Intro to Java Week 6 Coding Assignment

Points possible: 70

Category	Criteria	% of Grade
Functionality	Does the code work?	25
Organization	Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear.	25
Creativity	Student solved the problems presented in the assignment using creativity and out of the box thinking.	25
Completeness	All requirements of the assignment are complete.	25

Instructions: In Eclipse, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your Java project code, to the repository. Lastly, in the Learning Management System, click the "Add Submission" button and paste the URL to your GitHub repository.

Coding Steps:

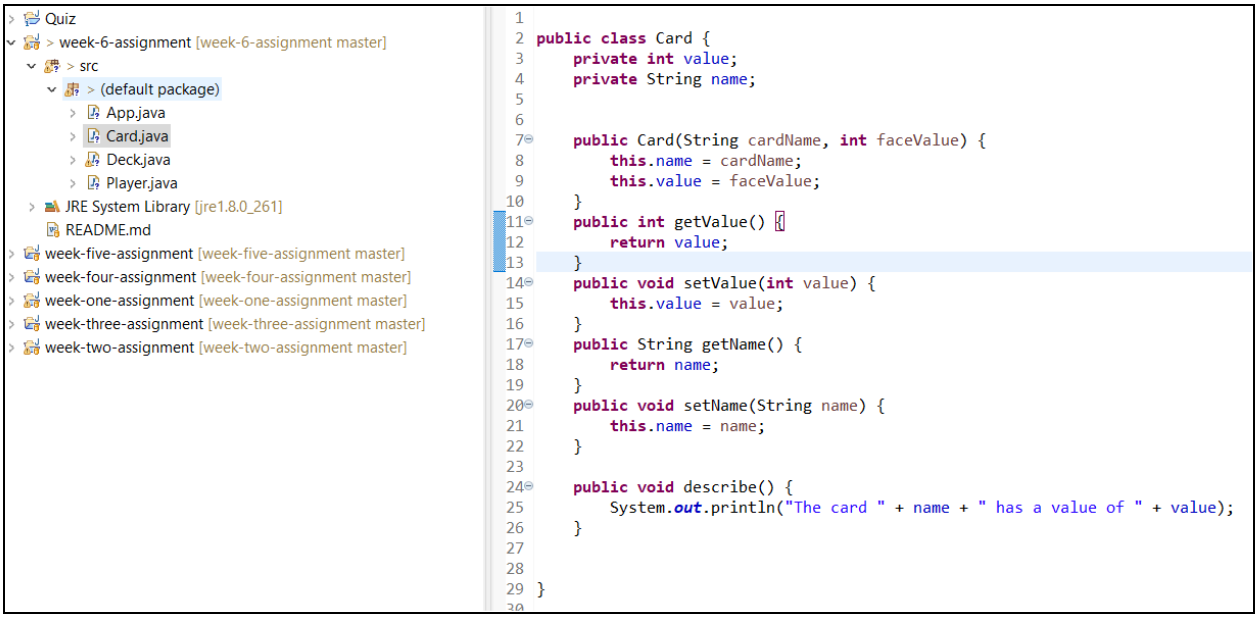
For the final project you will be creating an automated version of the classic card game *WAR*.

1. Create the following classes.
    a. Card
        i. Fields
            1. **value** (contains a value from 2-14 representing cards 2-Ace)
            2. **name** (e.g. Ace of Diamonds, or Two of Hearts)
        ii. Methods
            1. Getters and Setters
            2. **describe** (prints out information about a card)
    b. Deck
        i. Fields
            1. **cards** (List of Card)
        ii. Methods
            1. **shuffle** (randomizes the order of the cards)
            2. **draw** (removes and returns the top card of the Cards field)

