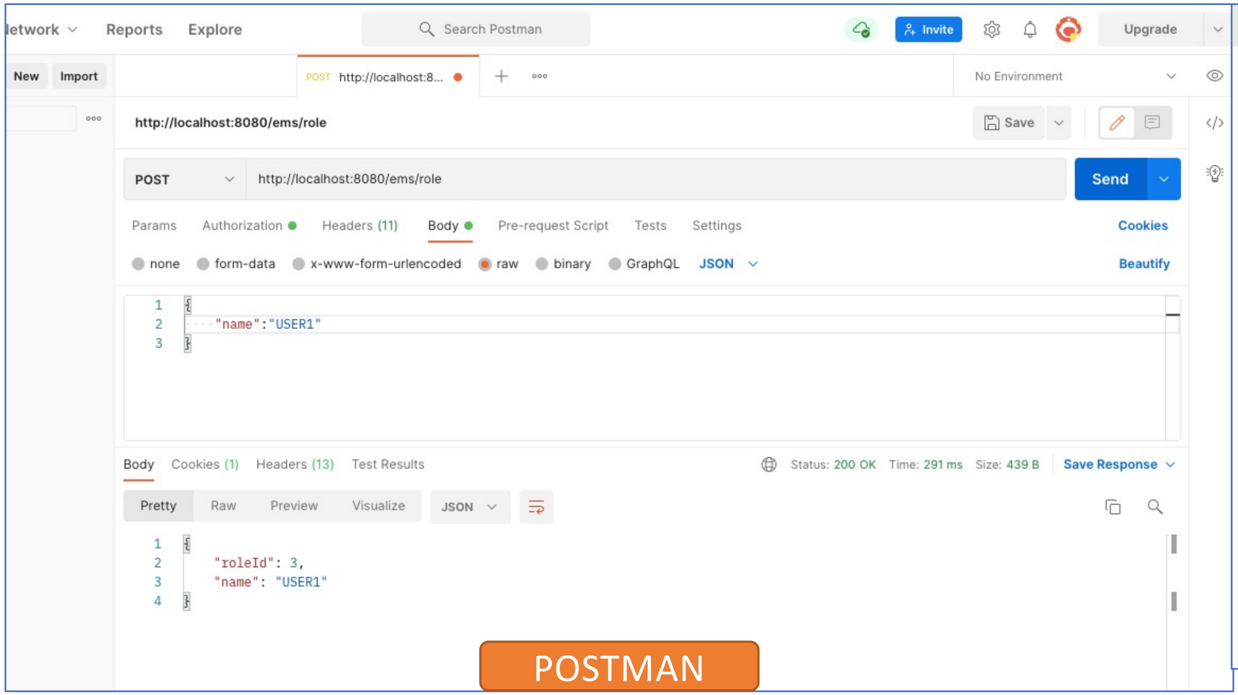
Graded Coding Assigment-6 **Riajuddin_RestAPIAssignmentSolution**_Screenshots

1. Adding A new Role

URL: http://localhost:8080/ems/role



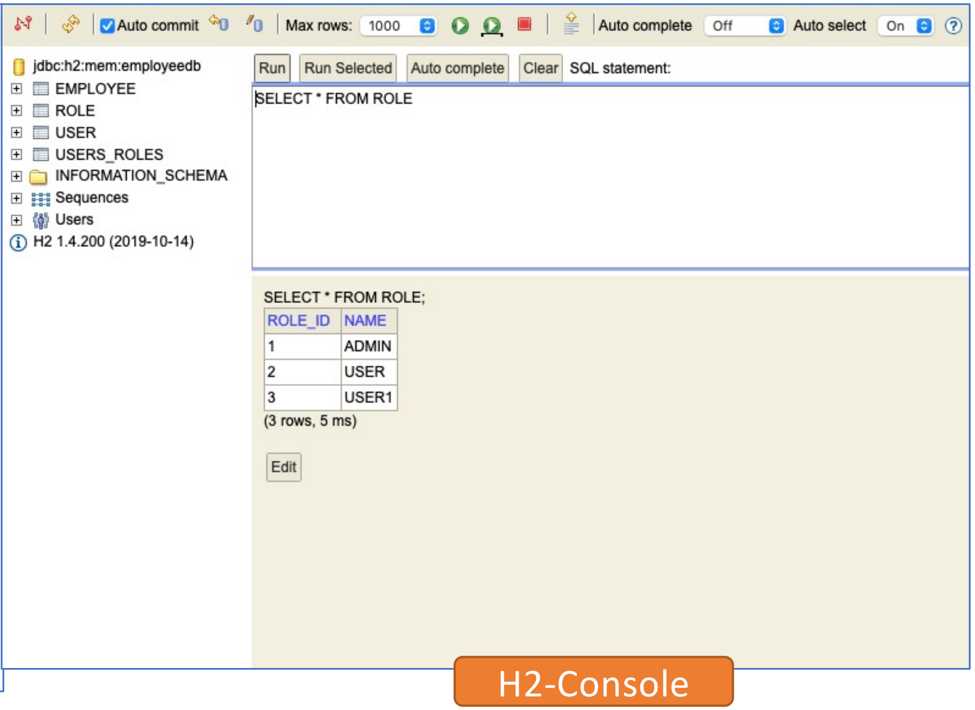
POST http://localhost:8080/ems/role

```
{
    "name":"USER1"
}
```

