**Create REST controller for CRUD operations with JPA**

Steps:

* Open spring initializer in google and select project and language and version then add two dependency like Spring boot Devtools and spring web 🡪click on Generate.
* Then extract the downloaded zip file to our main project.
* Then open eclipse🡪import🡪maven🡪existing maven projects🡪give our extracted file path then click pom.xml🡪Finish.
* Create 4 package in the main Java file inside Main Package where Application file is present

1]com.fsd.crud.crudDemo.Service

2] com.fsd.crud.crudDemo.Controller

3] com.fsd.crud.crudDemo.Entity

4] com.fsd.crud.crudDemo.Repository.

* In com.fsd.crud.crudDemo.Controller Floder create Mycontroller.java file
* In com.fsd.crud.crudDemo.Entity Floder create Student.java file
* In com.fsd.crud.crudDemo.Repository Floder create StudentDao.java file
* In com.fsd.crud.crudDemo.Service Floder create StudentService.java and StudentServiceImplement.java file.

**In Mycontroller.java file.**

package com.fsd.crud.crudDemo.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.fsd.crud.crudDemo.Entity.Student;

import com.fsd.crud.crudDemo.Service.StudentService;

@RestController

public class Mycontroller {

@Autowired

private StudentService studentservice;

@GetMapping("/")

public String show1() {

return("<h1>Welcome to vivekananda polytechnic</h1>");

}

@GetMapping("student")

private List<Student>show(){

return this.studentservice.Display();

}

@PostMapping("student")

private Student show2(@RequestBody Student st)

{ return this.studentservice.addStudent(st);

}

@PutMapping("student")

private Student show3(@RequestBody Student st)

{ return this.studentservice.updateStudent(st);

}

@DeleteMapping("student/{studentreg}")

private void show4(@PathVariable int studentreg)

{ this.studentservice.deleteStudent(studentreg);

}

}

**In Student.java file**

package com.fsd.crud.crudDemo.Entity;

import jakarta.persistence.Entity;

import jakarta.persistence.GeneratedValue;

import jakarta.persistence.GenerationType;

import jakarta.persistence.Id;

import jakarta.persistence.Table;

@Entity

@Table(name="student")

public class Student {

@Id

//@GeneratedValue(strategy=GenerationType.AUTO)

int regNo;

String Name;

public int getRegNo() {

return regNo;

}

public void setRegNo(int regNo) {

this.regNo = regNo;

}

public String getName() {

return Name;

}

public void setName(String name) {

Name = name;

}

public Student(int regNo, String name) {

super();

this.regNo = regNo;

Name = name;

}

public Student() {

super();

}

@Override

public String toString() {

return "{ \"regNo\": " + regNo + ", \"Name\": \"" + Name + "\" }";

}

}

**In StudentDao.java**

package com.fsd.crud.crudDemo.Repository;

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

public interface StudentDao extends JpaRepository<com.fsd.crud.crudDemo.Entity.Student,Integer> { }

In StudentService.java

package com.fsd.crud.crudDemo.Service;

import java.util.List;

//import org.springframework.stereotype.Component;

import com.fsd.crud.crudDemo.Entity.Student;

//@Component

public interface StudentService {

public List<Student>Display();

public Student addStudent(Student st);

public Student updateStudent(Student st);

public void deleteStudent(int studentregno);

}

**In StudentServiceImplement.java file**

package com.fsd.crud.crudDemo.Service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import com.fsd.crud.crudDemo.Entity.Student;

import com.fsd.crud.crudDemo.Repository.StudentDao;

@Service

public class StudentServiceImplement implements StudentService {

@Autowired

private final StudentDao StudentDao;

StudentServiceImplement(StudentDao StudentDao){

this.StudentDao=StudentDao;

}

@Override

public List<Student>Display(){

return StudentDao.findAll();

}

@Override

public Student addStudent(Student st){

StudentDao.save(st);

return st;

}

@Override

public Student updateStudent(Student st){

StudentDao.save(st);

return st;

}

@Override

public void deleteStudent(int regno){

Student stud=StudentDao.getReferenceById(regno);

StudentDao.delete(stud);

} }

**In Application.properties**

server.port=8080

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

spring.datasource.username=root

spring.datasource.password=root

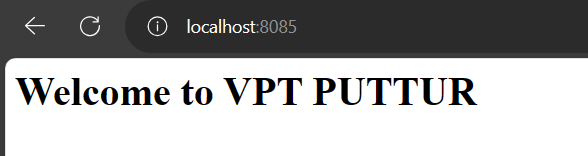
spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

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

* Then run our main file.
* Then enter URL in browser🡪http://localhost:8085

OUTPUT:



* Then run command prompt as a administrator
* Then run the database. To run database enter below command

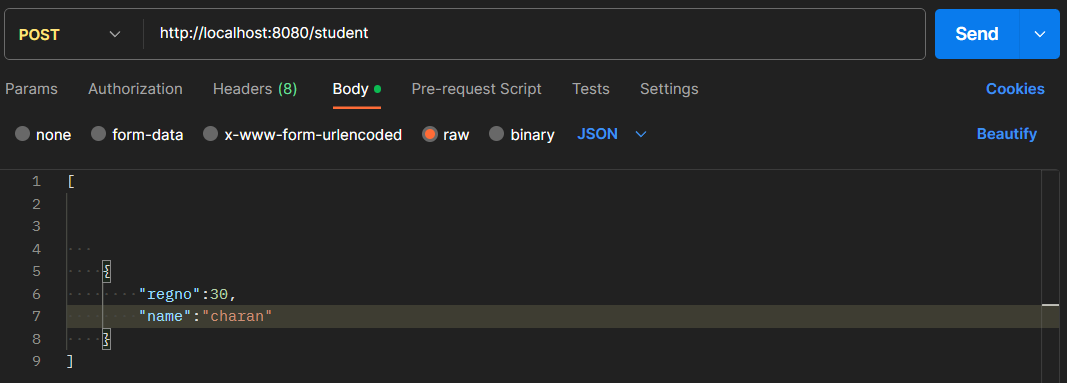
**Mysqld**

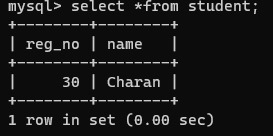
**Mysql -u root -p**

* Then enter password
* Then create database 🡪create database crudproj;
* use project1;
* Then go to postman and enter URL🡪 https://localhost:8080/student
* Use post method🡪body🡪raw🡪select JSON format and upload regno and name.
* {“regNo”:25,

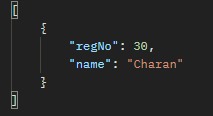
“name”: “Charan”

}

* Then click to send option.
* Then go to database🡪select\* from student;
* It displays student tables.



* Then use get method send the submit

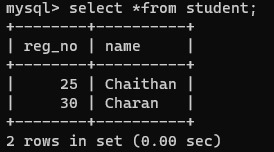


* Then use put method🡪Submit raw data in body using JSON format.

{“regNo”:25,

“name”: “Chaithan”}





* Then use delete method🡪Then enter student regno which we want to delete

**https://localhost:8080/student/156**

* Here 155 is students regNo.

