Spring Boot

1. Spring boot is a java based framework.
2. Spring Boot used to reduce the development efforts and for faster development.
3. This Framework is distributes into multiple modules.
4. Some of the spring modules
   1. Spring IOC
   2. Spring JDBC
   3. Spring ORM
   4. Spring REST
   5. Spring Batch
   6. Spring MVC
   7. Spring Cloud
   8. Spring JMS
   9. Spring Security
5. Spring provides easy integration for other frameworks. That’s why it is also known as framework of framework.
6. Spring Boot is majorly used for developing fullstack or microservices application.
7. Spring boot is also use to develop a production graded application.
8. Spring Boot document

<https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>

1. Spring boot is a wrapper of the spring framework. Spring boot resolve the issues of the spring framework.
   1. Spring boot provided a starter Project using which you can easily get the dependencies of a module.
   2. Spring Boot provided an auto configuration functionality using which you can get the module configuration automatically which is highly customizable.
   3. Spring boot use an embedded tomcat server and H2 database by default.

**Spring Boot Project Creation**

1. Spring CLI (Command Line Interface)
   1. In this approach you have to install CLI application on local and execute command to create project.

[**https://docs.spring.io/spring-boot/docs/current/reference/html/cli.html**](https://docs.spring.io/spring-boot/docs/current/reference/html/cli.html)

1. Spring STS
   1. In this you have to download and install STS IDE. And there are option provided to create a spring boot project.

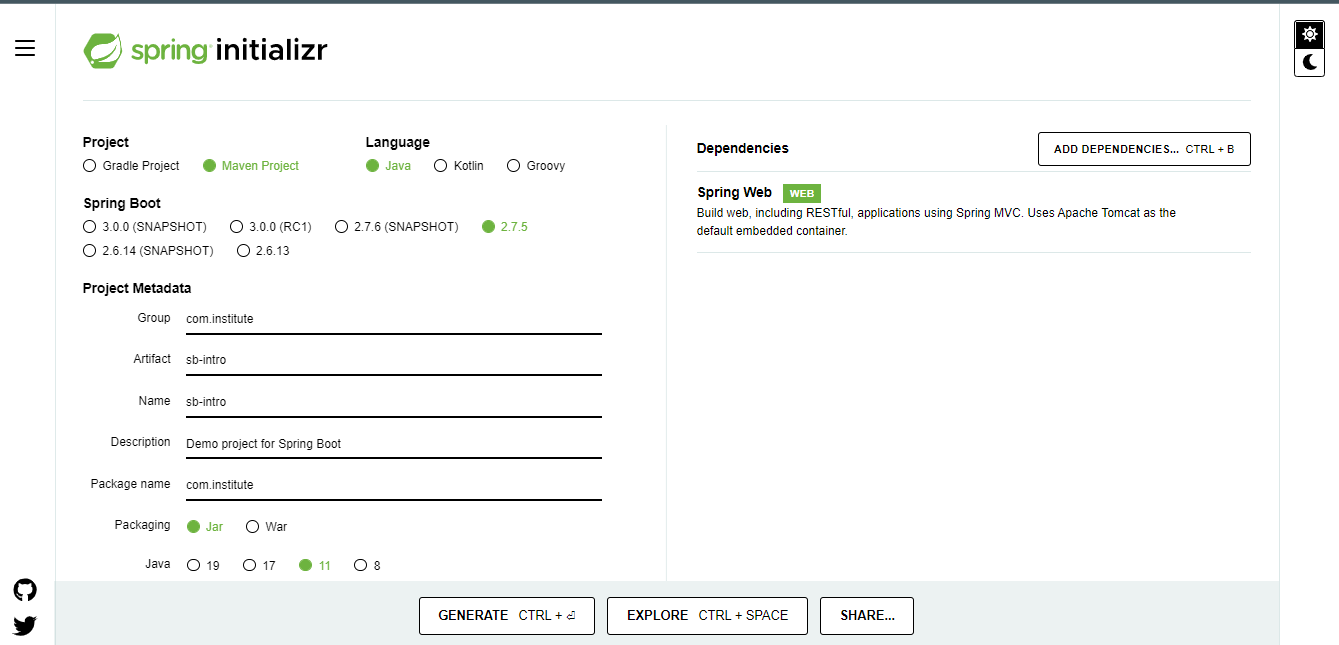
[**https://spring.io/tools**](https://spring.io/tools)

1. Spring Initializer
   1. It is a web application using which you can create s spring boot project.

[**https://start.spring.io/**](https://start.spring.io/)

**Create and Import spring boot project**

1. Create a spring boot project using spring initializer
2. Get the Project templated in the zip file.
3. Copy and paste the zip file into eclipse workspace
4. Extract the zip file
5. Steps to import project into eclipse
   1. File Menu -> select import option
   2. Search for maven option in the wizard
   3. Select “Existing Maven Project” option
   4. Click on Next -> Select Project path into root directory
   5. Click on Finish

****

**@SpringBootApplication**

1. This annotation is a combination of 3 annotation
   1. @Configuration: This annotation is use to mark any java class as a configuration class. These classes gets loaded into spring container at the application startup.
   2. @EnableAutoConfiguration: This will perform the autoconfiguration feature of the spring application. This configuration loaded based on the jar file added inside application.
   3. @ComponentScan: using this all spring bean classes will be loaded inside container. This annotation will scan all the spring bean classes and creates object of those java classes.

**Spring IOC**

1. IOC stands for Inverse of Control
2. You can get the Objects of java classes created by spring. Spring Bean classes are the classes which is manage by spring.
3. You can also inject the values inside object using CI(Constructor Injection) and SI (Setter injection)
4. You can perform Dependency Injection (DI).
5. Spring has a container which is used to store and manage the Object of spring bean classes, this container is also known as ApplicationContext.
6. Creating Spring Bean classes
   1. Create a java class and annotate java class with any of the following annotation.

@Component, @Controller, @RestController, @Service, @Repostiory, @ControllerAdvice

**Dependency Injection (DI)**

1. It is a process where you can create Object of one java class into another class.
2. It is also known as HAS-A relation in java.
3. This process can be achieve in spring which is known as DI.
4. This DI process can be automated using @Autowire annotation.
5. Using this annotation spring will inject the Object of java class into another java class.

**Download and Install Postman**

<https://www.postman.com/downloads/>

Full Stack Application Structure and REST API use



JSON

1. Stands for JavaScript Object Notation
2. This is use as a common language to communicate on cross platform and different languages.
3. JSON is in Key and Value pair, where Key and Value is called as a json element.
4. Key is always in String format, and values can be in different data type.
5. There is a JSON Object and JSON array which is used for a data communication.
6. JSON Object Example

{

"name":"John",

"age":30,

"car":null

}

1. JSON Array Example

["Ford", "BMW", "Fiat"]

[

{

"name":"John",

"age":30,

"car":null

},

{

"name":"John",

"age":30,

"car":null

}

]

JSON Example:

Employe Details

{

"name":"ABC",

"email" : "Abc@gmail.com",

"salary": 234234.23,

"contact": ["98978979870", "9899988776"],

"address" : [

{

"City":"Pune",

"Pincode":"98789",

state:"MH"

},

{

"City":"Mumbai",

"Pincode":"76778",

state:"MH"

}

],

"course" : {

"Name" : "Full Stack Development in Java",

"Duration" : "4 months",

"Technologies":["Java", "Advance Java", "Spring Boot", "Tools", "Html","Css", "JS", "ReactJS"]

}

}