

SPRING BOOT ASSIGNMENT - 2

1) What does Spring Boot mean?

Ans: Spring Boot is an open source Java-based framework used to create a micro Service. It is developed by Pivotal Team and is used to build stand-alone and production ready spring applications.

2) What are the various Advantages of using Spring Boot?

Ans: Advantages of using Spring Boot:

- Reduces the time spent on development and increases the overall efficiency of the development team.
- Helps to autoconfigure all components for a production-grade Spring app.
- Facilitates the creation and testing of Java-based applications by providing a default setup for unit and integration tests.
- Helps to avoid all the manual work of writing boilerplate code, annotations, and complex XML configurations.
- Comes with embedded HTTP servers like Jetty and Tomcat to test web applications.

3) What are the various features of Spring Boot?

Ans: Features of Spring Boot:

- Web Development.
- SpringApplication.
- Application events and listeners.
- Admin features.
- Externalized Configuration.
- Properties Files.
- YAML Support.
- Type-safe Configuration.

4) What is the reason to have a spring-boot-maven module?

Ans: Reasons to have a Spring Boot maven module:

- **spring-boot: run:** operates the Spring Boot application.
- **spring-boot: repackage:** it repackages the war to be executable.
- **spring-boot: start and spring-boot:** no need to deal with the lifecycle of spring boot app.

SPRING BOOT ASSIGNMENT - 2

5) How to make Spring Boot venture utilizing Spring Initializer?

Ans: There are three ways of utilizing Spring Initializr:

- Spring Boot CLI
- Online Interface
- Spring Tool Suit

6) What do Dev Tools in Spring boot mean?

Ans: DevTools stands for Developer Tool. The aim of the module is to try and improve the development time while working with the Spring Boot application. Spring Boot DevTools pick up the changes and restart the application.

7) What does Spring Boot Starter Pom mean? Why Is It Useful?

Ans: Spring Boot provides a number of starters that allow us to add jars in the classpath. Spring Boot built-in starters make development easier. The spring-boot-starter-bunch will import all the required conditions for the Spring Batch application. Also if we want to use Spring and JPA for database access, we need to include the spring-boot-starter-data-jpa dependency in our pom.

8) What does Actuator in Spring Boot mean?

Ans: Spring Boot Actuator is a sub-task of Spring Boot. It adds a few creation review administrations to the application. There are also has numerous features added to the application out-of-the-case for dealing with the administration in a any condition. They're basically used to uncover diverse kinds of data about the running application.

9) What Is the Configuration File Name Used By Spring Boot?

Ans: `application.properties`. It will overwrite all the default designs.

10) Why in spring boot “Opinionated ” is used?

Ans: The “Opinionated Defaults Configuration” Approach to avoid lot of boilerplate code and configuration to improve Development, Unit Test and Integration Test Process.

11) What are esteem properties of Spring Boot?

Ans: There are 16 catagories of Spring Boot properties, few of them are as follow:

- Core Properties.
- Cache Properties.
- Mail Properties.
- JSON Properties.

SPRING BOOT ASSIGNMENT - 2

- Data Properties.
- Transaction Properties.
- Data Migration Properties.
- Integration Properties.

12) What Is the Configuration File Name, which is used By Spring Boot?

Ans: application.properties

13) Would we be able to Use Spring Boot with Applications Which Are Not Using Spring?

Ans: No, Spring Boot is restricted to spring applications only.

14) What Is Name Of The Configuration File, Which You Use In Spring Boot?

Ans: application.properties

15) How Might You Implement Spring Security In Spring Boot Application?

Ans: We have to include spring-boot-starter-security starter in pom.xml. Then make spring config class, which will expand WebSecurity Configure Adapter and override expected strategy to accomplish security in Spring boot application.

16) Would you be able to Control Logging with Spring Boot? How?

Ans: Yes, Spring Boot's default configurations provides a support for the use of Java Util Logging, Log4j2, and Logback. Using these, we can configure the console logging as well as file logging.

17) Differentiate Between An Embedded Container And A War?

Ans:

EMBEDDED CONTAINER	WAR
1. It is only one component of spring boot	1. To deploy the app in external container, we need to create war file.
2. Embedded container utilized during improvements	2. war benefits a considerable measure from Spring Boot
3. Ex – Embedded tomcat server	4. Deploy spring boot app to an external server

18) What does Spring Security mean?

SPRING BOOT ASSIGNMENT - 2

Ans: Spring Security is very adjustable authentication and access-control structure. It is the true standard for securing Spring-based applications. Spring Security is a system that spotlights on giving both authentication and approval to Java applications.

19) What does Aspect-Oriented Programming (AOP) mean?

