PYTHON ASSIGNMENT-3

QUESTIONS-

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

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

3. What statement creates a function?

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

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

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

7. What is the concept of a return value? Is it 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?

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

10. What is the data type of None?

11. What does the sentence import areallyourpetsnamederic do?

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

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

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

SOLUTIONS:

1. **Advantages of Functions:**
   - Reusability: Functions allow you to reuse code by encapsulating it into a named block.
   - Modularity: Functions promote modularity, making code more organized and easier to understand.
   - Maintainability: Changes can be made to a function without affecting the rest of the code.
   - Abstraction: Functions hide the implementation details and provide an interface for interaction.
2. **When Does Code in a Function Run:**
   - The code in a function runs when the function is called or invoked.

3.Statement Creating a Function:

- The **def** statement is used to create a function. Example:

```
def my_function():
    # function code
```

4.Difference Between a Function and a Function Call:

- A function is a block of code defined with a name.
- A function call is the act of invoking or executing the function to perform its defined tasks.

5.Global and Local Scopes:

- There is one global scope in a Python program.
- Local scopes are created when a function is called. Each function call creates its own local scope.

6.Variables in Local Scope After Function Call Returns:

- Local variables cease to exist after the function call returns. They are only accessible within the function.

7.Return Value Concept:

- A return value is the value that a function provides as output.
- Yes, it is possible to have a return value in an expression, and it can be used in assignments or other expressions.

8.Return Value Without Return Statement:

- If a function does not have a return statement, the return value of a call to that function is **None**.

9.Make Function Variable Refer to Global Variable:

- To make a function variable refer to the global variable, use the **global** keyword. Example:
- def my_function():
- global global_variable
- global_variable = 10

10.Data Type of None:

- The data type of **None** is **NoneType**.

11.Import Statement import areallyourpetsnamederic:

- This is not a standard library or module in Python. It would raise an ImportError.

12.Calling Feature from Imported Module:

```
import spam
spam.bacon()
```

13.Handling Errors to Prevent Crashing:

- Use try and except blocks to catch and handle exceptions.
- try:
- # code that might cause an error
- except SomeException:
- # code to handle the exception

14.Purpose of Try and Except Clauses:

- The **try** clause contains the code that might raise an exception.
- The **except** clause contains the code to handle the exception if it occurs, preventing the program from crashing.