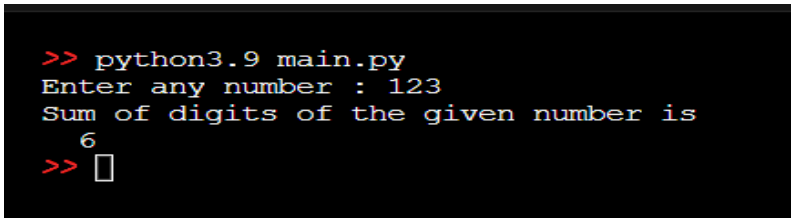
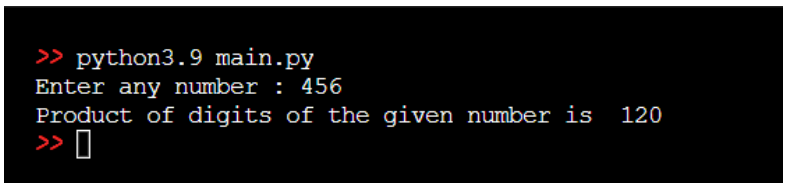
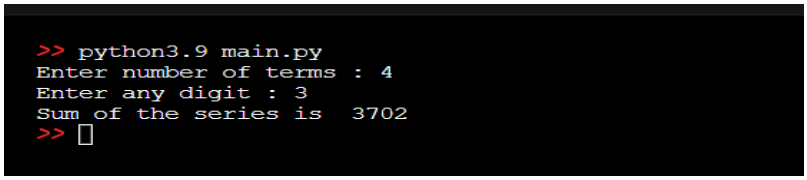
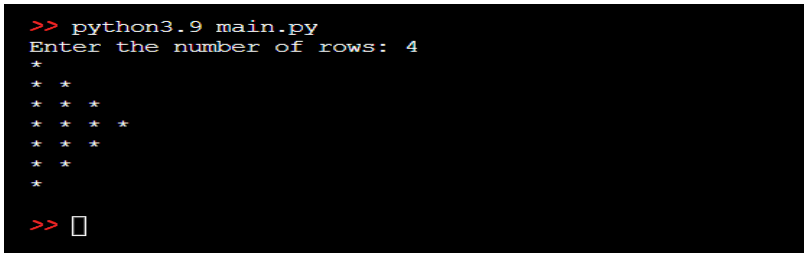
Name : Rajkumar Jaiswal
Enrolment no.: 0827CS191186
Class : CS-3

EXPERIMENT-3

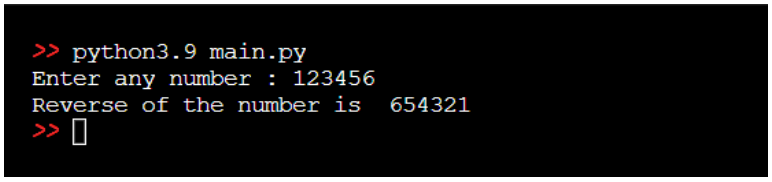
Program 1	Program to find digital sum of a given Number ex: n=123 Digital sum----->1+2+3=6
Solution	n=int(input("Enter any number : ")) sum=0 while(n!=0): digit=(n % 10) sum = sum + digit n = n//10 print("Sum of digits of the given number is ",sum)
Output (Screen Shot)	
Program 2	Program to find the digital product of a given number ex: n=43 Digital product ----->4*3=12
Solution	n=int(input("Enter any number : ")) product=1 while(n!=0): digit=(n % 10) product = product * digit n = n//10 print("Product of digits of the given number is ",product)
Output (Screen Shot)	
Program 3	Find the sum of the series 3 +33 + 333 + 3333 + .. n terms
Solution	n=int(input("Enter number of terms : ")) m=int(input("Enter any digit : ")) change=m sum=0 for i in range(n): sum+=change change=(change*10)+m print("Sum of the series is ",sum)\

