Introduction

In this book I'll show you how to program a One Player Poker game. I assume that you have worked with Android Studio before and are comfortable with the basics. That being said, the code I'm going to explain to you, can be used with other programming languages if you follow the same logic.

Our aim is to make an app that does the following:

1. Draws and displays 5 cards
2. Works out what the hand was (e.g. Flush, Straight, Full house, etc)
3. Allows the user to swap any of the 5 cards for new ones
4. Re-works out what the hand was
5. Displays the result and calculates the winnings.

Chapter 1:
Where does the code go?

Open up Android Studio and start a new project. Next go to the MainActivity.java

This is broken into a number of sections:

The first section starts with:

```
package mjprojects.poker;
```

Below this section you will have all the imports. The imports that we will be using are as follows:

```
import android.content.SharedPreferences;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.TextView;
import android.view.View.OnClickListener;
import android.widget.ImageView;
import java.util.Random;
```

You can include them now below **package.**

The next following sections are all contained in the:

```
public class MainActivity extends AppCompatActivity {
```

The second section just below this line is where we will declare all our variables.

After this we have the third section that contains three @override methods.

```
@Override
protected void onCreate(Bundle savedInstanceState) {…}

@Override
public boolean onCreateOptionsMenu(Menu menu) {…}

@Override
public boolean onOptionsItemSelected(MenuItem item) {…}
```

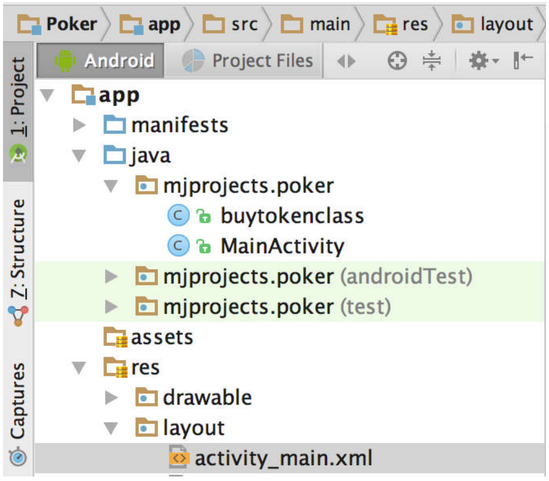
Finally we have the forth section below this that will contain all our methods.

So to recap we have 4 sections:

1. Import section
2. Variables section
3. Override Methods section
4. Our Methods section

I’ll refer to these four sections throughout the book to help you know where to place your code.

Chapter 2:
Setting up the User Interface

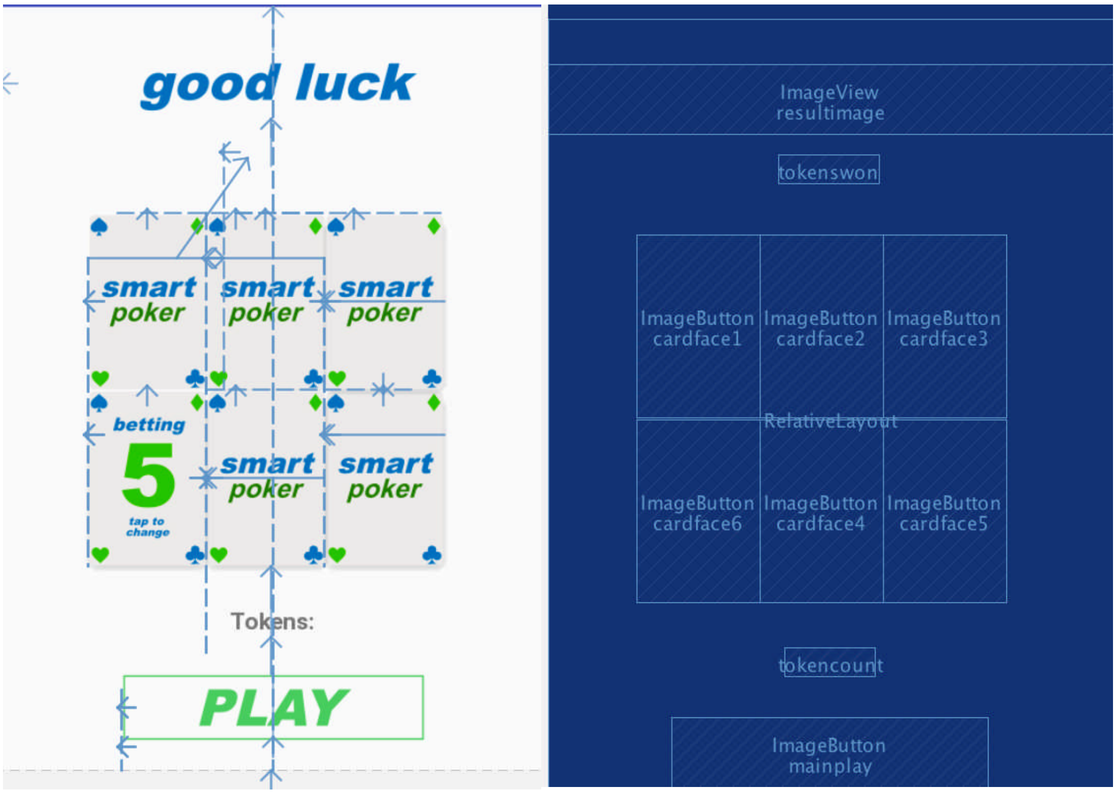


Go to res, layout, activity_main.xml.
At the bottom you will be able to switch to the design view.

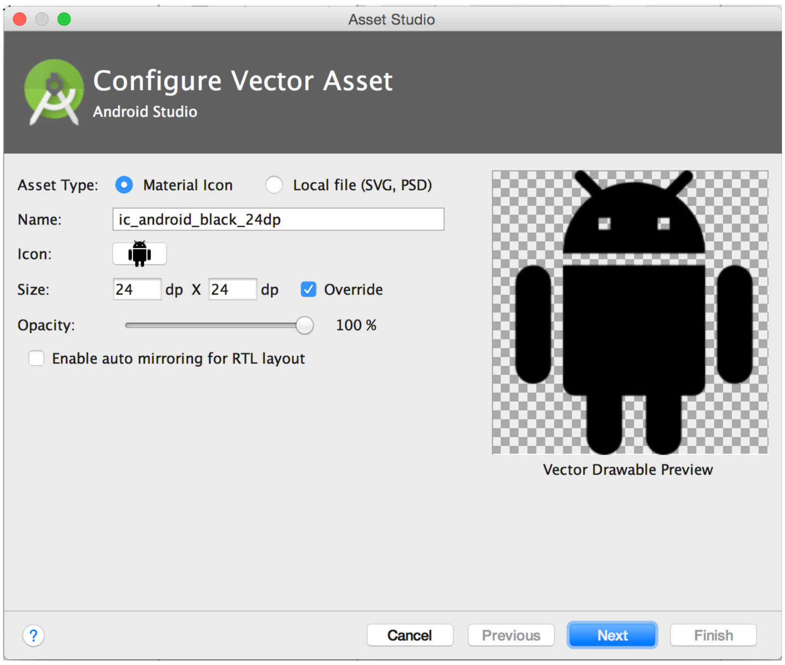
Add the following UI component:
(I've found RelativeLayout to be the easiest to work with)

1. Image to display the Results
2. A text field for debugging purposes
3. Five buttons to represent the cards
4. One button for changing the amount of the bet
5. A text field to display total coins
6. A play button.

