

=====

Spring Boot

=====

=> Spring Boot is an extension for existing spring framework.

=> SpringBoot is an approach to develop Spring Based applications with less configurations.

=> Spring Boot came into market in 2015 (SpringBoot 1.x version)

=> The current version of Spring Boot is 3.x

=> What type of projects we can build using spring, same type of projects we can build using springboot also with very less configuration.

Note: In spring framework programmers are responsible for configuration management. Those configurations are common in every project.

=> Spring Boot supports rapid application development.

=====

Advantages with Spring Boot

=====

- 1) POM starters
- 2) Auto configuration
- 3) Embedded Servers
- 4) Actuators

=====

What is POM Starter ?

=====

=> POM starters are used to simplify maven dependencies configuration in project pom.xml. We have several springboot starters like below

- a) spring-boot-starter
- b) spring-boot-starter-web
- c) spring-boot-starter-data-jpa
- d) spring-boot-starter-mail
- e) spring-boot-starter-security
- f) spring-boot-starter-actuator

Note: For all these starters we have a parent starter 'spring-boot-starter-parent'.

=====

What is Auto Configuration ?

=====

=> Auto Configuration is one of the most popular functionality in springboot.

=> It is used to identify what configuration required for the application and provide that configuration in the application runtime.

Note: Auto Configuration works based on pom starters.

=> if we add 'web-starter' : It provides tomcat server for deployment

=> If we add 'security-starter' : It provides login page for authentication

=> If we add 'jpa-starter' : It provides DB connections

=====
What is Embedded Server ?
=====

=> Embedded Server nothing but in built server provided by springboot to run our application.

=> When we add 'web-starter' then springboot will provide tomcat server by default to run our web application.

Note: We no need to download and install external servers from now..

=> SpringBoot supports 3 Embedded Servers

- 1) Tomcat (default)
- 2) Jetty
- 3) Netty

=====
What is Actuator ?
=====

=> Actuators are used to monitor and manage our springboot application.

=> Using Actuators we can get below details

- 1) how many beans loaded by our application
- 2) how many url patterns available in our application
- 3) What config props loaded by our application
- 4) App Env Details
- 5) Threads Details of our application
- 6) Heap details of our application

=====
How to create SpringBoot application
=====

=> To create spring boot application, spring team provided 'spring intializr website'.

Website URL : <https://start.spring.io/>

=> When we use intializr website we have to download project as zip file then

extract it and import that project into our IDE (eclipse/intellij/STS).

Note-1 : In Eclipse IDE we can't create Spring Boot Application directly.

Note-2 : In IntelliJ Community IDE we can't create Spring Boot application directly. IntelliJ Enterprise edition having direct support to create boot app.

Note-3 : Spring Team provided STS IDE to create Spring Boot application directly. STS IDE also internally uses 'start.spring.io' website to create the project.

STS IDE URL : <https://spring.io/tools>

=====
What is start class in Spring Boot ?
=====

=> When we create springboot application by default one java class will be created that is called as Start class of spring Boot Application.

=> It is entry point for springboot application execution. Application execution starts from here only.

```
-----  
@SpringBootApplication  
public class Application {  
  
    public static void main(String[] args) {  
        SpringApplication.run(Application.class, args);  
    }  
}  
-----
```

=====
What is happening inside start class run () method ?
=====

=> run () method contains bootstrapping(getting started) logic for springboot application.

=> Below logics are implemented inside run () method

- 1) create bootstrap context
- 2) Run listeners
- 3) prepareEnvironment
- 4) print banner
- 5) createApplicationContext (IOC)
- 6) call Runners
- 7) Return IOC container obj

=====
What is @SpringBootApplication annotation?
=====

=> This annotation is used at start class of the Spring Boot.

=> This one annotation is equal to below 3 annotations

- 1) @SpringBootConfiguration
- 2) @EnableAutoConfiguration
- 3) @ComponentScan

Note: In springboot application, by default start-class package will be considered as base package for Component Scanning.

=====
What is Banner ?
=====

=> Banner is a logo that is printing on the console when we start our Springboot application.

=> We can customize springboot banner by creating "banner.txt" file under "src/main/resources" folder.

=> Spring Boot Banner works on Modes. We have below 3 modes

- 1) CONSOLE (default)
- 2) LOG
- 4) OFF

=> We can set banner mode like below in application.properties file

spring.main.banner-mode=off

=====
What is Runner in Spring Boot ?
=====

=> If we have any requirement to the execute the logic only once when the spring boot application got started then we can use Runners in Spring Boot.

Ex:

Usecase-1 : Load static tables data into cache memory when app starts.

Usecase-2 : remove data from temp tables (staging tables) in our database when application starts.

Usecase-3 : Send an email when application got started.

=> We have 2 types of Runners in Springboot

- 1) ApplicationRunner (FI)
- 2) CommandLineRunner (FI)

```
@Component
public class MyAppRunner implements ApplicationRunner {

    @Override
    public void run(ApplicationArguments args) throws Exception {
        System.out.println("MyAppRunner - run() method called");
    }
}
```

}

}

Note: Our Runners will be called as part of SpringApplication.run () method.

=====

Summary

=====

- 1) Spring Vs SpringBoot
- 2) What is Spring Boot & Why
- 3) Spring Boot Advantages
- 4) What is POM Starter
- 5) What is Auto Configuration
- 6) What is Embedded Server
- 7) What is Actuator
- 8) What is start class in Spring Boot ?
- 9) What is happening inside SpringApplication.run () method ?
- 10) What is @SpringBootApplication annotation ?
- 11) What is Banner in SpringBoot and how to customize ?
- 12) What is Runner in springboot & usecases

=====

By using Spring Boot with Spring Core concepts we have to develop applications

- 1) Web Applications
- 2) ReST APIs
- 3) Microservices