AI Assignments

ARTIFICIAL INTELLIGENCE

2024

Name: BASIT IQBAL

Reg No: FA21-BSE-050

Submitted To:
Mam Zaib un Nisa

Comsats University Islamabad Abbottabad Campus

INTRODUCTION

SUBMITTED BY:

- Basit Iqbal (Fa21-bse-050)

REGISTRATION NO:

- (Fa21-bse-050)

Course Name:

- Artificial Intelligence

Submitted to:

- Mam Zaib un Nisa

University:

- Comsats University Islamabad Abbottabad Campus

Submission Date:

- 13th March 2024

Assignment Number:

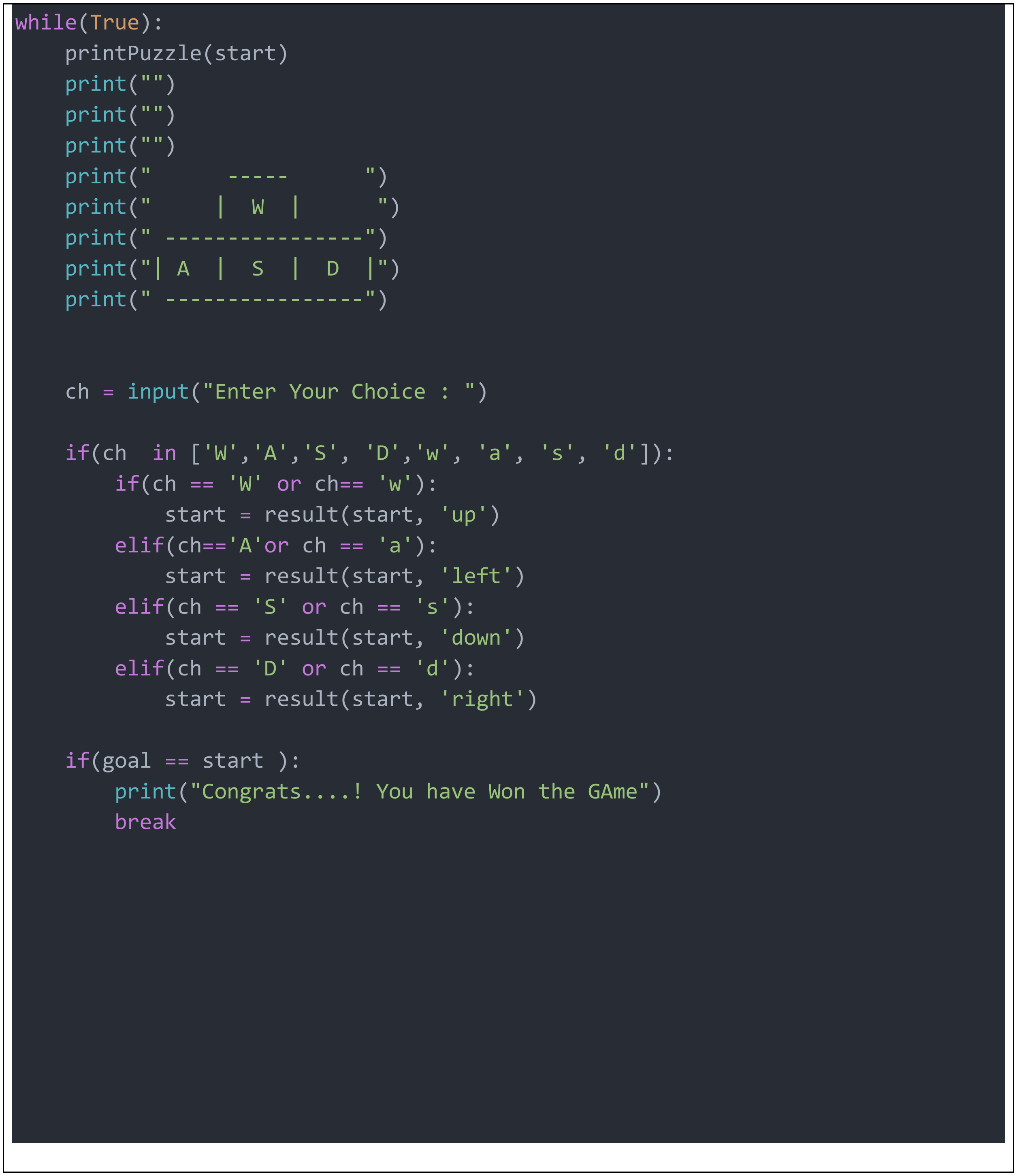
- Two

Question No: 01

You need to complete the game so that a player can play slide puzzle game?