Status: 200 OK Time: 291 ms Size: 439 B

```
{
    "roleId": 3,
    "name": "USER1"
}
```

POSTMAN



jdbc:h2:mem:employeedb
- EMPLOYEE
- ROLE
- USER
- USERS_ROLES
- INFORMATION_SCHEMA
- Sequences
- Users

H2 1.4.200 (2019-10-14)

SQL statement:

```
SELECT * FROM ROLE
```

SELECT * FROM ROLE;

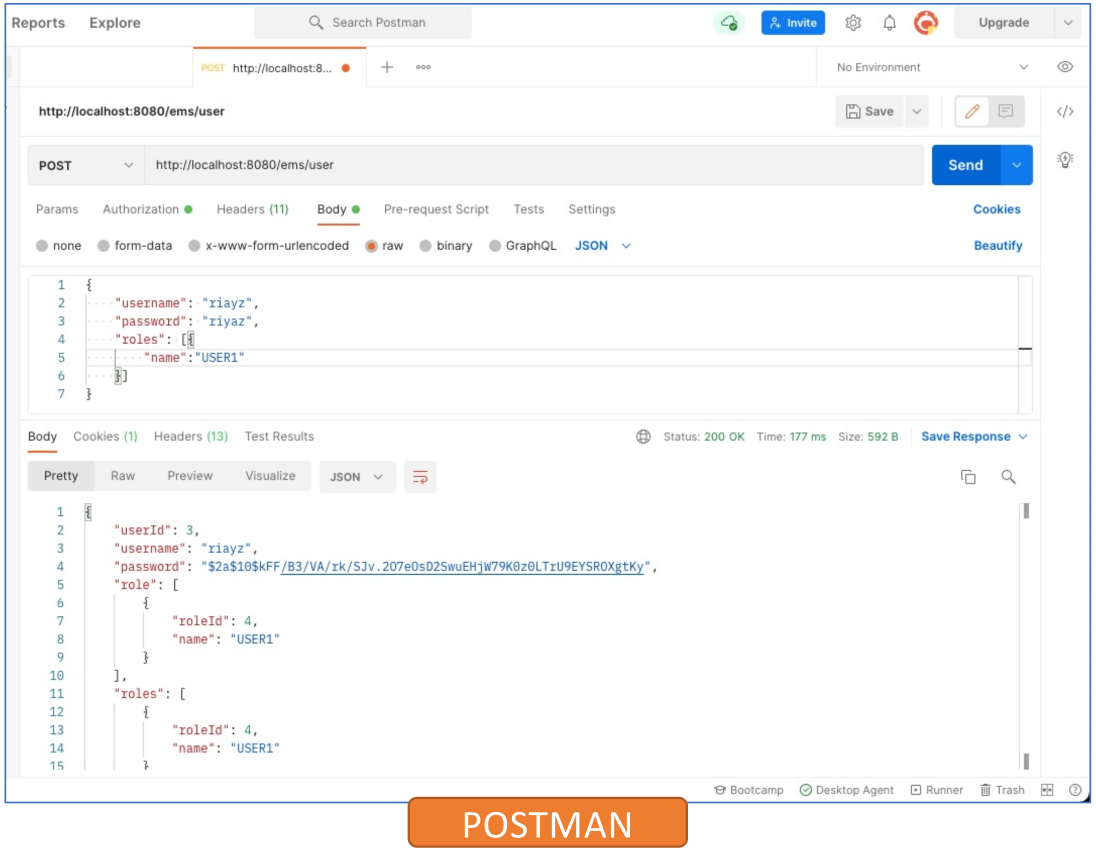
ROLE_ID	NAME
1	ADMIN
2	USER
3	USER1

(3 rows, 5 ms)

