JPA

* JPA stands for java persistence Api and it is a specification that specifies the rules on how the java objects must be persisted to relational tables
* It does not actually map data to relational tables but it will specify the rules and structures for java objects
* It provides a set of interfaces and annotations like @Entity, @Id, etc., and provide annotations for specifying mapping relations among entities like @OneToOne, etc.,

Hibernate

* Hibernate is a popular provider which implements the JPA
* It will implement the logic which will follow the rules specified by the JPA to map java objects to relational tables
* It introduces ORM which stands for object relationship mapping which lets us to map the java objects of POJO classes to be mapped with the relational tables and its records without much manual intervention
* It provides many key features like caching, etc.,

Spring Data JPA

* Spring data JPA is built on top of the JPA and spring framework
* It provides abstractions which will simplify the process of connecting to databases and running SQL queries
* It reduces boiler plate code by introducing repository interfaces for performing CRUD operations without write the queries directly or writing the implementation of the queries
* It also provides us with features to write our own custom queries using JPQL, native SQL
* It combines JPA and hibernate and to provide easy understanding and simplify the coding part it provides higher abstraction that allows us to easily access table data and perform CRUD operations