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

Ans:

The function of the functions are to make the task modular and minimize the code redundancy as per requirement.

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

Ans:

The function runs only when it’s called. Even the debuggers initialy don’t analyze it for functional issues if there is no syntax error.

1. What statement creates a function?

Ans:

The statement “def function\_name(params): function statements” creates a function.

The “def” keyword is used to define a function.

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

Ans:

The function is an entity that holds the code, while the function call executes the specified code.

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

Ans:

There is only one global scope in python, i.e. the variables and functions declared in the outermost block. There can be multiple local scopes, that are declared inside functions as well as class methods.

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

Ans:

When a function finishes running in Python and returns its result, any variables that were created inside the function disappear. They cease to exist and cannot be used or accessed from outside the function. It's as if they vanish into thin air, leaving no trace behind.

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

Ans:

In Python, a return value is like a special package that a function can give back to the code that called it. When a function runs, it might do some things and come up with a result. This result, called the return value, is like a gift that the function hands back to the code that asked for it. The code can then use this return value to do more things or get the final answer it needs. Yes. it 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 a function does not have a return statement, the return value of a call to that function is None. None is a special object in Python that represents the absence of a value. So, if a function doesn't explicitly return a value, it is assumed to implicitly return None.

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

Ans:

To make a function variable refer to a global variable in Python, you can use the global keyword before the variable inside the function. This lets Python know that the variable is a global one, not a local one.

1. What is the data type of None?

Ans: The datatype of None is NoneType.

1. What does the sentence “import areallyourpetsnamederic” do?

Ans: The “import areallyourpetsnamederic “ sentence imports module named areallyourpetsnamederic.

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

Ans:

There is atleast two ways to do it:

1. import spam

spam.bacon()

1. from spam import bacon

bacon()

1. from spam import bacon as b\_func

b\_func()

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

Ans:

We can use try and except method to either wait for the response of asynchronous functions or we can continue with the execution in except clause, that runs if the try block fails and it’s also capable of catching the error and log it using logging or print.

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

Ans:

Try clause is used on statements that can cause error or exceptions in the code. The try clause catches the exceptions and executes the except clause upon failing. There is also a finally clause that waits till the completion of try and except clause and then executes some other code. Personally I have used it extensively in web-scrapping and API calls.