

# Student Management Rest API

## 1. Pre given Data.

jdbc:h2:mem:testdb

SCHOOL

TEACHER

INFORMATION\_SCHEMA

Users

H2 2.3.232 (2024-08-11)

Run

Run Selected

Auto complete

Clear

SQL statement:

SELECT \* FROM TEACHER

SELECT \* FROM TEACHER;

ID	AGE	DOB	EXPERIENCE	QUALIFICATION	TEACHER_ADDRESS	TEACHER_NAME
1	45	1980-05-15	20	PHD	Hyderabad	Naga Chaitanya
2	50	1975-03-10	25	MSC	Mumbai	Jagapati Babu
3	38	1985-08-20	15	BTECH	Chennai	Ravi Teja
4	42	1981-11-30	18	MTECH	Bangalore	Maresh Babu
5	48	1977-04-05	22	PHD	Delhi	Prabhas
6	35	1986-07-14	12	MSC	Pune	Allu Arjun
7	40	1982-01-25	17	BSC	Kolkata	Ram Charan
8	55	1967-09-09	30	PHD	Chennai	Venkatesh

(8 rows, 1 ms)

Edit

Auto commit

Max rows: 1000

Auto complete

Off

Auto select

On

jdbc:h2:mem:testdb

SCHOOL

TEACHER

INFORMATION\_SCHEMA

Users

H2 2.3.232 (2024-08-11)

Run

Run Selected

Auto complete

Clear

SQL statement:

SELECT \* FROM SCHOOL

SELECT \* FROM SCHOOL;

ID	SCHOOL_ADDRESS	SCHOOL_NAME	TEACHER_ID
1	Hyderabad	Sunrise Public School	1
2	Mumbai	Greenwood High	2
3	Chennai	Springfield Academy	3
4	Bangalore	Maple Leaf School	4
5	Delhi	Crescent School	5
6	Pune	Oakridge International	6
7	Kolkata	Hillview School	7
8	Chennai	Lakewood School	8

(8 rows, 1 ms)

Edit

Auto commit

Max rows: 1000

Auto complete

Off

Auto select

On

## Implemented Endpoints

***@PostMapping("/addSchool")***

***@GetMapping("getAllSchools")***

***@PutMapping("/update/{id}")***

***@PatchMapping("/patch/{id}")***

***@DeleteMapping("/deleteSchool/{id}")***

***@GetMapping("/school/{schoolAddress}")***

***@DeleteMapping("/school/{schoolAddress}")***

***@GetMapping("/search")***

***@PostMapping("/addTeacher")***

***@GetMapping("getAllTeachers")***

***@PutMapping("/updateTeacher/{id}")***

***@PatchMapping("/patchTeacher/{id}")***

***@DeleteMapping("/deleteTeacher/{id}")***

***@GetMapping("/teachers/{qualification}")***

***@DeleteMapping("delete/{teacherAddress}")***

## 2. After (“/addSchool”)

The screenshot displays the Postman interface for a POST request to `http://localhost:8080/addSchool`. The left sidebar shows the 'Collections' and 'Environments' sections. The main panel is divided into tabs: Params, Authorization, Headers (8), Body, Scripts, and Settings. The 'Body' tab is selected, and the 'raw' radio button is chosen. The request body is a JSON object:

```
1 {  
2   "schoolName": "Jagadeesh Public School",  
3   "schoolAddress": "Hyderabad",  
4   "teacherId": "9"  
5 }
```

Below the request body, the 'Body' tab is selected in the response section, and the 'JSON' radio button is chosen. The response body is a JSON object:

```
1 {  
2   "id": 9,  
3   "schoolName": "Jagadeesh Public School",  
4   "schoolAddress": "Hyderabad",  
5   "teacherId": 9  
6 }
```



GET http://localhost:8080/ge

+

HTTP

http://localhost:8080/getAllSchools

GET

▼

http://localhost:8080/getAllSchools

Params

Authorization

Headers (8)

Body

Scripts

Settings

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

JSON ▼

```
1 {
2   "schoolName": "Jagadeesh Public School",
3   "schoolAddress": "Hyderabad",
4   "teacherId": "9"
5 }
```

Body

Cookies

Headers (5)

Test Results

↺

{ } JSON ▼

▶ Preview

🔗 Visualize

▼

```
1 [
2   {
3     "id": 1,
4     "schoolName": "Sunrise Public School",
5     "schoolAddress": "Hyderabad",
6     "teacherId": 1
7   },
8   {
9     "id": 2,
10    "schoolName": "Greenwood High",
11    "schoolAddress": "Mumbai",
12    "teacherId": 2
13  },
14  {
15    "id": 3,
16    "schoolName": "Springfield Academy",
17    "schoolAddress": "Chennai",
18    "teacherId": 3
19  },
```

