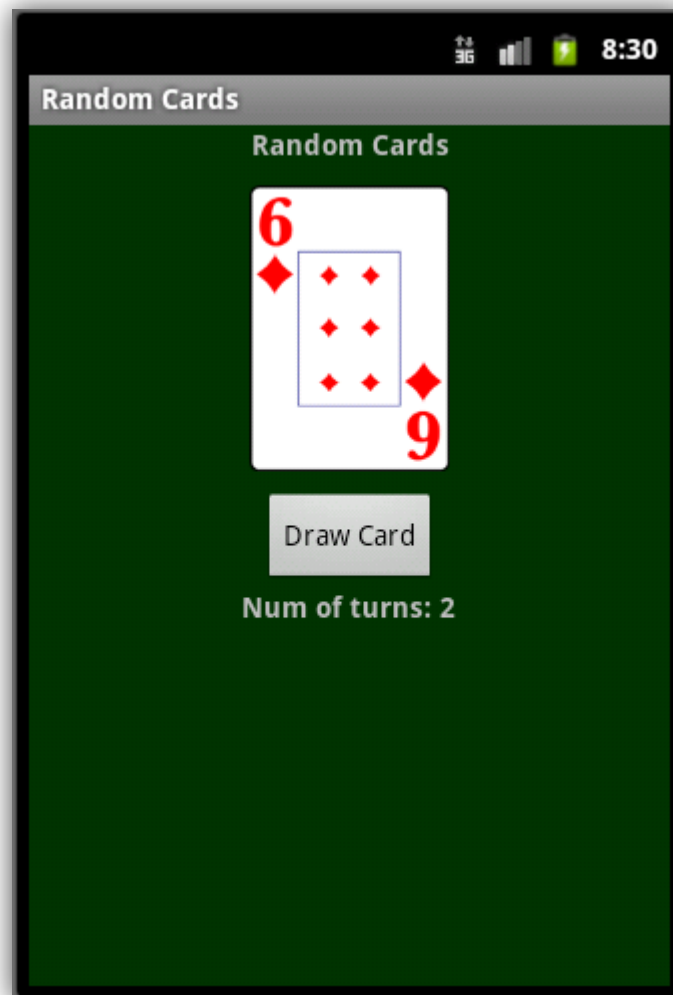


# Assignment # 2 – Random Playing Cards

---

In this exercise, you'll be building an application that allows the user to draw a random card from a deck. For each deck played, the user is not allowed to draw the same card twice, so in essence, once drawn, you are discarding the card in a different pile.

Your application will look like this:



The 52 playing cards, and an image of the card background are available in cards.zip file accompanying this lesson.

Use the blue\_back\_vert.png file as the default image in your application before the user has a chance to select any card.

Use #003300 as the background color for the linear layout used here in the application.

To help you get started, here are some tips and code samples you should work with while tackling this project.

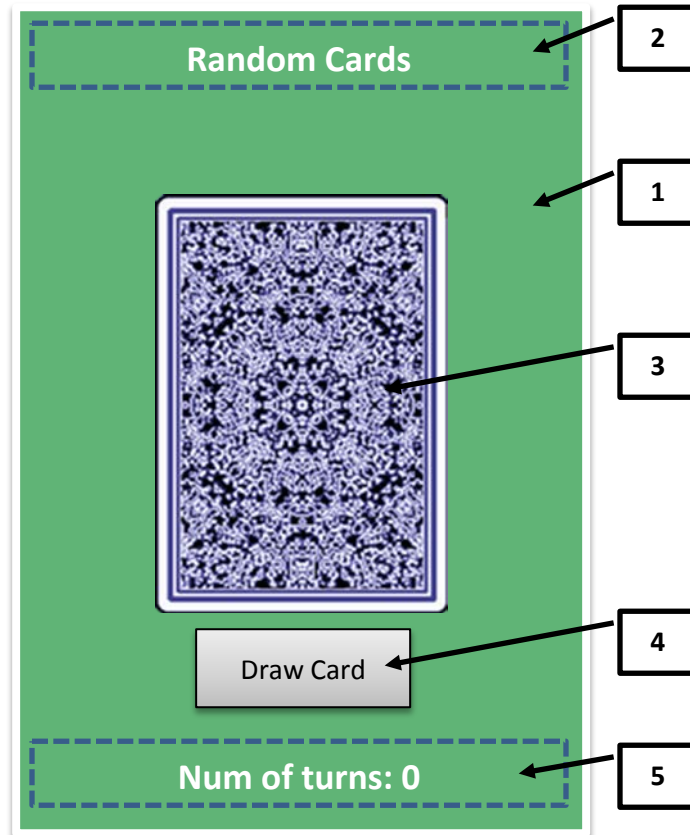
1 – You’ll need two global variables in this project. One to keep track of the card count, and the second to keep track of the cards in the discarded pile. Call the first variable “cardcount” and initialize it to zero, and the second an array of integers, called “usedcards”. You’ll need to initialize the usedcards array within your onCreate event to 52 elements (0 to 51).

