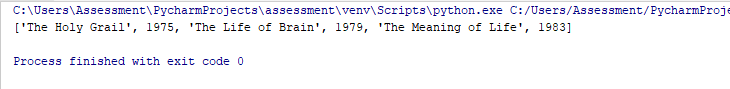
**Question 1:**

movies = [**"The Holy Grail"**, **"The Life of Brain"**, **"The Meaning of Life"**]  
years = [1975,1979,1983]  
*#Creating dictionary for merge two values*d = {}  
a = []  
b = []  
**for** i **in** range(3):  
 d[movies[i]] = years[i]  
*#'a' is convert the dict to list*a = list(d.items())  
**for** i **in** a:  
 *#to check the element in list is tuple or not* **if** type(i) == tuple:  
 **for** elem **in** i:  
 b.append(elem)  
*#print the output in list format*print(b)

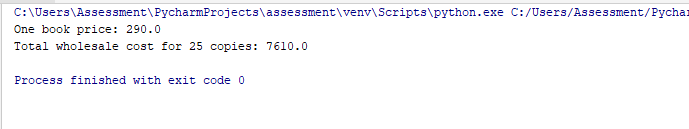
**OUTPUT:**



**Question 2:**

**def** total\_price():  
 cover\_price = 650  
 discount\_for\_book = 40/100 \* cover\_price  
 shipping\_cost = 30  
 Additional\_copies = 15  
 per\_book = discount\_for\_book + shipping\_cost  
 print(**"One book price: {}"**.format(per\_book))  
 price\_for\_25\_books = per\_book\*25 + (24\*Additional\_copies)  
 print(**"Total wholesale cost for 25 copies: {}"**.format(price\_for\_25\_books))  
total\_price()

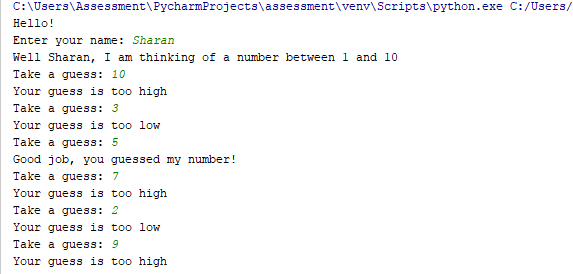
**OUTPUT:**



**Question 3:**

print(**"Hello!"**)  
a = input(**"Enter your name: "**)  
print(**"Well {}, I am thinking of a number between 1 and 10"**.format(a))  
*#Creating function***def** guess(num):  
*#The condition based on the input as "get six guesses"* **while** num != 7:  
 n = int(input(**"Take a guess: "**))  
 *#Check the condition to check which one is satisfied* **if** n < 5:  
 print(**"Your guess is too low"**)  
 **elif** n == 5:  
 print(**"Good job, you guessed my number!"**)  
 **else**:  
 print(**"Your guess is too high"**)  
 *#Increment the num, because loop going to execute upto six times which they mentioned in question* num = num + 1  
  
guess(1)

**OUTPUT:**

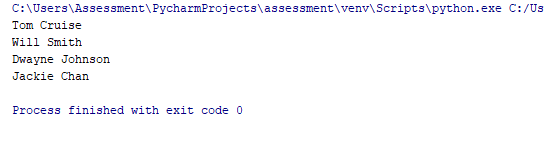


**Question 4:**

**Package Creation:**

*#Package created  
#Actors module created  
#input list*actors\_cast = [**'Tom Cruise'**, **'Will Smith'**,**'Dwayne Johnson'**, **'Jackie Chan'**, **'Arnold Schwarzenegger'**]  
*#by using function print the actors name***def** actor\_name():  
 **for** elem **in** range(0,len(actors\_cast)):  
 print(actors\_cast[elem])  
  
actor\_name()

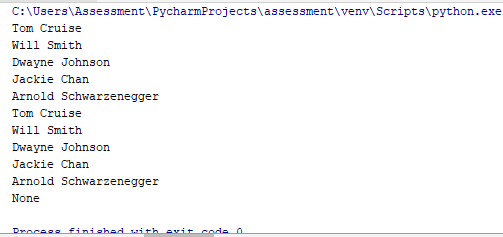
**OUTPUT:**



**Import data from “Actors” module:**

**from** package **import** actors  
*#Access data from actors in package  
#Print the actors name*print(actors.actor\_name())

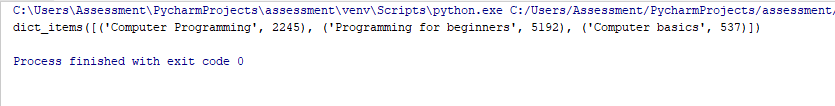
**OUTPUT:**



**Question 5:**

order\_no = [12354,12355,12356]  
book\_title = [**"Computer Programming"**,**"Programming for beginners"**,**"Computer basics"**]  
quantity = [5,8,3]  
price = [449,649,179]  
*#total price values for each book is append to 'a' list*a = []  
**for** i **in** range(len(quantity)):  
 a.append(quantity[i]\*price[i])  
*#covert list to dictionary*d = {}  
**for** j **in** range(len(book\_title)):  
 d[book\_title[j]] = a[j]  
*#print the required output*print(d.items())

