1. Functions are advantageous to have in programs because they promote code reuse, improve code organization, and make the code easier to understand and maintain. Functions allow you to break down a large program into smaller, more manageable chunks of code, and you can call the function multiple times from different parts of the program.

2. The code in a function runs when it is called by the program, not when it is specified. The program executes the function call statement, which then invokes the function code and executes it.

3. The statement that creates a function in Python is the def statement. For example, to create a function that adds two numbers, you could use the following code:

Ex:

def add\_numbers(x, y):

return x + y

4. A function is a block of code that performs a specific task, while a function call is the statement that invokes the function and executes the code within the function.

5. There is only one global scope in a Python program, but there can be multiple local scopes. Local scopes are created whenever a function is called, and they are destroyed when the function call returns.

6. When the function call returns, the variables in the local scope are destroyed, and their values are no longer accessible from outside the function.

7. A return value is the value that a function returns when it completes its task. It is possible to have a return value in an expression, such as assigning the return value to a variable or passing it as an argument to another function.

8. If a function does not have a return statement, the return value of a call to that function is None.

9. To make a function variable refer to a global variable, you can use the global keyword inside the function. For example:

Ex:

x = 10

def my\_function():

global x

x = 20

This code defines a global variable x with a value of 10, and then defines a function that uses the global keyword to modify the value of x to 20.

10. The data type of None is NoneType. None is a special constant in Python that represents the absence of a value.

11. The sentence import areallyourpetsnamederic does not have any meaning in Python. It is not a valid module name, so trying to import it will result in an error.

12. If you had a bacon() feature in a spam module, you could call it after importing spam using the following code:

Ex:

import spam

spam.bacon()

This code imports the spam module and then calls the bacon() function from within the module using dot notation.

13. To save a program from crashing if it encounters an error, you can use exception handling. Wrap the code that might raise an error in a try block, and then use an except block to handle the error if it occurs. You can also use the finally block to run code that should always execute, regardless of whether an error occurred.

14. The try clause is used to wrap code that might raise an exception. The purpose of the except clause is to handle the exception if it is raised. If an exception is raised within the try block, the program jumps to the except block and executes the code within it. The except block can also specify the type of exception it can handle.