**Spring-Data-JPA**

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Aspect** | **JPA (Java Persistence API)** | **Hibernate** | **Spring Data JPA** |
| **Type** | Specification (interface) | Implementation of JPA | Abstraction over JPA (helper framework) |
| **Provided By** | Java (by Oracle) | Red Hat | Spring Framework |
| **Purpose** | Defines standard APIs for ORM | Provides actual ORM functionality | Simplifies data access using repository abstraction |
| **What it is?** | A set of interfaces and annotations | A library that implements JPA (and adds extras) | A Spring module built on top of JPA |
| **Contains** | @Entity, @Id, EntityManager, etc. | Session, Criteria API, HQL, caching, etc. | Repositories like JpaRepository, custom queries |
| **Example Use** | EntityManager.persist(obj) | session.save(obj) | userRepo.save(obj) |
| **Configuration Required** | Needs manual setup of provider (like Hibernate) | Needs configuration of sessionFactory, etc. | Minimal setup — Spring Boot handles most automatically |
| **Ease of Use** | Moderate — lower-level APIs | More powerful but more verbose | Very easy — focuses on conventions and productivity |
| **Custom Queries** | Use JPQL or native queries | Supports JPQL, HQL, Criteria API | Supports JPQL, native SQL, derived methods |
| **Learning Curve** | Medium | Slightly higher (because of additional features) | Lowest (especially with Spring Boot) |

**Summary**

- JPA is just a standard — it defines how Java objects relate to database tables.  
- Hibernate is a JPA provider — it implements JPA and offers additional features.  
- Spring Data JPA is a Spring abstraction — it makes working with JPA easier by removing boilerplate code.

**Simple Example**

// JPA  
EntityManager em = emf.createEntityManager();  
em.persist(new User());  
  
// Hibernate  
Session session = sessionFactory.openSession();  
session.save(new User());  
  
// Spring Data JPA  
@Autowired  
UserRepository repo;  
repo.save(new User());