**The four core principles of Object-Oriented Programming (OOP)**

are Encapsulation, Abstraction, Inheritance, and Polymorphism. These principles promote code organization, reusability, and maintainability.

**1. Encapsulation:**

Encapsulation involves bundling data (properties) and methods that operate on that data within a single unit, typically a class. It also restricts direct access to some of an object's components, preventing external code from directly manipulating internal state.



**2. Abstraction:**

Abstraction focuses on showing only essential information and hiding complex implementation details. It allows users to interact with objects at a high level without needing to understand the underlying mechanisms.



**3. Inheritance:**

Inheritance allows a new class (subclass/child class) to inherit properties and methods from an existing class (superclass/parent class). This promotes code reuse and establishes a hierarchical relationship between classes.



**4. Polymorphism:**

Polymorphism means "many forms." In OOP, it allows objects of different classes to be treated as objects of a common type, and for methods with the same name to behave differently based on the object's type.