Answer:

```
import keyboard

goal = [1,2,3,4,5,6,7,8,0]
start =[8,2,0,4,7,6,3,5,1]


#printing the puzzle
def printPuzzle(state):
    print(' ___________\n| {} | {} | {} |\n ___________\n| {} | {} | {} |\n ___________\n| {} | {} | {} |\n -----------'.format(
        state[0], state[1], state[2],
        state[3], state[4], state[5],
        state[6], state[7], state[8]))

# Finding the result based on actions performed
def result(Incomming, action):
    outgoing = Incomming
    index = Incomming.index(0)
    if(action == "up" and index>2):
        outgoing[index] = Incomming[index-3]
        outgoing[index-3] = 0
    elif(action == "down" and index<6):
        outgoing[index] = Incomming[index+3]
        outgoing[index+3] = 0
    elif(action == "left" and (index%3!=0)):
        outgoing[index] = Incomming[index-1]
        outgoing[index-1] = 0
    elif(action == "right" and (index%3!=2)):
        outgoing[index] = Incomming[index+1]
        outgoing[index+1] = 0

    return outgoing







# Running the Game
print("...............You Can Use Arrow Keys to Move ..........")
```



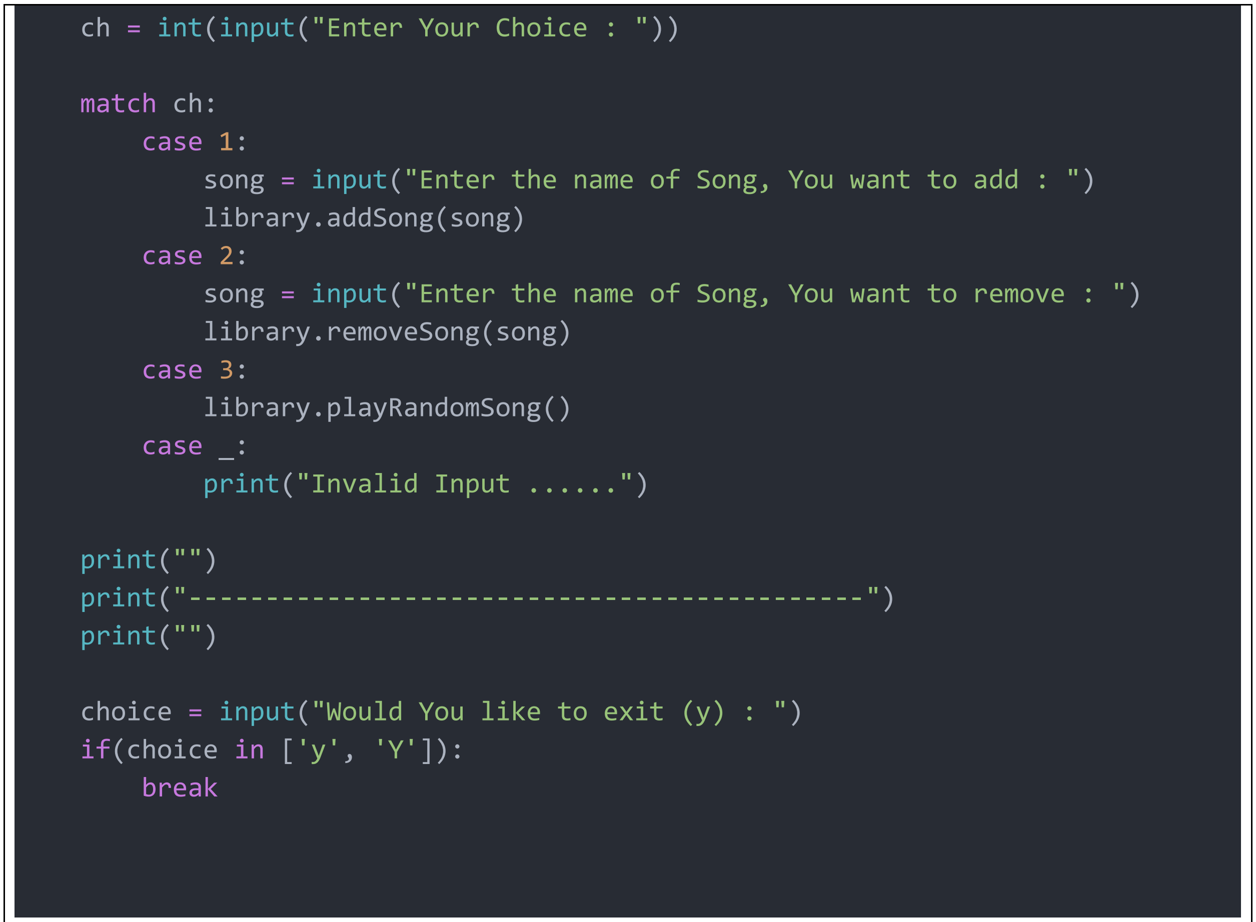
```
while(True):
    printPuzzle(start)
    print("")
    print("")
    print("")
    print("      -----      ")
    print("     |  W  |      ")
    print(" ----------------")
    print("| A  |  S  |  D  |")
    print(" ----------------")


    ch = input("Enter Your Choice : ")

    if(ch  in ['W','A','S', 'D','w', 'a', 's', 'd']):
        if(ch == 'W' or ch== 'w'):
            start = result(start, 'up')
        elif(ch=='A'or ch == 'a'):
            start = result(start, 'left')
        elif(ch == 'S' or ch == 's'):
            start = result(start, 'down')
        elif(ch == 'D' or ch == 'd'):
            start = result(start, 'right')

    if(goal == start ):
        print("Congrats....! You have Won the GAme")
        break
```