**OUTPUT:**

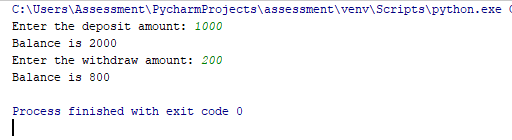


**Question 6:**

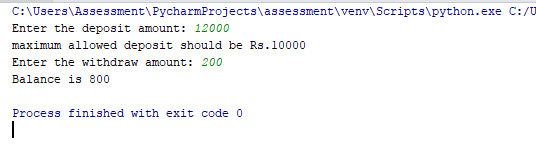
**class** Account:  
 *#set customer name and balance* **def** \_\_init\_\_(self,cust\_name):  
 self.cust\_name = cust\_name  
 self.balance = 1000  
  
 **def** deposit(self):  
 deposit\_amt = int(input(**"Enter the deposit amount: "**))  
 *#Deposit amount should be below 10000* **if** deposit\_amt <= 10000:  
 *#Deposit amount should be multiple's of 100's* **if** (deposit\_amt%100 == 0):  
 self.Deposit\_Balance = self.balance + deposit\_amt  
 print(**"Balance is {}"**.format(self.Deposit\_Balance))  
 **else**:  
 print(**"Deposit amount should be multiples of 100s"**)  
 **else**:  
 print(**"maximum allowed deposit should be Rs.10000"**)  
  
  
 **def** withdraw(self):  
 withdraw\_amt = int(input(**"Enter the withdraw amount: "**))  
 **if** withdraw\_amt%100 == 0:  
 *#Minimum balance should be minimum 500 else couldnot withdraw amount  
 #Withdraw amount should be multiples of 100's* **if** (self.balance >= 500):  
 self.Withdraw\_Balance = self.balance-withdraw\_amt  
 print(**"Balance is {}"**.format(self.Withdraw\_Balance))  
 **else**:  
 print(**"Customer should maintain minimum balance 500"**)  
 **else**:  
 print(**"Withdraw amount should be multiples of 100s"**)  
  
*#Creating objects and call that two functions are "deposit" and "withdraw"*obj = Account(**"Mouni"**)  
obj.deposit()  
obj.withdraw()

**OUTPUT:**

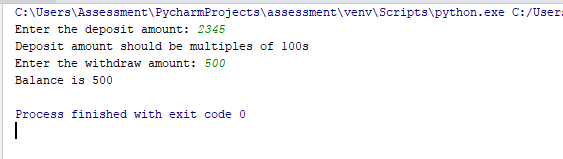
**Deposit and withdraw (both valid)**



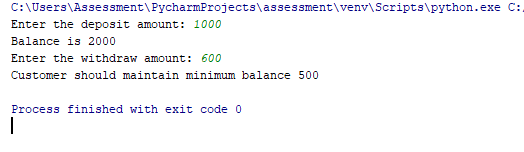
**Deposit (invalid) and withdraw (valid)**



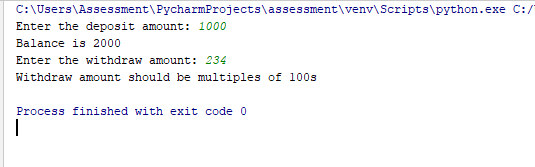
**Deposit (invalid) and withdraw (valid)**



**Deposit (valid) and withdraw (invalid)**



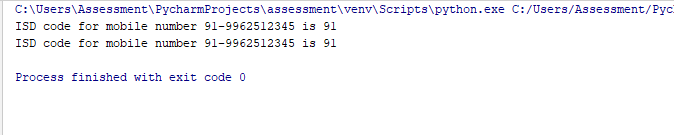
**Deposit (valid) and withdraw (invalid)**



**Question 7:**

**import** re  
  
*#Input mobile number is given*a = **'91-9962512345'**b = **'919962512345'***#ISD code pattern created*pattern = **'91'***#Match the pattern in the given input*matches1 = re.search(pattern,a)  
matches2 = re.search(pattern,b)  
*#Print output of ISD code for mobile numbers*print(**"ISD code for mobile number {} is {}"**.format(a,matches1.group(0)))  
print(**"ISD code for mobile number {} is {}"**.format(b,matches2.group(0)))

**OUTPUT:**



**Question 8:**

*#Opening file 1 for read the contents***with**(open(**"C:\\Users\\Assessment\\Desktop\\Saradha\\file1.txt"**,**'r'**)) **as** file1:  
 *#Opening file 2 for copying contents from file 1* **with**(open(**"C:\\Users\\Assessment\\Desktop\\Saradha\\file2.txt"**,**'w'**)) **as** file2:  
 *#Copy content from file 1 to file 2 by using looping method* **for** line **in** file1:  
 file2.write(line)  
*#Close the file after completing the copying content*file2.close()  
*#To read content from "file 2"***with**(open(**"C:\\Users\\Assessment\\Desktop\\Saradha\\file2.txt"**,**'r'**)) **as** file2:  
 print(**"File 2 contents copied from File 1\n"**)  
 line = file2.readline()  
 **while** line:  
 *#print the all contents from file 2* print(line)  
 line = file2.readline()

**OUTPUT:**