GET http://localhost:8080/getAllSchools

HTTP

http://localhost:8080/getAllSchools

GET

http://localhost:8080/getAllSchools

Params

Authorization

Headers (8)

Body

Scripts

Settings

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

☒ JSON

```
1 {
2   "schoolName": "Jagadeesh Public School",
3   "schoolAddress": "Hyderabad",
4   "teacherId": "9"
5 }
```

Body

Cookies

Headers (5)

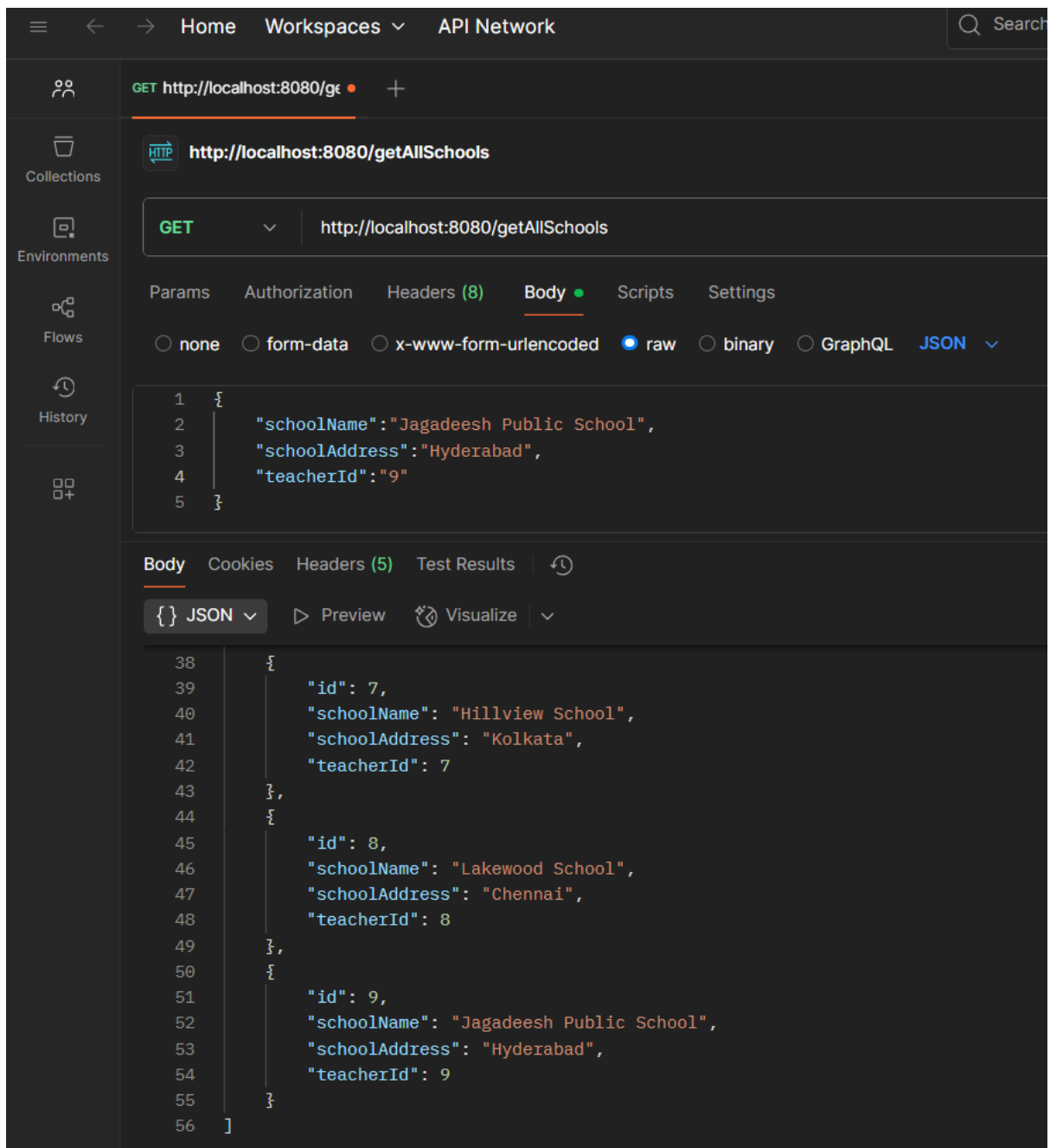
Test Results

{ } JSON

Preview

Visualize

```
20 {
21   "id": 4,
22   "schoolName": "Maple Leaf School",
23   "schoolAddress": "Bangalore",
24   "teacherId": 4
25 },
26 {
27   "id": 5,
28   "schoolName": "Crescent School",
29   "schoolAddress": "Delhi",
30   "teacherId": 5
31 },
32 {
33   "id": 6,
34   "schoolName": "Oakridge International",
35   "schoolAddress": "Pune",
36   "teacherId": 6
37 },
38 {
```



