Name : K.Taraka Lakshmi Prasanna

Superset ID : 6314755

Email : 22bq1a4268@vvit.net

Week-3

Topic : Difference between JPA , Hibernet and Spring Data JPA

1. JPA (Java Persistence API) – *Specification*

- **What it is**: A **standard/specification** provided by Java (part of Jakarta EE) for object-relational mapping (ORM) between Java objects and relational databases.
- **Who provides it**: Oracle (as part of Java EE / Jakarta EE).
- **What it does**:
    - Defines interfaces like EntityManager, Query, EntityTransaction.
    - Doesn't provide the implementation.
- **Needs an implementation** (e.g., Hibernate, EclipseLink).

Think of JPA as a **set of rules or interfaces**.

---

2. Hibernate – *JPA Implementation*

- **What it is**: A **popular implementation** of the JPA specification (and also offers extra features beyond JPA).
- **What it provides**:
    - Implements all JPA interfaces.
    - Offers its own APIs (Session, Criteria, etc.) in addition to JPA.
    - Allows caching, lazy loading, fetching strategies, etc.
- **You can use Hibernate directly**, or as the JPA provider in a JPA-compliant setup.

Think of Hibernate as a **tool that follows the JPA rules and adds extra features**.

---

3. Spring Data JPA – *Abstraction/Framework*

- **What it is**: A **Spring-based abstraction** over JPA (and hence Hibernate) to reduce boilerplate.
- **What it provides**:
    - Repository interfaces (CrudRepository, JpaRepository) for CRUD operations.
    - Auto-generated queries from method names (e.g., findByName()).
    - Custom query support with JPQL/SQL.

- Integration with Spring Boot.

Think of Spring Data JPA as **a productivity tool that makes JPA easier to use**.

Feature	**JPA**	**Hibernate**	**Spring Data JPA**
Type	Specification (interface)	Implementation (library)	Framework (Spring abstraction)
Provided by	Oracle (Jakarta EE)	Red Hat	Spring Team (Pivotal)
Provides implementation	No	Yes	Uses Hibernate or another JPA provider
Usage	With EntityManager	With Session or JPA API	With JpaRepository, etc.
Boilerplate code	Moderate	Moderate	Minimal (auto methods, annotations)
Extendable APIs	Standard only	Extended APIs (e.g., Criteria)	Query creation, paging, etc.

How They Work Together:

➞ **Spring Data JPA** ⟶ uses **JPA** ⟶ implemented by **Hibernate**
(You write interface-based repositories; Spring handles boilerplate using JPA interfaces which Hibernate implements.)