Design patterns and principles

SuperSet ID:6412063

Exercise 6: Implementing the Proxy Pattern

Code:

public interface Image {

void display();

}

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);

}

public void display() {

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

}

}

public 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); // Lazy initialization

} else {

System.out.println("Using cached image: " + fileName);

}

realImage.display();

}

}

public class ImageViewer {

public static void main(String[] args) {

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

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

image1.display();

image1.display();

image2.display();

image2.display();

}

}

Output:

A close-up of a computer screen

Description automatically generated