***Please refer the Day 9 and Day 10 for Spring MVC project and notes.***

***Spring core***

***Spring context***

***Spring mvc***

***Spring orm***

***Spring jdbc***

***Spring security***

***Spring aop***

***Spring boot : Spring boot is not to do specific task. Spring boot is use to bootstrap for spring existing modules.***

***Spring boot = all spring module – no xml file + few annotation + in build embedded web server ie tomcat.***

***Spring boot itself is core java project or standalone application(contains main methods) help to develop any type of application. This main method we are using in development mode.***

***Spring boot we need build tool mandatory.***

***Spring boot with maven build tool : xml base In maven we are using pom.xml file.***

***Spring boot with Gradle build tool : no xml file we using build file.***

***Spring boot modules***

1. ***Spring boot starter : Spring boot stater it provide different types of starter base upon type of application we are developing. This starter download all required dependencies which help to develop that type of application.***

***Web starter***

***Testing starter***

***Jpa starter***

***Jdbc starter***

***Security starter***

***Aop starter***

***Etc***

1. Auto configuration Spring boot provide auto configuration features. Base upon type of starter we added in project it provide default configuration. They are ready to inject us those resource in our application. In Spring boot no xml file configuration. In spring boot we use application.properties or application.yaml

Spring boot provide one of the annotation as

@SpringBootApplication = @Configutation + @ComponentScan + @AutoConfiguation

Spring Boot view technology

Thymeleaf : Thymeleaf as Java based library which provide dynamic html features in spring boot application. It use html as well as xhtml tags with dynamic features.

@SpringBootApplication

This annotation scan all those classes part of current package or sub package of current package. Inside that packages any classes with annotation @Controller, @RestController, @Service, @Repository. Then it will do DI.