Question No: 02

Write a program to create a class called "MusicLibrary" with a collection of songs and methods to add and remove songs, and to play a random song.

Answer:

```
from email.policy import default
import random

class MusicLibrary:
    def __init__(self):
        self.song=[]

    def addSong(self, song):
        self.song.append(song)
        print(f"{song} added to Music Library")

    def removeSong(self, song):
        if song in self.song:
            self.song.remove(song)
            print(f"{song} Removed Successfully")
        else:
            print(f"{song} Not found .... in Library")


    def playRandomSong(self):
        if self.song:
            random_song = random.choice(self.song)
            print(f"Now playing: ' {random_song} ' ")
        else:
            print("No Song in the Music Library")



library = MusicLibrary()
library.addSong("Tera Fittor")
library.addSong("Without Me")


while True:
    print("Select the Operation You want to Perform : ")
    print("1. Add Song")
    print("2. Remove Song")
    print("3. Play Random Song")
```

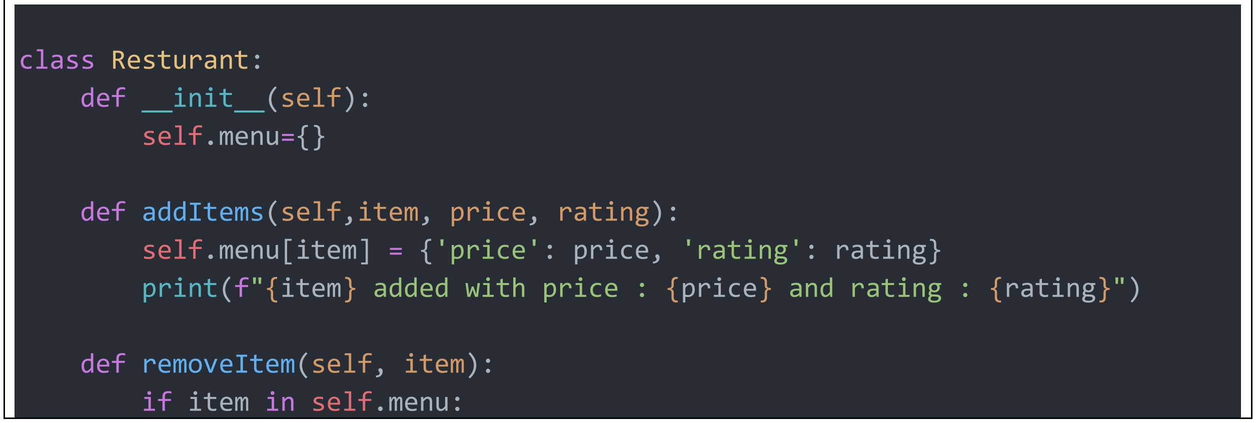


```
    ch = int(input("Enter Your Choice : "))

    match ch:
        case 1:
            song = input("Enter the name of Song, You want to add : ")
            library.addSong(song)
        case 2:
            song = input("Enter the name of Song, You want to remove : ")
            library.removeSong(song)
        case 3:
            library.playRandomSong()
        case _:
            print("Invalid Input ......")

    print("")
    print("--------------------------------------------")
    print("")

    choice = input("Would You like to exit (y) : ")
    if(choice in ['y', 'Y']):
        break
```

