**4. Create REST Controller for CRUD operations in spring boot using MySQL andJPA.**

Step1: Download and install POSTMAN and MySQL Community Edition.

Step2: Set the path of MySQL🡪 in environment variable.

Step3:Go to the command prompt and give command

**mysqld**- to login

Enter the user name and password as:

**mysql –u root –p**

**Enter Password: \*\*\*\*\*\*\*\*\*\*\***

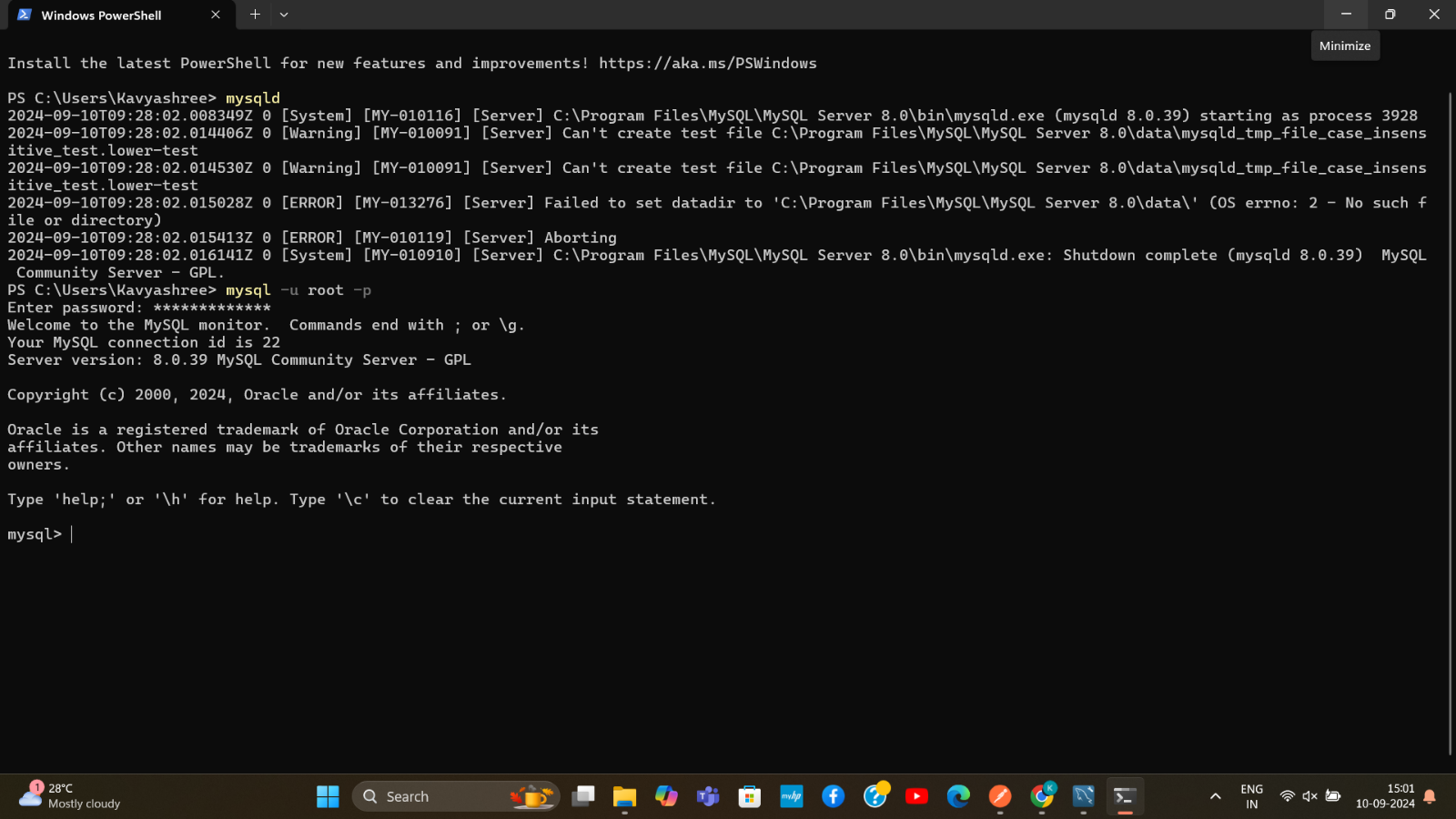
Create a database:

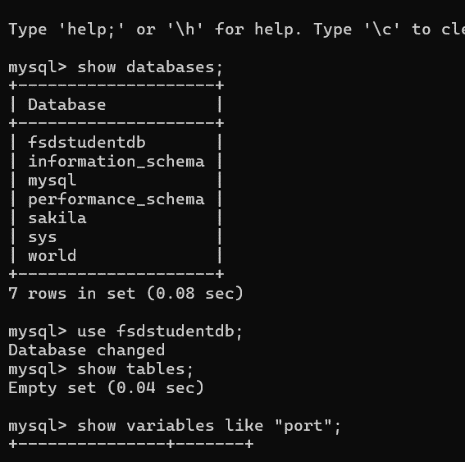
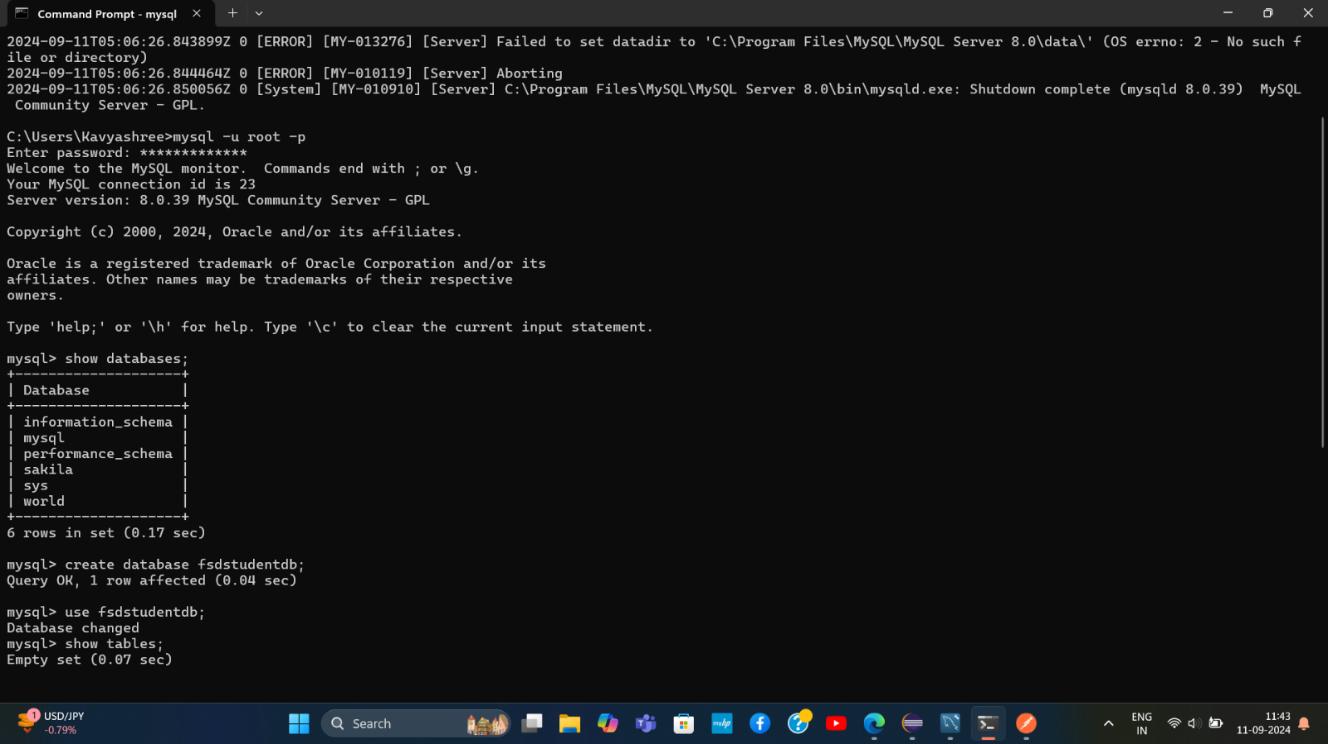
**create database fsdstudentdb**

**use fsdstudentdb**

to check in which port mysql is running:

