**10th June**

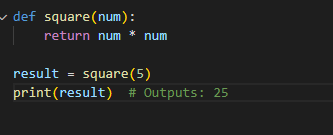
**Assignment - 8**

1. **In Python, what is the difference between a built-in function and a user-defined function? Provide an example of each.**

**Ans:** Built in functions: These ae the functions that provided as a part of the python language itself, and it is available for immediate use without writing any additional code. These functions cover a wide range of common tasks.

**Example:** “print()”, “len()”, range(), min(), max(), upend() etc.

**User defined functions:** These functions are created by the user to perform a specific task or set of tasks. User defined functions are created using “def” keyword followed by function name and parameters if any.

**Example:** 

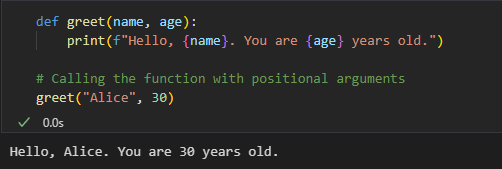
1. **How can you pass arguments to a function in Python? Explain the difference between positional arguments and keyword arguments.**

**Ans:** In python we can pass arguments to function by Positional argument and Keyword arguments.

* **Positional arguments:** Positional arguments are passed to a function in the order they are defined in the function’s parameter list.

When we call function and provide arguments without specifying the parameter names, then Python assigns the arguments to the corresponding parameters based on their position. This means first argument assigned to the first parameter and second argument to the second parameter.

Example:



* Keyword Argument: These arguments passed to the function by mentioning the parameter names during the function call. This way the order of arguments doesn’t matter.

Example:

A screenshot of a computer program

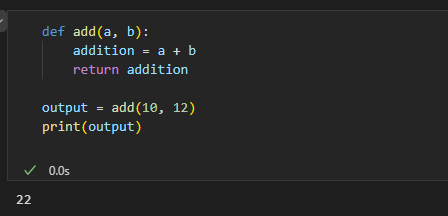
Description automatically generated

1. **What is the purpose of the return statement in a function? Can a function have multiple return statements? Explain with an example.**

**Ans:** When a function is called, it executes its code and performs some operations. The result of those operations can be returned to the caller using the “return” statement.

Yes, a function can have multiple “return” statements. However, when the function encounters a “return” statement, it immediately exits the function, and any subsequent code in the function is not executed. So, only one “return” statement will be executed during the function's call.

Example:



Below example is for multiple return statements.

A screen shot of a computer program

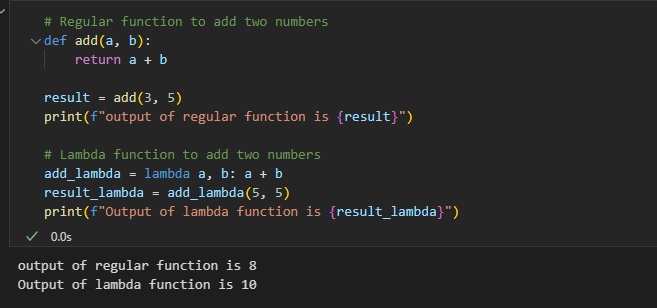
Description automatically generated

1. **What are lambda functions in Python? How are they different from regular functions? Provide an example where a lambda function can be useful.**

**Ans:** In Python, a lambda function is a small anonymous function. It is called "anonymous" because it doesn't have a name like a regular function defined with the def keyword. Instead, lambda functions are defined using the lambda keyword, followed by the function's parameters and a single expression. The result of the expression is automatically returned when the lambda function is called.

Difference between lambda functions and regular functions:

* Syntax: Lambda functions having more compact syntax compared to the regular function defined by ‘def’.
* Function name: Lambda functions do not have any name, while regular functions have a name which can be used to call the function.
* Number of expressions: Lambda functions are limited to the single line functions while regular functions does not having limitations of expressions.
* Example using Lambda function:



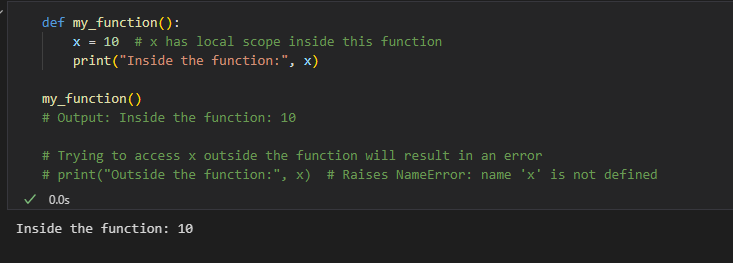
1. **How does the concept of "scope" apply to functions in Python? Explain the difference between local scope and global scope.**

**Ans:** In Python “scope” refers to the region of program where a particular variable can be accessible. The Local and Global scopes are explained below.

