In this project we will be writing a program to play blackjack (or 21). For those of you who are unfamiliar with the game, Blackjack is a card game where each card has a value:

- Numbered cards have the same values as their numbers
- Jacks, Queens, and Kings have a value of 10
- Aces have a value of 1 or 11

In our game we will have two players the user and the dealer. To start each hand the user and dealer are dealt an initial two card hand with the option of drawing cards to bring the total value to 21 or less without exceeding it, so that the dealer will lose by having a lesser hand than the player or by exceeding 21.

In each hand the game proceeds as follows:

1) The deck is shuffled.
2) The user and dealer are dealt 2 cards each -- so that the first card goes to the user, the second to the dealer, the third to the user, and the fourth to the dealer (the dealers fourth card will be hidden -- so only print out the first card).
3) Give the user the choice to hit (draw a card) or stand (end their turn).
4) Repeat 3) until the user chooses to stand or until they bust (go over 21).
5) If the user busts the hand is over and the dealer wins.
6) If the user doesn't bust the dealer continues to draw cards until she busts or her hand exceeds 17.
7) The player with the greatest hand, not exceeding 21, wins.

The Classes:

To complete this project you must write 4 classes:

1) Card - this class will represent individual cards
2) Deck - this class will represent the entire deck
3) Player - this class will represent both the user and dealer
4) Blackjack - this class will hold the main function, and will control the flow of the game

The following gives the specs (data, and methods) for each of the four classes -- feel free to add other data/methods to your classes, the specs below define the minimum.
The Card Class:

```java
public class Card {

    private int num; // num will represent the number on the card or
                    // the face.
                    // 1 = Ace
                    // 2 = 2
                    // 3 = 3
                    // ...
                    // 10 = 10
                    // 11 = Jack
                    // 12 = Queen
                    // 13 = King

    private char suit; // suit will be one of 4 letters:
                        // C for clubs,
                        // H for hearts,
                        // D for diamonds,
                        // or S for spades

    // Constructor
    public Card(int num, char suit) {
        // Initialize the private data with the parameters passed in
        ...
    }

    public boolean isAce() {
        // Return true if num == 1 false otherwise
        ...
    }

    public int getValue() {
        // Returns the blackjack value of the card
        // if the card is 2 thru 10 return num
        // if the card is a Jack, Queen or King return 10
        // if the card is an Ace return 11
        ...
    }

    public String toString() {
        // Return the card as a string. For example, if the card was a
        // king of diamonds return KD
        ...
    }

}```
The Deck Class:

```java
import java.util.Random; // Import Random to help shuffle the deck

public class Deck {

    private Card[] deck; // Array of cards to store all 52 cards
    private int nextCard; // Index of the next card to be dealt

    // Constructor
    public Deck() {
        // Set deck to a new Card array of size 52
        // Set nextCard to 0
        // Call initializeDeck
        ...
    }

    private void initializeDeck() {
        // Note that this is a private function
        // Create all 52 cards and add them to the deck
        // Hint: use a nested for-loop (a for-loop inside a for-loop)
        ...
    }

    public void shuffleDeck() {
        // Shuffle the deck:
        // Create a new random number generator
        // Randomly select two indexes and swap the
        // cards at these two indexes do this a
        // thousand times and the deck should be
        // well shuffled.
        // Also set nextCard back to 0
        ...
    }

    public Card nextCard() {
        // Return the next card in the deck
        // Don't forget to increment nextCard
        ...
    }
}
```
The Player Class

```java
public class Player {

    private Card[] hand; // Array of cards to hold the players hand
    private int emptyIndex; // Holds the index of the next empty
                           // index in the hand
    private int numAces; // Holds the number of aces

    // Constructor
    public Player() {
        // Set hand to a new card array large enough
        // hold any blackjack hand -- 20 should do it
        // Set emptyIndex to 0
        // Set numAces to 0
         ...
    }

    public void addCard(Card card) {
        // Add the card to the hand (at the emptyIndex index)
        // Don't forget to increment emptyIndex
        // If the card is an ace increment numAces
         ...
    }

    public Card firstCard() {
        return hand[0]; // this is used to display
                         // the dealers initial hand
                         // this method is complete
    }

    public int getValue() {
        // Return the blackjack value of the hand:
        // Create a sum variable and initialize it to 0
        // Loop though your hand and for each
        // card call getValue() and add that to sum
        // Remember that an ace can be either a 1 or 11,
        // but when the getValue() function only
        // returns an 11 for an ace. So if your sum
        // is greater than 21, then start turning your
        // aces into 1's instead of 11's. That is,
        // create a while loop that sets sum
        // to sum - 10, continue doing this until sum
        // is less than 22 or you run out of aces.
    }
}
```
public String toString() {
    // Return hand as a string -- the string should show
    // each card in your hand followed by a space (use the cards toString).
    // Then append the total blackjack value to the string
    // use getValue()
    ...
}

// Hint - you shouldn't loop through the entire
// hand array -- much of the hand array will
// be empty indexes
...
The Blackjack Class:

```java
import java.util.Scanner; //Import Scanner for user input

public class Blackjack {

    public static void main(String[] args) {

        // The main method controls the basic flow of the game
        // First create a Scanner object to get user input
        // Create an instance of the deck class
        // Print a welcome message to the user
        // Then code the following steps:

        // 1) Create two instances of the player class
        //    one for the user and one for the dealer.