My layout looks like this:



I made my own images in Affinity Designer. I saved the images as an SVG file. I then went to res, drawable and right clicked, new, Vector Asset to add them to my project



Asset Type: ● Material Icon ○ Local file (SVG, PSD)

Name: ic_android_black_24dp

Icon:

Size: 24 dp X 24 dp ☑ Override

Opacity: 100 %

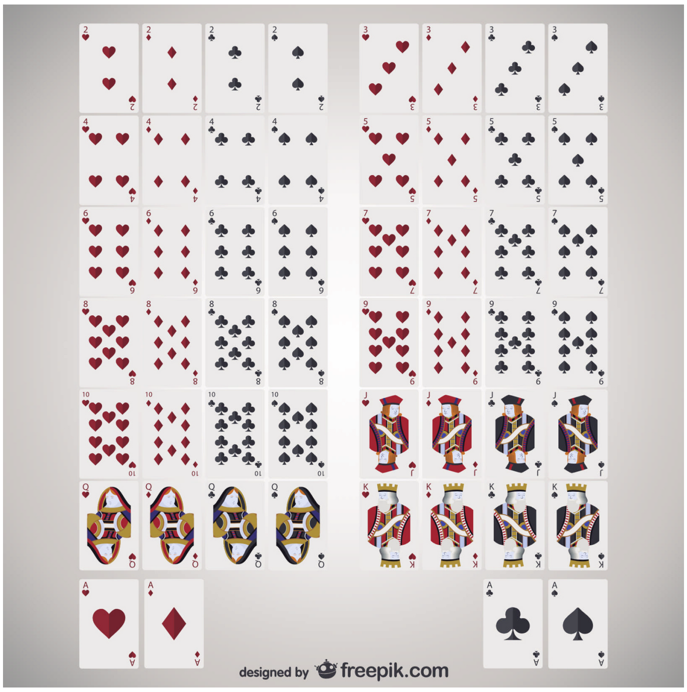
□ Enable auto mirroring for RTL layout

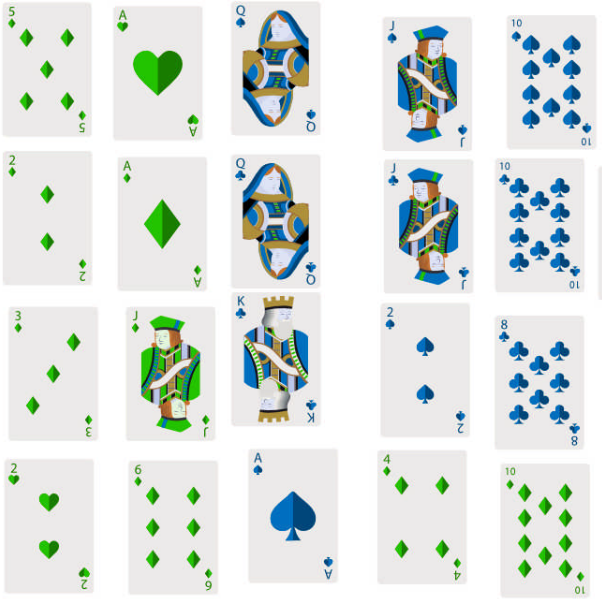
You will need patience while setting up the User Interface as it tricky to place everything where you want it to be.

For this project you will need the following images:

- 52 Card Images (Ace of Spades to King of Hearts)
- 8 Result Images (Flush, Straight, Fullhouse, etc)
- 3 Play Button States (I'll talk about them later)
- 3 Betting States (You can have more or less)
- 2 Message Images ("Good Luck" and "Out of Coins")
- 2 Information Images.

For the 52 Cards, I downloaded a free Vector Pack and then edited the colours to blue and green.





Chapter 3:
How to Draw 5 Cards.

We need to draw unique 5 cards from a pack of 52 to make up our starting hand.

In Section 4 (Our Method section) of the MainActivity we need to create a method called drawcards

The code starts as follows.

```
public void drawcards(){

    //First Card
    Random s1 = new Random();
    Random n1 = new Random();
    suit1 = s1.nextInt(5 - 1) + 1;
    num1 = n1.nextInt(14 - 1) + 1;

    nameofcard1 = "ic_"+suit1+""+num1;
    ((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
```

We generate two random numbers.
The first random number is between 1 and 4.
It determines the cards suit.
(Hearts, Spades, Diamonds, Clovers)
The second random number is between 1 and 13.
It determines the face of the card.
(Ace,2,3,4,5,6,7,8,9,10,Jack,Queen,King)

I assign these values to variables suit1 and num1.

Remember to go to Section 2 and declare these variables there.

```
int num1 =1;
int suit1 =1;
String nameofcard1
```

I give them a default value to prevent the program from crashing.

I have another variable called nameofcard1.
This is a string that will refer to one of the Card Vector Assets we created in the previous chapter.

It is important to save your Vectors in such a way that its easy to refer to them.

Finally the following code changes the Card Button to a picture that relates to the card generated.

```
((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1,”drawable",getPackageName()));
```

Drawing the Second Card

```
//Second Card
Random s2 = new Random();
Random n2 = new Random();
suit2 = s2.nextInt(5 - 1) + 1;
num2 = n2.nextInt(14 - 1) + 1;

while (suit1 == suit2 && num1 == num2) {
    n2 = new Random();
    num2 = n2.nextInt(14 - 1) + 1;
}

nameofcard2 = "ic_"+suit2+""+num2;
((ImageView)
findViewById(R.id.cardface2)).setImageResource(getResources().getI
dentifier(nameofcard2,"drawable",getPackageName()));
```

Again we generate two random numbers. One for the suit and one for the face number. But we are at risk of drawing exactly the same card as we did before.

To prevent this from happening we introduce a While Loop. The logic is as follows:
While the suit and number of both cards is the same, generate a new suit and number for the second card. Keep doing this until a different number is generated. Once this happens the While Loop breaks and the code can continue.

This way we can ensure that the second card is different from the first card.

Drawing the rest of the cards

The logic is the same for the last 3 cards. Except now we need to extend the While Loop to check; that the third card is different from the first two, that the fourth card is different from the first three, that the fifth card is different from the first four. Here is the code.

Remember to declare your variables in Section 2.

```
//Third Card
Random s3 = new Random();
Random n3 = new Random();
suit3 = s3.nextInt(5 - 1) + 1;
num3 = n3.nextInt(14 - 1) + 1;

while( (suit3 == suit1 && num3 == num1) || (suit3 == suit2 && num3
== num2)){

    s3 = new Random();
    n3 = new Random();
    suit3 = s3.nextInt(5 - 1) + 1;
    num3 = n3.nextInt(14 - 1) + 1;

}

nameofcard3 = "ic_"+suit3+""+num3;
((ImageView)
findViewById(R.id.cardface3)).setImageResource(getResources().getI
dentifier(nameofcard3,"drawable",getPackageName()));
```

```
//Forth Card
Random s4 = new Random();
Random n4 = new Random();
suit4 = s4.nextInt(5 – 1) + 1;
num4 = n4.nextInt(14 – 1) + 1;


while( (suit4 == suit1 && num4 == num1) || (suit4 == suit2 && num4
== num2) || (suit4 == suit3 && num4 == num3)){

    s4 = new Random();
    n4 = new Random();
    suit4 = s4.nextInt(5 – 1) + 1;
    num4 = n4.nextInt(14 – 1) + 1;

}
nameofcard4 = "ic_"+suit4+""+num4;
((ImageView)
findViewById(R.id.cardface4)).setImageResource(getResources().getI
dentifier(nameofcard4,"drawable",getPackageName()));


//Fifth Card
Random s5 = new Random();
Random n5 = new Random();
suit5 = s5.nextInt(5 – 1) + 1;
num5 = n5.nextInt(14 – 1) + 1;


while( (suit5 == suit1 && num5 == num1) || (suit5 == suit2 && num5
== num2) || (suit5 == suit3 && num5 == num3) || (suit5 == suit4 &&
num5 == num4)){

    s5 = new Random();
    n5 = new Random();
    suit5 = s5.nextInt(5 – 1) + 1;
    num5 = n5.nextInt(14 – 1) + 1;

}

nameofcard5 = "ic_"+suit5+""+num5;
((ImageView)
findViewById(R.id.cardface5)).setImageResource(getResources().getI
dentifier(nameofcard5,”drawable",getPackageName()));
```

Chapter 4:
Working out the hand

FLUSH

It is very easy to determine if our hand is a Flush.
We simply check if all 5 cards have the same suit.
If the answer is yes, then we have a Flush.

