***Introduction to JPA***

* JPA stands for Java Persistence API
* ORM (Object Relational Mapping )
* Hibernate is one of the implementations of JPA
* JPQL (Java Persistence Query Language)

**Note:** the only important difference between SQL and JPQL, SQL every query is default to its table and columns, JPQL we have mapping (class maps to the table and variables and fields inside the class are mapped to columns) ***[Hibernate and JPQL generates native query for the target database]***

***Spring boot- MySQL connection with JPA***

We need to add 2 dependencies in the “pom.xml” of Spring Boot

**<dependency>**

**<groupId>org.springframework.boot</groupId>**

**<artifactId>spring-boot-starter-data-jpa</artifactId>**

**</dependency>**

**<dependency>**

**<groupId>com.mysql</groupId>**

**<artifactId>mysql-connector-j</artifactId>**

**<scope>runtime</scope>**

**</dependency>**

***Note:*** you can search on the Spring initializer for (JPA, mysql)

***File .properties***

**spring.application.name=**{spring project name}

**spring.datasource.url=**{url from the database}

**spring.datasource.username=**{user name that owns the connection}

**spring.datasource.password=**{password of the connection}

**spring.jpa.database=**{name of the database}

***Creating first entity in JPA***

In the model class use

@Getter

@Setter

@AllArgsConstructor

@NoArgsConstructor

@Entity

@Table(name = “name of the table”)

@Column(name = “name of the column”) Note: one column per field

@Id the primary key

@GeneratedValue

Note: in the main class you can use @EntityScan(“entities path”)

***Creating first repository using @Repository annotation***

*Each repository is an interface, and add @Repository annotation and extends JpaRepository<identity class, datatype >*

*CrudRepository <T, ID>*

*PagingAndSortingRepository<T, ID>*

*Note: in the main class use @EnableJpaRepositories(“path”/{paths})*

***Get all records using JPA***

*In the repository extends the JPA repository , then go to the service and use the Autowired property in the field of the controllers*