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. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

Coding Steps:

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

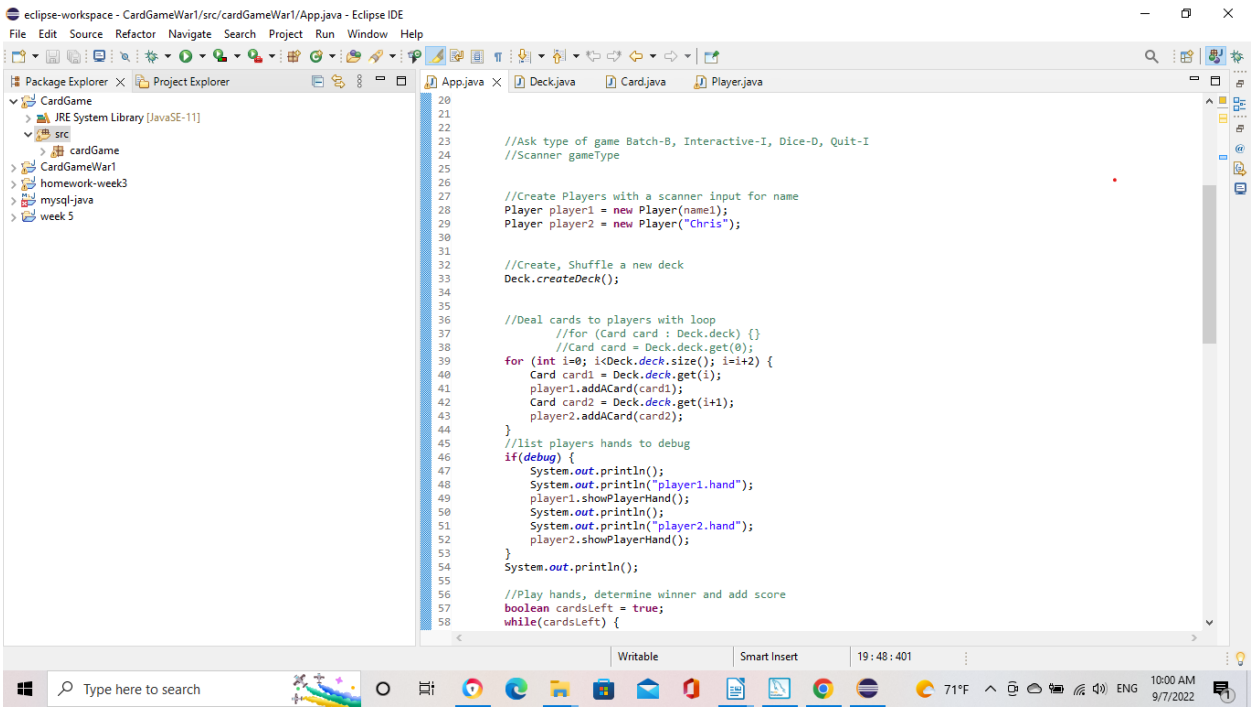
1. Create the following classes.
    a. Card
        i. Fields
            1. **value** (contains a value from 2-14 representing cards 2-Ace)
            2. **name** (e.g. Ace of Diamonds, or Two of Hearts)
        ii. Methods
            1. Getters and Setters
            2. **describe** (prints out 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)

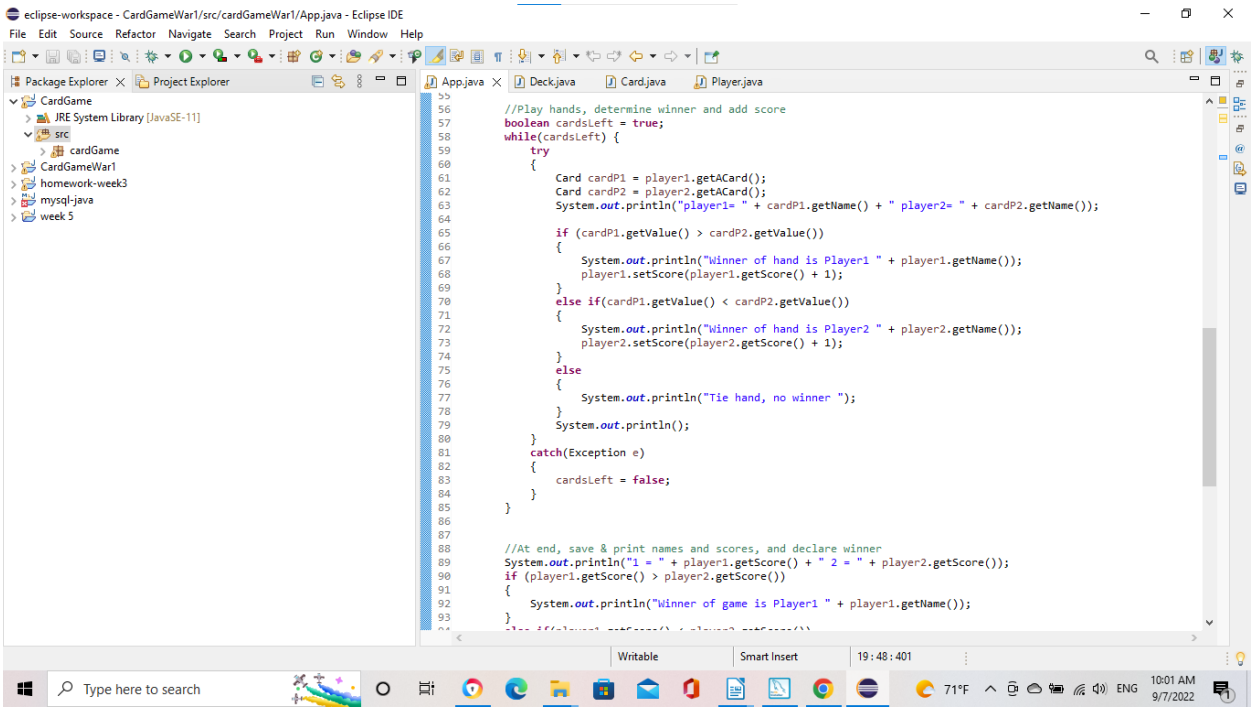
3. In the constructor, when a new Deck is instantiated, the Cards field should be populated with the standard 52 cards.

