

BY NAGOOR BABU

SPRING BOOT BY NAGOOR BABU

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

SPRING BOOT:

Spring Boot is an open source Java-based framework developed by Pivotal Team.

--> Spring Boot makes it easy to create stand-alone, production-grade Spring based Applications that you can "just run".

--> Spring Boot provides production-ready features such as metrics, health checks and externalized configuration.

--> It simplifies the bootstrapping and development of Spring Applications.

--> Spring Boot was designed on the top of existed Spring Framework to simplify and speedup the spring based applications.

--> Spring Boot can be used to develop Spring based applications either by using JAVA or by using Groovy.

--> Spring Boot Simplifies application development by having all configurations as defaults.

--> Spring Boot reduces application development time.

--> Spring Boot will improve Productivity.

--> Spring Boot will provide very simple approach to integrate Spring Boot Application with its Spring Ecosystem like Spring JDBC, Spring ORM, Spring Data, Spring Security etc.

--> Spring Boot follows "Opinionated Defaults Configuration" approach to reduce Developer effort, that is, Spring Boot avoids writing lots of boilerplate Code, Annotations and XML Configuration,...

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

--> Spring Boot provides Embedded HTTP servers like Tomcat, Jetty etc. to develop and test our web applications very easily.

--> Spring Boot provides CLI (Command Line Interface) tool to develop and test Spring Boot Applications from command prompt very easily and quickly.

--> Spring Boot provides lots of plugins to develop and test Spring Boot Applications very easily using Build Tools like Maven and Gradle

--> It provides lots of plugins to work with embedded and in-memory Databases very easily.

--> Where In-Memory databases are the databases, they will be created when application startup and it will be destroyed when application terminated.

EX: H2 Database.

--> In simple words, Spring Boot is the Existed Spring Framework with Embedded Web Servers like Tomcat and Jetty with No XML configurations and less Annotations.

SPRING BOOT FEATURES

To describe Spring Boot nature, Spring Boot has provided the following features.

- 1.SpringApplication
- 2.Externalized Configuration
3. Profiles

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

4. Logging
5. Internationalization
6. JSON
7. Developing Web Applications:
8. RSocket:
9. Security:
10. Working with SQL Databases:
11. Working with NoSQL Technologies:
12. Cache:
13. Messaging:
14. Calling REST Services with RestTemplate and WebClient:
15. Validations:
16. Sending Email:
17. Distributed Transactions with JTA:
18. Application events and listeners:
19. Admin features:
20. Properties Files
21. YAML Support:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

22.Type-safe Configuration:

1.SpringApplication:

SpringApplication is a class, it provides a convenient way to bootstrap a Spring application that is started from a main() method. SpringApplication has a static method run() to execute Spring application.

EX:

```
class MyApp{  
    public static void main(String[] args) {  
        SpringApplication.run(MyApp, args);  
    }  
}
```

When we run the above application then we will get the following output .

```
  ____  
 /__  '  _  ( )  _  _  \\\\  
( ) \_ | ' | ' | ' | ' V ' \\\\  
W _ ) | | | | | | | ( | | ) ) )  
' | _ | . _ | | | | | | \_ | / / / /
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
=====|_|=====|_|/=///
```

```
:: Spring Boot :: v2.2.0.BUILD-SNAPSHOT
```

```
2019-04-31 13:09:54.117 INFO 56603 --- [main] o.s.b.s.app.SampleApplication :  
Starting SampleApplication v0.1.0 on mycomputer with PID 56603 (/apps/myapp.jar  
started by pwebb)
```

```
2019-04-31 13:09:54.166 INFO 56603 --- [main]  
ationConfigServletWebServerApplicationContext : Refreshing  
org.springframework.boot.web.servlet.context.AnnotationConfigServletWebServerApplicat  
ionContext@6e5a8246: startup date [Wed Jul 31 00:08:16 PDT 2013]; root of context  
hierarchy
```

```
2019-04-01 13:09:56.912 INFO 41370 --- [main] .t.TomcatServletWebServerFactory :  
Server initialized with port: 8080
```

```
2019-04-01 13:09:57.501 INFO 41370 --- [main] o.s.b.s.app.SampleApplication  
: Started SampleApplication in 2.992 seconds (JVM running for 3.658)
```

If your application fails to start, registered FailureAnalyzers get a chance to provide a dedicated error message and a concrete action to fix the problem. For instance, if you start a web application on port 8080 and that port is already in use, you should see something similar to the following message:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

APPLICATION FAILED TO START

Description:Embedded servlet container failed to start. Port 8080 was already in use.

Action:Identify and stop the process that's listening on port 8080 or configure this application to listen on another port.

2.Externalized Configuration:

Spring Boot is giving an option to use like to externalize our application configurations so that we can work with the same application code in different environments. To provide Externalizable configurations we can use properties files, YAML files, environment variables, and command-line arguments,..

In Spring Boot applications, we are able to provide externalized configurations like

- 1.ServletConfig init parameters.
- 2.ServletContext init parameters.
- 3.JNDI attributes from java:comp/env.

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

4. Java System properties (System.getProperties()).

5. OS environment variables.

6. Profile-specific application properties outside of your packaged jar (application-{profile}.properties and YAML variants).

3. PROFILES:

Spring Profiles provide a way to segregate parts of your application configuration and make it be available only in certain environments.

IN Spring Boot applications, we are able to provide profiles configuration either through @Profile() annotation or through properties file or through Command Line Arguments.

EX: Through Annotation

```
@Configuration(proxyBeanMethods = false)
```

```
@Profile("production")
```

```
public class ProductionConfiguration {
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,


```
// ...
```

```
}
```

EX: Through Properties file

```
spring.profiles.active=dev,hsqldb
```

EX: Through Command Line Arguments.

```
--spring.profiles.active=dev,hsqldb.
```

4. Logging

Spring Boot uses Commons Logging for all internal logging but leaves the underlying log implementation open. Default configurations are provided for Java Util Logging, Log4J2, and Logback. In each case, loggers are pre-configured to use console output with optional file output also available.

5. Internationalization:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

Spring Boot supports localized messages so that your application can provide services to users of different languages . By default, Spring Boot looks for the presence of a messages resource bundle at the root of the classpath.

To provide Internationalization support we have to configure messages properties file base name in application configuration file like below.

```
spring.messages.basename=messages,config.i18n.messages
spring.messages.fallback-to-system-locale=false
```

6. JSON

Spring Boot provides integration with three JSON mapping libraries

1. Gson
2. Jackson
3. JSON-B

Spring Boot is using "Jackson" as the preferred and default library.

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

7. Developing Web Applications:

Spring Boot is well suited for web application development. We can create a self-contained HTTP server by using embedded Tomcat, Jetty, Undertow, or Netty.

Most web applications use the spring-boot-starter-web module to get up and running quickly.

8. RSocket:

RSocket is a binary protocol for use on byte stream transports. It enables symmetric interaction models via async message passing over a single connection.

The spring-messaging module of the Spring Framework provides support for RSocket requesters and responders, both on the client and on the server side.

9. Security:

If Spring Security is on the classpath, then web applications are secured by default. Spring Boot relies on Spring Security's content-negotiation strategy to determine whether to use

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

httpBasic or formLogin. To add method-level security to a web application, you can also add `@EnableGlobalMethodSecurity` with your desired settings

10. Working with SQL Databases:

The Spring Framework provides extensive support for working with SQL databases, from direct JDBC access using `JdbcTemplate` to complete “object relational mapping” technologies such as Hibernate. Spring Data provides an additional level of functionality like creating Repository implementations directly from interfaces and using conventions to generate queries from your method names.

11. Working with NoSQL Technologies:

Spring Data provides additional projects that help us access a variety of NoSQL technologies, including:

1. MongoDB

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

2.Neo4J

3.Elasticsearch

4.Cassandra

5.Couchbase

6.LDAP

12. Cache:

--> The Spring Framework provides support for transparently adding caching to an application. At its core, the abstraction applies caching to methods, thus reducing the number of executions based on the information available in the cache.

--> The caching logic is applied transparently, without any interference to the invoker. Spring Boot auto-configures the cache infrastructure as long as caching support is enabled via the @EnableCaching annotation.

13.Messaging:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

--> The Spring Framework provides extensive support for integrating with messaging systems, from simplified use of the JMS API using `JmsTemplate` to a complete infrastructure to receive messages asynchronously.

--> Spring AMQP provides a similar feature set for the Advanced Message Queuing Protocol.

--> Spring Boot also provides auto-configuration options for `RabbitTemplate` and `RabbitMQ`.

--> Spring WebSocket natively includes support for STOMP messaging, and Spring Boot has support for that through starters and a small amount of auto-configuration. Spring Boot also has support for Apache Kafka.

14. Calling REST Services with `RestTemplate` and `WebClient`:

Spring Framework has provided `RestTemplate` class to access remote Rest Services from our machine, but, Spring Boot has provided `RestSupportBuilder` to create `RestTemplate` instance as part of auto configurations.

If we want to use `Web Client` to call remote Rest Services then we have to use `WebFlux` module.

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

15. Validations:

Spring Boot is supporting Bean validations like Hibernate Bean Validations in order to validate data in bean objects.

16. Sending Email:

The Spring Framework provides an easy abstraction for sending email by using the `JavaMailSender` interface, and Spring Boot provides auto-configuration for it as well as a starter module.

17. Distributed Transactions with JTA:

Spring Boot supports distributed JTA transactions across multiple XA resources by using either an Atomikos or Bitronix embedded transaction manager. JTA transactions are also supported when deploying to a suitable Java EE Application Server.

When a JTA environment is detected, Spring's `JtaTransactionManager` is used to manage transactions. Auto-configured JMS, DataSource, and JPA beans are upgraded to support XA

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

transactions. You can use standard Spring idioms, such as `@Transactional`, to participate in a distributed transaction. If you are within a JTA environment and still want to use local transactions, you can set the `spring.jta.enabled` property to false to disable the JTA auto-configuration.

18.Application events and listeners:

Spring Boot uses events to handle variety of tasks. It allows us to create factories file that are used to add listeners. we can refer it by using `ApplicationListener` key.

Always create factories file in META-INF folder like: META-INF/spring.factories

19.Admin features:

Spring Boot provides the facility to enable admin related features for the application. It is used to access and manage application remotely. We can enable it by simply using `spring.application.admin.enabled` property.

20.Properties Files:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

Spring Boot provides rich set of Application Properties. So, we can use that in properties file of our project. Properties file is used to set properties like: server-port = 8082 and many others. It helps to organize application properties.

21.YAML Support:

It provides convenient way for specifying hierarchical configuration. It is a superset of JSON. The SpringApplication class automatically support YAML. It is successful alternative of properties.

22.Type-safe Configuration:

Strong type-safe configuration is provided to govern and validate the configuration of application. Application configuration is always a crucial task which should be type-safe. We can also use annotation provided by this library.

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

Key Components of Spring Boot Framework:

Spring Boot Framework has mainly four major Components.

- 1.Spring Boot Starters
- 2.Spring Boot AutoConfigurator
- 3.Spring Boot CLI
- 4.Spring Boot Actuator

1.Spring Boot Starters:

--> The main intention of Spring Boot Starter is to combine a group of common or related dependencies into single dependencies.

--> If we want to prepare any web application with tomcat server and with Database interaction then we have to add the following JARs minimum to the project library.

Spring beans jar file(spring-beans-xx.jar)

Spring core Jar file(spring-core-xx.jar)

Spring context Jar file(spring-context-xx.jar)

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

Spring Web Jar file(spring-web-xx.jar)

Spring Web MVC Jar file(spring-webmvc-xx.jar)

Servlet Jar file(servlet-xx.jar)

Spring JDBC Jar file(spring-jdbc-xx.jar)

Spring ORM Jar file(spring-orm-xx.jar)

Spring Transaction Jar file(spring-transaction-xx.jar)

The above approach will increase no of dependencies in our build file, it will increase build file size and it is very much difficult task to manage all these dependencies.

--> To overcome all the above problems, we have to use Spring Boot Starter Component.

--> Spring Boot Starter component combines all related jars into single jar file so that we can add only jar file dependency to our build files instead of adding above jars files to our build file, we need to add one and only one jar file like

“spring-boot-starter-web” jar file.

--> When we add “spring-boot-starter-web” jar file dependency to our build file, then Spring Boot Framework will automatically download all required jars and add to our project classpath.

Spring-boot-starter-web dependency will provide the following jars to our project library.

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

1. spring-boot-starter
 - a.spring-boot
 - b.spring-boot-autoconfigure
 - c.spring-boot-starter-logging
2. spring-web
3. spring-webmvc
4. spring-boot-starter-tomcat
 - a. tomcat-embedded-core
 - b.tomcat-embedded-logging-juli

2.Spring Boot AutoConfigurator:

In general, in spring based applications we have to provide lot of configuration details either through XML files or through Annotations.

EX: Beans configurations and their dependencies

HandlerMapping Configuration

ViewResolver configurations

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

To avoid all the above configurations Spring Boot has provided an autoConfigurator. Spring Boot Autoconfigurator will come automatically when we add "spring-boot-starter-web" dependency in build file, it will resolve views and Viewresolvers in Spring web MVC applications with out having Spring configuration file and if we use @SpringBootApplication annotation then it will avoid the explicit utilization of @Configuration, @ComponentScan and @EnableAutoConfiguration and provide them all these annotations to the Java Class internally.

@SpringBootApplication = @Configuration + @ComponentScan +
@EnableAutoConfiguration

3.Spring Boot CLI:

--> Spring Boot CLI(Command Line Interface) is a Spring Boot software, it will provide very good environment to run and test Spring Boot applications from command prompt.

--> When we run Spring Boot applications using CLI, then it internally uses Spring Boot Starter and Spring Boot AutoConfigurator components to resolve all dependencies and execute the application.

--> Spring Boot Command Line Interface has provided "spring" command to execute "Groovy" scripts from Command Prompt, that is, we have to prepare Controller classes in ".groovy" file.

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

Syntax: `spring run HelloWorld.groovy`

Where HelloWorld.groovy is like Java's HelloWorld.java file name.

--> When we run the above "spring" command on Command Line Interface then Spring Boot will activate GroovyC[Groovy Compiler] and Groovy Grap to add all the dependencies to project library and to execute and test Groovy script.

4.Spring Boot Actuator:

--> Spring Boot Actuators are providing no of Production-Ready features like

1. Spring Boot Actuator is providing environment to manage and monitor your application by using HTTP endpoints or with JMX.
2. Spring Boot Actuators are providing Auditing, health, and metrics automatically to your application.

--> When we run our Spring Boot Web Application using CLI, Spring Boot Actuator automatically provides hostname as "localhost" and default port number as "8080". We can access this application using "http://localhost:8080/" end point.

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

--> To activate Spring Boot Actuators we have to use the following dependencies in build file.

EX:

```
1. <dependency>
2.   <groupId>org.springframework.boot</groupId>
3.   <artifactId>spring-boot-starter-actuator</artifactId>
4. </dependency>
```

Steps to Prepare Spring Boot Applications:

Spring Boot Applications are prepared and executed in the following four ways.

1. Using Simple MAVEN project in Eclipse IDE.
2. STS IDE
3. Spring Boot Initializer
4. Spring Boot CLI[Command Line Interface]

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

1. Using Simple MAVEN project in Eclipse IDE:

1. Make Ready Setups for JDK1.8 and Eclipse IDE latest version.
2. Create MAVEN Project in Eclipse IDE.
3. Provide the required Dependencies in pom.xml file.
4. Prepare Java Class With main() method and the required Spring Boot Annotations.
5. Run Spring Boot Application.

1. Make Ready Setups for JDK1.8 and Eclipse IDE latest version:

Already we have JDK and Eclipse in our system, no need to reinstall them separately.

2. Create MAVEN Project in Eclipse IDE:

1. Right Click on Project Explorer.
2. Select "New".
2. Select "Others".

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

3. Search for "Maven".
4. Select "Maven Project".
5. Click on "Next" button.
6. select "Create a simple project(Skip Archutype)" checkbox.
7. Click on "Next" button.
8. Provide the following details in Wizard.
 1. group Id: com.durgasoft
 2. Artifact Id: helloapp
 3. Name : Hello Application
9. Click on "Finish" button.

3. Provide the required Dependencies in pom.xml file.

open pom.xml and provide the following dependencies in pom.xml.

- a) Provide Java-compiler-plugin in build path.

```
<build>
```

```
<plugins>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-compiler-plugin</artifactId>
  <version>3.6.0</version>
  <configuration>
    <source>1.8</source>
    <target>1.8</target>
  </configuration>
</plugin>
</plugins>
</build>
```

Right Click on Project, Select Maven and Select Update Project and Click on OK to get JAVA8 version in lib.

b) Provide Spring Boot Starter Parent

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

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
<artifactId>spring-boot-starter-parent</artifactId>
```

```
<version>1.5.1.RELEASE</version>
```

```
</parent>
```

c)Add Spring Boot Starter web in pom.xml

```
<dependencies>
```

```
<dependency>
```

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

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

```
</dependency>
```

```
</dependencies>
```

Observe all the dependent JARs under MAVEN Dependencies in MAVEN Project.

4. Prepare Java Class With main() method and the required Spring Boot Annotations:

```
package com.durgasoft;
```

```
import org.springframework.boot.SpringApplication;
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

```
@SpringBootApplication
```

```
1. public class HelloWorld {  
2.  
3.     public static void main(String[] args) {  
4.         SpringApplication.run(HelloWorld.class, args);  
5.         System.out.println("Hello From First Spring boot Application");  
6.     }  
7. }
```

Where @SpringBootApplication annotation will provide the required @Component, @ComponentScan and @EnableAutoConfigurator annotations.

Where SpringApplications provided run() method will start spring boot application.

If 8080 is busy in our system then Change Server port number from 8080 to 1111 [any number] by providing properties file:

src/main/resources/application.properties

server.port = 1111

5. Run Spring Boot Application:

1. Right Click on project

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

2. Select "RunAs"
3. Select "Java Application".
4. Select "HelloWord"

Then the following output on Console.

```
      . _ _ _ _ _ _ _ _ _ _  
  /\ / _ _ ' _ _ _ _ _ _ _ _  
  (( \ _ _ | ' _ | ' _ | ' _ V _ | \ \ \ \  
  W _ _ ) | | ) | | | | | | ( | | ) ) ) )  
  ' | _ _ | _ _ | | | | | | \ _ _ | / / / /  
  ===== | _ | ===== | _ _ / = / / / /  
  :: Spring Boot ::      (v1.5.1.RELEASE)
```

```
2019-10-06 22:02:48.309 INFO 14800 --- [      main] com.durgasoft.HelloWorld :  
Starting HelloWorld on DESKTOP-LTICM7V with PID 14800  
(D:\spring_boot_practice\helloapp\target\classes started by Nagoor in  
D:\spring_boot_practice\helloapp)  
2019-10-06 22:02:48.312 INFO 14800 --- [      main] com.durgasoft.HelloWorld :  
No active profile set, falling back to default profiles: default
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
2019-10-06 22:02:48.396 INFO 14800 --- [      main]
ationConfigEmbeddedWebApplicationContext : Refreshing
org.springframework.boot.context.embedded.AnnotationConfigEmbeddedWebApplication
Context@44a3ec6b: startup date [Sun Oct 06 22:02:48 IST 2019]; root of context hierarchy

2019-10-06 22:02:49.587 INFO 14800 --- [      main]
trationDelegate$BeanPostProcessorChecker : Bean
'org.springframework.boot.autoconfigure.validation.ValidationAutoConfiguration' of type
[class org.springframework.boot.autoconfigure.validation.ValidationAutoConfiguration] is
not eligible for getting processed by all BeanPostProcessors (for example: not eligible for
auto-proxying)

2019-10-06 22:02:49.686 INFO 14800 --- [      main]
trationDelegate$BeanPostProcessorChecker : Bean 'validator' of type [class
org.springframework.validation.beanvalidation.LocalValidatorFactoryBean] is not eligible
for getting processed by all BeanPostProcessors (for example: not eligible for auto-
proxying)

2019-10-06 22:02:50.277 INFO 14800 --- [      main]
s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat initialized with port(s): 1111 (http)

2019-10-06 22:02:50.294 INFO 14800 --- [      main]
o.apache.catalina.core.StandardService : Starting service Tomcat

2019-10-06 22:02:50.295 INFO 14800 --- [      main]
org.apache.catalina.core.StandardEngine : Starting Servlet Engine: Apache Tomcat/8.5.11

2019-10-06 22:02:50.424 INFO 14800 --- [ost-startStop-1] o.a.c.c.C.[Tomcat].[localhost].[/]
: Initializing Spring embedded WebApplicationContext
```

CONTACT US:**Mobile: +91- 8885 25 26 27**Mail ID: durgasoftonlinetraining@gmail.com**+91- 7207 21 24 27/28**WEBSITE: www.durgasoftonline.com**US NUM: 4433326786**

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
2019-10-06 22:02:50.424 INFO 14800 --- [ost-startStop-1] o.s.web.context.ContextLoader
: Root WebApplicationContext: initialization completed in 2033 ms

2019-10-06 22:02:50.635 INFO 14800 --- [ost-startStop-1]
o.s.b.w.servlet.ServletRegistrationBean : Mapping servlet: 'dispatcherServlet' to [/]

2019-10-06 22:02:50.644 INFO 14800 --- [ost-startStop-1]
o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'characterEncodingFilter' to: [/]

2019-10-06 22:02:50.645 INFO 14800 --- [ost-startStop-1]
o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'hiddenHttpMethodFilter' to: [/]

2019-10-06 22:02:50.645 INFO 14800 --- [ost-startStop-1]
o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'httpPutFormContentFilter' to: [/]

2019-10-06 22:02:50.645 INFO 14800 --- [ost-startStop-1]
o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'requestContextFilter' to: [/]

2019-10-06 22:02:51.217 INFO 14800 --- [      main]
s.w.s.m.m.a.RequestMappingHandlerAdapter : Looking for @ControllerAdvice:
org.springframework.boot.context.embedded.AnnotationConfigEmbeddedWebApplication
Context@44a3ec6b: startup date [Sun Oct 06 22:02:48 IST 2019]; root of context hierarchy

2019-10-06 22:02:51.387 INFO 14800 --- [      main]
s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/error]}" onto public
org.springframework.http.ResponseEntity<java.util.Map<java.lang.String,
java.lang.Object>>
org.springframework.boot.autoconfigure.web.BasicErrorController.error(javax.servlet.http.
HttpServletRequest)
```