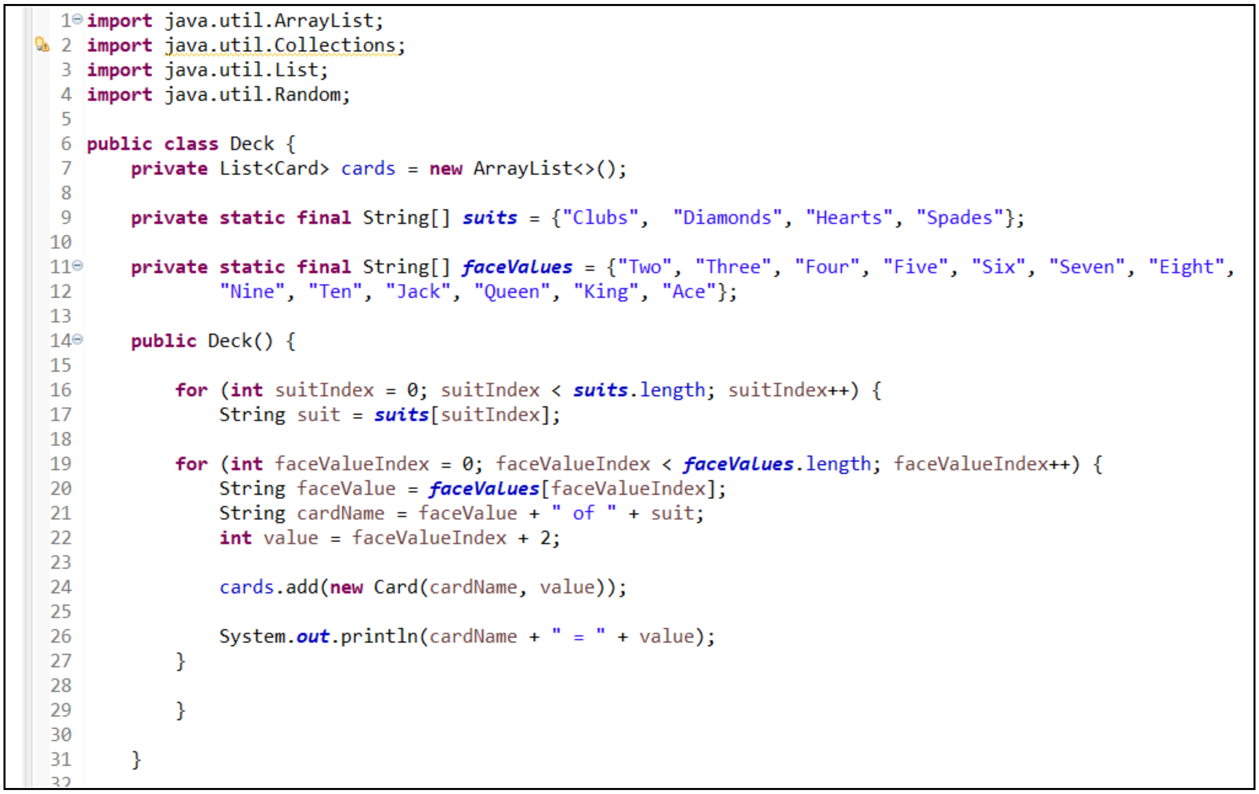
            3. In the constructor, when a new Deck is instantiated, the Cards field should be populated with the standard 52 cards.
    c. Player
        i. Fields
            1. **hand** (List of Card)
            2. **score** (set to 0 in the constructor)
            3. **name**
        ii. Methods
            1. **describe** (prints out information about the player and calls the describe method for each card in the Hand List)
            2. **flip** (removes and returns the top card of the Hand)
            3. **draw** (takes a Deck as an argument and calls the draw method on the deck, adding the returned Card to the hand field)
            4. **incrementScore** (adds 1 to the Player's score field)

2. Create a class called App with a main method.
3. Instantiate a Deck and two Players, call the shuffle method on the deck.
4. Using a traditional for loop, iterate 52 times calling the Draw method on the other player each iteration using the Deck you instantiated.
5. Using a traditional for loop, iterate 26 times and call the flip method for each player.
    a. Compare the value of each card returned by the two player's flip methods. Call the incrementScore method on the player whose card has the higher value.
6. After the loop, compare the final score from each player.
7. Print the final score of each player and either "Player 1", "Player 2", or "Draw" depending on which score is higher or if they are both the same.

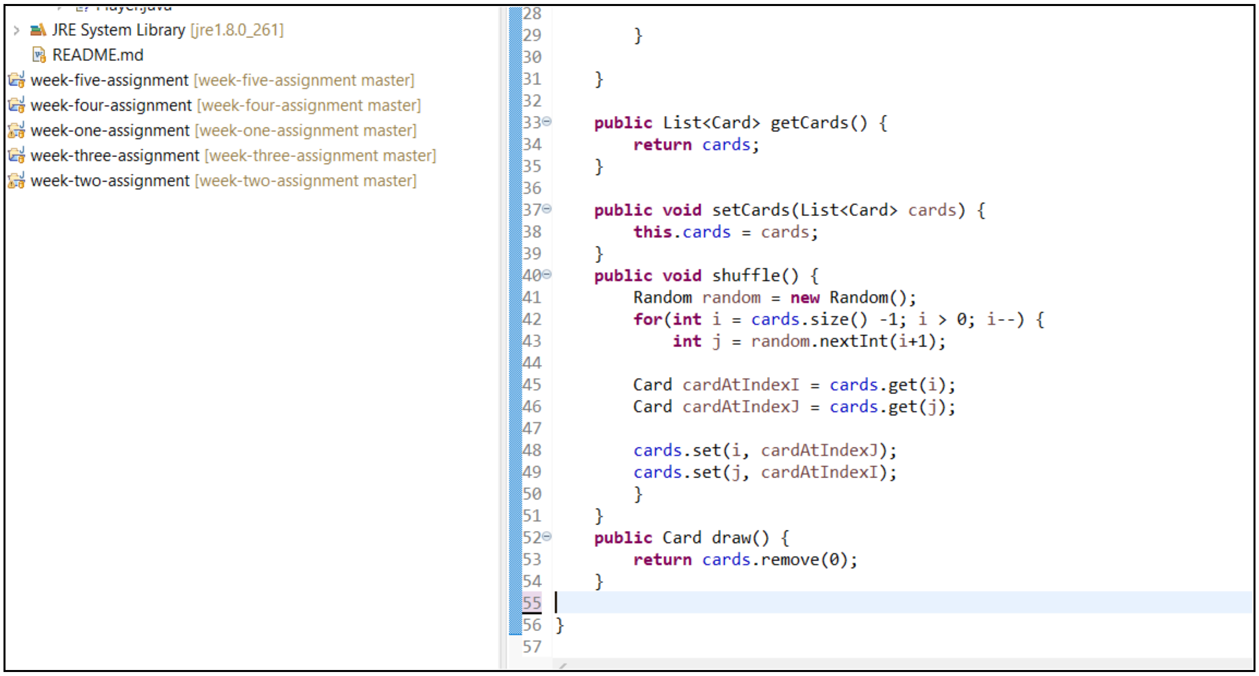
Screenshots of Code:


