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

Functions help organize our code into reusable pieces of code, functions can be called from anywhere in your Python code, even within other functions. This helps:

* running same piece of code multiple times
* breaking long complex programs into smaller components
* functions can be shared and used by other developers as well

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

The code inside a function runs only when it is called

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

‘def’ statement is used in python to create a function

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

Function or function definition is creation or declaring the function, example: what parameters to accept, what actions to perform and what values to return. On the other hand function call is invoking this function to get some required actions/processing done.

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

Local scope includes all variables defined in the current function and global scope includes variable defined outside of the current function. So one global scope and one local scope

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

The local scope variables will be deleted from the memory (local stack) and they will not be available after the function call returns.

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

Return values are used by a function to enable them to send Python objects back to the caller code, so that they can use these return values to perform further computation in their program.

The result of an expression can be sent as a return, also an expression can rely on a return value from a function.

**Example:** 4 – sqrt(16)

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

If a function does not have a return statement, then the return value of a call to that function is ‘None’ in Python

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

You can use a global variable within other functions by declaring it as global within each function that assigns a value to it, example:

globvar = 0

def set\_globvar\_to\_one():

global globvar # Needed to modify global copy of globvar

globvar = 1

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

None keyword is an object and is a data type of none type class in PYthon

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

The above import statement in Python will allow our python code or file to make use of the reusable components defined in areallyourpetsnamederic module.

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

spam.beacon()

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

Exception handling should be in place for handling runtime errors if encountered while running a program.

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

The try except statement can handle exceptions. Exceptions may happen when you run a program.

Exceptions are errors that happen during execution of the program. Python won’t tell you about errors like syntax errors (grammar faults), instead it will abruptly stop.

* **try**: the code with the exception(s) to catch. If an exception is raised, it jumps straight into the except block.
* **except**: this code is only executed if an exception occured in the try block. The except block is required with a try block, even if it contains only the pass statement.