### 3. After (“/update/{id}”)

Home

Workspaces

API Network

Search Postman

Ctrl K

PUT http://localhost:8080/update/9

PUT http://localhost:8080/update/9

ParamsAuthorizationHeaders (8)BodyScriptsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

1 {

2   "schoolName": "Jagadeesh High School",

3   "schoolAddress": "Hyderabad",

4   "teacherId": "9"

5 }

BodyCookiesHeaders (5)Test Results

{ } JSON

PreviewVisualize

1 {

2   "id": 9,

3   "schoolName": "Jagadeesh High School",

4   "schoolAddress": "Hyderabad",

5   "teacherId": 9

6 }

Auto commit

Max rows: 1000

Auto complete Off

Auto select On

jdbc:h2:mem:testdb

SCHOOL

TEACHER

INFORMATION\_SCHEMA

Users

H2 2.3.232 (2024-08-11)

RunRun SelectedAuto completeClearSQL statement:

SELECT \* FROM SCHOOL

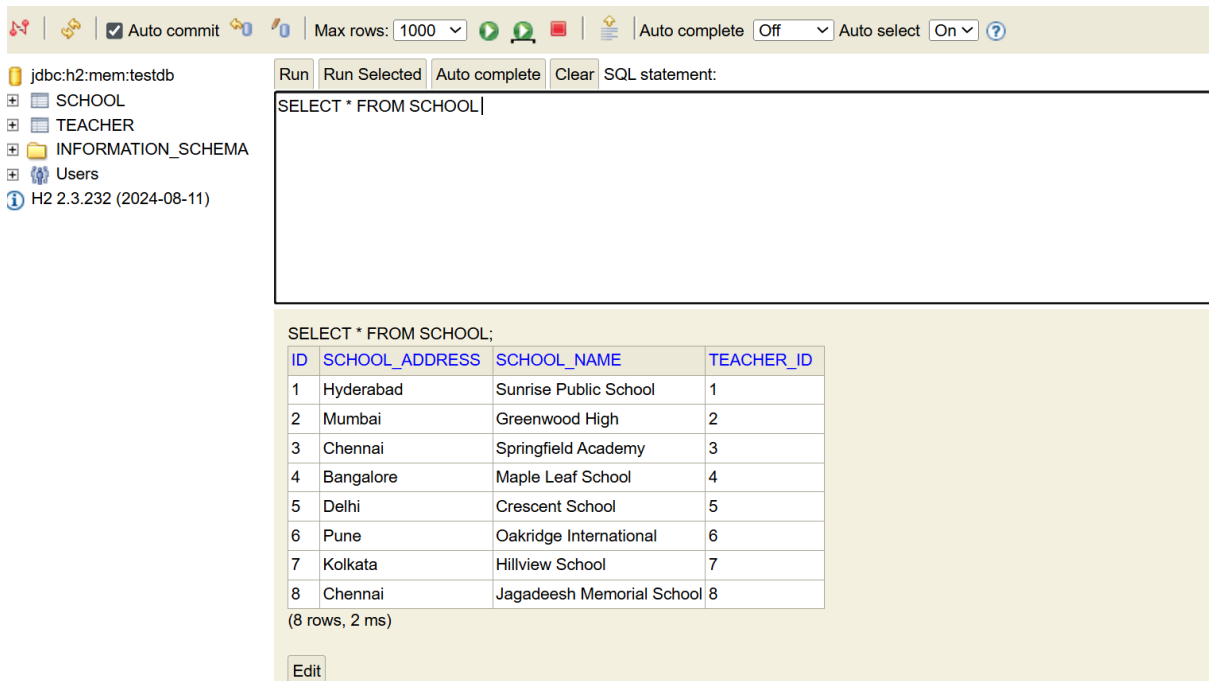
