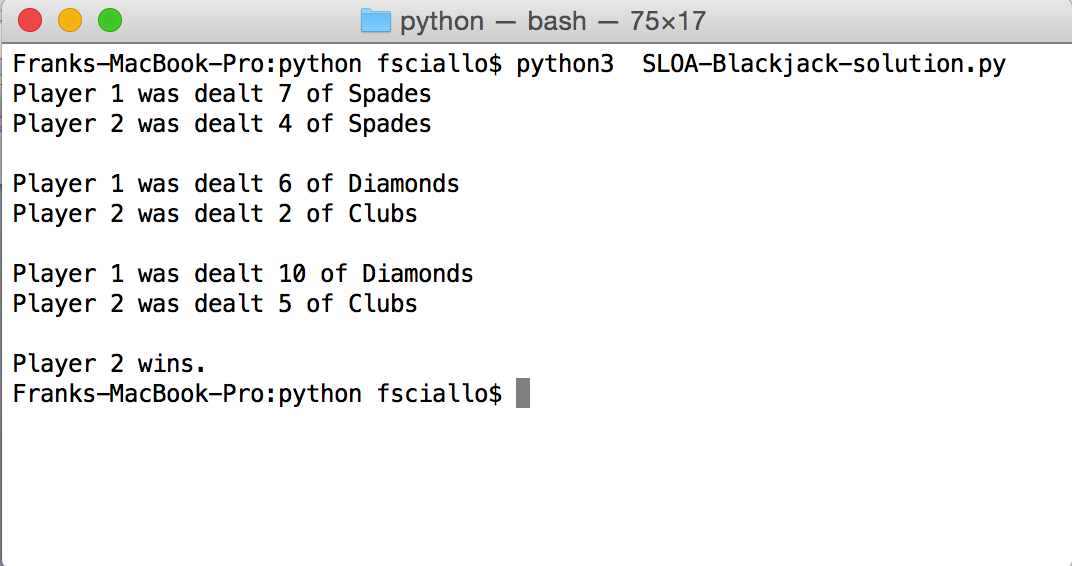
Program requirements:

The script below, written in python 3, should simulate a game of Blackjack between 2 players.

When executed, the script should display the cards dealt to each player until a player wins or both players lose.

No input is required from the user and the output should look similar to the following:



The code below will not run error free in the python interpreter.

Your task is to debug the given code making the change or changes required to deliver a functioning python script.

You may need to adjust the program to meet the requirements provided above.

You may need to provide additional documentation to ensure the readability of the script by future maintainers of the code.

#!/local/python3

MAX = 21

# main function

def main():

# Local variables

hand1 = 0

hand2 = 0

deck = create\_decks()

while hand1 <= MAX and hand2 <= MAX:

card1, value1 = deck.popitem()

hand1 = update\_hand\_value(hand1, value1, card1)

card2, value2 = deck.popitem()

hand2 = update\_hand\_value(hand2, value2, card2)

print('Player 1 was dealt', card1)

print('Player 2 was dealt', card2)

print()

# Determine the winner.

if hand1 > MAX and hand2 > MAX:

print("There is no winner.")

elif hand1 > 21:

print("Player 2 wins.")

else:

print("Player 1 wins.")

def create\_deck():

# Set up local variables

suits = ['Spades','Hearts','Clubs','Diamonds']

special\_values = {'Ace':1, 'King':10, 'Queen':10, 'Jack':10}

numbers = ['Ace', 'King', 'Queen', 'Jack']

for i in range(2,11):

numbers.append(str(i))

# Initialize deck

deck = {}

for suit in suits:

for num in numbers:

# Values 2-10.

if num.isnumeric():

deck[num + ' of ' + suit] = int(num)

return deck

def update\_hand\_value(hand, value, card):

if not card.startswith('Ace'):

return hand+value

# Adding 11 would cause to go over the maximum.

elif hand > 10:

# Value is 1 by default.

return hand + value

else:

return hand + 11

# Call the main function.

main()