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

JPA is not a tool or a framework; it's an official Java specification that describes how to manage relational data in Java applications. It provides a set of concepts, interfaces, and annotations that define a standard for Object-Relational Mapping (ORM).

**2. Hibernate:**

Hibernate is a popular, open-source ORM framework and one of the most widely used implementations of the JPA specification

**JPA Provider:** Hibernate implements the interfaces and lifecycle rules defined by the JPA specification.

**3. Spring Data JPA: The Abstraction Layer**

Spring Data JPA is part of the larger Spring Data family and is not a JPA implementation itself. Instead, it's a framework that adds a layer of abstraction on top of a JPA provider like Hibernate.

**Simplifies Data Access:** Its primary goal is to significantly reduce the amount of boilerplate code required to implement data access layers.

**Repository Pattern:** Spring Data JPA provides a repository abstraction that allows you to create repository interfaces. Spring then automatically generates the implementation for these repositories at runtime.

**Query Generation:** It can automatically generate database queries based on method names in your repository interfaces. For example, a method named findByName(String name) would automatically generate a query to find a record by its name.