6. **PROGRAMMING ASSIGNMENT 6**

Read Deitel: Chapter 25; Liang: Chapter 14-16
Programming: Name your file for PA#6: **P6.java**
DUE: Saturday, August 29, 2015 @ 6am

You will write a JavaFX UI program with javadoc comments to display a deck of playing card images, shuffle, sort in ascending and descending order using Buttons. Layout Panes include: GridPane, BorderPane, and HBox.
Random generates random numbers for the shuffle using the System.currentTimeMillis() seed. javadoc comments will be written only in PA#6. Work in a directory named "PA6". NOTE: Card images are in: "/home/linux/ieng6/cs11v/public/image/card/". NO error checking of input data because there are only buttons for user interface.

```
/**
 * public void start(Stage primaryStage)  - Initialize JavaFX application
 * public void ascendSort( int [] a)      - Sorts in ascending order
 * public void descendSort( int [] a)     - Sorts in descending order
 * public void shuffle( int [] a)         - Random swap of cards
 * public void swap( int j, int k)        - Swap of class array
 * public void main( String []args )      - Not needed for command line execution
 */
```

```
 public void start(Stage primaryStage)
 {
    // …
    for ( i = k = 0; i < ASIZE; i++ )    // ASIZE is 54
      aCardDeck[i] /*…*/ ;               // Populate with elements values 1-54

    GridPane gPane = new GridPane();     // Instantiate Grid for cards in rows/cols

    for ( i = k = 0 ; i < ROWS && k < ASIZE; i++)   // 6 rows, 9 columns
      for (j = 0; j < COLS ; j++)
        gPane.add
         (new ImageView("file:/home/linux/ieng6/cs11v/public/image/card/" +
                        aCardDeck[k++] + ".png"),j,i);

    // Create a button to shuffle
    Button btShuffle = new Button("Shuffle");
    btShuffle.setOnAction
    ( e ->                                    // Lamda Event Handler
     {                                       // Like anonymous inner class
      int r, c, n;
      shuffle();                            // Random shuffle
      gPane.getChildren().clear();
      for ( r = n = 0; r < ROWS && n < ASIZE; r++)
        for (c = 0; c < COLS ; c++)
         gPane.add(new ImageView("file:/home/linux/ieng6/cs11v/public/image/card/" +
                                 aCardDeck[n++] + ".png"),c,r);
     }
    );

    HBox hBox = new HBox(7);                  // Row of buttons
    hBox.getChildren().add( btShuffle );      // Add button to box
       // Repeat for 6 other buttons
    BorderPane pane = new BorderPane();
    pane.setCenter(gPane);                    // Layout rows of cards in center
    pane.setBottom(hBox);                     // Layout buttons at bottom
    BorderPane.setAlignment(hBox, Pos.CENTER);

    Scene scene = new Scene(pane, 650, 600);  // Create scene, place in stage
    primaryStage.setTitle("P6");              // Set the stage title
    primaryStage.setScene(scene);             // Place the scene in the stage
    primaryStage.show();                      // Display the stage
 }

 public static void main(String[] args)
 {
     launch(args);
 }
}
```

a) An ascending sorted order of 6 rows and 9 columns is displayed upon start.
b) One private instance array is allocated. ASIZE is a final constant with a value of 54.

```
private int aCardDeck[] = new int[ASIZE]; // 54 cards in a deck (52+2 jokers)
```

c) Use final constants.
d) In **start()**, as given above initializes the application and starts execution.

e) **shuffle()** will index through each element in the deck and swap positions with another element chosen at random in the same deck of cards.
e.g. deck[0] is indexed,
"14" is randomly generated,
deck[0] is swapped with deck[14]

For random number generation:

- **import java.util.Random;** // Insert at top of file

Create a local variable, Random object, and set the seed with current time in milliseconds.

