**DESIGN PATTERNS AND PRINCIPLES**

**Exercise 6: Implementing the Proxy Pattern**

**Scenario:**

You are developing an image viewer application that loads images from a remote server. Use the Proxy Pattern to add lazy initialization and caching.

**Steps:**

1. **Create a New Java Project:**
   * Create a new Java project named ProxyPatternExample.
2. **Define Subject Interface:**
   * Create an interface Image with a method display().
3. **Implement Real Subject Class:**
   * Create a class RealImage that implements Image and loads an image from a remote server.
4. **Implement Proxy Class:**
   * Create a class ProxyImage that implements Image and holds a reference to RealImage.
   * Implement lazy initialization and caching in ProxyImage.
5. **Test the Proxy Implementation:**
   * Create a test class to demonstrate the use of ProxyImage to load and display images.

**DESCRIPTION:**

This program demonstrates the Proxy Pattern to delay loading of images until they are actually needed.

* Image is the interface with the display() method.
* RealImage loads and displays the image from a remote server.
* ProxyImage acts as a placeholder that loads RealImage only when required, enabling lazy initialization and caching.

**PROGRAM:**

public class ProxyPatternExample {

public static void main(String[] args) {

Image image1 = new ProxyImage("sunset.jpg");

Image image2 = new ProxyImage("mountain.jpg");

image1.display();

System.out.println("---");

image1.display();

System.out.println("---");

image2.display();

}

}

interface Image {

void display();

}

class RealImage implements Image {

private String fileName;

public RealImage(String fileName) {

this.fileName = fileName;

loadFromServer();

}

private void loadFromServer() {

System.out.println("Loading " + fileName + " from remote server...");

}

public void display() {

System.out.println("Displaying " + fileName);

}

}

class ProxyImage implements Image {

private RealImage realImage;

private String fileName;

public ProxyImage(String fileName) {

this.fileName = fileName;

}

public void display() {

if (realImage == null) {

realImage = new RealImage(fileName);

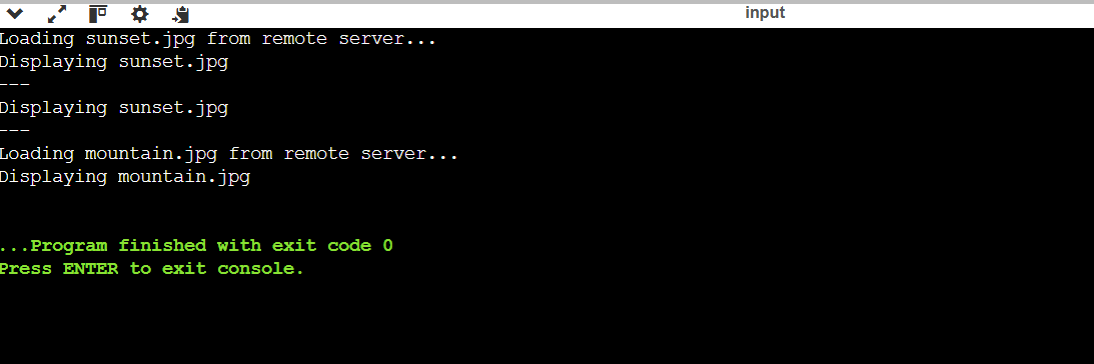
}

realImage.display();

}

}

**OUTPUT:**

****