Module 3 Summary: Python Programming Fundamentals

Congratulations! You have completed this module. At this point, you know that:

- Python conditions use "if" statements to execute code based on true/false conditions created by comparisons and Boolean expressions.

- Comparison operations require using comparison operators such as == (equal to), > (greater than), and < (less than).

- Python uses the "!=" operator to determine whether two values are not equal.

- You can compare integers, strings, and floats.

- Python branching directs program flow by using conditional statements (for example, if, else, elif) to execute different code blocks based on conditions or tests.

- You can use the "if" statement with conditions to define actions if true.

- To perform actions when all previous conditions are false, you can use the "else" statement without a condition.

- The elif statement allows for additional checks only if the initial condition is false.

- To execute various operations on Boolean values, we use Boolean logic operators.

- Python loops are control structures that automate repetitive tasks and iterate over data structures like lists or dictionaries.

- The range() function generates a sequence of numbers with a specified start, stop, and step value for loops in Python.

- A for loop in Python iterates over a sequence, such as a list, tuple, or string, and executes a block of code for each item in the sequence.

- A while loop in Python executes a block of code as long as a specified condition remains true.

- Python functions are reusable code blocks that perform specific tasks, take input parameters, and often return results, enhancing code modularity and reusability.

- You may or may not have written the codes that are often included in functions.

- Python has a set of built-in functions such as "len" to find the length of a sequence or "sum" to find the total sum of a sequence.

- The "sorted" function creates a new sorted list, while "sort" sorts items in the original list.

- You can also create your own functions in Python.

- To ensure clarity and organization and facilitate understanding and maintenance of the code, developers must document functions using a documentation string enclosed in three quotes.

- The help command will return the documentation defined for a particular function.

- A function can have multiple parameters.

- If a function does not include a return statement, it returns None by default.

- You can use the "pass" keyword in a function to indicate that it does nothing (a placeholder for future code).

- A function will usually perform more than one task.

- In Python, the scope of a variable determines where you can access or modify that variable. Global scope allows access from anywhere, while local scope restricts it to a block or function.

- In Python, a programmer defines a local variable within a specific block or function, which can only be accessed or modified within that block or function.

- In Python, a global variable is a variable defined at the top level of a program that any part of the code can access or modify.

- Exception handling in Python is a mechanism for managing and responding to errors and exceptions that may occur during program execution, preventing them from crashing the program.

- In Python, you use the "try-except" statement to attempt a block of code and specify alternative actions to execute if an error occurs, allowing you to handle exceptions.

- In Python, you use the "try-except-else" statement to attempt a block of code, handle exceptions in the "except" block, and execute code in the "else" block when no exceptions occur.

- Python developers use the "try-except-else-finally" statement to attempt a block of code, catch exceptions in the "except" block, execute code in the "else" block when no exceptions occur, and ensure that the "finally" block always runs, regardless of whether exception was raised or not.

- In Python, objects are instances of classes that encapsulate data and behavior, serving as the foundation for creating and working with various data types and custom data structures.

- To determine the type of an object in Python, you can use the `type()` command.

- Methods may modify an object's internal state, but the object's type usually remains the same.

- Classes in Python are blueprints for creating objects, defining their attributes and methods, enabling code organization, and object-oriented programming.

- Function "init" is a special method used to initialize data attributes.

- We can create instances of a class in Python.

- Data attributes consist of the data defining the objects.

- Methods are functions that interact and change the data attributes.

- The method has a function that requires the self as well as other parameters.

✓