**show variables like “port”**

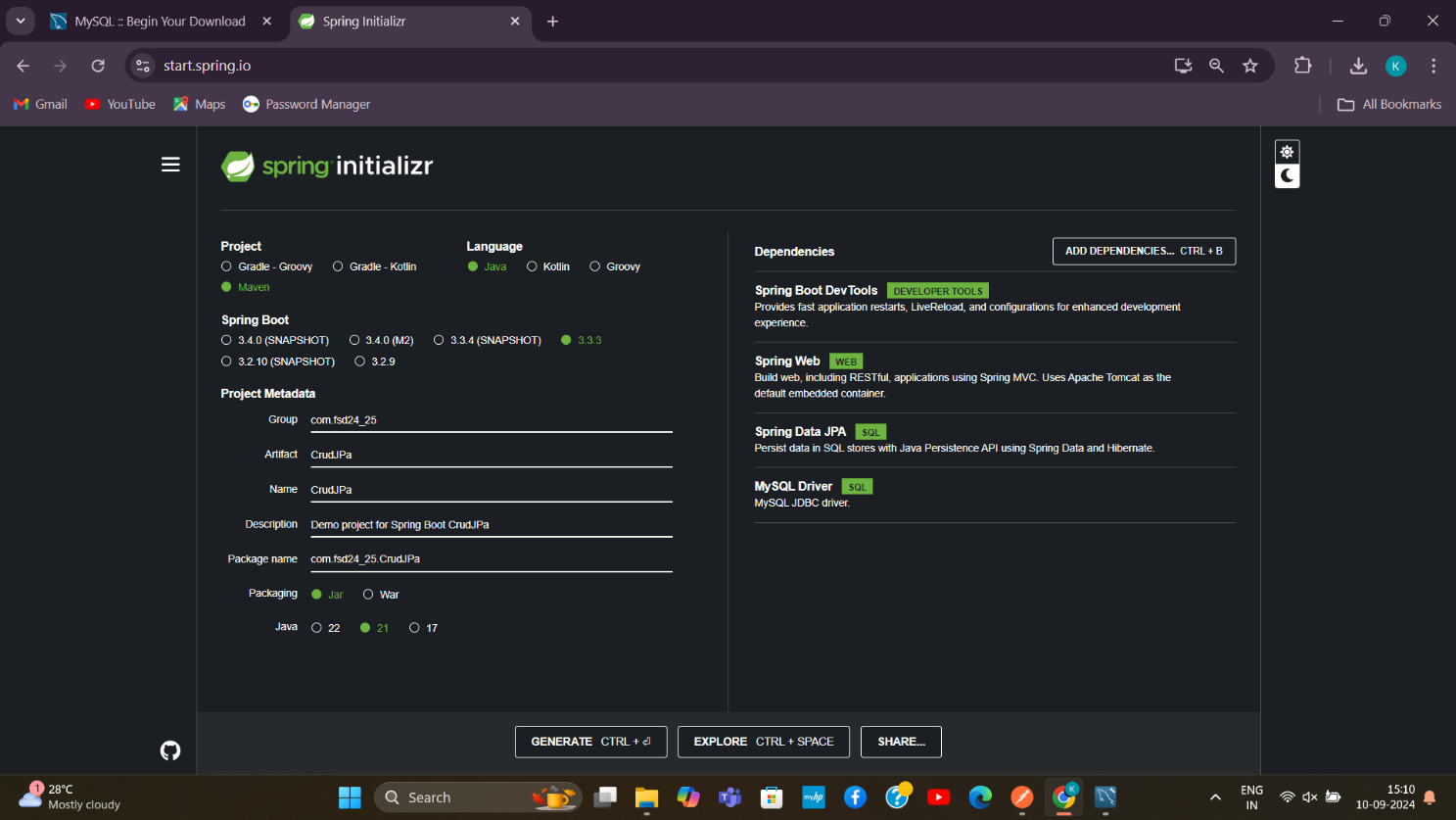




Step4: To set project configuration🡪Go to browser🡪Search **start.spring.io**

Step5: In spring initializr🡪Select Project-Maven🡪Language-Java🡪Spring Boot-3.3.3🡪 Group- com.fsd24\_25🡪Artifact-CrudJPa🡪Description-Demo project for spring boot CrudJpa🡪Packaging-Jar🡪Java-21.

Dependencies🡪ADD DEPENDENCIES🡪Select-Spring Boot Dev Tools ,Spring Web,MySQL Driver and Spring Data JPA🡪 Click on Generate.

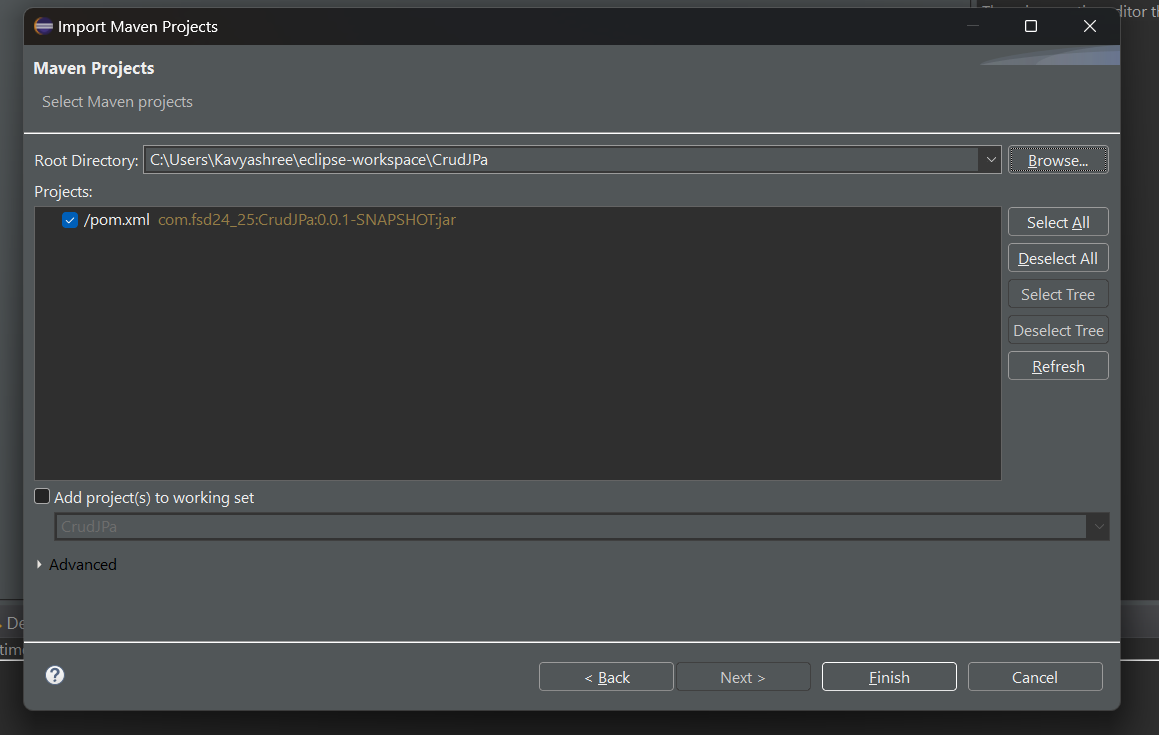


Step6: Next extract your zip file to workspace.

Step7: Select a destination.

Step8: Go to Eclipse🡪File🡪Import🡪Maven🡪Existing maven🡪Click on Browse and select the folder from where you extracted🡪Tick mark on pom.xml🡪Click on Finish.

Save the pom.xml



Step9: Create four packages:

com.fsd24\_25.CrudJPa.Entity

com.fsd24\_25.CrudJPa.Controller

com.fsd24\_25.CrudJPa.Repository

com.fsd24\_25.CrudJPa.Service

Step10: Create a Student class in com.fsd24\_25.CrudJPa.Entity

**Student.java:**

package com.fsd24\_25.CrudJPa.Entity;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.Table;

@Entity

@Table(name="Student")

public class Student {

@Id

private String regno;

private String name;

private String branch;

private int sem;

public Student() {}

public Student(String regno, String name, String branch, int sem) {

super();

this.regno = regno;

this.name = name;

this.branch = branch;

this.sem = sem;

}

public String getRegno() {

return regno;

}

public void setRegno(String regno) {

this.regno = regno;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getBranch() {

return branch;

}

public void setBranch(String branch) {

this.branch = branch;