Quiz
week-6-assignment [week-6-assignment master]
src
(default package)
App.java
Card.java
Deck.java
Player.java
JRE System Library [jre1.8.0_261]
README.md
week-five-assignment [week-five-assignment master]
week-four-assignment [week-four-assignment master]
week-one-assignment [week-one-assignment master]
week-three-assignment [week-three-assignment master]
week-two-assignment [week-two-assignment master]

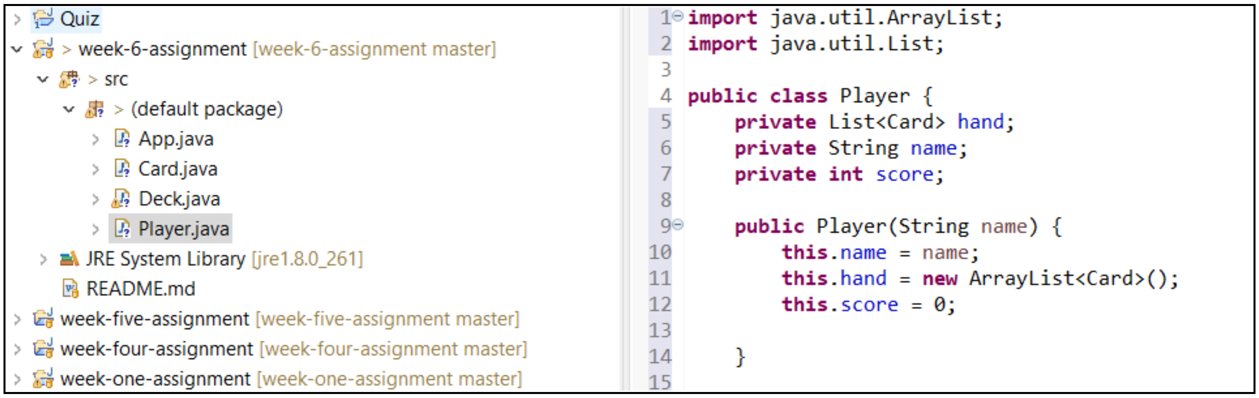
```java

public class Card {
    private int value;
    private String name;


    public Card(String cardName, int faceValue) {
        this.name = cardName;
        this.value = faceValue;
    }
    public int getValue() {
        return value;
    }
    public void setValue(int value) {
        this.value = value;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }

    public void describe() {
        System.out.println("The card " + name + " has a value of " + value);
    }


}
```



```java
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;
import java.util.Random;

public class Deck {
    private List<Card> cards = new ArrayList<>();

    private static final String[] suits = {"Clubs",  "Diamonds", "Hearts", "Spades"};

    private static final String[] faceValues = {"Two", "Three", "Four", "Five", "Six", "Seven", "Eight",
            "Nine", "Ten", "Jack", "Queen", "King", "Ace"};

    public Deck() {

        for (int suitIndex = 0; suitIndex < suits.length; suitIndex++) {
            String suit = suits[suitIndex];

        for (int faceValueIndex = 0; faceValueIndex < faceValues.length; faceValueIndex++) {
            String faceValue = faceValues[faceValueIndex];
            String cardName = faceValue + " of " + suit;
            int value = faceValueIndex + 2;

            cards.add(new Card(cardName, value));

            System.out.println(cardName + " = " + value);
        }

        }

    }
```

