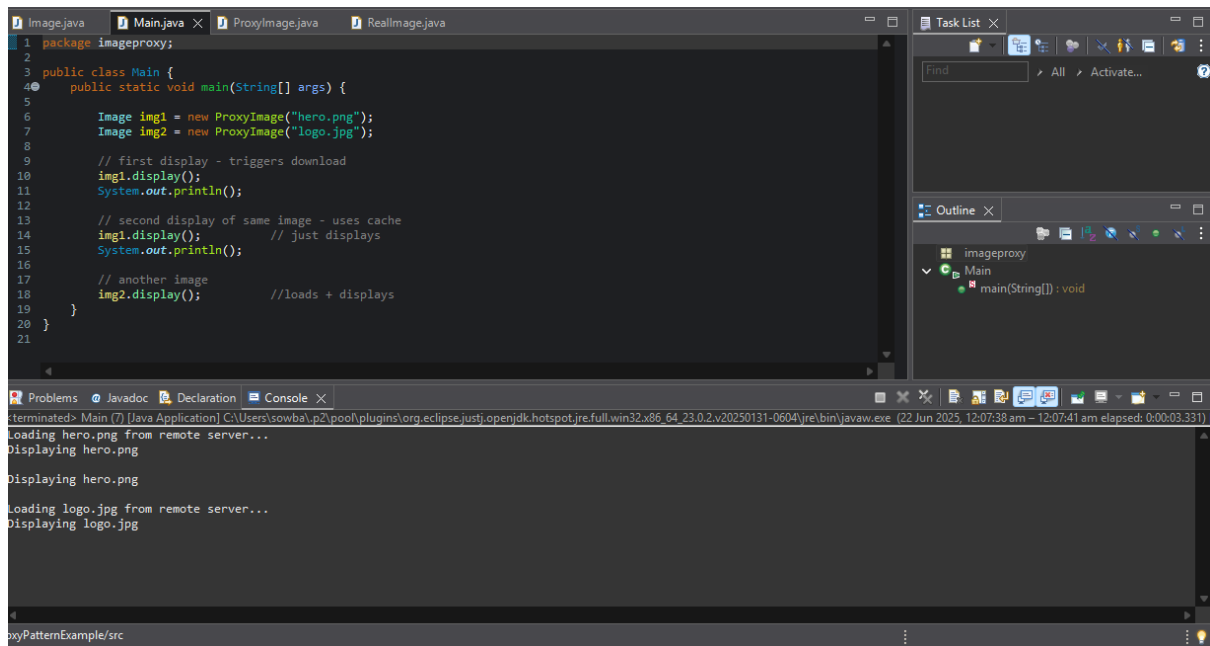


Exercise 6: Implementing the Proxy Pattern

OUTPUT:



The screenshot shows the Eclipse IDE with the following components:

- Editor:** Displays the `Main.java` file with the following code:

```
1 package imageproxy;
2
3 public class Main {
4     public static void main(String[] args) {
5
6         Image img1 = new ProxyImage("hero.png");
7         Image img2 = new ProxyImage("logo.jpg");
8
9         // first display - triggers download
10        img1.display();
11        System.out.println();
12
13        // second display of same image - uses cache
14        img1.display(); // just displays
15        System.out.println();
16
17        // another image
18        img2.display(); // loads + displays
19    }
20 }
21
```
- Outline:** Shows the package structure: `imageproxy` containing `Main` with the method `main(String[]): void`.
- Console:** Shows the output of the program:

```
terminated> Main [?] [Java Application] C:\Users\sowba\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_23.0.2.v20250131-0604\jre\bin\javaw.exe (22 Jun 2025, 12:07:38 am - 12:07:41 am elapsed: 0:00:03.331)
Loading hero.png from remote server...
Displaying hero.png
Displaying hero.png
Loading logo.jpg from remote server...
Displaying logo.jpg
```

The Proxy Pattern was used to add lazy loading and caching for images. It improves performance by delaying image loading until needed, reducing unnecessary resource usage and enhancing application efficiency.