**Pom.xml**

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<parent>

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

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

<version>2.7.10</version>

<relativePath /> <!-- lookup parent from repository -->

</parent>

<groupId>com.cdac</groupId>

<artifactId>SB-Many2One-Mapping-Project</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>SB-Many2Many-Mapping-Project</name>

<description>Spring Boot with Many2Many mapping</description>

<properties>

<java.version>11</java.version>

</properties>

<dependencies>

<dependency>

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

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

</dependency>

<dependency>

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

<artifactId>spring-boot-starter-thymeleaf</artifactId>

</dependency>

<dependency>

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

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

</dependency>

<dependency>

<groupId>com.mysql</groupId>

<artifactId>mysql-connector-j</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

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

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

<dependency>

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

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

**Application.properties**

# JDBC PROPERTIES

spring.datasource.url=jdbc:mysql://localhost:3306/pune?useSSL=false&allowPublicKeyRetrieval=true

spring.datasource.username=root

spring.datasource.password=cdac

# HIBERNATE PROPERTIES

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL5Dialect

spring.jpa.hibernate.ddl-auto=update

## Show sql query.

spring.jpa.show-sql=true

**Subject.java**

package com.cdac.model;

import java.util.HashSet;

import java.util.Set;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.ManyToMany;

import javax.persistence.Table;

@Entity

@Table(name = "subject\_table")

public class Subject {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int subject\_Id;

private String subject\_name;

private String subject\_department;

@ManyToMany(mappedBy = "subjects")

private Set<Student> students = new HashSet<>();

public Subject() {

}

public Subject(int subject\_Id) {

this.subject\_Id = subject\_Id;

}

public Subject(int subject\_Id, String subject\_name, String subject\_department) {

super();

this.subject\_Id = subject\_Id;

this.subject\_name = subject\_name;

this.subject\_department = subject\_department;

}

public int getSubject\_Id() {

return subject\_Id;

}

public void setSubject\_Id(int subject\_Id) {

this.subject\_Id = subject\_Id;

}

public String getSubject\_name() {

return subject\_name;

}

public void setSubject\_name(String subject\_name) {

this.subject\_name = subject\_name;

}

public String getSubject\_department() {

return subject\_department;

}

public void setSubject\_department(String subject\_department) {

this.subject\_department = subject\_department;

}

public Set<Student> getStudents() {

return students;

}

public void setStudents(Set<Student> students) {

this.students = students;

}

@Override

public String toString() {

return subject\_Id + " " + subject\_name + " " + subject\_department + " " + students;

}

}

**Student.java**

package com.cdac.model;

import java.util.HashSet;

import java.util.Set;

import javax.persistence.CascadeType;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.JoinColumn;

import javax.persistence.JoinTable;

import javax.persistence.ManyToMany;

import javax.persistence.Table;

@Entity

@Table(name="student\_table")

public class Student {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int student\_Id;

private String student\_FirstName;

private String student\_LastName;

@ManyToMany(cascade = CascadeType.ALL)

@JoinTable(name = "student\_subject", joinColumns = {@JoinColumn(name="student\_Id")},

inverseJoinColumns = {@JoinColumn(name="subject\_Id")})

private Set<Subject> subjects = new HashSet<>();

public Student() {

}

public Student(int student\_Id) {

this.student\_Id = student\_Id;

}

public Student(int student\_Id, String student\_FirstName, String student\_LastName) {

this.student\_Id = student\_Id;

this.student\_FirstName = student\_FirstName;

this.student\_LastName = student\_LastName;

}

public int getStudent\_Id() {

return student\_Id;

}

public void setStudent\_Id(int student\_Id) {

this.student\_Id = student\_Id;

}

public String getStudent\_FirstName() {

return student\_FirstName;

}

public void setStudent\_FirstName(String student\_FirstName) {

this.student\_FirstName = student\_FirstName;

}

public String getStudent\_LastName() {

return student\_LastName;

}

public void setStudent\_LastName(String student\_LastName) {

this.student\_LastName = student\_LastName;

}

public Set<Subject> getSubjects() {

return subjects;

}

public void setSubjects(Set<Subject> subjects) {

this.subjects = subjects;

}

@Override

public String toString() {

return student\_Id + " " + student\_FirstName + " " + student\_LastName + " " + subjects;