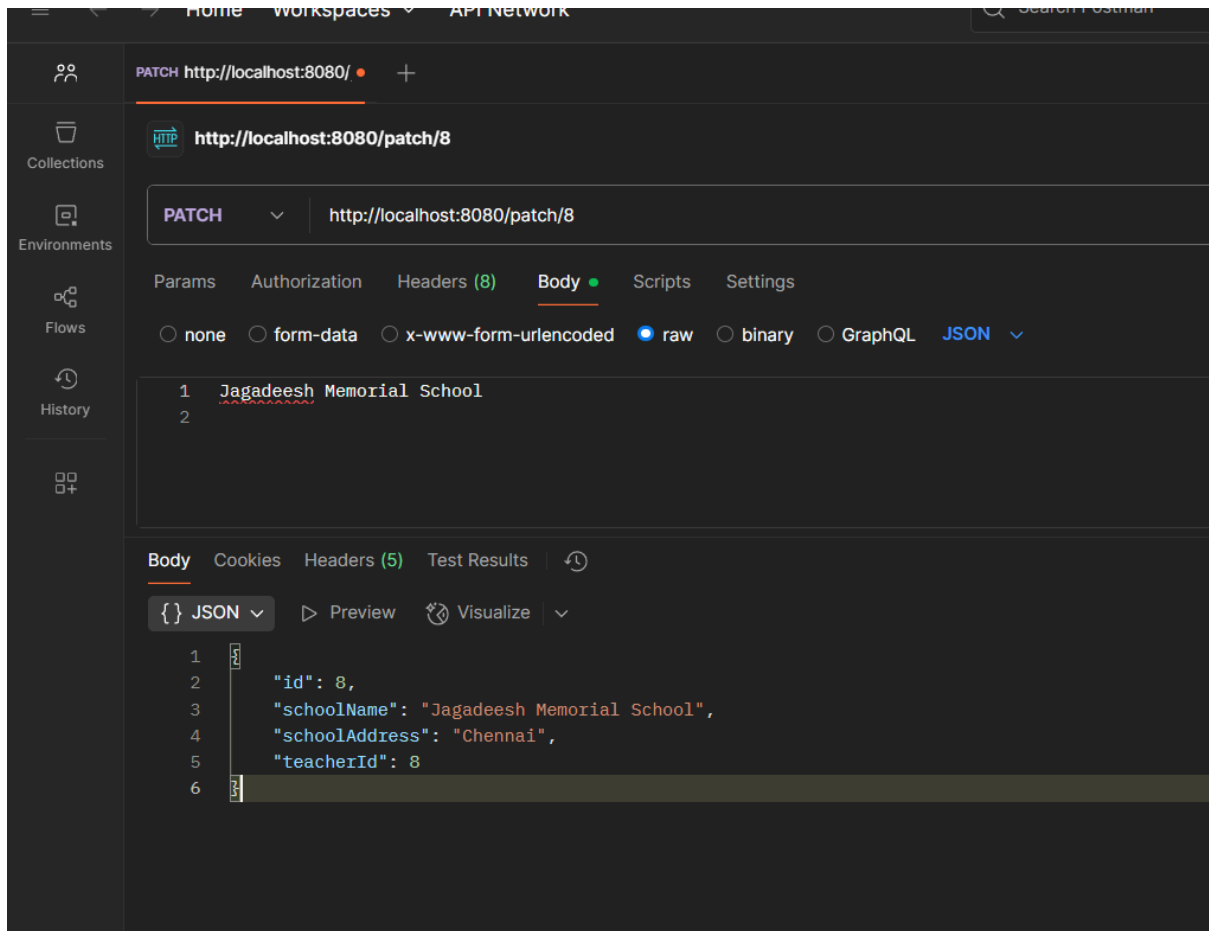
SELECT \* FROM SCHOOL;

ID	SCHOOL_ADDRESS	SCHOOL_NAME	TEACHER_ID
1	Hyderabad	Sunrise Public School	1
2	Mumbai	Greenwood High	2
3	Chennai	Springfield Academy	3
4	Bangalore	Maple Leaf School	4
5	Delhi	Crescent School	5
6	Pune	Oakridge International	6
7	Kolkata	Hillview School	7
8	Chennai	Lakewood School	8
9	Hyderabad	Jagadeesh High School	9

(9 rows, 1 ms)



## 4. After (“/patch/{id}”)



## 5. After (“/deleteSchool/{id}”)

The screenshot displays two applications side-by-side. The top application is Postman, showing a DELETE request to `http://localhost:8080/deleteSchool/8`. The request body is raw text containing `Jagadeesh Memorial School`. The response status is `200 OK`. The bottom application is the H2 database console, showing the SQL statement `SELECT * FROM SCHOOL` and its result set.

