Spring Tool Suite is a software written by Vmware and is an IDE for making Spring Boot Applications.

What is Spring Boot ?

It provides an easier and faster way to set up, configure, and run both simple and web-based applications.

It is a Spring module that provides the **RAD (*Rapid Application Development*)** feature to the Spring Framework. It is used to create a stand-alone Spring-based application that you can just run because it needs minimal Spring configuration.

Spring Boot is the combination of **Spring Framework** and **Embedded Servers**.

We can use Spring **STS IDE (Spring Tool Suite)** or **Spring Initializr** to develop Spring Boot Java applications.

In this tutorial Spring Tool Suite Version 4 is used to make a simple web application to insert a record in mysql database.

download and install mysql server from www.wampserver.com

open MySql Console.

Create a database in mysql with name as contacts.

Following commands can be used to create a database and tables in contacts table.

create database contacts;

use contacts;

create table contacts(cname varchar(20),caddress varchar(20),cmobileno varchar(20),cemail varchar(20));

What is annotation in Spring Boot?

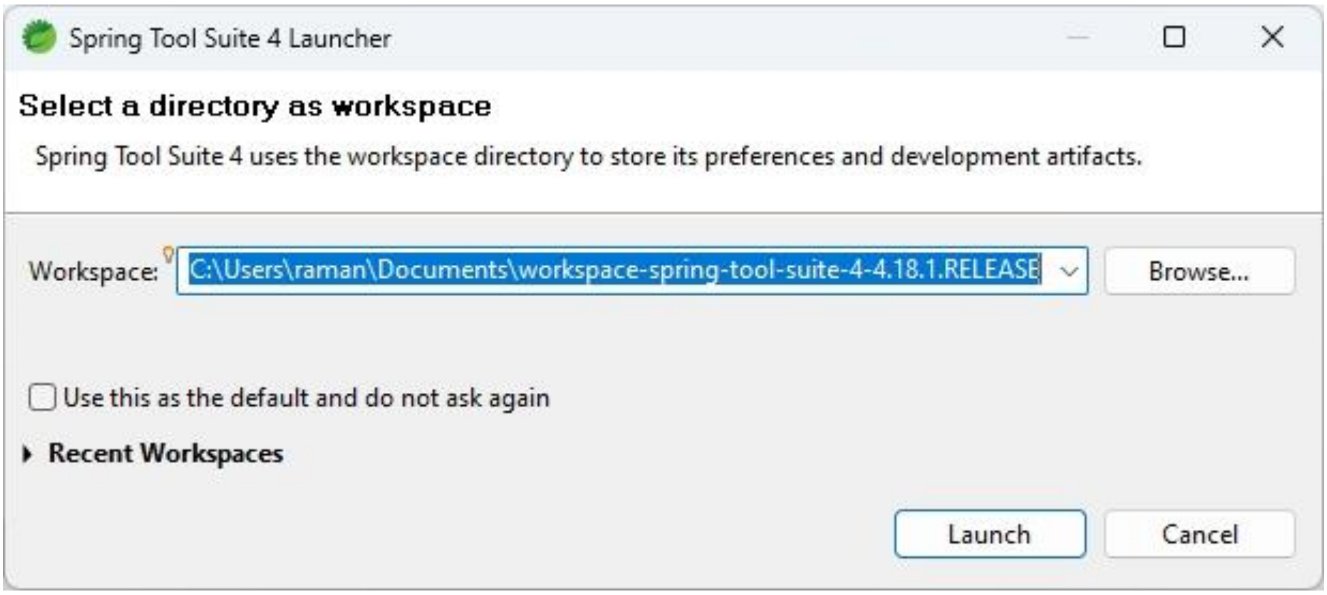
Spring Boot Annotations is a form of metadata that provides data about a program. In other words, annotations are used to provide supplemental information about a program. It is not a part of the application that we develop. It does not have a direct effect on the operation of the code they annotate.

Let's start with creating a simple web application with Spring Tool Suite 4 which will insert a record in mysql database.

Download Spring Tool Suite from Spring | Tools

Extract the **zip** file and install the STS.

Run SpringToolSuite4.exe and Spring Tool Suite 4 will start.



Select a directory as workspace

Spring Tool Suite 4 uses the workspace directory to store its preferences and development artifacts.