Output (Screen Shot)	
Program 4	Print the following pattern *

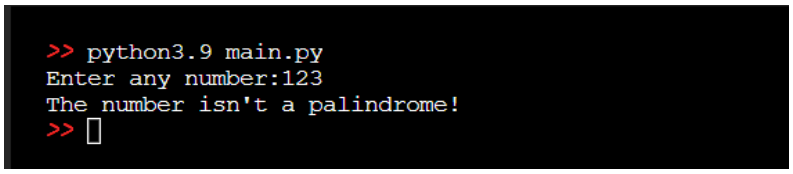
Solution	rows = int(input("Enter the number of rows: "))
	for i in range(0, rows):
	  for j in range(0, i + 1):
	  print("*", end=' ')
	  print(" ")
	for i in range(rows , 0, -1):
	  for j in range(0, i - 1):
	  print("*", end=' ')
	  print(" ")

Output (Screen Shot)	
Program 5	Program to reverse a given Number. ex: n=123 Reversed no is 321

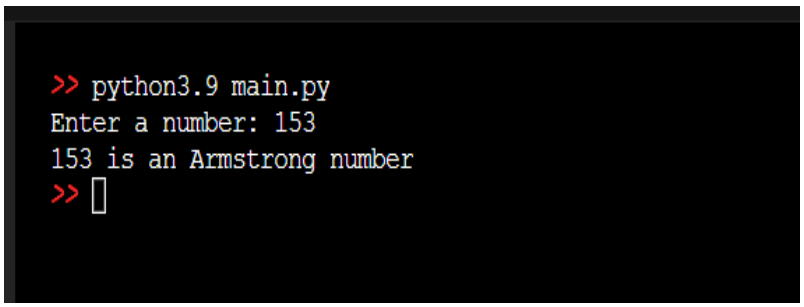
Solution	n = int(input("Enter any number : "))
	rev = 0
	while(n > 0):
	  a = n % 10
	  rev = rev * 10 + a
	  n = n // 10
	print("Reverse of the number is ",rev)

Output (Screen Shot)	
Program 6	Program to check whether a given number is a palindrome or not

Solution	n=int(input("Enter any number:"))
	temp=n
	rev=0
	while(n>0):
	  dig=n%10
	  rev=rev*10+dig
	  n=n//10
	if(temp==rev):
	  print("The number is a palindrome!")
	else:
	  print("The number isn't a palindrome!")

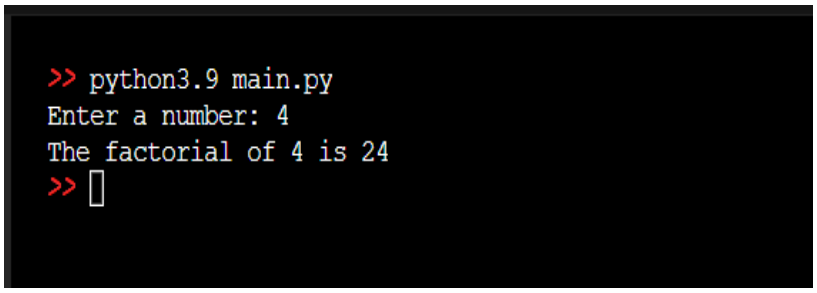
Output (Screen Shot)	
Program 7	Program to check whether a given number is an Armstrong number or not.

Solution	num = int(input("Enter a number: "))
	sum = 0
	temp = num
	while temp > 0:
	  digit = temp % 10
	  sum += digit ** 3
	  temp //= 10
	if num == sum:
	  print(num,"is an Armstrong number")
	else:
	  print(num,"is not an Armstrong number")

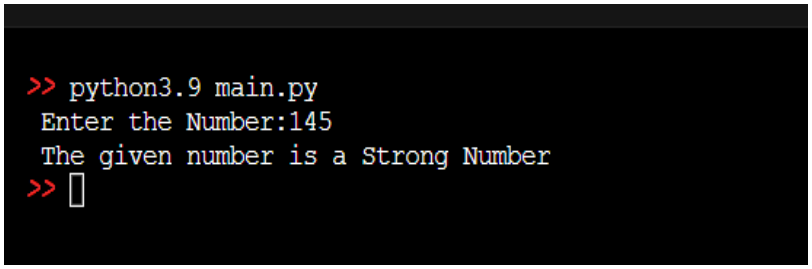
Output (Screen Shot)	

Program 8	Program to find factorial of a given number.