- **Random rand = new Random();** // local variable in **shuffle()**
- **seed = System.currentTimeMillis();** // Current time milliseconds type long
- **rand.setSeed(seed);** // Assign seed to random object
- **for(…)**
  r = rand.nextInt(ASIZE); // Generate next int in range 0-53, swap

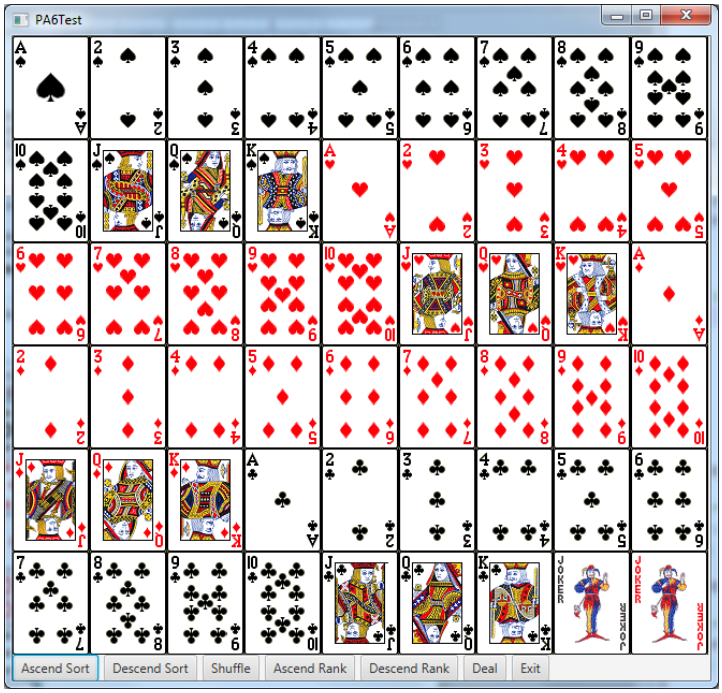
f) Compile your program as in previous applications.
A new window appears displaying your card deck images!
VERIFY your picture is VISIBLY DISPLAYED on EBU-3B #B260 computer monitor screens. IF a picture is NOT seen by grader, it will NOT be awarded any points.

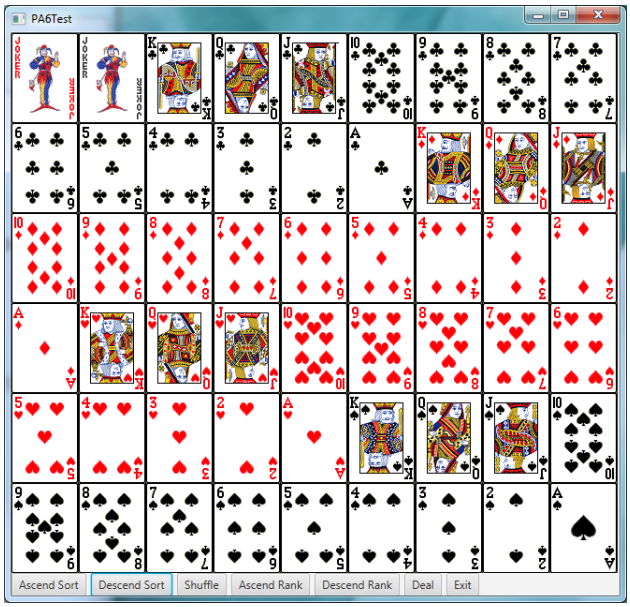
g) To produce the resulting javadoc html files, add these tags: "@param @author @version". These @see tags will allow a hyperlink to the Java 1.8 API and the JavaFX API.
" @see JavaFX docs "
" @see Java 1.8 API "
At the Unix command line, type: **javadoc –author – private – version P6.java**

After successful javadoc execution in #B260 Linux, click on "cs11v home" folder icon on desktop. Click open your PA6 folder. Click on the P6.html icon for javadoc result.

h) Test your program with the "PA/PA6Test.class" file on ieng6.
i) Submit your program for grading with the command, " **bundleP6** ".

PA#6 SAMPLE OUTPUT:





PA#6 SAMPLE INPUT:

```
Click on DescendSort
Click on Shuffle
Click on Deal 2 hands
Click on DescendRankSort
Click on Exit
```

PA#6 SAMPLE OUTPUT: