Python

Learn clean and efficient Python programs using data structure

About the Author

- Created By: Mohammad Salman
- Experience: 19 Years +
- Designation: Corporate Trainer .NET

Microsoft CERTIFIED Professional

Microsoft Specialist

Microsoft CERTIFIED Solutions Developer

Icons Used

Questions

Tools

Hands-on Exercise

Coding Standards

Questions?

Reference

Try it Out

Informative Slide

Mandatory Slide

Welcome Break

DATA STRUCTURE IN PYTHON

- Learn clean and efficient Python programs using the right data structure
- By: Mohd Salman

PYTHON DATA STRUCTURES

- Lists, Tuples, Sets, Dictionaries, Comprehension
- By: Mohd Salman

PYTHON DATA STRUCTURES

- Understand the purpose and characteristics of Lists, Tuples, Sets, and Dictionaries
- dentify key differences among these data structures
- Apply appropriate data structures for different problem scenarios
- Perform basic operations such as adding, updating, deleting, and accessing elements
- Use dictionary functions (get(), keys(), values(), items()) effectively
- Write clean and efficient Python programs using the right data structure

LIST - DESCRIPTION & SYNTAX

- A list is an ordered, mutable collection of items.
- Allows duplicates and mixed data types.
- Syntax:
- my_list = [10, 20, 30, "Python"]

LIST - EXERCISES

- Beginner: Create list of numbers, print 2nd and last element.
- Intermediate: Add, remove, and sort elements.
- Advanced: New list with elements > 50 using list comprehension.

TUPLE - DESCRIPTION & SYNTAX

- A tuple is ordered and immutable.
- Once created, values cannot be changed.
- Syntax:
- my_tuple = (1, 2, 3, 4)

TUPLE - EXERCISES

- Beginner: Print subjects from tuple.
- Intermediate: Find index of a given value.
- Advanced: Count frequency of each number in tuple.

SETS - WILL NOT PRINT THE REPEATED DATA

- A set is unordered and contains unique elements.
- Syntax:
- my_set = {1, 2, 3}

SET - DESCRIPTION & SYNTAX

SET - EXERCISES

- Beginner: Create set with duplicates and print it.
- Intermediate: Find union and intersection of two sets.
- Advanced: Print elements in set1 not in set2.

DICTIONARY - DESCRIPTION & SYNTAX

- Dictionary stores key-value pairs. Keys are unique.
- Syntax:
- my_dict = {"name": "John", "age": 25}

DICTIONARY - EXERCISES

- Beginner: Create dictionary and print key-value pairs.
- Intermediate: Add new key and delete another.
- Advanced: Find student with highest marks from dictionary.

DICTIONARY FUNCTIONS

- Common functions:
- .get(key) - Returns value
- .keys() - All keys
- .values() - All values
- .items() - Key-value pairs

DICTIONARY FUNCTION EXERCISES

- Beginner: Print all keys.
- Intermediate: Safely access missing key using get().
- Advanced: Print employees with salary > 50,000 using comprehension.

COMPARISON

Feature	List	Tuple	Set	Dictionary
Syntax	`[]`	`()`	`{}`	`{key: value}`
Ordered	Yes	Yes	NO	Yes (since Python 3.7)
Mutable	Yes	NO	Yes	Yes
Indexed	Yes	Yes	NO	Yes (by key)
Duplicates	Yes	Yes	NO	NO (keys unique)
Use Case	General-purpose ordered data	Fixed data that shouldn't change	Unique items collection	Key-value mapping

mutable-can change the print

intersection, union etx

- Data structures are organized ways to store, manage, and access data efficiently in a program

THANK YOU!