2 – You’ll need a second array of cards, which represents the deck that you draw from. Because the imported images (card1.png through card52.png) are written into the R.drawable class as integers, we can create an array of int, and specify the R.drawable cards as each element. Name the array “cards”, and initialize it within your onCreate event. To save you some time, use the code below:

```
final int[] cards = {  
    R.drawable.card1, R.drawable.card2, R.drawable.card3, R.drawable.card4,  
        R.drawable.card5, R.drawable.card6, R.drawable.card7,  
        R.drawable.card8, R.drawable.card9, R.drawable.card10,  
        R.drawable.card11, R.drawable.card12, R.drawable.card13,  
        R.drawable.card14, R.drawable.card15, R.drawable.card16,  
        R.drawable.card17, R.drawable.card18, R.drawable.card19,  
        R.drawable.card20, R.drawable.card21, R.drawable.card22,  
        R.drawable.card23, R.drawable.card24, R.drawable.card25,  
        R.drawable.card26, R.drawable.card27, R.drawable.card28,  
        R.drawable.card29, R.drawable.card30, R.drawable.card31,  
        R.drawable.card32, R.drawable.card33, R.drawable.card34,  
        R.drawable.card35, R.drawable.card36, R.drawable.card37,  
        R.drawable.card38, R.drawable.card39, R.drawable.card40,  
        R.drawable.card41, R.drawable.card42, R.drawable.card43,  
        R.drawable.card44, R.drawable.card45, R.drawable.card46,  
        R.drawable.card47, R.drawable.card48, R.drawable.card49,  
        R.drawable.card50, R.drawable.card51, R.drawable.card52  
};
```

Be sure to import the images into your **res/drawable-hdpi** folder first.

3 – Your layout will consist of the following elements:



1 – LinearLayout – **Background** set to #003300.

2 – TextView – **Text** set to “Random Cards”, **Text Style** set to **bold**, **Gravity** set to **center\_horizontal**.

3 – ImageView – **ID** set to @+id/imgCard, **Src** set to @drawable/blue\_back\_vert, **Layout gravity** set to **center\_horizontal**, **Layout margin** set to 10dp.

4 – Button – **ID** set to @+id/btnDraw, **Text** set to “Draw Card”, **Layout gravity** set to **center\_horizontal**.

5 – TextView – **ID** set to @+id/txtTurns, **Text style** set to **bold**, **Layout gravity** set to **center\_horizontal**, and **Text** property is blank.

4 – To generate random numbers, we can use the Math.random() function. Since we need to draw a single card out of a deck of 52, we need a random number between 0 and 51. Here’s the code that will accomplish that for you (note that CardNum is an INT variable):

```
CardNum = (int) (Math.random() * (52)); // 0 to 51
```

This is a challenging project. Take your time and plan it correctly. Write out pseudo-code. This can help you plan your routines, and loops.

If you find that the tips here help, use them in your code. If, however, you feel you have a better approach, be sure to use that instead!