coursera

Menu

Module 2 Summary: Python Data Structures

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

- In Python, we often use tuples to group related data together.Tuples refer to ordered and immutable collections of elements.
- Tuples are usually written as comma-separated elements in parentheses “()".
- You can include strings, integers, and floats in tuples and access them using both positive and negative indices.
- You can perform operations such as combining, concatenating, and slicing on tuples.
- Tuples are immutable, so you need to create a new tuple to manipulate it.
- Tuples, termed nesting, can include other tuples of complex data types.
- You can access elements in a nested tuple through indexing.
- Lists in Python contain ordered collections of items that can hold elements of different types and are mutable, allowing for versatile data storage and manipulation.
- A list is an ordered sequence, represented with square brackets "[]".
- Lists possess mutability, rendering them akin to tuples.
- A list can contain strings, integers, and floats; you can nest lists within it.
- You can access each element in a list using both positive and negative indexing.
- Concatenating or appending a list will result in the modification of the same list.
- You can perform operations such as adding, deleting, splitting, and so forth on a list.
- You can separate elements in a list using delimiters.
- Aliasing occurs when multiple names refer to the same object.
- You can also clone a list to create another list.
- Dictionaries in Python are key-value pairs that provide a flexible way to store and retrieve data