**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**.

**CODE:**

using System;

interface ICommand

{

void Execute();

}

class Light

{

public void TurnOn()

{

Console.WriteLine("The light is ON.");

}

public void TurnOff()

{

Console.WriteLine("The light is OFF.");

}

}

class LightOnCommand : ICommand

{

private Light \_light;

public LightOnCommand(Light light)

{

\_light = light;

}

public void Execute()

{

\_light.TurnOn();

}

}

class LightOffCommand : ICommand

{

private Light \_light;

public LightOffCommand(Light light)

{

\_light = light;

}

public void Execute()

{

\_light.TurnOff();

}

}

class RemoteControl

{

private ICommand \_command;

public void SetCommand(ICommand command)

{

\_command = command;

}

public void PressButton()

{

\_command.Execute();

}

}

class Program

{

static void Main()

{

// Receiver

Light livingRoomLight = new Light();

// Commands

ICommand lightsOn = new LightOnCommand(livingRoomLight);

ICommand lightsOff = new LightOffCommand(livingRoomLight);

// Invoker

RemoteControl remote = new RemoteControl();

// Turn light ON

remote.SetCommand(lightsOn);

remote.PressButton();

// Turn light OFF

remote.SetCommand(lightsOff);

remote.PressButton();

}

}

**OUTPUT:**

