

[Composite Design Pattern]

The **Composite Pattern** is a **structural design pattern** that allows you to treat individual objects and compositions of objects uniformly. It is often used to represent part-whole hierarchies, such as a tree structure.

Key Concepts:

- **Component:** An abstract class or interface that defines common operations for both simple and composite objects.
 - **Leaf:** Represents individual objects in the composition that do not have children.
 - **Composite:** Represents a container of objects (including other composites and leaves) and implements operations for managing children.
-

Real-World Analogy:

Think of a company's organizational chart:

- Employees (Leaf) report to Managers (Composite).
 - Managers, in turn, may report to Directors (Composite), forming a hierarchy.
-

Example: File System Structure

We will model a file system where:

- **Files** are leaves.
- **Folders** are composites containing files or other folders.

