**Exercise 9: Implementing the Command Pattern**

**SCENARIO:**

You are developing a home automation system where commands can be issued to turn devices on or off. Use the Command Pattern to achieve this.

**STEPS:**

1. **Create a New Java Project:**
   * Create a new Java project named CommandPatternExample.
2. **Define Command Interface:**
   * Create an interface Command with a method execute().
3. **Implement Concrete Commands:**
   * Create classes LightOnCommand, LightOffCommand that implement Command.
4. **Implement Invoker Class:**
   * Create a class RemoteControl that holds a reference to a Command and a method to execute the command.
5. **Implement Receiver Class:**
   * Create a class Light with methods to turn on and off.
6. **Test the Command Implementation:**
   * Create a test class to demonstrate issuing commands using the RemoteControl.

**SOLUTION:**

**Step 1:** Create a new Java project named CommandPatternExample.

**Step 2:** Define Command Interface

// Command.java

interface Command {

void execute();

}

**Step 3:** Implement Concrete Commands

// LightOnCommand

class LightOnCommand implements Command {

private Light light;

public LightOnCommand(Light light) {

this.light = light;

}

@Override

public void execute() {

light.turnOn();

}

}

// LightOffCommand

class LightOffCommand implements Command {

private Light light;

public LightOffCommand(Light light) {

this.light = light;

}

@Override

public void execute() {

light.turnOff();

}

}

**Step 4:** Implement Invoker Class

// RemoteControl

class RemoteControl {

private Command command;

public void setCommand(Command command) {

this.command = command;

}

public void pressButton() {

command.execute();

}

}

**Step 5:** Implement Receiver Class

// Light

class Light {

public void turnOn() {

System.out.println("The light is on.");

}

public void turnOff() {

System.out.println("The light is off.");

}

}

**Step 6:** Test the Command Implementation

public class CommandPatternExample {

public static void main(String[] args) {

Light livingRoomLight = new Light();

// Create concrete commands

Command lightOn = new LightOnCommand(livingRoomLight);

Command lightOff = new LightOffCommand(livingRoomLight);

// Create invoker

RemoteControl remote = new RemoteControl();

// Turn on the light

remote.setCommand(lightOn);

remote.pressButton();

// Turn off the light

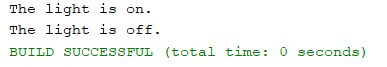
remote.setCommand(lightOff);

remote.pressButton();

}

}

**SAMPLE OUTPUT:**

****