WEEK – 1

Implementing the Proxy Pattern

Nipuna Amanapu

[namanapu@gitam.in](mailto:namanapu@gitam.in)

superset id : 6432842

**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.

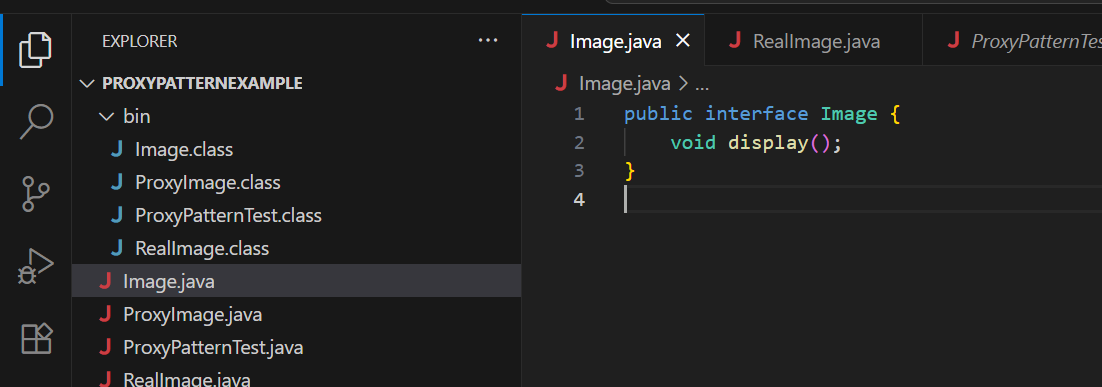
Solution:

1)Image.java

public interface Image {

    void display();

}



2)realImage.java

public class RealImage implements Image {

    private String filename;

    public RealImage(String filename) {

        this.filename = filename;

        loadFromRemoteServer();

    }

    private void loadFromRemoteServer() {

        System.out.println("Loading image from remote server: " + filename);

        try {

            Thread.sleep(1000);

        } catch (InterruptedException e) {

            e.printStackTrace();

        }

    }

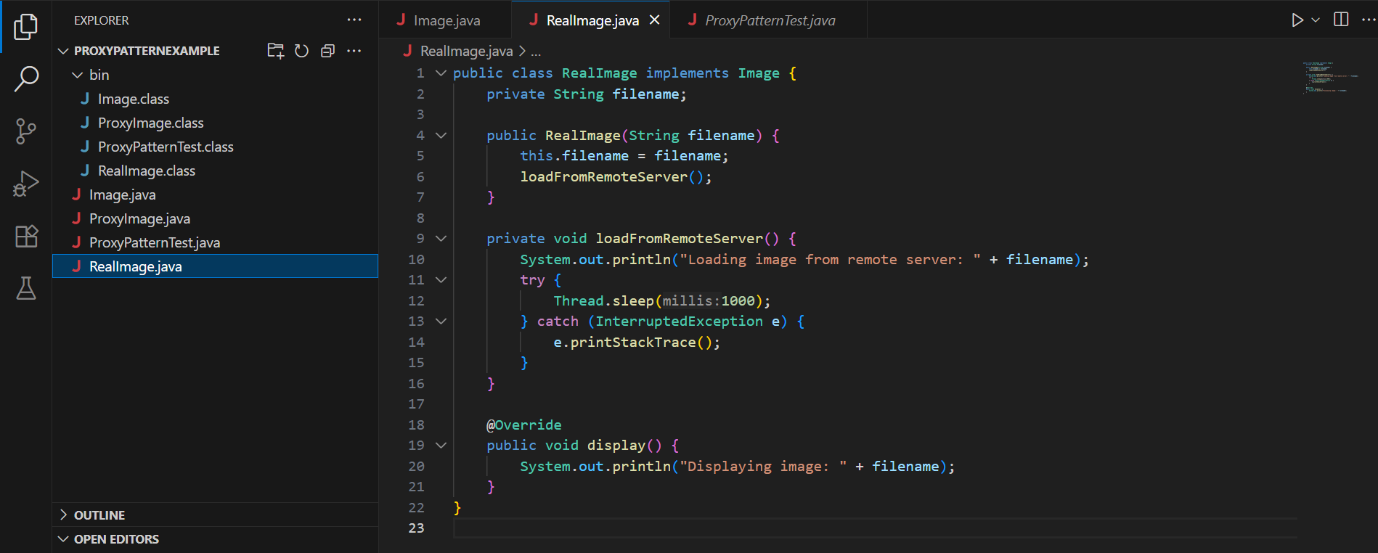
    @Override

    public void display() {

        System.out.println("Displaying image: " + filename);