        // 2) Shuffle the deck

        // 3) Print the main menu:
        //    "Please choose from the following"
        //    "1) Start Game"
        //    "2) Exit"

        // 4) Get user input -- if it's 2 display a
        //    message and end the program (use break)

        // 5) Deal the hands (two cards to the user
        //    two to the dealer)

        // 6) Show the dealers first card

        // 7) Show the users hand -- use toString

        // 8) Display hit/stand menu to user
        //    "Please choose from the following:
        //    " 1) Hit
        //    " 2) Stand

        // 9) Get the users input
        //    if it's a 1, then deal the user another card

        // 10) Repeat steps 7 to 9 until the user enters a 2 or
             the user busts (goes over 21).

        // 11) If the user busts, print a message
    }
}
```
// "Sorry you lose" and go back to step 1)

// 12) Continue dealing the dealer cards until
// her hand is greater than or equal to 17

// 13) Print the dealers hand -- use toString

// 14) Determine the winner
// if the dealer busted, the user wins
// if the dealers hand is greater than the user, the dealer wins
// if the two hands are equal they push
// else the dealer wins

// 15) Display the winner, and go back to step 1

}
Sample Output:

Welcome to the CS 152 Blackjack Table

Please select from the following:
1) Start Game
2) Exit
1
The dealer is showing: 3C
Your hand is:
2C 8C , Total = 10
Please select from the following:
1) Hit
2) Stand
1
Your hand is:
2C 8C 9C , Total = 19
Please select from the following:
1) Hit
2) Stand
2
The dealer has:
3C 6H AS , Total = 20
Sorry you lose

Please select from the following:
1) Start Game
2) Exit
1
The dealer is showing: 6S
Your hand is:
AH 4H , Total = 15
Please select from the following:
1) Hit
2) Stand
1
Your hand is:
AH 4H QS , Total = 15
Please select from the following:
1) Hit
2) Stand
1
Your hand is:
AH 4H QS 8S , Total = 23
Sorry you bust

Please select from the following:
1) Start Game
2) Exit
1
The dealer is showing: 3C
Your hand is:
QC 5C , Total = 15
Please select from the following:
1) Hit
2) Stand
2
The dealer has:
3C AC 2H 7D 4C , Total = 17
Sorry you lose

Please select from the following:
1) Start Game
2) Exit
1
The dealer is showing: 9H
Your hand is:
7S 5D , Total = 12
Please select from the following:
1) Hit
2) Stand
1
Your hand is:
7S 5D AH , Total = 13
Please select from the following:
1) Hit
2) Stand
1
Your hand is:
7S 5D AH 4H , Total = 17
Please select from the following:
1) Hit
2) Stand
2
The dealer has:
9H 7D 10H , Total = 26
Dealer busts -- You WIN!!!
Please select from the following:
1) Start Game
2) Exit

1
The dealer is showing: 5S
Your hand is:
4C 9S , Total = 13
Please select from the following:
1) Hit
2) Stand

1
Your hand is:
4C 9S 5C , Total = 18
Please select from the following:
1) Hit
2) Stand

2
The dealer has:
5S AC KC 6S , Total = 22
Dealer busts -- You WIN!!!

Please select from the following:
1) Start Game
2) Exit

1
The dealer is showing: 4H
Your hand is:
7S 9C , Total = 16
Please select from the following:
1) Hit
2) Stand

2
The dealer has:
4H 5S 8D , Total = 17
Sorry you lose

Please select from the following:
1) Start Game
2) Exit

1
The dealer is showing: 9S
Your hand is:
4H 3D , Total = 7
Please select from the following:
1) Hit
2) Stand
1
Your hand is:
4H 3D 6D , Total = 13
Please select from the following:
1) Hit
2) Stand
1
Your hand is:
4H 3D 6D 9C , Total = 22
Sorry you bust

Please select from the following:
1) Start Game
2) Exit
1
The dealer is showing: 10H
Your hand is:
2S 2D , Total = 4
Please select from the following:
1) Hit
2) Stand
1
Your hand is:
2S 2D QD , Total = 14
Please select from the following:
1) Hit
2) Stand
1
Your hand is:
2S 2D QD AD , Total = 15
Please select from the following:
1) Hit
2) Stand
1
Your hand is:
2S 2D QD AD 6S , Total = 21
The dealer has:
10H 9D , Total = 19
You WIN!!!!
Please select from the following:
1) Start Game
2) Exit

2
Thanks for playing