**SPRING DATA JPA WITH SPRING BOOT, HIBERNATE**

**Difference between JPA, Hibernate and Spring Data JPA**

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

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Aspect** | **Explanation** | | **What it is** | **A Java specification (interface) for accessing, persisting, and managing data between Java objects and a relational database.** | | **Who defines it** | **Part of Jakarta EE (formerly Java EE) maintained by Eclipse Foundation.** | | **What it does** | **Defines how to map Java classes to database tables using annotations like @Entity, @Table, @Id, etc.** | | **Key point** | **It is only an API — it does not implement anything. You need a provider (like Hibernate) to use it.** | | **Example Annotations** | **@Entity, @Id, @OneToMany, @ManyToOne, etc.** | | **Example API** | **EntityManager, Query, CriteriaBuilder, etc.** |   **Think of JPA as the interface that standardizes ORM.** |

**2. Hibernate**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Aspect** | **Explanation** | | **What it is** | **A popular implementation of JPA. It is a framework that provides the actual ORM functionality.** | | **Who maintains it** | **Developed and maintained by Red Hat.** | | **JPA Implementation?** | **Yes. Hibernate implements JPA, but also provides additional features beyond the JPA specification.** | | **Additional Features** | **Caching, custom SQL, dirty checking, enhanced fetch strategies, @Where, @Filter, etc.** | | **Configuration** | **Can use hibernate.cfg.xml or be configured via Spring Boot application.properties.** | | **Query Support** | **HQL (Hibernate Query Language), Criteria API, native SQL.** |   **Think of Hibernate as the engine that powers JPA.** |

**3. Spring Data JPA**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Aspect** | **Explanation** | | **What it is** | **A Spring module that builds on top of JPA. It simplifies JPA-based repository access using interfaces and query methods.** | | **Who maintains it** | **Maintained by Spring Team / VMware.** | | **Goal** | **Eliminate boilerplate code for repositories and queries.** | | **Features** | **Auto-generated queries from method names (e.g., findByNameContaining()), custom queries with @Query, pagination, sorting, projections, auditing, etc.** | | **Abstraction level** | **High-level abstraction over JPA and Hibernate.** | | **No implementation needed** | **Just define an interface like extends JpaRepository<Entity, ID> — Spring auto-generates the implementation.** |   **Think of Spring Data JPA as a powerful abstraction layer on top of JPA + Hibernate.** |