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

🡪A function is a group of statements that are executed whenever the function is called to perform a specific designated task.

🡪Functions reduce the need for duplicate code. This makes programs shorter, easier to read, and easier to update.

🡪Using a function, it is possible to reduce the size of a program by calling and using the function at different places in the program.

🡪Functions help in code modularity, which means that the entire code is divided into separate blocks, each of which is self-contained and performs a different task.

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

* A function declaration tells the compiler about a function name and how to call the function.
* The code in a function executes when the function is called, not when the function is defined.
* When a function is "called" the program "leaves" the current section of code and begins to execute the first line inside the function.

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

To create a function, we must first declare it and give it a name, and then we follow it by a function definition,

def function\_name() :

{ function body

}

**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.
* A function call is the code used to pass control to a function.
* The code in a function executes when the function is called, not when the function is defined.

#def of func

def my\_func():

pass

my\_func() #calling of function

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

* Variables defined inside the function are known as Local variable
* Global varaibles can be accessed anywhere in the script unlike local variable who's scope is defined within the function only.
* There's only one global Python scope per program execution. This scope remains in existence until the program terminates .

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

When the execution of the function terminates (returns), the local variables are destroyed.

**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. A return value can be any one of the four variable types: handle, integer, object, or string.
* A return statement is used to end the execution of the function call and “returns” the result (value of the expression following the return keyword) to the caller. The statements after the return statements are not executed. If the return statement is without any expression, then the special value None is returned.

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

* If no return statement appears in a function definition, control automatically returns to the calling function after the last statement of the called function is executed. In this case, the return value of the called function is undefined.
* If the return statement is without any expression, then the special value None is returned.

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

global keyword

Normally, when you create a variable inside a function, that variable is local, and can only be used inside that function. To create a global variable inside a function, you can use the global keyword.

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

None is a data type of its own (NoneType) and only None can be None.

NoneType in Python is a data type that simply shows that an object has no value/has a value of None.

None keyword is an object, and it is a data type of the class NoneType .

None – None is an instance of the NoneType object type. And it is a particular variable that has no object value.

None is not the same as 0, False, or an empty string.

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

imports a module named areallyourpetsnamederic

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

import spam   
spam.bacon()

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

error handling can be used to notify the user of why the error occurred and gracefully exit the process that caused the error.

Error-Handling Practices

Exception handling using try - except and and try-except-else and try - except - finally

**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. The else block lets you execute code when there is no error.