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

**Answer-** The use of functions makes a program more readable. It's frequently difficult to read a large program. Breaking the code down into smaller functions keeps the program structured, understandable, and reusable. The function can be reused countless times after it is defined.

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

**Answer-** 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?**

**Answer-** The “def” keyword is a statement for defining a function in Python.

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

**Answer-** 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.

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

**Answer-** There's only one global Python scope per program execution & four local scopes.

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

**Answer-** 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?**

**Answer-** 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. The type of value your function returns depends largely on the task it performs.

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

**Answer-** 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.

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

**Answer-** To create a global variable inside a function, you can use the global keyword.

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

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

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

**Answer-** 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?**

**Answer-** 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-** If an error occurs in a program, we don't want the program to unexpectedly crash on the user. Instead, error handling can be used to notify the user of why the error occurred and gracefully exit the process that caused the error.

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

**Answer-** 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. The finally block lets you execute code, regardless of the result of the try- and except blocks.