STRAIGHT

It is a little more difficult to determine if our hand is a Straight. For a hand to be a Straight, the 5 cards need to be in a consecutive order.

This means we first need to order our cards.
To do this we need to create a new method and create an array

```
public void displayresults(){

int[] input = {num1, num2, num3, num4, num5};
bubble_srt(input);
```

}

Next we need to make the bubble_srt method which will sort the cards for us.

```
public void bubble_srt(int array[]) {
    int n = array.length;
    int k;
    for (int m = n; m >= 0; m--) {
        for (int i = 0; i < n - 1; i++) {
            k = i + 1;
            if (array[i] > array[k]) {
                swapNumbers(i, k, array);
            }
        }
    }
    high5 = array[0];
    high4 = array[1];
    high3 = array[2];
    high2 = array[3];
    high1 = array[4];

}
```

Remember to declare the high1 to high5 variables

This method works by running a for loop on the array and then rearranging the array.

It does it by checking each element of the array with the others and if one of them is bigger, it swaps them by calling the swapNumbers method

```
private void swapNumbers(int i, int j, int[] array) {

    int temp;
    temp = array[i];
    array[i] = array[j];
    array[j] = temp;
}
```

Remember to create these methods in Section 4. Now we have our cards in order.

Run the following code in the displayResult method just under the bubble_srt(input) line

```
if( suit1 == suit2 && suit2 == suit3 && suit3 == suit4 && suit4 ==
suit5){

    if ( high1 == (high2 +1) && high1 == (high3 + 2) && high1 ==
(high4 + 3) && (high1 == (high5 + 4) || high1 == (high5 + 12))){

        //STRAIGHT FLUSH RESULT
        result = "Straight Flush";
        winnings = 2500 * (bet/5);
        ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_straightflush","drawable",getPackageName()));


    }else{
        //FLUSH RESULT
        result = "Flush";
        winnings = 450 * (bet/5);
        ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_flush","drawable",getPackageName()));
    }

}else{
    if (high1 == (high2 +1) && high1 == (high3 + 2) && high1 ==
(high4 + 3) && (high1 == (high5 + 4) || high1 == (high5 + 12))){
        //STRAIGHT RESULT
        result = "Straight";
        winnings = 100 * (bet/5);
        ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_straight","drawable",getPackageName()));
    } else{
        //HIGH CARD RESULT
        result = "High Card";
        winnings = 0 * (bet/5);
        ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_highcard","drawable",getPackageName()));
    }
}
```

The logic of this code is as follows:

If all the suits are the same:
And nested in this statement, if high1, high2, high3, high4 and high5 are consecutive, then we have a Straight Flush. Else, we have a Flush.

Else if all the suits are not the same, but the cards are consecutive then we have a Straight. If the cards are not all the same suit and are not consecutive then we just have a High Card Hand for the time being.

It is important to check that the highest card is either equal to the lowest card + 5 OR equal to the lowest card + 12. This captures the Straight that is, Ace, King, Queen, Jack, 10 as well as allowing us to capture the Straight that is, Ace, 2, 3, 4, 5.

But we are not done yet. We now need to see if our hand has any matching pairs.

Matching Pairs

This was one of hardest algorithms to code for the poker project so I'll explain the logic step by step.

Essentially what we want to do is compare the first card with the other four and see if it is the same as any of them.

Then we want to compare the second card with the other three and see if it is the same as any of them.

Then we want to compare the third card with the other two and see if it is the same as any of them

And finally we want to compare the forth card with the last card and see if they are the same.

By same, I mean the cards have the same number. Remember its impossible for any two cards to have the same suit and the same number. By keeping the cards' number and suit separate, it makes the rest of the code simpler to compute.

Remember that array we created? Well this time we run it in a method that I call Bubble Match.

```
public void displayresults(){


    int[] input = {num1, num2, num3, num4, num5};
    bubble_srt(input);

    pair = 0;
    bubble_match2(input);
```

We need to declare an integer called “pair” in Section 2 of our code.

The Bubble Method could be a for loop within a for loop but I have written it out like this so that it is easier to see what is happening.

```
public void bubble_match2 (int array[]){

    for( int n = 1; n <= 4; n++) {
        if (array[0] == array[n]) {
            pair = pair + 1;
        }
    }

    for( int n = 2; n <= 4; n++) {
        if (array[1] == array[n]) {
            pair = pair + 1;
        }
    }

    for( int n = 3; n <= 4; n++) {
        if (array[2] == array[n]) {
            pair = pair + 1;
        }
    }

    for( int n = 4; n <= 4; n++) {
        if (array[3] == array[n]) {
            pair = pair + 1;
        }
    }
}
```

What we want to do is count every time one of the cards match with another card. But it is important to reset this counter at the start of the previous method, hence why we said pair = 0 in the displayResult method.

Now that we have counted the amount of matches, we need to determine what hand our cards have made.

If we have 1 match it means only two cards were the same and therefore the hand is a “One Pair”

If we have 2 matches it means that we have two accounts of two cards matching and therefore the hand is a “Two Pair”

If we have 3 matches it means that we have three cards that were the same and therefore the hand is a “Three of a kind”

If we have 4 matches it means that we have three cards that were the same and that the other two cards were matching each other. Therefore the hand is a “Full House”

If we have 6 matches it means that we have four cards that are the same and therefore the hand is a “Four of a Kind”

The code for the above is a simple set of if statements.

```java
if(pair == 1){
    //ONE PAIR RESULT
    result = "One Pair";
    ((ImageView) findViewById(R.id.resultimage)).setImageResource(getResources().getIdentifier("ic_1pair","drawable",getPackageName()));
    winnings = 2 * (bet/5);
}
if (pair == 2){
    //TWO PAIR RESULT
    result = "Two Pair";
    winnings = 20 * (bet/5);
    ((ImageView) findViewById(R.id.resultimage)).setImageResource(getResources().getIdentifier("ic_2pairs","drawable",getPackageName()));
}
if (pair == 3){
    //THREE OF A KIND RESULT
    result = "Three of a Kind";
    winnings = 40 * (bet/5);
    ((ImageView) findViewById(R.id.resultimage)).setImageResource(getResources().getIdentifier("ic_3ofakind","drawable",getPackageName()));
}
if (pair == 4){
    //FULL HOUSE RESULT
    result = " Full House";
    winnings = 500 * (bet/5);
    ((ImageView) findViewById(R.id.resultimage)).setImageResource(getResources().getIdentifier("ic_fullhousebar","drawable",getPackageName()));
}
if (pair == 6){
    //FOUR OF A KIND RESULT
    result = "Four of a Kind";
    winnings = 1000 * (bet/5);
    ((ImageView) findViewById(R.id.resultimage)).setImageResource(getResources().getIdentifier("ic_4ofakind","drawable",getPackageName()));
}
```

Its important to reward the player for getting a special hand and that is what the winnings variable is taking account of.

It is also important to communicate to the user what there hand was and so I made a picture message for each hand result that gets displayed when that hand is formed.

This code is run after the initial check. The initial check determines if the hand is a Straight Flush, Straight, Flush or High Card. If pair = 0 then the hand will remain as a High Card hand.

Chapter 5:
Swapping Cards

Poker is a game of skill and chance. The first 5 cards you receive are based on chance. The skill is knowing which of these cards to swap out for a better hand.

Each card in the app is actually a button that becomes enabled after the first 5 cards are drawn. The user can tap on any card to signify that they want it swapped. When they do this the card face changes to a placeholder face. After the user has tapped all the cards that they want to swap, they will then click the Swap Button (previously the Play Button) and receive new cards. The program will then re run the code that checks to see what hand the user has.

