URL to GitHub Repository: https://github.com/aliustunyer/Card-Game-War-.git

URL to Public Link of my Video: https://youtu.be/v6cxxzRrSxo

Main Class (APP);

```
package week6CodingAssignment;
import java.util.Scanner;
public class App {
	static Scanner scanner = new Scanner(System.in);

	public static void main(String[] args) {

		//Instantiating two players with the names entered by the user.

		Player player1= new Player(getPlayer1Name());
		Player player2= new Player(getPlayer2Name());

	    // the user should press 1 to start the game, press 2 to exit,in case of another user input
	    // the system gives a warning to prompt user to enter a valid choice.

	startGame();

	    // if the user chooses to start the game a new deck is instantiated with 52 cards

		Deck deck = new Deck();

		// the method below shuffles the cards in the deck randomly.

	deck.shuffle();

	    // the loop with the method below removes cards from the deck and add the cards to each player's hand;
	    // this method also uses deck.draw(); method that removes and returns the top card of the Cards field

	 for (int i=1 ; i<=26; i++) {

	 player1.draw(deck);
	 player2.draw(deck);
	 }

	  // the method below with a loop compares the value of the each card returned by the two player's flip() methods,
	  // increments the score for the winner of the turn, shows the number of the turn, calls the incrementScore()
	  // gives information about each player,their score,name using the player.describe(),
	  // and the cards in their hands by using the card.describe(),finally, prints out the winner of each turn.

	for (int i =1; i<=26; i++) {
```

```
      infoEachTurn(player1,  player2, i);

        // the two methods below in the loop show the updated total score for each
player to indicate game progression.

      player1.progression();
      player2.progression();

        // the method below in the loop informs user that the previous turn is over
and asks user input to proceed the game;
        // the user should press 1 to continue, press 2 to exit,
        // in case of another user input, the system gives a warning to prompt user
to enter a valid choice.
        // if the turn number is 26, it informs the user that was the last turn.

      proceedGame(i);
          }

        // the method below informs the user when the game is over,
        // compares the final scores and also prints out the final score of each
player
        // with the name of the winner; or it prints out as draw if the result is a
tie.

      endGame(player1, player2);

      }



  public static String getPlayer1Name() {

        System.out.print("Enter the name of the first player :");
        return  scanner.next();
      }


  public static String getPlayer2Name() {

        System.out.print("Enter the name of the second player :");
        return  scanner.next();
      }


  public static void startGame() {

        System.out.println "Press 1 to start the game!");
        System.out.println "Press 2 to exit!");
        String a =scanner.next();

        if (a.equals("1")) {
        System.out.println "The game starts, GOOD LUCK!!!");
        System.out.println "=============================================");
        Deck deck = new Deck();
        deck.shuffle();
      }
      else if (a.equals("2")) {
```

```
        System.out.println "Goodbye!!!");
        System.exit 0);
            }
        else {
        System.out.println "Please enter a valid choice!!!");
        startGame();
            }
    }


    public static void infoEachTurn(Player player1, Player player2, int i) {

        System.out.println("Turn Number : " + i);
        System.out.println("-----------------------------------------------");

        Card player1Hand = player1.flip();
        System.out.print  player1.getName() +"'s hand: ");
          player1Hand.describe();

        Card  player2Hand = player2.flip();
        System.out.print  player2.getName() +"'s hand: ");
            player2Hand.describe();

         if (player1Hand.getValue()> player2Hand.getValue()) {
           player1.incrementScore();
        System.out.println("        "+player1.getName()+" wins this turn!!!
");
         }
         else if (player1Hand.getValue()< player2Hand.getValue()) {
           player2.incrementScore();
        System.out.println("        "+player2.getName()+" wins this turn!!!
");
         }
         else {
           System.out.println("           !!!draw!!!      ");
         }
         System.out.println("-----------------------------------------------");
    }


    public static void proceedGame(int i) {

          if (i!=26) {
          System.out.println("-----------------------------------------------");
          System.out.println("Turn Number " +i +" is over ");
          System.out.println("-----------------------------------------------");
          System.out.println("Press 1 to proceed game!");
          System.out.println("Press 2 to exit!");
          String a =scanner.next();

          if (a.equals("1")) {

          System.out.println("===============================================");
    }
          else if (a.equals("2")) {
          System.out.println("Goodbye!!!");
          System.exit(0);
              }
          else {
```

```
            System.out.println("Please enter a valid choice!!!");
            proceedGame(i);
        }
            }
            else {
            System.out.println("================================================");
            System.out.println(">>>          This was the last turn          <<<");
            System.out.println("================================================");
            }
}


  public static void endGame(Player player1, Player player2) {

        System.out.println(">>>                 END OF GAME                <<<");
        System.out.println("================================================");
        System.out.print("Final Score >>> "+ player1.getName()+ ": ");
        System.out.println(player1.getScore());
        System.out.print("Final Score >>> "+ player2.getName()+ ": ");
        System.out.println(player2.getScore());

    if (player1.getScore()> player2.getScore()) {

       System.out.println("================================================");
       System.out.println("                    "+player1.getName()+" WINS!");
       System.out.println("================================================");


    } else if (player1.getScore()< player2.getScore()) {

       System.out.println("================================================");
       System.out.println("                    "+player2.getName()+" WINS!");
       System.out.println("================================================");

    } else {

        System.out.println("================================================");
        System.out.println("                  !!!DRAW!!!                    ");
        System.out.println("================================================");
    }
  }
}
```