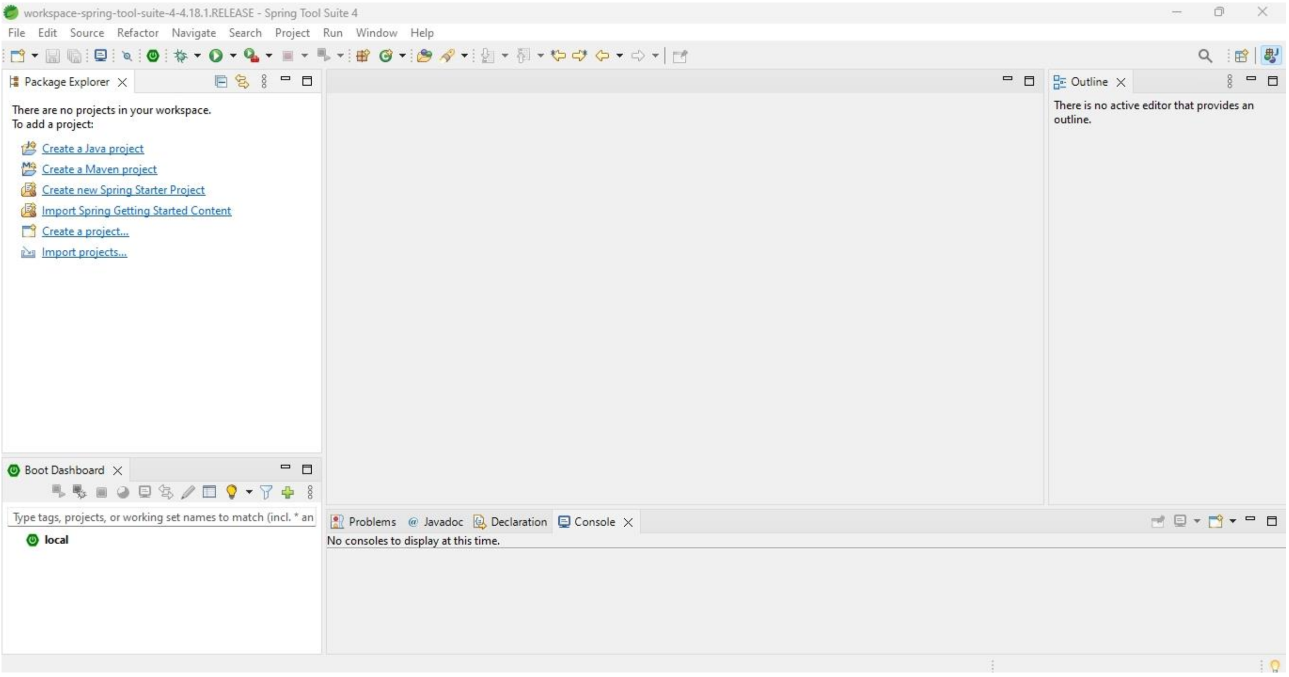
Workspace: C:\Users\raman\Documents\workspace-spring-tool-suite-4-4.18.1.RELEASE

☐ Use this as the default and do not ask again

▸ **Recent Workspaces**

Click on Launch.

Following window will appear



There are no projects in your workspace.
To add a project:

- Create a Java project
- Create a Maven project
- Create new Spring Starter Project
- Import Spring Getting Started Content
- Create a project...
- Import projects...

There is no active editor that provides an outline.

