

## CRUID operation in normal postman

### 1. Student.java

```
package com.example.StudentExample.Model;

import jakarta.persistence.Entity;
import jakarta.persistence.Id;
import org.springframework.stereotype.Component;
@Entity
@Component
public class Student {
    @Id
    private int regno;
    private String name;
    private int marks;

    public int getRegno() {
        return regno;
    }

    public void setRegno(int regno) {
        this.regno = regno;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getMarks() {
        return marks;
    }

    public void setMarks(int marks) {
        this.marks = marks;
    }

    public Student(int regno, String name, int marks) {
        this.regno = regno;
        this.name = name;
        this.marks = marks;
    }

    public Student() {
    }
}
```

### 2. studentController

```
package com.example.StudentExample.Controller;

import com.example.StudentExample.Model.Student;
import com.example.StudentExample.Service.StudentService;
import org.springframework.beans.factory.annotation.Autowired;
```

```

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

import java.util.List;

@RestController
public class StudentController {
    @Autowired
    StudentService service;

    @GetMapping("/students")
    public List<Student> getAllStudentDetails() {
        return service.getAllStudents();
    }

    @GetMapping("students/{stid}")
    public Student getStudentById(@PathVariable int stid) {
        return service.getStudentById(stid);
    }

    @PostMapping("/students")
    public void addStudents(@RequestBody Student s) {
        service.addStudent(s);
    }

    @PutMapping("/students")
    public void updateStudent(@RequestBody Student s) {
        service.updateStudent(s);
    }

    @DeleteMapping("/students/{sid}")
    public void deleteStudentById(@PathVariable int sid) {
        service.deleteStudentById(sid);
    }
}

```

### 3. StudentService

```

package com.example.StudentExample.Service;

import com.example.StudentExample.Model.Student;
import org.springframework.stereotype.Service;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

@Service
public class StudentService {

    List<Student> students=new ArrayList<> (
        Arrays.asList(
            new Student(111,"Kumar",95),
            new Student(112,"sathish",90),
            new Student(113,"ramesh",85)
        ));

    public List<Student> getAllStudents() {
        return students;
    }

    public Student getStudentById(int stid) {
        return students.stream()
            .filter(p->p.getRegno()==stid)
            .findFirst();
    }
}

```

```

        .get();
    }

    public void addStudent(Student s) {
        students.add(s);
    }

    public void updateStudent(Student s) {
        int index=0;
        for(int i=0;i<students.size();i++){
            if( students.get(i).getRegno()==s.getRegno())
                index=i;
        }

        students.set(index,s);
    }

    public void deleteStudentById(int sid) {
        int index=0;
        for(int i=0;i<students.size();i++){
            if( students.get(i).getRegno()==sid)
                index=i;
        }

        students.remove(index);
    }
}

```

## **CRUID operation in h2-console.**

### **1. Application.Properties**

spring.application.name=StudentExample  
 spring.datasource.url=jdbc:h2:mem:testdb  
 spring.datasource.driverClassName=org.h2.Driver  
 spring.jpa.show-sql=true

### **2. Student.java**

```

import jakarta.persistence.Entity;
import jakarta.persistence.Id;
import org.springframework.stereotype.Component;

@Entity
@Component
public class Student {
    @Id
    private int regno;
    private String name;
    private int marks;

    public int getRegno() {
        return regno;
    }

    public void setRegno(int regno) {
        this.regno = regno;
    }
}

```

```

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getMarks() {
        return marks;
    }

    public void setMarks(int marks) {
        this.marks = marks;
    }

    public Student(int regno, String name, int marks) {
        this.regno = regno;
        this.name = name;
        this.marks = marks;
    }

    public Student() {
    }

```

### 3. StudentController

```

package com.example.StudentExample.Controller;

import com.example.StudentExample.Model.Student;
import com.example.StudentExample.Service.StudentService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;

import java.util.List;

@RestController
public class StudentController {
    @Autowired
    StudentService service;

    @GetMapping("/students")
    public List<Student> getAllStudentDetails() {
        return service.getAllStudents();
    }

    @GetMapping("students/{stid}")
    public Student getStudentById(@PathVariable int stid) {
        return service.getStudentById(stid);
    }

    @PostMapping("/students")
    public void addStudents(@RequestBody Student s) {
        service.addStudent(s);
    }

    @PutMapping("/students")
    public void updateStudent(@RequestBody Student s) {
        service.updateStudent(s);
    }
}

