**Assignment -3**

1. **Why are functions advantageous to have in your programs?**

Ans=> Functions enable reuse of code, improve maintainability and scalability.

1. **When does the code in a function run: when it’s specified or when it’s called?**

Ans=> When it’s called.

1. **What statement creates a function?**

Ans=> ‘def’ keyword is a statement of defining the functions. You start a function with the def keyword, specify a name followed by a colon (:)

1. **What is the difference between a function and a function call?**

Ans=> **A function** is a block of code that does a particular operation and returns a result. It usually accepts inputs as parameters and returns a result. The parameters are not mandatory.

**A function call** is the code used to pass control to a function.

**5. How many global scopes are there in a Python program? How many local scopes?**

Ans=> We also declare a variable outside any other python Variable scope, this makes it global scope. So

There's only one global Python scope per program execution.

Local scopes : Whenever you define a variable within a function, its scope lies ONLY within the function. It is accessible from the point at which it is defined until the end of the function and exists for as long as the function is executing. So the scope of local scopes are n times inside n number of functions.

**6. What happens to variables in a local scope when the function call returns?**

Ans=> When the function call returns, the local scop of variables will be reassigned the same value declared in that variables.

**7. What is the concept of a return value? Is it possible to have a return value in an expression?**

Ans=> A return is a value of a function when it completes its task.

You can use that value in a math expression or any other kind of expression in which the value has a logical or coherent meaning.

**8. If a function does not have a return statement, what is the return value of a call to that function?**

Ans=> A function without **‘return’** statement, returns None.

**9. How do you make a function variable refer to the global variable?**

Ans=> If I want to refer the global variable into a function variable, I can use ‘global’ keyword inside the function in time of declaration of a function.

Example –

def set\_globvar():

global globvar

globvar = 1

def display\_globvar():

print(globvar)

set\_globvar()

display\_globvar()

1**0. What is the data type of None?**

Ans=> None is data type of its own, basically it’s a NonType.

**11. What does the sentence import areallyourpetsnamederic do?**

Ans=>

**12. If you had a bacon() feature in a spam module, what would you call it after importing spam?**

Ans=>

import spam

spam.bacon()

**13. What can you do to save a programmed from crashing if it encounters an error?**

Ans=> Error handling can be used to notify the user of why the error occurred and gracefully exit the process that caused the error.

**14. What is the purpose of the try clause? What is the purpose of the except clause?**

Ans=> Try clause allows you to catch one or more exceptions in the ‘try’ clause and handle each of them in the expect clauses.

Example - a = 10

b = 0

try:

c = a / b

print(c)

except ZeroDivisionError as error:

print(error)

finally:

print('Finishing up.')