}

}

**StudentDao.java**

package com.cdac.dao;

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

import org.springframework.stereotype.Repository;

import com.cdac.model.Post;

@Repository

public interface StudentDao extends JpaRepository<Student, Integer>{

}

**SubjectDao.java**

package com.cdac.dao;

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

import org.springframework.stereotype.Repository;

import com.cdac.model.Comment;

@Repository

public interface SubjectDao extends JpaRepository<Subject, Integer>{

}

**StudentService.java**

package com.cdac.service;

import java.util.List;

import com.cdac.model.Student;

public interface StudentService {

void add\_Student(Student student);

void update\_Student(Student student);

void delete\_Student\_By\_Id(int id);

Student get\_Student\_By\_Id(int id);

List<Student> get\_All\_Student();

}

**StudentServiceImpl.java**

package com.cdac.service;

import java.util.ArrayList;

import java.util.List;

import java.util.Optional;

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

import org.springframework.stereotype.Service;

import com.cdac.dao.StudentDao;

import com.cdac.model.Student;

@Service

public class StudentServiceImpl implements StudentService {

@Autowired

private StudentDao studentDao;

@Override

public void add\_Student(Student movie) {

studentDao.save(movie);

}

@Override

public void update\_Student(Student movie) {

studentDao.save(movie);

}

@Override

public void delete\_Student\_By\_Id(int id) {

studentDao.deleteById(id);

}

@Override

public Student get\_Student\_By\_Id(int id) {

Optional<Student> opt = studentDao.findById(id);

if (opt.isPresent()) {

return opt.get();

}

return null;

}

@Override

public List<Student> get\_All\_Student() {

Iterable<Student> itr = studentDao.findAll();

List<Student> lst = new ArrayList<Student>();

itr.forEach(ele -> lst.add(ele));

return lst;

}

}

**SubjectService.java**

package com.cdac.service;

import java.util.List;

import com.cdac.model.Subject;

public interface SubjectService {

void add\_Subject(Subject subject);

void update\_Subject(Subject subject);

void delete\_Subject\_By\_Id(int id);

Subject get\_Subject\_By\_Id(int id);

List<Subject> get\_All\_Subject();

}

**SubjectServiceImpl.java**

package com.cdac.service;

import java.util.ArrayList;

import java.util.List;

import java.util.Optional;

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

import org.springframework.stereotype.Service;

import com.cdac.dao.SubjectDao;

import com.cdac.model.Subject;

@Service

public class SubjectServiceImpl implements SubjectService {

@Autowired

private SubjectDao subjectDao;

@Override

public void add\_Subject(Subject subject) {

subjectDao.save(subject);

}

@Override

public void update\_Subject(Subject subject) {

subjectDao.save(subject);

}

@Override

public void delete\_Subject\_By\_Id(int id) {

subjectDao.deleteById(id);

}

@Override

public Subject get\_Subject\_By\_Id(int id) {

Optional<Subject> opt = subjectDao.findById(id);

if (opt.isPresent()) {

return opt.get();

}

return null;

}

@Override

public List<Subject> get\_All\_Subject() {

Iterable<Subject> itr = subjectDao.findAll();

List<Subject> lst = new ArrayList<Subject>();

itr.forEach(ele -> lst.add(ele));

return lst;

}

}

**StudentController.java**

package com.cdac.controller;

import java.util.List;

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

import org.springframework.http.HttpStatus;

import org.springframework.http.ResponseEntity;

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.RequestParam;

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

import com.cdac.model.Student;

import com.cdac.service.StudentService;

@RestController

public class StudentController {

@Autowired

private StudentService studentService;

@PostMapping(value = { "/student" })

public String add\_Student(@RequestBody Student student) {

studentService.add\_Student(student);

return "Student added successfully";

}

@PutMapping(value = { "/student" })

public String update\_Student(@RequestBody Student student) {

studentService.update\_Student(student);

return "Student updated successfully";

}

@DeleteMapping(value = { "/student" })

public String delete\_Student(@RequestParam int id) {

System.out.println("first delete");

studentService.delete\_Student\_By\_Id(id);

return "Student deleted successfully";

}

@DeleteMapping(value = { "/student/{id}" })

public String delete\_Student1(@PathVariable int id) {

System.out.println("second delete");

studentService.delete\_Student\_By\_Id(id);

return "Student deleted successfully";