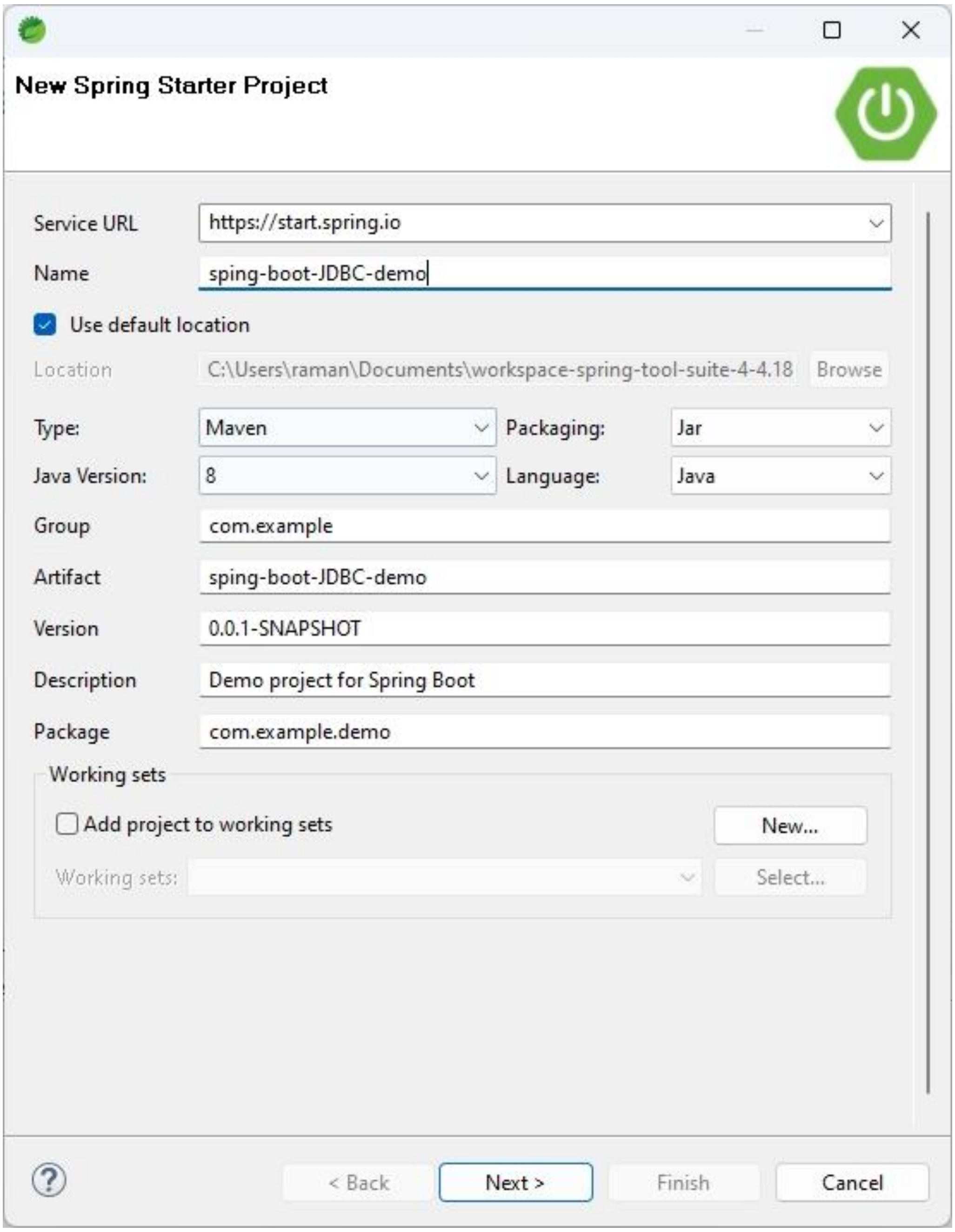
Type tags, projects, or working set names to match (incl. * an

local

No consoles to display at this time.