H2-Console

2. Adding User

URL: http://localhost:8080/ems/user



http://localhost:8080/ems/user

POST http://localhost:8080/ems/user

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

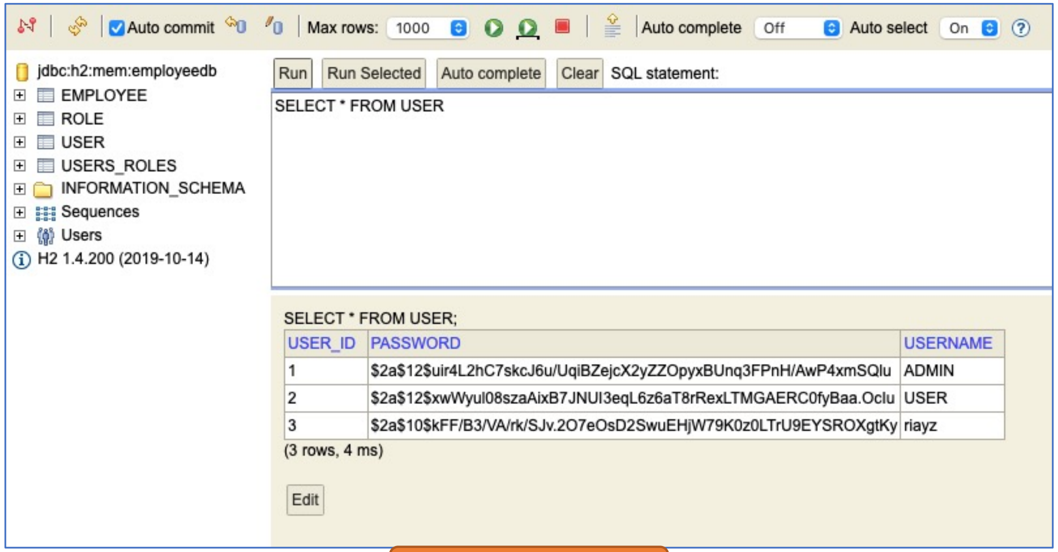
```
{
    "username": "riayz",
    "password": "riyaz",
    "roles": [{
        "name":"USER1"
    }]
}
```

Body Cookies (1) Headers (13) Test Results — Status: 200 OK Time: 177 ms Size: 592 B Save Response

JSON

```
{
    "userId": 3,
    "username": "riayz",
    "password": "$2a$10$kFF/B3/VA/rk/SJv.2O7eOsD2SwuEHjW79K0z0LTrU9EYSROXgtKy",
    "role": [
        {
            "roleId": 4,
            "name": "USER1"
        }
    ],
    "roles": [
        {
            "roleId": 4,
            "name": "USER1"
        }
```

POSTMAN



jdbc:h2:mem:employeedb
- EMPLOYEE
- ROLE
- USER
- USERS_ROLES
- INFORMATION_SCHEMA
- Sequences
- Users
- H2 1.4.200 (2019-10-14)

```
SELECT * FROM USER
```

SELECT * FROM USER;