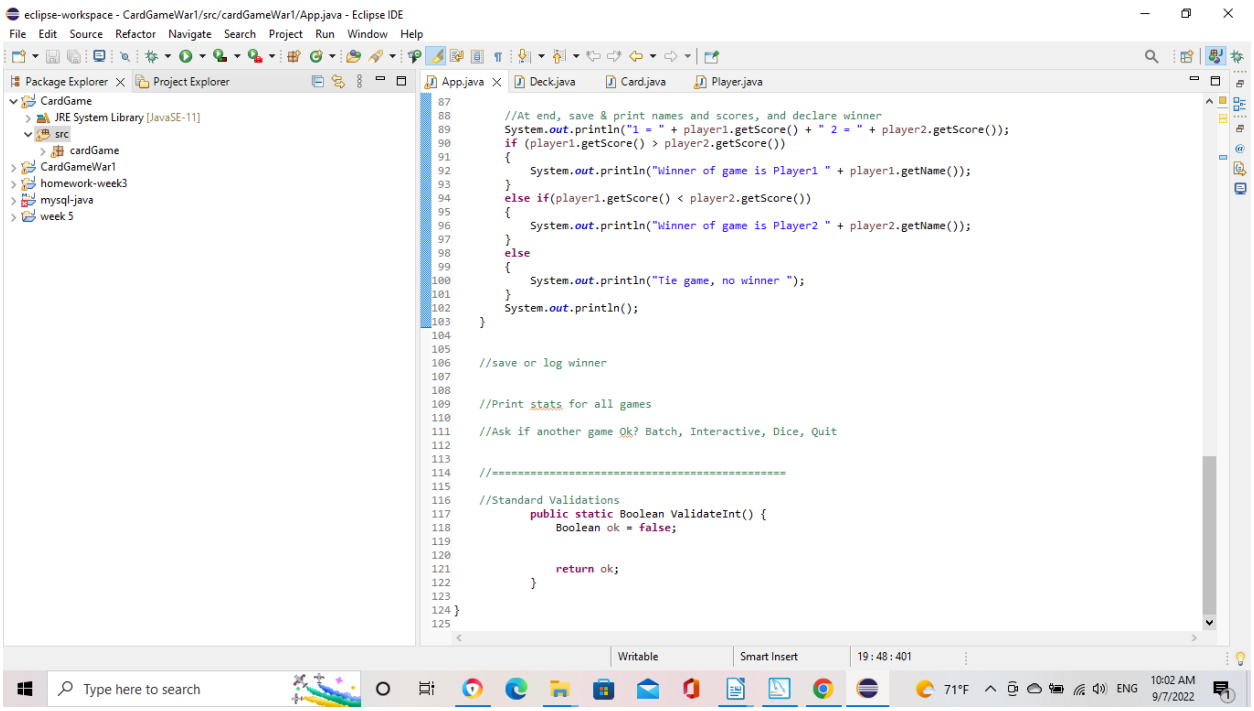
c. Player
   i. Fields
      1. **hand** (List of Card)
      2. **score** (set to 0 in the constructor)
      3. **name**
   ii. Methods
      1. **describe** (prints out information about the player and calls the describe method for each card in the Hand List)
      2. **flip** (removes and returns 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)

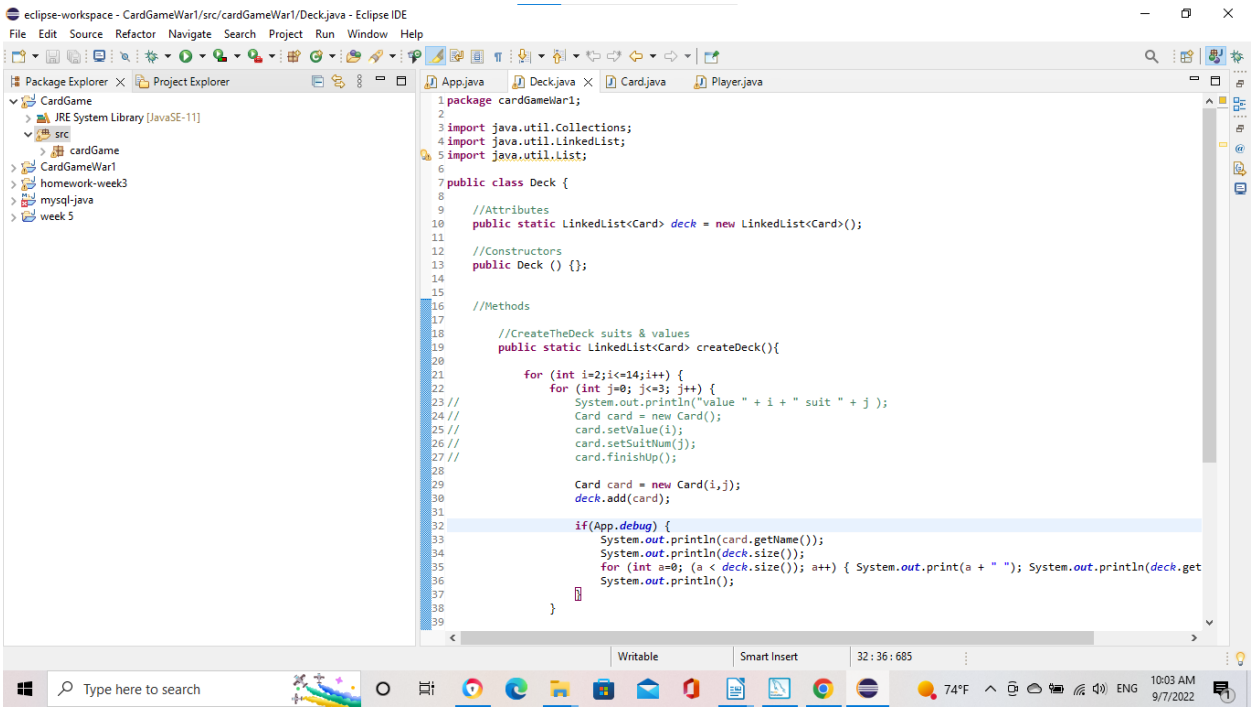
2. Create a class called App with a main method.
3. Instantiate a Deck and two Players, call the shuffle method on the deck.
4. Using a traditional for loop, iterate 52 times calling the Draw method on the other player each iteration using the Deck you instantiated.
5. Using a traditional for loop, iterate 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.

