# **Exercise 1: Configuring a Basic Spring Application**

```
eclipse-workspace - LibraryManagement/src/main/java/com/library/MainApp.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
7
8 public class MainApp {
99 public static void main(String[] args) {
10 ApplicationContext context=new ClassPathXmlApplicationContext("applicationContext.xml");
11
12 BookService bookService = (BookService)
                                                                                                                                          $ □ 1ª × × × • × • 8
                                                                                                                                          tom.library
> H InitEvample
                                                                                                                                        ✓ ₽ MainApp
 ∨ 🎥 src/main/java
                                                                                                                                           5 main(String[]): void

▼ A com.library

                                            BookService bookService = (BookService)context.getBean("bookService");
bookService.displayBook();
       > 🔃 MainApp.java
    > D BookRepository.java
                                            BookRepository bookRepository = (BookRepository)context.getBean("bookRepository");
bookRepository.saveBook();

▼ 

⊕ com.library.service

■ applicationContext.xml

  > 🕮 src/test/java
                                                                                                                         III Data Source Explorer ☐ Properties ☐ Console X ☐ Progress Jʊ JUnit
                                <terminated> MainApp [Java Application] C\Program Files\Java\jdk-24\bin\javaw.exe (04-Jul-2025, 7:46:16pm - 7:46:17pm elapsed: 0:00:01.244) [pid: 9812] BookService: Displaying book information. BookRepository: Saving book to database.
  > NRE System Library [JavaSE-17]
  > 🔑 src
    Imx.mog
 > 🔛 LoggingExampleApp
> MockitoExample
```

# **Exercise 2: Implementing Dependency Injection**

```
 \fbox{ eclipse-workspace - Library Management DI/src/main/java/com/library/Main App. java - Eclipse IDE and State of the St
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       ð
                                                                                                                                                                           package com.library;

a package com.library;

package com.library;

a package com.library;

a package com.library;

a package com.library;

a package com.library;

b package com.library;

a package com.library;

a package com.library;

b package com.library;

a package com.library;

b package com.library;

a package com.library;

b package com.library;

c terminated> MainApp (1) [Java Application] C\Program Files\Java\Jdk-24\bin\Javaw.exe (04-Jul-2025, 832:57p = BookService: Displaying book information.

b package com.library;

a package com.library;

c terminated> MainApp (1) [Java Application] C\Program Files\Java\Jdk-24\bin\Javaw.exe (04-Jul-2025, 832:57p = BookService: Displaying book information.

b package com.library;

a package com.library;

c terminated> MainApp (1) [Java Application] C\Program Files\Java\Jdk-24\bin\Javaw.exe (04-Jul-2025, 832:57p = BookService: Displaying book information.

b package com.library;

a package com.library;

b package com.library;

a package com.library;

b package com.library;

b package com.library;

b package com.library;

c program Files\Javav\Jdk-24\bin\Javaw.exe (04-Jul-2025, 832:57p = BookService: Displaying book information.

b pookService: Displaying book information.

   File Edit Source Refactor Navigate Search Project Run Window Help
 Project Explorer X
                                                                                                                                                🗎 🗎 LibraryManagementDI/pom.xml 🔹 applicationContext.xml 🛴 🛍 Data Source Explorer 🛄 Properties 💆 Console 🗴 📆 Progress 💆 Ju JUnit
       > 🔛 Library Management

✓ 

LibraryManagementDI

✓ 

## src/main/iava

▼ 

⊕ com.library.repository

                                                     BookRepository.java

▼ 

⊕ com.library.service

                                                     > D BookService.java
                                       > <section-header> MainApp.java
                            16 }
17
                                        > 🗾 BookRepository.java
                            >  BookService.java

✓ 

Æ src/main/resources

    ■ applicationContext.xml

                > 🎩 src/test/java
```

