

spring starters\*

Spring boot starter is a maven template that contains a collection of all the relevant transitive dependencies that are needed to start a particular functionality.

Like we need to import spring-boot-starter-web dependency for creating a web application.

for database access, just include the spring-boot-starter-data-jpa dependency in your project.

<dependency>

<groupId> org.springframework.boot</groupId>

<artifactId> spring-boot-starter-web </artifactId>

</dependency>

SPRINGBOOT

springboot is a spring module which provies rapid application develeopment feature to spring framework. we need springboot

to develop stand alone webapplications which means we can run the application directly without adding the external server.

What are the advantages of Spring Boot?

1.use to Create stand-alone Spring applications,where we can run the application directly without adding any server.

2.it has Embedded Tomcat, Jetty or Undertow directly. You don't need to deploy WAR files.

3.It provides opinionated 'starter' POMs to simplify your Maven configuration.

4.It automatically configure Spring whenever possible.

5.Provides production-ready features such as metrics, health checks, and externalized configuration.

6.Absolutely no requirement for XML configuration.

7.when u use devtool dependencies,it reloads automatically when u do any changes.

8.Increases productivity and reduces development time.

if u dont want to do auto scan a particular package/class how will u do dat?

by using exclude option in the component scan.

\*@controller

it indicates dat it is a controller class and a bean will be created for that class, and also it helps to identify the request mapping

of handler methods and it will register with the application context and handler mapper.

@restcontroller

restcontroller is used to create restfull webservices.

it is nothing but it is combination of @controller and @responsebody .

(@responsebody-it will convert java objects into json.)

Spring RestController takes care of request mapping of particular handler method. and also it will take responsiblity

of converting java objs to json or xml and send back in response.

(spring internally uses jackson api for json and jaxb api for converting the java obj to json/xml)

HTTP methods supported by REST are:

GET: It requests a resource at the request URL. It should not contain a request body as it will be discarded. Maybe it can be cached locally or on the server.

POST: It submits information to the service for processing; it should typically return the modified or new resource

PUT: At the request URL it update the resource

DELETE: At the request URL it removes the resource

OPTIONS: It indicates which techniques are supported

HEAD: About the request URL it returns meta information