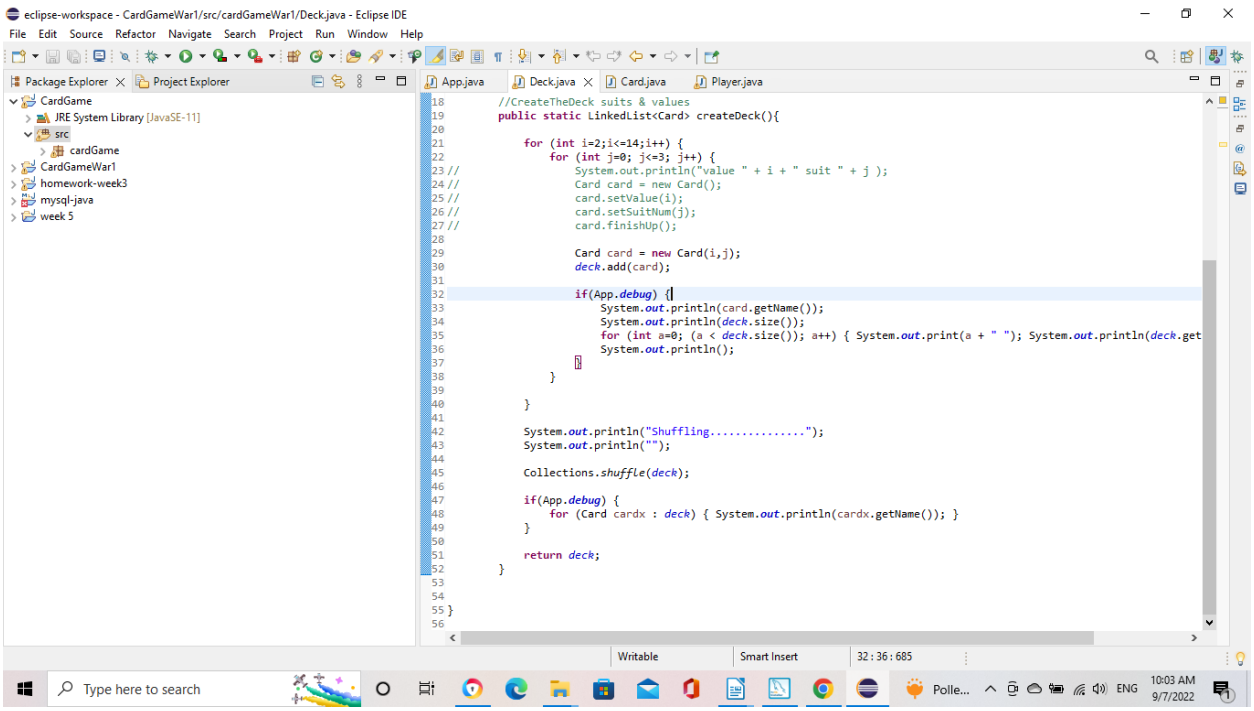
Screenshots of Code:

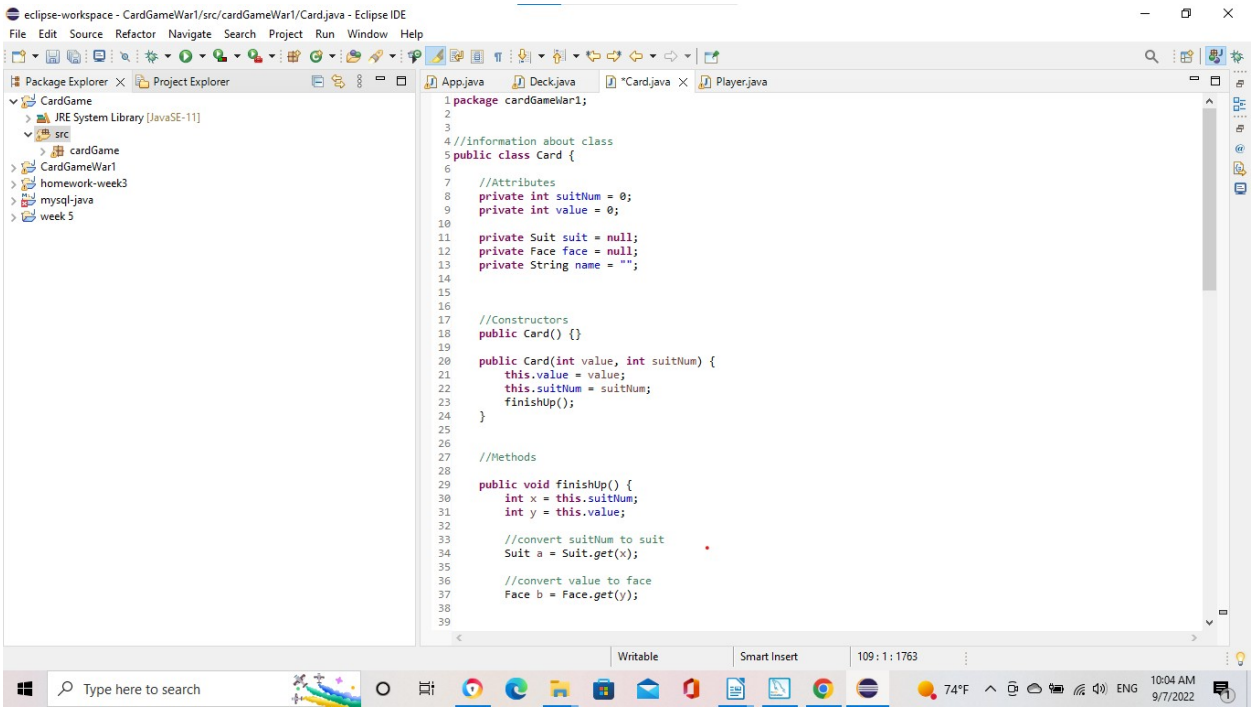

eclipse-workspace - CardGameWar1/src/cardGameWar1/App.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
CardGame
JRE System Library [JavaSE-11]
src
cardGame
CardGameWar1
homework-week3
mysql-java
week 5
App.java
Deck.java
Card.java
Player.java
//Ask type of game Batch-B, Interactive-I, Dice-D, Quit-I
//Scanner gameType
//Create Players with a scanner input for name
Player player1 = new Player(name1);
Player player2 = new Player("Chris");
//Create, Shuffle a new deck
Deck.createDeck();
//Deal cards to players with loop
//for (Card card : Deck.deck) {}
//Card card = Deck.deck.get(0);
for (int i=0; i<Deck.deck.size(); i=i+2) {
Card card1 = Deck.deck.get(i);
player1.addACard(card1);
Card card2 = Deck.deck.get(i+1);
player2.addACard(card2);
}
//list players hands to debug
if(debug) {
System.out.println();
System.out.println("player1.hand");
player1.showPlayerHand();
System.out.println();
System.out.println("player2.hand");
player2.showPlayerHand();
}
System.out.println();
//Play hands, determine winner and add score
boolean cardsLeft = true;
while(cardsLeft) {
Writable
Smart Insert
19 : 48 : 401
Type here to search
71°F
ENG
10:00 AM
9/7/2022

