



3 Classes exist in this program.

Player, CardDeck, Blackjack

Player class is for holding information about the player since they will be multiple player in the game so it's make sense to make a class to hold those information individually.

CardDeck class is for holding the method related to getting the card and store a list of card.

Blackjack class is for containing all the method to operate the game and include all the attributes need for operating the game.

Method in Player class:

 __init__ : Set up attributes for player object Method

in CardDeck class:

 __init__(self): Set up attributes for deck object

shuffle(self): Shuffle the deck draw_cards(self,n):

Return n card drawn from deck.

Method in Blackjack class:

 __init__(self): Set up attributes for blackjack game object

start(self): Start the game(Draw card and adjust value for all players including computer)

 status_update(player): Check whether current status of a player is legit or not.

value_cal(player): Calculate the value in the hand of a player.

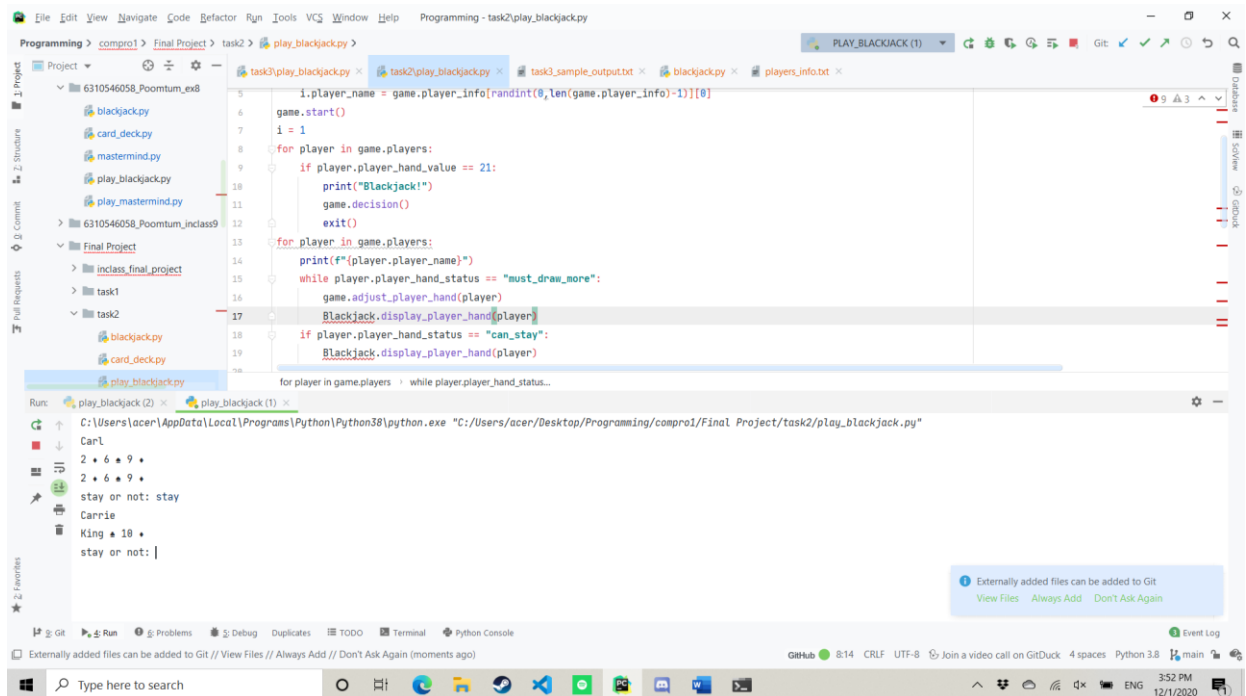
 ace_cal(player , num_ace): Calculate whether the ace in the hand should be 1 or 11.

`adjust_player_hand(self, player)`: Adjust the hand of player (Draw more) according to the status of the player.

`display_player_hand(player)` : Show the hand of the player to the screen.

`decision(self)`: Check who is winning and print out result + plus the point to the winner.

Sample:



The screenshot shows an IDE with a Python script for a Blackjack game. The script defines a `game` object with methods `start()`, `adjust_player_hand()`, `display_player_hand()`, and `decision()`. The `start()` method initializes the game, including player names and hands. The `adjust_player_hand()` method draws cards until the player's hand value is 21 or less. The `display_player_hand()` method prints the player's hand. The `decision()` method checks for a blackjack and prints the result.

```
1. player_name = game.player_info[randint(0, len(game.player_info)-1)][0]
2. game.start()
3. i = 1
4. for player in game.players:
5.     if player.player_hand_value == 21:
6.         print("Blackjack!")
7.         game.decision()
8.         exit()
9.     for player in game.players:
10.        print(f"{player.player_name}")
11.        while player.player_hand_status == "must_draw_more":
12.            game.adjust_player_hand(player)
13.        Blackjack.display_player_hand(player)
14.        if player.player_hand_status == "can_stay":
15.            Blackjack.display_player_hand(player)
16.        for player in game.players:
17.            while player.player_hand_status == "must_draw_more":
18.                game.adjust_player_hand(player)
19.            Blackjack.display_player_hand(player)
```

The Run console shows the output of the script:

```
Run: play_blackjack (2) x play_blackjack (1) x
C:\Users\acer\AppData\Local\Programs\Python\Python38\python.exe "C:/Users/acer/Desktop/Programming/compro1/Final Project/task2/play_blackjack.py"
Carl
2 * 6 * 9 *
2 * 6 * 9 *
stay or not: stay
Carrie
King * 18 *
stay or not: |
```

