Java OOP House Project – Step-by-Step Study Summary

Step 1 – Created a Class (Blueprint)

• Defined BuildHouse class.
• Added fields (rooms, color, hasCatDoor).
• Fields describe what a house can have.

Step 2 – Constructor (Build Day)

• Created a constructor to assign values.
• Used 'this.' to store values in the object.
• Constructor builds the actual house object.

Step 3 – Methods (House Behavior)

• Added describe() method to print house details.
• Added repaint() to modify object state.
• Methods make the house DO things.

Step 4 – Separate Demo File

• Moved main() into DemoBuildHouse.java.
• Demonstrated separation of logic and execution.

Step 5 – Encapsulation

• Made fields private.
• Kept constructor and methods public.
• Protected internal state of object.

Step 6 – Inheritance

• Created FancyHouse extends BuildHouse.
• Used super() to call parent constructor.
• Added subclass-specific behavior (hot tub).

Step 7 – Method Overriding & Polymorphism

- Overrode describe() in FancyHouse.
- Used parent reference with child object.
- Java selects method based on object type at runtime.

Step 8 – Abstract Class (Foreman)

- Created AbstractHouse (abstract class).
- Added shared structure (roofType).
- Added abstract method buildFoundation() (IOU).
- Subclasses implemented required method.

Core Concepts Demonstrated

- Class vs Object
- Fields vs Methods
- Constructor behavior
- Encapsulation
- Inheritance
- Overriding
- Polymorphism
- Abstract classes and abstract methods