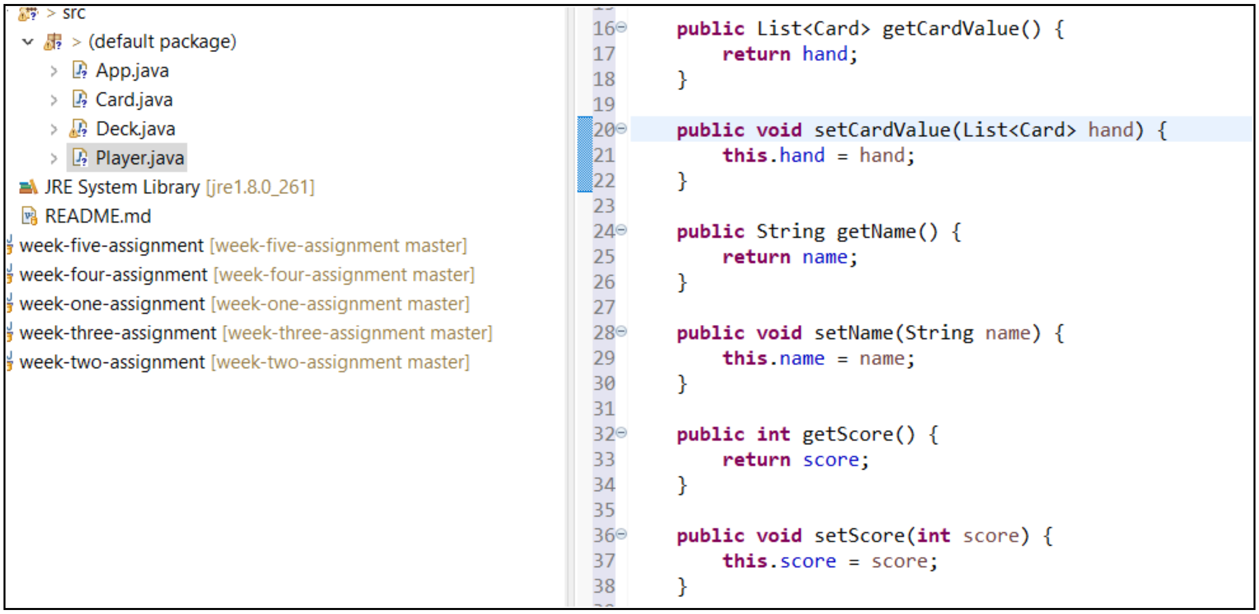

Player.java
JRE System Library [jre1.8.0_261]
README.md
week-five-assignment [week-five-assignment master]
week-four-assignment [week-four-assignment master]
week-one-assignment [week-one-assignment master]
week-three-assignment [week-three-assignment master]
week-two-assignment [week-two-assignment master]

```java
        }

    }

    public List<Card> getCards() {
        return cards;
    }

    public void setCards(List<Card> cards) {
        this.cards = cards;
    }
    public void shuffle() {
        Random random = new Random();
        for(int i = cards.size() -1; i > 0; i--) {
            int j = random.nextInt(i+1);

        Card cardAtIndexI = cards.get(i);
        Card cardAtIndexJ = cards.get(j);

        cards.set(i, cardAtIndexJ);
        cards.set(j, cardAtIndexI);
        }
    }
    public Card draw() {
        return cards.remove(0);
    }

}
```



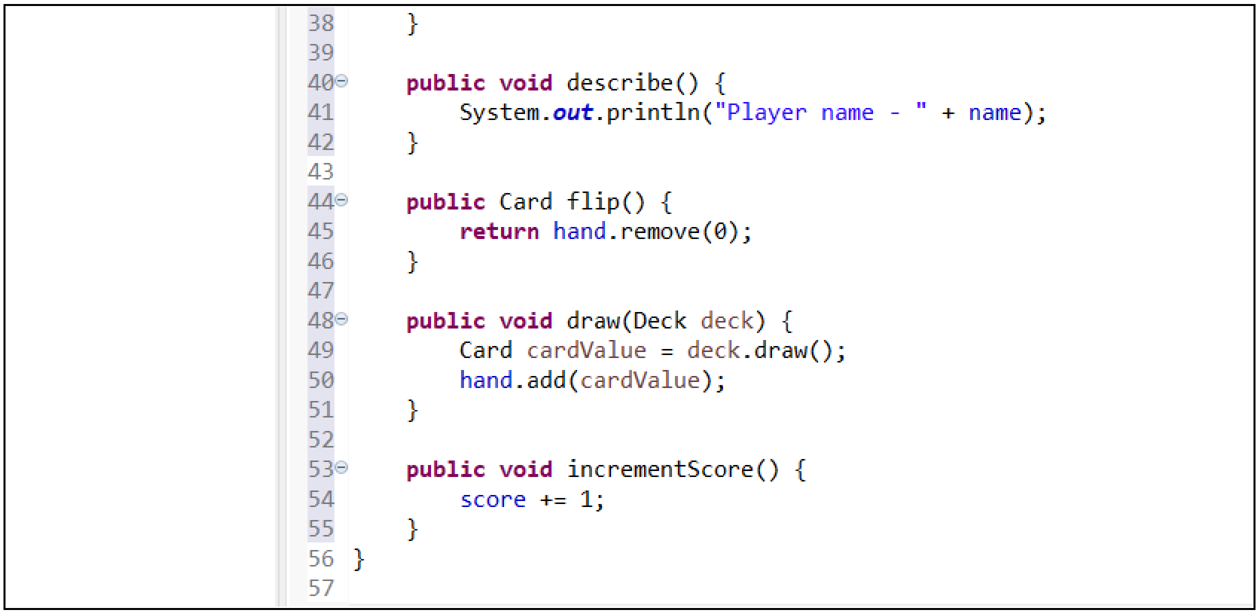
```
> Quiz
v > week-6-assignment [week-6-assignment master]
  v > src
    v > (default package)
      > App.java
      > Card.java
      > Deck.java
      > Player.java
  > JRE System Library [jre1.8.0_261]
    README.md
> week-five-assignment [week-five-assignment master]
> week-four-assignment [week-four-assignment master]
> week-one-assignment [week-one-assignment master]
```

```java
import java.util.ArrayList;
import java.util.List;

public class Player {
    private List<Card> hand;
    private String name;
    private int score;

    public Player(String name) {
        this.name = name;
        this.hand = new ArrayList<Card>();
        this.score = 0;

    }

