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

Functions stop us from writing the same logic various times. We can bind the logic in one def and then call the same over and over so functions reduce the need for duplicate code.. Functions divide the large programs into small groups so that we can read the code, and debug the program faster and better. Many persons can work on the same program by assigning different methods to each of them.

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

The code in a function executes when the function is called, not when the function is specified.

3. What statement creates a function?

The “def” keyword is a statement for defining a function in Python. You start a function with the def keyword, specify a name followed by a colon (:) sign. The “def” call creates the function object and assigns it to the name given.

def function\_name(arguments):

return arguments

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

A function is a block of code that does a particular operation and returns a result. It usually accepts inputs as parameters and returns a result. The parameters are not mandatory and function call is the code used to pass control to a function.

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

There’s only one global Python scope per program execution. This scope remains in existence until the program terminates and all its names are forgotten. Otherwise, the next time you were to run the program, the names would remember their values from the previous run.

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

A local variable **retains its value** until the next time the function is called. A local variable becomes **undefined** after the function call completes.  When a function returns, the local scope is destroyed, and all the variables in it are forgotten.

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

A return is a value that a function returns to the calling script or function when it completes its task. yes it is possible to have return value in an expression.

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

 if you totally omit the return statement, then Python will implicitly return a default value for you. That default return value will always be None .

9. How do you make a function variable 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?

None is used to define a null value. It is not the same as an empty string, False, or a zero. It is a data type of the class NoneType object.

11. What does the sentence import areallyourpetsnamederic do?

That import statement imports a module named areallyourpetsnamederic.

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

spam.bacon()

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

I  will use try and catch for preventing crash

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

The try block lets you test a block of code for errors. The except block lets you handle the error.