}

public int getSem() {

return sem;

}

public void setSem(int sem) {

this.sem = sem;

}

@Override

public String toString() {

return ("{\n\"RegNo\":\""+this.regno+"\",\n\"Name\":\""+this.name+"\",\n\"Branch\":\""+this.branch+"\",\n\"Sem\":"+this.sem+"\"\n}");

}

}

Step11: Create another class StudentRepo inside com.fsd24\_25.CrudJPa.Repository

**StudenRepo.java:**

package com.fsd24\_25.CrudJPa.Repository;

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

import com.fsd24\_25.CrudJPa.Entity.Student;

public interface StudentRepo extends JpaRepository<Student, String> {}

Step12: Create another class StudService in com.fsd24\_25.CrudJPa.Service

**StudService.java:**

package com.fsd24\_25.CrudJPa.Service;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import com.fsd24\_25.CrudJPa.Entity.Student;

import com.fsd24\_25.CrudJPa.Repository.StudentRepo;

@Service

public class StudService {

@Autowired

private StudentRepo studRep;

public List <Student> getAllStudents() {

return studRep.findAll();

}

public Student insertStudent(Student st) {

studRep.save(st);

return st;

}

public Student updateStudent(Student s) {

studRep.save(s);

return s;

}

public void deleteStudent(String id) {

Student s=studRep.getReferenceById(id);

studRep.delete(s);

}

}

Step13: Create StudCtrl class inside com.fsd24\_25.CrudJPa.Controller

**studCtrl.java:**

package com.fsd24\_25.CrudJPa.Controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.DeleteMapping;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.PutMapping;

import org.springframework.web.bind.annotation.RequestBody;

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

import com.fsd24\_25.CrudJPa.Dto.studDetail;

import com.fsd24\_25.CrudJPa.Entity.student;

import com.fsd24\_25.CrudJPa.Service.StudService;

@RestController

public class StudCtrl {

@Autowired

private StudService s;

@GetMapping("/students")

public List<student> getStudentList(){

return s.getAllStudents();

}

@GetMapping("/")

public String disp(){

return "Welcome";

}

@DeleteMapping("/del/{regno}")

public void delStud(@PathVariable String regno) {

s.deleteStudent(regno);

}

@PostMapping("/student")

public student insertStud(@RequestBody student st){

return s.insertStudent(st);

}

@PutMapping("/student")

public student updateStud(@RequestBody student st){

return s.updateStudent(st);

}

@GetMapping("/details")

private List<studDetail> dispstudDetails() {

return s.getstudDetail();

}

}

Step14: Now write the following code in applicaton.properties

Src/main/resources

**Application.properties:**

spring.application.name=CrudJPa

spring.datasource.url=jdbc:MySql://localhost:3306/fsdstudentdb

spring.datasource.username=root

spring.datasource.password=@Kavyagowda23

spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

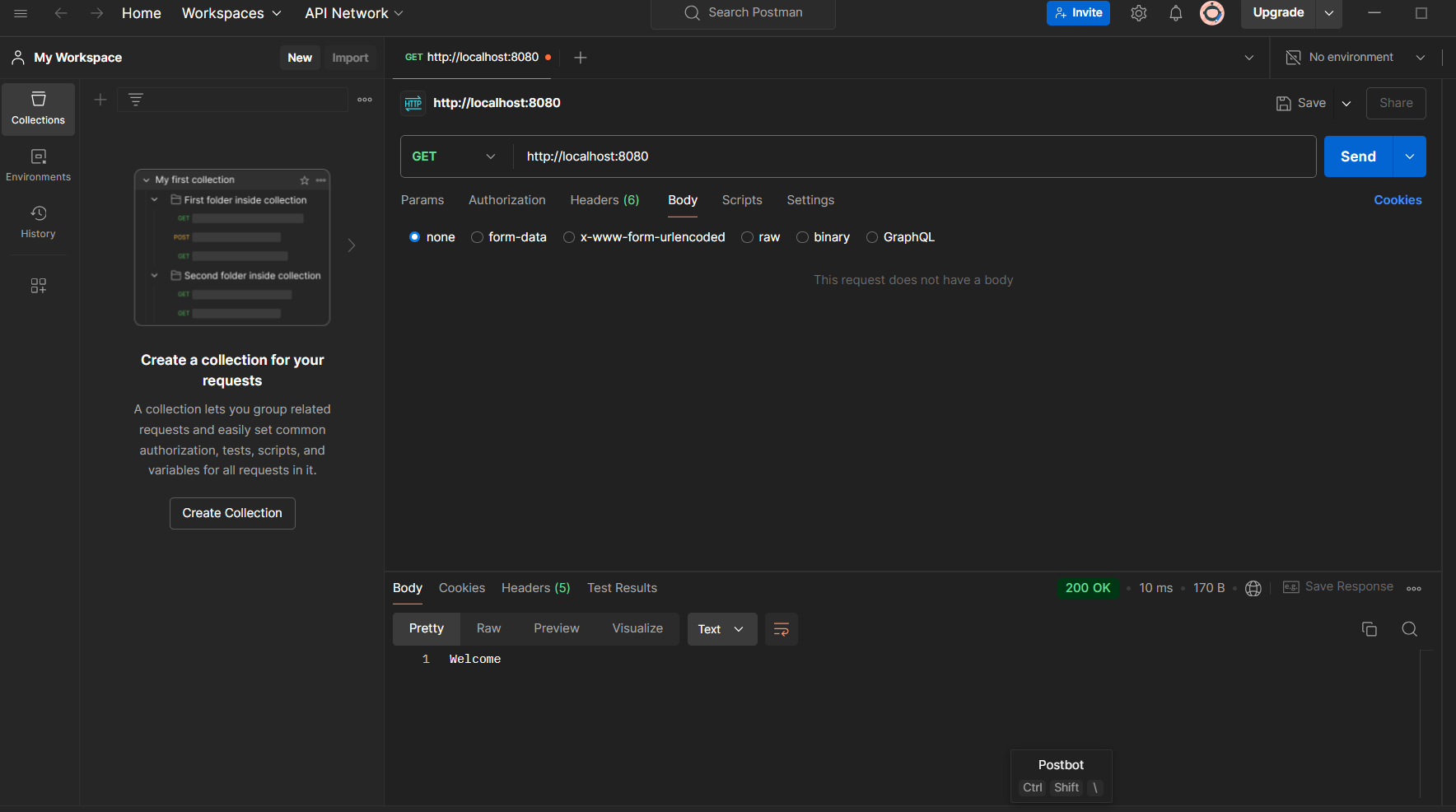
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQLDialect