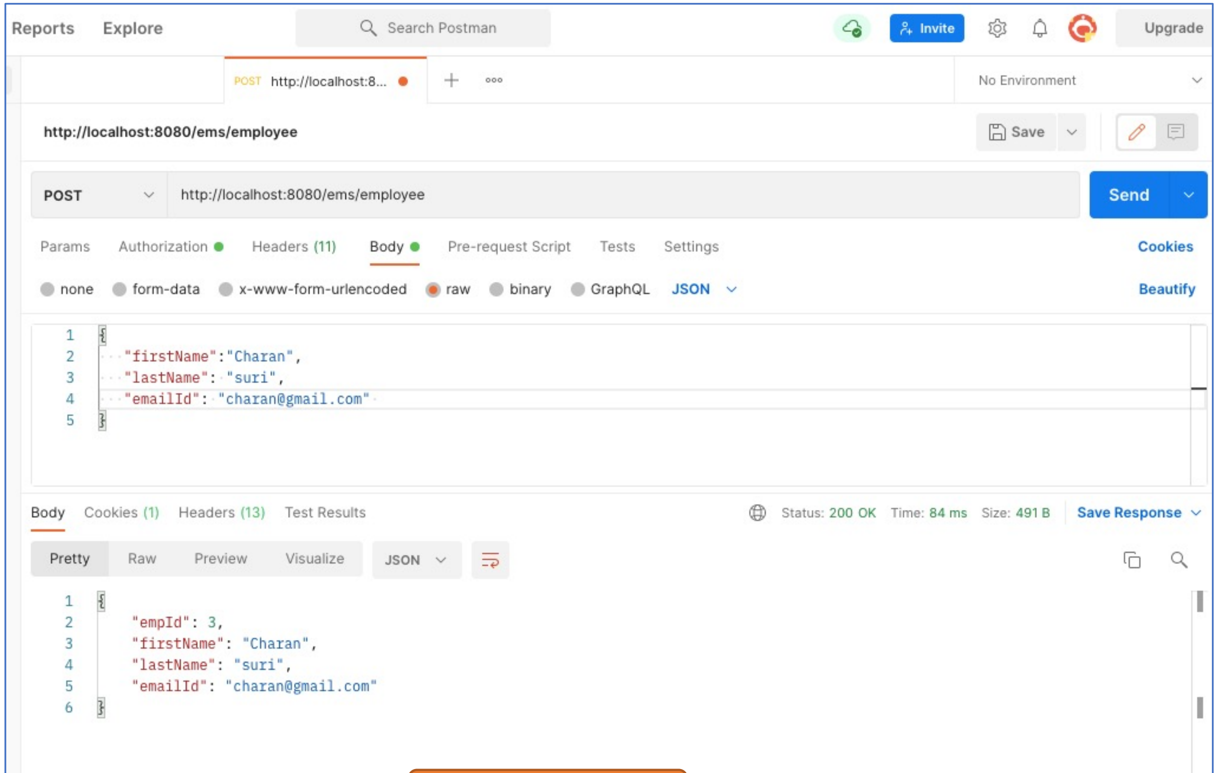
USER_ID	PASSWORD	USERNAME
1	$2a$12$uir4L2hC7skcJ6u/UqiBZejcX2yZZOpyxBUnq3FPnH/AwP4xmSQlu	ADMIN
2	$2a$12$xwWyul08szaAixB7JNUI3eqL6z6aT8rRexLTMGAERC0fyBaa.Oclu	USER
3	$2a$10$kFF/B3/VA/rk/SJv.2O7eOsD2SwuEHjW79K0z0LTrU9EYSROXgtKy	riayz

(3 rows, 4 ms)

H2-Console

3. Adding New Employee

URL: http://localhost:8080/ems/employee



http://localhost:8080/ems/employee

POST http://localhost:8080/ems/employee

Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
  "firstName":"Charan",
  "lastName": "suri",
  "emailId": "charan@gmail.com"
}
```

Body Cookies (1) Headers (13) Test Results

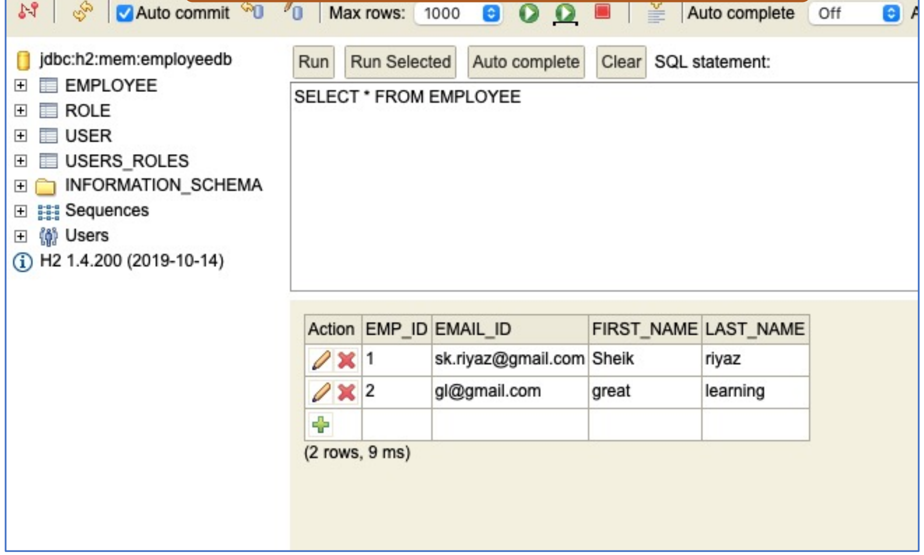
Status: 200 OK Time: 84 ms Size: 491 B

JSON

