Validation & Polymorphism – Explanation for the OOP Library Project

1. Validation

Validation means checking that data is correct before performing an operation. In the Library project, validation ensures safe, correct, and reliable behavior.

Validation in Borrowing:

- Checks if the book exists.

- Checks if the book is already borrowed.

Invalid operations return false.

Validation in Returning:

- Checks if the book exists.

- Checks if the book is marked as borrowed.

This prevents returning books that were never borrowed.

Why Validation Matters:

- Prevents invalid operations.

- Maintains correct system state.

- Protects data integrity.

2. Polymorphism

Polymorphism allows classes to share a common method name but implement different behaviors.

Example: Different Book Types

Different book types can override a method like GetBorrowDuration(), giving each type its own borrowing rules.

Example Classes:

- Novel: 14 days

- Magazine: 3 days

- ReferenceBook: not borrowable

Calling book.GetBorrowDuration() results in different behavior depending on the object type.

Why Polymorphism Matters:

- Makes the system scalable.

- Allows different objects to behave uniquely.

- Adds flexibility to the OOP design.