In order to achieve this we need a toggle variable to keep track of what state our app is in.

Declare the toggle variable and set it to zero.

```
int toggle = 0;
```

When the play button is tapped, the toggle value will change. Then we cards are tapped to be swapped the toggle value will change. The toggle states are as follows:

Toggle = 0; beginning of game
Toggle = 1; a card has been tapped to be swapped
Toggle = 2; cards have been swapped
Toggle = 3; the first 5 cards have been drawn

Depending on what value the toggle has, determines whether cards can be tapped and determines whether the main button is either the Play Button, the Swap Button or the Accept Hand Button.

Toggle = 0; Play Button
Toggle = 1; Swap Button
Toggle = 2; Accept Hand Button
Toggle = 3; Accept Hand Button

Cards can only be tapped when the toggle variable is either 1 or 3. Lets now look at the code.

```
public void CardOneButton (View v){


    if(toggle == 3 || toggle ==1) {

        if (select1 == 0) {

            select1 = 1;
            nameofcard1 = "ic_swap";
            ((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1, "drawable", getPackageName()));

        } else {
            select1 = 0;
            nameofcard1 = "ic_" + suit1 + "" + num1;
            ((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1, "drawable", getPackageName()));
        }

        if (select1 == 1 || select2 == 1 || select3 == 1 ||
select4 == 1 || select5 == 1) {
            toggle = 1;
            playbuttonmessage = "ic_swapbutton";
            ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
        } else {
            toggle = 3;
            playbuttonmessage = "ic_accepthand";
            ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
        }
    }
}
```

Each card will also have its own toggle variable which I have called them select1, select2, select3, select4, select5.

All cards start with their select variable as 0.
When they are tapped, their select variable becomes 1 if it was 0 and it becomes 0 if it was 1.

When the Swap Button is pressed, all cards that have a select value of 1, will be swapped.

All this method is doing is changing the select value, the picture of the card and the toggle state.

There are 5 of these methods, one for each of the 5 cards.

As long as one of the select variables has a value of 1, the toggle state will become 1. If all select variables are 0, the toggle state will become 3.

Remember when we generated the 5 random cards. Well just below that code, in the same method, we are going to generate 5 more cards. The code follows the same logic as before:

```
// THE EXTRA CARDS
// SIXTH EXTRA CARD
Random s6 = new Random();
Random n6 = new Random();
suit6 = s6.nextInt(5 - 1) + 1;
num6 = n6.nextInt(14 - 1) + 1;

while( (suit6 == suit1 && num6 == num1) || (suit6 == suit2 && num6
== num2) || (suit6 == suit3 && num6 == num3) || (suit6 == suit4 &&
num6 == num4) || (suit6 == suit5 && num6 == num5)){

    s6 = new Random();
    n6 = new Random();
    suit6 = s6.nextInt(5 - 1) + 1;
    num6 = n6.nextInt(14 - 1) + 1;

}

// SEVENTH EXTRA CARD
Random s7 = new Random();
Random n7 = new Random();
suit7 = s7.nextInt(5 - 1) + 1;
num7 = n7.nextInt(14 - 1) + 1;

while( (suit7 == suit1 && num7 == num1) || (suit7 == suit2 && num7
== num2) || (suit7 == suit3 && num7 == num3) || (suit7 == suit4 &&
num7 == num4) || (suit7 == suit5 && num7 == num5) || (suit7 ==
suit6 && num7 == num6)){

    s7 = new Random();
    n7 = new Random();
    suit7 = s7.nextInt(5 - 1) + 1;
    num7 = n7.nextInt(14 - 1) + 1;
}

// EIGHTH EXTRA CARD
Random s8 = new Random();
Random n8 = new Random();
suit8 = s8.nextInt(5 - 1) + 1;
num8 = n8.nextInt(14 - 1) + 1;

while( (suit8 == suit1 && num8 == num1) || (suit8 == suit2 && num8
== num2) || (suit8 == suit3 && num8 == num3) || (suit8 == suit4 &&
num8 == num4) || (suit8 == suit5 && num8 == num5) || (suit8 ==
suit6 && num8 == num6) || (suit8 == suit7 && num8 == num7)){

    s8 = new Random();
    n8 = new Random();
    suit8 = s8.nextInt(5 - 1) + 1;
    num8 = n8.nextInt(14 - 1) + 1;
}

// NINTH EXTRA CARD
Random s9 = new Random();
Random n9 = new Random();
suit9 = s8.nextInt(5 - 1) + 1;
num9 = n8.nextInt(14 - 1) + 1;

while( (suit9 == suit1 && num9 == num1) || (suit9 == suit2 && num9
== num2) || (suit9 == suit3 && num9 == num3) || (suit9 == suit4 &&
num9 == num4) || (suit9 == suit5 && num9 == num5) || (suit9 ==
suit6 && num9 == num6) || (suit9 == suit7 && num9 == num7) ||
(suit9 == suit8 && num9 == num8)){

    s9 = new Random();
    n9 = new Random();
    suit9 = s9.nextInt(5 - 1) + 1;
    num9 = n9.nextInt(14 - 1) + 1;
}


// TENTH EXTRA CARD
Random s10 = new Random();
Random n10 = new Random();
suit10 = s10.nextInt(5 - 1) + 1;
num10 = n10.nextInt(14 - 1) + 1;

while( (suit10 == suit1 && num10 == num1) || (suit10 == suit2 &&
num10 == num2) || (suit10 == suit3 && num10 == num3) || (suit10 ==
suit4 && num10 == num4) || (suit10 == suit5 && num10 == num5) ||
(suit10 == suit6 && num10 == num6) || (suit10 == suit7 && num10 ==
num7) || (suit10 == suit8 && num10 == num8) || (suit10 == suit9 &&
num10 == num9)){

    s10 = new Random();
    n10 = new Random();
    suit10 = s10.nextInt(5 - 1) + 1;
    num10 = n10.nextInt(14 - 1) + 1;
}
```

Chapter 6:
Bringing it all together

We now create the most important method. The Play method. It does 4 different things depending on what state our toggle position is in.

If Toggle is in state 0 then it does the following:

- Calls the drawcards() method
- Calls the displayresults() method
- Reduces total coins by the bet
- Changes the toggle state to 3
- Changes the Play Button text to Accept Hand
- Reset the select toggles to 0

If Toggle is in state 1 then it does the following:

- Swaps the cards that were previously tapped
- Changes the toggle state to 2
- Chances the Play Button text to Accept Hand
- Card 1 becomes Card 6
- Card 2 becomes Card 7
- Card 3 becomes Card 8
- Card 4 becomes Card 9
- Card 5 becomes Card 10

But only if it the select toggle for that card is 1

If Toggle is in state 2 or 3 then it does the following:

- Changes the Play Button text to Play
- Changes the toggle state to 0
- Changes all the card faces to the card backs
- Save the coins

