Polymorphism & Encapsulation

Polymorphism:

- **Meaning:** "Poly" = many, "Morph" = forms → One thing with multiple forms.
- In Python, polymorphism allows the **same method name** to behave differently depending on the object.

- **Built-in Polymorphism**
- **Polymorphism with Methods**
- **Method Overriding (Polymorphism in OOP)**

Encapsulation:

- **Encapsulation = Binding data (variables) and methods (functions) into one unit (class).**
- It hides implementation details from the user and only exposes necessary functionality.

Access Modifiers in Python:

- **Public** → Accessible everywhere.
- **Protected (_var)** → Indicate it should not be accessed outside the class (just a convention).
- **Private (__var)** → Strictly hidden, cannot be accessed directly outside class.

Key Points:

- **Polymorphism** → One interface, multiple implementations.
- **Encapsulation** → Data hiding + security by restricting direct access.
- Together, they make code flexible, secure, and reusable.