1. ***Why are functions advantageous to have in your programs?***
   Ans :-
   - With the help of functions, we can avoid rewriting the same logic or code again and again in a program.
   - In a single Program, we can call Python functions anywhere and also call multiple times.
   - We can easily track a large Python program when divided into multiple functions.

2. ***When does the code in a function run: when it's specified or when it's called?***
   Ans :- The code inside a function runs when the function is called.

3. ***What statement creates a function?***
   Ans :-

- Function is created with the **def** keyword, then write the function identifier (name) followed by parentheses and a colon.
  def functionName():

- The next thing is to specify what the function does.

```
def hello():
        print("Hello World")
```

- To call this function **,** write the name of the function followed by parentheses: hello()

4. ***What is the difference between a function and a function call?***
   Ans:- A function is a piece of code that enhance the reusability of your program. It means that piece of code need not be written again.
   A function call means invoking or calling that function. Unless a function is called there is no use of that function.

5. ***How many global scopes are there in a Python program? How many local scopes?***
   Ans :- In a Python program, there is only one global scope. It is accessible throughout the entire program.
   The number of local scopes in a Python program can vary depending on the number of function or method invocations. Each time a function or method is called, a new local scope is created.

6. ***What happens to variables in a local scope when the function call returns?***
   Ans:- When a function call returns, the local scope of that function, including its variables, is destroyed. Variables declared within the function's local scope are no longer available and their values no longer exist.

7. ***What is the concept of a return value? Is it possible to have a return value in an expression?***
   Ans :- A return value is the value that a function "returns" after it has completed its execution. When a function is called, it may perform some operations and then return a value back to the caller. This value can then be used by the caller for further processing or display.
   Yes, it is possible to have a return value in an expression.

8. ***If a function does not have a return statement, what is the return value of a call to that function?***
   Ans :- If a function does not have a return statement, the return value of a call to that function is None in Python.

```
def neuron():
   print("Hello, world!")

result =neuron()
print(result)
```

   Output:

```
Hello, Coders!
None
```

   In the example above, the neuron () function does not have a return statement. When we call neuron (), it prints "Hello, Coders!" but doesn't explicitly return anything. As a result, the value of the result is None.

9. ***How do you make a function variable refer to the global variable?***
   Ans :- To make a function variable refer to a global variable, you can use the global keyword.

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

Ans:- The data type of None is NoneType. It is a special type that represents the absence of a value or the lack of a specific object. None is often used to indicate that a variable or expression does not have a value assigned to it.

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

Ans:- It gives error. There is no such module found.

12. ***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().

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

Ans:- 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?***

Ans :- Try blocks let you test for errors within a block of code.

Using the except block, you are able to handle errors.