The code is as follows:

```
public void Play (View v) {
        if( toggle == 0) {

            playbuttonmessage = "ic_accepthand";
            ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage,"drawable",getPackageName()));
            ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_taporaccept","drawable",getPackageName()));
            bettoggle = 3;


            select1 = 0;
            select2 = 0;
            select3 = 0;
            select4 = 0;
            select5 = 0;

            drawcards();
            displayresults();
            toggle = 3;
            coins = coins - bet;

            ((TextView)
findViewById(R.id.tokencount)).setText("Tokens " + coins+ " +
"+winnings);


        }else
    if(toggle == 1) {

        if (select1 == 1) {

            num1 = num6;
            suit1 = suit6;

        }
        if (select2 == 1) {

            num2 = num7;
            suit2 = suit7;

        }
        if (select3 == 1) {

            num3 = num8;
            suit3 = suit8;
        }
        if (select4 == 1) {

            num4 = num9;
            suit4 = suit9;

        }
        if (select5 == 1) {

            num5 = num10;
            suit5 = suit10;

        }

        toggle = 2;
        displayresults();
        playbuttonmessage = "ic_accepthand";
        ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage,"drawable",getPackageName()));
        ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_acceptwinnings","drawable",getPackageName()));
        bettoggle = 3;

        ((TextView) findViewById(R.id.tokencount)).setText("Tokens
" + coins+" + "+winnings);

    }else
    if( toggle == 2 || toggle == 3){

        toggle = 0;

        playbuttonmessage = "ic_playbuttonpoker";
        ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage,"drawable",getPackageName()));

        nameofcard1 = "ic_cardbacknew";
        ((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
```

```
        ((ImageView)
findViewById(R.id.cardface2)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
        ((ImageView)
findViewById(R.id.cardface3)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
        ((ImageView)
findViewById(R.id.cardface4)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
        ((ImageView)
findViewById(R.id.cardface5)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));

        ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_goodluck","drawable",getPackageName()));
        coins = coins + winnings;
        ((TextView) findViewById(R.id.tokencount)).setText("Tokens
" + coins);


        if( bet == 5){
            ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet5","drawable",getPackageName()));
            bettoggle = 0;
        }
        else if (bet == 10){
            ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet10","drawable",getPackageName()));
            bettoggle = 1;
        }
        else if (bet == 50){
            ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet50","drawable",getPackageName()));
            bettoggle = 2;
        }


        SharedPreferences settings =
getSharedPreferences(PREFS_NAME, 0);
        SharedPreferences.Editor editor = settings.edit();
        editor.putInt("coins", coins);
        editor.commit();

    }
}
```

Chapter 7:
Remember to save

It is important that when a user closes the app and then reopens it later, they still have the same amount of coins.

To save a value we first write the following code in section 2:

```
public static final String PREFS_NAME = "MyPrefsFile";
```

Then in the procteced void onCreate method we have the following code:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.newhome);


 SharedPreferences settings = getSharedPreferences(PREFS_NAME,0);
    coins = settings.getInt("coins",coins);

    ((TextView) findViewById(R.id.tokencount)).setText("Tokens " +
coins);


    ImageButton buytokens = (ImageButton)
findViewById(R.id.buytokenbtn);
    buytokens.setOnClickListener(new OnClickListener() {
        @Override
        public void onClick(View v) {
            setContentView(R.layout.activity_main);
        }
    });

}
```

And then after each method that changes the value of the total coins we want to include the following code:

```
SharedPreferences settings = getSharedPreferences(PREFS_NAME, 0);
SharedPreferences.Editor editor = settings.edit();
editor.putInt("coins", coins);
editor.commit();
```

So to recap we do the following:

We declare a file that we want to save.
We then recall the contents of this file when we start our app
And finally we save to this file when the value changes.

Chapter 8
All the code to copy and paste

You are welcome to copy and paste the code below into your MainActivity.java

But remember you will need to change the names of the image references. Also I've left in some other code that allows for the saving of pass codes which I didn't talk about in this book.

You are also going to need to link up the methods to your own User Interface and change the names of the UI elements in my code.

Good luck and thank you for reading this book.

```java
package mjprojects.poker;

import android.content.Intent;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;
import android.support.design.widget.Snackbar;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;
import android.view.MotionEvent;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.TextView;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.ImageView;
import android.view.ViewGroup;
import android.*;


import java.util.Random;

public class MainActivity extends AppCompatActivity {

    //GameVariables
    private int coins = 1000;
    private int jackpots = 0;
    private String message;
    private int randomNum;
    private int count;

    private String first;
    private int firstNum;

    private String second;
    private int secondNum;

    private String third;
    private int thirdNum;

    private String forth;
    private int forthNum;

    private String code;
```

```
EditText mEdit;


private int cardSuit1;
private int cardNum1;

private int cardSuit2;
private int cardNum2;

private int cardSuit3;
private int cardNum3;

private int cardSuit4;
private int cardNum4;

private int cardSuit5;
private int cardNum5;

int high1=0;
int high2=0;
int high3=0;
int high4=0;
int high5=0;


int num1 =1;
int num2 =1;
int num3 =1;
int num4 =1;
int num5 =1;
int num6 =1;
int num7 =1;
int num8 =1;
int num9 =1;
int num10 =1;

int suit1 =1;
int suit2 =1;
int suit3 =1;
int suit4 =1;
int suit5 =1;
int suit6 =1;
int suit7 =1;
int suit8 =1;
int suit9 =1;
int suit10 =1;

int select1 = 0;
int select2 = 0;
int select3 = 0;
int select4 = 0;
int select5 = 0;
```

```
    int toggle = 0;
    int bettoggle =0;
    int cellnumbertoggle =0;
    int bet = 5;
    int winnings;
    int secretNum;
    int secretsSaved = 1;


    String nameofcard1, nameofcard2, nameofcard3, nameofcard4,
nameofcard5, playbuttonmessage;


    String cellnumber;
    int cellnumbersaved;
    float passcodesaved = 1;
    int lastthreedigits;
    int passcode;
    int passcodecheck;



    int pair = 0;
    String result;

    public static final String PREFS_NAME = "MyPrefsFile";


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.newhome);
        //Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        //setSupportActionBar(toolbar);


        SharedPreferences settings =
getSharedPreferences(PREFS_NAME,0);
        SharedPreferences settings2 =
getSharedPreferences(PREFS_NAME,1);
        SharedPreferences settings3 =
getSharedPreferences(PREFS_NAME,2);
        SharedPreferences settings4 =
getSharedPreferences(PREFS_NAME,3);

        cellnumbersaved =
settings2.getInt("cellnumbersaved",cellnumbersaved);
        coins = settings.getInt("coins",coins);
        cellnumbertoggle =
settings3.getInt("cellnumbertoggle",cellnumbertoggle);
        secretsSaved =
settings4.getInt("secretsSaved",secretsSaved);


        ((TextView) findViewById(R.id.tokencount)).setText("Tokens
" + coins);


        ImageButton buytokens = (ImageButton)
findViewById(R.id.buytokenbtn);
        buytokens.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                setContentView(R.layout.activity_main);
            }
        });

    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if
it is present.
        getMenuInflater().inflate(R.menu.menu_main, menu);





        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        // Handle action bar item clicks here. The action bar will
        // automatically handle clicks on the Home/Up button, so
long
        // as you specify a parent activity in
AndroidManifest.xml.
        int id = item.getItemId();

        //noinspection SimplifiableIfStatement
        if (id == R.id.action_settings) {
            return true;
        }

        return super.onOptionsItemSelected(item);
    }
```

```
public void returntogame (View v){
    setContentView(R.layout.newhome);
    //((ImageView)
findViewById(R.id.imageView2)).setImageResource(getResources().get
Identifier("ic_mainmessage2","drawable",getPackageName()));
    toggle = 0;
    bettoggle = 0;
}

    public void gotobuytokens (View v){
        setContentView(R.layout.activity_main);

    }


    public void ChangeBet (View v){

        if (bettoggle == 0){

            bet = 10;
            ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet10","drawable",getPackageName()));
            bettoggle = 1;
        }
        else if (bettoggle == 1){

            bet = 50;
            ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet50","drawable",getPackageName()));
            bettoggle = 2;
        }
        else if (bettoggle == 2){

            bet = 5;
            ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet5","drawable",getPackageName()));
            bettoggle = 0;
        }


    }


    public void Play (View v) {
        // TESTING
      /*
      num1 = 6;
        num2 = 5;
        num3 = 8;
        num4 = 3;
        num5 = 7;

        suit1=1;
        suit2=1;
        suit3=1;
        suit4=1;
        suit5=1;
        */
            if( toggle == 0) {

                playbuttonmessage = "ic_accepthand";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage,"drawable",getPackageName()));
                ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_taporaccept","drawable",getPackageName()));
                bettoggle = 3;


                select1 = 0;
                select2 = 0;
                select3 = 0;
                select4 = 0;
                select5 = 0;

                drawcards();
                displayresults();
                toggle = 3;
                coins = coins - bet;