**Postman Request Details:**

- Method: DELETE
- URL: `http://localhost:8080/deleteSchool/8`
- Body: `Jagadeesh Memorial School`
- Status: 200 OK

**H2 Database Console:**

Run | Run Selected | Auto complete | Clear | SQL statement:

```
SELECT * FROM SCHOOL
```

SELECT \* FROM SCHOOL;

ID	SCHOOL_ADDRESS	SCHOOL_NAME	TEACHER_ID
1	Hyderabad	Sunrise Public School	1
2	Mumbai	Greenwood High	2
3	Chennai	Springfield Academy	3
4	Bangalore	Maple Leaf School	4
5	Delhi	Crescent School	5
6	Pune	Oakridge International	6
7	Kolkata	Hillview School	7

(7 rows, 0 ms)

Edit

## 6. After (“/school/{schoolAddress}”) - GET

GET http://localhost:8080/sc

HTTP http://localhost:8080/school/Hyderabad

GET http://localhost:8080/school/Hyderabad

ParamsAuthorizationHeaders (8)BodyScriptsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

1Jagadeesh Memorial School

2

BodyCookiesHeaders (5)Test Results

{ } JSON

PreviewVisualize

1[

2{

3" id": 1,

4" schoolName": "Sunrise Public School",

5" schoolAddress": "Hyderabad",

6" teacherId": 1

7}

8]

Collections

Environments

Flows

History

Auto commit

Max rows: 1000

Auto complete Off

Auto select

jdbc:h2:mem:testdb

SCHOOL

TEACHER

INFORMATION\_SCHEMA

Users

H2 2.3.232 (2024-08-11)

RunRun SelectedAuto completeClear

SQL statement:

SELECT \* FROM SCHOOL

SELECT \* FROM SCHOOL;

ID	SCHOOL_ADDRESS	SCHOOL_NAME	TEACHER_ID
1	Hyderabad	Sunrise Public School	1
2	Mumbai	Greenwood High	2
3	Chennai	Springfield Academy	3
4	Bangalore	Maple Leaf School	4
5	Delhi	Crescent School	5
6	Pune	Oakridge International	6
7	Kolkata	Hillview School	7

(7 rows, 0 ms)

Edit

## 8. After(“/addTeacher”)

jdbc:h2:mem:testdb
 

SCHOOL

TEACHER

INFORMATION\_SCHEMA

Users

H2 2.3.232 (2024-08-11)

Max rows: 1000

Auto complete

Clear

SQL statement:

Run

Run Selected

Auto complete

Clear

SQL statement:

```
SELECT * FROM TEACHER
```

2	50	1975-03-10	25	MSC	Mumbai	Jagapati Babu
3	38	1985-08-20	15	BTECH	Chennai	Ravi Teja
4	42	1981-11-30	18	MTECH	Bangalore	Mahesh Babu
5	48	1977-04-05	22	PHD	Delhi	Prabhas
6	35	1986-07-14	12	MSC	Pune	Allu Arjun
7	40	1982-01-25	17	BSC	Kolkata	Ram Charan
8	55	1967-09-09	30	PHD	Chennai	Venkatesh
9	35	1990-05-15	10	MSC		John
10	35	1990-05-15	10	MSC	Hyderabad	John

(10 rows, 2 ms)

Edit

The screenshot displays the Postman REST client interface. At the top, the URL bar shows `http://localhost:8080/addTeacher`. The request method is set to **POST**. The request body is configured as **raw** with **JSON** format. The body content is a JSON object representing a teacher's details.

**Request Body:**

```
{
  "teacherName": "John",
  "teacherAddress": "Hyderabad",
  "qualification": "MSC",
  "age": 35,
  "dob": "1990-05-15",
  "experience": 10
}
```

The bottom section shows the **Body** tab of the response, which contains the same JSON object returned by the server.