Click on Create New Spring Starter Project

Following window will appear



New Spring Starter Project

Service URL: https://start.spring.io

Name: sping-boot-JDBC-demo

☑ Use default location

Location: C:\Users\raman\Documents\workspace-spring-tool-suite-4-4.18

Type: Maven

Packaging: Jar

Java Version: 8

Language: Java

Group: com.example

Artifact: sping-boot-JDBC-demo

Version: 0.0.1-SNAPSHOT

Description: Demo project for Spring Boot

Package: com.example.demo

Working sets

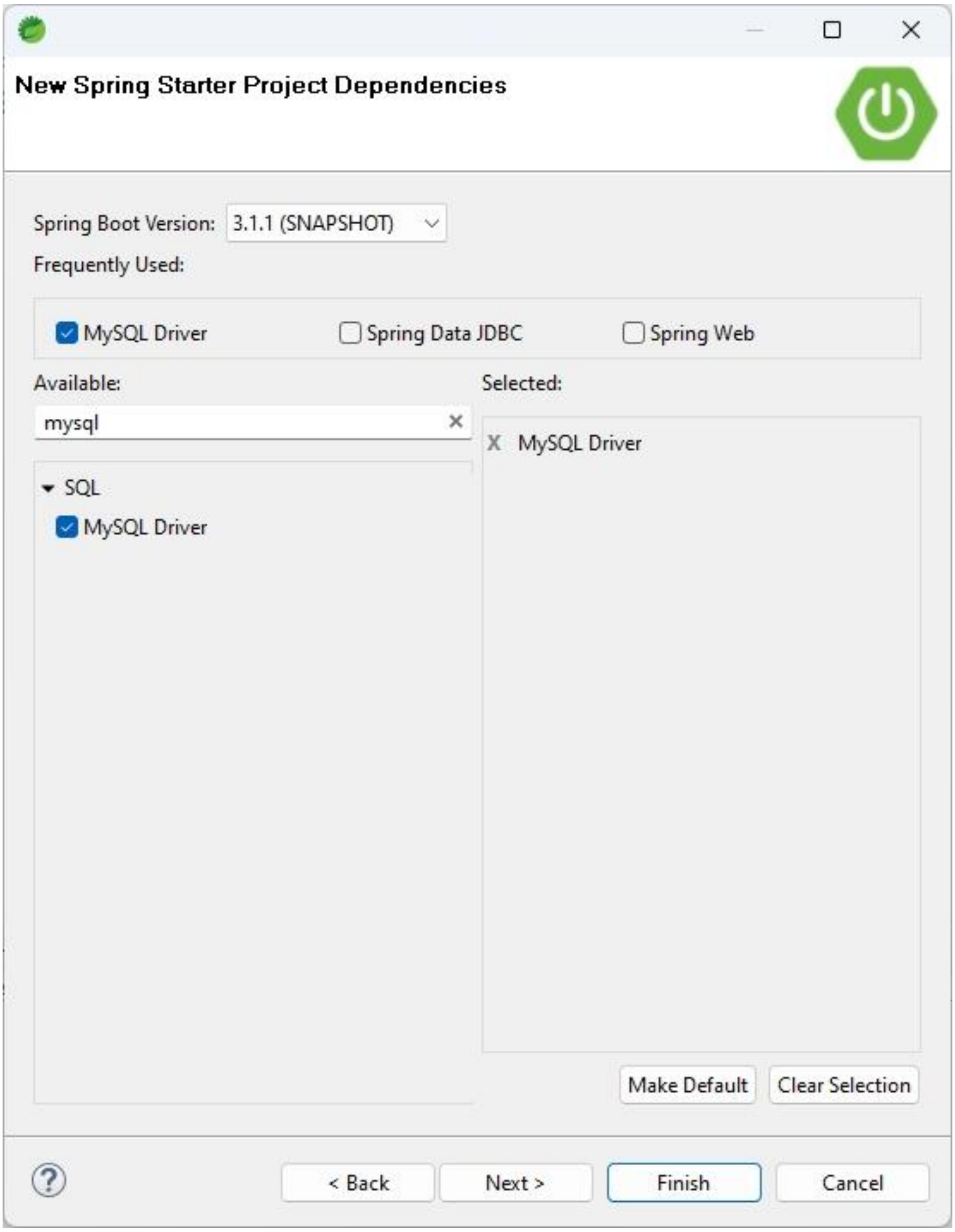
□ Add project to working sets

Working sets:

Click on Next

Now in this window you have to select dependencies for the project.

Dependency means a jar file required for a particular task in the project for example to connect spring boot application with mysql you have to include dependency for that purpose which is MySQL Driver.



New Spring Starter Project Dependencies

Spring Boot Version: 3.1.1 (SNAPSHOT)

Frequently Used:

☑ MySQL Driver ☐ Spring Data JDBC ☐ Spring Web

Available: mysql

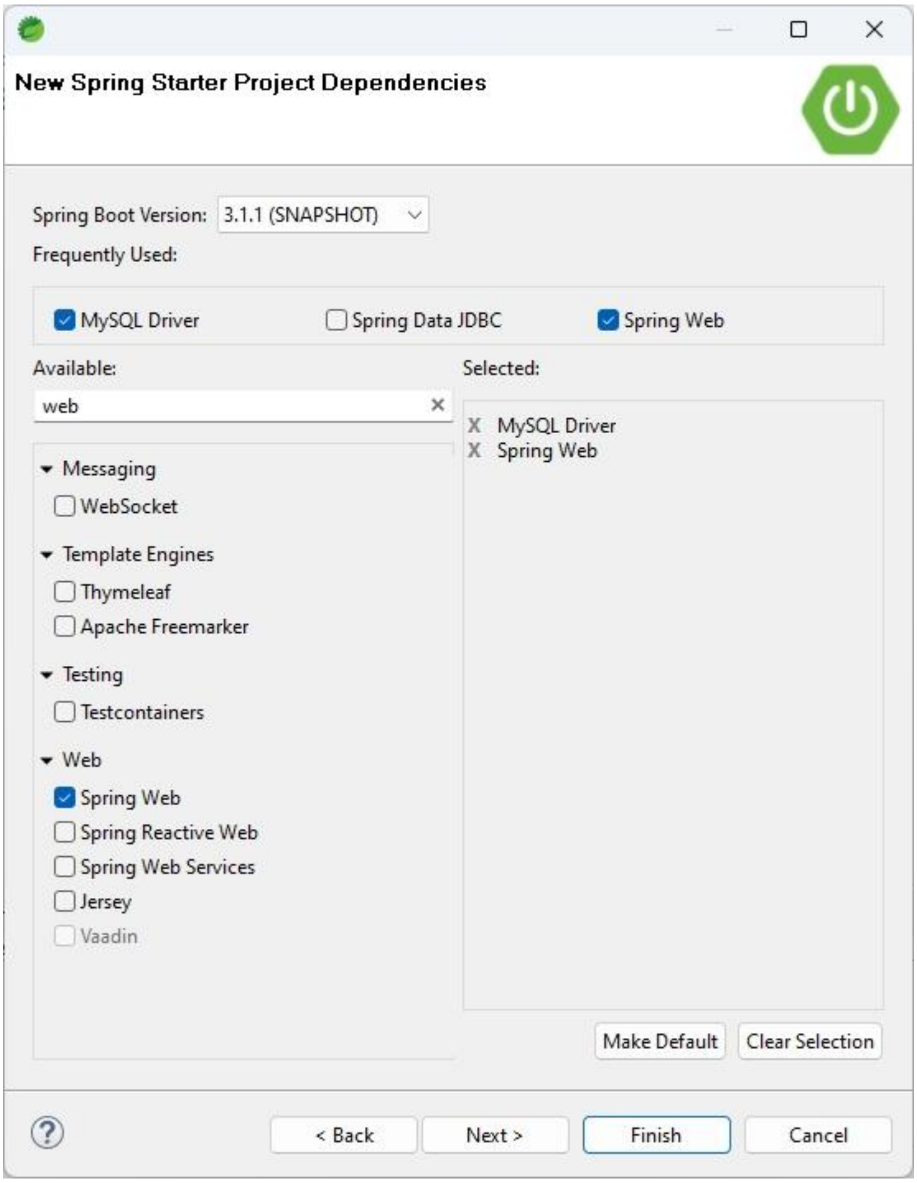
SQL

☑ MySQL Driver

Selected:

X MySQL Driver

Now add another dependency Web



New Spring Starter Project Dependencies

Spring Boot Version: 3.1.1 (SNAPSHOT)

Frequently Used:

☑ MySQL Driver ☐ Spring Data JDBC ☑ Spring Web

Available:

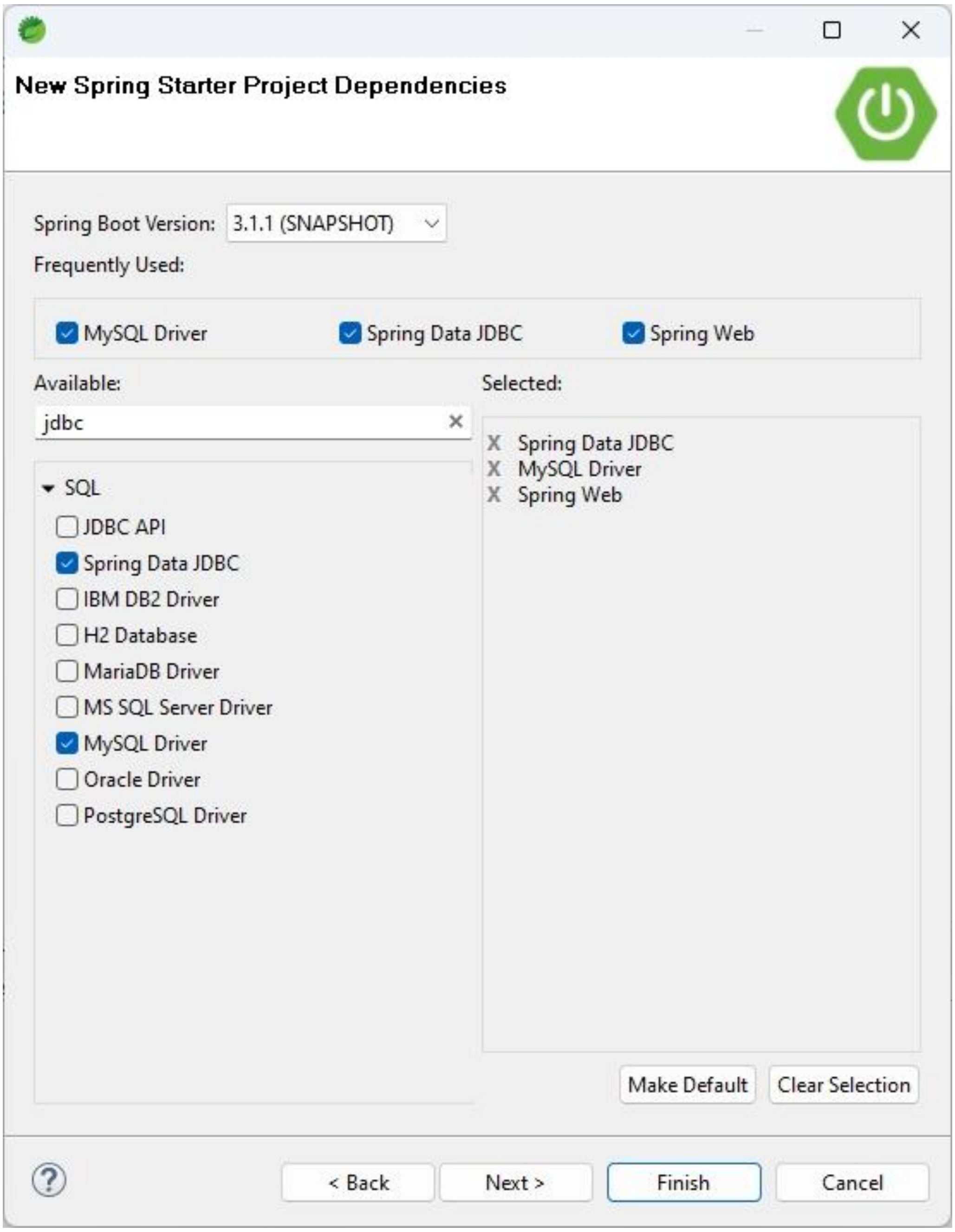
web

- Messaging
  - ☐ WebSocket
- Template Engines
  - ☐ Thymeleaf
  - ☐ Apache Freemarker
- Testing
  - ☐ Testcontainers
- Web
  - ☑ Spring Web
  - ☐ Spring Reactive Web
  - ☐ Spring Web Services
  - ☐ Jersey
  - ☐ Vaadin