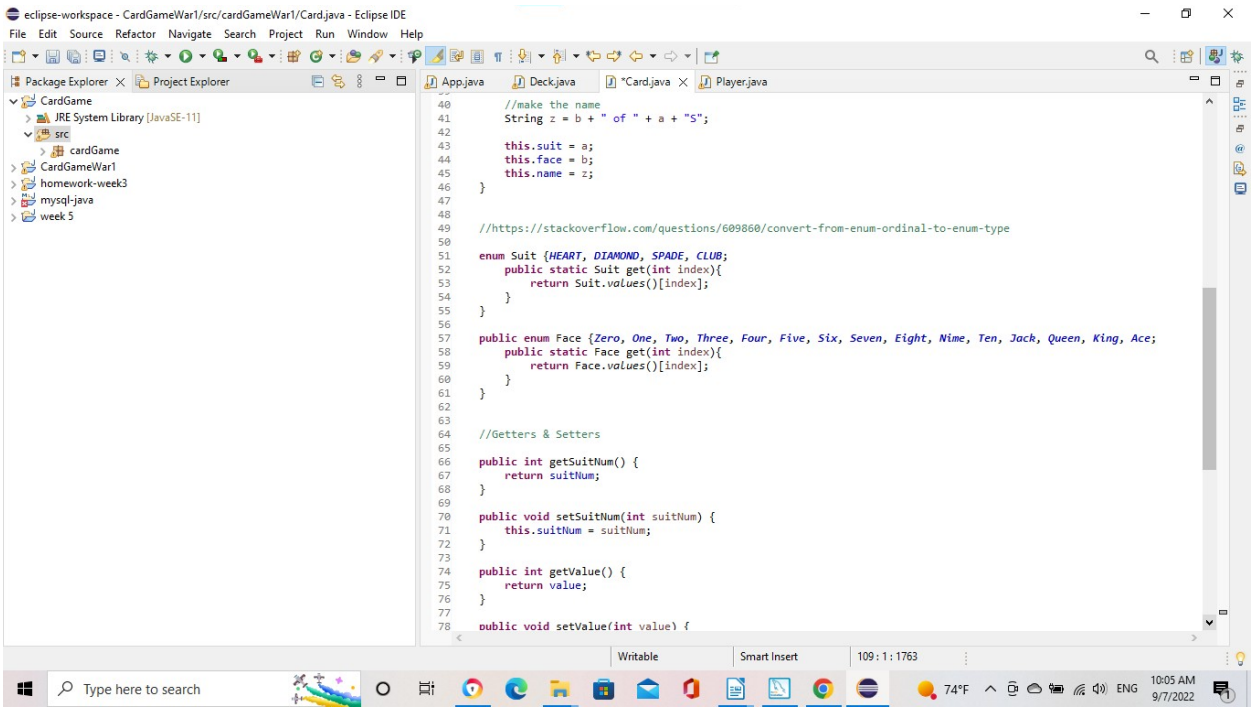

eclipse-workspace - CardGameWar1/src/cardGameWar1/App.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
CardGame
JRE System Library [JavaSE-11]
src
cardGame
CardGameWar1
homework-week3
mysql-java
week 5
App.java
Deck.java
Card.java
Player.java
//Play hands, determine winner and add score
boolean cardsLeft = true;
while(cardsLeft) {
try
{
Card cardP1 = player1.getACard();
Card cardP2 = player2.getACard();
System.out.println("player1= " + cardP1.getName() + " player2= " + cardP2.getName());
if (cardP1.getValue() > cardP2.getValue())
{
System.out.println("Winner of hand is Player1 " + player1.getName());
player1.setScore(player1.getScore() + 1);
}
else if(cardP1.getValue() < cardP2.getValue())
{
System.out.println("Winner of hand is Player2 " + player2.getName());
player2.setScore(player2.getScore() + 1);
}
else
{
System.out.println("Tie hand, no winner ");
}
System.out.println();
}
catch(Exception e)
{
cardsLeft = false;
}
}
//At end, save & print names and scores, and declare winner
System.out.println("1 = " + player1.getScore() + " 2 = " + player2.getScore());
if (player1.getScore() > player2.getScore())
{
System.out.println("Winner of game is Player1 " + player1.getName());
}
Writable
Smart Insert
19 : 48 : 401
Type here to search
71°F
ENG
10:01 AM
9/7/2022