**Response Body:**

```
{
  "id": 10,
  "teacherName": "John",
  "teacherAddress": "Hyderabad",
  "qualification": "MSC",
  "age": 35,
  "dob": "1990-05-15",
  "experience": 10
}
```

## 9. After (“getAllTeachers”)

Body Cookies Headers (5) Test Results

{ } JSON Preview Visualize

```
1  [
2    {
3      "id": 1,
4      "teacherName": "Naga Chaitanya",
5      "teacherAddress": "Hyderabad",
6      "qualification": "PHD",
7      "age": 45,
8      "dob": "1980-05-15",
9      "experience": 20
10   },
11   {
12     "id": 2,
13     "teacherName": "Jagapati Babu",
14     "teacherAddress": "Mumbai",
15     "qualification": "MSC",
16     "age": 50,
17     "dob": "1975-03-10",
18     "experience": 25
19   },
20   {
```

{ } JSON Preview Visualize

```
13     "teacherName": "Jagapati Babu",
14     "teacherAddress": "Mumbai",
15     "qualification": "MSC",
16     "age": 50,
17     "dob": "1975-03-10",
18     "experience": 25
19   },
20   {
21     "id": 3,
22     "teacherName": "Ravi Teja",
23     "teacherAddress": "Chennai",
24     "qualification": "BTECH",
25     "age": 38,
26     "dob": "1985-08-20",
27     "experience": 15
28   },
29   {
30     "id": 4,
31     "teacherName": "Mahesh Babu",
32     "teacherAddress": "Bangalore",
33     "qualification": "MTECH",
```

{ } JSON ▾ ▶ Preview ↺ Visualize ▾

```
35     "dob": "1981-11-30",
36     "experience": 18
37 },
38 {
39     "id": 5,
40     "teacherName": "Prabhas",
41     "teacherAddress": "Delhi",
42     "qualification": "PHD",
43     "age": 48,
44     "dob": "1977-04-05",
45     "experience": 22
46 },
47 {
48     "id": 6,
49     "teacherName": "Allu Arjun",
50     "teacherAddress": "Pune",
51     "qualification": "MSC",
52     "age": 35,
53     "dob": "1986-07-14",
54     "experience": 12
55 }
```