```
{
    "empId": 3,
    "firstName": "Charan",
    "lastName": "suri",
    "emailId": "charan@gmail.com"
}
```

POSTMAN

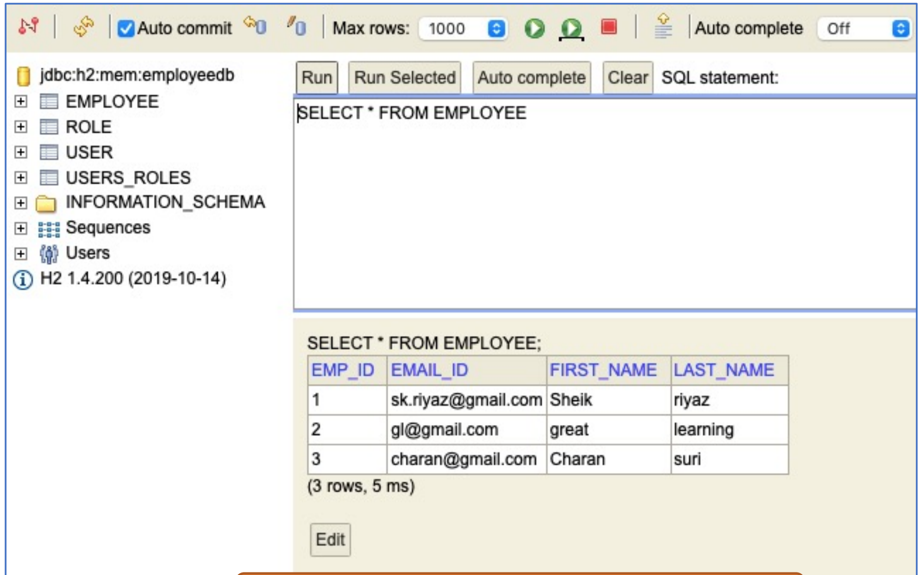
H2-Console before adding charan



SELECT * FROM EMPLOYEE

Action	EMP_ID	EMAIL_ID	FIRST_NAME	LAST_NAME
✎ ✖	1	sk.riyaz@gmail.com	Sheik	riyaz
✎ ✖	2	gl@gmail.com	great	learning
✚				

(2 rows, 9 ms)



SELECT * FROM EMPLOYEE

SELECT * FROM EMPLOYEE;