}

@GetMapping(value = { "/student/{id}" })

public Student get\_Student\_By\_Id(@PathVariable int id) {

return studentService.get\_Student\_By\_Id(id);

}

@GetMapping(value = { "/student" })

public List<Student> get\_All\_Student() {

return studentService.get\_All\_Student();

}

}

**SBMany2OneMappingProject.java**

package com.cdac;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SBMany2ManyMappingProjectApplication {

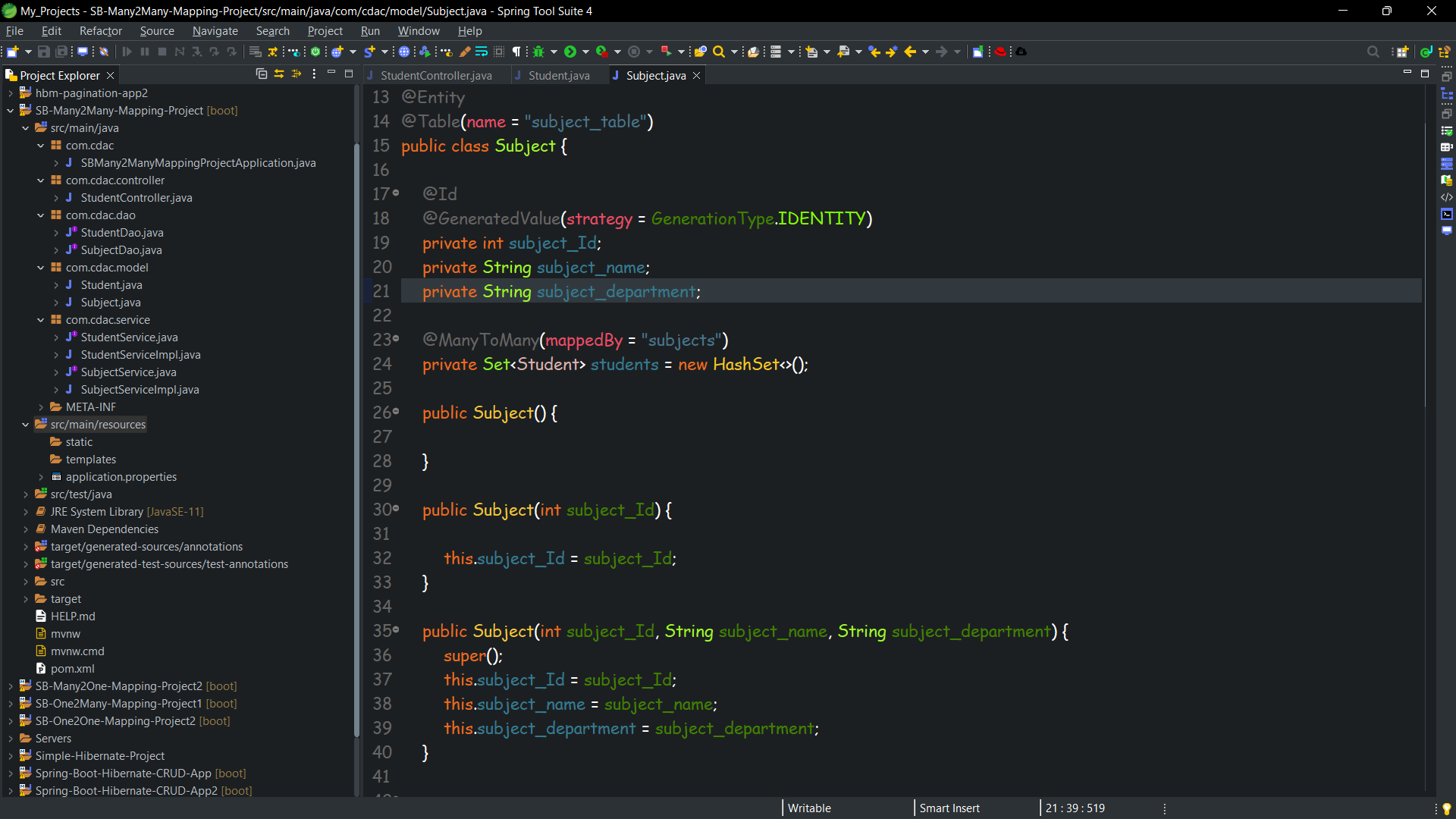
public static void main(String[] args) {

