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

Answer 1:- Functions reduce the need for duplicate codes and if certain set of codes are to be used again and again, then in such cases we can define the function. Afterwards rather than writing the same set of codes again and again we can call it using function. It makes are program shorter and easy to understand.

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

Answer 2:- It runs in the function when it is called.

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

Answer 3:- def function\_name ():

Pass

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

Answer 4:- function is when it is defined using ‘def’ keyword and when we have to use the function at somewhere, we call the function.

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

Answer 5: There's only one global Python scope per program execution. A variable created inside a function belongs to the local scope of that function, and can only be used inside that function whereas A variable created in the main body of the Python code is a global variable and belongs to the global scope

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

Answer 6: When a function returns, the local scope is destroyed, and all the variables in it are forgotten. A local variable becomes undefined after the function call completes

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

Answer 7: The Python return statement is a key component of functions and methods. We can use the return statement to make functions send Python objects back to the caller code. These objects are known as the function’s return value. A return value is the value that a function call evaluates to. Like any value, a return value can be used as part of an expression.

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

Answer 8:- The return value is None. The function then returns None Type object.

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

Answer 9:- A global statement will force a variable in a function to refer to the global variable. If you want to refer to a global variable in a function, you can use the global keyword to declare which variables are global.

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

Answer 10:- The data type of None is NoneType

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

Answer 11: It will import a module named “areallyourpetsnamederic”.

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

Answer 12:-This function can be called with spam.bacon().

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

Answer 13:- we can use exception handling. If there is possibility of encountering an error in a code, we can put it into try block and then use except block to handle the exceptions

try:

print(x)

except:

print("An exception occurred")

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

Answer 14:- The code that could potentially cause an error goes in the try clause. The code that executes if an error happens goes in the except clause.