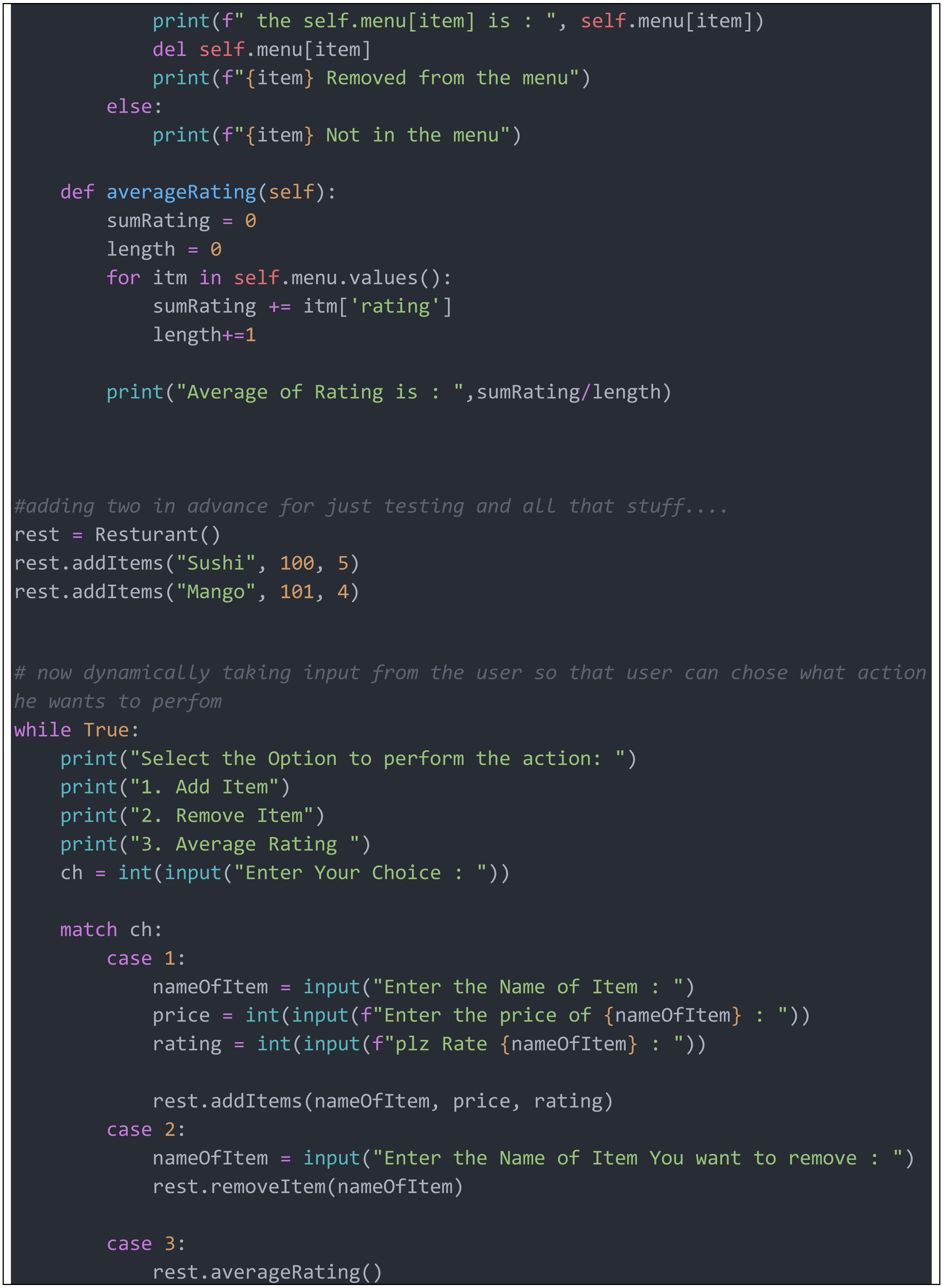
Question No: 03

Write a program to create a class called "Restaurant" with attributes for menu items, prices, and ratings, and methods to add and remove items, and to calculate average rating.

