**WEEK-3**

**MODULE-6**

**SPRING CORE AND MAVEN**

**Superset id:6407550**

**Difference between Java Persistence API, Hibernate and Spring Data JPA**

**JPA (Java Persistence API)**

JPA is part of **Jakarta EE (formerly Java EE)**.

It defines a standard for **ORM mapping, entity relationships**, and **query language** (JPQL).

Annotations like @Table, @Column, @GeneratedValue come from JPA.

Cannot perform any database operation on its own — must use a provider like Hibernate.

Focuses on **standardization and portability** of data persistence logic.

**Hibernate**

Hibernate is the most widely used **JPA implementation**.

Provides a powerful query language: **HQL (Hibernate Query Language)**.

Supports advanced features not in standard JPA:

* **First-level and Second-level Caching**
* **Batch processing**
* **Custom Interceptors and Filters**
* **Lazy Initialization Proxy Handling**

You can use **Hibernate-specific annotations** if you want more control (e.g., @Fetch, @LazyCollection).

Allows **XML- or annotation-based** configurations.

**Spring Data JPA**

Built on top of **Spring ORM + JPA + Hibernate**.

Removes boilerplate by allowing simple **Repository interfaces** like:

public interface BookRepository extends JpaRepository<Book, Long> {}

Supports **paging, sorting, and specifications** with almost no code.

Enables **declarative query methods** and also supports @Query for custom queries.

Integrates with **Spring Boot** to auto-configure DataSource, EntityManager, and TransactionManager.

Makes application code **cleaner, maintainable, and scalable**.