Solution	num = int(input("Enter a number: "))
	factorial = 1
	if num < 0:
	  print("Sorry, factorial does not exist for negative numbers")
	elif num == 0:
	  print("The factorial of 0 is 1")
	else:
	  for i in range(1,num + 1):
	  factorial = factorial*i
	  print("The factorial of",num,"is",factorial)

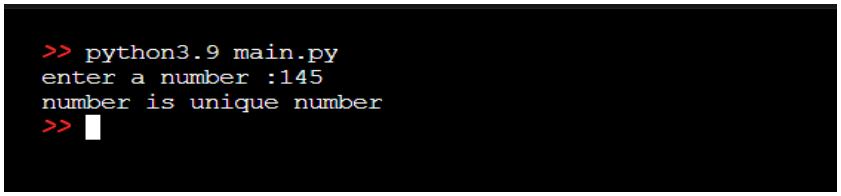
Output (Screen Shot)	
Program 9	Program to find whether a given number is a strong number or not. e.g. n=145 1!+4!+5!==145

Solution	num = int(input(" Enter the Number:"))
	sum = 0
	temp = num
	while(temp > 0):
	  fact = 1
	  rem = temp % 10
	  for i in range(1, rem + 1):
	  fact = fact * i
	  sum = sum + fact
	  temp = temp // 10
	if (sum == num):
	  print(" The given number is a Strong Number")
	else:
	  print(" The given number is not a Strong Number")

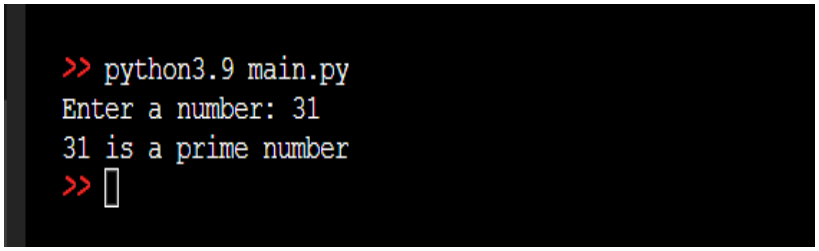
Output (Screen Shot)	

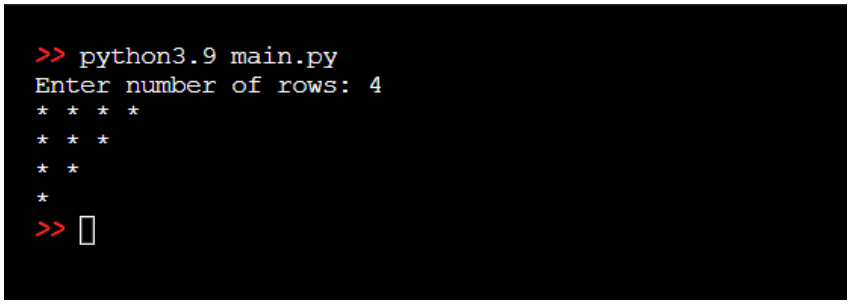
Program 10	Program to find whether a given number is a unique number. For example, 20, 56, 9863, 145, etc. are the unique numbers while 33, 121, 900, 1010, etc. are not unique numbers

Solution	def checkunique(numb):
	  visited=[0,0,0,0,0,0,0,0,0,0]
	  while(numb):
	  if visited[numb%10]==1:
	  break
	  else :
	  visited[numb%10]=1
	  numb=numb//10
	  return numb
	num=int(input("enter a number :"))
	numb=checkunique(num)
	if (numb==0):
	  print("number is unique number")
	else :
	  print("number is not unique")

