Basic Python Programming –UNIT 1	

1	Write a Python Program to Check Prime Number.
	`# To take input from the user`
	`#num = int(input("Enter a number: "))`
	
	`# define a flag variable`
	`flag = False`
	
	`# prime numbers are greater than 1`
	`if num > 1:`
	`  # check for factors`
	`  for i in range(2, num):`
	`  if (num % i) == 0:`
	`  # if factor is found, set flag to True`
	`  flag = True`
	`  # break out of loop`
	`  break`
	
	`# check if flag is True`
	`if flag:`
	`  print(num, "is not a prime number")`
	`else:`
	`  print(num, "is a prime number")`

2	Write a Python Program to Find the Largest Among Three Numbers.
	`num1 = float(input("Enter first number: "))`
	`num2 = float(input("Enter second number: "))`
	`num3 = float(input("Enter third number: "))`
	`if (num1 >= num2) and (num1 >= num3):`
	`  largest = num1`
	`elif (num2 >= num1) and (num2 >= num3):`
	`  largest = num2`
	`else:`
	`  largest = num3`
	
	`print("The largest number is", largest)`

3	Write a Python Program to Print the Fibonacci sequence.
	`# Program to display the Fibonacci sequence up to n-th term`
	`nterms = int(input("How many terms? "))`
	`# first two terms`
	`n1, n2 = 0, 1`
	`count = 0`
	`# check if the number of terms is valid`
	`if nterms <= 0:`
	`  print("Please enter a positive integer")`
	`# if there is only one term, return n1`
	`elif nterms == 1:`
	`  print("Fibonacci sequence upto",nterms,":")`
	`  print(n1)`
	`# generate fibonacci sequence`
	`else:`

	print("Fibonacci sequence:")
	while count < nterms:
	print(n1)
	nth = n1 + n2
	# update values
	n1 = n2
	n2 = nth
	count += 1

4	Write a python program to find the roots of a quadratic equation.
	# Python Program to find roots of a Quadratic Equation
	import math
	
	a = int(input("Please Enter a Value of a Quadratic Equation : "))
	b = int(input("Please Enter b Value of a Quadratic Equation : "))
	c = int(input("Please Enter c Value of a Quadratic Equation : "))
	
	discriminant = (b * b) - (4 * a * c)
	
	if(discriminant > 0):
	root1 = (-b + math.sqrt(discriminant) / (2 * a))
	root2 = (-b - math.sqrt(discriminant) / (2 * a))
	print("Two Distinct Real Roots Exists: root1 = %.2f and root2 = %.2f" %(root1, root2))
	elif(discriminant == 0):
	root1 = root2 = -b / (2 * a)
	print("Two Equal and Real Roots Exists: root1 = %.2f and root2 = %.2f" %(root1, root2))
	elif(discriminant < 0):
	root1 = root2 = -b / (2 * a)
	imaginary = math.sqrt(-discriminant) / (2 * a)
	print("Two Distinct Complex Roots Exists: root1 = %.2f+%.2f and root2 = %.2f-%.2f" %(root1, imaginary, root2, imaginary))

5	Write a python program to check whether a given number is palindrome or not.
	number = int(input("Please Enter any Number: "))
	reverse = 0
	temp = number
	while(temp > 0):
	Reminder = temp % 10
	reverse = (reverse * 10) + Reminder
	temp = temp //10
	
	print("Reverse of a Given number is = %d" %reverse)
	
	if(number == reverse):
	print("%d is a Palindrome Number" %number)
	else:
	print("%d is not a Palindrome Number" %number)

	Python Program Flow Control – UNIT 2
6	Write a Python Program to Print Pascal Triangles. # Print Pascal's Triangle in Python from math import factorial # input n

	n = int(input("Enter how many rows:")
	for i in range(n):
	  for j in range(n-i+1):
	  # for left spacing
	  print(end=" ")
	  for j in range(i+1):
	  # nCr = n!/((n-r)!*r!)
	  print(factorial(i)//(factorial(j)*factorial(i-j)), end=" ")
	  # for new line
	  print()

7	Write a Python program to Calculate Electricity Bill.
	1 - 100 unit – Rs.2.60/-
	101-200 unit – Rs. 3.25/-
	201-300 unit – Rs. 5.26/-
	Above 300 unit – Rs. 8.45/-
	# Python Program to Calculate Electricity Bill
	units = int(input(" Please enter Number of Units you Consumed : "))
	if(units < 50):
	  amount = units * 2.60
	  surcharge = 25
	elif(units <= 100):
	  amount = 130 + ((units - 50) * 3.25)
	  surcharge = 35
	elif(units <= 200):
	  amount = 130 + 162.50 + ((units - 100) * 5.26)
	  surcharge = 45
	else:
	  amount = 130 + 162.50 + 526 + ((units - 200) * 8.45)
	  surcharge = 75
	total = amount + surcharge
	print("\nElectricity Bill = %.2f" %total)

8	Write a python program to display a multiplication table. number = int(input ("Enter the number of which the user wants to print the multiplication table: ")) # We are using "for loop" to iterate the multiplication 10 times print ("The Multiplication Table of: ", number) for count in range(1, 11): print (number, 'x', count, '=', number * count)
9	Write a python program to find Student Grade. This python program allows users to enter five different values for five subjects English, Mathematics, Computer Science, Physics and Chemistry. Next, it finds the Total, and Percentage of those Five Subjects. Grades are calculated as follows based on percentage: Percentage Grade 91% - 100% S 81% - 90% A 71% - 80% B 61% - 70% C 51% - 60% D 41% - 50% E <40% FAIL # Python Program to find Student Grade

```
english = float(input(" Please enter English Marks: "))
math = float(input(" Please enter Math score: "))
computers = float(input(" Please enter Computer Marks: "))
physics = float(input(" Please enter Physics Marks: "))
chemistry = float(input(" Please enter Chemistry Marks: "))

total = english + math + computers + physics + chemistry
percentage = (total / 500) * 100

print("Total Marks = %.2f"  %total)
print("Marks Percentage = %.2f"  %percentage)

if(percentage >= 90):
   print("S Grade")
elif(percentage >= 80):
   print("A Grade")
elif(percentage >= 70):
   print("B Grade")
elif(percentage >= 60):
   print("CGrade")
elif(percentage >= 50):
   print("D Grade")
elif(percentage >=40):
   printf(“E Grade”)
else:
   print("Fail”)
```