                ((TextView)
findViewById(R.id.tokencount)).setText("Tokens " + coins+ " +
"+winnings);


            }else
        if(toggle == 1) {

            if (select1 == 1) {

                num1 = num6;
                suit1 = suit6;

            }
            if (select2 == 1) {
                num2 = num7;
                suit2 = suit7;

            }
            if (select3 == 1) {

                num3 = num8;
                suit3 = suit8;
            }
            if (select4 == 1) {

                num4 = num9;
                suit4 = suit9;

            }
            if (select5 == 1) {

                num5 = num10;
                suit5 = suit10;

            }

            toggle = 2;
            displayresults();
            playbuttonmessage = "ic_accepthand";
            ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage,"drawable",getPackageName()));
            ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_acceptwinnings","drawable",getPackageName()));
            bettoggle = 3;

            ((TextView)
findViewById(R.id.tokencount)).setText("Tokens " + coins+" +
"+winnings);

        }else
        if( toggle == 2){

            toggle = 0;

            playbuttonmessage = "ic_playbuttonpoker";
            ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage,"drawable",getPackageName()));

            nameofcard1 = "ic_cardbacknew";
            ((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
```

```
            ((ImageView)
findViewById(R.id.cardface2)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
            ((ImageView)
findViewById(R.id.cardface3)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
            ((ImageView)
findViewById(R.id.cardface4)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
            ((ImageView)
findViewById(R.id.cardface5)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));

            ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_goodluck","drawable",getPackageName()));
            coins = coins + winnings;
            ((TextView)
findViewById(R.id.tokencount)).setText("Tokens " + coins);


            if( bet == 5){
                ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet5","drawable",getPackageName()));
                bettoggle = 0;
            }
            else if (bet == 10){
                ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet10","drawable",getPackageName()));
                bettoggle = 1;
            }
            else if (bet == 50){
                ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet50","drawable",getPackageName()));
                bettoggle = 2;
            }


            SharedPreferences settings =
getSharedPreferences(PREFS_NAME, 0);
            SharedPreferences.Editor editor = settings.edit();
            editor.putInt("coins", coins);
            editor.commit();




        }else
        if (toggle == 3){
            toggle = 0;
            playbuttonmessage = "ic_playbuttonpoker";
            ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage,"drawable",getPackageName()));

            nameofcard1 = "ic_cardbacknew";
            ((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
            ((ImageView)
findViewById(R.id.cardface2)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
            ((ImageView)
findViewById(R.id.cardface3)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
            ((ImageView)
findViewById(R.id.cardface4)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));
            ((ImageView)
findViewById(R.id.cardface5)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));

            ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_goodluck","drawable",getPackageName()));
            coins = coins + winnings;
            ((TextView)
findViewById(R.id.tokencount)).setText("Tokens " + coins);

            if( bet == 5){
                ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet5","drawable",getPackageName()));
            }
            else if (bet == 10){
                ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet10","drawable",getPackageName()));
            }
            else if (bet == 50){
                ((ImageView)
findViewById(R.id.cardface6)).setImageResource(getResources().getI
dentifier("ic_bet50","drawable",getPackageName()));
            }


            SharedPreferences settings =
getSharedPreferences(PREFS_NAME, 0);
            SharedPreferences.Editor editor = settings.edit();
            editor.putInt("coins", coins);
            editor.commit();


        }






    }

    public void CardOneButton (View v){


        if(toggle == 3 || toggle ==1) {

            if (select1 == 0) {

                select1 = 1;
                nameofcard1 = "ic_swap";
                ((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1, "drawable", getPackageName()));

            } else {
                select1 = 0;
                nameofcard1 = "ic_" + suit1 + "" + num1;
                ((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1, "drawable", getPackageName()));
            }

            if (select1 == 1 || select2 == 1 || select3 == 1 ||
select4 == 1 || select5 == 1) {
                toggle = 1;
                playbuttonmessage = "ic_swapbutton";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
            } else {
                toggle = 3;
                playbuttonmessage = "ic_accepthand";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
            }
        }
    }

    public void CardTwoButton (View v){


        if(toggle == 3 || toggle ==1) {

            if (select2 == 0) {

                select2 = 1;
                nameofcard2 = "ic_swap";
                ((ImageView)
findViewById(R.id.cardface2)).setImageResource(getResources().getI
dentifier(nameofcard2, "drawable", getPackageName()));

            } else {
                select2 = 0;
                nameofcard2 = "ic_" + suit2 + "" + num2;
                ((ImageView)
findViewById(R.id.cardface2)).setImageResource(getResources().getI
dentifier(nameofcard2, "drawable", getPackageName()));
            }

            if (select1 == 1 || select2 == 1 || select3 == 1 ||
select4 == 1 || select5 == 1) {
                toggle = 1;
                playbuttonmessage = "ic_swapbutton";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
            } else {
                toggle = 3;
                playbuttonmessage = "ic_accepthand";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
            }
        }

    }

    public void CardThreeButton (View v){

        if(toggle == 3 || toggle ==1) {

            if (select3 == 0) {

                select3 = 1;
                nameofcard3 = "ic_swap";
                ((ImageView)
findViewById(R.id.cardface3)).setImageResource(getResources().getI
dentifier(nameofcard3, "drawable", getPackageName()));
```

```
            } else {
                select3 = 0;
                nameofcard3 = "ic_" + suit3 + "" + num3;
                ((ImageView)
findViewById(R.id.cardface3)).setImageResource(getResources().getI
dentifier(nameofcard3, "drawable", getPackageName()));
            }

            if (select1 == 1 || select2 == 1 || select3 == 1 ||
select4 == 1 || select5 == 1) {
                toggle = 1;
                playbuttonmessage = "ic_swapbutton";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
            } else {
                toggle = 3;
                playbuttonmessage = "ic_accepthand";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
            }
        }
    }

    public void CardFourButton (View v){


        if(toggle == 3 || toggle ==1) {


            if (select4 == 0) {

                select4 = 1;
                nameofcard4 = "ic_swap";
                ((ImageView)
findViewById(R.id.cardface4)).setImageResource(getResources().getI
dentifier(nameofcard4, "drawable", getPackageName()));

            } else {
                select4 = 0;
                nameofcard4 = "ic_" + suit4 + "" + num4;
                ((ImageView)
findViewById(R.id.cardface4)).setImageResource(getResources().getI
dentifier(nameofcard4, "drawable", getPackageName()));
            }

            if (select1 == 1 || select2 == 1 || select3 == 1 ||
select4 == 1 || select5 == 1) {
                toggle = 1;
                playbuttonmessage = "ic_swapbutton";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
            } else {
                toggle = 3;
                playbuttonmessage = "ic_accepthand";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
            }

        }
    }

    public void CardFiveButton (View v){


        if(toggle == 3 || toggle ==1) {
            if (select5 == 0) {

                select5 = 1;
                nameofcard5 = "ic_swap";
                ((ImageView)
findViewById(R.id.cardface5)).setImageResource(getResources().getI
dentifier(nameofcard5, "drawable", getPackageName()));

            } else {
                select5 = 0;
                nameofcard5 = "ic_" + suit5 + "" + num5;
                ((ImageView)
findViewById(R.id.cardface5)).setImageResource(getResources().getI
dentifier(nameofcard5, "drawable", getPackageName()));
            }

            if (select1 == 1 || select2 == 1 || select3 == 1 ||
select4 == 1 || select5 == 1) {
                toggle = 1;
                playbuttonmessage = "ic_swapbutton";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
            } else {
                toggle = 3;
                playbuttonmessage = "ic_accepthand";
                ((ImageView)
findViewById(R.id.mainplay)).setImageResource(getResources().getId
entifier(playbuttonmessage, "drawable", getPackageName()));
            }
        }
    }
    public void swapCard ( int oldnum, int oldsuit, int newnum,
int newsuit){


        oldnum = newnum;
        oldsuit = newsuit;
    }

    public void drawcards(){

        //First Card
        Random s1 = new Random();
        Random n1 = new Random();
        suit1 = s1.nextInt(5 - 1) + 1;
        num1 = n1.nextInt(14 - 1) + 1;
        cardSuit1 = suit1;
        cardNum1 = num1;

        nameofcard1 = "ic_"+suit1+""+num1;
        ((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));