```

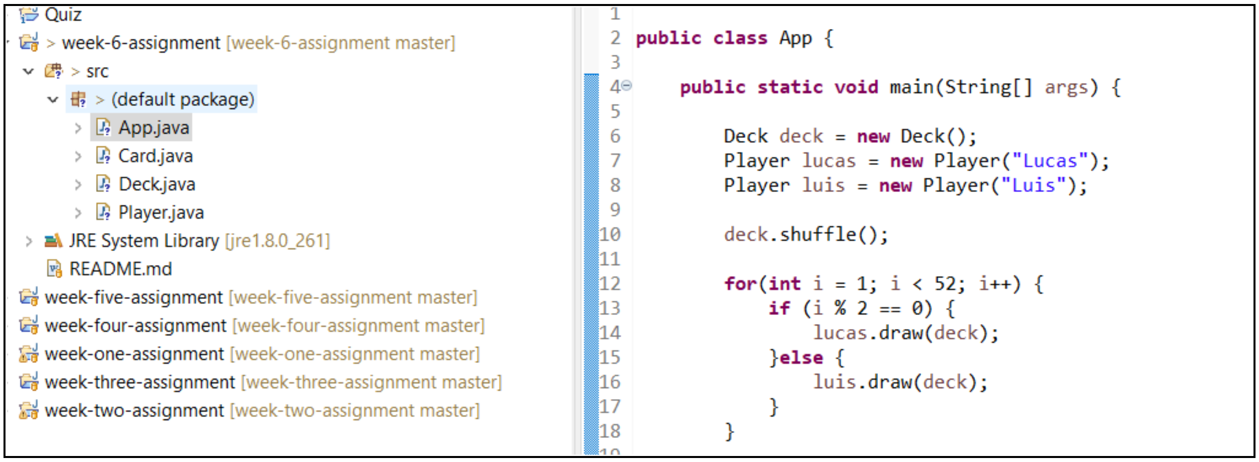


```
  > src
  v > (default package)
    > App.java
    > Card.java
    > Deck.java
    > Player.java
  JRE System Library [jre1.8.0_261]
  README.md
week-five-assignment [week-five-assignment master]
week-four-assignment [week-four-assignment master]
week-one-assignment [week-one-assignment master]
week-three-assignment [week-three-assignment master]
week-two-assignment [week-two-assignment master]
```

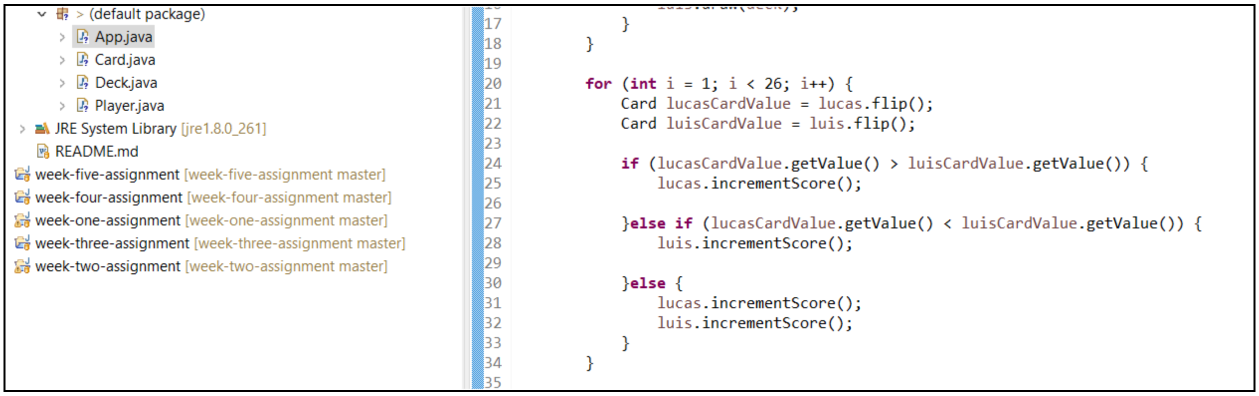
```java
    public List<Card> getCardValue() {
        return hand;
    }

    public void setCardValue(List<Card> hand) {
        this.hand = hand;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getScore() {
        return score;
    }

    public void setScore(int score) {
        this.score = score;
    }
```