CONTACT US:**Mobile: +91- 8885 25 26 27**Mail ID: durgasoftonline training@gmail.com**+91- 7207 21 24 27/28**WEBSITE: www.durgasoftonline.com**US NUM: 4433326786**

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,


```
2019-10-06 22:02:51.388 INFO 14800 --- [      main]
s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/error],produces=[text/html]}"
onto public org.springframework.web.servlet.ModelAndView
org.springframework.boot.autoconfigure.web.BasicErrorController.errorHtml(javax.servlet.
http.HttpServletRequest,javax.servlet.http.HttpServletResponse)

2019-10-06 22:02:51.511 INFO 14800 --- [      main]
o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/webjars/**] onto handler
of type [class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]

2019-10-06 22:02:51.512 INFO 14800 --- [      main]
o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/**] onto handler of type
[class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]

2019-10-06 22:02:51.618 INFO 14800 --- [      main]
o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/**/favicon.ico] onto
handler of type [class
org.springframework.web.servlet.resource.ResourceHttpRequestHandler]

2019-10-06 22:02:51.810 INFO 14800 --- [      main] o.s.j.e.a.AnnotationMBeanExporter
: Registering beans for JMX exposure on startup

2019-10-06 22:02:51.937 INFO 14800 --- [      main]
s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port(s): 1111 (http)

2019-10-06 22:02:51.942 INFO 14800 --- [      main] com.durgasoft.HelloWorld      :
Started HelloWorld in 4.047 seconds (JVM running for 4.509)

Hello From First Spring boot Application
```

CONTACT US:**Mobile: +91- 8885 25 26 27**Mail ID: durgasoftonlinetraining@gmail.com**+91- 7207 21 24 27/28**WEBSITE: www.durgasoftonline.com**US NUM: 4433326786**

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

Note: If we want to access the same application through Web Client then we have to prepare a Controller class , where we have to provide handler method with a particular return value.

EX:

HelloController.java:

```
package com.durgasoft.controller;
```

```
import org.springframework.boot.autoconfigure.EnableAutoConfiguration;
```

```
import org.springframework.web.bind.annotation.RequestMapping;
```

```
import org.springframework.web.bind.annotation.RestController;
```

```
@RestController
```

```
@EnableAutoConfiguration
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
public class HelloController {  
    @RequestMapping("/hello")  
    public String sayHello() {  
        return "Hello User!";  
    }  
}
```

Access the above controller class by using the following URL at client browser after executing HelloWorld standalone application.

<http://localhost:1111/hello>

Note: in URL, application name is not required to specify.

2.Steps to prepare Spring Boot Application by using STS[Spring Tool Suit] IDE:

- a)Download and instal STS in our System
- b)Create Spring Starter project
- c)

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

a)Download and install STS in our System:

- a)Open "https://spring.io/tools" in Browser.
- b)Click on "Spring Tool Suite for Eclipse "Windows 64 bit".
- c)Get "spring-tool-suite-4-4.4.0.RELEASE-e4.13.0-win32.win32.x86_64.zip" from downloads and unzip at C drive.
- d)Find "sts-4.4.0.RELEASE" folder at C driver and open "sts-4.4.0.RELEASE" and double click on "SpringToolSuite4" Application.
- e)Provide workspace "D:\springbootpractice" .

b)Create Spring Starter project:

- a)Select "File"--> "New"--> "Spring Starter Project".
- b)Provide the following details.
 - a)Name: helloapp

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

- b)Group : com.durgasoft
- c)Artifact: helloapp
- d)Description: HelloApp for Spring Boot
- e)package: com.durgasoft
- c)Click on "Next" button.
- d)Select the required modules like "web" --"Spring Web".
- e)Click on "Next".
- f)Click on "Finish" button.

If we do the above steps then we will get SpringBoot project at Project Explorer part.

- c)Open HelloApplication.java file which is existed at "src\main\java\com.durgasoft.HelloApplication.java" and write sop(-).

EX: HelloApplication.java:

```
package com.durgasoft;
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,



BY NAGOOR BABU

```
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class HelloappApplication {

    public static void main(String[] args) {
        SpringApplication.run(HelloappApplication.class, args);
        System.out.println("HelloApplication from Spring");
    }
}
```

d)Run "HelloApplication" by clicking on "Run" button then we will get Output.

~~Λ\ / _ _ _ _ () _ _ _ \ \ \~~

~~(() \ _ | ' _ | ' _ | ' _ V _ ` | \ \ \~~

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,



BY NAGOOR BABU

```
\V _)|_| | | | | |( |_| ))))
' |__| ._|_| |_|_| |_|_\,|////
=====|_|=====|___/=/_/_/_/
:: Spring Boot ::      (v2.1.9.RELEASE)
```

```
2019-10-07 10:43:43.483 INFO 16632 --- [      main] com.durgasoft.HelloappApplication
: Starting HelloappApplication on DESKTOP-LTICM7V with PID 16632
(D:\spring_boot_practice\sts\helloapp\target\classes started by Nagoor in
D:\spring_boot_practice\sts\helloapp)

2019-10-07 10:43:43.486 INFO 16632 --- [      main] com.durgasoft.HelloappApplication
: No active profile set, falling back to default profiles: default

2019-10-07 10:43:44.444 INFO 16632 --- [      main]
o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 2222 (http)

2019-10-07 10:43:44.475 INFO 16632 --- [      main]
o.apache.catalina.core.StandardService : Starting service [Tomcat]

2019-10-07 10:43:44.475 INFO 16632 --- [      main]
org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.26]

2019-10-07 10:43:44.603 INFO 16632 --- [      main] o.a.c.c.C.[Tomcat].[localhost].[/] :
Initializing Spring embedded WebApplicationContext

2019-10-07 10:43:44.603 INFO 16632 --- [      main] o.s.web.context.ContextLoader
: Root WebApplicationContext: initialization completed in 1065 ms
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
2019-10-07 10:43:44.796 INFO 16632 --- [      main]
o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService
'applicationTaskExecutor'

2019-10-07 10:43:44.925 INFO 16632 --- [      main]
o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 2222 (http) with
context path ""

2019-10-07 10:43:44.928 INFO 16632 --- [      main] com.durgasoft.HelloappApplication
: Started HelloappApplication in 1.831 seconds (JVM running for 2.894)
```

HelloApplication from Spring

Note: If 8080 port is busy in our system then change server port number by using key-value pair "server.port=2222" in application.properties which is available under "src\main\resources\application.properties".

e)If we want to execute through Web Client then prepare controller class with handler method.

EX: HelloController.java

```
package com.durgasoft.controller;
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController

public class HelloController {

    @RequestMapping("/hello")

    public String sayHello() {

        return "Hello User!";

    }

}
```

f)Execute HelloApplication application and open browser and use the following URL at address bar.

<http://localhost:2222/hello>

3. Spring Boot Initializr:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

-->Spring Boot initializr is a web tool , it will provide option to generate Spring boot projects directly as per the provided project details.

-->Spring Boot Initializr will give an option to download the project directly from web and to execute through Eclipse IDE or some other.

Steps:

1. Create Spring Boot Project and Download project:

2. Import Spring boot Application in STS:

1. Create Spring Boot Project and Download project:

a)Open "https://start.spring.io" URL at address bar in browser.

b)Provide the following details:

a)project: Maven Project

b)Language : JAVA

c)Spring Boot: 2.1.9

d)Project Metadata :

group: com.durgasoft

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

artifact: helloapp

Options:

Name: helloapp

Description: Helloapp for Spring Boot

Packaging : jar

JAVA: 8

Dependencies: web --> Spring Web

c)Click on "Generate" button.

If we do the above step then helloapp.zip file is generated and downloaded.

2. Import Spring boot Application in STS:

a)Copy helloapp.zip" file to "D:\mywebapps" location from downloads.

b)Unzip "helloapp.zip" file.

c)Open STS and import the project as existed Maven project:

1)Select "File" --> "import" --> "Maven" --> "Existed Maven Projects".

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

2)Click on "Next" button

3)provide Root Directory "D:\mywebapps".

4)Click on "Finish" button.

d)Open "com.durgasoft.HelloappApplication.java" from src\main\java and do required modifications.

EX: HelloappApplication.java

```
package com.durgasoft;
```

```
import org.springframework.boot.SpringApplication;
```

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

```
@SpringBootApplication
```

```
public class HelloappApplication {
```

```
    public static void main(String[] args) {
```

```
        SpringApplication.run(HelloappApplication.class, args);
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
}
```

```
}
```

Note: If 8080 port is busy in our system then change port number from 8080 to some other by providing "server.port" in application.properties file which is existed in "src\Main\resources\application.properties".

e)Execute HelloappApplication.java file by clicking on "Run" button.

Output:

```
.  _ _ _ _ _
/\ / _ ' _ _ _ _ _ \ \ \
(( \ _ | ' | ' | ' | ' V _ ' | \ \ \
\ W _ ) | | ) | | | | | ( | | ) ) )
' | _ | . _ | | | | | | \ _ | / / / /
=====|_|=====|_|/=///
:: Spring Boot ::      (v2.1.9.RELEASE)
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
2019-10-07 11:37:04.769 INFO 3492 --- [      main] com.durgasoft.HelloappApplication
: Starting HelloappApplication on DESKTOP-LTICM7V with PID 3492
(D:\mywebapps\helloapp\target\classes started by Nagoor in D:\mywebapps\helloapp)
2019-10-07 11:37:04.771 INFO 3492 --- [      main] com.durgasoft.HelloappApplication
: No active profile set, falling back to default profiles: default
2019-10-07 11:37:05.744 INFO 3492 --- [      main]
o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 5555 (http)
2019-10-07 11:37:05.769 INFO 3492 --- [      main]
o.apache.catalina.core.StandardService : Starting service [Tomcat]
2019-10-07 11:37:05.769 INFO 3492 --- [      main]
org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.26]
2019-10-07 11:37:05.876 INFO 3492 --- [      main] o.a.c.c.C.[Tomcat].[localhost].[/] :
Initializing Spring embedded WebApplicationContext
2019-10-07 11:37:05.876 INFO 3492 --- [      main] o.s.web.context.ContextLoader
: Root WebApplicationContext: initialization completed in 1063 ms
2019-10-07 11:37:06.098 INFO 3492 --- [      main]
o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService
'applicationTaskExecutor'
2019-10-07 11:37:06.279 INFO 3492 --- [      main]
o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 5555 (http) with
context path ""
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
2019-10-07 11:37:06.283 INFO 3492 --- [      main] com.durgasoft.HelloappApplication
: Started HelloappApplication in 1.89 seconds (JVM running for 2.8)
```

HelloApp from Spring Boot initializer

e)If we want to prepare Spring Boot application through Web Client then prepare Controller class with handler method and execute that:

EX: HelloController.java