SpringApplication.run(SBMany2OneMappingProjectApplication.class, args);

}

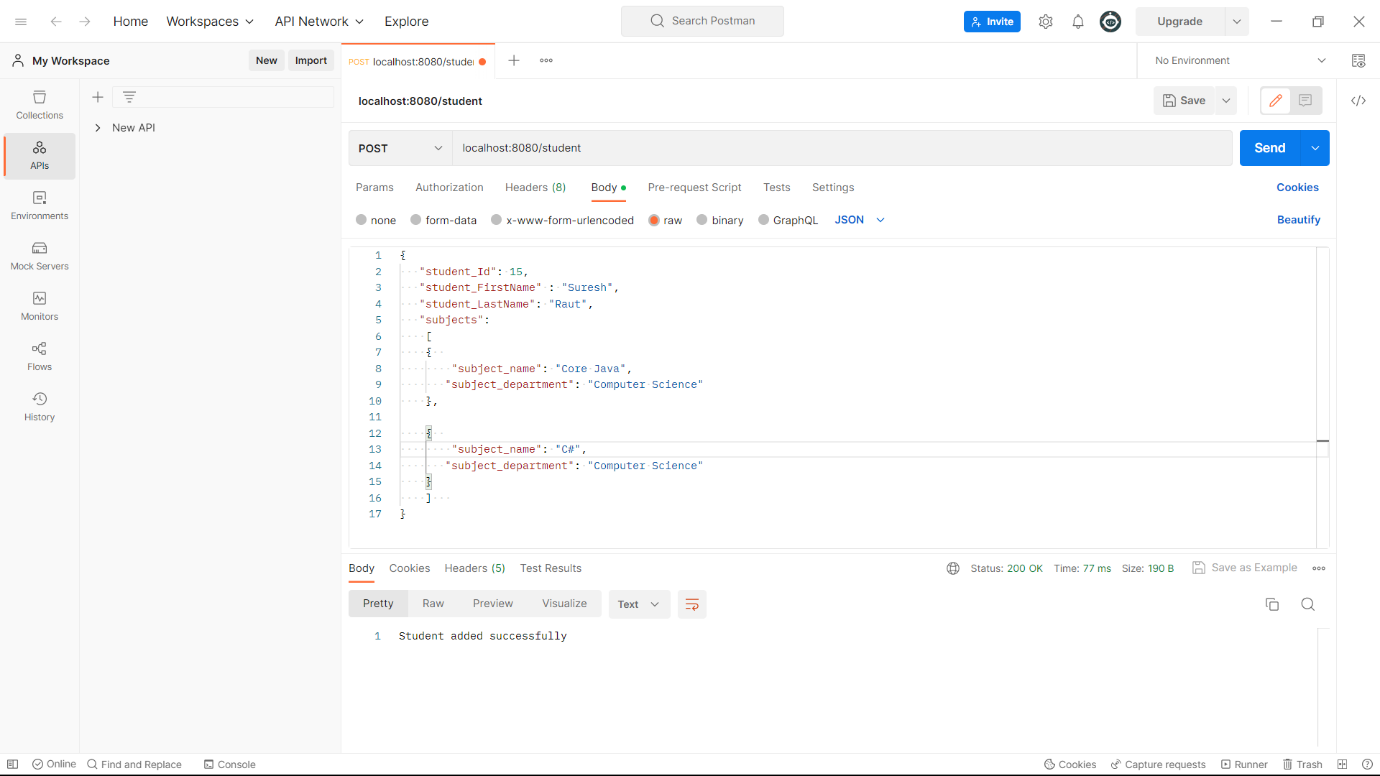
