* **Spring-boot-starter-parent:**
  + Powerful part of spring boot
  + This is a concept of parent poms.
  + Gives us a set of curated dependencies
    - Spring guys do a release
    - They know the dependencies of all the dozens of jars that are involved in this and they do a curated release. So we are inheriting all that version information from them.
* **RUN THE APP:**
  + Install an eclipse extension for batch editor
  + Set java home
    - Go the properties of this pc
    - Click on advanced
    - Click on environment variables
    - Set Java\_Home variable to current jdk location
    - Hit ok.
  + In cmd , go to current project location i.e spring5webapp and write: **mvnw spring-boot:run**
  + Now, the tomcat webapplication is running in port 8080
* **WHAT IS JPA?**
  + JPA stands for Java Persitence API
  + JPA is the official API for working with relational data in Java
  + JPA is only a specification
    - JPA is not a concrete implementation. That means there are other vendors to choose from it and there’s different implementations to choose for JPA
* **WHAT DOES JPA DO?**
  + JPA is a bridge from Java’s object world to how data is stored in relational databases
  + ORM – Object Relational Mapping
  + JPA offers Java developers database independence
  + One API will support many relational databases
* **Hibernate 5**
  + Hibernate 5 released in September 2015
  + Commercial support available from Red Hat
  + Adopted by Spring Framework in version 4.2 (July 2015)
  + Adopted by spring boot in version 1.4 (Jan 2016)
  + Database independent
* **Annotation:**
  + @entity: Convert the class into JPA entity
    - use javax.persistence.Entity – It is the official JPA API
* **Process:**
  + Port 8080 must be free
    - Check for the available port:
      * netstat -ao |find /i "listening"
    - Kill the task using port 8080
      * Taskkill /F /IM 12704, 12704 is just an example of process id of port 8080
  + Run the Spring5WebApplication.java in eclipse
  + Open src/main/resources/application.properties
    - Enable h2 database
    - Put spring.h2.console.enabled = true
  + On the web browser, type: localhost:8080/h2-console/
    - Put jdbc url : jdbc:h2:mem:testdb and connect
  + Everything else is on the code.
* **SPRING DATA REPOSITORIES**
  + Provides an implementation of the Repository Pattern
  + Concept is originally from Eric Evans’ book Domain Driven Design
  + A repository has methods for retrieving domain objects should delegate to a specialized Repository object such that alternative storage implementations may be interchanged.
  + This is important. It allows you to easily substitute the persistence layer.
    - i.e. going from SQL to NoSQL
* **SPRING DATA JPA**
  + Is a part of a larger family of Spring Data projects
  + Uses Hibernate for persistence to supported RDMS systems
    - Just about any major relational database
  + You extend a Java Repository Interface
  + Spring Data JPA provides the implementation at run time
  + No SQL required