**You have the option of doing Number 1 or 2 for this assignment**

Assignment # 6: Functions and Methods

1. You have name your file Blackjack.java.

Create a game that allows the user to play Black Jack. Black Jack is a widely popular casino game. How to play:

Each player bets a certain amount of money each hand, say $10.

The player is then dealt 2 cards face up and the dealer is dealt one card face down (that you can’t see) and one card face up. The user then completes their hand before the dealer does by taking as many more cards as they want. If they go over 21 then they are finished and they automatically lose.

The dealer than completes their hand by flipping over the facedown card and taking more cards. The dealer always stops once they have at least 17 in their hand.

The hand with the highest total wins as long as it doesn't exceed 21; a hand with a higher total than 21 is said to *bust* or *too many*. Cards 2 through 10 are worth their face value, and face cards (jack, queen, king) are all worth 10. An ace's value is 11.

Each player's goal is to beat the dealer by having the higher, unbusted hand. Note that if the player busts they lose, even if the dealer also busts. If both the player and the dealer have the same point value, it is called a "push", and neither player nor dealer wins the hand. Each player has an independent game with the dealer, so it is possible for the dealer to lose to some players but still beat the other players in the same round.

There are other additional rules that casinos have, but for our purposes we will go with these rules. The user should start with $500 and the game starts by allowing the player to place their bet. The player should be able to play as long as they want to unless they have lost all their money, then the game ends.

Each time a player gets a card, the card should be displayed. (You should have a method called **deal()** that deals one random card.) It is alright for the same card to be dealt more than once in a hand because casinos often deal with 6 to 8 packs of cards together.

Modification: If the user gets BlackJack, meaning their first two cards consist of an Ace and a facecard for a total of 21, then the person gets 1.5 times their bet for winning.

Also, only allow the user to bet between $10 and $100 per hand.

1. You have to name your file Nim.java.

The game of Nim starts with a random number of stones between 15 and 30. Two players alternate turns and on each turn may take either 1, 2 or 3 stones from the pile. The player is forced to take the last stone loses. Create a Nim application that allows the user to play against the computer. In this version of the game, the application generates the number of stones to begin with, the number of stones the computer takes, and the user goes first. The Nim application code should:

* Prevent the user and the computer from taking an illegal number of stones. For example, neither should be allowed to take three stones when there are only 1 or 2 left.
* Include an isValidEntry() method to check user input.
* Include a drawStones() method that generates a random number from 1 to 3 for the number of stones the computer draws.
* Include separate methods to handle the user’s turn and the computer’s turn.

Application output should look similar to:

**There are 22 stones. How many would you like? 3  
There are 19 stones. The computer takes 2 stones.  
There are 17 stones. How many would you like? 3  
There are 14 stones. The computer takes 2 stones.  
There are 12 stones. How many would you like? 3  
There are 9 stones. The computer takes 3 stones.  
There are 6 stones. How many would you like? 3  
There are 3 stones. The computer takes 3 stones.  
The player beats the computer!**