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

```
Ð
eclipse-workspace - LibraryManagement2/src/main/java/com/library/MainApp.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
🗀 🗎 LibraryManagement2/pom.xml 🔃 applicationContext.xml 🔃 Mill Data Source Explorer 🔲 Properties 🖳 Console 🗡 😽 Progress 🛭 Ju Junit
Project Explorer X
                                                                                                                                      ■ X ¾ 🖟 🖟 🖟 🗗 🗗 🛨 🚉
                   ☐ 🤄 🎖 🐉 🖇 1 package com.library;
                                   a import org.springframework.context.ApplicationCont import org.springframework.context.ApplicationCont Book saved to repository.

import com.library.service.BookService;

import com.library.service.BookService;
 > 📂 Library Management

▼ LibraryManagement2

▼ 

## src/main/java

                                                                                                                                                                          public class MainApp {

▼ A com,library

                                         public static void main(String[] args) {
    ApplicationContext context=new ClassPathXn
                                                                                                                                                                           =0
      > 🕖 MainApp.java
                                                BookService bookService=(BookService)conte

▼ 

⊕ com.library.repository

                                                bookService.addBook();
                                  13 }
      > DookRepository.java

▼ 

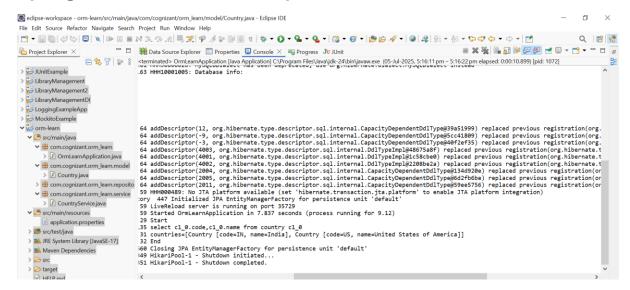
⊕ com.library.service

      > D BookService.iava
   ∨ ∰ src/main/resources

    ■ applicationContext.xml

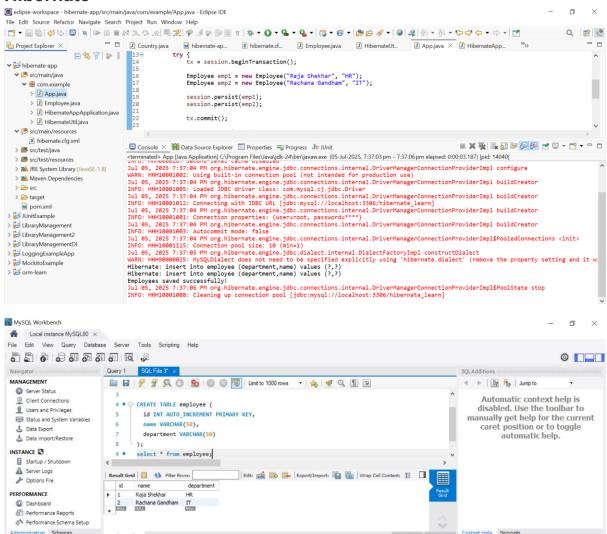
   > 🎩 src/test/java
   > 👺 src/test/resources
  > NRE System Library [JavaSE-1.8]
  > Mayen Dependencies
```

# **Spring Data JPA - Quick Example**

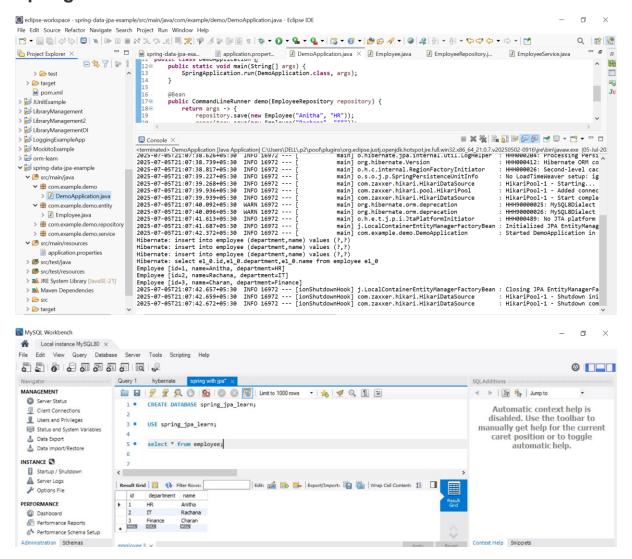


# Difference between JPA, Hibernate and Spring Data JPA

### Hibernate



# **Spring Data JPA**



### **Java Persistence API (JPA)**

## • What it is:

JPA is a specification (defined by JSR 338) for mapping Java objects to relational database tables and managing persistent data.

# Key point:

It is just an interface/standard, and does not provide its own implementation.

#### Usage:

You still need a JPA provider (like Hibernate, EclipseLink, etc.) to actually perform the persistence operations.

#### • Features:

Defines standard annotations (@Entity, @Id, etc.) and APIs for CRUD, querying (JPQL), and transaction management.

### Hibernate

#### What it is:

Hibernate is a framework (ORM tool) and a JPA implementation.

## • Key point:

It provides the actual implementation of the JPA specification, with additional proprietary features.

# • Usage:

You write code using JPA (or Hibernate-specific) APIs, and Hibernate translates it into SQL queries and manages database interactions.

### • Features:

- o Implements JPA.
- Adds advanced features like caching, lazy/eager loading, batch fetching, criteria API, etc.
- o Requires more boilerplate code than Spring Data JPA.

# **Spring Data JPA**

### • What it is:

Spring Data JPA is a Spring project that builds on top of JPA (and its implementation, like Hibernate).

### • Key point:

It is not a JPA implementation, but a higher-level abstraction that reduces boilerplate code and simplifies repository and query creation.

# • Usage:

You define interfaces that extend JpaRepository or CrudRepository, and Spring auto-generates implementations at runtime.

### • Features:

- o Automatically implements CRUD methods.
- Supports query derivation from method names (findByName, deleteById, etc.).
- Manages transactions and sessions for you.
- o Greatly reduces the amount of boilerplate code compared to plain Hibernate.