Pillars of Python

Python follows the **four pillars of Object-Oriented Programming (OOP)**, which are fundamental to structuring and designing robust, scalable applications.

- Encapsulation
- Abstraction
- Inheritance
- Polymorphism

Encapsulation is the bundling of data (attributes) and methods (functions) into a single unit (class). It restricts direct access to variables and allows controlled modification through getter and setter methods.

Abstraction hides implementation details and exposes only the necessary functionality to the user. It allows working with high-level concepts without worrying about underlying complexity.

Inheritance allows one class (child) to derive attributes and methods from another class (parent). It promotes code reuse and establishes a relationship between classes.

Polymorphism allows methods or functions to have multiple implementations depending on the object or context. This enables flexibility and dynamic behaviour in programs.

Pillar	Purpose
Encapsulation	Hides data, allows controlled access
Abstraction	Hides complexity, shows only necessary details
Inheritance	Enables reusability by deriving new classes from existing ones
Polymorphism	Provides flexibility by using a single interface for different implementations