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

* 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.
* 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. This makes each block implementation and debugging much easier.
* In top-down structured programming, dividing a program into function is more efficient and easy to understand.
* If you are just using the function in your program then you don’t have to worry about how it works inside.  
  **Example:** printf()

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

The code inside a function is executed when the function is **invoked**. It is common to use the term "**call a function**" instead of "**invoke a function**". It is also common to say "call upon a function", "start a function", or "execute a function".

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

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

You start a function with the def keyword, specify a name followed by a colon (:) sign.

The “def” call creates the function object and assigns it to the name given.

1. **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.  
E.g: Function add(a,b)  
return a+ b  
A function call is the code used to pass control to a function.  
E.g.: b = add(5,6)  
Now b will have the value 11.

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

There is one global scope, and a local scope is created whenever a function is called

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

Yes they will be deleted. Every time you call a function you put a "stack frame" on top of the stack where all the local variables live in, and when the function is done, the stack frame will be "teared down".

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

When you create a new function that returns a string value, the Script Manager places the following function beginning line into your script file:

String Function MyFunction ()

The "string" key word that precedes the "function" key word tells you that the MyFunction function returns a string value to the calling script or user-defined function.

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

If there is no return statement for a function, its return value is None.

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

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

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

The data type of None is NoneType.

1. **What does the sentence import are all your pets namederic do?**

That import statement imports a module named areallyourpetsnamederic. (This isn’t a real Python module, by the way.)

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

This function can be called with spam.bacon().

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

Place the line of code that might cause an error in a try clause.

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

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.