**Assignment-3**

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

**Ans:-** Functions are the block of code which is callable or reusable within project code. It is advantageous when we have to perform specific task multiple times within our project. It helps to reduce repetitive work.

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

**Ans:-** The code within a function in most programming languages runs when the function is called, not when it is specified or defined.

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

**Ans:-** By using “def” at initial of code we can create a function.

e.g. = *def function1():*

in above example we create function named *function1* by using def statement.

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

**Ans:-** Functions and Functions call are relatable concepts but they have different meanings.

Functions= Functions are the block of code which is callable or reusable within project code. It is advantageous when we have to perform specific task multiple times within our project. It helps to reduce repetitive work. Functions can take inputs, perform operations on them and return a result.

We can define a function with *def fun\_1():*

Function Call= A function call is the act of executing the function. By using this statement, we can trigger the block of code within a function. When we make a function call, we have to specify the function's name followed by parentheses.

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

**Ans:-** In Python there is only one global scope but there can be multiple local scopes.

* Global scope:- The global scope is the outermost scope in the Python program.

It exists outside of the function or class. Variable defined in global scope are accessible throughout the program, including functions and classes.

e.g. =

A screenshot of a computer screen

Description automatically generated

* Local Scope:- A local scope is created whenever function is called or entering a code block like loop or conditional statements e.g For loop, if else statement etc. Variable defined in local scope are only accessible within that specific scope. Once the scope is exit, the variable defined in that scope will get expired.

e.g.=

A screen shot of a computer error

Description automatically generated

In above example y is defined within a function and it is accessible only in function not outside of the function.

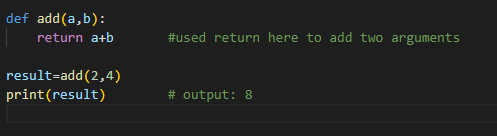
1. **What happens to variables in a local scope when the function call returns?**

**Ans:-** The variable defined within a function is only accessible for that specific function, this is called local scope. When we call function that variable used in that function and returns a result and memory got freed from that variable. We can use that variable in global scope also.

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

**Ans:-** The concept of a return value refers to the value that a function can send back or "return" to the code that called it. When a function is executed and reaches a return statement, it immediately exits the function and provides a value as the result of the function call.

**e.**g.=

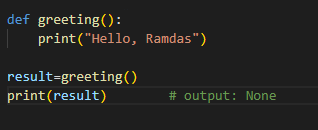


In above example we used return statement to get the addition of two values within a expression.

1. **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, or if it reaches the end of the function block without encountering a return statement, the function call will still produce a return value. In Python, the default return value in such cases is None.

**e.**g.:-



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

**Ans:-** To make function variable refer to the global variable we need to declare the variable within a function by using “global” keyword.

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

**Ans:-** In Python, the data type of ‘None’ is ‘NoneType’. ‘NoneType’ is a built-in type in Python that represents the absence of a value or the null value.

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

**Ans:-** The importstatement is used to access the modules or specific objects from module.

We can access the functionality of a module by using import statement.

In above sentence we can import **areallyourpetsnamederic** module and makes it accessible using the module name. To use functions, classes, or other objects from the module, we would need to prefix them with the module name followed by a dot (.).

Like **areallyourpetsnamederic.Function\_1()**

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

**Ans:-** We can call bacon() feature from spam module after importing spam module by using (.) after module name, as follows.

**Spam.**bacon()

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

**Ans:-** We can use exception handling to save program from crashing if it encounters an error. There are multiple types of error in Python, and all of them having certain exception handling methods.

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

**Ans:-** The try and except clauses are used together in exception handling in Python. They serve the following purposes:

* **try:** The try clause is used to enclose a block of code that might raise an exception. It allows you to specify a section of code where exceptions can occur. The purpose of the try clause is to attempt the execution of this code block and monitor for any raised exceptions.

If the exception occurs within a try block the execution of code within try block get halted and transfer to the “except” clause.

* **except:** The “except” clause is used to define how specific exceptions should be handled when they occur within the corresponding “try” block. It allows you to specify the code that should be executed when a particular exception is raised.