Selected:

X MySQL Driver
X Spring Web

Now add another dependency Spring Data JDBC



New Spring Starter Project Dependencies

Spring Boot Version: 3.1.1 (SNAPSHOT)

Frequently Used:

☑ MySQL Driver ☑ Spring Data JDBC ☑ Spring Web

Available:

jdbc

SQL

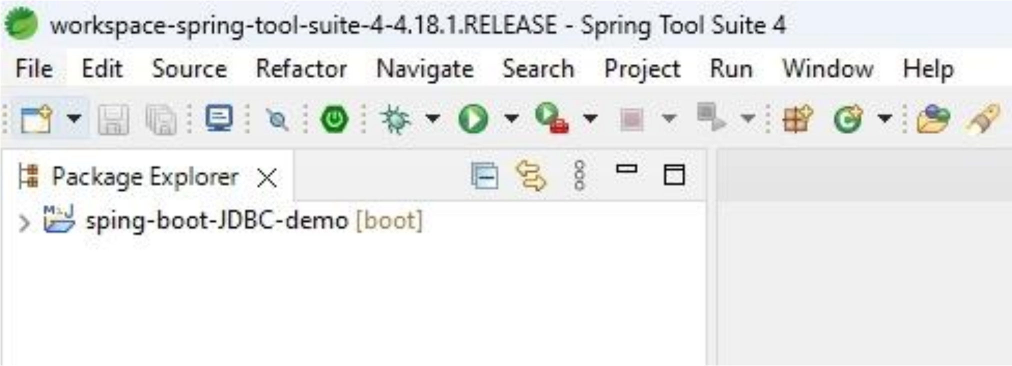
- ☐ JDBC API
- ☑ Spring Data JDBC
- ☐ IBM DB2 Driver
- ☐ H2 Database
- ☐ MariaDB Driver
- ☐ MS SQL Server Driver
- ☑ MySQL Driver
- ☐ Oracle Driver
- ☐ PostgreSQL Driver

Selected:

- X Spring Data JDBC
- X MySQL Driver
- X Spring Web

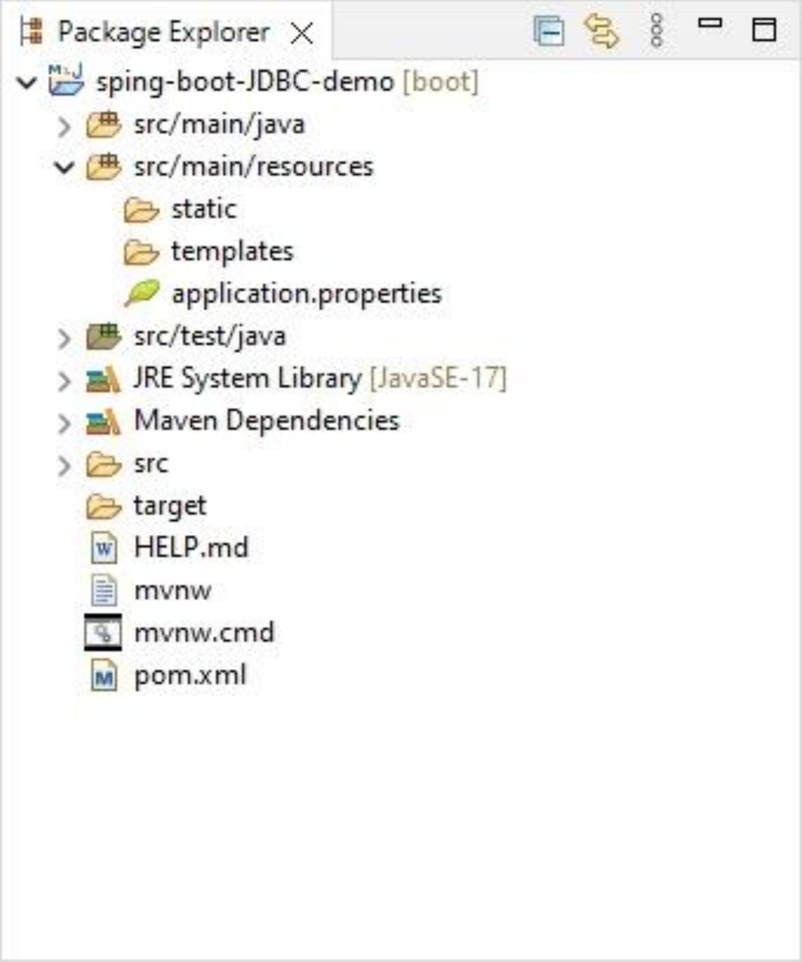
Click on Finish

You will see the following output


workspace-spring-tool-suite-4-4.18.1.RELEASE - Spring Tool Suite 4
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
sping-boot-JDBC-demo [boot]

