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

**Function helps code organizaztion,reusability,maintainability making them fundemental building blocks of software development.they enable us to write cleaner,more efficient and more managable code which is crucial for both small and large scale software project.**

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

**Code in a function runs when it is called.**

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

**Def function\_name():**

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

**Function**:

* 1. A function is a block of code that performs a specific task or set of tasks.
  2. It is defined with a name and may take input parameters (arguments).
  3. Functions can be thought of as reusable units of code that encapsulate a particular functionality.
  4. Functions are typically defined once in your code and can be called multiple times from different parts of your program.

**Function Call**:

* 1. A function call is the action of invoking (or running) a function to execute its code.
  2. When we call a function, you provide any required input values (arguments) based on the function's parameter list.
  3. The function call triggers the execution of the code within the function's body, using the provided arguments.
  4. After the function completes its task, it may return a value to the calling code, if specified.

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

**In a Python program, there is typically one global scope and multiple local scopes**

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

**When a function call returns in Python, the variables that were defined within the local scope of that function typically go out of scope and are no longer accessible. This means that the local variables cease to exist once the function execution is completed .**

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

**functions can return a value to the caller, which allows us to use the result of the function's.**

**def add\_num(x,y):**

**result=x+y**

**return result**

**total=add\_num(4,5)**

**print(total)**

**we can also use functions that return values within expressions. This means we can directly use the returned value in another operation or expression.**

**result=add\_num(5,6)**

**Print(result)**

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

**If a function in Python does not have a return statement, or if it has a return statement without an expression to specify a value, the function will implicitly return a special value called None. None is a built-in constant in Python that represents the absence of a value or a null value.**

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

**In Python, if you want to make a function variable refer to a global variable, you can use the global keyword.**

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

**Datatype of None is ‘Nonetype’.**

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

**ImportError: No module named 'areallyourpetsnamederic'**

1. **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?

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

**Python, the try and except clauses are used together to handle exceptions, which are runtime errors or exceptional situations that can occur during the execution of a program.**

**Try clause is used to enclosed a block of code where we anticipate possibility of exception occuring**

* + **It allows us to test a block of code for exceptions without actually halting the program's execution when an exception is raised.**
  + **If an exception occurs within the try block, the execution of the try block is immediately terminated, and the program jumps to the corresponding except block.**

1. **except Clause:**
   * **The except clause is used to define how to handle specific exceptions that may occur within the associated try block.**
   * **You can have multiple except clauses to handle different types of exceptions.**
   * **When an exception occurs in the try block, Python checks each except clause to see if it matches the type of exception that was raised. If a match is found, the code within that except block is executed.**
   * **If there is no matching except clause, the exception is propagated up the call stack, and if not caught there, it may result in the program terminating with an error message.**