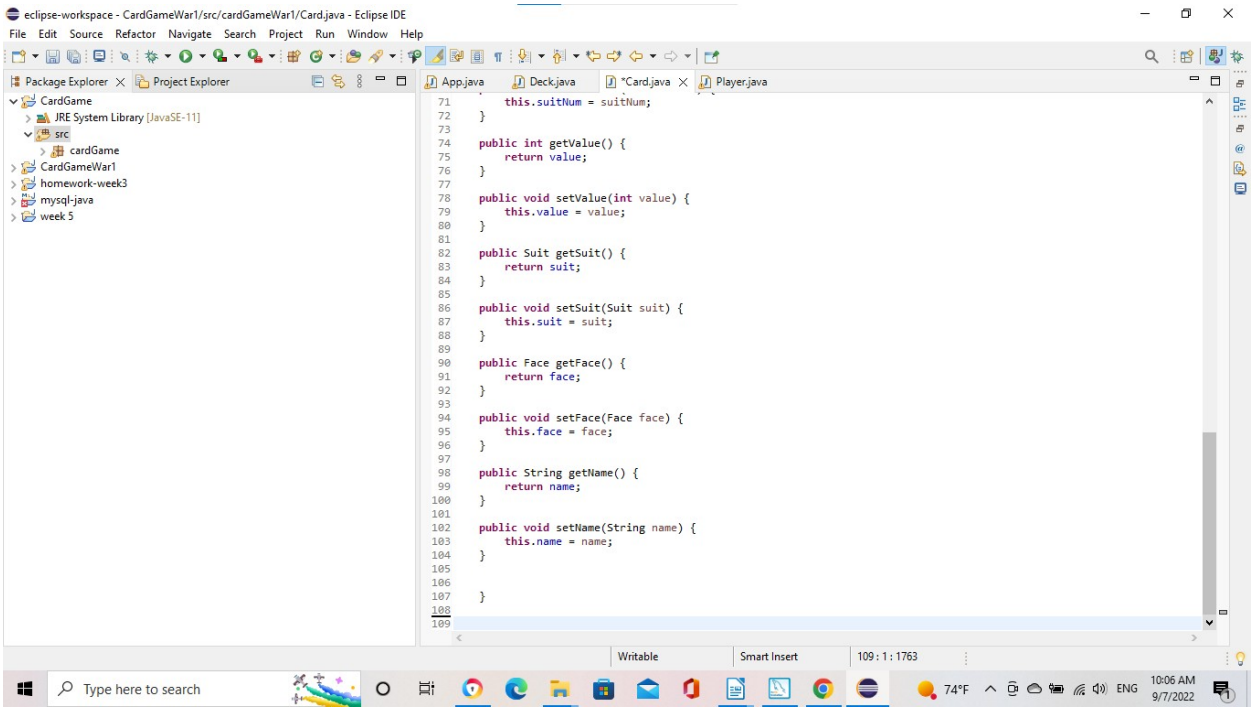

eclipse-workspace - CardGameWar1/src/cardGameWar1/App.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
CardGame
JRE System Library [JavaSE-11]
src
cardGame
CardGameWar1
homework-week3
mysql-java
week 5
App.java
Deck.java
Card.java
Player.java
//At end, save & print names and scores, and declare winner
System.out.println("1 = " + player1.getScore() + " 2 = " + player2.getScore());
if (player1.getScore() > player2.getScore())
{
System.out.println("Winner of game is Player1 " + player1.getName());
}
else if(player1.getScore() < player2.getScore())
{
System.out.println("Winner of game is Player2 " + player2.getName());
}
else
{
System.out.println("Tie game, no winner ");
}
System.out.println();
}
//save or log winner
//Print stats for all games
//Ask if another game Ok? Batch, Interactive, Dice, Quit
//==========
//Standard Validations
public static Boolean ValidateInt() {
Boolean ok = false;
return ok;
}
}
Writable
Smart Insert
19 : 48 : 401
Type here to search
71°F
ENG
10:02 AM
9/7/2022

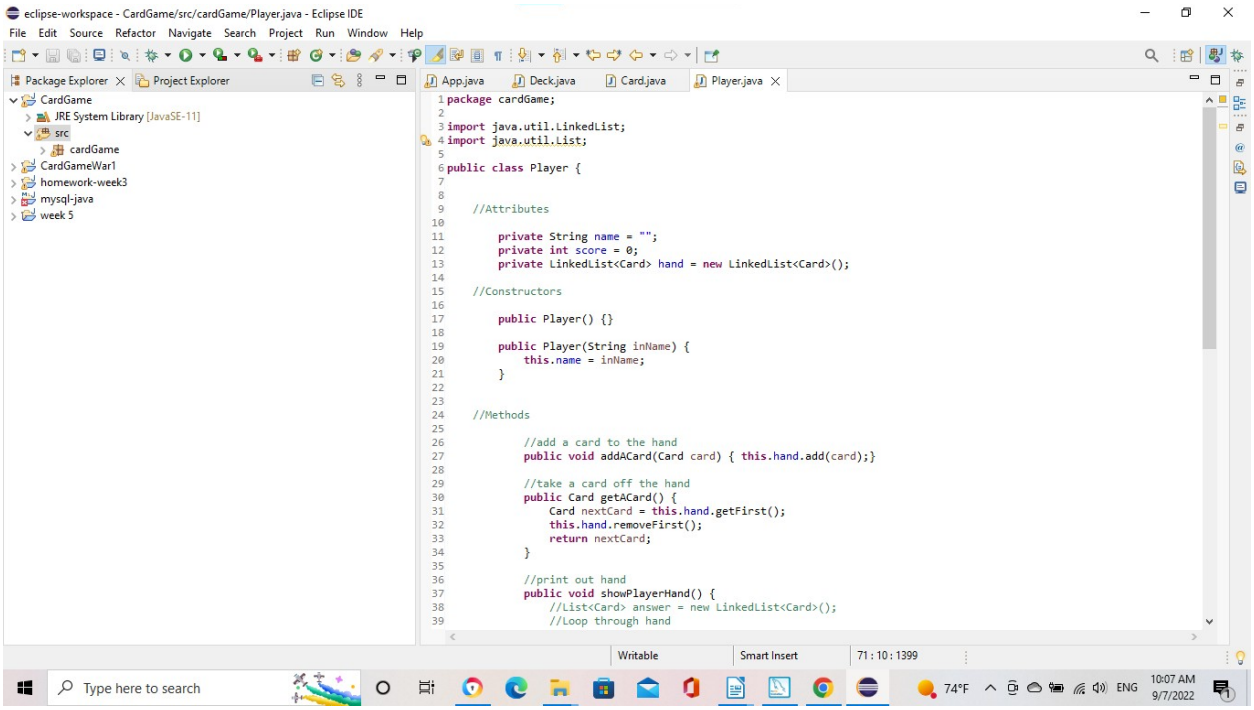

eclipse-workspace - CardGameWar1/src/cardGameWar1/Deck.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
CardGame
JRE System Library [JavaSE-11]
src
cardGame
CardGameWar1
homework-week3
mysql-java
week 5
App.java
Deck.java
Card.java
Player.java
package cardGameWar1;
import java.util.Collections;
import java.util.LinkedList;
import java.util.List;
public class Deck {
//Attributes
public static LinkedList<Card> deck = new LinkedList<Card>();
//Constructors
public Deck () {};
//Methods
//CreateTheDeck suits & values
public static LinkedList<Card> createDeck(){
for (int i=2;i<=14;i++) {
for (int j=0; j<=3; j++) {
// System.out.println("value " + i + " suit " + j);
// Card card = new Card();
// card.setValue(i);
// card.setSuitNum(j);
// card.finishUp();
Card card = new Card(i,j);
deck.add(card);
if(App.debug) {
System.out.println(card.getName());
System.out.println(deck.size());
for (int a=0; (a < deck.size()); a++) { System.out.print(a + " "); System.out.println(deck.get
System.out.println();
}
}
Writable
Smart Insert
32 : 36 : 685
Type here to search
74°F
ENG
10:03 AM
9/7/2022