}

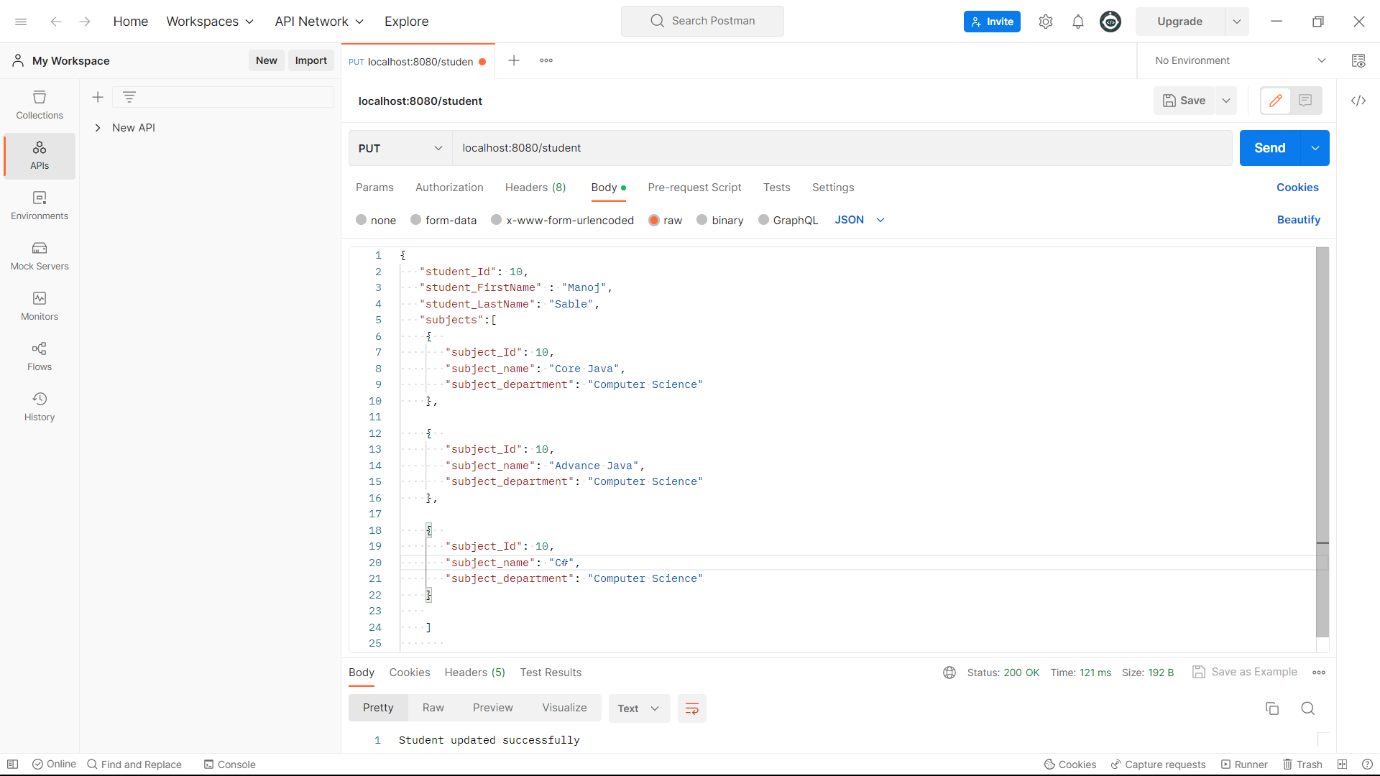
**Project Structrure**



