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

**Ans:** Following are the benefits of having function in a program

* Functions can be used and shared by other programmers.
* Same piece of code can be used for several times with the help of functions.
* Complex programs can be broken up into smaller programs with the help of functions.
* Same function can be called several times with different inputs.

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

**Ans:** To run the code in a function, the function needs to be called with appropriate input arguments.

1. What statement creates a function?

**Ans:** A function can be created by using ***def*** statement.

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

**Ans:** In Python, a function is a block of code which can perform a specific task. A function can be defined using ***def*** keyword, followed by the function name and a set of parentheses that may include input arguments. Example of a function in Python to add two numbers is depicted below:

def add\_num(a,b):

  c = a+b

  print("The sum of two numbers =",c)

The above function can be executed by calling it using the name add\_num with two input arguments e.g.

add\_num(3,5)

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

**Ans:** There is one global scope and one local scope in Python.

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

Ans: Local variables in Python are initialized inside a function which belongs only to that particular function. Once the function call returns, the local variable is erased from the memory. So, local variables cannot be accessed outside the function.

Example:

def My\_Name():

  Name="Arnav Kumar Sahu"

  print(Name)

My\_Name() # This function will print the Name

print(Name) # This statement cannot print the Name, as it is a local variable inside the function My\_Name.

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

**Ans:** A ‘return value’ is a value that is returned after the evaluation of a script or function file. Yes it is possible to have a return value in an expression.

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

**Ans:** If there is no return statement in a function, it will not return anything.

Example:

def add(x,y):

  result=x+y

add(5,4) # It will not return any output.

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

**Ans:** A variable declared inside a function, is by default a local variable. A local variable can be made global using the keyword ‘global’.

Example:

def My\_Name():

  global Name

  Name="Arnav Kumar Sahu"

print(Name) # The variable 'Name' will be printed, as it is declared as a global variable

1. What is the data type of None?

**Ans:** The data type of None is ‘NoneType’.

1. What does the sentence import areallyourpetsnamederic do?

**Ans:** It will import a module named as areallyourpetsnamederic.

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

**Ans:** The function can be called with the statement spam.bacon().

1. What can you do to save a programme from crashing if it encounters an error?

**Ans:** By keeping the lines which are giving error within a try clause.

Example:

try:

  print(x)

except:

  print("The variable x is not defined")

1. What is the purpose of the try clause? What is the purpose of the except clause?

**Ans:** The **try clause** is used to check some code for errors i.e the code inside the try block will get executed when there is no error.

On the other hand, **except clause** is executed if there is some error in the try clause.