```
-----
package com.durgasoft.controller;

import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HelloController {

    @RequestMapping("/hello")
    public String sayhello() {
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
        return "<h1>Hello User!</h1>";  
    }  
}
```

To run this , first we have to execute "HelloappApplication" then open browser and use the following URL.

`http://localhost:5555/hello`

4. Spring Boot CLI[Command Line Interface]:

--> Spring Boot CLI(Command Line Interface) is a Spring Boot software to run and test Spring Boot applications from command prompt. When we run Spring Boot applications using CLI, then it internally uses Spring Boot Starter and Spring Boot AutoConfigure components to resolve all dependencies and execute the application

--> Spring Boot internally contains Groovy and Grape (JAR Dependency Manager) to add Spring Boot Defaults and resolve all dependencies automatically.

--> To use Spring Boot CLI we have to use the following steps.

1. Download and Install Spring Boot CLI.

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

2. Prepare Groovy Script Code.

3. Run Groovy Script code through Spring Boot CLI.

1. Download and Install Spring Boot CLI:

a) Open the link in Browser address bar "<https://docs.spring.io/spring-boot/docs/current/reference/html/getting-started-installing-spring-boot.html#getting-started-installing-the-cli>". or else search for it in Google.

b) click on "spring-boot-cli-2.1.9.RELEASE-bin.zip" link under Manual Installation.

c) Copy the downloaded "spring-boot-cli-2.1.9.RELEASE-bin.zip" in C drive and extract it then we will find "spring-2.1.9.RELEASE" folder in C Driver.

d) Set "path" environment variable to "C:\spring-2.1.9.RELEASE\bin" location.

```
set path=%path%;C:\spring-2.1.9.RELEASE\bin;
```

e) Check whether Spring command is running or not in Command prompt.

Spring Boot CLI commands:

1. `spring --version` --> To get Spring CLI version.

2. `spring --help` --> To get help for Spring CLI.

3. `spring run Groovy_Script_File_Name.groovy` --> to run Groovy Script

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

2. Prepare Groovy Script Code:

It is almost all same as JAVA code only, but, the file extension should be .groovy.

EX: D:\spring_boot_practice\cli\HelloController.groovy

@RestController

```
public class HelloController{  
    @RequestMapping("/hello")  
    public String sayhello(){  
        return "Hello User!";  
    }  
}
```

3. If 8080 is busy then change tomcat port number from 8080 to 1234 by providing application.properties file at the same location:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

D:\spring_boot_practice\cli\application.properties

server.port=1234

4. Run Groovy Script code through Spring Boot CLI.

1)Run Groovy Script code by using "spring run" command.

D:\spring_boot_practice\cli>spring run HelloController.groovy

2)Use the following URL at Browser:

http://localhost:1234/hello

Postman:

Postman is a web tool, it can be used to send different types of request like GET, POST, PUT, OPTIONS,....., but, it required HttpServer environment.

Steps:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

1.Download and install Postman tool.

2.Open Postman tool and send request to Spring boot Application which is running at present.

1.Download and install Postman tool:

a)Open "https://www.getpostman.com/downloads/" link in browser Addressbar.

b)Click on Download button and select Windows 64 bit

With the above Postman tool will download with the setup file "Postman-win64-7.8.0-Setup.exe.

c)Execute Postman setup file by double clicking on "Postman-win64-7.8.0-Setup.exe".

d)Provide signin details or click on Open with google Account.

With the above Postman tool will open.

2.Send request to Spring boot Application which is running at present:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

1. Enter Application URL
2. Select the request type like GET or POST,...
3. Click on Send button and check the response in Body part.

EX:

application.properties

server.port=1234

HelloappApplication.java

package com.durgasoft;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

@SpringBootApplication

```
public class HelloappApplication {  
  
    public static void main(String[] args) {  
        SpringApplication.run(HelloappApplication.class, args);  
    }  
}
```

HelloController.java

```
package com.durgasoft.controller;  
  
import org.springframework.web.bind.annotation.RequestMapping;  
import org.springframework.web.bind.annotation.RequestMethod;  
import org.springframework.web.bind.annotation.RestController;
```

@RestController

```
public class HelloController {
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
@RequestMapping(value="/hello",method = RequestMethod.GET )
```

```
public String sayHello() {
```

```
    return "Hello User!";
```

```
}
```

```
@RequestMapping(value="/hello",method = RequestMethod.POST )
```

```
public String sayWelcome() {
```

```
    return "Welcome User!";
```

```
}
```

```
}
```

pom.xml:

```
1. <?xml version="1.0" encoding="UTF-8"?>
2. <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001
   /XMLSchema-instance"
3.     xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/x
   sd/maven-4.0.0.xsd">
4.     <modelVersion>4.0.0</modelVersion>
5.     <parent>
6.         <groupId>org.springframework.boot</groupId>
7.         <artifactId>spring-boot-starter-parent</artifactId>
8.         <version>2.1.9.RELEASE</version>
9.         <relativePath/> <!-- lookup parent from repository -->
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
10. </parent>
11. <groupId>com.durgasoft</groupId>
12. <artifactId>helloapp</artifactId>
13. <version>0.0.1-SNAPSHOT</version>
14. <name>helloapp</name>
15. <description>helloapp for Spring Boot</description>
16.
17. <properties>
18.     <java.version>1.8</java.version>
19. </properties>
20.
21. <dependencies>
22.     <dependency>
23.         <groupId>org.springframework.boot</groupId>
24.         <artifactId>spring-boot-starter-web</artifactId>
25.     </dependency>
26.
27.     <dependency>
28.         <groupId>org.springframework.boot</groupId>
29.         <artifactId>spring-boot-starter-test</artifactId>
30.         <scope>test</scope>
31.     </dependency>
32. </dependencies>
33.
34. <build>
35.     <plugins>
36.         <plugin>
37.             <groupId>org.springframework.boot</groupId>
38.             <artifactId>spring-boot-maven-plugin</artifactId>
39.         </plugin>
40.     </plugins>
41. </build>
42. </project>
```

In Postman.

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

If we send GET request with the url "http://localhost:1234/hello" then we will get "Hello User!" response.

If we send POST request with the url "http://localhost:1234/hello" then we will get "Welcome User!" response.

How to use User forms in Spring boot Application:

1. Prepare Spring starter project in STS.
2. Prepare "webapp/WEB-INF" folders under "src/main" folders to provide JSP and html pages.
3. Add "tomcat-jasper" dependency in pom.xml inorder to support for JSP.
4. Prepare HTML, JSP pages under "WEB-INF" folder.
5. Prepare Controller classes as per Spring WEB MVC and as per the requirement.
6. Provide server.port=1234, spring.mvc.view.prefix=/WEB-INF/, spring.mvc.view.suffix=.jsp key value pairs in application.properties file.
7. Access SpringBoot application as standalone application.
8. Open browser and send request.

EX:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,


```
1. helloapp
2. |---/src/main/java
3. |   |---com.durgasoft
4. |       |---HelloappApplication.java
5. |---com.durgasoft.controller
6. |       |HelloController.java
7. |---/src/main/resources
8. |       |---static
9. |       |---templates
10. |           |---application.properties
11. |---/src/test/java
12. |---src
13. |   |---main
14. |       |---webapp
15. |           |---WEB-INF
16. |               |---hellopage.jsp
17. |               |---wishpage.jsp
18. |---/src/test
19. |---target
20. |---HELP.md
21. |---mvnw
22. |---mvnw.cmd
23. |---pom.xml
```

hellopage.jsp

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.   pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html>
4. <html>
5. <head>
6. <meta charset="ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
9. <body>
10.     <h2 style="color: red;" align="center">Durga Software Solutions</h2>
11.     <h3 style="color: blue;" align="center">User Hello Form</h3>
12.     <form method="post" action="wish">
13.         <table align="center">
14.             <tr>
15.                 <td>User Name</td>
16.                 <td><input type="text" name="username"></td>
17.             </tr>
18.             <tr>
19.                 <td><input type="submit" value="SayHello"/></td>
20.             </tr>
21.         </table>
22.     </form>
23. </body>
24. </html>
```

wishpage.jsp:

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html>
4. <html>
5. <head>
6. <meta charset="ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <h1 style="color: red;" align="center">Hello ${username}!</h1>
11. <h2 align="center">
12. <a href="hellopage">|User Hello Page|</a>
13. </h2>
14. </body>
15. </html>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

HelloController.java:

```
package com.durgasoft.controller;
```

```
1. import org.springframework.stereotype.Controller;
2. import org.springframework.ui.ModelMap;
3. import org.springframework.web.bind.annotation.RequestMapping;
4. import org.springframework.web.bind.annotation.RequestMethod;
5. import org.springframework.web.bind.annotation.RequestParam;
6.
7. @Controller
8. public class HelloController {
9.
10.     @RequestMapping(value="/hellopage",method = RequestMethod.GET )
11.     public String sayHello() {
12.         return "hellopage";
13.     }
14.
15.     @RequestMapping(value="/wish",method = RequestMethod.POST )
16.     public String wish(@RequestParam String username, ModelMap map) {
17.         map.addAttribute("username", username);
18.         return "wishpage";
19.     }
20. }
```

HelloappApplication.java:

```
package com.durgasoft;
```

```
import org.springframework.boot.SpringApplication;
```

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
@SpringBootApplication
```

```
public class HelloappApplication {
```

```
    public static void main(String[] args) {
```

```
        SpringApplication.run(HelloappApplication.class, args);
```

```
    }
```

```
}
```

Application.properties:

server.port=1234

spring.mvc.view.prefix = /WEB-INF/

spring.mvc.view.suffix = .jsp

pom.xml:

```
1. <?xml version="1.0" encoding="UTF-8"?>
2. <project xmlns="http://maven.apache.org/POM/4.0.0"
3.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
4.      xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/x
sd/maven-4.0.0.xsd">
5.      <modelVersion>4.0.0</modelVersion>
6.      <parent>
7.          <groupId>org.springframework.boot</groupId>
8.          <artifactId>spring-boot-starter-parent</artifactId>
9.          <version>2.1.9.RELEASE</version>
10.         <relativePath /> <!-- lookup parent from repository -->
11.     </parent>
12.     <groupId>com.durgasoft</groupId>
13.     <artifactId>helloapp</artifactId>
14.     <version>0.0.1-SNAPSHOT</version>
15.     <name>helloapp</name>
16.     <description>helloapp for Spring Boot</description>
17.
18.     <properties>
19.         <java.version>1.8</java.version>
20.     </properties>
21.
22.     <dependencies>
23.         <dependency>
24.             <groupId>org.springframework.boot</groupId>
25.             <artifactId>spring-boot-starter-web</artifactId>
26.         </dependency>
27.
28.
29.         <dependency>
30.             <groupId>org.apache.tomcat</groupId>
31.             <artifactId>tomcat-jasper</artifactId>
32.             <version>9.0.26</version>
33.         </dependency>
34.
35.         <dependency>
36.             <groupId>org.springframework.boot</groupId>
37.             <artifactId>spring-boot-starter-test</artifactId>
38.             <scope>test</scope>
39.         </dependency>
40.     </dependencies>
41.
42.     <build>
43.         <plugins>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
44.         <plugin>
45.             <groupId>org.springframework.boot</groupId>
46.             <artifactId>spring-boot-maven-plugin</artifactId>
47.         </plugin>
48.     </plugins>
49. </build>
50.</project>
```

EX:

loginapp

```
|---/src/main/java
|   |---com.durgasoft
|       |----LoginappApplication.java
|---com.durgasoft.controller
|       |LoginController.java
|---/src/main/resources
|   |---static
|   |---templates
|   |---application.properties
|---/src/test/java
|---src
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
| |---main
| |----webapp
| |----WEB-INF
| |    ---failure.jsp
| |    ---loginform.jsp
| |    ---success.jsp
|---/src/test
|---target
|---HELP.md
|---mvnw
|---mvnw.cmd
|---pom.xml
```

loginform.jsp:

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.    pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html>
4. <html>
5. <head>
6. <meta charset="ISO-8859-1">
```

CONTACT US:**Mobile: +91- 8885 25 26 27**Mail ID: durgasoftonline@training@gmail.com**+91- 7207 21 24 27/28**WEBSITE: www.durgasoftonline.com**US NUM: 4433326786**

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
7. <title>Insert title here</title>
8. </head>
9. <body>
10.     <h2 style="color: red;" align="center">Durga Software Solutions</h2>
11.     <h3 style="color: blue;" align="center">User Login Form</h3>
12.     <form method="post" action="/login">
13.         <table align="center">
14.             <tr>
15.                 <td>User Name</td>
16.                 <td><input type="text" name="uname"></td>
17.             </tr>
18.             <tr>
19.                 <td>Password</td>
20.                 <td><input type="password" name="upwd"></td>
21.             </tr>
22.             <tr>
23.                 <td><input type="submit" value="Login"/></td>
24.             </tr>
25.         </table>
26.     </form>
27. </body>
28. </html>
```

success.jsp

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html>
4. <html>
5. <head>
6. <meta charset="ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <h1 style="color: red;" align="center">Login Success</h1>
11. <h2 align="center">
12. <a href="loginpage">|Login Page|</a>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
13. </h2>
14. </body>
15. </html>
```

failure.jsp

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html>
4. <html>
5. <head>
6. <meta charset="ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <h1 style="color: red;" align="center">Login Success</h1>
11. <h2 align="center">
12. <a href="loginpage">|Login Page|</a>
13. </h2>
14. </body>
15. </html>
```

LoginController.java

```
1. package com.durgasoft.controller;
2.
3. import org.springframework.stereotype.Controller;
4. import org.springframework.web.bind.annotation.RequestMapping;
5. import org.springframework.web.bind.annotation.RequestMethod;
6. import org.springframework.web.bind.annotation.RequestParam;
7.
8. @Controller
9. public class LoginController {
10.
11.     @RequestMapping(value="/loginpage", method = RequestMethod.GET)
12.     public String showLoginPage() {
13.         return "loginform";
14.     }
15. }
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
14.     }
15.
16.     @RequestMapping(value="/login", method = RequestMethod.POST)
17.     public String login(@RequestParam String uname, @RequestParam String upwd) {
18.         String status = "";
19.         if(uname.equals("durga") && upwd.equals("durga")) {
20.             status = "success";
21.         }else {
22.             status = "failure";
23.         }
24.         return status;
25.     }
26. }
```

LoginappApplication.java

```
package com.durgasoft;
```

```
1. import org.springframework.boot.SpringApplication;
2. import org.springframework.boot.autoconfigure.SpringBootApplication;
3.
4. @SpringBootApplication
5. public class LoginappApplication {
6.
7.     public static void main(String[] args) {
8.         SpringApplication.run(LoginappApplication.class, args);
9.     }
10.
11. }
```

Application.properties

```
server.port = 1234
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

spring.mvc.view.prefix=/WEB-INF/

spring.mvc.view.suffix=.jsp

pom.xml

```
1. <?xml version="1.0" encoding="UTF-8"?>
2. <project xmlns="http://maven.apache.org/POM/4.0.0"
3.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4.     xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
5.     <modelVersion>4.0.0</modelVersion>
6.     <parent>
7.         <groupId>org.springframework.boot</groupId>
8.         <artifactId>spring-boot-starter-parent</artifactId>
9.         <version>2.1.9.RELEASE</version>
10.        <relativePath /> <!-- lookup parent from repository -->
11.    </parent>
12.    <groupId>com.durgasoft</groupId>
13.    <artifactId>loginapp</artifactId>
14.    <version>0.0.1-SNAPSHOT</version>
15.    <name>loginapp</name>
16.    <description>Spring boot Login app</description>
17.
18.    <properties>
19.        <java.version>1.8</java.version>
20.    </properties>
21.
22.    <dependencies>
23.        <dependency>
24.            <groupId>org.springframework.boot</groupId>
25.            <artifactId>spring-boot-starter-web</artifactId>
26.        </dependency>
27.
28.
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
29.     <dependency>
30.         <groupId>org.apache.tomcat</groupId>
31.         <artifactId>tomcat-jasper</artifactId>
32.         <version>9.0.26</version>
33.     </dependency>
34.
35.
36.     <dependency>
37.         <groupId>org.springframework.boot</groupId>
38.         <artifactId>spring-boot-starter-test</artifactId>
39.         <scope>test</scope>
40.     </dependency>
41. </dependencies>
42.
43. <build>
44.     <plugins>
45.         <plugin>
46.             <groupId>org.springframework.boot</groupId>
47.             <artifactId>spring-boot-maven-plugin</artifactId>
48.         </plugin>
49.     </plugins>
50. </build>
51.
52. </project>
```

First Execute LoginappApplication as standalone application then open browser and access the application by using the following URL.

<http://localhost:1234/loginpage>.

Spring Boot Application with Spring JDBC Application:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

To prepare Spring Boot Application with Spring-JDBC , we have to use the following Steps.

1. Install Maven in our System:

a)Download Maven software from "<http://maven.apache.org/download.cgi>" url.

b)Unzip "apache-maven-3.6.2-bin.zip" file.

c)Set "path" environment variable to "C:\apache-maven-3.6.2\bin" either temporarily or permanently.

```
set path=C:\apache-maven-3.6.2\bin;
```

2. Prepare ojdbc6 dependency manually:

Oracle Driver is a driver provided Oracle , it is licenced, it is not possible to get dependency details manually, so we must prepare ojdbc6.jar dependency.

To prepare ojdbc6 dependency manually we have to use the following command on Command prompt.

```
D:\>mvn install:install-file -
Dfile=C:\oracle\app\oracle\product\11.2.0\server\jdbc\lib\ojdbc6.jar -
DgroupId=com.oracle - DartifactId=ojdbc6 -Dversion=11.2.0 -Dpackaging=jar
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

For ojdbc6.jar file, we have to use the following dependency.

groupid: com.oracle

artifactid: ojdbc6

version:11.2.0

3. Prepare Spring boot Starter project in STS with Spring Web and JDBC API in SQL part.

4. Open pom.xml and add "ojdbc6" and tomcat-jasper:

Where ojdbc6.jar file is to interact with Oracle Database.

Where tomcat jasper is to provide JSPs support.

```
1. <project>
2.   ----
3.   <dependencies>
4.     <dependency>
5.       <groupId>com.oracle</groupId>
6.       <artifactId>ojdbc6</artifactId>
7.       <version>11.2.0</version>
8.     </dependency>
9.     <dependency>
10.      <groupId>org.apache.tomcat</groupId>
11.      <artifactId>tomcat-jasper</artifactId>
12.      <version>9.0.26</version>
13.    </dependency>
14.    -----
15.  </dependencies>
16.  ----
17. </project>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

5. Provide the following details in application.properties file:

```
server.port=1234  
spring.mvc.view.prefix=/WEB-INF/  
spring.mvc.view.suffix=.jsp  
spring.datasource.driver-class-name=oracle.jdbc.OracleDriver  
spring.datasource.url=jdbc:oracle:thin:@localhost:1521:xe  
spring.datasource.username=system  
spring.datasource.password=durga
```

Note: datasource name is not required, because, Spring boot is using Apache's "Hikary Connection pooling Mechanism" by default.

6. Prepare "webapp/WEB-INF" folder under "src/main" folder.

7. Prepare all jsp pages under "src/main/webapp/WEB-INF" folder.

8. Prepare all Java Resources as per the structer.

EX:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

application.properties:

```
-----  
server.port=1234  
spring.mvc.view.prefix=/WEB-INF/  
spring.mvc.view.suffix=.jsp  
spring.datasource.driver-class-name=oracle.jdbc.OracleDriver  
spring.datasource.url=jdbc:oracle:thin:@localhost:1521:xe  
spring.datasource.username=system  
spring.datasource.password=durga
```

welcome.jsp:

```
-----  
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"  
2.     pageEncoding="ISO-8859-1"%>  
3. <!DOCTYPE html PUBLIC "-  
   //W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">  
4. <html>  
5. <head>  
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">  
7. <title>Insert title here</title>  
8. </head>  
9. <body>  
10. <h1 style="color: red;" align="center">Durga Software Solutions</h1>  
11. <hr>  
12. <table align="center" width="100%" height="100%">  
13. <tr>  
14.     <td align="center">
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com

+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com

US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
15.      <a href="addform">ADD Student</a>
16.      <a href="searchform">SEARCH Student</a>
17.      <a href="updateform">UPDATE Student</a>
18.      <a href="deleteform">DELETE Student</a>
19.    </td>
20.  </tr>
21. </table>
22. </body>
23. </html>
```

Addform.jsp

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.   pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <jsp:include page="/WEB-INF/welcome.jsp"/>
11.
12. <br><br><br><br><br>
13. <form method="post" action="add">
14. <table align="center">
15. <tr>
16.   <td>Student Id</td>
17.   <td><input type="text" name="sid"></td>
18. </tr>
19. <tr>
20.   <td>Student Name</td>
21.   <td><input type="text" name="sname"></td>
22. </tr>
23. <tr>
24.   <td>Student Address</td>
25.   <td><input type="text" name="saddr"></td>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
26.</tr>
27.<tr>
28.    <td><input type="submit" value="ADD"/></td>
29.</tr>
30.</table>
31.</form>
32.</body>
33.</html>
```

searchform.jsp:

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.    pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <jsp:include page="/WEB-INF/welcome.jsp"/>
11.
12. <br><br><br><br><br>
13. <form method="post" action="search">
14. <table align="center">
15. <tr>
16.    <td>Student Id</td>
17.    <td><input type="text" name="sid"></td>
18. </tr>
19.
20. <tr>
21.    <td><input type="submit" value="SEARCH"/></td>
22. </tr>
23. </table>
24. </form>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
25. </body>
26. </html>
```

updateform.jsp:

