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

**Ans**: Because of code reusability

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

**Ans**: Code in function is executed when function is executed, i.e when it get called

1. What statement creates a function?

**Ans**: def function\_name(different parameters):

Body

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

**Ans:** function is a piece of code which increases code reusability, and function call is stmt by which we use this feature.

**Example:**

def fun\_name(): # function

body

return\_value = fun\_name() # calling function

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

**Ans:** Only one global scope

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

**Ans** They get deleted by garbage collectors, after completion of function local scope function li

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

**Ans** return is value return by function after completing its task.

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

**Ans:** None

Example:

def non\_return():

print("abbcd",end=" ")

r = non\_return()

print(r)

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

**Ans**: use the global keyword

10. What is the data type of None?

**Ans:** print(type(r))

<class ‘NoneType’>

11. What does the sentence import areallyourpetsnamederic do?

**Ans:**

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

**Ans:** import spam as s

s.bacon()

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

Ans:

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

**Ans:**

Try block let you test you block of code and except let you handle the error.