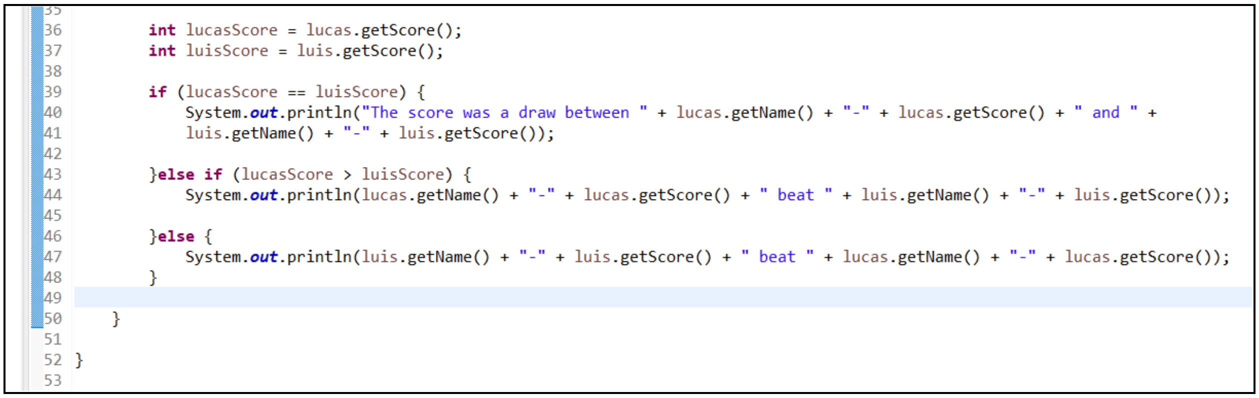
```java
    }

    public void describe() {
        System.out.println("Player name - " + name);
    }

    public Card flip() {
        return hand.remove(0);
    }

    public void draw(Deck deck) {
        Card cardValue = deck.draw();
        hand.add(cardValue);
    }

    public void incrementScore() {
        score += 1;
    }
}

```


Quiz
week-6-assignment [week-6-assignment master]
src
(default package)
App.java
Card.java
Deck.java
Player.java
JRE System Library [jre1.8.0_261]
README.md
week-five-assignment [week-five-assignment master]
week-four-assignment [week-four-assignment master]
week-one-assignment [week-one-assignment master]
week-three-assignment [week-three-assignment master]
week-two-assignment [week-two-assignment master]

```
public class App {

    public static void main(String[] args) {

        Deck deck = new Deck();
        Player lucas = new Player("Lucas");
        Player luis = new Player("Luis");

        deck.shuffle();

        for(int i = 1; i < 52; i++) {
            if (i % 2 == 0) {
                lucas.draw(deck);
            }else {
                luis.draw(deck);
            }
        }
```