{ } JSON ▾

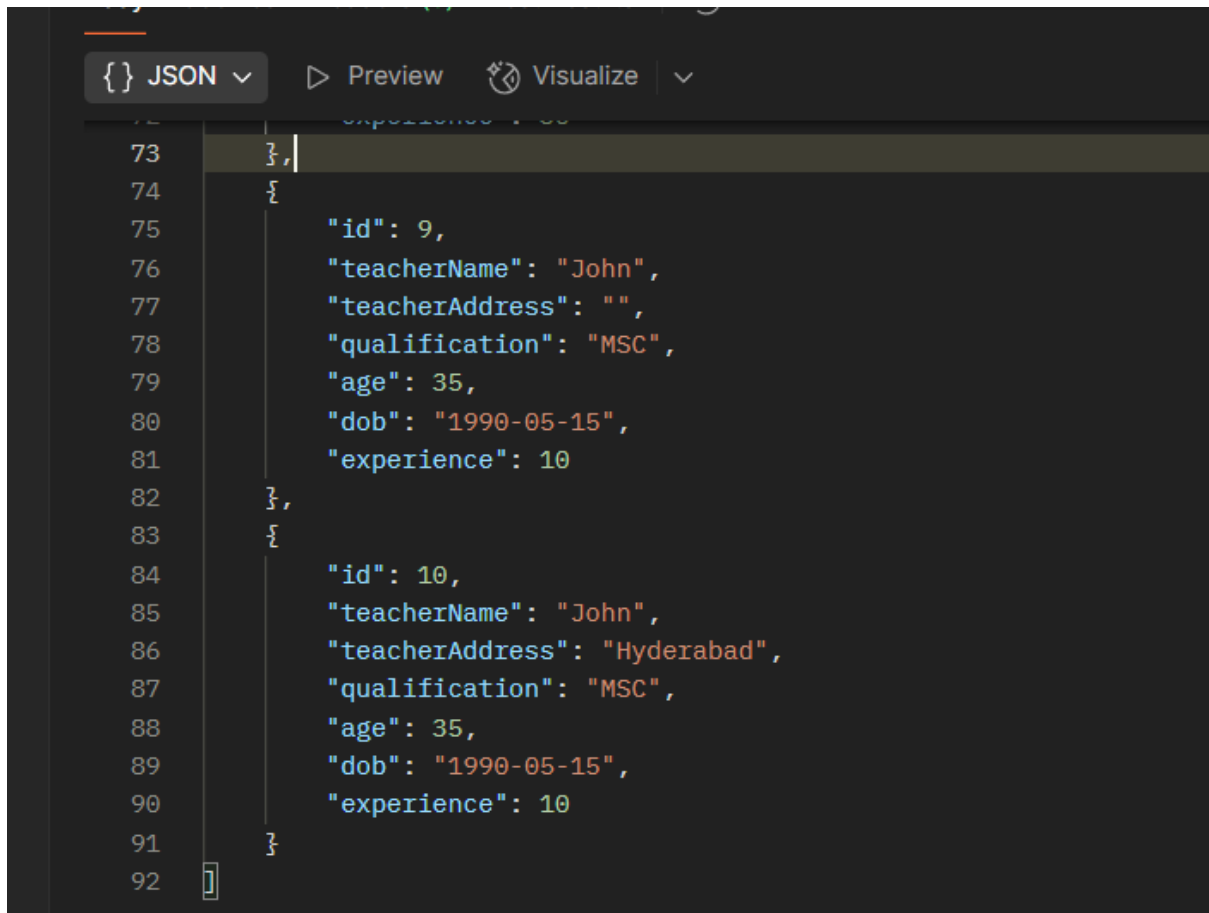
▶ Preview

🔄 Visualize ▾

```
57     "id": 7,  
58     "teacherName": "Ram Charan",  
59     "teacherAddress": "Kolkata",  
60     "qualification": "BSC",  
61     "age": 40,  
62     "dob": "1982-01-25",  
63     "experience": 17  
64   },  
65   {  
66     "id": 8,  
67     "teacherName": "Venkatesh",  
68     "teacherAddress": "Chennai",  
69     "qualification": "PHD",  
70     "age": 55,  
71     "dob": "1967-09-09",  
72     "experience": 30  
73   },  
74   {  
75     "id": 9,  
76     "teacherName": "John",  
77     "teacherAddress": "Kolkata",  
78     "qualification": "BSC",  
79     "age": 35,  
80     "dob": "1987-03-10",  
81     "experience": 10  
82   }  
83 ]
```

🔍 Find and replace

📄 Console



The screenshot shows a JSON editor interface with a dark theme. At the top, there is a toolbar with a dropdown menu set to 'JSON', a 'Preview' button, and a 'Visualize' button. The main area displays a JSON array containing two objects. The first object has an 'id' of 9, 'teacherName' of 'John', 'teacherAddress' of an empty string, 'qualification' of 'MSC', 'age' of 35, 'dob' of '1990-05-15', and 'experience' of 10. The second object has an 'id' of 10, 'teacherName' of 'John', 'teacherAddress' of 'Hyderabad', 'qualification' of 'MSC', 'age' of 35, 'dob' of '1990-05-15', and 'experience' of 10. Line numbers 73 through 92 are visible on the left side of the editor.

```
73  },  
74  {  
75    "id": 9,  
76    "teacherName": "John",  
77    "teacherAddress": "",  
78    "qualification": "MSC",  
79    "age": 35,  
80    "dob": "1990-05-15",  
81    "experience": 10  
82  },  
83  {  
84    "id": 10,  
85    "teacherName": "John",  
86    "teacherAddress": "Hyderabad",  
87    "qualification": "MSC",  
88    "age": 35,  
89    "dob": "1990-05-15",  
90    "experience": 10  
91  }  
92  ]
```

## 10. After (“"/updateTeacher/{id}”)



ents

POST http://localhost:8080/updateTeacher/9

ParamsAuthorizationHeaders (8)BodyScriptsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

```
1 {
2   "teacherName": "John sena",
3   "teacherAddress": "Hyderabad",
4   "qualification": "MSC",
5   "age": 45,
6   "dob": "1990-05-15",
7   "experience": 20
8 }
9
```

BodyCookiesHeaders (5)Test Results

{ } JSON

PreviewVisualize

```
1 {
2   "id": 9,
3   "teacherName": "John sena",
4   "teacherAddress": "Hyderabad",
5   "qualification": "MSC",
6   "age": 45,
7   "dob": "1990-05-15",
8   "experience": 20
9 }
```

OnlineFind and replaceConsole

jdbc:h2:mem:testdb

SCHOOLTEACHERINFORMATION\_SCHEMAUsersH2 2.3.232 (2024-08-11)

RunRun SelectedAuto completeClearSQL statement:

SELECT \* FROM TEACHER

2	50	1975-03-10	25	MSC	Mumbai	Jagapati Babu
3	38	1985-08-20	15	BTECH	Chennai	Ravi Teja
4	42	1981-11-30	18	MTECH	Bangalore	Mahesh Babu
5	48	1977-04-05	22	PHD	Delhi	Prabhas
6	35	1986-07-14	12	MSC	Pune	Allu Arjun
7	40	1982-01-25	17	BSC	Kolkata	Ram Charan
8	55	1967-09-09	30	PHD	Chennai	Venkatesh
9	45	1990-05-15	20	MSC	Hyderabad	John sena
10	35	1990-05-15	10	MSC	Hyderabad	John

