

Code to add to your Card, Hand and PokerHand classes for PokerSolitaire

Card class Code:

- 1) You will need to add code to keep track of an image for each Card
Add the following code into the variable section of the Card class:

```
private final static Image background = new
    ImageIcon("images\\redback.png").getImage();
public final static int WIDTH = background.getWidth(null);
public final static int HEIGHT = background.getHeight(null);
private Image image;
```

- 2) Add the following code to the Card constructor (the one with 2 integer parameters) to load up each Card's image. This code should go after you have set the Card's rank and suit.

```
// Load up the appropriate image file for this card
String imageFileName = "" + " dchs".charAt(suit) + rank + ".png";
imageFileName = "images\\" + imageFileName;
image = new ImageIcon(imageFileName).getImage();
```

- 3) You will also need to add a draw method to your Card class:

```
/**
 * Draws a card in a Graphics context
 * @param g Graphics to draw the card in
 */
public void draw(Graphics g)
{
    if (isFaceUp)
        g.drawImage(image, x, y, null);
    else
        g.drawImage(background, x, y, null);
}
```

- 4) To keep track of a Card's position, height and width you could add some new instance variables and methods but an easier way is to have your Card class extend the Rectangle class. This should be your new Card heading:

```
public class Card extends Rectangle implements Comparable<Card>
```

- 5) Since your Card class extends Rectangle, you will need to add the following code to your Card constructor. Initially all Cards will have position (0, 0).

Add this code at the top of this constructor:

```
super(0, 0, 0, 0);
```

Add this code after you load up the Card's image:

```
// Set the size of the card based on the image size
setSize(image.getWidth(null), image.getHeight(null));
```

- 6) You will inherit many new methods from the Rectangle class but one method used to drag Cards in PokerSolitaire, but not included in Rectangle, will need to be added. This code calls the translate method from the Rectangle class.

```
public void move (Point initialPos, Point finalPos)
{
    translate (finalPos.x - initialPos.x, finalPos.y - initialPos.y);
}
```

Hand class Code:

- 1) You will need to include code to draw each Hand. Add the following draw method to the Hand class.

```
/**
 * Displays the Cards in this Hand
 * @param g Graphics context to display the deck
 */
public void draw(Graphics g)
{
    for (Card next : hand)
        next.draw(g);
}
```

- 2) The following code will be used to remove Cards from the Hand. This code is used when dragging and dropping Cards. Add it to your Hand class.

```
public void removeCard(Card card)
{
    hand.remove(card);
}
```

- 3) Finally, in order to find a selected Card at a given point, we will include the following code that searches a Hand to find a Card that contains the given point. Add it to your Hand class.

```
public Card getCardAt(Point point)
{
    for (Card next : hand)
        if (next.contains(point))
            return next;
    return null;
}
```

PokerHand class Code:

- 1) You will need to add the following code to your PokerHand class to get the score of each hand. Notice that the points for a straight are higher than the points for a flush or a full house, since it is usually harder to get a straight in Poker Solitaire.

```
static final int[] SCORES = { 0, 1, 3, 6, 12, 5, 10, 16, 30, 50 };

public int getScore()
{
    return SCORES[getType()];
}
```

Note: The above scoring system is for the English version of the game that reflects the difficulty of getting each type of Hand in Poker Solitaire. The American version of the scoring system that reflects the order of normal poker hands is given below:

```
static final int[] SCORES = { 0, 2, 5, 10, 15, 20, 25, 50, 75, 100 };
```