Unit 4

Function

Q.1 Explain Python Mathematical function (4) or
→ Explain Following mathematical function with example. (7)

→ 1) abs()
2) divmod()
3) min()
4) max()
5) pow()
6) sum()
7) round()

Explanation :-

1) abs() : આ function દ્વારા પણ આપેલ નો Absolute value return કરે છે. જો આપેલ ~~નું~~ Negative હોય, તો તે આપો-આપ positive થઈ જાય છે.

Syntax :- abs(number)

Example :-

```
Print (abs (5))
Print (abs (-5))
Print (abs (-2.24))
Print (abs (2.24))
```

Output :-

```
5
5
2.24
2.24
```

2) divmod():- આ Fu. બે નંબર લે છે અને તેનો ભાગાકાર કરી તે એક Tuple return કરે છે. આ tuple માં પહેલો value Quotient અને બીજો remainder હોય છે.

Syntax :- divmod (num1, num2)

Example :-

```
Print (divmod (5, 2))
Print (divmod (2.5, 2))
Print (divmod (4, 2))
Print (divmod (18, 5))
```

Output :

```
(2, 1)
(1.0, 0.5)
(2, 0)
(3, 3)
```

3) max() : આ function આપેલી બધી જ values કે sequence (List કે Tuple) માંથી મોટી મોટી value return કરે છે.

Syntax :- max (item1, item2, ...)

Example :-

```
Print([12, 5, 2, 45, 3])
Print([45, 23, 67, 3])
```

Output :-

4) min() :- આ max() ની બિલકુલ ઉલટું છે. તે આપેલી બધા values માંથી નીચેની વાળી value નીકળે છે તે આપે છે.

Syntax :- min (item1, item2, ...)

Example :-

```
Print (min([12, 5, 6, 45]))
Print (min([2.4, 4.5, -1]))
```

Output :-

```
5
-1
```

5) pow() :- આ function power calculate કરે છે. તે x ની ઉપર y ની power ની વેલ્યુ return કરે છે.

Syntax : pow (base, exponent)

Example :

```
Print ( pow(2, 3))
Print ( pow (4, 2))
```

Output : 8
16

6) Sum() :- આ function દ્વારા પણ iterable માં આપેલા બધા જ numbers નો સરવાળો કરીને return કરે છે.

Syntax : sum (iterable)

Example :

```
Print(sum([2,3,4,5]))
Print(sum([1.2, 2.2, 3.2]))
```

Output :-

```
14
6.6
```

7) round() :-

Sytax :

Example :

```
Print(round(5.6))
Print(round(5.4367,2))
```

Output :

```
5
5.44
```

8) len() :-

Syntax :

Example :-

```
Print (len('Hello'))
Print (len([1,2,3,4]))
```

Output :-

5
4

Q.2 Explain random module & its function (4/7)

Introduction to random module :-

-> Python ના random module દ્વારા functions આપેલ છે જેની મદદથી random numbers generate કરવા છે, sequence માંથી કોઈ random item select કરવા માટે વપરાય છે.

-> Function :-

1) random.random() :- આ function 0.0 થી 1.0 ની વચ્ચે આવતાં કોઈ પણ random float value generate કરે છે. આમાં 0.0 include થાય છે, પણ 1.0 include નથી થતું.

Syntax : random.random()

Example :

```
Print (random.random ())
```

Output :

0.1232

2) random.randint(a, b) :- આ બીજી રીતે use થતું function છે. આ આપેલી range a અને b ની વચ્ચે નો કોઈ random Integer number આપે છે. આમાં ~~અને~~ a અને b બન્ને include થાય છે.

Syntax : random.randit (Start, end)

Example :

```
Print (random.randint(1, 49))
```

Output :

24

3) random.randrange (Start, Stop, Step) :- આ function Python ના normal range () function જેવું જ કામ કરે છે. પણ તે એ range માંથી એક random number return કરે છે.

Syntax : random.randrange (Start, Stop, Step)

Example :-

```
Print (random.randrange (35))
Print (random.randrange (0, 49, 2)
```

Output :

3

24

Q.3 What is Function? Write down Characteristics OF Function.