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <jsp:include page="/WEB-INF/welcome.jsp"/>
11. <br><br><br><br><br>
12. <form method="post" action="editform">
13. <table align="center">
14. <tr>
15.     <td>Student Id</td>
16.     <td><input type="text" name="sid"></td>
17. </tr>
18.
19. <tr>
20.     <td><input type="submit" value="GetEditForm"/></td>
21. </tr>
22. </table>
23. </form>
24. </body>
25. </html>
```

Editform.jsp

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html PUBLIC "-
   //W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10.<jsp:include page="/WEB-INF/welcome.jsp"/>
11.<br><br><br><br><br>
12.<form method="post" action="update">
13.<table align="center">
14.<tr>
15.     <td>Student Id</td>
16.     <td>${std.sid}</td>
17.</tr>
18.<tr>
19.     <td>Student Name</td>
20.     <td><input type="text" name="sname" value="${std.sname}"></td>
21.</tr>
22.<tr>
23.     <td>Student Address</td>
24.     <td><input type="text" name="saddr" value="${std.saddr}"></td>
25.</tr>
26.<tr>
27.     <td><input type="submit" value="Update"></td>
28.</tr>
29.
30.</table>
31.<input type="hidden" name="sid" value="${std.sid}">
32.</form>
33.</body>
34.</html>
```

Deleteform.jsp:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <jsp:include page="/WEB-INF/welcome.jsp"/>
11.
12. <br><br><br><br><br>
13. <form method="post" action="delete">
14. <table align="center">
15. <tr>
16.     <td>Student Id</td>
17.     <td><input type="text" name="sid"></td>
18. </tr>
19.
20. <tr>
21.     <td><input type="submit" value="DELETE"/></td>
22. </tr>
23. </table>
24. </form>
25. </body>
26. </html>
```

studentdetails.jsp:

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7. <title>Insert title here</title>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
8. </head>
9. <body>
10. <jsp:include page="/WEB-INF/welcome.jsp"/>
11. <br><br><br><br><br>
12. <table align="center" border="1">
13. <tr>
14.     <td>Student Id</td>
15.     <td>${std.sid}</td>
16. </tr>
17. <tr>
18.     <td>Student Name</td>
19.     <td>${std.sname}</td>
20. </tr>
21. <tr>
22.     <td>Student Address</td>
23.     <td>${std.saddr}</td>
24. </tr>
25. </table>
26. </body>
27. </html>
```

success.jsp:

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <jsp:include page="/WEB-INF/welcome.jsp"/>
11. <br><br><br><br><br>
12. <h1 style="color: red;" align="center">${operation} Success</h1>
13. </body>
14. </html>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

failure.jsp

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <jsp:include page="/WEB-INF/welcome.jsp"/>
11. <br><br><br><br><br><br>
12. <h1 style="color: red;" align="center">${operation} Failure</h1>
13. </body>
14. </html>
```

existed.jsp

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <jsp:include page="/WEB-INF/welcome.jsp"/>
11. <br><br><br><br><br><br>
12. <h1 style="color: red;" align="center">Student Existed Already</h1>
13. </body>
14. </html>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

Notexisted.jsp

```
1. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2.     pageEncoding="ISO-8859-1"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
7. <title>Insert title here</title>
8. </head>
9. <body>
10. <jsp:include page="/WEB-INF/welcome.jsp"/>
11. <br><br><br><br><br><br>
12. <h1 style="color: red;" align="center">Student Not Existed</h1>
13. </body>
14. </html>
```

StudentController.java:

```
1. package com.durgasoft.controller;
2.
3. import org.springframework.beans.factory.annotation.Autowired;
4. import org.springframework.stereotype.Controller;
5. import org.springframework.ui.ModelMap;
6. import org.springframework.web.bind.annotation.RequestMapping;
7. import org.springframework.web.bind.annotation.RequestMethod;
8. import org.springframework.web.bind.annotation.RequestParam;
9.
10. import com.durgasoft.dto.Student;
11. import com.durgasoft.service.StudentService;
12.
13. @Controller
14. public class StudentController {
15.
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
16. @Autowired
17. StudentService studentService;
18.
19. @RequestMapping("/welcomepage")
20. public String showWelcomePage() {
21.     return "welcome";
22. }
23. @RequestMapping("/addform")
24. public String showAddPage() {
25.     return "addform";
26. }
27. @RequestMapping("/searchform")
28. public String showSearchPage() {
29.     return "searchform";
30. }
31. @RequestMapping("/updateform")
32. public String showUpdatePage() {
33.     return "updateform";
34. }
35. @RequestMapping("/editformform")
36. public String showEditPage() {
37.     return "editform";
38. }
39. @RequestMapping("/deleteform")
40. public String showDeletePage() {
41.     return "deleteform";
42. }
43.
44. @RequestMapping(value="/add", method = RequestMethod.POST)
45. public String add(ModelMap map, @RequestParam String sid, @RequestParam String s
    name,@RequestParam String saddr) {
46.     map.addAttribute("operation", "Student Insertion");
47.     Student std = new Student();
48.     std.setSid(sid);
49.     std.setName(sname);
50.     std.setSaddr(saddr);
51.
52.     String status = studentService.addStudent(std);
53.
54.     return status;
55. }
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
56.
57.     @RequestMapping(value="/search", method=RequestMethod.POST)
58.     public String search(ModelMap map, @RequestParam String sid) {
59.         String status = "";
60.         Student std = studentService.searchStudent(sid);
61.         if(std == null) {
62.             status = "notexisted";
63.         }else {
64.             map.addAttribute("std", std);
65.             status = "studentdetails";
66.         }
67.         return status;
68.     }
69.
70.     @RequestMapping(value="editform", method = RequestMethod.POST)
71.     public String getEditForm(ModelMap map, @RequestParam String sid) {
72.         String status = "";
73.         Student std = studentService.searchStudent(sid);
74.         if(std == null) {
75.             status = "notexisted";
76.         }else {
77.             map.addAttribute("std", std);
78.             status="editform";
79.         }
80.         return status;
81.     }
82.
83.     @RequestMapping(value="update", method = RequestMethod.POST)
84.     public String update(ModelMap map,@RequestParam String sid, @RequestParam String
sname, @RequestParam String saddr) {
85.         String status = "";
86.         map.addAttribute("operation", "Student Updation");
87.         Student std = new Student();
88.         std.setSid(sid);
89.         std.setSname(sname);
90.         std.setSaddr(saddr);
91.
92.         status = studentService.updateStudent(std);
93.         return status;
94.     }
95.
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
96. @RequestMapping(value = "delete", method = RequestMethod.POST)
97. public String delete(ModelMap map, @RequestParam String sid) {
98.     map.addAttribute("operation", "Student Deletion");
99.     String status = studentService.deleteStudent(sid);
100.     return status;
101. }
102. }
```

StudentService.java:

```
1. package com.durgasoft.service;
2.
3. import com.durgasoft.dto.Student;
4.
5. public interface StudentService {
6.     public String addStudent(Student std);
7.     public Student searchStudent(String sid);
8.     public String updateStudent(Student std);
9.     public String deleteStudent(String sid);
10. }
```

StudentServiceImpl.java:

```
1. package com.durgasoft.service;
2.
3. import org.springframework.beans.factory.annotation.Autowired;
4. import org.springframework.stereotype.Service;
5.
6. import com.durgasoft.dao.StudentDao;
7. import com.durgasoft.dto.Student;
8.
9. @Service
10. public class StudentServiceImpl implements StudentService {
11.
12.     @Autowired
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
13. StudentDao studentDao;
14. @Override
15. public String addStudent(Student std) {
16.     String status = studentDao.add(std);
17.     return status;
18. }
19.
20. @Override
21. public Student searchStudent(String sid) {
22.     Student std = studentDao.search(sid);
23.     return std;
24. }
25.
26. @Override
27. public String updateStudent(Student std) {
28.     String status = studentDao.update(std);
29.     return status;
30. }
31.
32. @Override
33. public String deleteStudent(String sid) {
34.     String status = studentDao.delete(sid);
35.     return status;
36. }
37.
38. }
```

StudentDao.java

```
package com.durgasoft.dao;
```

```
import com.durgasoft.dto.Student;
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
public interface StudentDao {  
    public String add(Student std);  
    public Student search(String sid);  
    public String update(Student std);  
    public String delete(String sid);  
}
```

StudentDaoImpl.java:

```
package com.durgasoft.dao;
```

```
1. import java.util.List;  
2.  
3. import org.springframework.beans.factory.annotation.Autowired;  
4. import org.springframework.jdbc.core.JdbcTemplate;  
5. import org.springframework.stereotype.Repository;  
6.  
7. import com.durgasoft.dto.Student;  
8.  
9. @Repository  
10. public class StudentDaoImpl implements StudentDao {  
11.  
12.     @Autowired  
13.     JdbcTemplate jdbcTemplate;
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
14.  @Override
15.  public String add(Student std) {
16.      String status = "";
17.      Student student = search(std.getSid());
18.      if(student == null) {
19.          int rowCount = jdbcTemplate.update("insert into student values('"+std.ge
tSid()+"', '"+std.getSname()+"', '"+std.getSaddr()+"'");
20.          if(rowCount == 1) {
21.              status = "success";
22.          }else {
23.              status = "failure";
24.          }
25.      }else {
26.          status = "existed";
27.      }
28.      return status;
29.  }
30.
31.  @Override
32.  public Student search(String sid) {
33.      Student std = null;
34.      List<Student> studentsList = jdbcTemplate.query("select * from student where
SID = '"+sid+"'", (rs, rowNum) ->{
35.          Student s = new Student();
36.          s.setSid(rs.getString("SID"));
37.          s.setSname(rs.getString("SNAME"));
38.          s.setSaddr(rs.getString("SADDR"));
39.          return s;
40.      });
41.      if(studentsList.isEmpty() == true) {
42.          std = null;
43.      }else {
44.          std = studentsList.get(0);
45.      }
46.      return std;
47.  }
48.
49.  @Override
50.  public String update(Student std) {
51.      String status = "";
```

CONTACT US:**Mobile: +91- 8885 25 26 27**Mail ID: durgasoftonlinetraining@gmail.com**+91- 7207 21 24 27/28**WEBSITE: www.durgasoftonline.com**US NUM: 4433326786**

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
52.         int rowCount = jdbcTemplate.update("update student set SNAME = '"+std.getSna
me()+"', SADDR = '"+std.getSaddr()+"' where SID = '"+std.getSid()+"'");
53.         if(rowCount == 1) {
54.             status = "success";
55.         }else {
56.             status = "failure";
57.         }
58.         return status;
59.     }
60.
61.     @Override
62.     public String delete(String sid) {
63.         String status = "";
64.         Student std = search(sid);
65.         if(std == null) {
66.             status = "notexisted";
67.         }else {
68.             int rowCount = jdbcTemplate.update("delete from student where SID = '"+s
id+"'");
69.             if(rowCount == 1) {
70.                 status = "success";
71.             }else {
72.                 status = "failure";
73.             }
74.         }
75.         return status;
76.     }
77. }
```

