```java
import java.util.ArrayList;
import java.util.Objects;
import java.util.Scanner;

public class Game {

    //GET USERS NAME
    static String getUser(String currentUser) {

        Scanner name = new Scanner(System.in); //new instance of scanner
        System.out.print("\nPlease tell me your name to continue: ");

        currentUser = String.valueOf(name.nextLine());
        return currentUser; //returns the currentUser / users name
    }

    //GAME METHOD
    static String game(String currentScore) {

        //instance variables
        Card computerCard; //this will store the current card held by the
computer
        Card selectedCard = null; //this will hold the value of the
selected card
        int selectedCardIndex = 0; //this will be used as an index to
select the users card
        int roundNumber = 1; //saves round number
        boolean play = true; //if false game will stop

        //ARRAYLIST FOR STORING GAME HISTORY
        ArrayList <String> history = new ArrayList<>();
        /*
        * roundNumber, userDeck, computerCard, cardPlayed, score*/

        //SETTING UP ARRAYS FOR USER AND COMPUTERS DECKS
        Deck deck = new Deck(); //new instance of deck

        ArrayList<Card> mainDeck = new ArrayList<>(); //this will be used
to hold the values of the array
        ArrayList<Card> userDeck = new ArrayList<Card>(); //ArrayList for
holding users cards in hand
        Card[] userCard = new Card[1]; //holds the card the user selects
        ArrayList<Card> computerDeck = new ArrayList<>(); //used for
holding the computers deck, will only add one card and remove used card


        //ADD ALL CARDS TO mainDeck
        for(int i = 0; i < 52; i++){
            mainDeck.add(deck.deal()); //add next value of deck array
            //System.out.println(mainDeck.get(i));
        }

        System.out.println("\n\n");

        //ADD 5 CARDS TO USER ARRAY
        for (int i = 0; i < 5; i++) { //for loop will add 5 cards to the
users deck when game starts
            userDeck.add(mainDeck.get(0)); //THIS WILL ADD THE REMOVED CARD
TO THE USERS CARDS!!
            mainDeck.remove(0); //Remove card added to user deck from main
deck
```

```
        }



        int score = 0; //used for keeping track of score
        String convScore = ""; //used to make score a string

        //USER INPUTS FOR GAME
        while (play) { //while play = true

            System.out.println("===Round " + roundNumber + "!===");
            history.add(String.valueOf(roundNumber)); //adds round to game
history

            //OUTPUT USER DECK
            System.out.println("\nHere are your cards: ");
            for (int i = 0; i < 5; i++) {
                System.out.println((i + 1) + ". " + userDeck.get(i));
//prints out the users deck, NEED TO CHANGE TO MAKE MORE PRESENTABLE
                history.add(String.valueOf(userDeck.get(i))); //adds each
card from deck to history
            }


            //GENERATE COMPUTERS CARD
            System.out.print("\nThe computers card is: ");
            computerDeck.add(mainDeck.get(0)); //adds card to computer deck
            mainDeck.remove(0); //removes card from main deck
            computerCard = computerDeck.get(0); //stores value of first
card to card in array
            System.out.println(computerCard); //outputs computers card
            history.add(String.valueOf(computerCard)); //adds computers
card to game history

            //SCANNER FOR KEYBOARD INPUT


            //USERS INPUT
            boolean o = true;
            Scanner selectCard = new Scanner(System.in);
            while(o) {
                System.out.print("\nPlease select a card by typing a number
between 1 and 5: "); //MAY CHANGE THIS LINE
                selectedCardIndex = selectCard.nextInt(); //takes users
input to select their card from arraylist
                if ((selectedCardIndex == 1) || (selectedCardIndex == 2) ||
(selectedCardIndex == 3) || (selectedCardIndex == 4) || (selectedCardIndex
== 5)) {
                    selectedCard = userDeck.get(selectedCardIndex - 1);
//takes 1 away from user input to better match indexing
                    history.add(String.valueOf(selectedCard)); //adds users
selected card to history
                    o = false;
                } else {
                    System.out.println("INVALID INPUT! TRY AGAIN!");
                }
            }

            //CONFORMATION TEST ONLY
            System.out.print("\n You have selected: ");
            System.out.print(selectedCard + "\n");
```

```
            //SPLIT CARDS TO COMPARE SUIT AND RANK
            int userSelectedRank;
            int computerSelectedRank;