**Now run the code:**

Right click to your project🡪Run as🡪Java Application🡪Search your project🡪Click on OK.

Go to the POSTMAN🡪Set the request GET🡪Enter the URL

http://localhost:8080/



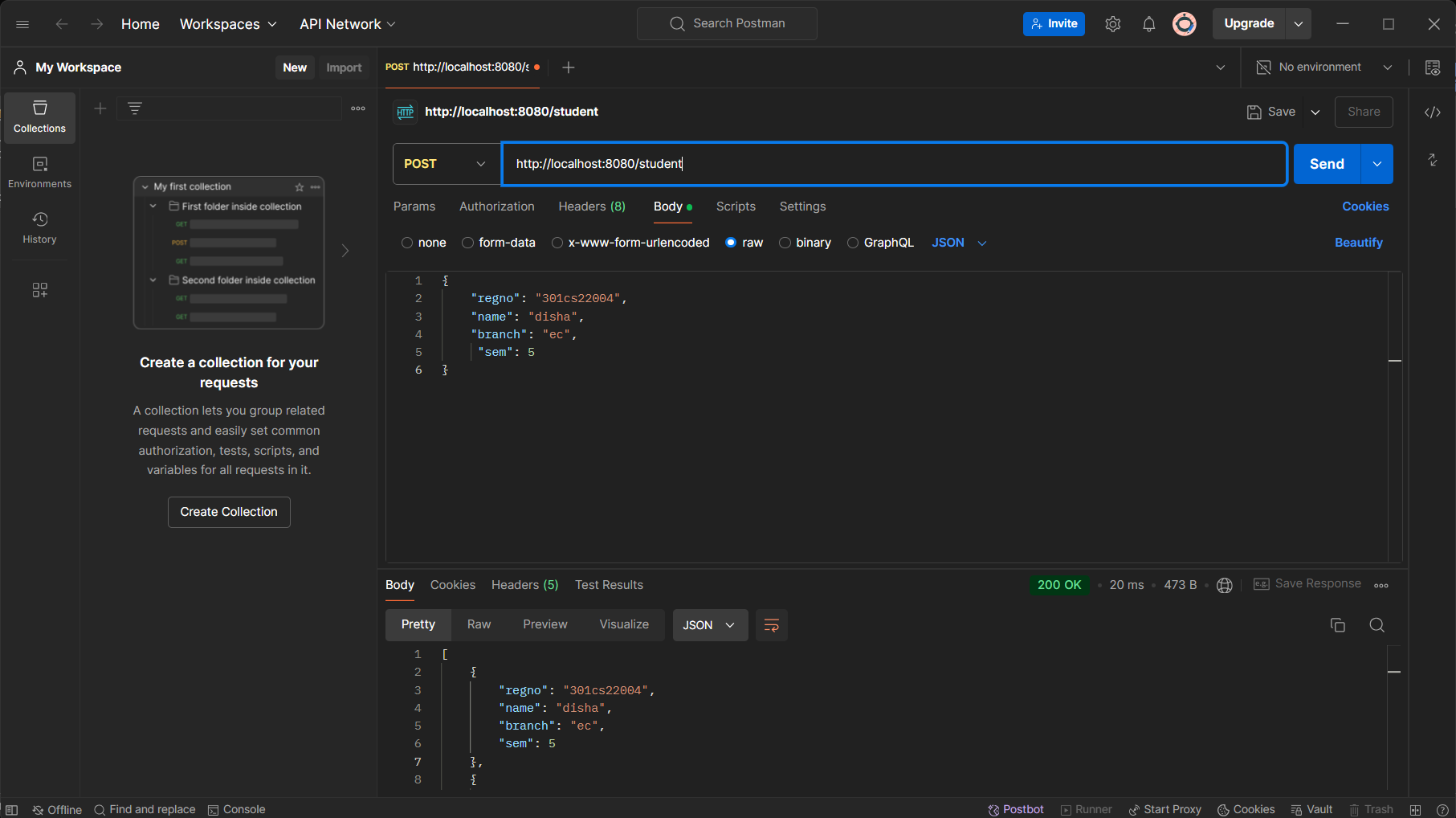
Set the url to GET request 🡪 <http://localhost:8080/students>

Click on send.

