UNDERSTANDING SPRING BOOT STARTER PROJECTS

1. We need a lot of frameworks to build application features:
    a. Build a REST API: We need Spring, Spring MVC, Tomcat, JSON conversion... etc.
    b. Write Unit Tests: We need Spring Test, JUnit, Mockito, ... etc.
2. How can I group them and make it easy to build applications?
    a. Starters: Convenient dependency descriptors for different features.
3. Spring Boot provides variety of starter projects:
    a. Web Application & REST API - Spring Boot Starter Web (spring-webmvc, spring – web, spring – boot – starter – tomcat, spring – boot – starter – json)
    b. Unit Tests - Spring Boot Starter Test
    c. Talk to database using JPA – Spring Boot Starter Data JPA
    d. Talk to database using JDBC – Spring Boot Starter JDBC
    e. Secure your web application or REST API – Spring Boot Starter Security
4. (REMEMBER) Starters: Define all application dependencies.

A Spring Starter project is a type of project template that allows developers to quickly bootstrap a Spring-based application with a set of preconfigured dependencies and settings. It is a convenient way to get started with building a new Spring-based application, as it eliminates the need to manually configure the project and its dependencies.

Spring Starter projects can be generated using Spring Initializr, a web-based tool that allows developers to select the necessary dependencies and settings for their project. Once the necessary options have been selected, the tool generates a project template that can be used as a starting point for the application.

Spring Starter projects typically include a basic structure for the application, such as a main application class, a configuration file, and a set of predefined dependencies such as Spring Web, Spring Data, and Spring Security. This allows developers to quickly start building their application without having to spend time setting up the project structure or worrying about configuring the dependencies correctly.