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

It helps to divide the large programs into small groups so that we can read the code and debug the program faster and better. Functions stop us from writing the same logic various times. We can bind the logic in one def and then call the same over and over.

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

When a function is "called" the program "leaves" the current section of code and begins to execute the first line inside the function.

1. What statement creates a function?

A function is created with the def keyword. The statements in the block of the function must be indented. The def keyword is followed by the function name with round brackets and a colon. The indented statements form a body of the function.

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

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.

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

When we use an unqualified name inside a function, Python searches three scopes—the local (L), then the global (G), and then the built-in (B).

Variables that are defined inside a function body have a local scope, and those defined outside have a global scope. This means that local variables can be accessed only inside the function in which they are declared, whereas global variables can be accessed throughout the program body by all functions.

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

The local variable scope ends as function call completed. We cannot directly use local variable outside the function call.

There is a way to access the local variables of a function using pointers, by creating another pointer variable that points to the variable to be returned and returning the pointer variable itself.

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

A return is a value that a function returns to the calling script or function when it completes its task.

The statements after the return statements are not executed. If the return statement is without any expression, then the special value None is returned

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

The return value of a call to that function Should be None here.

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

To create a global variable inside a function, you can use the global keyword before he variable.

1. What is the data type of None?

None is used to define null objects and variables, None in Python is actually an object implemented by NoneType class. Just like False is a boolean object implemented by the bool class.

1. What does the sentence import areallyourpetsnamederic do?

That import statement imports a module named areallyourpetsnamederic.(not a defined module in python).

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

Spam.bacon()

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

Place the line of code which might cause error in try clause.

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

Place the line of code which might cause error in try clause.

The code that executes if an error happens goes in except clause.