Answer:



```
class Resturant:
    def __init__(self):
        self.menu={}

    def addItems(self,item, price, rating):
        self.menu[item] = {'price': price, 'rating': rating}
        print(f"{item} added with price : {price} and rating : {rating}")

    def removeItem(self, item):
        if item in self.menu:
```



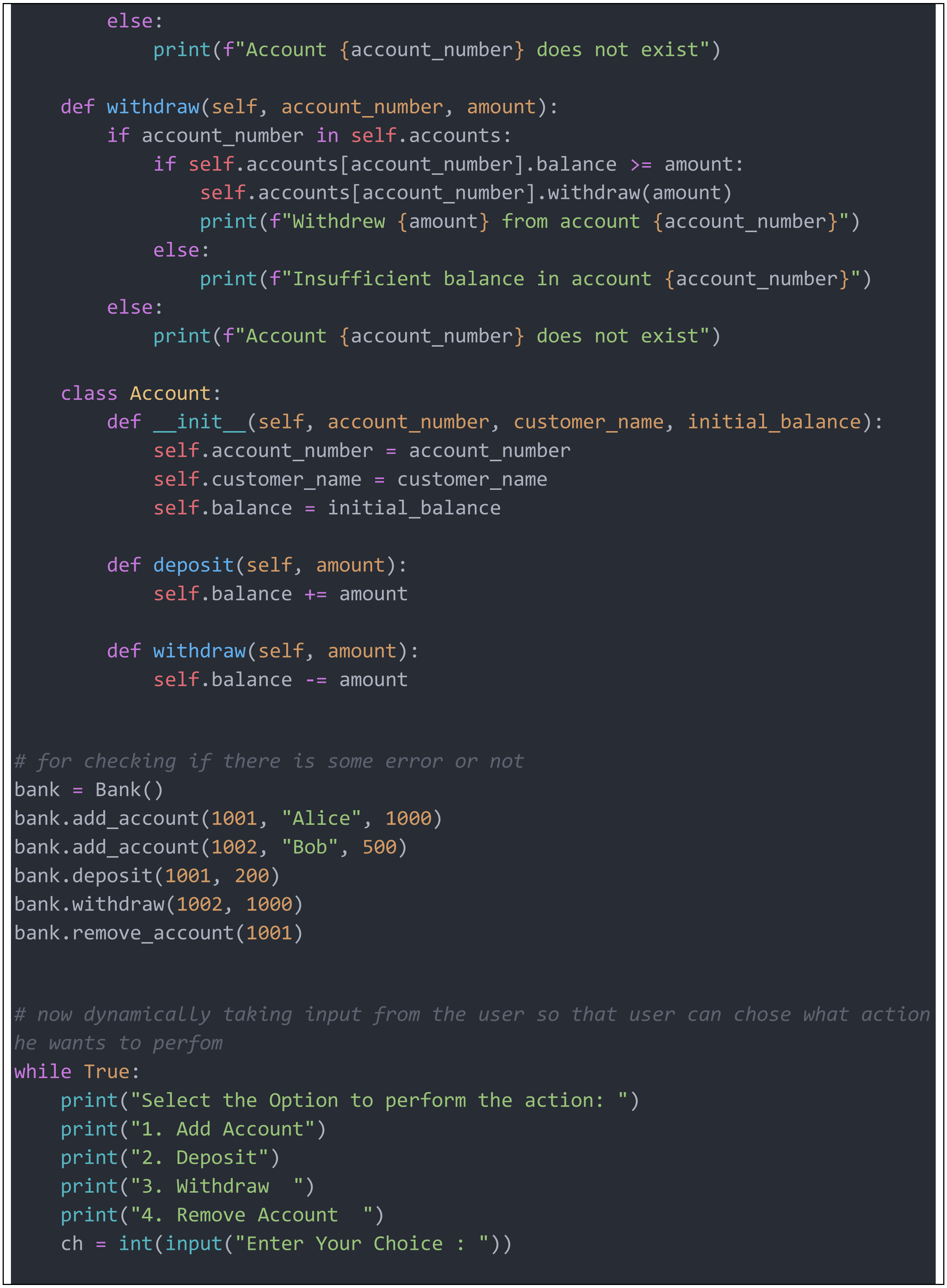
```
            print(f" the self.menu[item] is : ", self.menu[item])
            del self.menu[item]
            print(f"{item} Removed from the menu")
        else:
            print(f"{item} Not in the menu")

    def averageRating(self):
        sumRating = 0
        length = 0
        for itm in self.menu.values():
            sumRating += itm['rating']
            length+=1

        print("Average of Rating is : ",sumRating/length)



#adding two in advance for just testing and all that stuff....
rest = Resturant()
rest.addItems("Sushi", 100, 5)
rest.addItems("Mango", 101, 4)


# now dynamically taking input from the user so that user can chose what action
he wants to perfom
while True:
    print("Select the Option to perform the action: ")
    print("1. Add Item")
    print("2. Remove Item")
    print("3. Average Rating ")
    ch = int(input("Enter Your Choice : "))

    match ch:
        case 1:
            nameOfItem = input("Enter the Name of Item : ")
            price = int(input(f"Enter the price of {nameOfItem} : "))
            rating = int(input(f"plz Rate {nameOfItem} : "))

            rest.addItems(nameOfItem, price, rating)
        case 2:
            nameOfItem = input("Enter the Name of Item You want to remove : ")
            rest.removeItem(nameOfItem)

        case 3:
            rest.averageRating()
        case _:
            print("invalid Input")

    choice = input("Would You like to exit (y) : ")
    if(choice in ['y', 'Y']):
        break
```