StudentApplication.java

```
package com.durgasoft;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

@SpringBootApplication

```
public class StudentappApplication {  
    public static void main(String[] args) {  
        SpringApplication.run(StudentappApplication.class, args);  
    }  
}
```

pom.xml:

```
1. <?xml version="1.0" encoding="UTF-8"?>  
2. <project xmlns="http://maven.apache.org/POM/4.0.0"  
3.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
4.     xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">  
5.     <modelVersion>4.0.0</modelVersion>  
6.     <parent>  
7.         <groupId>org.springframework.boot</groupId>  
8.         <artifactId>spring-boot-starter-parent</artifactId>  
9.         <version>2.1.9.RELEASE</version>  
10.        <relativePath /> <!-- lookup parent from repository -->  
11.    </parent>  
12.    <groupId>com.durgasoft</groupId>  
13.    <artifactId>employee</artifactId>  
14.    <version>0.0.1-SNAPSHOT</version>  
15.    <name>employeeapp</name>  
16.    <description>Spring boot employee app</description>  
17.  
18.    <properties>  
19.        <java.version>1.8</java.version>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
20. </properties>
21.
22. <dependencies>
23.   <dependency>
24.     <groupId>org.springframework.boot</groupId>
25.     <artifactId>spring-boot-starter-jdbc</artifactId>
26.   </dependency>
27.   <dependency>
28.     <groupId>com.oracle</groupId>
29.     <artifactId>ojdbc6</artifactId>
30.     <version>11.2.0</version>
31.   </dependency>
32.   <dependency>
33.     <groupId>org.apache.tomcat</groupId>
34.     <artifactId>tomcat-jasper</artifactId>
35.     <version>9.0.26</version>
36.   </dependency>
37.
38.   <dependency>
39.     <groupId>org.springframework.boot</groupId>
40.     <artifactId>spring-boot-starter-web</artifactId>
41.   </dependency>
42.
43.   <dependency>
44.     <groupId>org.springframework.boot</groupId>
45.     <artifactId>spring-boot-starter-test</artifactId>
46.     <scope>test</scope>
47.   </dependency>
48. </dependencies>
49.
50. <build>
51.   <plugins>
52.     <plugin>
53.       <groupId>org.springframework.boot</groupId>
54.       <artifactId>spring-boot-maven-plugin</artifactId>
55.     </plugin>
56.   </plugins>
57. </build>
58. </project>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

Connection Pooling In Spring Boot:

Spring Boot will follow an algorithm to find Connection pooling mechanism internally.

1. Spring boot will search for Hikary Connection Pooling if it is existed.
2. If Hikary Connection pooling is not existed then Spring boot will search for Tomcat Pooling DataSource.
3. If both Hikary and Tomcat Pooling DataSource are not existed then Spring boot will search for Commons DBCP2 mechanism if it is existed.

To check the above algorithm then we have to use the following Steps:

1. Prepare SpringBootApplication By extending CommandLineRunner and by providing run(String... args) method:

```
package com.durgasoft;  
  
import javax.sql.DataSource;  
  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.boot.CommandLineRunner;  
import org.springframework.boot.SpringApplication;
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.context.annotation.ComponentScan;

@SpringBootApplication
@ComponentScan
public class StudentappApplication implements CommandLineRunner{

    @Autowired
    private DataSource dataSource;

    public static void main(String[] args) {
        SpringApplication.run(StudentappApplication.class, args);
    }

    @Override
    public void run(String... args) throws Exception {
        System.out.println("Connection Pooling : "+dataSource);
    }
}
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

2. Apply a particular Connection pooling Mechanism:

a)case-1: only Hikary Connection Pooling, No Tomcat JDBC and No Apache DBCP2

In pom.xml

No changes , normal pom.xml.

By default, Spring boot will search for Hikary Connection pooling mechanism.

If we run the application then we will get the following output.

---Spring Boot logo----

Connection Pooling : HikariDataSource (HikariPool-1)

2)case-2:Only Tomcat Jdbc Connection pooling, No Hikary Connection Pooling and No Apache DBCP2 Connection pooling

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

In pom.xml: Exclude Hikary Connection Pooling in spring-boot-starter-jdbc and provide tomcat-jdbc dependency.

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-jdbc</artifactId>
    <exclusions>
      <exclusion>
        <groupId>com.zaxxer</groupId>
        <artifactId>HikariCP</artifactId>
      </exclusion>
    </exclusions>
  </dependency>
  <dependency>
    <groupId>org.apache.tomcat</groupId>
    <artifactId>tomcat-jdbc</artifactId>
    <version>9.0.10</version>
  </dependency>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
-----  
</dependencies>
```

If we run the application then we will get the following output.

```
---Spring Boot logo---
```

```
----
```

Connection Pooling : [org.apache.tomcat.jdbc.pool.DataSource@6438a7fe{----}](#)

3)Case-3: Only Apache DBCP2, No Hikary Connection Pooling and No Tomcat-Jdbc Connection pooling.

In pom.xml: Exclude "Hikary" and "Tomcat Jdbc" Connection Pooling mechanisms in "Spring-Boot-Starter-jdbc". and include Apache DBCP2 dependency.

EX:

```
----
```

```
<dependencies>
```

```
  <dependency>
```

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

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
<artifactId>spring-boot-starter-jdbc</artifactId>

<exclusions>

    <exclusion>

        <groupId>org.apache.tomcat</groupId>

        <artifactId>tomcat-jdbc</artifactId>

    </exclusion>

    <exclusion>

        <groupId>com.zaxxer</groupId>

        <artifactId>HikariCP</artifactId>

    </exclusion>

</exclusions>

</dependency>

<dependency>

    <groupId>org.apache.commons</groupId>

    <artifactId>commons-dbcp2</artifactId>

    <version>2.1</version>

</dependency>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
-----
</dependencies>

If we run the application then we will get the following output.

---Spring Boot logo---

-----
Connection Pooling : org.apache.commons.dbcp2.BasicDataSource@7c8f9c2e

For all the above cases we have to use the following content in application.properties file.
application.properties
-----
server.port=1234
spring.datasource.driver-class-name=oracle.jdbc.driver.OracleDriver
spring.datasource.url=jdbc:oracle:thin:@localhost:1521:xe
spring.datasource.username=system
spring.datasource.password=durga
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

If we want to use any other Connection Pooling Mechanisms like C3PO or proxool in Spring boot applications then we have to use the following steps.

1. Provide Connection pooling mechanisms dependency in pom.xml file.
2. Provide "spring.datasource.type" property in application.properties file.

EX: C3PO Connection Pooling Mechanisms:

1. Prepare c3p0.properties file under "src/main/resources"

c3p0.properties

```
c3p0.initialPoolSize = 10
c3p0.maxPoolSize = 10
c3p0.maxStatements = 20
c3p0.maxIdleTime = 20000
```

2. Provide "spring.datasource.type" property in application.properties file:
- application.properties

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
-----  
spring.datasource.type = com.mchange.v2.c3p0.ComboPooledDataSource  
-----  
-----
```

3. Add c3p0 dependency in pom.xml file and exclude Hikary in pom.xml:

pom.xml

```
-----  
<dependencies>  
  <dependency>  
    <groupId>org.springframework.boot</groupId>  
    <artifactId>spring-boot-starter-jdbc</artifactId>  
  
    <exclusions>  
      <exclusion>  
        <groupId>com.zaxxer</groupId>  
        <artifactId>HikariCP</artifactId>  
      </exclusion>  
    </exclusions>  
  </dependency>  
</dependencies>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
<dependency>  
  <groupId>c3p0</groupId>  
  <artifactId>c3p0</artifactId>  
  <version>0.9.1.2</version>  
</dependency>
```

```
-----  
</dependencies>
```

EX:

```
-----  
c3p0.properties
```

```
-----  
c3p0.initialPoolSize = 10  
c3p0.maxPoolSize = 10  
c3p0.maxStatements = 20  
c3p0.maxIdleTime = 20000
```

```
application.properties
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
-----  
server.port=1234  
spring.datasource.driver-class-name=oracle.jdbc.OracleDriver  
spring.datasource.url==jdbc:oracle:thin:@localhost:1521:xe  
spring.datasource.username=system  
spring.datasource.password=durga  
spring.datasource.type = com.mchange.v2.c3p0.ComboPooledDataSource
```

```
CustomConnectionPoolingApplication.java  
-----
```

```
package com.durgasoft;  
  
import javax.sql.DataSource;  
  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.boot.CommandLineRunner;  
import org.springframework.boot.SpringApplication;  
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

CONTACT US:**Mobile: +91- 8885 25 26 27**Mail ID: durgasoftonline training@gmail.com**+91- 7207 21 24 27/28**WEBSITE: www.durgasoftonline.com**US NUM: 4433326786****FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,**

BY NAGOOR BABU

@SpringBootApplication

```
public class CustomconnectionpoolingApplication implements CommandLineRunner {
```

```
    @Autowired
```

```
    private DataSource dataSource;
```

```
    public static void main(String[] args) {
```

```
        SpringApplication.run(CustomconnectionpoolingApplication.class, args);
```

```
    }
```

```
    @Override
```

```
    public void run(String... args) throws Exception {
```

```
        System.out.println("Connection Pooling Data Source : "+dataSource);
```

```
    }
```

```
}
```

pom.xml:

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
1. <?xml version="1.0" encoding="UTF-8"?>
2. <project xmlns="http://maven.apache.org/POM/4.0.0"
3.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4.     xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
5.     <modelVersion>4.0.0</modelVersion>
6.     <parent>
7.         <groupId>org.springframework.boot</groupId>
8.         <artifactId>spring-boot-starter-parent</artifactId>
9.         <version>2.1.9.RELEASE</version>
10.        <relativePath /> <!-- lookup parent from repository -->
11.    </parent>
12.    <groupId>com.durgasoft</groupId>
13.    <artifactId>customconnectionpooling</artifactId>
14.    <version>0.0.1-SNAPSHOT</version>
15.    <name>customconnectionpooling</name>
16.    <description>Spring boot application on Customized Connection Pooling</description>
17.
18.    <properties>
19.        <java.version>1.8</java.version>
20.    </properties>
21.
22.    <dependencies>
23.        <dependency>
24.            <groupId>org.springframework.boot</groupId>
25.            <artifactId>spring-boot-starter-jdbc</artifactId>
26.            <exclusions>
27.
28.                <!-- <exclusion>
29.                    <groupId>org.apache.tomcat</groupId>
30.                    <artifactId>tomcat-jdbc</artifactId>
31.                </exclusion> -->
32.            </exclusion>
33.            <groupId>com.zaxxer</groupId>
34.            <artifactId>HikariCP</artifactId>
35.        </exclusion>
36.    </exclusions>
37.    </dependency>
38.    <dependency>
39.        <groupId>com.oracle</groupId>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
40.         <artifactId>ojdbc6</artifactId>
41.         <version>11.2.0</version>
42.     </dependency>
43.
44.     <dependency>
45.         <groupId>c3p0</groupId>
46.         <artifactId>c3p0</artifactId>
47.         <version>0.9.1.2</version>
48.     </dependency>
49.     <dependency>
50.         <groupId>org.springframework.boot</groupId>
51.         <artifactId>spring-boot-starter-web</artifactId>
52.     </dependency>
53.
54.     <dependency>
55.         <groupId>org.springframework.boot</groupId>
56.         <artifactId>spring-boot-starter-test</artifactId>
57.         <scope>test</scope>
58.     </dependency>
59. </dependencies>
60.
61. <build>
62.     <plugins>
63.         <plugin>
64.             <groupId>org.springframework.boot</groupId>
65.             <artifactId>spring-boot-maven-plugin</artifactId>
66.         </plugin>
67.     </plugins>
68. </build>
69.
70.</project>
```

In Memory Databases in Spring Boot:

IN enterprise applications, if we want to use traditional Databases then we have to provide lot of configurations like

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

1. Install Database
2. Providing Database setups
3. Create Schema
4. Create table
5. Populating Data.
6. Creating dataSource to connect with database

To overcome the above problems, we will use In Memory databases.

In Memory Databases are providing the following Advantages.

Zero project setup or infrastructure

Zero Configuration

Zero Maintainance

Easy to use for Learning, POCs and Unit Tests

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

Spring Boot provides Simple Configuration to switch between a real database and an in memory database like H2

In general, In Memory databases are started when application is started and destroyed when application is stopped.

EX: H2

H2 is a relational database management system written in Java. It can be embedded in Java applications or run in the client-server mode.

If we want to use H2 In Memory Database for our applications then we have to use the following steps.

1. Prepare Spring boot starter project with "web, JDBC" "h2" Dependencies in STS.
2. Provide the following properties in application.properties file:

```
spring.h2.console.enabled=true  
spring.datasource.url=jdbc:h2:mem:testdb  
spring.datasource.driverClassName=org.h2.Driver  
spring.datasource.username=sa
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
spring.datasource.password=
```

3. Prepare SQL script files under "src/main/resources".

schema.sql

```
drop table if exists emp1;
```

```
create table emp1(ENO number(5) primary key, ENAME varchar2(10), ESAL float(5),  
EADDR varchar2(10));
```

data.sql

```
insert into emp1 values(111, 'AAA', 5000, 'Hyd');
```

```
insert into emp1 values(222, 'BBB', 6000, 'Hyd');
```

```
insert into emp1 values(333, 'CCC', 7000, 'Hyd');
```

```
insert into emp1 values(444, 'DDD', 8000, 'Hyd');
```

```
insert into emp1 values(555, 'EEE', 9000, 'Hyd');
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

Spring boot application will load and execute these sql scripting files.

EX:

schema.sql

drop table if exists emp1;

create table emp1(ENO number(5) primary key, ENAME varchar2(10), ESAL float(5), EADDR varchar2(10));

data.sql

insert into emp1 values(111, 'AAA', 5000, 'Hyd');

insert into emp1 values(222, 'BBB', 6000, 'Hyd');

insert into emp1 values(333, 'CCC', 7000, 'Hyd');

insert into emp1 values(444, 'DDD', 8000, 'Hyd');

insert into emp1 values(555, 'EEE', 9000, 'Hyd');

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
application.properties
```

```
-----  
server.port=1234
```

```
spring.mvc.view.prefix=/WEB-INF/
```

```
spring.mvc.view.suffix=.jsp
```

```
spring.h2.console.enabled=true
```

```
spring.datasource.url=jdbc:h2:mem:testdb
```

```
spring.datasource.driverClassName=org.h2.Driver
```

```
spring.datasource.username=sa
```

```
spring.datasource.password=
```

```
H2DatabaseApplication.java
```

```
-----  
package com.durgasoft;
```

```
import org.springframework.boot.SpringApplication;
```

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

@SpringBootApplication

```
public class H2databaseApplication //implements CommandLineRunner
{

    public static void main(String[] args) {

        SpringApplication.run(H2databaseApplication.class, args);

    }

}
```

EmployeeController.java

```
-----
package com.durgasoft.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
import org.springframework.ui.ModelMap;
import org.springframework.web.bind.annotation.RequestMapping;

import com.durgasoft.dao.EmployeeDao;
import com.durgasoft.dto.Employee;

@Controller
public class EmployeeController {

    @Autowired
    private EmployeeDao employeeDao;

    @RequestMapping("getallemployees")
    public String getAllEmployees(ModelMap map) {
        List<Employee> allEmployees = employeeDao.getAllEmployees();
        map.addAttribute("allEmployees", allEmployees);
        return "employeesdetails";
    }
}
```

CONTACT US:**Mobile: +91- 8885 25 26 27**Mail ID: durgasoftonlinetraining@gmail.com**+91- 7207 21 24 27/28**WEBSITE: www.durgasoftonline.com**US NUM: 4433326786****FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,**

BY NAGOOR BABU

```
}

EmployeeDao.java
-----

package com.durgasoft.dao;

import java.util.List;

import com.durgasoft.dto.Employee;

public interface EmployeeDao {
    public List<Employee> getAllEmployees();
}


```

Employee.java:

```
-----

package com.durgasoft.dto;


```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
1. public class Employee {
2.     private int eno;
3.     private String ename;
4.     private float esal;
5.     private String eaddr;
6.
7.     public int getEno() {
8.         return eno;
9.     }
10.    public void setEno(int eno) {
11.        this.eno = eno;
12.    }
13.    public String getEname() {
14.        return ename;
15.    }
16.    public void setEname(String ename) {
17.        this.ename = ename;
18.    }
19.    public float getEsal() {
20.        return esal;
21.    }
22.    public void setEsal(float esal) {
23.        this.esal = esal;
24.    }
25.    public String getEaddr() {
26.        return eaddr;
27.    }
28.    public void setEaddr(String eaddr) {
29.        this.eaddr = eaddr;
30.    }
31.}
```

EmployeeDaoImpl.java

package com.durgasoft.dao;

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.jdbc.core.JdbcTemplate;
import org.springframework.stereotype.Repository;

import com.durgasoft.dto.Employee;

@Repository
public class EmployeeDaoImpl implements EmployeeDao {

    @Autowired
    private JdbcTemplate jdbcTemplate;

    @Override
    public List<Employee> getAllEmployees() {
        List<Employee> list = jdbcTemplate.query("select * from emp1", (rs, rowNum)
-> {
            Employee emp = new Employee();
            emp.setEno(rs.getInt("ENO"));
        });
    }
}
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonline@training@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,



BY NAGOOR BABU

```
emp.setEname(rs.getString("ENAME"));

emp.setEsal(rs.getFloat("ESAL"));

emp.setEaddr(rs.getString("EADDR"));

return emp;

});

return list;

}

}
```

employeeedetails.jsp:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h2 style="color: red;" align="center">Employees Details</h2>
<table align="center" border="1">
<tr><th>ENO</th><th>ENAME</th><th>ESAL</th><th>EADDR</th></tr>
<c:forEach var="emp" items="{allEmployees}">
<tr>
    <td>${emp.eno}</td>
    <td>${emp.ename}</td>
    <td>${emp.esal}</td>
    <td>${emp.eaddr}</td>
</tr>
</c:forEach>
</table>
</body>
```

CONTACT US:**Mobile: +91- 8885 25 26 27**Mail ID: durgasoftonline training@gmail.com**+91- 7207 21 24 27/28**WEBSITE: www.durgasoftonline.com**US NUM: 4433326786**

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

</html>

pom.xml

```
1. <?xml version="1.0" encoding="UTF-8"?>
2. <project xmlns="http://maven.apache.org/POM/4.0.0"
3.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4.     xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/x
sd/maven-4.0.0.xsd">
5.     <modelVersion>4.0.0</modelVersion>
6.     <parent>
7.         <groupId>org.springframework.boot</groupId>
8.         <artifactId>spring-boot-starter-parent</artifactId>
9.         <version>2.1.9.RELEASE</version>
10.        <relativePath /> <!-- lookup parent from repository -->
11.    </parent>
12.    <groupId>com.durgasoft</groupId>
13.    <artifactId>h2database</artifactId>
14.    <version>0.0.1-SNAPSHOT</version>
15.    <name>h2database</name>
16.    <description>Spring boot application on H2 In Memory Database</description>
17.
18.    <properties>
19.        <java.version>1.8</java.version>
20.    </properties>
21.
22.    <dependencies>
23.        <dependency>
24.            <groupId>org.springframework.boot</groupId>
25.            <artifactId>spring-boot-starter-web</artifactId>
26.        </dependency>
27.
28.
29.        <dependency>
30.            <groupId>org.apache.tomcat</groupId>
```

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com

+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com

US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

```
31.         <artifactId>tomcat-jasper</artifactId>
32.         <version>9.0.26</version>
33.     </dependency>
34.
35.     <dependency>
36.         <groupId>javax.servlet</groupId>
37.         <artifactId>jstl</artifactId>
38.         <version>1.2</version>
39.     </dependency>
40.
41.
42.     <dependency>
43.         <groupId>org.springframework.boot</groupId>
44.         <artifactId>spring-boot-starter-jdbc</artifactId>
45.     </dependency>
46.
47.     <dependency>
48.         <groupId>com.h2database</groupId>
49.         <artifactId>h2</artifactId>
50.         <scope>runtime</scope>
51.     </dependency>
52.     <dependency>
53.         <groupId>org.springframework.boot</groupId>
54.         <artifactId>spring-boot-starter-test</artifactId>
55.         <scope>test</scope>
56.     </dependency>
57. </dependencies>
58.
59. <build>
60.     <plugins>
61.         <plugin>
62.             <groupId>org.springframework.boot</groupId>
63.             <artifactId>spring-boot-maven-plugin</artifactId>
64.         </plugin>
65.     </plugins>
66. </build>
67.
68. </project>
```

To get H2 In memory Database use the following End Point.

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,

BY NAGOOR BABU

`http://localhost:1234/h2-console` and click on "Connect" button.

To get all Employees list use the following End Point.

`http://localhost:1234/getallemployees`

CONTACT US:

Mobile: +91- 8885 25 26 27

Mail ID: durgasoftonlinetraining@gmail.com



+91- 7207 21 24 27/28

WEBSITE: www.durgasoftonline.com



US NUM: 4433326786

FLAT NO: 202, HMDA MYTRIVANUM, AMEERPET, HYDERABAD.,