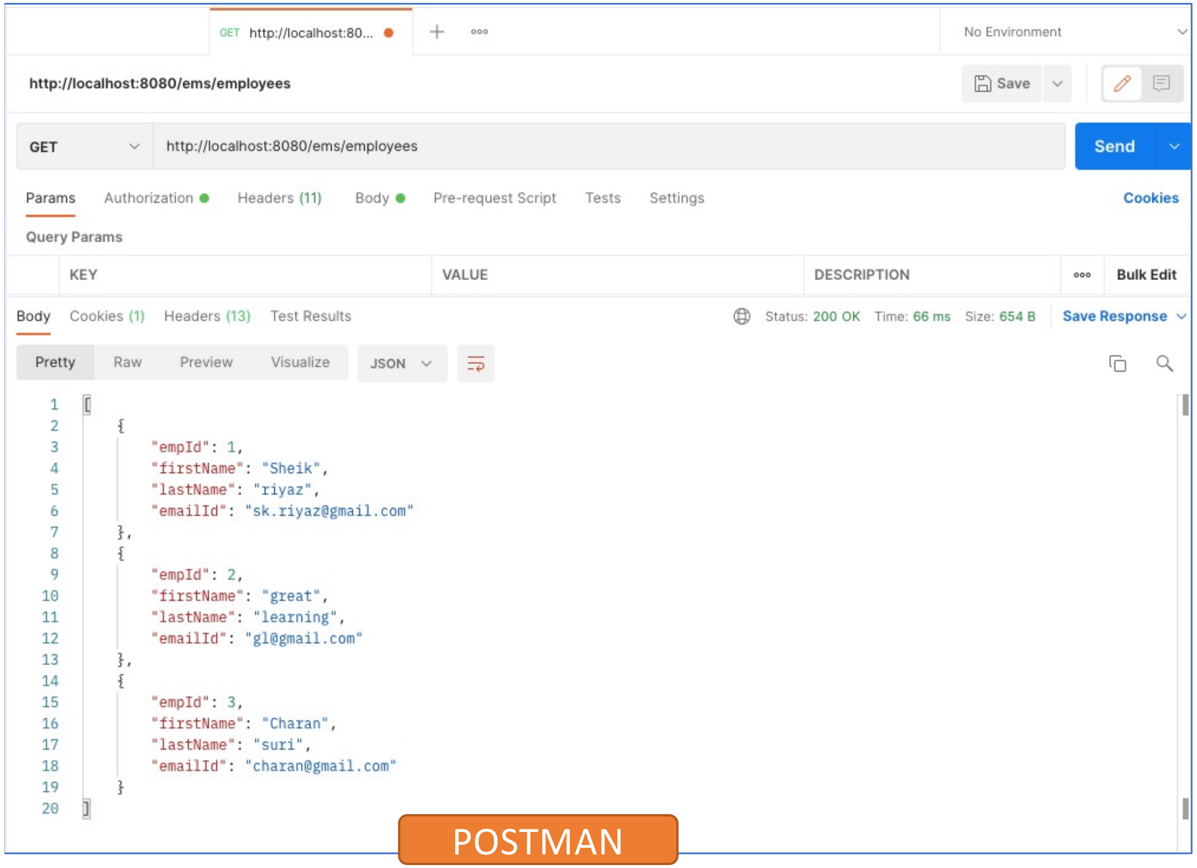
EMP_ID	EMAIL_ID	FIRST_NAME	LAST_NAME
1	sk.riyaz@gmail.com	Sheik	riyaz
2	gl@gmail.com	great	learning
3	charan@gmail.com	Charan	suri

(3 rows, 5 ms)

H2-Console after adding charan

4. Adding A New Employee

URL: http://localhost:8080/ems/employees

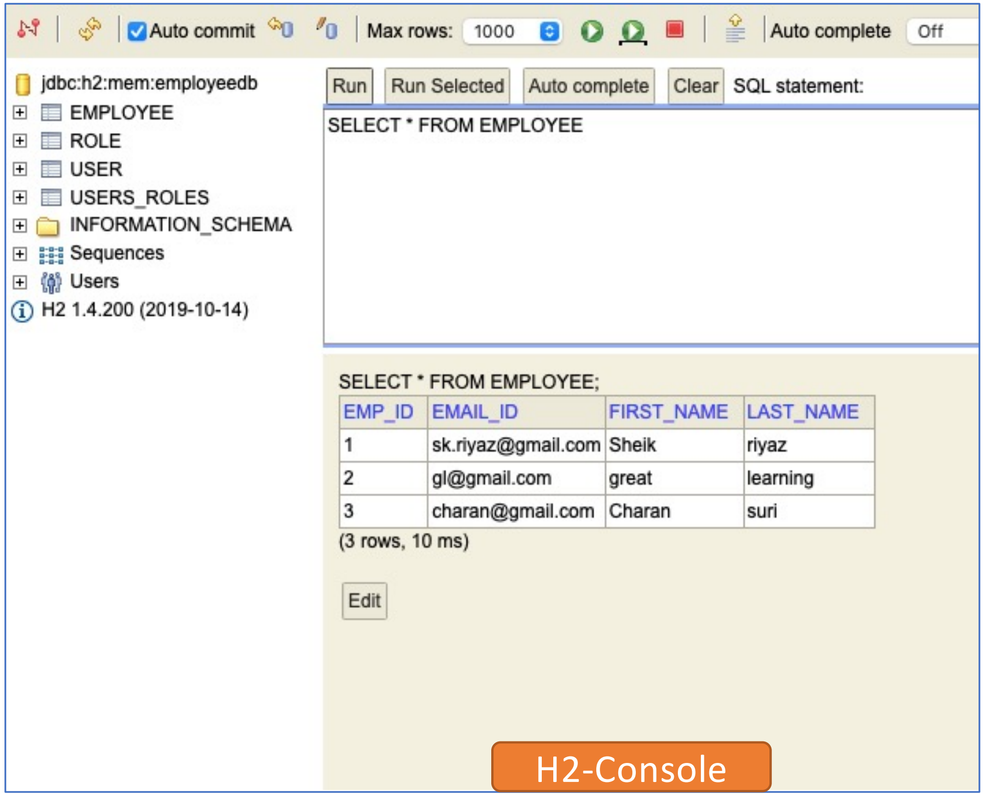


GET http://localhost:8080/ems/employees

Status: 200 OK Time: 66 ms Size: 654 B