        //Second Card
        Random s2 = new Random();
        Random n2 = new Random();
        suit2 = s2.nextInt(5 - 1) + 1;
        num2 = n2.nextInt(14 - 1) + 1;

        cardSuit2 = suit2;
        cardNum2 = num2;

        while (suit1 == suit2 && num1 == num2) {
            n2 = new Random();
            num2 = n2.nextInt(14 - 1) + 1;
            cardSuit2 = suit2;
            cardNum2 = num2;
        }

        nameofcard2 = "ic_"+suit2+""+num2;
        ((ImageView)
findViewById(R.id.cardface2)).setImageResource(getResources().getI
dentifier(nameofcard2,"drawable",getPackageName()));


        //Third Card
        Random s3 = new Random();
        Random n3 = new Random();
        suit3 = s3.nextInt(5 - 1) + 1;
        num3 = n3.nextInt(14 - 1) + 1;
        cardSuit3 = suit3;
        cardNum3 = num3;

        while( (suit3 == suit1 && num3 == num1) || (suit3 == suit2
&& num3 == num2)){

            s3 = new Random();
            n3 = new Random();
            suit3 = s3.nextInt(5 - 1) + 1;
            num3 = n3.nextInt(14 - 1) + 1;

            cardSuit3 = suit3;
            cardNum3 = num3;
        }

        nameofcard3 = "ic_"+suit3+""+num3;
        ((ImageView)
findViewById(R.id.cardface3)).setImageResource(getResources().getI
dentifier(nameofcard3,"drawable",getPackageName()));


        //Forth Card
        Random s4 = new Random();
        Random n4 = new Random();
        suit4 = s4.nextInt(5 - 1) + 1;
        num4 = n4.nextInt(14 - 1) + 1;


        cardSuit4 = suit4;
        cardNum4 = num4;

        while( (suit4 == suit1 && num4 == num1) || (suit4 == suit2
&& num4 == num2) || (suit4 == suit3 && num4 == num3)){

            s4 = new Random();
            n4 = new Random();
            suit4 = s4.nextInt(5 - 1) + 1;
            num4 = n4.nextInt(14 - 1) + 1;

            cardSuit4 = suit4;
            cardNum4 = num4;
        }
        nameofcard4 = "ic_"+suit4+""+num4;
        ((ImageView)
findViewById(R.id.cardface4)).setImageResource(getResources().getI
dentifier(nameofcard4,"drawable",getPackageName()));


        //Fifth Card
        Random s5 = new Random();
        Random n5 = new Random();
        suit5 = s5.nextInt(5 - 1) + 1;
        num5 = n5.nextInt(14 - 1) + 1;


        cardSuit5 = suit5;
        cardNum5 = num5;

        while( (suit5 == suit1 && num5 == num1) || (suit5 == suit2
&& num5 == num2) || (suit5 == suit3 && num5 == num3) || (suit5 ==
suit4 && num5 == num4)){

            s5 = new Random();
            n5 = new Random();
            suit5 = s5.nextInt(5 - 1) + 1;
            num5 = n5.nextInt(14 - 1) + 1;

            cardSuit5 = suit5;
            cardNum5 = num5;
        }

        nameofcard5 = "ic_"+suit5+""+num5;
        ((ImageView)
findViewById(R.id.cardface5)).setImageResource(getResources().getI
dentifier(nameofcard5,"drawable",getPackageName()));





        // THE EXTRA CARDS
        // SIXTH EXTRA CARD
        Random s6 = new Random();
        Random n6 = new Random();
        suit6 = s6.nextInt(5 - 1) + 1;
        num6 = n6.nextInt(14 - 1) + 1;

        while( (suit6 == suit1 && num6 == num1) || (suit6 == suit2
&& num6 == num2) || (suit6 == suit3 && num6 == num3) || (suit6 ==
suit4 && num6 == num4) || (suit6 == suit5 && num6 == num5)){

            s6 = new Random();
            n6 = new Random();
            suit6 = s6.nextInt(5 - 1) + 1;
            num6 = n6.nextInt(14 - 1) + 1;

        }

        // SEVENTH EXTRA CARD
        Random s7 = new Random();
        Random n7 = new Random();
        suit7 = s7.nextInt(5 - 1) + 1;
        num7 = n7.nextInt(14 - 1) + 1;
```

```
    while( (suit7 == suit1 && num7 == num1) || (suit7 == suit2
&& num7 == num2) || (suit7 == suit3 && num7 == num3) || (suit7 ==
suit4 && num7 == num4) || (suit7 == suit5 && num7 == num5) ||
(suit7 == suit6 && num7 == num6)){

        s7 = new Random();
        n7 = new Random();
        suit7 = s7.nextInt(5 - 1) + 1;
        num7 = n7.nextInt(14 - 1) + 1;
    }

    // EIGHTH EXTRA CARD
    Random s8 = new Random();
    Random n8 = new Random();
    suit8 = s8.nextInt(5 - 1) + 1;
    num8 = n8.nextInt(14 - 1) + 1;

    while( (suit8 == suit1 && num8 == num1) || (suit8 == suit2
&& num8 == num2) || (suit8 == suit3 && num8 == num3) || (suit8 ==
suit4 && num8 == num4) || (suit8 == suit5 && num8 == num5) ||
(suit8 == suit6 && num8 == num6) || (suit8 == suit7 && num8 ==
num7)){

        s8 = new Random();
        n8 = new Random();
        suit8 = s8.nextInt(5 - 1) + 1;
        num8 = n8.nextInt(14 - 1) + 1;
    }

    // NINTH EXTRA CARD
    Random s9 = new Random();
    Random n9 = new Random();
    suit9 = s8.nextInt(5 - 1) + 1;
    num9 = n8.nextInt(14 - 1) + 1;

    while( (suit9 == suit1 && num9 == num1) || (suit9 == suit2
&& num9 == num2) || (suit9 == suit3 && num9 == num3) || (suit9 ==
suit4 && num9 == num4) || (suit9 == suit5 && num9 == num5) ||
(suit9 == suit6 && num9 == num6) || (suit9 == suit7 && num9 ==
num7) || (suit9 == suit8 && num9 == num8)){

        s9 = new Random();
        n9 = new Random();
        suit9 = s9.nextInt(5 - 1) + 1;
        num9 = n9.nextInt(14 - 1) + 1;
    }


    // TENTH EXTRA CARD
    Random s10 = new Random();
    Random n10 = new Random();
    suit10 = s10.nextInt(5 - 1) + 1;
```

```
        num10 = n10.nextInt(14 - 1) + 1;

        while( (suit10 == suit1 && num10 == num1) || (suit10 == 
suit2 && num10 == num2) || (suit10 == suit3 && num10 == num3) || 
(suit10 == suit4 && num10 == num4) || (suit10 == suit5 && num10 == 
num5) || (suit10 == suit6 && num10 == num6) || (suit10 == suit7 && 
num10 == num7) || (suit10 == suit8 && num10 == num8) || (suit10 == 
suit9 && num10 == num9)){

            s10 = new Random();
            n10 = new Random();
            suit10 = s10.nextInt(5 - 1) + 1;
            num10 = n10.nextInt(14 - 1) + 1;
        }


