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

Functions help us avoid rewriting the same code again as again. We can recall the function whenever we require.  We can track a large program easily when it is divided into multiple functions.

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

When it is called

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

We define function in python by def keyword which is followed by an identifier (any name) and then by parenthesis () and a colon. The next thing is to hit enter and provide a tab or 4 spaces for writing the code or logic. At the end we provide return statement to exit from the function. The return statement can contain an expression to execute once the function is called.

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

Defining a function properly and providing the necessary code or logic is called function. On returning the value by providing return statement we exit out from the function. But this does not run or execute the statement, code or logic. Once that function is called and a value is provided to the necessary argument, the function will run and execution will happen.

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

Variables used in function can be global or local as per their usage. A variable when created and is used inside a particular function then it is called local scope. They are available in two forms, primarily when a variable is created inside a function and is available inside the function; secondly a local variable can also be accessed from a function within another function. While a variable when declared outside a function or in main body in a python code is called global variable. But they can be available locally and also globally. A global variable if is in the main body and also within the function body; python will treat them as two different variable. One who is within the function will be printed first and execution of variable in main body will be followed.

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

When the function call returns it runs the code as described within the function. Once execution is done and function is terminated the local variable dies or of no use. This is because a when function is called, the local variables are not called at the same time. Hence once we are exiting the function by a return statement, the local variables are not used anywhere in the program.

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

The return value is the expression following a return statement in a function. It is possible to have a return value in an expression.

For example of return in an expression:

def myfunc():

x = 300

return x+1

myfunc()

Output----> 301

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

If we don't explicitly use a return value in a return statement, or if we totally omit the return statement, then Python will implicitly return a default value for us. That default return value will always be none.

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

A local variable inside a function can be referred as global when we use keyword ‘global’ followed by the variable.

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

None keyword is used when the value is null or no value but is different from 0, false or empty string. It is a data type on its own, None type.

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

It will execute an error, as we haven’t assigned any module named areallyourpetsnamederic. Hence there is no question of importing it.

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

import spam

spam.bacon()

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

This can be avoided by using try and exception handling within each module. This is done by writing additional code under except block in a program to give proper messages or instructions to the user on encountering an exception. If the try block fails it will automatically generate the code under exception.

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

Try clause provide the original code or logic for which we are programming, it will execute the code that the user wants and this helps in avoid unnecessary crashing if it fails. The exception handling after try block fails to execute. It is known as exception handling.