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

JPA:

* Java Persistence API is a specification that defines a set of interfaces and annotations for working with relational databases in Java applications.
* JPA provides a common API for Object-Relational Mapping (ORM) frameworks like Hibernate, Entity Framework (.NET) etc., making it easier to switch between different ORM frameworks without changing the code.

Hibernate:

* Hibernateis an Object-Relational Mapping (ORM) framework for Java that provides a high-level API for managing database interactions.
* It allows programmer to map Java objects to database tables and perform database operations using a high-level API, instead of writing low-level SQL code.

Spring Data JPA:

* Spring Data JPA is a part of the larger Spring Data family that makes it easier to build Spring-powered applications that use data access technologies.
* By leveraging this abstraction, common operations like CRUD (Create, Read, Update, Delete) are handled automatically, eliminating the need to write repetitive code for basic queries. Thus, it reduces the boilerplate code.

Conclusion:

|  |  |  |  |
| --- | --- | --- | --- |
| Approach | JPA | Hibernate | Spring Data JPA |
| Type | Specification / Standard API | ORM Framework | Abstraction on top of JPA/Hibernate |
| Purpose | Defines standard Object-Relational Mapping (ORM) | Provides actual ORM functionality | Simplifies data access with repository abstraction (auto-queries) |
| Boilerplate | High | Medium | Lowest |
| Query Language | JPQL (Java Persistence Query Language) | HQL (Hibernate Query Language, extends JPQL) | Supports derived queries |