Spring Data JPA

Spring Data JPA is a part of the Spring Data family that makes it easier to work with Java Persistence API (JPA). It provides a set of abstractions and tools to simplify database operations in Spring applications.

Technologies Used

- Java

- Spring Boot

- Spring Data JPA

- Hibernate (as the JPA provider)

- H2 (in-memory database)

- RESTful API

Step 1: Add Dependencies in pom.xml

Add the following dependencies to your pom.xml:

```
<dependencies>
   <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-data-jpa</artifactId>
   </dependency>
   <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
   </dependency>
   <dependency>
      <groupId>com.h2database</groupId>
      <artifactId>h2</artifactId>
      <scope>runtime</scope>
   </dependency>
</dependencies>
```

Step 2: Configure application.properties

Create or edit src/main/resources/application.properties file:

```
spring.datasource.url=jdbc:h2:mem:librarydb
spring.datasource.driverClassName=org.h2.Driver
spring.datasource.username=sa
spring.datasource.password=
spring.jpa.database-platform=org.hibernate.dialect.H2Dialect
spring.jpa.hibernate.ddl-auto=update
spring.h2.console.enabled=true
```

Step 3: Create the Book Entity Class

```
import jakarta.persistence.*;
@Entity
public class Book {
   @Id
   @GeneratedValue(strategy = GenerationType.IDENTITY)
   private Long id;
   private String title;
   private String author;
   public Book() {}
   public Book(String title, String author) {
      this.title = title;
      this.author = author;
   }
   // Optional: Getters and Setters
}
```

Step 4: Create Repository Interface

```
import org.springframework.data.jpa.repository.JpaRepository;
public interface BookRepository extends JpaRepository<Book, Long> {
   // No need to add anything — basic CRUD is automatically handled!
}
```

Step 5: Create a REST Controller

```
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;
import java.util.List;
@RestController
@RequestMapping("/books")
public class BookController {
    @Autowired
    private BookRepository bookRepository;
    @PostMapping
    public Book createBook(@RequestBody Book book) {
        return bookRepository.save(book);
    }
    @GetMapping
    public List<Book> getAllBooks() {
        return bookRepository.findAll();
    }
}
```

Step 6: Run and Test the Application

- Run the Application using your IDE or terminal:

```
mvn spring-boot:run
```