    }

    public void displayresults(){


        int[] input = {num1, num2, num3, num4, num5};
        bubble_srt(input);

        pair = 0;
        bubble_match2(input);


        nameofcard1 = "ic_"+suit1+""+num1;
        ((ImageView)
findViewById(R.id.cardface1)).setImageResource(getResources().getI
dentifier(nameofcard1,"drawable",getPackageName()));

        nameofcard2 = "ic_"+suit2+""+num2;
        ((ImageView)
findViewById(R.id.cardface2)).setImageResource(getResources().getI
dentifier(nameofcard2,"drawable",getPackageName()));

        nameofcard3 = "ic_"+suit3+""+num3;
        ((ImageView)
findViewById(R.id.cardface3)).setImageResource(getResources().getI
dentifier(nameofcard3,"drawable",getPackageName()));

        nameofcard4 = "ic_"+suit4+""+num4;
        ((ImageView)
findViewById(R.id.cardface4)).setImageResource(getResources().getI
dentifier(nameofcard4,"drawable",getPackageName()));

        nameofcard5 = "ic_"+suit5+""+num5;
        ((ImageView)
findViewById(R.id.cardface5)).setImageResource(getResources().getI
dentifier(nameofcard5,"drawable",getPackageName()));
```

```
        if( suit1 == suit2 && suit2 == suit3 && suit3 == suit4 &&
suit4 == suit5){

            if ( high1 == (high2 +1) && high1 == (high3 + 2) &&
high1 == (high4 + 3) && (high1 == (high5 + 4) || high1 == (high5 +
12))){

                //STRAIGHT FLUSH RESULT
                result = "Straight Flush";
                winnings = 2500 * (bet/5);
                ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_straightflush","drawable",getPackageName()));


            }else{
                //FLUSH RESULT
                result = "Flush";
                winnings = 450 * (bet/5);
                ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_flush","drawable",getPackageName()));
            }

        }else{
            if (high1 == (high2 +1) && high1 == (high3 + 2) &&
high1 == (high4 + 3) && (high1 == (high5 + 4) || high1 == (high5 +
12))){
                //STRAIGHT RESULT
                result = "Straight";
                winnings = 100 * (bet/5);
                ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_straight","drawable",getPackageName()));
            } else{
                //HIGH CARD RESULT
                result = "High Card";
                winnings = 0 * (bet/5);
                ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_highcard","drawable",getPackageName()));
            }
        }

        if(pair == 1){
            //ONE PAIR RESULT
            result = "One Pair";
            ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_1pair","drawable",getPackageName()));
            winnings = 2 * (bet/5);
        }
        if (pair == 2){
            //TWO PAIR RESULT
            result = "Two Pair";
            winnings = 20 * (bet/5);
            ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_2pairs","drawable",getPackageName()));
        }
        if (pair == 3){
            //THREE OF A KIND RESULT
            result = "Three of a Kind";
            winnings = 40 * (bet/5);
            ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_3ofakind","drawable",getPackageName()));
        }
        if (pair == 4){
            //FULL HOUSE RESULT
            result = " Full House";
            winnings = 500 * (bet/5);
            ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_fullhousebar","drawable",getPackageName()));
        }
        if (pair == 6){
            //FOUR OF A KIND RESULT
            result = "Four of a Kind";
            winnings = 1000 * (bet/5);
            ((ImageView)
findViewById(R.id.resultimage)).setImageResource(getResources().ge
tIdentifier("ic_4ofakind","drawable",getPackageName()));
        }

        //((TextView) findViewById(R.id.tokenswon)).setText(" " +
pair);

    }


    public void checkPasscode(View v){




        EditText cellnumbervalue =
(EditText)findViewById(R.id.entercell);
        EditText codevalue =
(EditText)findViewById(R.id.entercode);

if( cellnumbervalue.getText().toString().equals("") ||
codevalue.getText().toString().equals("")){
    cellnumber = "12345";
    passcode = 12345;
}else {

    cellnumber = cellnumbervalue.getText().toString();
    passcode = Integer.parseInt(codevalue.getText().toString());
}

        if(cellnumber.length() > 3){
            lastthreedigits =
Integer.parseInt(cellnumber.substring(cellnumber.length()-3));
        }
        else{
            lastthreedigits = 123;
        }

      passcodecheck = passcode - 23 - lastthreedigits;

        secretNum = (passcodecheck/11) - 909;





        if(   ((passcodecheck%11) == 0)  &&
( (cellnumbertoggle==0)||((cellnumbersaved == lastthreedigits) &&
((secretsSaved%secretNum) != 0 ) )) ){

            //SAVE COINS
            coins = coins + 500;
            SharedPreferences settings =
getSharedPreferences(PREFS_NAME, 0);
            SharedPreferences.Editor editor = settings.edit();
            editor.putInt("coins", coins);
            editor.commit();

            //SAVE CELLNUMBER
            cellnumbersaved = lastthreedigits;
            //cellnumbersaved = 0;
            SharedPreferences settings2 =
getSharedPreferences(PREFS_NAME,1);
            SharedPreferences.Editor editor2 = settings2.edit();
            editor2.putInt("cellnumbersaved", cellnumbersaved);
            editor2.commit();
            //SAVE CELLNUMBER TOGGLE
            cellnumbertoggle = 1;
            SharedPreferences settings3 =
getSharedPreferences(PREFS_NAME,2);
            SharedPreferences.Editor editor3 = settings3.edit();
            editor3.putInt("cellnumbertoggle", cellnumbertoggle);
            editor3.commit();


            //SAVE PASSCODES SECRETS
            //secretsSaved = 1;
            //secretNum = 1;
            secretsSaved = secretsSaved * secretNum;
            SharedPreferences settings4 =
getSharedPreferences(PREFS_NAME,3);
            SharedPreferences.Editor editor4 = settings4.edit();
            editor4.putInt("secretsSaved", secretsSaved);
            editor4.commit();

            ((TextView)
findViewById(R.id.textView)).setText("Last3 " + cellnumbersaved +
" Toggle "+ cellnumbertoggle + " pass " + secretsSaved);





            ((ImageView)
findViewById(R.id.imageView2)).setImageResource(getResources().get
Identifier("ic_codeaccepted","drawable",getPackageName()));
        }
        else{
            ((TextView)
findViewById(R.id.textView)).setText("Last3 " + cellnumbersaved +
" Toggle "+ cellnumbertoggle + " pass " + secretsSaved);
            ((ImageView)
findViewById(R.id.imageView2)).setImageResource(getResources().get
Identifier("ic_failmessage","drawable",getPackageName()));
        }


    }


    // logic to sort the elements
    public void bubble_srt(int array[]) {
        int n = array.length;
        int k;
        for (int m = n; m >= 0; m--) {
            for (int i = 0; i < n - 1; i++) {
```

```
            k = i + 1;
            if (array[i] > array[k]) {
                swapNumbers(i, k, array);
            }
        }
    }
    high5 = array[0];
    high4 = array[1];
    high3 = array[2];
    high2 = array[3];
    high1 = array[4];

}

public void bubble_match(int arraylist[]){

    int nbm = arraylist.length;
    int kbm;

    for (int mbm = nbm; mbm >= 0; mbm--){


        for(int j = 0; j < nbm - 1; j++){
            kbm = j + 1;

            if(arraylist[j] == arraylist[kbm]){
                pair = pair + 1;
            }
        }
    }
}

public void bubble_match2 (int array[]){

    for( int n = 1; n <= 4; n++) {
        if (array[0] == array[n]) {
            pair = pair + 1;
        }
    }

    for( int n = 2; n <= 4; n++) {
        if (array[1] == array[n]) {
            pair = pair + 1;
        }
    }

    for( int n = 3; n <= 4; n++) {
        if (array[2] == array[n]) {
            pair = pair + 1;
        }
    }
```

```
        for( int n = 4; n <= 4; n++) {
            if (array[3] == array[n]) {
                pair = pair + 1;
            }
        }




    }


    private void swapNumbers(int i, int j, int[] array) {

        int temp;
        temp = array[i];
        array[i] = array[j];
        array[j] = temp;
    }

}
```