<u>Question No: 04</u>

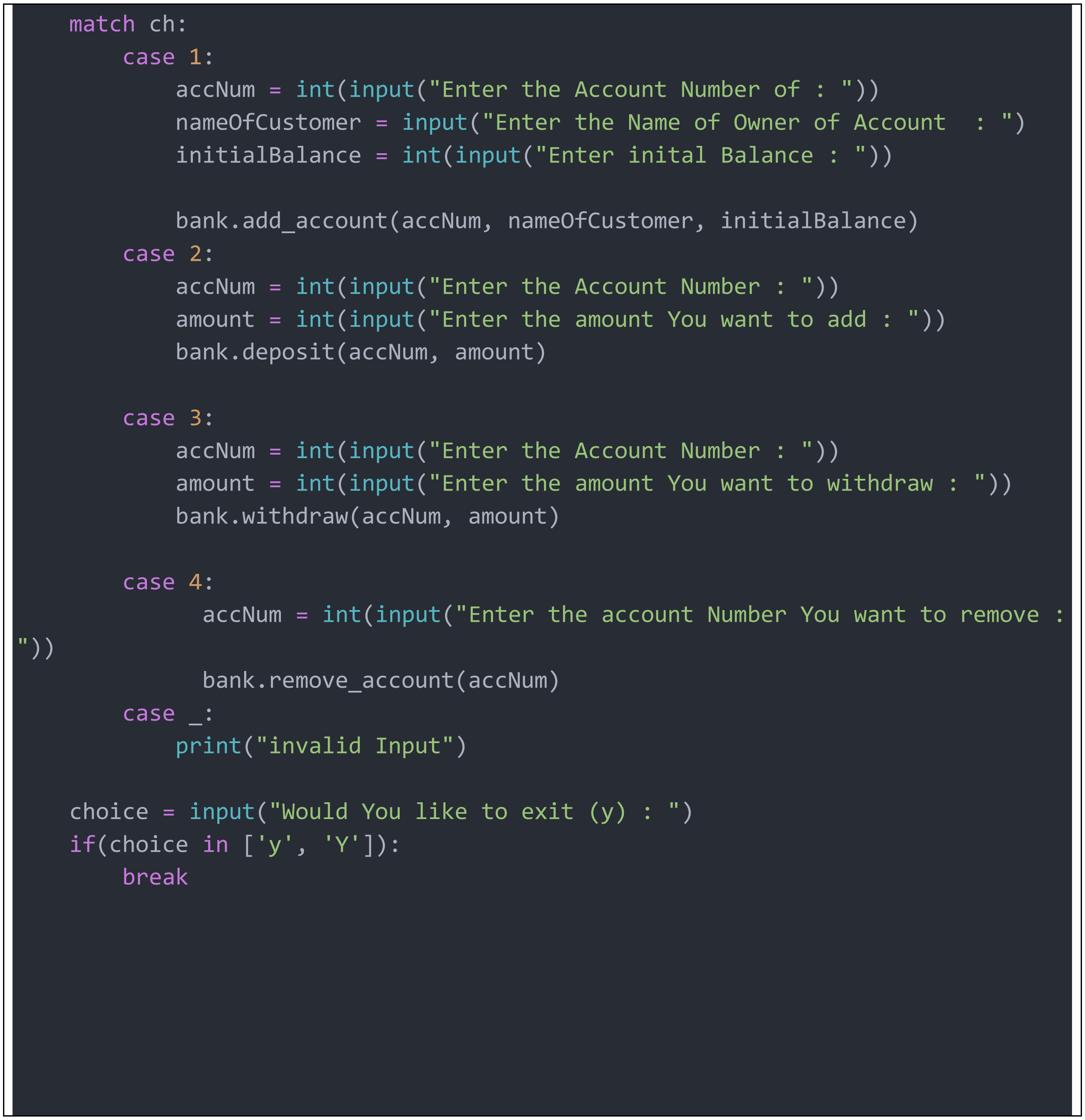
Write a program to create a class called "Bank" with a collection of accounts and methods to add and remove accounts, and to deposit and withdraw money. Also define a class called "Account" to maintain account details of a particular customer