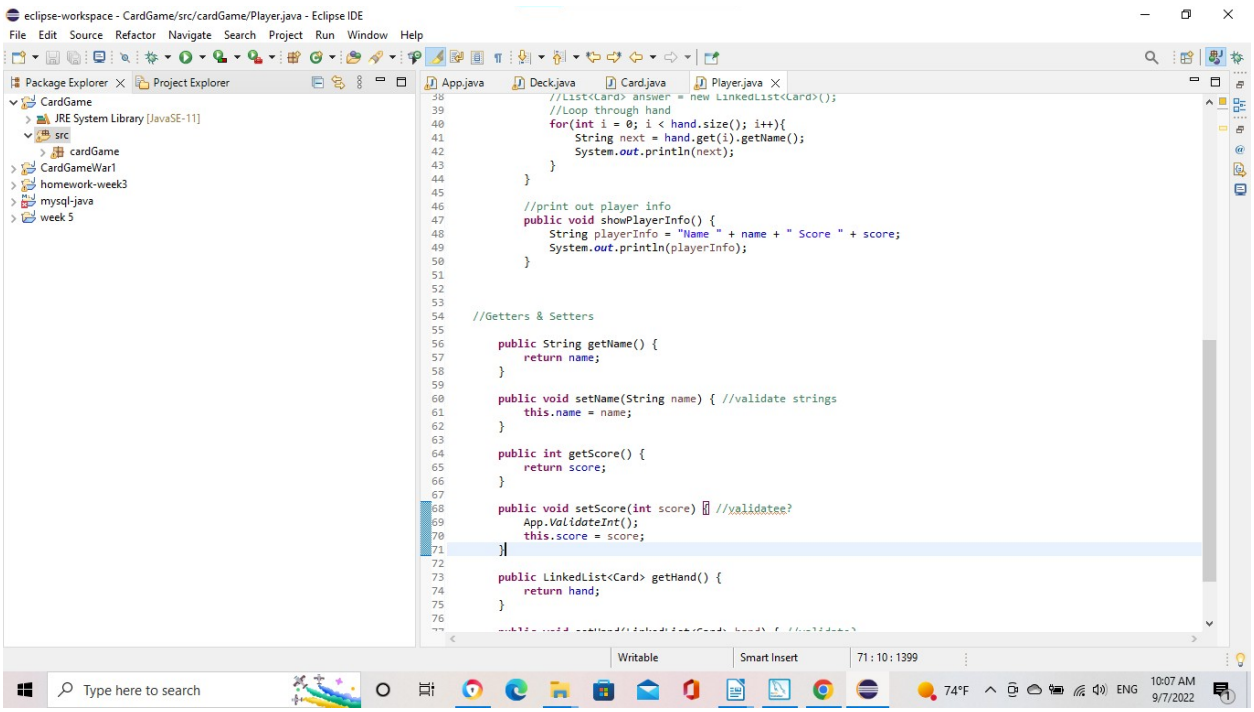

eclipse-workspace - CardGameWar1/src/cardGameWar1/Deck.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
CardGame
JRE System Library [JavaSE-11]
src
cardGame
CardGameWar1
homework-week3
mysql-java
week 5
App.java
Deck.java
Card.java
Player.java
//CreateTheDeck suits & values
public static LinkedList<Card> createDeck(){
for (int i=2;i<=14;i++) {
for (int j=0; j<=3; j++) {
// System.out.println("value " + i + " suit " + j);
// Card card = new Card();
// card.setValue(i);
// card.setSuitNum(j);
// card.finishUp();
Card card = new Card(i,j);
deck.add(card);
if(App.debug) {
System.out.println(card.getName());
System.out.println(deck.size());
for (int a=0; (a < deck.size()); a++) { System.out.print(a + " "); System.out.println(deck.get
System.out.println();
}
}
}
System.out.println("Shuffling.....");
System.out.println("");
Collections.shuffle(deck);
if(App.debug) {
for (Card cardx : deck) { System.out.println(cardx.getName()); }
}
return deck;
}
}
Writable
Smart Insert
32 : 36 : 685
Type here to search
Polle...
ENG
10:03 AM
9/7/2022

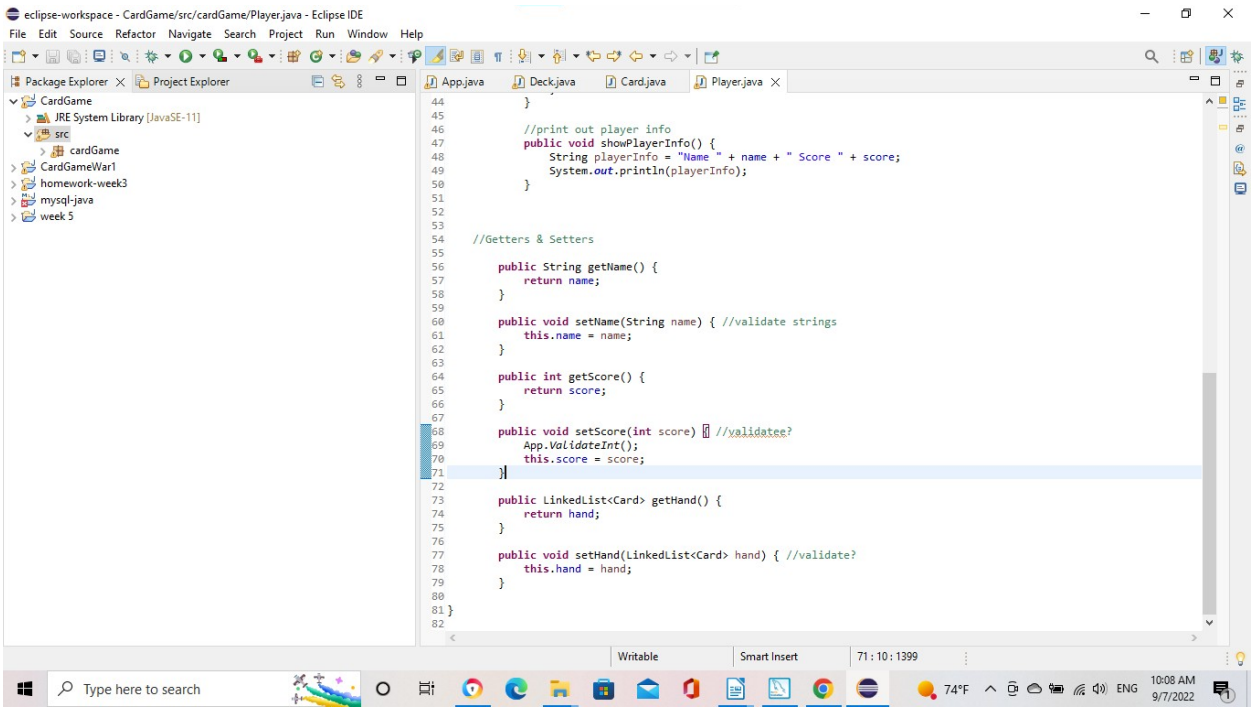

eclipse-workspace - CardGameWar1/src/cardGameWar1/Card.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
CardGame
JRE System Library [JavaSE-11]
src
cardGame
CardGameWar1
homework-week3
mysql-java
week 5
App.java
Deck.java
*Card.java
Player.java
package cardGameWar1;
//information about class
public class Card {
//Attributes
private int suitNum = 0;
private int value = 0;
private Suit suit = null;
private Face face = null;
private String name = "";
//Constructors
public Card() {}
public Card(int value, int suitNum) {
this.value = value;
this.suitNum = suitNum;
finishUp();
}
//Methods
public void finishUp() {
int x = this.suitNum;
int y = this.value;
//convert suitNum to suit
Suit a = Suit.get(x);
//convert value to face
Face b = Face.get(y);
Writable
Smart Insert
109 : 1 : 1763
Type here to search
74°F
ENG
10:04 AM
9/7/2022