    }

}



3)ProxyImage.java

public class ProxyImage implements Image {

    private RealImage realImage;

    private String filename;

    public ProxyImage(String filename) {

        this.filename = filename;

    }

    @Override

    public void display() {

        if (realImage == null) {

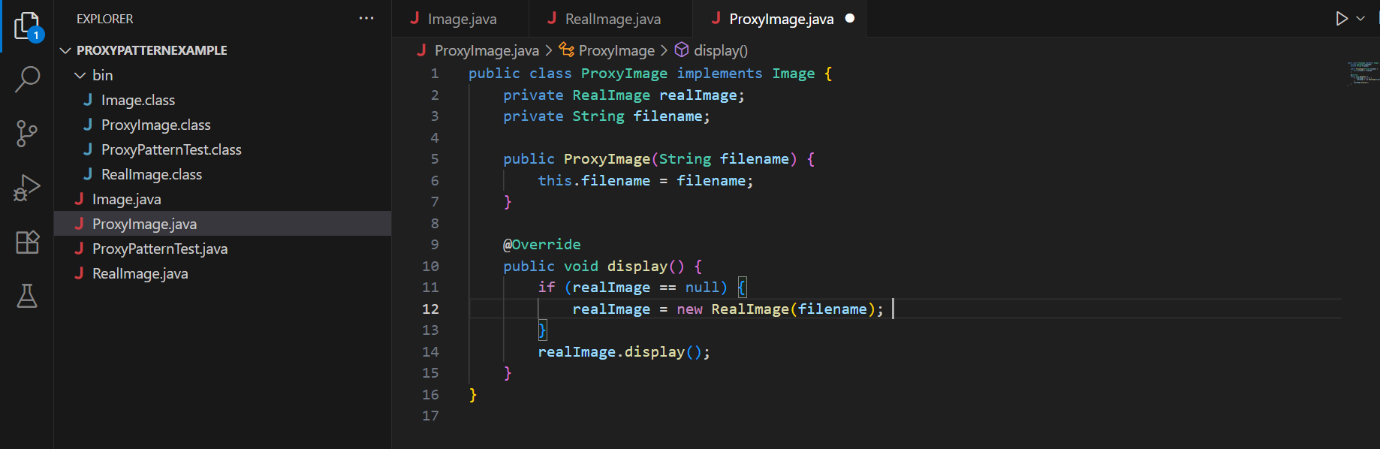
            realImage = new RealImage(filename);

        }

        realImage.display();

    }

}



4)ProxyPatterTest.java

public class ProxyPatternTest {

    public static void main(String[] args) {

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

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

        System.out.println("Calling display on image1 (first time):");

        image1.display();

        System.out.println("\nCalling display on image1 (second time):");

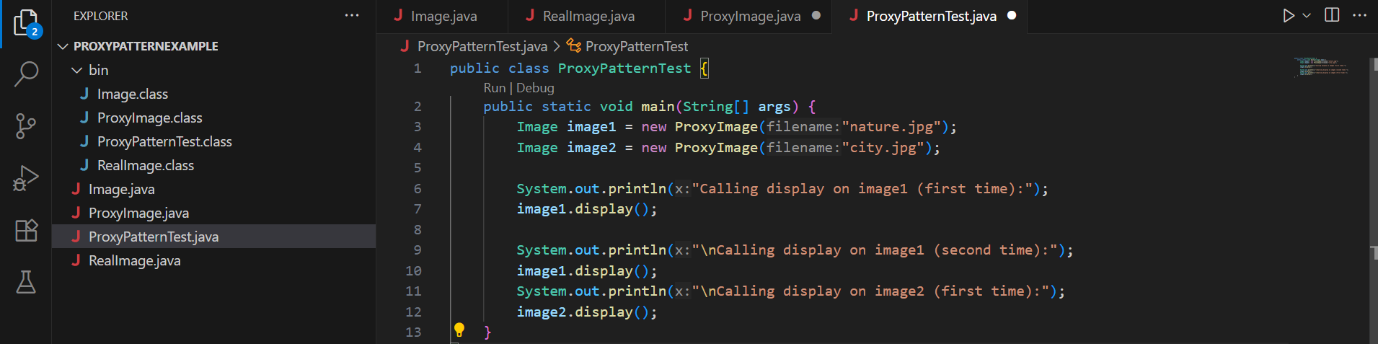
        image1.display();

        System.out.println("\nCalling display on image2 (first time):");

        image2.display();

    }

}



Output:

