle_no">No</string>
    <string name="toggle_yes">Yes</string>

    <string name="toggle_more">More</string>
    <string name="toggle_less">Less</string>

    <string name="noage"></string>
    <string name="age1">1</string>
    <string name="age2">2</string>
    <string name="age3">3</string>
    <string name="age4">4</string>
    <string name="age5">5</string>

```

```
<string name="age6">6</string>
<string name="age7">7</string>
<string name="age8">8</string>
<string name="age9">9</string>
<string name="age10">10</string>
<string name="age11">11</string>
<string name="age12">12</string>
<string name="age13">13</string>
<string name="age14">14</string>
<string name="age15">15</string>
<string name="age16">16</string>
<string name="age17">17</string>
<string name="age18">18</string>
<string name="age19">19</string>
<string name="age20">20</string>
<string name="age21">21</string>
<string name="age22">22</string>
<string name="age23">23</string>
<string name="age24">24</string>
<string name="age25">25</string>
<string name="age26">26</string>
<string name="age27">27</string>
<string name="age28">28</string>
<string name="age29">29</string>
<string name="age30">30</string>
<string name="age31">31</string>
<string name="age32">32</string>
<string name="age33">33</string>
<string name="age34">34</string>
<string name="age35">35</string>
<string name="age36">36</string>
<string name="age37">37</string>
<string name="age38">38</string>
<string name="age39">39</string>
<string name="age40">40</string>
<string name="age41">41</string>
<string name="age42">42</string>
<string name="age43">43</string>
<string name="age44">44</string>
<string name="age45">45</string>
<string name="age46">46</string>
<string name="age47">47</string>
<string name="age48">48</string>
<string name="age49">49</string>
<string name="age50">50</string>

<string name="none"></string>
<string name="Freshman">Freshman</string>
<string name="Sophomore">Sophomore</string>
<string name="Junior">Junior</string>
<string name="Senior">Senior</string>
<string name="Transfer">Transfer</string>
<string name="RSSenior">RS Senior</string>
```

```
</resources>
```

## styles.xml

```
<resources>

    <style name="AppTheme" parent="android:Theme.Light" />

</resources>
```

## AndroidManifest.xml

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.mattrankin.sportdoc"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="15" />

    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"
/>
    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@android:style/Theme.NoTitleBar" >
        <activity
            android:name=".MainActivity"
            android:label="@string/title_activity_main">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".CompHealthHistoryPart1"
            android:configChanges="keyboardHidden|orientation"
            android:label="@string/title_activity_CompHealthHistoryPart1" >
        </activity>
        <activity
            android:name=".PrePartMedQuestPart2"
            android:configChanges="keyboardHidden|orientation"
            android:label="@string/title_activity_PrePartMedQuestPart2" >
        </activity>
        <activity
            android:name=".PhysicalExamPart3"
            android:configChanges="keyboardHidden|orientation"
            android:label="@string/title_activity_PhysicalExamPart3" >
        </activity>
        <activity
            android:name=".AthletesInsurInfoPart4"
```

```

        android:configChanges="keyboardHidden|orientation"
        android:label="@string/title_activity_AthletesInsurInfoPart4" >
    </activity>
    <activity
        android:name=".ProvMedTreatPart5"
        android:configChanges="keyboardHidden|orientation"
        android:label="@string/title_activity_ProvMedTreatPart5" >
    </activity>
    <activity
        android:name=".InterimhealthformPart6"
        android:configChanges="keyboardHidden|orientation"
        android:label="@string/title_activity_InterimhealthformPart6" >
    </activity>
    <activity
        android:name=".Signatures_saveActivity"
        android:configChanges="keyboardHidden|orientation"
        android:label="@string/title_activity_signatures_save" >
    </activity>
    <activity
        android:name=".CaptureSignatureActivity"
        android:configChanges="keyboardHidden|orientation"
        android:label="@string/title_activity_capture_signature" >
    </activity>
    <activity
        android:name=".CaptureSignature"
        android:configChanges="keyboardHidden|orientation"
        android:label="@string/title_activity_capture_signature" >
    </activity>
</application>

</manifest>

```