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

**Solution1:**

Functions help us to avoid writing a lengthy code again and again for the same task.

We can enter the input in the function and get the desired result in one go.

Example: Let us suppose we have a list which can consist of integers, floats, complex numbers, list, tuple, set and dictionary.

We want to extract keys of the dictionary. Then the following code can be written to extract keys of dictionary if any kind of list is provided inside the function.

Code:-

lst=[1,2,"alka",{45,66},(23,"asdf"),34+5j,{"key1":1,"key2":2}]

key\_lst=[]

def enter\_lst(a):

for i in lst:

if type(i)==dict:

key\_lst.append(i.keys())

return key\_lst

enter\_lst(lst)

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

**Solution2:**

Code in a function runs when it's called.

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

**Solution3:**

The word “def” creates a function like:-

def fun(a):

if a==int:

print(a)

else:

print("a is not an integer")

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

**Solution4:**

A function call means invoking or calling that function. Unless a function is called there is no use of that function. ... So the difference between the function and function call is, A function is procedure to achieve a particular result while function call is using this function to achieve that task.

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

**Solutio5:**

One global scope, and a local scope is created whenever a function is called.

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

**Solution6:**

When the execution of the function terminates (returns), the local variables are destroyed.

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

**Solution7:**

Return value: Return values are just what they sound like — the values that a function returns when it has completed.

def enter\_number():

return “enter\_number”

this code will return ‘enter\_number’

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

**Soluton8:**

It will not give any output.

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

**Solution9:**

We can define a variable outside the function. That variable will be global variable.

Ex: Variables that are created outside of a function (as in all of the examples above) are known as global variables.

Global variables can be used by everyone, both inside of functions and outside.

x = "awesome"

def myfunc():

print("Python is " + x)

myfunc()

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

**Solution10:**

None is used to define a null value. It is not the same as an empty string, False, or a zero. It is a data type of the class NoneType object. Assigning a value of None to a variable is one way to reset it to its original, empty state.

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

**Solution11:**

ModuleNotFoundError: No module named 'areallyourpetsnamederic'

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

**Solution12:**

This function can be called with spam.bacon().

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

**Solution13:**

Place the line of code that might cause an error in a try clause.

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

**Solution14:**

The code that could potentially cause an error goes in the try clause.The code that executes if an error happens goes in the except clause.