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

**Ans:** Functions reduce the need for duplicate code. This makes programs shorter, easier to read, and easier to update. ... A function call is what moves the program execution into the function, and the function call evaluates to the function's return value.

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

**Ans:** The code in a function executes when the function is called, not when the function is defined.

1. What statement creates a function?

**Ans:** A statement that executes a function. The def statement defines (that is, creates) a function. It consists of the name of the function followed by a list of arguments enclosed in parentheses. Using the output from one function call as the input to another. A statement that creates a new function, specifying its name, parameters, and the statements it executes.

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

**Ans:** 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.

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

**Ans:** 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. There is only one  a local scope is created whenever a function is called.

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

**Ans:** A Local Variable Retains Its Value Until the Next Time The Function Is Called A Local Variable Becomes Undefined After. The Function Call Completes. The Local Variable Can Be Used Outside the Function Any Time After the Function Call Completes. When a function returns, the local scope is destroyed, and all the variables in it are forgotten.

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

**Ans:** 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. A return value is the value that a function call evaluates to. Like any value, a return value can be used as part of an expression.

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

**Ans:** A function without an explicit return statement returns None . In the case of no arguments and no return value, the definition is very simple. Calling the function is performed by using the call operator () after the name of the function.

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

**Ans**: A global statement will force a variable in a function to refer to the global variable.

1. What is the data type of None?

**Ans:** The None keyword is used to define a null value, or no value at all. None is not the same as 0, False, or an empty string. None is a data type of its own (NoneType) and only None can be None.

1. What does the sentence import areallyourpetsnamederic do?

**Ans:** That import statement imports a module named areallyourpetsnamederic . (This is not a real Python module, by the way.) This function can be called with spam.

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

**Ans:** This function can be called with spam.bacon().

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

**Ans:** Prevents program from crashing if an error occurs instead, error handling can be used to notify the user of why the error occurred and gracefully exit the process that caused the error.

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

**Ans:** A try clause is executed up until the point where the first exception is encountered. Inside the except clause, or the exception handler, one can determine how the program responds to the exception. One can anticipate multiple exceptions and differentiate how the program should respond to them. The code that could potentially cause an error goes in the try clause. The code that executes if an error happens goes in the except clause.