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

**Ans: for reusability and reduces the duplication in code**

2. When does the code in a function run: when it's specified or when it's called? **Ans: When it is called, the code inside the function will execute**

3. What statement creates a function? **Ans: def statement creates a function**

4. What is the difference between a function and a function call**? Ans: function defines the code inside it and function call executes those codes inside the function**

5. How many global scopes are there in a Python program? How many local scopes? **Ans: Only one global scope and a local scope whenever the function is called.**

6. What happens to variables in a local scope when the function call returns? Ans: **The value is destroyed and is available only inside that function not outside of it**

7. What is the concept of a return value? Is it possible to have a return value in an expression**? Ans: Return value is the result of an evaluation of a function and this can be part of the expression like any other value.**

8. If a function does not have a return statement, what is the return value of a call to that function? **Ans: None, no value is returned**

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

**Ans: If the variable is defined inside the function, it should be defined using global keyword**

10. What is the data type of None? **Ans: Nonetype**

11. What does the sentence import areallyourpetsnamederic do? **Ans : import the specified module** **areallyourpetsnamederic**

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

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

**Ans: enclose the program inside a try clause**

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

**Ans: Try – let us to execute a block of code to check for errors**

**Except – lets you to handle the error**