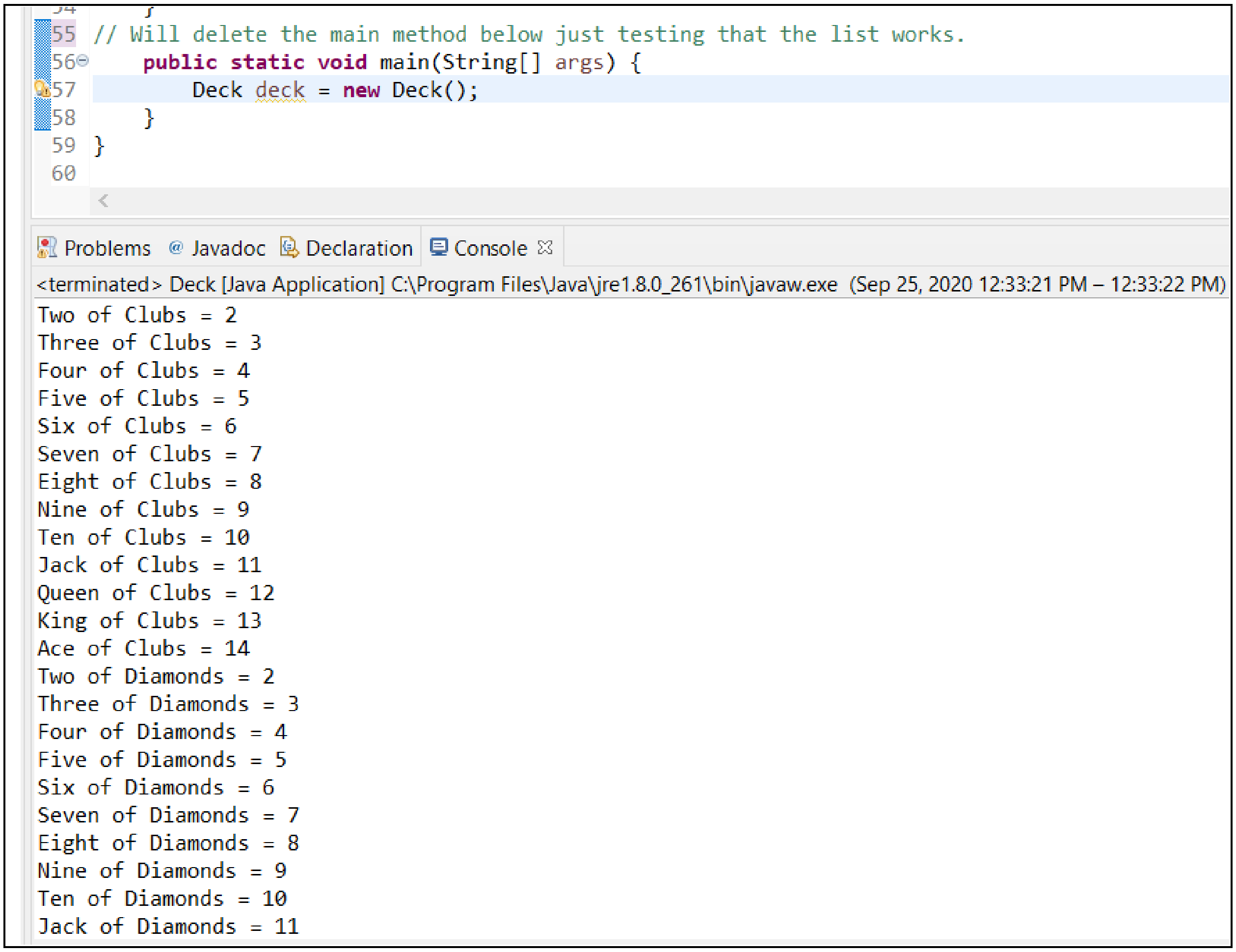

(default package)
App.java
Card.java
Deck.java
Player.java
JRE System Library [jre1.8.0_261]
README.md
week-five-assignment [week-five-assignment master]
week-four-assignment [week-four-assignment master]
week-one-assignment [week-one-assignment master]
week-three-assignment [week-three-assignment master]
week-two-assignment [week-two-assignment master]

```
            }
        }

        for (int i = 1; i < 26; i++) {
            Card lucasCardValue = lucas.flip();
            Card luisCardValue = luis.flip();

            if (lucasCardValue.getValue() > luisCardValue.getValue()) {
                lucas.incrementScore();

            }else if (lucasCardValue.getValue() < luisCardValue.getValue()) {
                luis.incrementScore();

            }else {
                lucas.incrementScore();
                luis.incrementScore();
            }
        }
```



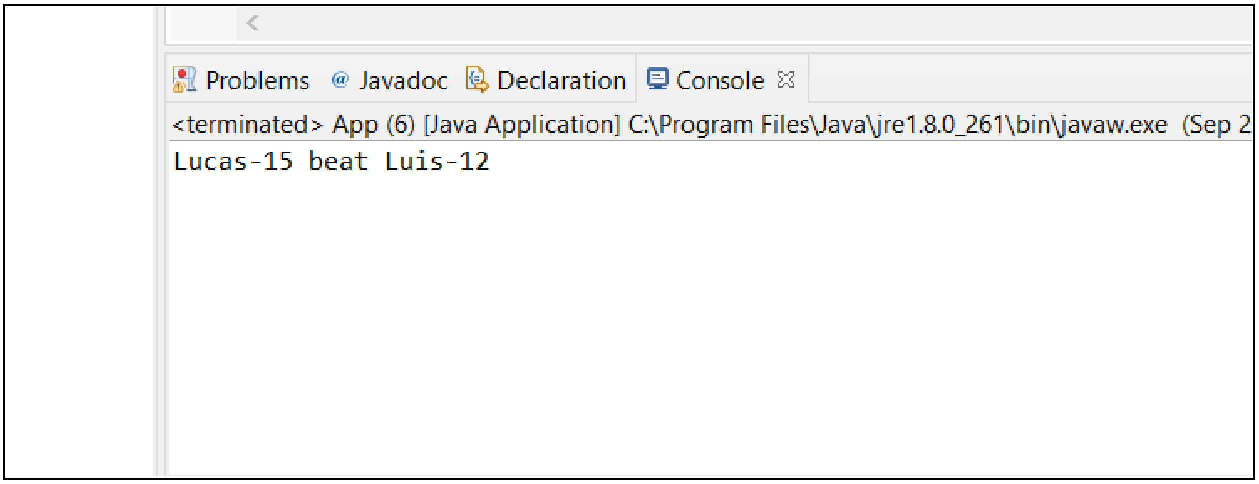
```
        int lucasScore = lucas.getScore();
        int luisScore = luis.getScore();

        if (lucasScore == luisScore) {
            System.out.println("The score was a draw between " + lucas.getName() + "-" + lucas.getScore() + " and " +
            luis.getName() + "-" + luis.getScore());

        }else if (lucasScore > luisScore) {
            System.out.println(lucas.getName() + "-" + lucas.getScore() + " beat " + luis.getName() + "-" + luis.getScore());

        }else {
            System.out.println(luis.getName() + "-" + luis.getScore() + " beat " + lucas.getName() + "-" + lucas.getScore());
        }

    }

}
```

Screenshots of Running Application:

