1.Why are functions advantageous to have in your programs?
A: reusability of code

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

3.What statement creates a function?
A: def <function_name>

4.What is the difference between a function and a function call?
A: Function is a snippet of code where the logic lies whereas function call is when you invoke the function which runs the snipper of code.

5.

6.What happens to variables in a local scope when the function call returns?
A: local scope variable is accessible only inside the function. So when we return a value from the function only the value is returned and those variables inside the function cannot be accessed outside the function.

7.What is the concept of a return value? Is it possible to have a return value in an expression?
A: return is used to come out of the function with a value once the desired value is obtained. It is possible to return an expression
Eg :
a= 10
b=20
return a+b

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

9.How do you make a function variable refer to the global variable?
A: using the keyword global

10.What is the data type of None?
A: NoneType

11.What does the sentence import areallyourpetsnamederic do?
A: imports the function from another python file

12.If you had a bacon() feature in a spam module, what would you call it after importing spam?
A: you can access the bacon function

13.What can you do to save a programme from crashing if it encounters an error?
A: can add try except block to catch the error and handle error scenarios safely

14.What is the purpose of the try clause? What is the purpose of the except clause?
A: inside try clause we need to have the code that we think can cause error. Inside clause we need to mention how the error needs to be handled.