**14. JPA Entities**

JPA = Java Persistance API

In JPA we will use POJO ( Plain Old Java Objects)

The packages that make reference to JPA entities are named “domain” or “model”

JPA requires none Constructor

Spring Data JPA it is made to work with Hibernate

Hibernate it is generating DDL SQL statement

**MVC** = it was created in 70 by Xerox

Keep the controler light! It should be like a traffic cop! No database connections in it.

Annotate Controller Class with @Controller. This will register the class as a Spring Bean and as a Controller in Spring MVC

For maping methods to http request paths use @RequestMapping

**Thymeleaf** = Java Template Engine, este o alternativa la JSP (Java Server Pages)

**Spring Boot** = if there it is no Spring Web context, the spring will start and after that stop.

**Toate aceste stereotipuri(@Controller, @Service) nu difera una de cealalta.** Singurul lucru este este ca acestea ii spun spring-ului ca sunt **Spring managed component**.

**@Autowired = iti leaga un field sa zicem de un bean.**

**@Qualifier = spune care Bean alege**

**@Primary = daca nu ai pus niciun Qualifier, foloseste automat Bean-ul care are stereotipul @Primary**

**@Profile =** o caracteristica din Spring Framework foarte puternica. Te ajuta sa iti controlezi aplicaia in medii de rulare diferite.

**Implementation Naming =**

* < Interface Name > + Impl  **for a class that implements just one interface**