**Hands-on 4: Difference between JPA, Hibernate, and Spring Data JPA**

**Java Persistence API (JPA)**

* JPA is a Java specification (JSR 338) for persisting, reading, and managing data from Java objects to relational databases.
* It defines a set of interfaces and annotations but does not provide any implementation.
* It is used to standardize object-relational mapping (ORM) in Java.
* Requires an implementation provider like Hibernate to work.

**Hibernate**

* Hibernate is an ORM framework that provides the actual implementation of the JPA specification.
* It allows developers to map Java classes to database tables.
* It provides additional features such as caching, lazy loading, and HQL (Hibernate Query Language).
* Developers need to manually handle sessions, transactions, and exception handling.

**Spring Data JPA**

* Spring Data JPA is not a JPA implementation but an abstraction over JPA implementations like Hibernate.
* It reduces boilerplate code for data access by providing repository interfaces.
* Automatically handles common database operations like save, delete, update, and find.
* Manages transactions and simplifies integration with Spring applications.
* Allows developers to focus on the business logic rather than the persistence layer.

### Code Comparison: Hibernate vs Spring Data JPA

### Hibernate

* Requires manual session management using openSession() and close().
* Transactions must be explicitly started and committed using Transaction.
* Exception handling is manual using try-catch blocks.
* You need to write boilerplate code for common database operations like saving and fetching.
* More verbose and lower-level control over persistence logic.

#### Spring Data JPA

* Session management is handled automatically by Spring.
* Transactions are managed using annotations like @Transactional.
* No need for manual exception handling; Spring manages most of it internally.
* Provides built-in repository interfaces like JpaRepository which include methods for CRUD operations.
* Reduces boilerplate and simplifies data access with cleaner, declarative code.