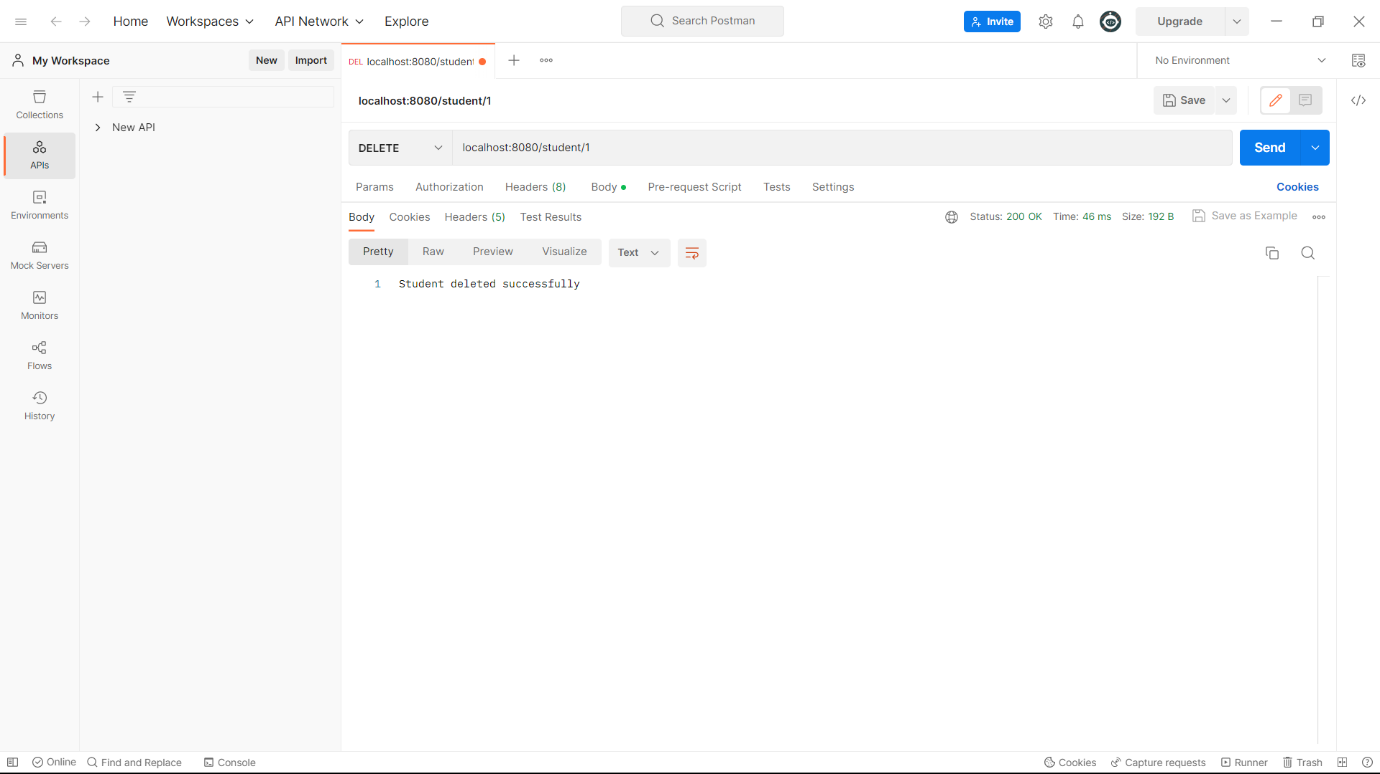
**PostMan and Database ScreenShots**

**Insert**

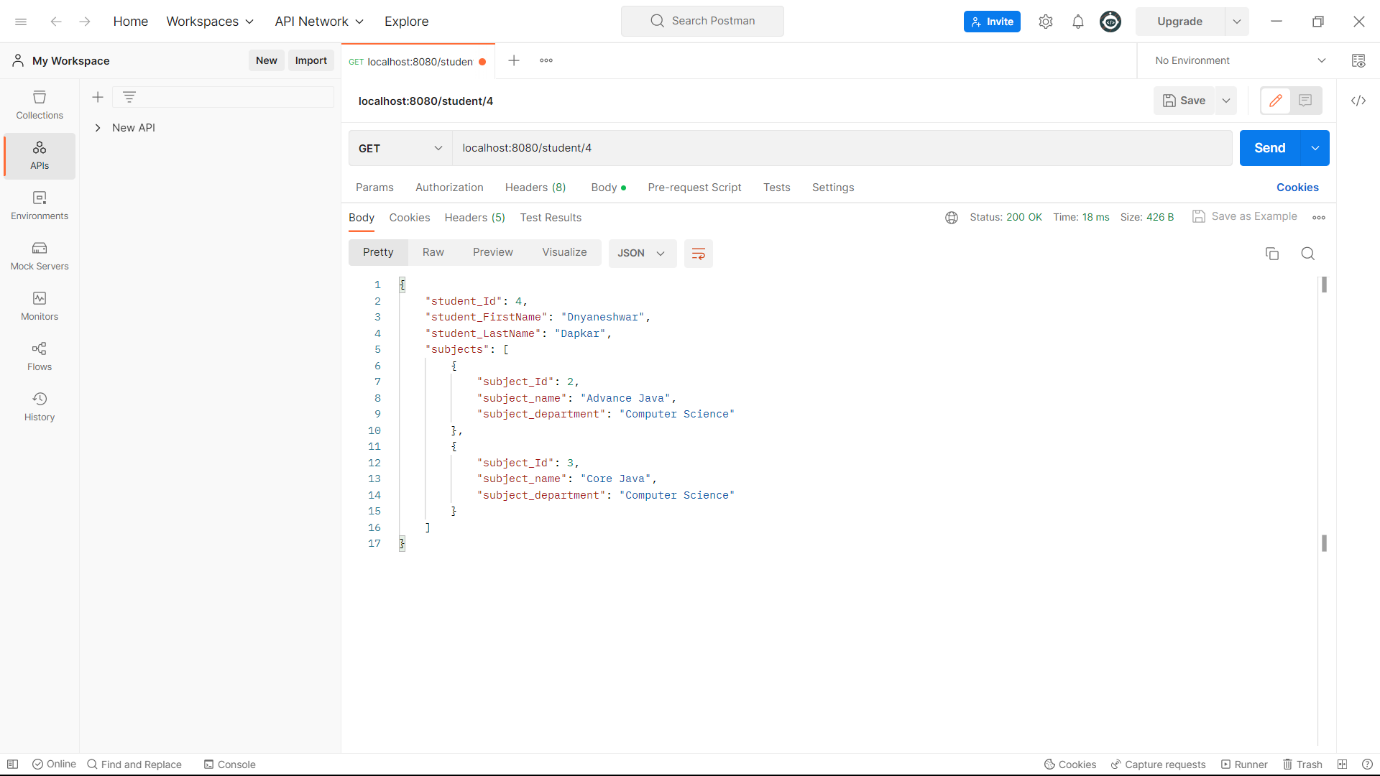