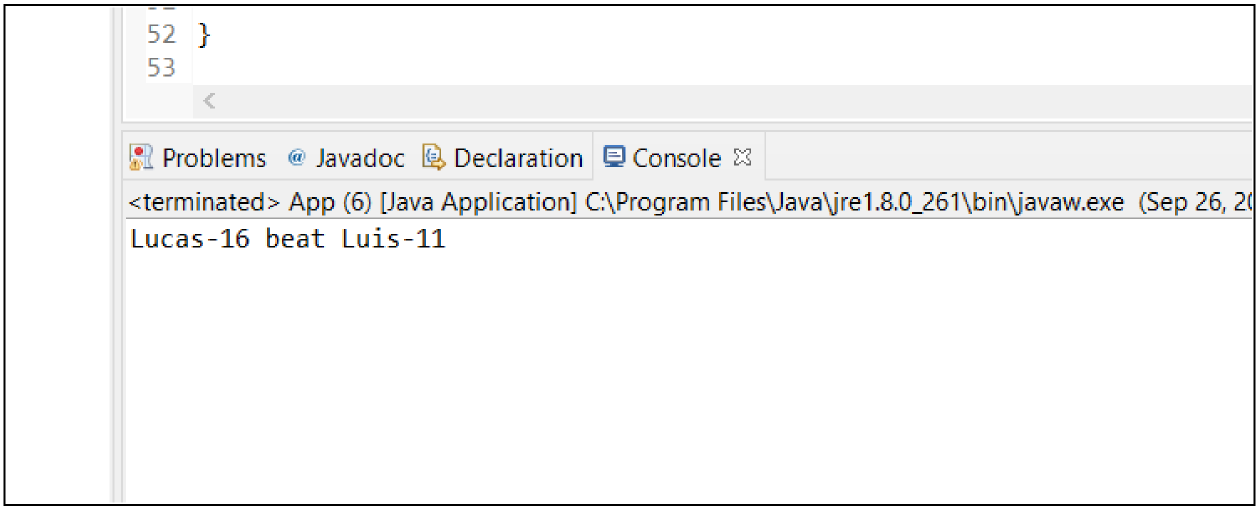


```
    }
// Will delete the main method below just testing that the list works.
    public static void main(String[] args) {
        Deck deck = new Deck();
    }
}

```

```
<terminated> Deck [Java Application] C:\Program Files\Java\jre1.8.0_261\bin\javaw.exe  (Sep 25, 2020 12:33:21 PM – 12:33:22 PM)
Two of Clubs = 2
Three of Clubs = 3
Four of Clubs = 4
Five of Clubs = 5
Six of Clubs = 6
Seven of Clubs = 7
Eight of Clubs = 8
Nine of Clubs = 9
Ten of Clubs = 10
Jack of Clubs = 11
Queen of Clubs = 12
King of Clubs = 13
Ace of Clubs = 14
Two of Diamonds = 2
Three of Diamonds = 3
Four of Diamonds = 4
Five of Diamonds = 5
Six of Diamonds = 6
Seven of Diamonds = 7
Eight of Diamonds = 8
Nine of Diamonds = 9
Ten of Diamonds = 10
Jack of Diamonds = 11
```

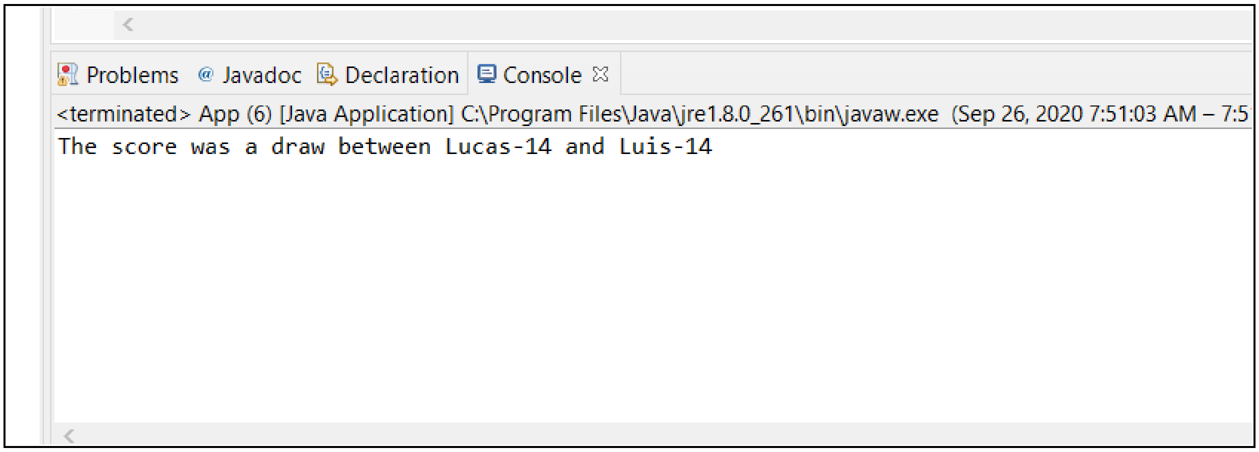


```
<terminated> App (6) [Java Application] C:\Program Files\Java\jre1.8.0_261\bin\javaw.exe  (Sep 2
Lucas-15 beat Luis-12
```



```
}
```

```
<terminated> App (6) [Java Application] C:\Program Files\Java\jre1.8.0_261\bin\javaw.exe  (Sep 26, 2(
Lucas-16 beat Luis-11
```



```
<terminated> App (6) [Java Application] C:\Program Files\Java\jre1.8.0_261\bin\javaw.exe  (Sep 26, 2020 7:51:03 AM – 7:5
The score was a draw between Lucas-14 and Luis-14
```

URL to GitHub Repository:

https://github.com/lcuevas6/week-6-assignment.git