

## Practice Test

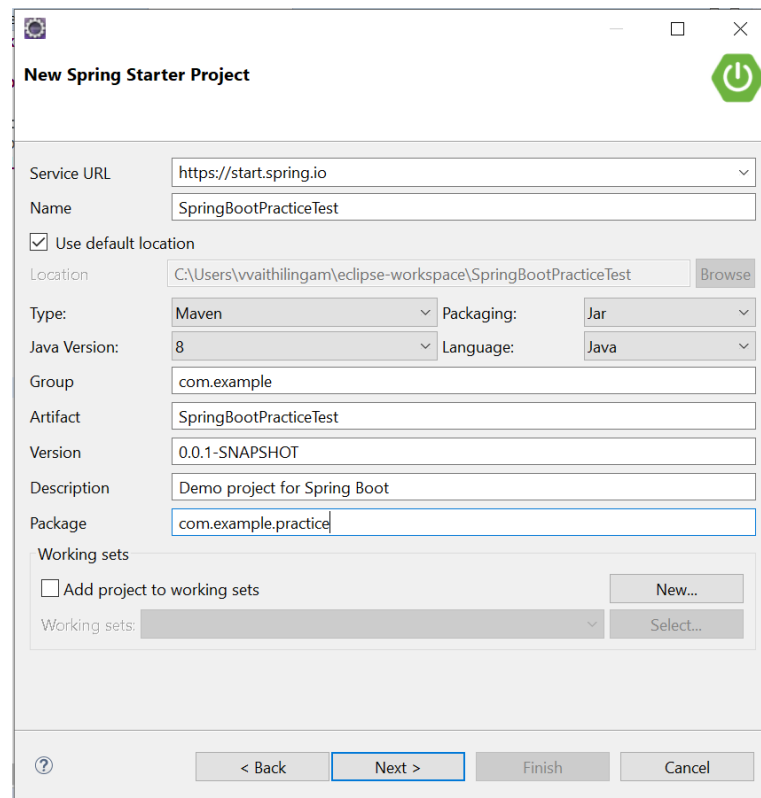
**Step 1:** Create a table “**Student**” under “**centennial**” database with the following fields in MySQL

stdNo integer

stdName String

course String

**Step 2:** Create a new Spring Starter project in eclipse or create a project using spring.io web site named “SpringBootPracticeTest”.

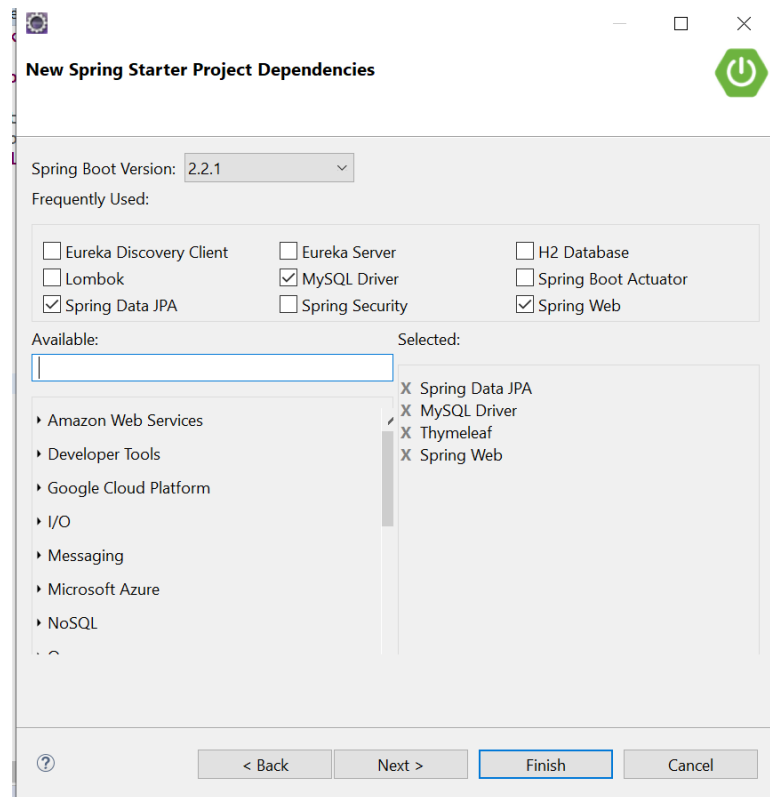


The screenshot shows the 'New Spring Starter Project' dialog box in Eclipse. The dialog is titled 'New Spring Starter Project' and features a green power button icon in the top right corner. The fields are as follows:

- Service URL: <https://start.spring.io>
- Name: SpringBootPracticeTest
- ☒ Use default location
- Location: C:\Users\vaithilingam\workspace\SpringBootPracticeTest (with a 'Browse' button)
- Type: Maven (dropdown)
- Packaging: Jar (dropdown)
- Java Version: 8 (dropdown)
- Language: Java (dropdown)
- Group: com.example
- Artifact: SpringBootPracticeTest
- Version: 0.0.1-SNAPSHOT
- Description: Demo project for Spring Boot
- Package: com.example.practice

At the bottom, there is a 'Working sets' section with an unchecked checkbox 'Add project to working sets', a 'New...' button, and a 'Working sets:' dropdown with a 'Select...' button. The bottom navigation bar includes a help icon, '< Back', 'Next >', 'Finish', and 'Cancel' buttons.

Choose the necessary dependencies like Spring Web, MySQL Driver, Spring Data JPA, Thymeleaf, and Spring Dev Tools.



**Step 3:** Write this code in student.java class.

```
public class Student {

    private int stdId;
    private String stdName;
    private String Course;

    .....
    .....
}
```

Include all the necessary annotations like @Entity, @Id etc and validation annotations like @NotNull, @Size or @Range etc.

**Step 4:** Write this code for StudentRepository.java class

```
package com.example.practice;

import org.springframework.data.jpa.repository.JpaRepository;

public interface StudentRepository extends JpaRepository<Student, Integer> {

}
```

**Step 5:** Write this code for StudentController.java file

```
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.bind.annotation.RestController;
```

@Controller

```
public class StudentController {

    @Autowired
    private StudentRepository stdRepository;


    @RequestMapping("/")
    public String home()
    {
        return "index";
    }

}
```

```

    @RequestMapping("/del")
    public String del()
    {
        return "delete";
    }

    @RequestMapping("/update")
    public String update()
    {
        return "update";
    }

    @PostMapping("/add")
    public @ResponseBody String add(@RequestParam("stdId") int stdId,
        @RequestParam("stdName") String stdName,
        @RequestParam("course") String course);
    {
        Student student=new Student(stdId,stdName,course);
        stdRepository.save(student);
        return "A student info added";
    }

    @RequestMapping("/show")
    public String show(Model model)
    {
        model.addAttribute("student", stdRepository.findAll());
        return "show";
    }

```

```

        @RequestMapping("/edit")
public @ResponseBody String edit(@RequestParam("stdId") int stdId,
        @RequestParam("stdName") String stdName,
        @RequestParam("course") String course)
{
    Student student=new Student(stdId,stdName,course);
    stdRepository.save(student);
    return "A student info updated";
}

@RequestMapping("/delete")
public @ResponseBody String delete(@RequestParam("stdId") int stdId)
{
    stdRepository.deleteByld(stdId);
    return "one row deleted";
}
}

```

**Step 6:** Ensure that you have boot application class named “SpringBootPracticeTestApplication”

```

@SpringBootApplication
public class SpringBootPracticeTestApplication {
    public static void main(String[] args) {
        SpringApplication.run(SpringBootRestEmployeeApplication.class, args);
        System.out.println("Spring Boot JPA started.");
    }
}

```

**Step 7:** Create your input and view files under src/main/resources/templates folder

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<meta charset="ISO-8859-1">
<title>Get Students</title>
</head>
<body>
    <form action="add" method="POST">
        Student Id : <input type="text" name="stdId"/> </br>
        Student Name : <input type="text" name="stdName"/></br>
        Course Name : <input type="text" name="course"/></br>
        <input type="submit" name="Add Student"/>
    </form>
</body>
</html>
```

Show.html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<meta charset="ISO-8859-1">
<title>Student list</title>
</head>
<body>
<table>
    <thead>
        <tr>
            <th>Id</th>
            <th>Name</th>
            <th>Job</th>
            <th>Salary</th>
        </tr>
    </thead>
    <tbody>
        <tr th:each="std:${student}">
            <td th:text="${std.stdId}">Id</td>
            <td th:text="${std.stdName}">Name</td>
            <td th:text="${std.course}">courser</td>
        </tr>
    </tbody>
</table>
</body>
</html>
```

Like create other html files too

**Step 8:** Run this spring boot project