eclipse-workspace - CardGameWar1/src/cardGameWar1/Card.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
CardGame
JRE System Library [JavaSE-11]
src
cardGame
CardGameWar1
homework-week3
mysql-java
week 5
App.java
Deck.java
*Card.java
Player.java
//make the name
String z = b + " of " + a + "S";
this.suit = a;
this.face = b;
this.name = z;
}
//https://stackoverflow.com/questions/609860/convert-from-enum-ordinal-to-enum-type
enum Suit {HEART, DIAMOND, SPADE, CLUB;
public static Suit get(int index){
return Suit.values()[index];
}
}
public enum Face {Zero, One, Two, Three, Four, Five, Six, Seven, Eight, Nime, Ten, Jack, Queen, King, Ace;
public static Face get(int index){
return Face.values()[index];
}
}
//Getters & Setters
public int getSuitNum() {
return suitNum;
}
public void setSuitNum(int suitNum) {
this.suitNum = suitNum;
}
public int getValue() {
return value;
}
public void setValue(int value) {
Writable
Smart Insert
109 : 1 : 1763
Type here to search
74°F
ENG
10:05 AM
9/7/2022

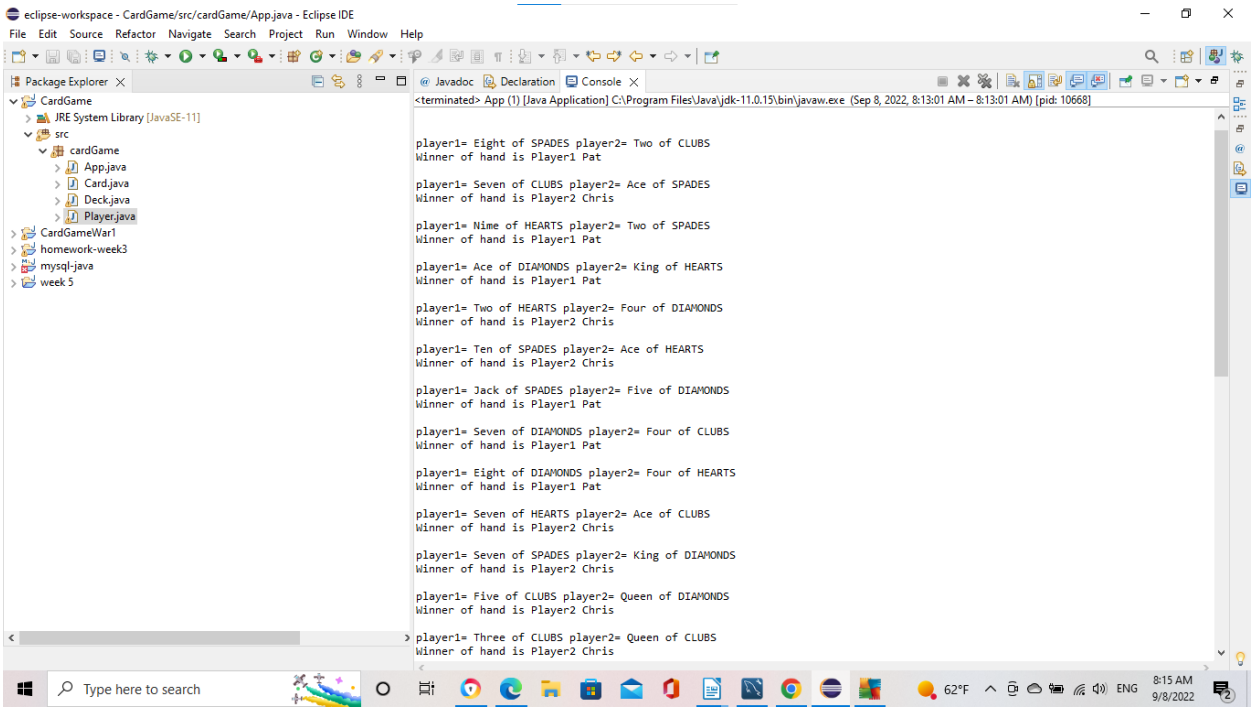

eclipse-workspace - CardGameWar1/src/cardGameWar1/Card.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
CardGame
JRE System Library [JavaSE-11]
src
cardGame
CardGameWar1
homework-week3
mysql-java
week 5
App.java
Deck.java
*Card.java
Player.java
this.suitNum = suitNum;
}
public int getValue() {
return value;
}
public void setValue(int value) {
this.value = value;
}
public Suit getSuit() {
return suit;
}
public void setSuit(Suit suit) {
this.suit = suit;
}
public Face getFace() {
return face;
}
public void setFace(Face face) {
this.face = face;
}
public String getName() {
return name;
}
public void setName(String name) {
this.name = name;
}
}
Writable
Smart Insert
109 : 1 : 1763
Type here to search
74°F
ENG
10:06 AM
9/7/2022

