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

They make the code more readable and reusable. They make a lot of sense especially when we want to repeat the same function over and over again within our project.

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

The code in a function runs by the time it is called.

3. What statement creates a function?

def name_function(argument):

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

The function contains :
the keyword 'def', the name of the function, the arguments, the body of the function (the instruction that will be executed when it'll be called) and one or more objects that the function will return (or nothing).

The function call contains only the name and the arguments and can be executed multiple times.

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

It exists only one global scope in Python while as many local scopes as the number of functions created.

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

When the function terminates its execution, the local variables are initialized to their original value.

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

The return is used to store only the result, or the most useful information within a function.

No, it is used only by a function.

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

NoneType

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

By using the keyword global.

10. What is the data type of None?

The data type of None is Nonetype and it means that no value is returned to that variable or function.

11. What does the sentence import areallyourpetsnamederic do?

It imports the library called areallyourpetsnamederic and all the methods that it contains.

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

To call bacon function I would use the following syntax: spam.bacon()

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

I would put the code within the try structure and use except like this:

```
try:
        code

except:
        pass
```

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

The try structure allows me to test a block of code.
The except block enables you to handle the error.