Smart Home System Programming Exercise

# Problem Statement

Create a simulation for a Smart Home System that allows the user to control different smart devices such as lights, thermostats, and door locks via a central hub. The user should be able to set schedules, automate tasks, and view the status of each device.

## Functional Requirements

1. Initialize the Smart Home System with different devices, each having their own unique ID and type (light, thermostat, door lock).

2. Implement features to:

• Turn devices on/off

• Schedule devices to turn on/off at a particular time

• Automate tasks based on triggers (e.g., turning off lights when the thermostat reaches a certain temperature)

3. Optional: Provide the ability to add or remove devices dynamically.

## Key Focus

1. Behavioral Pattern: Use the Observer Pattern to update all devices when a change occurs in the system.

2. Creational Pattern: Use the Factory Method for creating instances of different smart devices.

3. Structural Pattern: Use the Proxy Pattern to control access to the devices.

4. OOP: Ensure strong encapsulation, modularity, and the application of inheritance and polymorphism.

## Possible Inputs

Devices:

[{id: 1, type: 'light', status: 'off'}, {id: 2, type: 'thermostat', temperature: 70}, {id: 3, type: 'door', status: 'locked'}]

Commands:

['turnOn(1)', 'setSchedule(2, "06:00", "Turn On")', 'addTrigger("temperature", ">", 75, "turnOff(1)")']

## Possible Outputs

Status Report:

"Light 1 is On. Thermostat is set to 70 degrees. Door is Locked."

Scheduled Tasks:

[{device: 2, time: "06:00", command: "Turn On"}]

Automated Triggers:

[{condition: "temperature > 75", action: "turnOff(1)"}]

## Evaluation

1. Code Quality: Evaluation criteria remain consistent with best practices, SOLID principles, and effective use of design patterns.

2. Functionality: Does the solution meet all the requirements and handle edge cases gracefully?

3. Global Convention: Is the code globally understandable and well-documented?

4. Gold Standards: Is the code up to the gold standard in terms of logging, error handling, and performance optimization?

5. Code Walkthrough: Can the candidate explain their solution coherently, focusing on the architecture, design decisions, and patterns used?

## Intent

The intent of this Smart Home System exercise is to bring together elements of real-world applicability and technical depth.

# How to Install from GitHub to Local

1. Clone the repository:

Open your terminal and run the following command:

git clone https://github.com/your-repo/smart-home-system.git

2. Navigate to the project directory:

cd smart-home-system

3. Install the required dependencies:

npm install

4. Run the project:

npm start or ts-node src/index.ts

5. For TypeScript compilation:

npm run build