Open the project in Package Explorer

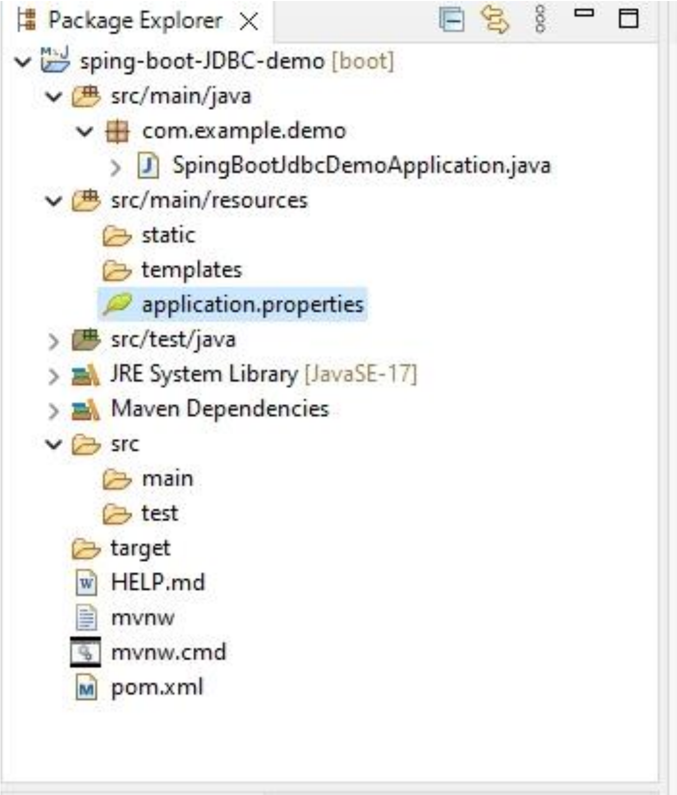
You will see following output



Go to application.properties file

and add following code in it

```
spring.datasource.url=jdbc:mysql://localhost:3306/contacts
spring.datasource.username=root
spring.datasource.password=
spring.jpa.hibernate.ddl-auto=create-drop
```



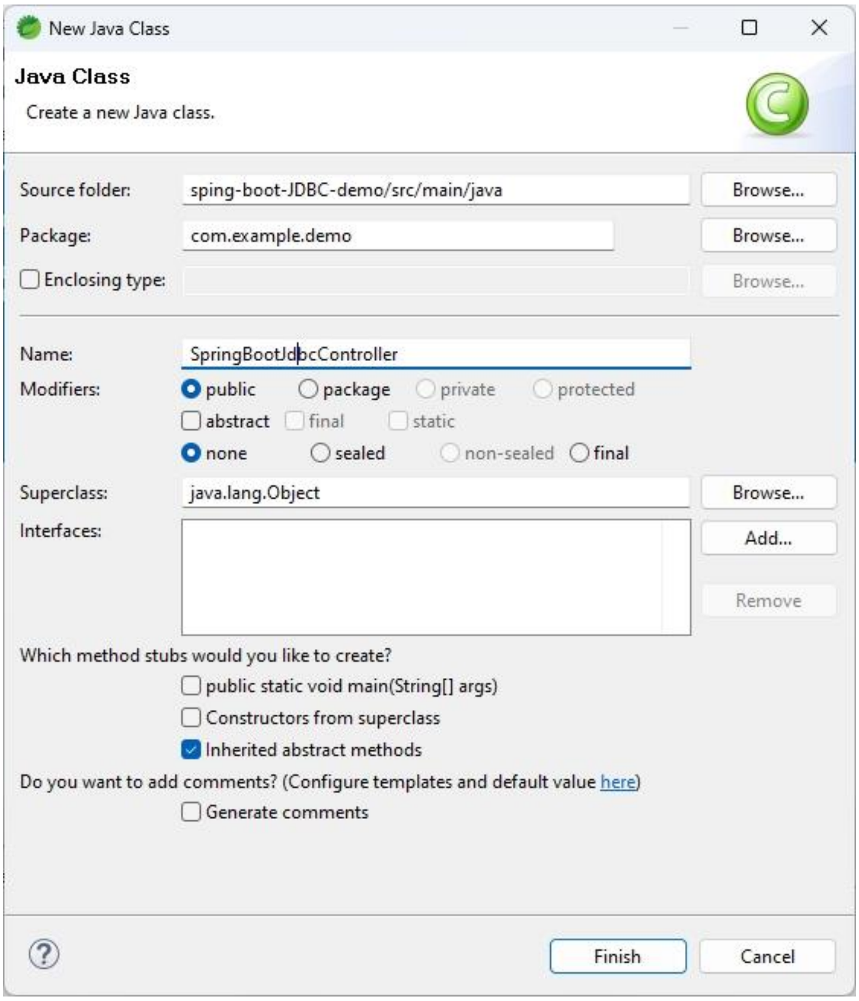
```
Package Explorer
sping-boot-JDBC-demo [boot]
  src/main/java
    com.example.demo
      SpingBootJdbcDemoApplication.java
  src/main/resources
    static
    templates
    application.properties
  src/test/java
  JRE System Library [JavaSE-17]
  Maven Dependencies
  src
    main
    test
  target
  HELP.md
  mvnw
  mvnw.cmd
  pom.xml
```