→ Function :-

Function એ code ના logically related statements ની group છે, જેનો ઉપયોગ કોઈ specific કામ કરવા માટે થાય છે.

Function ના ઉપયોગ થી મોટા અને અઘરા Program ને નાના-નાના ભાગમાં વેંચી શકાય છે.

એક વાર Function બનાવી ને test કરી લીધા પછી, તેને Program ના બીજા ત્રિજી વાર વારંવાર use કરી શકાય છે.

→ Characteristics / Advantages OF using Function.

1. Reduces Code length :- એક ની એક Code વારંવાર લખવી નાખી પડતી, જેના Program ની લંબાઈ ઘટી જાય છે.

2. Easy to Understand :- Code repeat ના ઘટતી હોવા થી આખા Program ને ~~સમજી શકાય~~ વાંચવી અને સમજવી એકદમ easy થઈ જાય છે.

3. Easy Error Handling and Debugging :-

જો Program માં કોઈ ભૂલ આવે, તો આખો જો specific function માં અંદર જ સુધારી શકાય છે, આખા Program માં નહિ.

4. Work Distribution in Team :- મોટા Project માં અલગ-અલગ team members ને અલગ-અલગ Function / Module બનાવવાનું કામ આપી શકાય છે. જેથી સમય બચે.

Q. 4. Creat a User-defined Function to Print the Fibonacci series of 0 to N numbers. -4

```
--> def Fibonacci(n):
        a = 0
        b = 1

        Print ("Fibonacci series:")

        For i in range(n):
            print(a, end = " ")

            c = a+b
            a = b
            b = c

num = int(input("Enter value of N: "))

Fibonacci(num)
```

```
num = int(input(" Enter a number : "))

result = factorial (num)

print ("Factorial is :", result)
```

Q.5 Develop a Python Program to find the Factorial of a number using a user-defined function. (4)

→

```
def Find_Factorial (n):
    if n <0 :
        return " Factorial not defined for
negative numbers"

    elif n == 0 or n == 1:
        return 1
    else:
        fact = 1
        for i in range (1, n+1):
            fact = fact * i
        return fact
```

Q.6 Explain built in functions in Python. Or Explain different built in functions of Python standard library.

→ Explanation :-

Python માં અમુક functions પહેલા થી જ બનાવેલા હોય છે, જેને Built-in functions કહેવાય છે.

આ functions Python ની standard library માં જ હોય directly available હોય છે. એટલે આપણે use કરવા માટે આપણે કોઈ પણ module (like import math) import કરવાની જરૂર નથી પડતી.

આનો મેન ફાયદો એ છે કે તે standard tasks માટે ready-made code આપે છે, જેથી Programmer નો time બચે છે.

Built in function :-

1. Print () :-

આ function નો ઉપયોગ Screen પર Output display કરવા માટે થાય છે.

Syntax : Print (value)

Example :- Print (24)
Output :- 24

2. input ()

આ Function ની મદદથી user પાસેથી keyword દ્વારા input લેવા માટે વપરાય છે.

Syntax :- input ("Prompt Message")

Example :- input ("write a message:")

3. abs ()

આ Function એક Built in Mathematical Function છે જેનો ઉપયોગ એ absolute value return કરવાનો થાય છે. કે જો integer નો વપરાશ Float માં કરીને કરવાનો.

Syntax :- abs (number)

Example :- Print (abs (-5))

Output :- 5

4. len () :

આ Function કોઈ પણ sequence (like String, list, Tuple) ની length, એટલે કે કેટલા કરતા items છે તે count કરીને આપે છે.

Syntax :- len (sequence)

Example :- a = [1, 2, 4, 5, 6]

Print (len(a))

Output:- 5

Q.7 Define Scope of a Variable. Apply global and local scope of variable Concepts in Python Program. (7) Or
Write a Python Program that explains scope of variable. (4)

→ Definition of Scope of a Variable:-

- Programming માં Variable નો scope એટલે Program નો એ ભાગ જ્યાં એ variable ને access કે modify કરી શકાય છે.

- Program માં variable ક્યાં declare કરીએ છીએ તેના આધારે તેનો scope બદલી પામે છે.