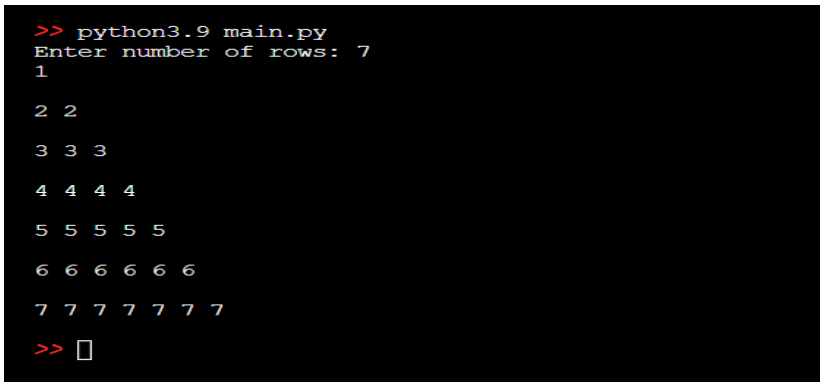
Output (Screen Shot)	
Program 11	Program to find whether a given number is a prime number or not.

Solution	num = int(input("Enter a number: "))
	flag = False
	if num > 1:
	  for i in range(2, num):
	  if (num % i) == 0:
	  flag = True
	  break
	if flag:
	  print(num, "is not a prime number")
	else:
	  print(num, "is a prime number")

Output (Screen Shot)	

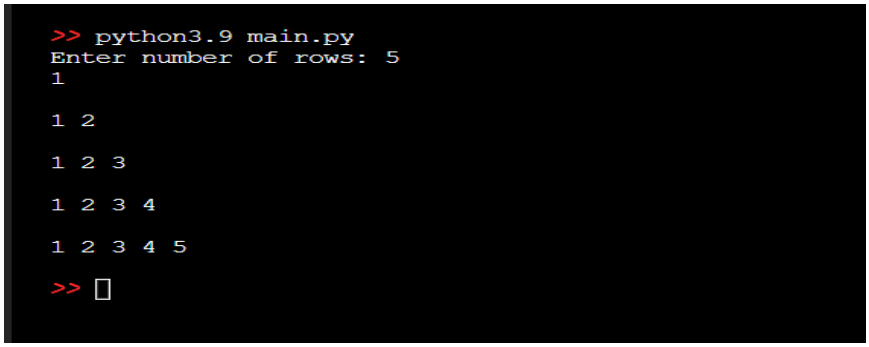
Program 12	Print downward Half-Pyramid Pattern with Star (asterisk) * * * * * * * * * * * * * * *
Solution	rows = int(input("Enter number of rows: ")) for i in range(rows, 0, -1): print("* "*i)
Output (Screen Shot)	
Program 13	Print following pattern: 1 22 333 4444 55555 666666 7777777 88888888 999999999

Solution	rows = int(input("Enter number of rows: "))
	for i in range(rows):
	  for j in range(i+1):
	  print(i+1, end=" ")
	  print("\n")

Output (Screen Shot)	

Program 14	Print following pattern: 1 12 123 1234 12345 123456 1234567 12345678 123456789

Solution	rows = int(input("Enter number of rows: "))
	for i in range(rows):
	  for j in range(i+1):
	  print(j+1, end=" ")
	  print("\n")

Output (Screen Shot)	
Program 15	Print following pattern: A BB CCC DDDD EEEEE FFFFFF GGGGGGG FFFFFFFF

Solution	rows = int(input("Enter number of rows: "))
	ascii_value = 65
	for i in range(rows):
	  for j in range(i+1):
	  alphabet = chr(ascii_value)
	  print(alphabet, end=" ")
	  ascii_value += 1
	  print("\n")

Output (Screen Shot)