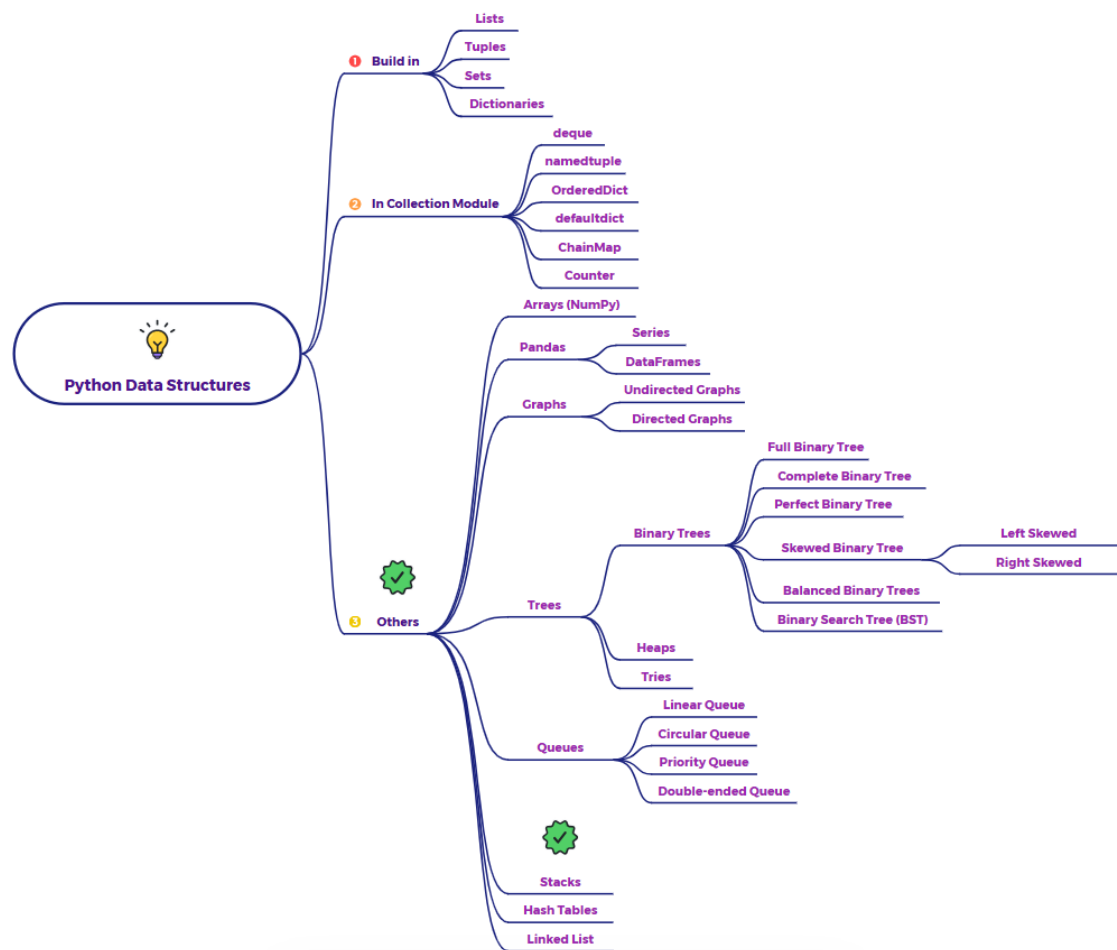


# Explain Stacks as a data structure in python



Imagine a stack of plates. You can only add a new plate to the top, and when you want to take a plate, you can only take the topmost one. The last plate you put on the stack is the first one you take off. This "last-in, first-out" (LIFO) principle is the core idea behind a **Stack** data structure.

## What is a Stack?

A **Stack** is a linear data structure that follows the **Last-In, First-Out (LIFO)** principle. This means that the most recently added element is the first one to be removed. Think of it like a stack of items where you can only access the top.

## Key Operations of a Stack:

- **Push:** Adding a new element to the **top** of the stack.
- **Pop:** Removing and returning the element from the **top** of the stack.

- **Peek (or top):** Viewing the element at the top of the stack without removing it.
- **isEmpty:** Checking if the stack is empty.
- **size:** Getting the number of elements in the stack.

### Why Use Stacks?

Stacks are fundamental in many areas of computer science and have various applications, including:

- **Function Call Stack:** Used by programming languages to manage function calls and returns. When a function is called, its information is pushed onto the stack, and when it returns, that information is popped off.
- **Undo/Redo Functionality:** Many applications use stacks to implement undo and redo operations. Each action is pushed onto the stack, and undoing pops the last action.
- **Expression Evaluation:** Stacks are used in algorithms for evaluating arithmetic expressions (e.g., converting infix to postfix notation).
- **Backtracking Algorithms:** Algorithms that explore possibilities step by step often use stacks to keep track of the path taken and backtrack when a dead end is reached.
- **Browser History:** The back button in a web browser often uses a stack to keep track of visited pages.
- **Depth-First Search (DFS):** A graph traversal algorithm that explores as far as possible along each branch before backtracking.

In summary, a Stack is a linear data structure that follows the LIFO principle. In Python, the built-in list provides an efficient and straightforward way to implement a stack using its `append()` for push and `pop()` for pop operations.