```

```

    @DeleteMapping("/students/{sid}")
    public void deleteStudentById(@PathVariable int sid) {
        service.deleteStudentById(sid);
    }

```

#### 4. StudentRepository.java(interface)

```

com.example.StudentExample.Repository;

import com.example.StudentExample.Model.Student;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

@Repository
public interface StudentRepository extends JpaRepository<Student,Integer> {

}

```

#### 5. StudentServices.\

```

package com.example.StudentExample.Service;

import com.example.StudentExample.Model.Student;
import com.example.StudentExample.Repository.StudentRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

@Service
public class StudentService {

    @Autowired
    StudentRepository repo;

    public List<Student> getAllStudents() {

        return repo.findAll();
    }

    public Student getStudentById(int stid) {
        return repo.findById(stid).orElse(new Student());
    }

    public void addStudent(Student s) {
        repo.save(s);
    }

    public void updateStudent(Student s) {
        repo.save(s);
    }

    public void deleteStudentById(int sid) {
        repo.deleteById(sid);
    }
}

```

```
}  
}
```

web, devtool h2 lom

## **CRUID operation in MongoDB.**

### **1.Student.java**

```
package com.example.mongoDbInSpringBoot.model;  
  
import org.springframework.data.mongodb.core.mapping.Document;  
  
@Document(collection = "StudentDB")  
public class Student {  
    private int id;  
    private String name;  
    private int regono;  
    private int sem;  
  
    public Student() {  
    }  
  
    public Student(int id, String name, int regono, int sem) {  
        this.id = id;  
        this.name = name;  
        this.regono = regono;  
        this.sem = sem;  
    }  
  
    public int getId() {  
        return id;  
    }  
  
    public void setId(int id) {  
        this.id = id;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public void setName(String name) {  
        this.name = name;  
    }  
  
    public int getRegono() {  
        return regono;  
    }  
  
    public void setRegono(int regono) {  
        this.regono = regono;  
    }  
  
    public int getSem() {  
        return sem;  
    }  
  
    public void setSem(int sem) {  
        this.sem = sem;  
    }  
}
```

## **2. StudentController.java**

```
package com.example.mongoDbInSpringBoot.controller;

import com.example.mongoDbInSpringBoot.model.Student;
import com.example.mongoDbInSpringBoot.repository.StudentRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;

import java.util.List;
import java.util.List;

@RestController
@RequestMapping("/Student")
public class StudentController {
    @Autowired
    private StudentRepository studentRepository;
    @PostMapping("/post")
    public String addData(@RequestBody Student student)
    {
        try{
            this.studentRepository.save(student);
            return "data inserted success fully";

        } catch (Exception e){
            return "data not inserted";
        }
    }
    @GetMapping("/data")
    public List<Student> getData() {
        return studentRepository.findAll();
    }
    @DeleteMapping("/delete")
    public String deleteAllData()
    {
        try{
            this.studentRepository.deleteAll();
            return "data delete suceessfully";
        } catch (Exception e){
            return "data not deleted";
        }
    }
}
```

```

@DeleteMapping("/{id}")
public String deleteAllData(@PathVariable int id)
{
    try{
        this.studentRepository.deleteById(id);
        return "data deleted suceess fully";
    }catch (Exception e){
        return "not deleted data";
    }
}
@PutMapping("/{id}")
public String updateData(@RequestBody Student student,@PathVariable int id){
    try{
        this.studentRepository.save(student);
        return "data update successfully";
    }catch(Exception e )
    {
        return "data not updated";
    }
}
}
}

```

### 3. StudentRepositroy(in).

```

package com.example.mongoDbInSpringBoot.repository;

import com.example.mongoDbInSpringBoot.model.Student;
import org.springframework.data.mongodb.repository.MongoRepository;

public interface StudentRepository extends
MongoRepository<Student,Integer> {

}

```

### 4. ApplicationProperties.

```

spring.data.mongodb.host=localhost
spring.data.mongodb.port=27017
spring.mongodb.database=MYDB

```