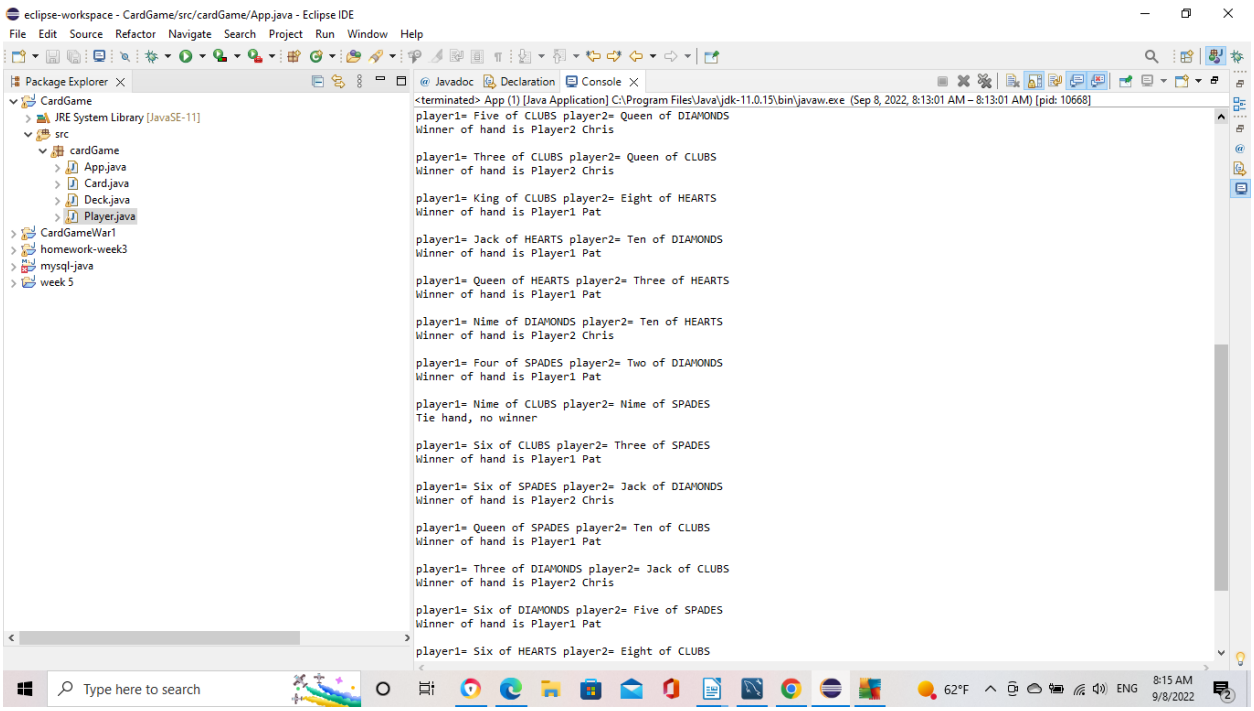

eclipse-workspace - CardGame/src/cardGame/Player.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
CardGame
JRE System Library [JavaSE-11]
src
cardGame
CardGameWar1
homework-week3
mysql-java
week 5
App.java
Deck.java
Card.java
Player.java
1 package cardGame;
3 import java.util.LinkedList;
4 import java.util.List;
6 public class Player {
9 //Attributes
11 private String name = "";
12 private int score = 0;
13 private LinkedList<Card> hand = new LinkedList<Card>();
15 //Constructors
17 public Player() {}
19 public Player(String inName) {
20 this.name = inName;
21 }
24 //Methods
26 //add a card to the hand
27 public void addACard(Card card) { this.hand.add(card);}
29 //take a card off the hand
30 public Card getACard() {
31 Card nextCard = this.hand.getFirst();
32 this.hand.removeFirst();
33 return nextCard;
34 }
36 //print out hand
37 public void showPlayerHand() {
38 //List<Card> answer = new LinkedList<Card>();
39 //Loop through hand
Writable
Smart Insert
71 : 10 : 1399
Type here to search
74°F
ENG
10:07 AM
9/7/2022

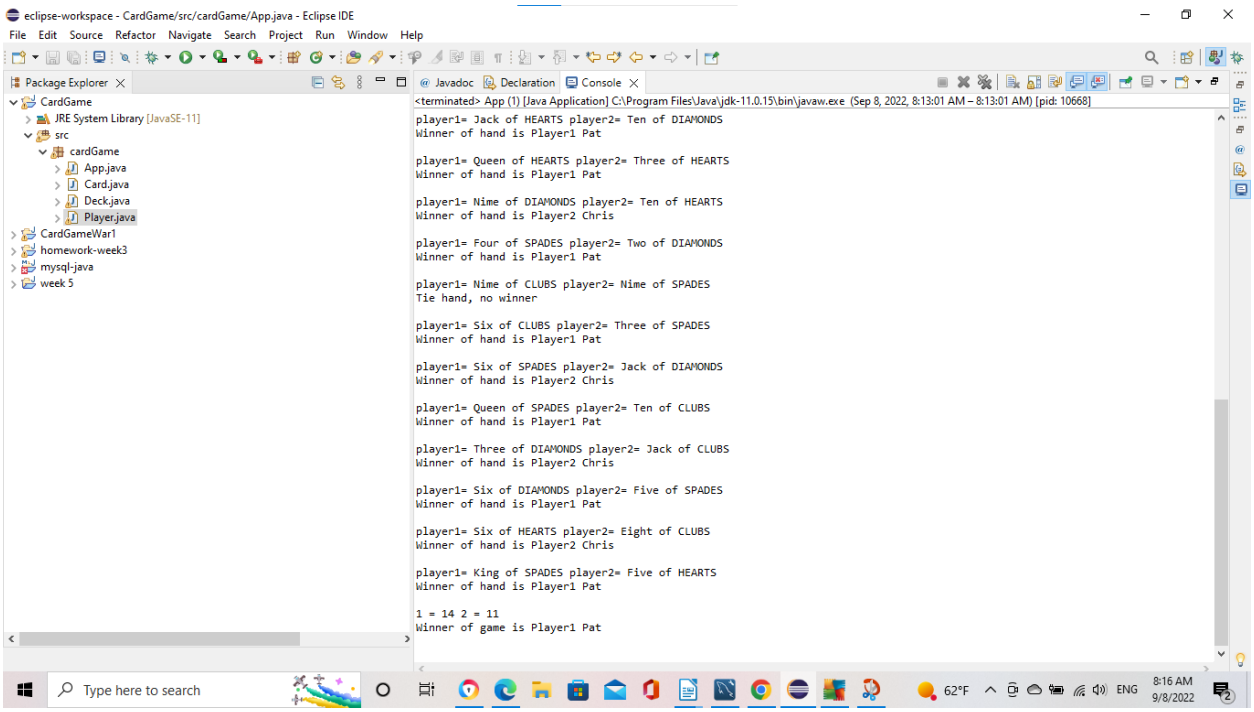

eclipse-workspace - CardGame/src/cardGame/Player.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Project Explorer
CardGame
JRE System Library [JavaSE-11]
src
cardGame
CardGameWar1
homework-week3
mysql-java
week 5
App.java
Deck.java
Card.java
Player.java
//List<Card> answer = new LinkedList<Card>();
//Loop through hand
for(int i = 0; i < hand.size(); i++){
String next = hand.get(i).getName();
System.out.println(next);
}
//print out player info
public void showPlayerInfo() {
String playerInfo = "Name " + name + " Score " + score;
System.out.println(playerInfo);
}
//Getters & Setters
public String getName() {
return name;
}
public void setName(String name) { //validate strings
this.name = name;
}
public int getScore() {
return score;
}
public void setScore(int score) { //validatee?
App.ValidateInt();
this.score = score;
}
public LinkedList<Card> getHand() {
return hand;
}
Writable
Smart Insert
71 : 10 : 1399
Type here to search
74°F
ENG
10:07 AM
9/7/2022



Screenshots of Running Application:







URL to GitHub Repository: