**Credit Card Number Validation program**

**Objective** To write a program that validates a credit card number.

**Description**

Program takes input from user for the credit card number.

The isValid() function checks if user has entered valid input, counts the number of digits and checks if the count is 13, 14, 15 or 16.

The cardType() function checks the first 2 digits of the number and announces that the card is of one of the following types:

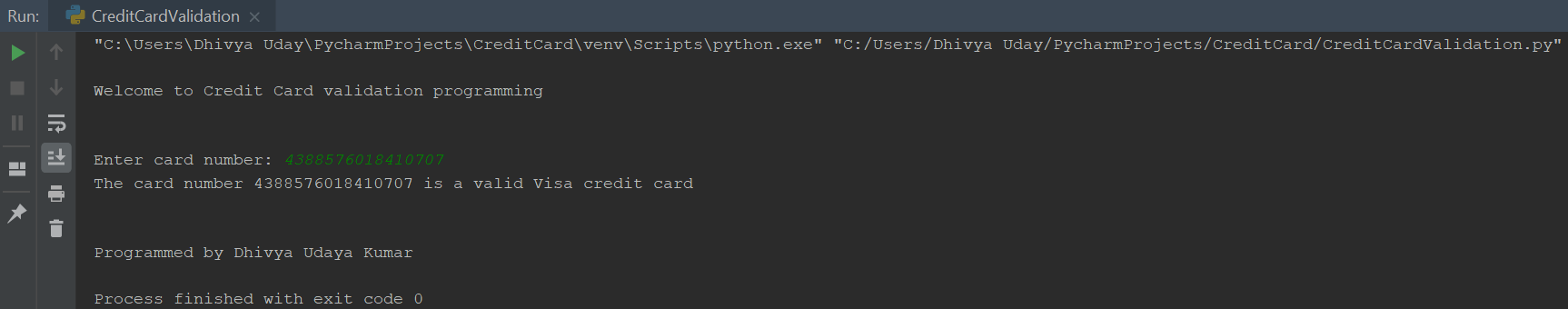
1. Visa
2. MasterCard
3. American Express
4. Discover

If not, the card is marked invalid and prompts user to enter a different number.

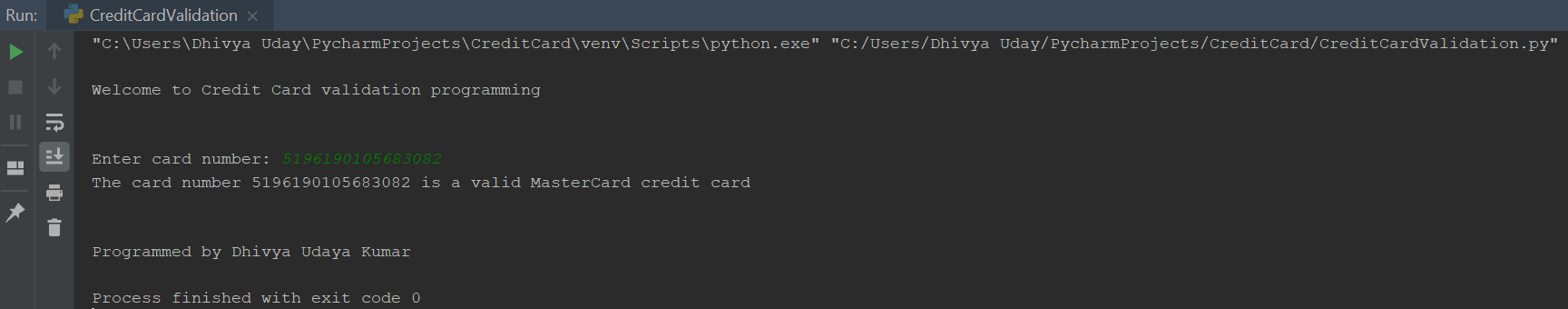
IF the number passes the above two functions, a Luhn check is performed through sumOfDoubleEvenPlace(), sumOfOddPlace(), getDigit() methods and then the results are added and checked to see if the sum is divisible by 10. If yes, the card is marked valid.

**Snapshots of Testcases:**

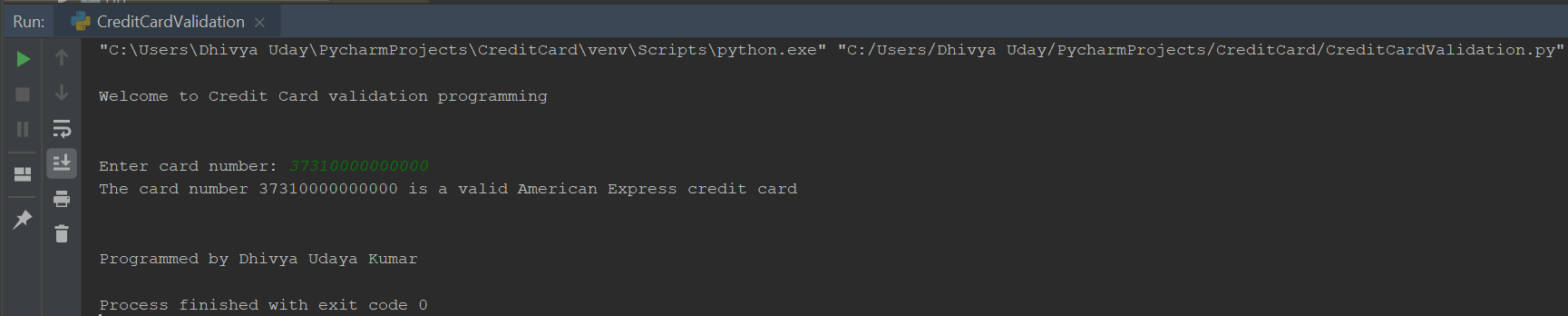
1. Scenario 1: Valid Visa card



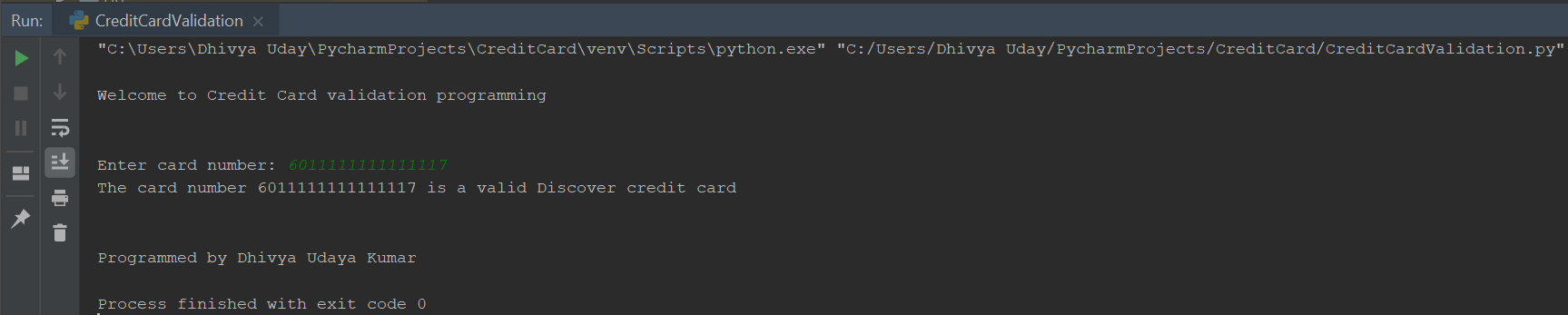
1. Scenario 2: Valid MasterCard



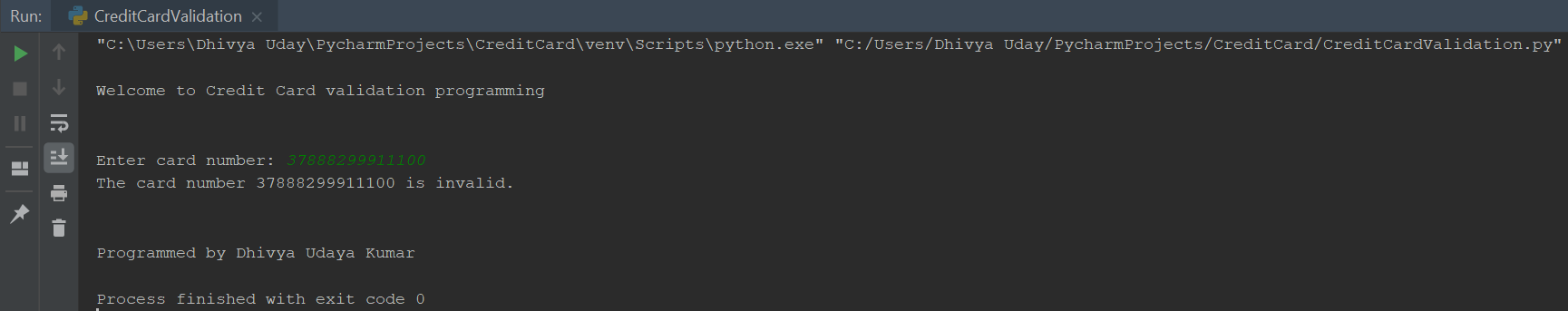
1. Scenario 3: Valid American Express card



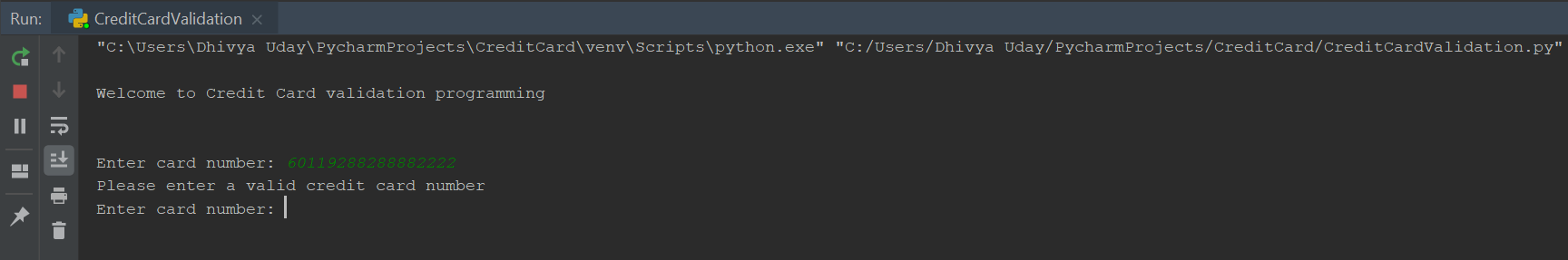
1. Scenario 4: Valid Discover card

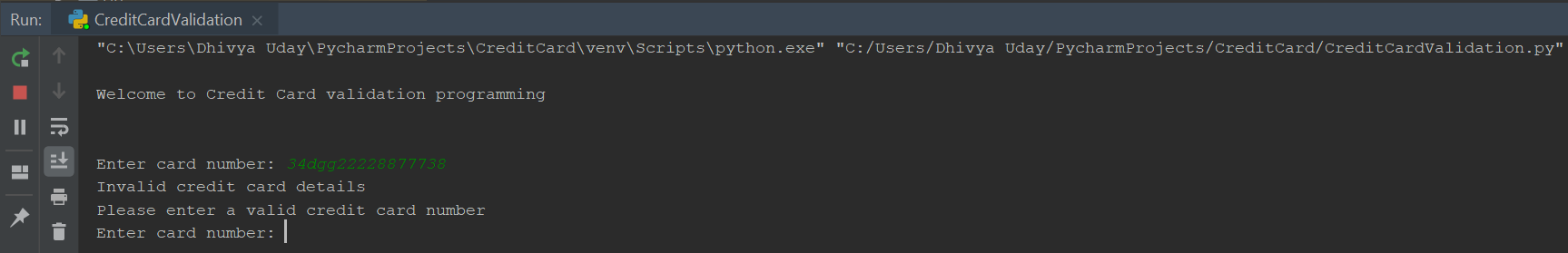


1. Scenario 5: Invalid card



1. Scenario 6: Invalid inputs





**Source code**

#Program to perform Credit Card validation

#Programmed by Dhivya Udaya Kumar

print("\nWelcome to Credit Card validation programming")

print("\n")

#Function to identify the card type

def cardType(number):

if(number[0] == '4'):

return "Visa"

elif(number[0] == '5'):

return "MasterCard"

elif(number[:2] == '37'):

return "American Express"

elif(number[0] == '6'):

return "Discover"

else:

return "Invalid"

#Function to validate number of digits in the card

def isValid(number):

count = 0

curr = number

while curr != '':

digit = curr[len(curr) - 1] #to get final character

curr = curr[:len(curr) - 1] #to trim last character off the string

if not digit.isdigit(): #check if non-digits are included

print("Invalid credit card details")

return False

count += 1

if count == 13 or count == 14 or count == 15 or count == 16 or cardType(number) != 'Invalid':

return True

return False

#Function to find if number is single digit, else return the sum of 2 digits

def getDigit(number):

curr = str(number)

if len(curr) == 1:

return curr

else:

return int(curr[len(curr) - 1]) + int(curr[len(curr) - 2])

#Function to get sum of double of digit in even place from right to left

def sumOfDoubleEvenPlace(number):

sum = 0

curr = number

while curr != '':

digit = curr[len(curr) - 2]

curr = curr[:len(curr) - 2]

doubleOfDigit = int(digit) \* 2

sum += int(getDigit(doubleOfDigit))

return sum

#Function to get sum of digits in odd place from right to left

def sumOfOddPlace(number):

sum = 0

curr = number

sum += int(curr[len(curr) - 1])

curr = curr[:len(curr) - 1]

while len(curr) > 1:

digit = curr[len(curr) - 2]

curr = curr[:len(curr) - 2]

sum += int(digit)

return sum

cardNumber = 0

#Credit Card number input

while True:

cardNumber = input("Enter card number: ")

isValidCard = isValid(cardNumber)

if isValidCard == False:

print("Please enter a valid credit card number")

else:

break

#Check for card number validity and print result

if((sumOfDoubleEvenPlace(cardNumber) + sumOfOddPlace(cardNumber)) % 10 == 0):

print("The card number "+cardNumber+" is a valid "+cardType(cardNumber)+" credit card")

else:

print("The card number "+cardNumber+" is invalid.")

print("\n")

print("Programmed by Dhivya Udaya Kumar")