**PYTHON BASIC ASSIGNMENT 3**

**Question1**. Why are functions advantageous to have in your programs?

**Answer:**  functions are used to avoid the rewriting the same code again and again in our program.

functions can be called any number of times from any place of our program.

When a program is divided into functions, then any part of our program can easily be tracked.

**Question2**. When does the code in a function run: when it's specified or when it's called?

**Answer:** When a function is "called" the program "leaves" the current section of code and begins to execute the first line inside the function

**Question3**. What statement creates a function?

**Answer:** 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. You can further re-assign the same function object to other names.

**Question4**. What is the difference between a function and a function call?

**Answer:** Using a function to do a particular task any point in program is called as function call.

A function is procedure to achieve a particular result while function call is using this function to achive that task.

**Question5**. How many global scopes are there in a Python program? How many local scopes?

**Answer:** There's only one global Python scope per program.

There are four local scopes in python local scope, enclosed scope, global scope, and built-in scope.

**Question6**. What happens to variables in a local scope when the function call returns?

**Answer:** When the execution of the function terminates (returns), the local variables are destroyed. Codelens helps you visualize this because the local variables disappear after the function returns.

**Question7**. What is the concept of a return value? Is it possible to have a return value in an expression?

**Answer:** The Python [return statement](https://en.wikipedia.org/wiki/Return_statement) is a key component of [functions](https://realpython.com/defining-your-own-python-function/) and [methods](https://realpython.com/python3-object-oriented-programming/#instance-methods). You can use the return statement to make your functions send Python objects back to the caller code. These objects are known as the function’s **return value**. You can use them to perform further computation in your programs.

**Question8**. If a function does not have a return statement, what is the return value of a call to that function?

**Answer:** If no return statement appears in a function definition, control automatically **returns** to the calling function after the last statement of the called function is executed. In this case, the return value of the called function is undefined. ... If a return value isn't required, declare the function to have void return type

**Question9**. How do you make a function variable refer to the global variable?

**Answer:** The global Keyword

Normally, when you create a variable inside a function, that variable is local, and can only be used inside that function. To create a global variable inside a function, you can use the global keyword.

**Question10**. What is the data type of None?

**Answer:** None is used to define a null value. It is not the same as an empty string, False, or a zero. It is a data type of the class NoneType object.

**Question11**. What does the sentence import areallyourpetsnamederic do?

**Answer:** A global statement will force a variable in a function to refer to the global variable. The data type of None is NoneType . That import statement imports a module named areallyourpetsnamederic

**Question12**. If you had a bacon() feature in a spam module, what would you call it after importing spam?

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

**Question13**. What can you do to save a programme from crashing if it encounters an error?

**Answer:** Exceptions are thrown and caught so the code can recover and handle the situation and not enter an error state. Exceptions can be thrown and caught so the application can recover or continue gracefully

**Question14**. What is the purpose of the try clause? What is the purpose of the except clause?

**Answer:** Try and Except statement is used to handle these errors within our code in Python. The try block is used to check some code for errors i.e the code inside the try block will execute when there is no error in the program. Whereas the code inside the except block will execute whenever the program encounters some error in the preceding try block.