Assignment:3

Neeraj Pal (neerajpal1801@gmail.com)

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

Ans: With the help of functions, we can avoid rewriting the same logic or code again and again in a program. In a single Program, we can call Python functions anywhere and also call multiple times.

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

Ans: A function defined inside another function is called a nested function.

3. What statement creates a function?

Ans: The “def” keyword is a statement for defining a function in Python. we start a function with the def keyword, specify a name followed by a colon (:) sign. The “def” call creates the function object and assigns it to the name given. You can further re-assign the same function object to other names.

4. 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: In Python, there are four types of scopes, which are as follows: Global Scope, Local Scope, Enclosing Scope  and built-in—depending on where you are in the code.

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

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

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 that a function returns to the calling script or function when it completes its task. A return value can be any one of the four variable types: handle, integer, object, or string. The type of value your function returns depends largely on the task it performs.

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

Ans: If no return statement appears in a function definition, control automatically returns to the calling function after the last statement of the called function is executed. In this case, the return value of the called function is undefined.

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

Ans: Normally, when you create a variable inside a function, that variable is local, and can only be used inside that function. To create a global variable inside a function, you can use the global keyword.

10. What is the data type of None?

Ans: Definition and Usage. The None keyword is used to define a null value, or no value at all. None is not the same as 0, False, or an empty string. None is a data type of its own (NoneType) and only None can be None.

11. What does the sentence import areallyourpetsnamederic do?

Ans: What does the import areallyourpetsnamederic statement do? That import statement imports a module named areallyourpetsnamederic.

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

Ans: If you had a function named bacon() inside a module named spam, how would you call it after importing spam ? This function can be called with spam. bacon().

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

Ans: When it encounters an error, the control is passed to the except block, skipping the code in between. As seen in the above code, we have moved our code inside a try and except statement. Try running the program and it should throw an error message instead of crashing the program.

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

Ans: The try block lets you test a block of code for errors. The except block lets you handle the error. The else block lets you execute code when there is no error. The finally block lets you execute code, regardless of the result of the try- and except blocks.