Step-by-Step Plan to Start Coding the Nursery System

Step 1: Create the Main Class (Base Classes and Interface)

Layer	Classes/Interfaces	Purpose
Mediator layer	Mediator, InteractionMediator, InventoryMediator, Colleague	Set up communication flow
Command layer	Command, PurchasePlantCommand, WaterPlantCommand, AddPotCommand, UpdateStockCommand, RemovePlantCommand	Define action contracts
Decorator layer	Plant, PotDecorator, GiftWrapDecorator, SpecialArrangement	Enable plant customization
Strategy layer	InteractionStrategy, AdviceStrategy, AvailabilityStrategy, AccessoriesStrategy DiscountStrategy, BulkDiscount, SeasonalDiscount, FamilyDiscount, WitheringDiscount	Handle different staff interaction styles
Concrete Colleagues	Customer, Staff, GreenHouseStaff, SalesStaff	Your main actors
Utility/Domain	Inventory, ShoppingCart, PlantType,	Core domain logic

Step 2: Implement the Mediator Core

Start with **Mediator** because it's your system's backbone — all communication routes through it.

Files to focus on first:

- Mediator (interface)
- InteractionMediator
- InventoryMediator
- Colleague
- Customer
- Staff, GreenHouseStaff, SalesStaff

Test early:
Simulate a conversation between a Customer and a GreenHouseStaff through the InteractionMediator.

Step 3: Add Command Layer

Once communication works, add **Command** objects to encapsulate actions.

Focus on these first:

- Command
- PurchasePlantCommand
- WaterPlantCommand
- AddPotCommand
- UpdateStockCommand
- RemovePlantCommand

Test:
Let the InteractionMediator or Staff create and execute commands on Plant or Inventory.
You can print: “Executing: PurchasePlantCommand on Fern”

Step 4: Add the Decorator Layer (Plant Customisation)

Add your plant hierarchy:

- PlantType (abstract)
- PotDecorator, GiftWrapDecorator, SpecialArrangement

Test:
Plant fern = new BasicPlant("Fern");

Plant wrappedFern = new WrapDecorator(new PotDecorator(fern));

System.out.println(wrappedFern.getDescription());

Step 5: Implement the Factories

Now create plant families using the **Abstract Factory**.

- PlantFactory
- TreeFactory
- FlowerFactory

Test:

- AbstractPlantFactory factory = new FlowerFactory();

Plant aloe = factory.createPlant("Aloe Vera");

Step 6: Add Strategies for Staff Interactions

- InteractionStrategy (interface)
- Concrete: AdviceStrategy, AvailabilityStrategy, AccessoriesStrategy

Test:
Staff sales = new SalesStaff(new AvailabilityStrategy());

sales.interactWithCustomer();

Step 7: Add the Chain of Responsibility

Step 9: Build Scenarios (End-to-End Tests)

Examples:

- **Scenario 1:** Customer buys a plant → Mediator routes → Command executes → Inventory updates.
- **Scenario 2:** Customer asks for watering advice → Mediator routes → CoR handles → Strategy decides tone or detail.
- **Scenario 3:** Customer customizes plant → Decorator applied → PurchaseCommand executes.

WORKFLOW

Scenario: Customer Purchases a Plant

A customer buys one or more plants, optionally with decorations (e.g., pot, gift wrap).
The system updates the inventory and confirms the sale.

Step by Step:

1. Customer initiates Purchase
    - The customer selects a plant or bundle and chooses “Buy”
    - This request goes to the InteractionMediator, which manages all customer–staff communication.
2. Mediator Delegates to Staff
    - The InteractionMediator finds the appropriate Staff member (via Chain of Responsibility if needed).
    - The SalesStaff receives the purchase request.
3. Staff Creates and Executes a Command
    - It creates a PurchasePlantCommand, passing references to:
        - the PlantType (Receiver)
        - the InventoryMediator (Receiver)
        - the ShoppingCart
4. Command Executes the Purchase
    - The PurchasePlantCommand.execute() method:
        - Deducts the purchased plant from InventoryMediator.
        - Records the transaction in ShoppingCart
        - Notifies Observers
5. Observer Updates Inventory Automatically
    - The InventoryMediator notifies all observers (such as the display system or other staff terminals) that stock has changed.
    - This ensures real-time updates across the system.
6. Confirmation Returned to Customer
    - Once the command executes successfully,
      the Mediator sends a confirmation message to the Customer.

Scenario: A customer interacts with the nursery system to request information or advice about a plant.

Step by Step

Step 1: Request Initiation (Customer → Mediator)

- The Customer initiates an action (e.g., asks for plant care advice or availability).
- This request is sent to the InteractionMediator — a centralized communication hub that manages how different system components (customers, staff, inventory) talk to each other.
- The Mediator decouples the Customer from the Staff hierarchy, meaning neither side needs to know who they're directly interacting with.

Step 2: Request Delegation (Mediator → CoR)

- The InteractionMediator forwards this request to the Chain of Responsibility (CoR) of staff members.
- This chain could include:
  - SalesStaff
  - GreenhouseStaff
  - Manager
- Each staff member evaluates whether they can handle the request based on its type or context.
- If the current staff cannot process it, they pass it along to the next in the chain.

Example:

SalesStaff: "That's not my area. Let me forward this to the GreenhouseStaff."

S**tep 3: Request Handling (CoR → Strategy)**

- When a capable staff member is found (e.g., GreenhouseStaff), that staff member handles the request using a **Strategy** pattern.
- The Strategy encapsulates the specific interaction behavior.

Different Strategy types:

- AdviceStrategy → give plant care instructions
- AvailabilityStrategy → check stock and availability
- AccessoriesStrategy → suggest complementary products

This allows staff behavior to be **easily changed or extended** without modifying their base class.

Example:

GreenhouseStaff (using AdviceStrategy): "This plant prefers light watering twice a week."

Step 4: System Coordination (Mediator → InventoryMediator)

- If the request affects stock or inventory (e.g., checking plant availability or completing a purchase),
  the InteractionMediator communicates with the InventoryMediator.
- The InventoryMediator updates stock levels or retrieves product information — ensuring all departments stay synchronized.

Example:

- InventoryMediator: “There are 12 of this plant species available.”

Step 5: Response (Mediator → Customer)

- Once the request is successfully handled, the Mediator compiles the response and sends it back to the Customer.
- The customer receives clear feedback — such as advice, availability, or confirmation of a purchase.

Example:

Mediator to Customer: “This plant is in stock and requires watering twice a week.”