<u>Answer:</u>

```
class Bank:
    def __init__(self):
        self.accounts = {}

    def add_account(self, account_number, customer_name, initial_balance=0):
        if account_number not in self.accounts:
            self.accounts[account_number] = self.Account(account_number, 
customer_name, initial_balance)
            print(f"Account {account_number} created for {customer_name} with 
initial balance {initial_balance}")
        else:
            print(f"Account {account_number} already exists")

    def remove_account(self, account_number):
        if account_number in self.accounts:
            del self.accounts[account_number]
            print(f"Account {account_number} removed")
        else:
            print(f"Account {account_number} does not exist")

    def deposit(self, account_number, amount):
        if account_number in self.accounts:
            self.accounts[account_number].deposit(amount)
            print(f"Deposited {amount} into account {account_number}")
```



```python
        else:
            print(f"Account {account_number} does not exist")

    def withdraw(self, account_number, amount):
        if account_number in self.accounts:
            if self.accounts[account_number].balance >= amount:
                self.accounts[account_number].withdraw(amount)
                print(f"Withdrew {amount} from account {account_number}")
            else:
                print(f"Insufficient balance in account {account_number}")
        else:
            print(f"Account {account_number} does not exist")

    class Account:
        def __init__(self, account_number, customer_name, initial_balance):
            self.account_number = account_number
            self.customer_name = customer_name
            self.balance = initial_balance

        def deposit(self, amount):
            self.balance += amount

        def withdraw(self, amount):
            self.balance -= amount


# for checking if there is some error or not
bank = Bank()
bank.add_account(1001, "Alice", 1000)
bank.add_account(1002, "Bob", 500)
bank.deposit(1001, 200)
bank.withdraw(1002, 1000)
bank.remove_account(1001)


# now dynamically taking input from the user so that user can chose what action he wants to perfom
while True:
    print("Select the Option to perform the action: ")
    print("1. Add Account")
    print("2. Deposit")
    print("3. Withdraw  ")
    print("4. Remove Account  ")
    ch = int(input("Enter Your Choice : "))
```



```
    match ch:
        case 1:
            accNum = int(input("Enter the Account Number of : "))
            nameOfCustomer = input("Enter the Name of Owner of Account  : ")
            initialBalance = int(input("Enter inital Balance : "))

            bank.add_account(accNum, nameOfCustomer, initialBalance)
        case 2:
            accNum = int(input("Enter the Account Number : "))
            amount = int(input("Enter the amount You want to add : "))
            bank.deposit(accNum, amount)

        case 3:
            accNum = int(input("Enter the Account Number : "))
            amount = int(input("Enter the amount You want to withdraw : "))
            bank.withdraw(accNum, amount)

        case 4:
              accNum = int(input("Enter the account Number You want to remove :
"))
              bank.remove_account(accNum)
        case _:
            print("invalid Input")

    choice = input("Would You like to exit (y) : ")
    if(choice in ['y', 'Y']):
        break
```

The End....!