You will receive an empty table

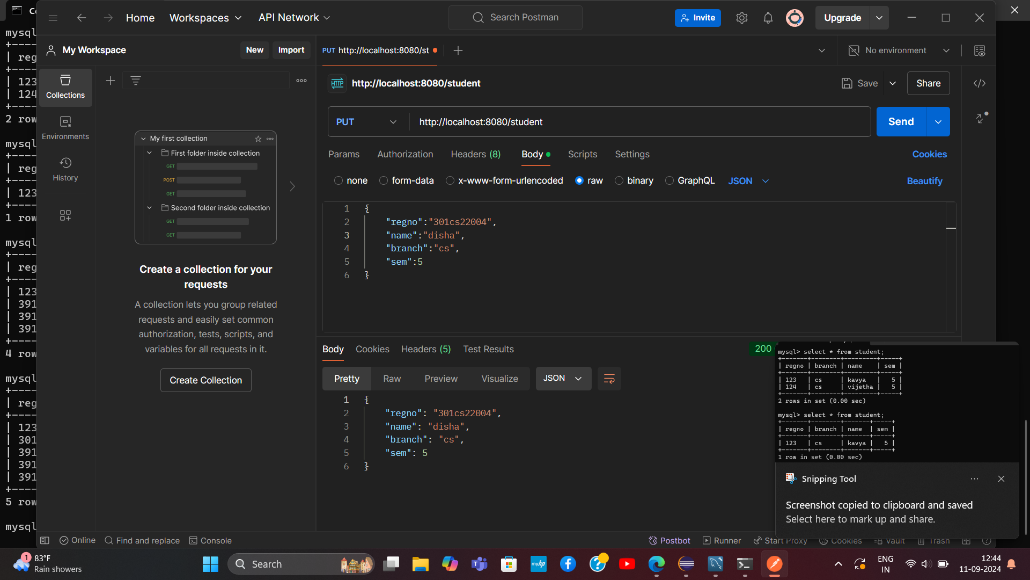
Set the request in POST🡪Body-select row🡪Write students details to insert into student table🡪Click on send

<http://localhost:8080/student>



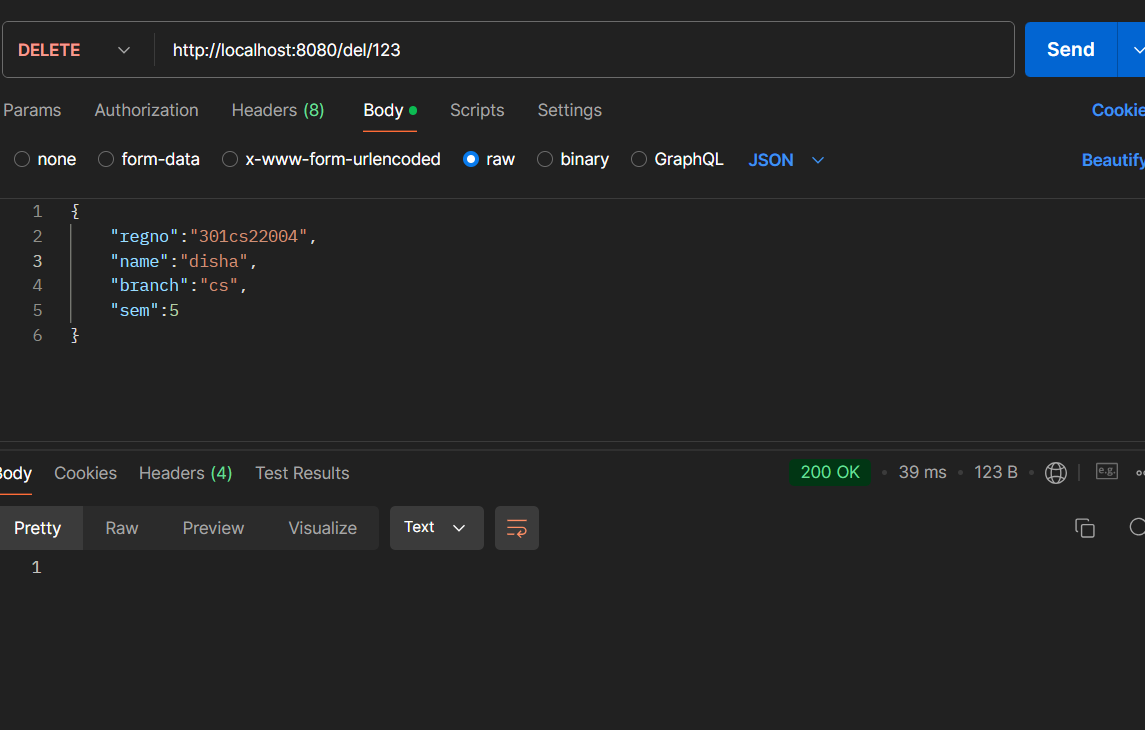
Set the request in PUT🡪Do the changes in name🡪click on send