Go to SpringBootJdbcDemoApplication.java

You should see following code in this file

```
package com.example.demo;

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

@SpringBootApplication
public class SpingBootJdbcDemoApplication {

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

}
```

Now create a Controller class in com.example.demo package by Clicking on New Class Option



New Java Class

Java Class

Create a new Java class.

Source folder: sping-boot-JDBC-demo/src/main/java

Package: com.example.demo

□ Enclosing type:

Name: SpringBootJdbcController

Modifiers: ● public ○ package ○ private ○ protected

□ abstract □ final □ static

● none ○ sealed ○ non-sealed ○ final

Superclass: java.lang.Object

Interfaces:

Which method stubs would you like to create?

□ public static void main(String[] args)

□ Constructors from superclass

☑ Inherited abstract methods

Do you want to add comments? (Configure templates and default value here)

□ Generate comments

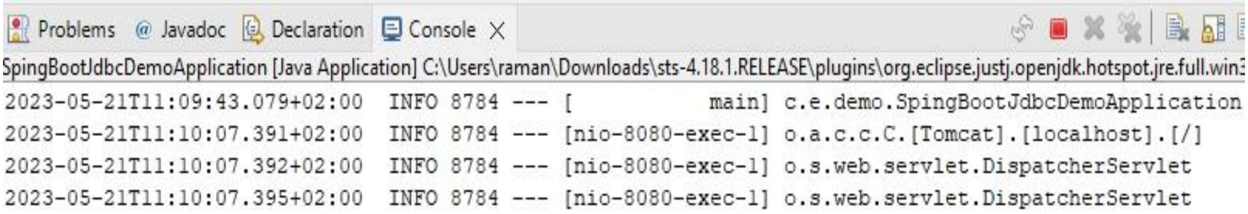
Click on Finish

Write following code in SpringBootJdbcController.java

```
package com.example.demo;

import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.jdbc.core.JdbcTemplate;
import org.springframework.web.bind.annotation.RestController;
@RestController
public class SpringBootJdbcController {
    @Autowired
    JdbcTemplate jdbc;
    @RequestMapping("/insert")
    public String index(){
        jdbc.execute("insert into contacts values('raman','raman
address','1234567890','raman@abc.com')");
        return "Data Saved to Database";
    }
}
```

Now Right Click on SpringBootJdbcDemoApplication.java and Select (Run As) -> (Java Application)

You will see the following output in Console

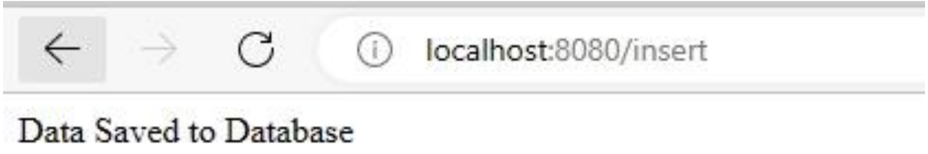


```
SpingBootJdbcDemoApplication [Java Application] C:\Users\raman\Downloads\sts-4.18.1.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win3
2023-05-21T11:09:43.079+02:00  INFO 8784 --- [           main] c.e.demo.SpingBootJdbcDemoApplication
2023-05-21T11:10:07.391+02:00  INFO 8784 --- [nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/]
2023-05-21T11:10:07.392+02:00  INFO 8784 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet
2023-05-21T11:10:07.395+02:00  INFO 8784 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet
```

Now open Browser and in address type

http://localhost:8080/insert

You will see the following output



Data Saved to Database

After the above command data is inserted in the database.

You can open table contacts in database contacts and you will see the record just inserted in the database.