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

**A function is a block of code which only runs when it is called. Its advantages are:**

* Reducing duplication of code.
* Decomposing complex problems into simpler pieces.
* Improving clarity of the code.
* Reuse of code.
* Information hiding.

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

**When it is called**

3. What statement creates a function?

**Function blocks begin with the keyword def followed by the function name and parentheses. Any input parameters or arguments should be placed within these parentheses**

**The first statement of a function can be an optional statement - the documentation string of the function or *docstring*.**

**The statement return [expression] exits a function, optionally passing back an expression to the caller. A return statement with no arguments is the same as return None.**

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

**Defining a function only gives it a name, specifies the parameters that are to be included in the function and structures the blocks of code.**

**On the other hand, a function is executed only when it is called.**

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

**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.**

**However, it is possible to have global scope using ‘global’ keyword inside a function. Therefore, there are 2 global scopes, while there is only one local scope**

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

**A local variable, however, has a limited scope: it exists only within the block that it is declared in. Once that block ends, the variable is destroyed and its values lost**

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

**The python return statement is used in a** [**function**](https://www.askpython.com/python/python-functions) **to return something to the caller program. If the return statement contains an expression, it’s evaluated first and then the value is returned.**

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

**None**

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

**By using the ‘global’ keyword**

10. What is the data type of None?

**NoneType**

11. What does the sentence import areallyourpetsnamederic do?

**ModuleNotFoundError: No module named 'areallyourpetsnamederic'**

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

**Spam.bacon()**

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

**Exception handling using try - except**

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

**When the function is called, the try clause will run. If no exceptions are raised, the program will run as expected.**

**But if an exception is raised in the try clause, the flow of execution will immediately jump to the except clause to handle the exception.**