6.1:

1. public class Card {

String suit, name;

int points;

Card(int n1, int n2) {

suit = getSuit(n1);

name = getName(n2);

points = getPoints(name);

}

public String toString() {

return "The " + name + " of " + suit;

}

public String getName(int i) {

if (i == 1) return "Ace";

if (i == 2) return "Two";

if (i == 3) return "Three";

if (i == 4) return "Four";

if (i == 5) return "Five";

if (i == 6) return "Six";

if (i == 7) return "Seven";

if (i == 8) return "Eight";

if (i == 9) return "Nine";

if (i == 10) return "Ten";

if (i == 11) return "Jack";

if (i == 12) return "Queen";

if (i == 13) return "King";

return "error";

}

public int getPoints(String n) {

if (n.equals("Jack") || n.equals("Queen") || n.equals("King") || n.equals("Ten"))

return 10;

if (n.equals("Ace"))

return 11;

return Integer.parseInt(n);

}

public String getSuit(int i) {

if (i == 1) return "Diamonds";

if (i == 2) return "Clubs";

if (i == 3) return "Spades";

if (i == 4) return "Hearts";

return "error";

}

}

2. import java.util.Collections;

import java.util.List;

import java.util.ArrayList;

public class Deck {

Card[] cardArray = new Card[52];

private List<Card> cardList;

Deck() {

int suits = 4;

int cardType = 13;

int cardCount = 0;

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

for (int j = 1; j <= cardType; j++) {

cardArray[cardCount] = new Card(i, j);

cardCount++;

}

}

cardList = new ArrayList<Card>(List.of(cardArray));

}

public void print() {

for (Card card : cardArray) {

System.out.println(card);

}

}

public void shuffle() {

Collections.shuffle(cardList);

cardList.toArray(cardArray); // Refill cardArray with shuffled cards

}

}

3. import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class Hand {

private List<Card> hand;

private int points;

public Hand() {

hand = new ArrayList<>();

points = 0;

}

public void addCard(Card card) {

hand.add(card);

points += card.points;

if (points > 21) {

for (Card c : hand) {

if (c.points == 11) {

c.points = 1;

points -= 10;

if (points <= 21) break;

}

}

}

}

public int getPoints() {

return points;

}

public void display() {

for (Card card : hand) {

System.out.println(card);

}

System.out.println("Total points: " + points);

}

}

4. import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Deck deck = new Deck();

deck.shuffle(); // Shuffle the deck

Scanner scanner = new Scanner(System.in);

Hand playerHand = new Hand();

Hand dealerHand = new Hand();

// Deal initial cards

playerHand.addCard(deck.cardArray[0]);

deck.cardArray[0] = null; // Remove the card from deck

playerHand.addCard(deck.cardArray[1]);

deck.cardArray[1] = null; // Remove the card from deck

dealerHand.addCard(deck.cardArray[2]);

deck.cardArray[2] = null; // Remove the card from deck

dealerHand.addCard(deck.cardArray[3]);

deck.cardArray[3] = null; // Remove the card from deck

boolean playerBust = false;

while (playerHand.getPoints() <= 21 && playerHand.hand.size() < 5) {

playerHand.display();

System.out.println("Do you want another card? (yes/no)");

String response = scanner.next();

if (response.equalsIgnoreCase("yes")) {

// Deal next card

for (int i = 4; i < 52; i++) {

if (deck.cardArray[i] != null) {

playerHand.addCard(deck.cardArray[i]);

deck.cardArray[i] = null;

break;

}

}

} else {

break;

}

}

if (playerHand.getPoints() > 21) {

playerBust = true;

}

// Dealer's turn

while (dealerHand.getPoints() < 17) {

for (int i = 4; i < 52; i++) {

if (deck.cardArray[i] != null) {

dealerHand.addCard(deck.cardArray[i]);

deck.cardArray[i] = null;

break;

}

}

}

dealerHand.display();

if (playerBust) {

System.out.println("Player busts! Dealer wins.");

} else {

if (dealerHand.getPoints() > 21 || playerHand.getPoints() > dealerHand.getPoints()) {

System.out.println("Player wins!");

} else if (playerHand.getPoints() < dealerHand.getPoints()) {

System.out.println("Dealer wins!");

} else {

System.out.println("It's a tie!");

}

}

}

}