```
[
    {
        "empId": 1,
        "firstName": "Sheik",
        "lastName": "riyaz",
        "emailId": "sk.riyaz@gmail.com"
    },
    {
        "empId": 2,
        "firstName": "great",
        "lastName": "learning",
        "emailId": "gl@gmail.com"
    },
    {
        "empId": 3,
        "firstName": "Charan",
        "lastName": "suri",
        "emailId": "charan@gmail.com"
    }
]
```

POSTMAN



SELECT * FROM EMPLOYEE

SELECT * FROM EMPLOYEE;

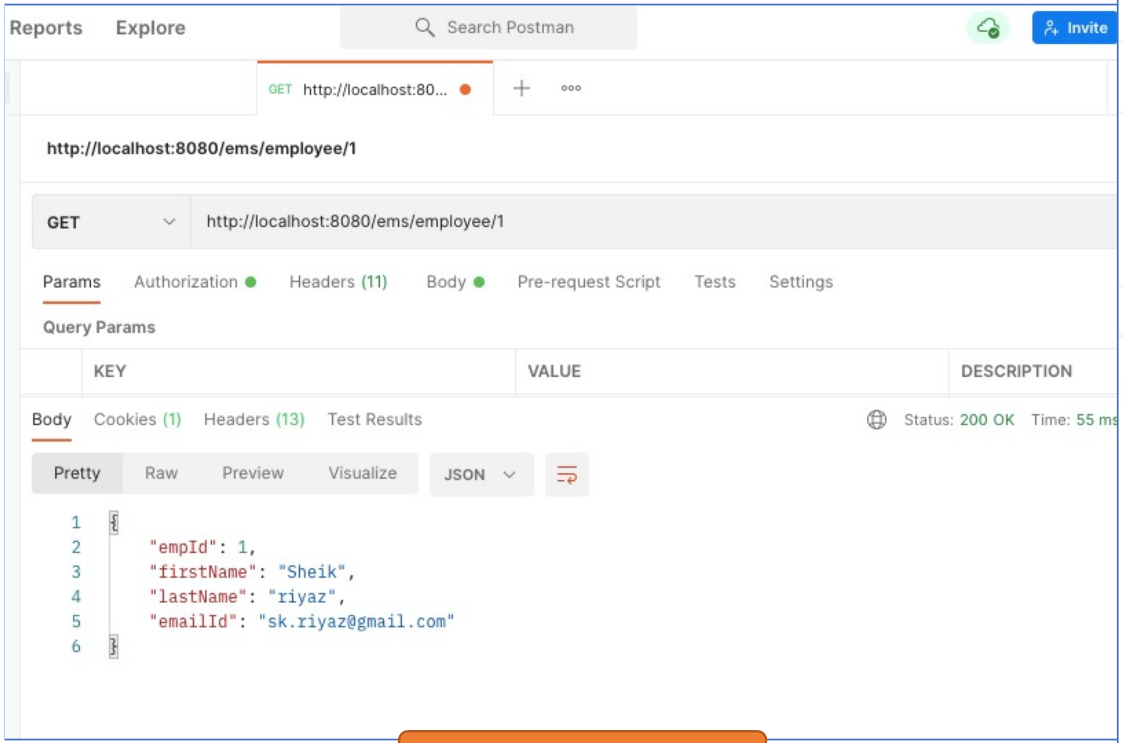
EMP_ID	EMAIL_ID	FIRST_NAME	LAST_NAME