(10 rows, 0 ms)

Edit

# 11. After (“/patchTeacher/{id}”)

Collections

Environments

Flows

History

HTTP

http://localhost:8080/patchTeacher/9

PATCH

http://localhost:8080/patchTeacher/9

Params

Authorization

Headers (8)

Body

Scripts

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

John Don

Body

Cookies

Headers (5)

Test Results

JSON

Preview

Visualize

1

{

2

"id": 9,

3

"teacherName": "John Don",

4

"teacherAddress": "Hyderabad",

5

"qualification": "MSC",

6

"age": 45,

7

"dob": "1990-05-15",

8

"experience": 20

9

}

jdbc:h2:mem:testdb

SCHOOL

TEACHER

INFORMATION\_SCHEMA

Users

H2 2.3.232 (2024-08-11)

Run

Run Selected

Auto complete

Clear

SQL statement:

SELECT \* FROM TEACHER

2	50	1975-03-10	25	MSC	Mumbai	Jagapati Babu
3	38	1985-08-20	15	BTECH	Chennai	Ravi Teja
4	42	1981-11-30	18	MTECH	Bangalore	Mahesh Babu
5	48	1977-04-05	22	PHD	Delhi	Prabhas
6	35	1986-07-14	12	MSC	Pune	Allu Arjun
7	40	1982-01-25	17	BSC	Kolkata	Ram Charan
8	55	1967-09-09	30	PHD	Chennai	Venkatesh
9	45	1990-05-15	20	MSC	Hyderabad	John Don
10	35	1990-05-15	10	MSC	Hyderabad	John

(10 rows, 1 ms)

# 12. After (“/deleteTeacher/{id}”)

collections

environments

flows

history

DELETE

http://localhost:8080/deleteTeacher/9

Params

Authorization

Headers (8)

Body

Scripts

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

John

Don

Body

Cookies

Headers (4)

Test Results

Raw

Preview

Visualize

200 OK

1

Auto commit

Max rows: 1000

Auto complete

Auto select

Run

Run Selected

Auto complete

Clear

SQL statement:

SELECT \* FROM TEACHER|

SELECT \* FROM TEACHER;

	ID	AGE	DOB	EXPERIENCE	QUALIFICATION	TEACHER_ADDRESS	TEACHER_NAME
1	45	1980-05-15	20		PHD	Hyderabad	Naga Chaitanya
2	50	1975-03-10	25		MSC	Mumbai	Jagapati Babu
3	38	1985-08-20	15		BTECH	Chennai	Ravi Teja
4	42	1981-11-30	18		MTECH	Bangalore	Mahesh Babu
5	48	1977-04-05	22		PHD	Delhi	Prabhas
6	35	1986-07-14	12		MSC	Pune	Allu Arjun
7	40	1982-01-25	17		BSC	Kolkata	Ram Charan
8	55	1967-09-09	30		PHD	Chennai	Venkatesh
10	35	1990-05-15	10		MSC	Hyderabad	John

(9 rows, 0 ms)

jdbc:h2:mem:testdb

SCHOOL

TEACHER

INFORMATION\_SCHEMA

Users

H2 2.3.232 (2024-08-11)

## 13. After (“/teachers/{qualification}”)

GET http://localhost:8080/te

+

Collections

Environments

Flows

History

GET http://localhost:8080/teachers/MS

GET http://localhost:8080/teachers/MS

Params Authorization Headers (8) Body • Scripts Settings

Body Cookies Headers (5) Test Results ↺

{ } JSON ▾ ▶ Preview ↻ Visualize ▾

```
1  [
2    {
3      "id": 2,
4      "teacherName": "Jagapati Babu",
5      "teacherAddress": "Mumbai",
6      "qualification": "MSC",
7      "age": 50,
8      "dob": "1975-03-10",
9      "experience": 25
10   },
11   {
12     "id": 6,
13     "teacherName": "Allu Arjun",
14     "teacherAddress": "Pune",
15     "qualification": "MSC",
16     "age": 35,
17     "dob": "1986-07-14",
18     "experience": 12
19   },
20   {
21     "id": 10,
22     "teacherName": "John",
23     "teacherAddress": "Hyderabad",
24     "qualification": "MSC",
25     "age": 35,
26     "dob": "1990-05-15",
27     "experience": 10
28   }
29 ]
```