<http://localhost:8080/student>

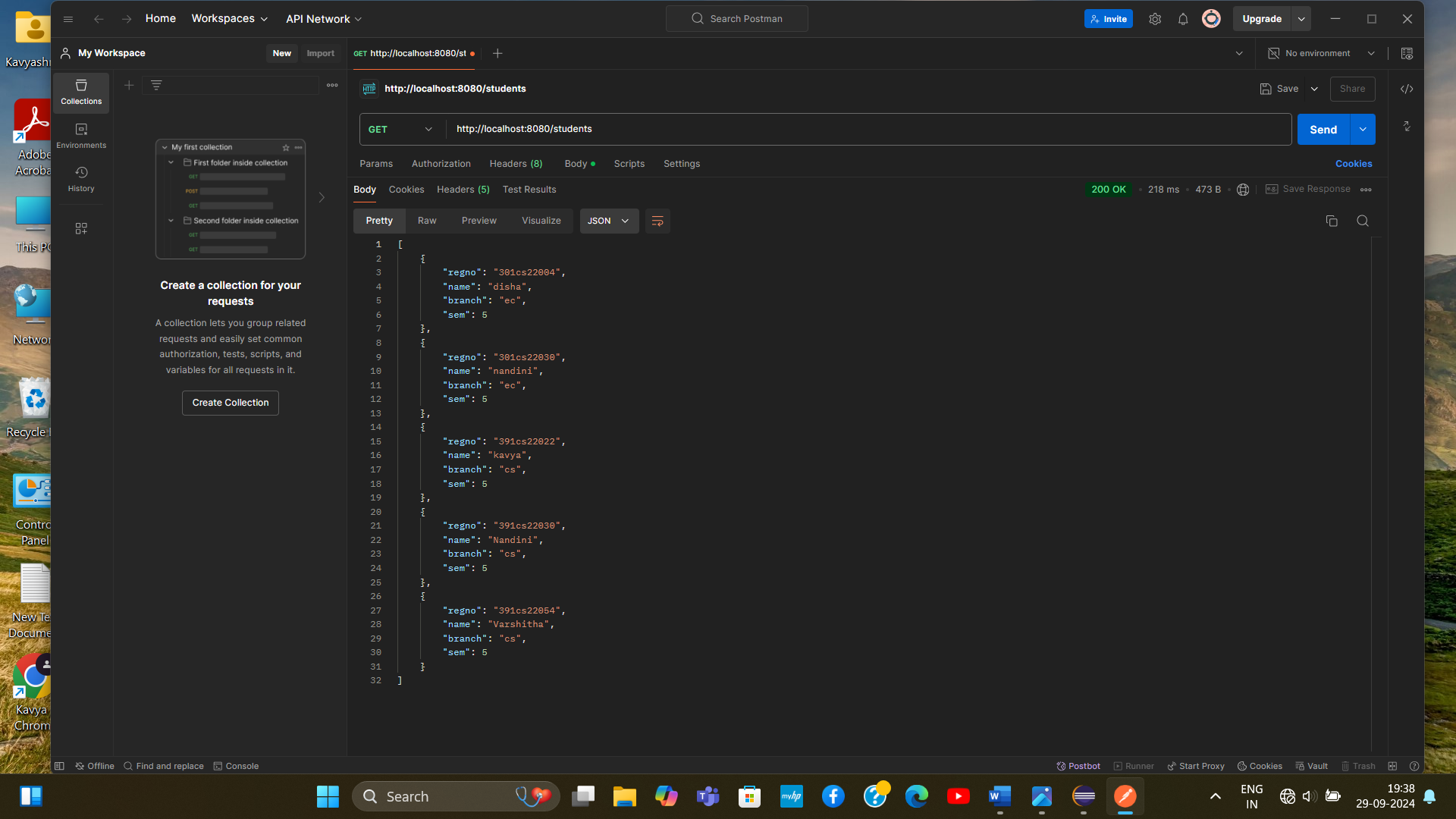


Set the request in Delete🡪To delete specify the regno

<http://localhost:8080/del/123>



Now we can see the student details after insertion in GET request🡪 <http://localhost:8080/students>



Step15: Create another department class inside com.fsd24\_25.CrudJPa.Entity

**Department.java:**

package com.fsd24\_25.CrudJPa.Entity;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.Table;

@Entity

@Table(name="department")

public class department {

@Id

@Column(name="deptid")

private String deptid;

private String dname;

private String hod;

public department() {}

public department(String deptid, String dname, String hod) {

super();

this.deptid = deptid;

this.dname = dname;

this.hod = hod;

}

public String getDeptid() {

return deptid;

}

public void setDeptid(String deptid) {

this.deptid = deptid;

}

public String getDname() {

return dname;

}

public void setDname(String dname) {

this.dname = dname;

}

public String getHod() {

return hod;

}

public void setHod(String hod) {

this.hod = hod;

}

@Override

public String toString() {

return ("{\"DeptId\":\""+deptid+"\",\n\"Dname\":\""+dname+"\",\n\"Hod\":\""+hod+"\"}");

}

}

Add the code to the service and controller and repository class:

**StudCtrl.java:**

package com.fsd24\_25.CrudJPa.Controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.DeleteMapping;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.PutMapping;

import org.springframework.web.bind.annotation.RequestBody;

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

import com.fsd24\_25.CrudJPa.Dto.studDetail;

import com.fsd24\_25.CrudJPa.Entity.student;

import com.fsd24\_25.CrudJPa.Service.StudService;

@RestController

public class StudCtrl {

@Autowired

private StudService s;

@GetMapping("/students")

public List<student> getStudentList(){

return s.getAllStudents();

}

@GetMapping("/")

public String disp(){

return "Welcome";

}

@DeleteMapping("/del/{regno}")

public void delStud(@PathVariable String regno) {

s.deleteStudent(regno);

}

@PostMapping("/student")

public student insertStud(@RequestBody student st){

return s.insertStudent(st);

}

@PutMapping("/student")

public student updateStud(@RequestBody student st){

return s.updateStudent(st);

}

@GetMapping("/details")

private List<studDetail> dispstudDetails() {

return s.getstudDetail();

}

}

**StudentRepo.java:**

package com.fsd24\_25.CrudJPa.Repository;

import java.util.List;

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

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

import com.fsd24\_25.CrudJPa.Entity.Student;

public interface StudentRepo extends JpaRepository<Student, String> {

@Query(value = "SELECT s.regno, s.name, s.sem, d.dname, d.hod " +

"FROM Student s, department d " +

"WHERE s.branch = d.deptid", nativeQuery = true)

public List<Object[]> getstudDetails();

}

**StudService.java:**

package com.fsd24\_25.CrudJPa.Service;

import java.util.ArrayList;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service

import com.fsd24\_25.CrudJPa.Dto.studDetail;

import com.fsd24\_25.CrudJPa.Entity.student;

import com.fsd24\_25.CrudJPa.Repository.StudentRepo;