Types of Scope in Python:-

1. local Variable (local Scope):-

- જે variable કોઈ fun. ની અંદર declare કરવામાં આવતો હોય, તેને local variable કહેવાય છે.

- આ variable ને માત્ર Function ની અંદર જ access કરી શકાય છે. Function ની બહાર તેનો કોઈ ઉપયોગ નથી અને બહાર થી

તેની call કરવામાં આવશે તો error આવશે.

B. Global variable (Global scope):-

- જે variable Program માં કોઈ functions ની બહાર declare કરવામાં આવતી હોય, તેને Global variable કહેવામાં આવે છે.

- આ variable ને Program માં ગમે ત્યાંથી access કરી શકાય છે.

Example :-

```
global_var = 100

def my_function():
    local_var = 50

    Print ("Inside function:")
    Print ("-> Global variable value is:",
        global_var)
    Print ("-> local variable value is:",
        local_var)

my_function()

Print ("Outside function:")
Print ("-> Global variable value is:",
    global_var")
```

Q. 8 list any six built-in-function of list. Or Explain list methods and its built in functions.

→ list :- Python માં list એ એક mutable data structure છે.

list ની અંદર data ને add કરવા કે remove કે modify કરવા માટે Python માં ઘણા built in Function આવે છે.

Built in Function :-

1. Append (item) :-

આ method list ની એકદમ end માં એક નવા item add કરવા માટે વપરાય છે.

Syntax : list. append (item)

Example :-

```
my_list = [10, 20, 30]
Print (my_list. append (40))
```

Out Put :

[10, 20, 30, 40]

2. Extend (iterable)

જ્યારે એક list ની અંદર બીજા ઘણાબધા list add કરવા હોય, ત્યારે extend () નો ઉપયોગ ક

Syntax : list. extend (another_list)

Example :-

Print (my_list. extend ([50, 60]))

Output :-

[10, 20, 30, 40, 50, 60]

3. insert (index, item)

આ મેથડ ની ઉપયોગ list માં કોઈ ચોક્કસ position પર નવો item add / insert કરવા માટે થાય છે.

Syntax : list. insert (index, item)

Example :-

Print (my_list. insert (1, 15))

Output :-

[10, 15, 20, 40, 50, 60]

4. remove (item)

આ મેથડ list માંથી આપેલો item નો પહેલો Occurrence remove કરવા માટે ઉપયોગ થાય છે.

Syntax : list. remove (item)

Page
Date

Example :

Print (my_list. remove (30))

Output :-

[10, 15, 20, 40, 50, 60]

5. Pop (index) :-

આ મેથડ આપેલી index પર રહેલ item ને remove કરે છે અને તેની item return પણ કરે છે. જો index ના આપેલ ન હોય તો છેલ્લી item remove કરે છે.

Syntax: list. Pop (index)

Example :-

Print (my_list. Pop(2))

Output :-

[10, 15, 40, 50, 60]

6. sort ()

આ મેથડ list ની items ને ascending કે descending order માં sort કરે છે.

Syntax : list.sort()

Example :-

```
Print( my_list.sort(reverse = True))
```

Output :-

[60, 50, 40, 15, 10]

Q.9. Explain tuple operations and functions. 4 OR Write a program to demonstrate tuple functions and operations. (4)

→ Explanation :-

- Python માં Tuple એ એક ordered અને immutable data structure છે. Tuple ની items ને round brackets () ની અંદર ગોઠવવામાં આવે છે.

- list ની જેમ આમાં items add, remove કે modify નથી કરી શકતા, પણ આપેલ આમાં પર અલગ-અલગ operations અને functions apply કરી શકીએ છીએ.

Operations :-

Page
Date

Q.9 Write a Program to traverse the String using for and while loop.

→ Traversing using for loop.

```
Str1 = input(" Enter a String: ")

Print (" Traversing using for loop: ")

For ch in str1 :
    Print (ch)
```

→ Traversing using while loop.

```
Str1 = input (" Enter a String: ")

i = 0

Print (" Traversing using while loop.")

While i < len (Str1):
    Print (Str1 [i])
    i = i+1
```