Ans: Aspect-Oriented Programming (AOP) is a programming paradigm that aims to increase modularity by allowing the separation of cross-cutting concerns and make the application adaptable to change. Aspect-oriented programming entails breaking down program logic into distinct parts (so-called concerns, cohesive areas of functionality).

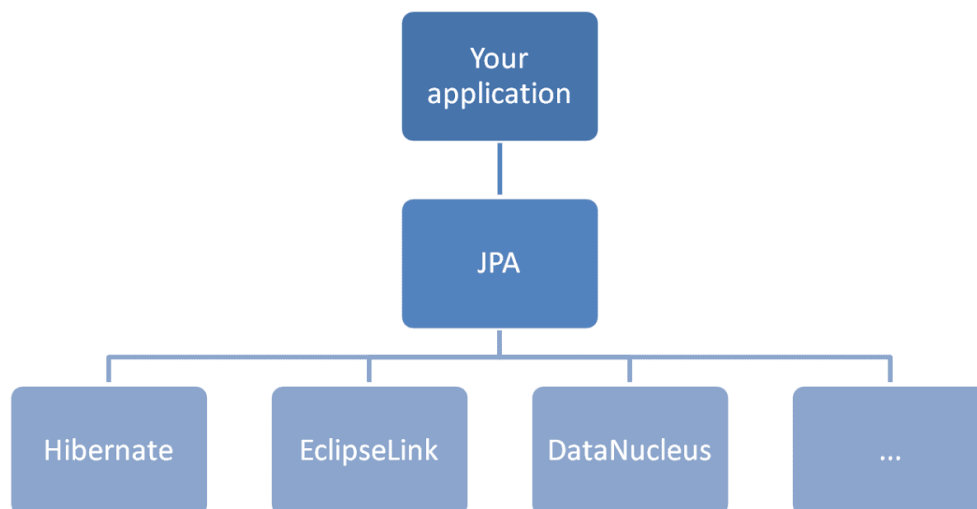
20) Describe some of the spring sub-projects briefly?

Ans: Spring sub-projects:

- **JDBC:** this module empowers a JDBC-deliberation layer that evaluates the need to do JDBC coding for vendor databases
- **Core:** a key module that gives basic parts of the system, as IoC or DI
- **MVC system:** a web module executing the Model View Controller configuration design
- **Web:** a web-situated joining module, giving multipart document upload, listeners members, and web-arranged application context functionalities
- **AOP module:** perspective-oriented programming execution is permitting the meaning of clean strategy interceptors and pointcuts.

21) Explain the difference between JPA and Hibernate?

Ans:



(Fig: 21.1 JPA vs Hibernate)

SPRING BOOT ASSIGNMENT - 2

Hibernate is a JPA implementation, while Spring Data JPA is a JPA Data Access Abstraction. It can also generate JPA queries on your behalf through method name conventions. Hibernate provides a reference implementation of the Java Persistence API that makes it a great choice as an ORM tool with benefits of loose coupling.

22) How to connect to an external database like MSSQL or oracle with Spring boot?

Ans:

Step 1 - The first step to connect the database like Oracle or MySql is adding the dependency for your database connector to pom.xml.

Step 2 - The next step is the elimination of H2 Dependency from pom.xml

Step 3 - Step 3 includes the schema and table to establish your database.

Step 4 - The next step is configuring of the database by using Configure application.properties to connect to your database.

Step 5- In the end, restart the machine.

23) How to add custom JS code in Spring Boot?

Ans: The steps to add a custom JS code with Spring Boot are as follows:

- Now, create a folder and name it static under the resources folder
- In this folder, you can put the static content in that folder

24) List minimum requirements for Spring boot System?

Ans:

Spring Boot 1.5.10.

- Java 7 +
- Spring 4.3.13 +

Build Support

- Maven 3.2+
- Gradle 2.9+

SPRING BOOT ASSIGNMENT - 2

Container Support

- Tomcat 7+

25) What is Auto Configuration in Spring boot?

Ans: Auto-configuration: It is a way in Spring Boot to configure a spring application automatically based on the dependencies in the classpath. When we use the Spring Boot Auto Configuration, automatically the spring-**boot-starter-data-jpa** dependency gets added to the pom.xml file.

EXAMPLE:

```
@Configuration
```

```
@ConditionalOnClass({ DataSource.class, EmbeddedDatabaseType.class })
```

```
@EnableConfigurationProperties(DataSourceProperties.class)
```

```
@Import({ Registrar.class, DataSourcePoolMetadataProvidersConfiguration.class })
```

```
public class DataSourceAutoConfiguration {.....}
```