            String[] selectedCardSplit = selectedCard.toString().split(" of
"); //split users selected card to compare
            String[] computerCardSplit = computerCard.toString().split(" of
"); //splits computers card to compare

            //System.out.println(selectedCardSplit[0] + " " +
selectedCardSplit[1]);
            //System.out.println(computerCardSplit[0] + " " +
computerCardSplit[1]);

            //IF RANK == STRING VALUE THEN KEEP AS STRING VALUE
            //User
            if(Objects.equals(selectedCardSplit[0], "King")){
                userSelectedRank = 10; //sets value to value of the card
            } else if (Objects.equals(selectedCardSplit[0], "Queen")){
                userSelectedRank = 10; //sets value to value of the card
            } else if (Objects.equals(selectedCardSplit[0], "Jack")){
                userSelectedRank = 10; //sets value to value of the card
            } else if(Objects.equals(selectedCardSplit[0], "Ace")) {
                userSelectedRank = 1; //sets value to value of card
            } else{
                userSelectedRank = Integer.parseInt(selectedCardSplit[0]);
//converts selected rank to an integer
            }

            //Computer
            if(Objects.equals(computerCardSplit[0], "King") ||
Objects.equals(computerCardSplit[0], "Queen") ||
Objects.equals(computerCardSplit[0], "Jack")){
                computerSelectedRank = 10; //sets value to value of the
card
            } else if(Objects.equals(computerCardSplit[0], "Ace")) {
                computerSelectedRank = 1; //sets value to value of card
            } else{
                computerSelectedRank =
Integer.parseInt(computerCardSplit[0]); //converts computer rank to an int
            }

            //COMPARE CARDS //0 == rank, 1 == suit

            if (userSelectedRank + computerSelectedRank == 11) { //if total
score == 11
                System.out.println("\nThat's 11!! You have gained one
point!!!");
                score++; //add one to the score total
                userDeck.remove(selectedCardIndex - 1); //removes used card
from deck
                computerDeck.remove(0); //removes computers card

                //OUTPUT SCORE
                System.out.println("\nYOU'RE SCORE IS: " + score + "\n");
                history.add(String.valueOf(score)); //adds current score to
history

                //EXCHANGE CARD
                Scanner exchange = new Scanner(System.in); //new instance
```

```
of scanner to get user input
                System.out.print("\nWould you like to exchange one of your
cards for another? [y/n]: ");
                String uChoice = exchange.nextLine(); //read user  input
                System.out.println("\n");

                if((Objects.equals(uChoice, "y")) ||
(Objects.equals(uChoice, "Y"))) { //if user wants to exchange their card
                    //for loop to add deck
                    for (int i = 0; i < 4; i++) {
                        System.out.println((i+1) + ". " + userDeck.get(i) +
" ");
                    }

                    boolean p = true;

                    while(p) {
                        System.out.print("\nPlease choose a card [1-4]: ");

                        selectedCardIndex = exchange.nextInt();

                        if ((selectedCardIndex == 1) || (selectedCardIndex
== 2) || (selectedCardIndex == 3) || (selectedCardIndex == 4)) {
                            userDeck.remove(selectedCardIndex - 1);
                            if (!mainDeck.isEmpty()) { //checks is deck is
empty before grabbing new card
                                userDeck.add(mainDeck.get(0));
                                mainDeck.remove(0);
                                p = false;
                            }
                            else{
                                System.out.println("Invalid Input! Try
Again!");
                            }
                        }
                    }
                }

                //ADD NEW CARD TO USER DECK
                if(!mainDeck.isEmpty()) { //checks is deck is empty before
grabbing new card
                    userDeck.add(mainDeck.get(0)); //adds a new card from
deck
                    mainDeck.remove(0); //removes collected card
                }

            } else if ((userSelectedRank + computerSelectedRank != 11) &&
(!Objects.equals(selectedCardSplit[1], computerCardSplit[1]))){ //if total
!= 11 and suits not equal
                System.out.println("\nTHAT'S GAME OVER!");
                System.out.println("\nYour final total is = " + score +
"\n");
                history.add(String.valueOf(score)); //adds score to history

                System.out.print("\nWould you like to see the game history?