* **Local Scope:** Local scope refers to the area within a function where variable is defined. In other words, any variable declared inside the function is considered to have Local scope.

These variables only can be accessed within a function. Once the execution completes, the local variable is destroyed, and their memory is released.

Example:



* **Global Scope:** Global scope refers to the outermost level of the program, outside any function or block.

Variables defined at this level have global scope and can be accessed from anywhere in the code, including within functions.

Example:

A computer screen with text and numbers

Description automatically generated

* **Difference between Local scope and Global scope:**

When a variable is accessed within a function, Python first looks for that variable's value in the local scope. If it finds the variable in the local scope, it uses that value. If the variable is not present in the local scope, Python then searches for it in the global scope.

1. **How can you use the "return" statement in a Python function to return multiple values?**

**Ans:** In Python, we can use the “return” statement in a function to return multiple values as a single object. To achieve this, we can return a tuple, list, or any other data structure that can hold multiple values.

Example to use “return” statement to give multiple values.

A computer screen shot of a program code

Description automatically generated

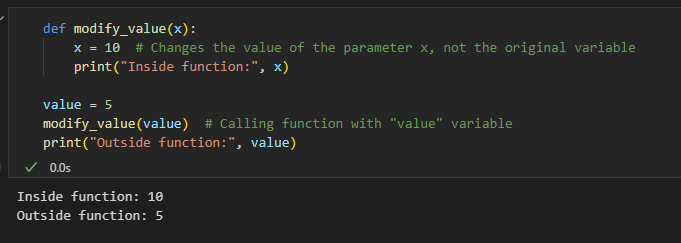
1. **What is the difference between the "pass by value" and "pass by reference" concepts when it comes to function arguments in Python?**

**Ans:** In Python, the concepts of "pass by value" and "pass by reference" refer to how function arguments are handled during function calls.

* **Pass by value:**

In "pass by value," when we pass a variable to a function as an argument, a copy of the variable's value is created, and this copy is used within the function. Any modifications made to the parameter (variable) within the function do not affect the original variable in the calling code.

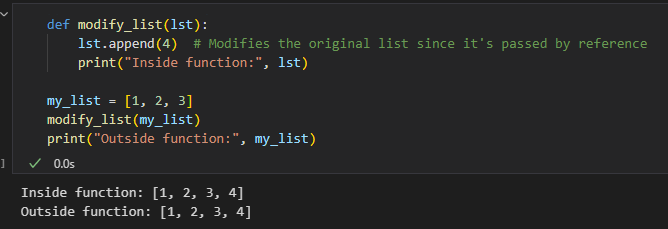
Example:



* **Pass by reference:**

In "pass by reference," when we pass a variable to a function as an argument, we are passing a reference to the memory location where the variable is stored. Any changes made to the parameter (variable) within the function will directly affect the original variable in the calling code.

Example:



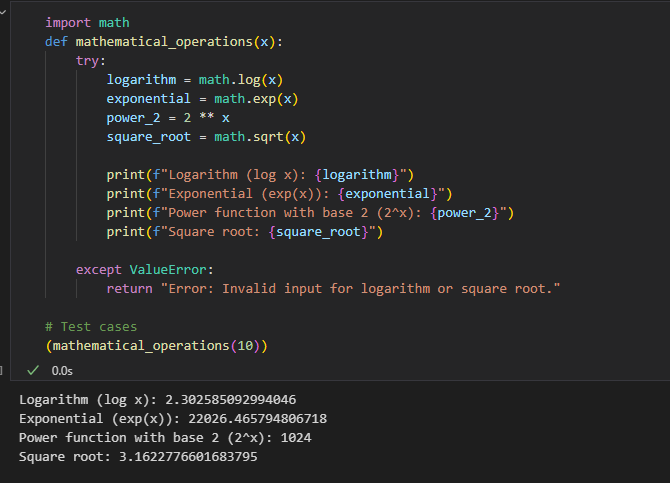
**8. Create a function that can intake integer or decimal value and do following operations:**

**a. Logarithmic function (log x)**

**b. Exponential function (exp(x))**

**c. Power function with base 2 (2x)**

**d. Square root**

**Ans: **

1. **Create a function that takes a full name as an argument and returns first name and last name.**

**Ans:**

**A computer screen shot of a code

Description automatically generated**