Inquiry Question
Can you accurately simulate a game of Blackjack using Python's random module?

Name: _____ Date: _____

General Instructions
Blackjack is a popular card game in which your objective as a player is to beat the dealer. Your goal is to reach a total hand value of 21, or a value higher than the dealer's without busting (going over 21). Using the Python's "random" module, as well as functions and loops, can you write a Blackjack computer program which allows a user (you), to play against a computerized dealer? Is this an accurate representation of the game of Blackjack?

Materials you'll need:

- Pencil
- Computer

Project submission:

Submit the completed pages of this project as well as the .py code file for your Blackjack program.

Design Specifications

- ❏ Output the player's initial hand of 2 cards that is dealt.
- ❏ Output the dealer's initial hand that is dealt (but keep one card as hidden).
- ❏ Ask the player to stand or hit. If they hit, add a random card to their hand.
- ❏ If the player "busts" (goes over 21) while hitting, the game is over and they lose.
- ❏ Once the player "stands" (stops hitting), the dealer must hit while they have 16 or less, and must stand on 17 through 21.
- ❏ If the dealer busts, the player wins.
- ❏ The player wins when their hand totals higher than the dealer's hand, or they have 21 or less when the dealer busts. This logic should be within a function.
- ❏ If the player and dealer both end with 21, it is a tie.
- ❏ You must use the Python "random" module, at least 2 functions, and at least 1 loop.

Bonus Options

- ❏ Add the ability to play multiple games without exiting and restarting. Keep an overall score.
- ❏ Allow more than one player to play against the dealer at once.

Hints and Resources

Here is a short example of how your program might look when you run it in the console.

```
>>> %Run blackjack.py
 The dealer has two cards: 2, and a hidden card
 You are at 9
 Do you want to stand, or hit? hit
 You are at 16
 Do you want to stand, or hit? stand
 Dealer hits and is now at 9
 Dealer hits and is now at 19
 Dealer ends turn with 19
 The dealer has a greater hand value, you lose.
```

The player wins when:

- The dealer busts (goes over 21).
- By drawing a hand value of 21 on your first two cards, when the dealer does not.
- By having a hand value greater than the dealer's once the dealer is done hitting.

The player loses when:

- The player busts.
- The round is over, and the dealer has a greater hand value than the player.

How to find a hand's total value?

1. 2 through 10 count as face value (2 counts as two, 9 counts as nine).
2. Face cards (Jack, Queen, King) count as 10.
3. Ace can count as 1 or an 11 depending what helps the hand the most.

For this program, keep it simple and generate a random number between 1 (inclusive) and 11 (inclusive).

Questions

What parameters will you pass to the function *randrange* to generate a random card?

Using pseudocode, write a function to detect whether the player or the dealer has won once the round is over.

Is generating a random number an accurate representation of drawing a card from a deck? Why or why not?