@Service

public class StudService {

@Autowired

private StudentRepo StudRep ;

public List<student> getAllStudents(){

return StudRep.findAll();

}

public student insertStudent(student st) {

StudRep.save(st);

return st;

}

public student updateStudent(student s) {

StudRep.save(s);

return s;

}

public void deleteStudent(String id) {

student s = StudRep.getReferenceById(id);

StudRep.delete(s);

}

public List<studDetail> getstudDetail() {

List<studDetail> detail\_list=new ArrayList<>();

List<Object[]> result= StudRep.getstudDetail();

for (Object[] row : result){

studDetail d=new studDetail();

d.setRegno((String)row[0]);

d.setName((String)row[1]);

d.setSem((int)row[2]);

d.setBranch((String)row[3]);

d.setHod((String)row[4]);

detail\_list.add(d);

}

return detail\_list;

}

}

Step16: Create another package com.fsd24\_25.CrudJPa.Dto

Step17: Create a studDetail class inside com.fsd24\_25.CrudJPa.Dto

**studDetail.java:**

package com.fsd24\_25.CrudJPa.Dto;

import org.springframework.stereotype.Component;

*@Component*

public class studDetail {

private String regno;

private String name;

private String branch;

private int sem;

private String hod;

public studDetail () {}

public studDetail(String regno, String name, String branch, int sem, String hod) {

super();

this.regno = regno;

this.name = name;

this.branch = branch;

this.sem = sem;

this.hod = hod;

}

public String getRegno() {

return regno;

}

public void setRegno(String regno) {

this.regno = regno;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getBranch() {

return branch;

}

public void setBranch(String branch) {

this.branch = branch;

}

public int getSem() {

return sem;

}

public void setSem(int sem) {

this.sem = sem;

}

public String getHod() {

return hod;

}

public void setHod(String hod) {

this.hod = hod;

}

*@Override*

public String toString() {

return ("{\n\"regno\":\""+this.regno+"\",\n\"name\":\""+this.name+"\",\n\"branch\":"+this.branch+",\n\"sem\":"+this.sem+"\n\"HOD\":\""+this.hod+"\"\n}");

}

}

**Now run the code:**

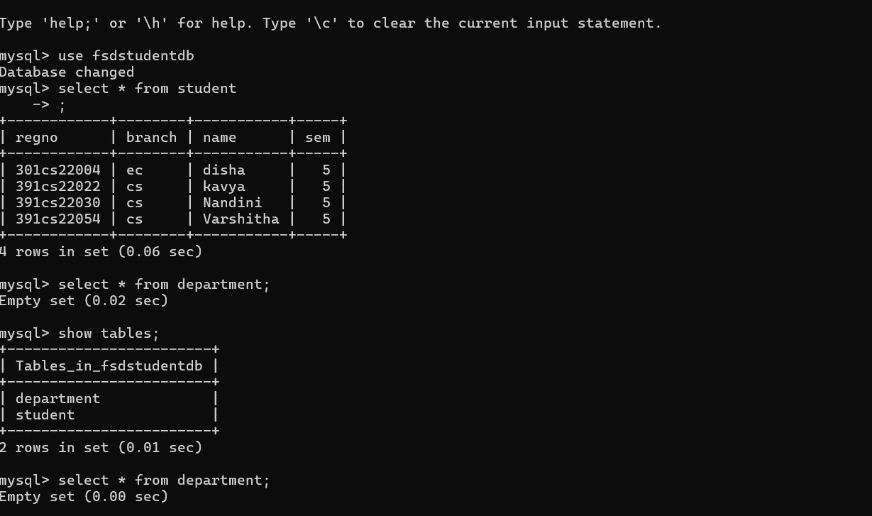
Right click to your project🡪Run as🡪Java Application🡪Search your project🡪Click on OK.

Go to the POSTMAN🡪Set the request to GET🡪

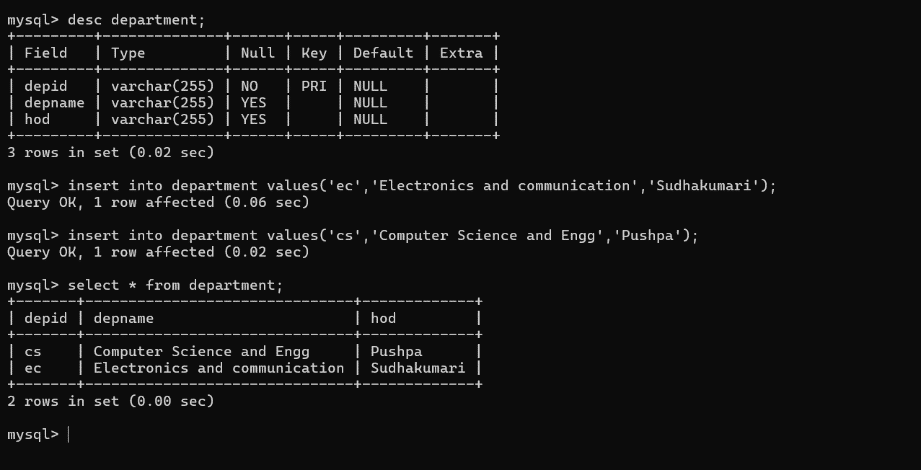
<http://localhsot:8080/details>

You will receive an empty table

We can see our output in command prompt also:



Now to insert the department details🡪Go to the command prompt and give the command as follows:



Now go to the POSTMAN🡪set the reuest to GET and see the output.

http://localhost:8080/details