Class (Card);

```
package week6CodingAssignment;

public class Card {

      String name;
      String suit;
      int value;

      Card (String name, String suit, int value) {
            this.name =name;
            this.suit = suit;
            this.value = value;
      }
```

```
	public String getName() {
		return name;
	}

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

	public String getSuit() {
		return suit;
	}

	public void setSuit String suit) {
		this.suit = suit;
	}

	public int getValue() {
		return value;
	}

	public void setValue int value) {
		this.value = value;
	}


	public void describe () {
		System.out.println ( name + " of " + suit + " --" + " Value: " +
value);


	}
```

Class (Deck);

```
package week6CodingAssignment;

import java.util.ArrayList;

public class Deck {

	List<Card> cards = new ArrayList<Card>();


	Deck (){

		String [] suits = {"Clubs", "Diamonds", "Hearts", "Spades"};
		String [] numbers = {"Two", "Three","Four", "Five","Six", "Seven",
"Eigth", "Nine", "Ten", "Jack","Queen","King","Ace"};
		for (int i=0 ; i<13; i++) {
			int count =2;
			count += i;
			for (int b=0; b<4;b++) {
				Card card = new Card(numbers[i],suits[b], count);
				cards.add(card);

			}
		}
```

```
	}


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

	public void setCards(List<Card> cards) {
		this.cards = cards;
	}


	// the  method below shows all the information about the cards in the deck
using cards.describe()

    public void describe() {

	int i =0;
	for (Card card : cards){
		i++;
		card.describe();
		System.out.println(i);
	}
    }


	public void shuffle() {
		Collections.shuffle(cards);
	}

	public Card draw(){
		Card card = cards.remove(0);
		return card;


	}
```

Class (Player);

```
package week6CodingAssignment;

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

public class Player {

	List<Card> hand = new ArrayList<Card>();
	int score =0;
	String name ;

	Player (String name){
		this.name =name;

	}

	Player (List<Card> hand, String name, int score){
		this.hand = hand;
		this.name = name;
		this.score = score;
```

```
      }

      public List<Card> gethand() {
            return hand;
      }
      public void sethand(List<Card> hand) {
            this.hand = hand;
      }
      public int getScore() {
            return score;
      }
      public void setScore(int score) {
            this.score = score;
      }
      public String getName() {
            return name;
      }
      public void setName(String name) {
            this.name = name;
      }


      //the method below is defined to show information for each player, their
score,name,
      //and the cards in their hands by using the card.describe(),

    public void describe() {

      int i = 1;
      System.out.println( "============================================");
      System.out.println( "Player Name : " + name + " == " + "Score :  " + score );
      System.out.println( "============================================");
      for (Card card : hand) {
      System.out.println(i);
      card.describe();
      i++;
}

}
    public Card flip () {

      Card card = hand.remove(0);
      return card;
}


    public void draw (Deck deck) {

      Card card = deck.draw();
      hand.add(card);
}

    public void incrementScore () {

      score++;
}
    public void progression() {

      System.out.println(  "Total Score >>> " +name + " : " + score );
```