**Exercise 4: Creating and Configuring a Maven Project**

Path: [\\LOGES\Users\loges\eclipse-workspace\LibraryManagemen](file:///\\LOGES\Users\loges\eclipse-workspace\LibraryManagemen)

**Step 1: Create a New Maven Project**

**In Eclipse:**

1. Go to **File > New > Maven Project**
2. Check **"Create a simple project"** → Click **Next**
3. Fill:
   * **Group Id**: com.library
   * **Artifact Id**: LibraryManagement
   * **Packaging**: jar (or war if you want to build a web app later)
4. Click **Finish**

**Step 2: Add Spring Dependencies in pom.xml**

Replace your pom.xml content with the following:

xml

Copy code

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0

http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.library</groupId>

<artifactId>LibraryManagement</artifactId>

<version>1.0-SNAPSHOT</version>

<properties>

<java.version>1.8</java.version>

</properties>

<dependencies>

<!-- Spring Core + Context -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.34</version>

</dependency>

<!-- Spring AOP -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-aop</artifactId>

<version>5.3.34</version>

</dependency>

<!-- Spring WebMVC -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>5.3.34</version>

</dependency>

<!-- Servlet API (needed for WebMVC) -->

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>javax.servlet-api</artifactId>

<version>4.0.1</version>

<scope>provided</scope>

</dependency>

</dependencies>

<build>

<plugins>

<!-- Maven Compiler Plugin for Java 1.8 -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.1</version>

<configuration>

<source>${java.version}</source>

<target>${java.version}</target>

</configuration>

</plugin>

</plugins>

</build>

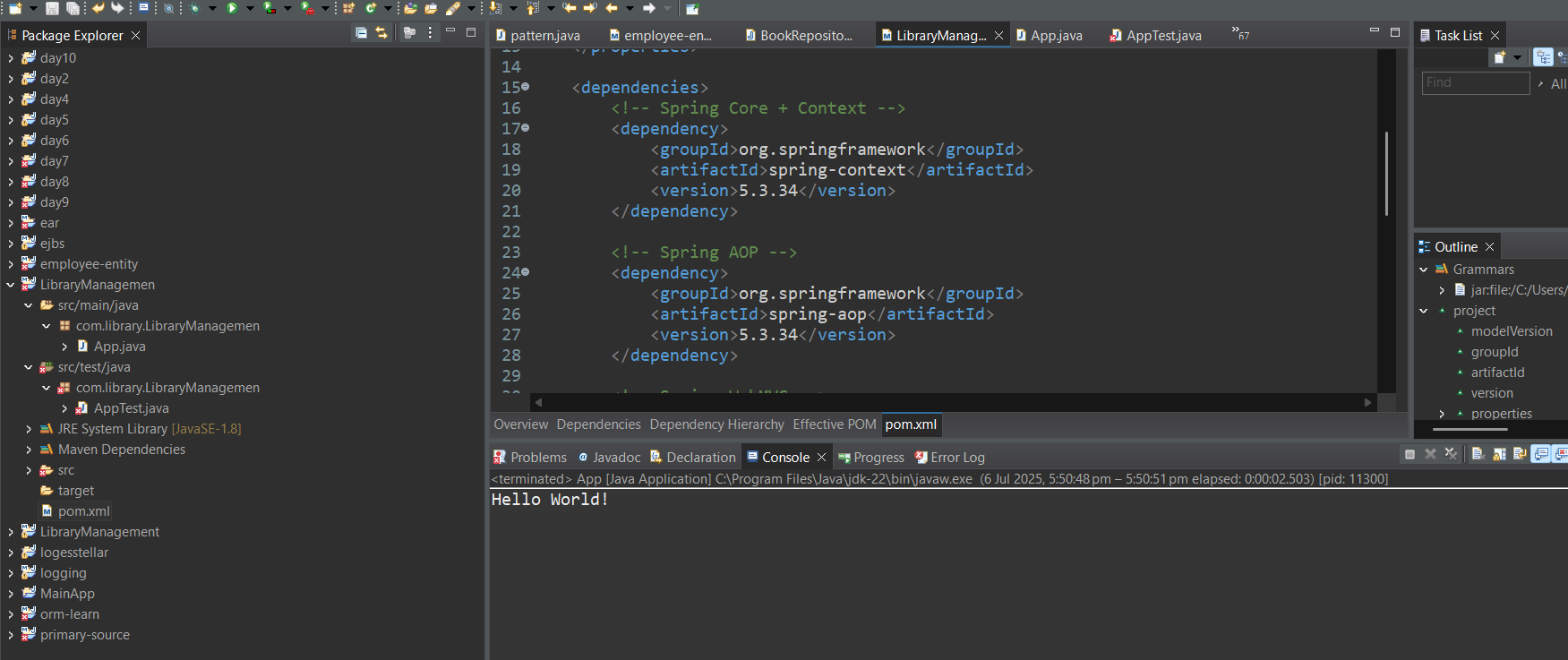
</project>

**Step 3: Run Maven Update**

After saving pom.xml:

* Right-click on the project → **Maven > Update Project**
* Check **"Force Update of Snapshots/Releases"** and click **OK**

**Output ScreenShot:**

****