[y/n]");
                Scanner seeHistory = new Scanner(System.in); //gets user
input to see if they wish to see the game history
                String uInput = seeHistory.nextLine();
                int i = 0; //index for stepping through history
                if((uInput.equals("Y")) || (uInput.equals("y"))){
```

```
                    while (!history.isEmpty()){ //while history is not
empty

                        //show round
                        System.out.println("\n===Round " + history.get(i) +
"!==="); //prints round number
                        history.remove(i); //removes first value of history
array
                        //i++;

                        //show user deck
                        System.out.println("Your cards");
                        for(int uDeck = 0; uDeck < 5; uDeck++){ //loops 5
times to output all of users deck
                            System.out.println((uDeck + 1) + ". " +
history.get(i)); //prints out the users deck
                            history.remove(i); //removes first value of
history array
                            //i++;
                        }

                        //show computers card
                        System.out.println("The computer had: " +
history.get(i));
                        history.remove(i); //removes first value of history
array
                        //i++;

                        //show played card
                        System.out.println("You played the: " +
history.get(i));
                        history.remove(i); //removes first value of history
array
                        //i++;

                        //show score
                        System.out.println("The score for this round was "
+ history.get(i));
                        history.remove(i); //removes first value of history
array

                        System.out.print("\nPress [ENTER] to continue!");
                        Scanner endHistory = new Scanner(System.in); //when
user hits enter then the while statement should continue
                        String endHistoryInput = endHistory.nextLine();
//reads next line

                    }

                }


                //END GAME
                play = false; //stops the game
                convScore = String.valueOf(score); //convert score to
String
                currentScore = convScore; //set current score to String
                return currentScore; //return final score


            } else if (userSelectedRank + computerSelectedRank != 11) {
```

```
//if ranks are not equal to 11
                //if suit == suit
                if(Objects.equals(selectedCardSplit[1],
computerCardSplit[1])){
                    System.out.println("\nYou matched the suits!! Play
on!");
                    System.out.println("\n YOU'RE SCORE IS: " + score +
"\n");
                    history.add(String.valueOf(score)); //adds score to
history
                    userDeck.remove(selectedCardIndex - 1); //removes used
card from deck
                    computerDeck.remove(0); //removes computers card
                    if (!mainDeck.isEmpty()) { //checks is deck is empty
before grabbing new card
                        userDeck.add(mainDeck.get(0)); //adds a new card
from deck
                        mainDeck.remove(0); //removes collected card
                    }
                }

            }

            if (mainDeck.isEmpty()) { //if the deck is empty, end game
                System.out.println("\nGAME OVER! THE DECK IS EMPTY!");
                System.out.println("\nYour final score is: " + score +
"\n");

                System.out.print("\nWould you like to see the game history?
[y/n]");
                Scanner seeHistory = new Scanner(System.in); //gets user
input to see if they wish to see the game history
                String uInput = seeHistory.nextLine();
                int i = 0; //index for stepping through history
                if((uInput.equals("Y")) || (uInput.equals("y"))){
                    while (!history.isEmpty()){ //while history is not
empty

                        //show round
                        System.out.println("\n===Round " + history.get(i) +
"!==="); //prints round number
                        history.remove(i); //removes first value of history
array
                        //i++;

                        //show user deck
                        System.out.println("Your cards");
                        for(int uDeck = 0; uDeck < 5; uDeck++){ //loops 5
times to output all of users deck
                            System.out.println((uDeck + 1) + ". " +
history.get(i)); //prints out the users deck
                            history.remove(i); //removes first value of
history array
                            //i++;
                        }

                        //show computers card
                        System.out.println("The computer had: " +
history.get(i));
                        history.remove(i); //removes first value of history
array
```

```
                        //i++;

                        //show played card
                        System.out.println("You played the: " +
history.get(i));
                        history.remove(i); //removes first value of history
array
                        //i++;

                        //show score
                        System.out.println("The score for this round was "
+ history.get(i));
                        history.remove(i); //removes first value of history
array


                        System.out.print("\nPress [ENTER] to continue!");
                        Scanner endHistory = new Scanner(System.in); //when
user hits enter then the while statement should continue
                        String endHistoryInput = endHistory.nextLine();
//reads next line

                    }

                }

                //END GAME
                play = false; //stops the game
                convScore = String.valueOf(score); //convert score to
String
                currentScore = convScore; //set current score to String
                return currentScore; //return final score
            }
            roundNumber++; //adds one to round number

        } return currentScore; //when game ends it returns the score the
user got


    }
}
```