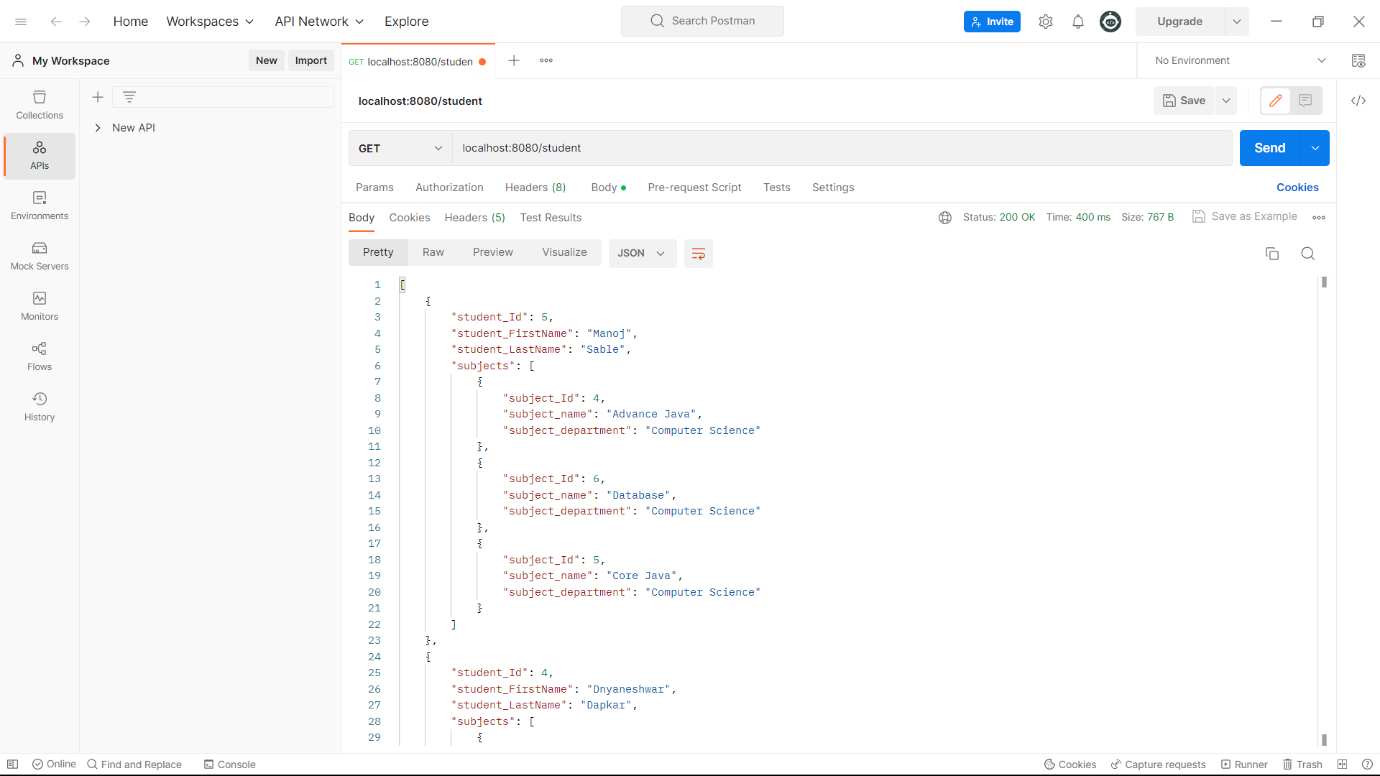
**Update** 

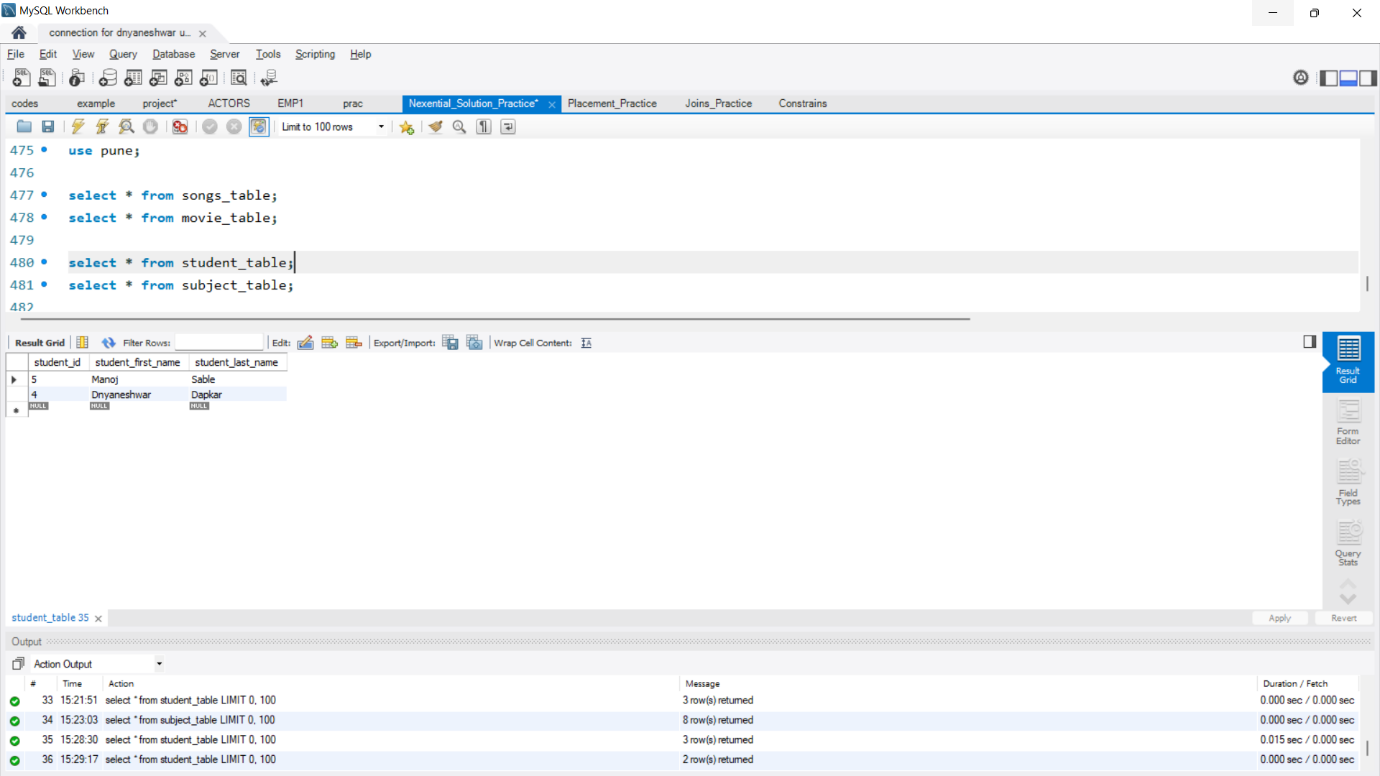
**delete**



**select by id**



**select all** 

**Table Student** 

**Table Subject** 