10 Write a python program to find perfect number.
Note: For example, 6 is a perfect number in Python because 6 is divisible by 1, 2, 3 and 6. So, the sum of these values are: 1+2+3 = 6 (Remember, we have to exclude the number itself. That’s why we haven’t added 6 here). Some of the perfect numbers are 6, 28, 496, 8128 and 33550336 so on.

```
#  find Perfect Number using For loop

Number = int(input(" Please Enter any Number: "))
Sum = 0
for i in range(1, Number):
   if(Number % i == 0):
      Sum = Sum + i
if (Sum == Number):
   print(" %d is a Perfect Number" %Number)
else:
   print(" %d is not a Perfect Number" %Number)
```

Function: -- UNIT3

11)Python code to check a number is Prime no or not

```
def prime(no):
    no=int(no)
    n=int(no/2)
    flag=0
    for i in range(2,n):
        if no%i==0:
            flag+=1
        else:
            continue
    if flag>0:
        print("This is not prime")
    else:
        print("Prime")

no=input("Enter a no")
prime(no)
```

12)Sum of n numbers

```
#Function Definition
def sum(n):
    sum=0
    for i in range(1,n):
        sum+=i
    print(sum)

n=int(input("Enter n"))
sum(n) #Function Calling with one Argument
```

13)Python code to find biggest of 3 numbers

```
def great(a,b,c):
      if a>b and a>c:
         print("A is greatest")
       elif b>c:
            print("B is greatest")
       else:
            print("C is greatest")

a=input("Enter first no")
b=input("Enter 2nd no")
c=input("Enter 3rd no")
great(a,b,c)
```

14)Left order triangle

```
x=int(input("Enter row number="))
for i in range(x):
    for j in range(1,i+1):
        print(j,end=' ')
    print("")
```

15)Program for checking whether a number is Armstrong no or not

```
# Python program to check if the number is an Armstrong number or not
def arm(num):
    sum=0
    # find the sum of the cube of each digit
    temp = num
    while temp > 0:
        digit = temp % 10
        sum += digit ** 3
        temp //= 10
    return sum


# take input from the user
num = int(input("Enter a number: "))


# Function calling
sum=arm(num)
# display the result based on returned value from the function
if num == sum:
   print(num,"is an Armstrong number")
else:
   print(num,"is not an Armstrong number")
```

PYTHON DATA STRUCTURES –UNIT5	

16	Write a Python Program to perform various slicing operation over tuple.
	
	Tuple1 = tuple('WELCOME TO PYTHON')
	
	# Removing First element
	print("Removal of First Element: ")
	print(Tuple1[1:])
	
	# Reversing the Tuple
	print("\nTuple after sequence of Element is reversed: ")
	print(Tuple1[::-1])
	
	# Printing elements of a Range
	print("\nPrinting elements between Range 4-9: ")
	print(Tuple1[4:9])

17	Write a python program to show various slicing operations in LIST.
	
	# Creating a List
	List = ['a','b','c','d','e','f','g','h','i','j','k','o']
	print("Initial List: ")
	print(List)
	
	# Print elements of a range
	# using Slice operation
	Sliced_List = List[3:8]
	print("\nSlicing elements in a range 3-8: ")
	print(Sliced_List)
	
	# Print elements from a
	# pre-defined point to end
	Sliced_List = List[5:]
	print("\nElements sliced from 5th "
	"element till the end: ")
	print(Sliced_List)
	
	# Printing elements from
	# beginning till end
	Sliced_List = List[:]
	print("\nPrinting all elements using slice operation: ")
	print(Sliced_List)

18	Write a python program to create SET , add elements and perform insert and delete operations set1 = set() print("Initial blank Set: ")

	print(set1) set1.add(8) set1.add(11) set1.add(58) set1.add(9) set1.add((6,7)) print("\nSet after Addition of Three elements: ") print(set1) set1.remove(58) set1.remove(8) print("\nSet after Removal of two elements: ") print(set1)
19	Write a python program to create 3 sets with elements and perform different methods to delete an elements from Set. set1 = set([1, 2, 3, 4, 5, 6,7, 8, 9, 10, 11, 12]) set2 = set([11, 22, 23, 54, 85, 96]) set3 = set([100,22,336,545] print("Initial Set: ") print(set1) set1.discard(8) set1.discard(9) print("\nSet after Discarding two elements: ") print(set1) set1.pop() print("\nSet after popping an element: ") print(set2) set3.remove(100) set3.remove(22) print("\nSet after Removal of two elements: ") print(set3)

19	Write a python program to create sets with elements and perform intersection() and Symmetric difference() on set.
	
	# Python3 program for intersection() function
	set1 = {2, 4, 5, 6}
	set2 = {4, 6, 7, 8}

	set3 = {4, 6, 8} # union of two sets print("set1 intersection set2 : ", set1.intersection(set2)) # union of three sets print("set1 intersection set2 intersection set3 :", set1.intersection(set2, set3)) print(set1.symmetric_difference(set2))