1	sk.riyaz@gmail.com	Sheik	riyaz
2	gl@gmail.com	great	learning
3	charan@gmail.com	Charan	suri

(3 rows, 10 ms)

H2-Console

5. Finding a employee with ID (1&3)

URL: http://localhost:8080/ems/employee/1 & http://localhost:8080/ems/employee/3



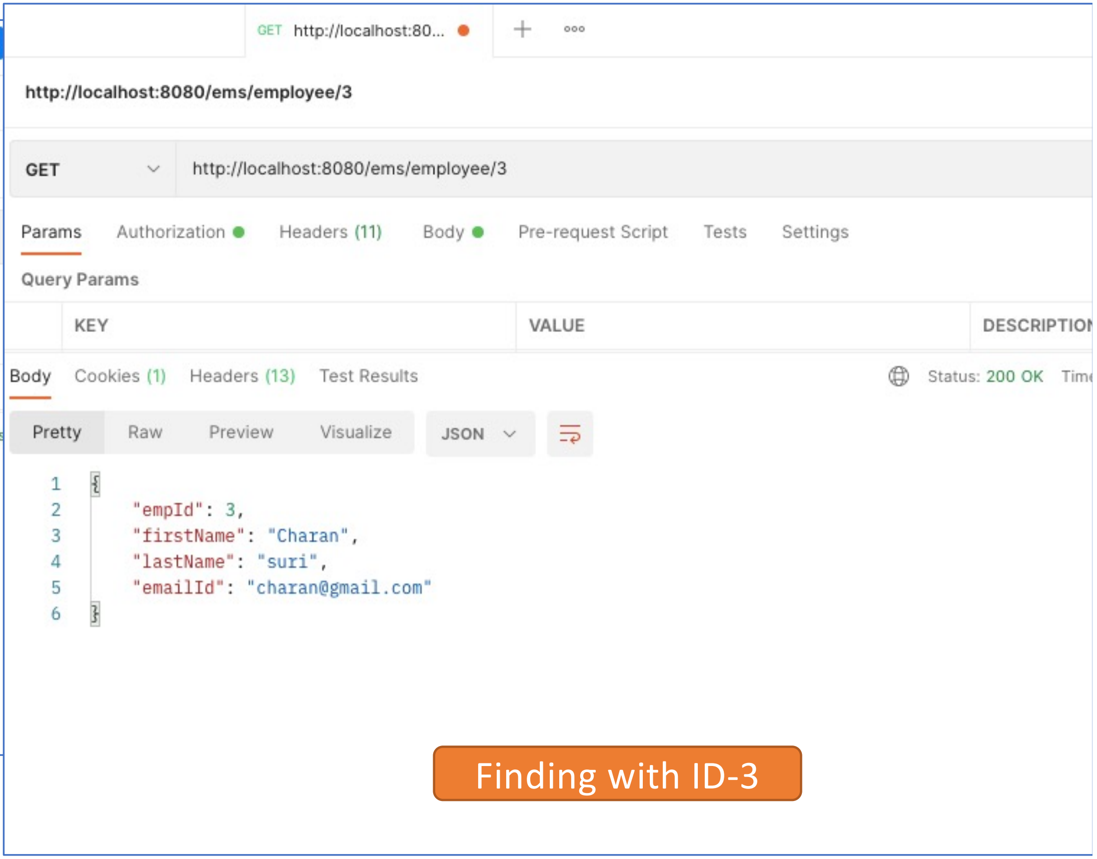
http://localhost:8080/ems/employee/1

GET http://localhost:8080/ems/employee/1

Status: 200 OK Time: 55 ms

```
{
    "empId": 1,
    "firstName": "Sheik",
    "lastName": "riyaz",
    "emailId": "sk.riyaz@gmail.com"
}
```

Finding with ID-1



http://localhost:8080/ems/employee/3

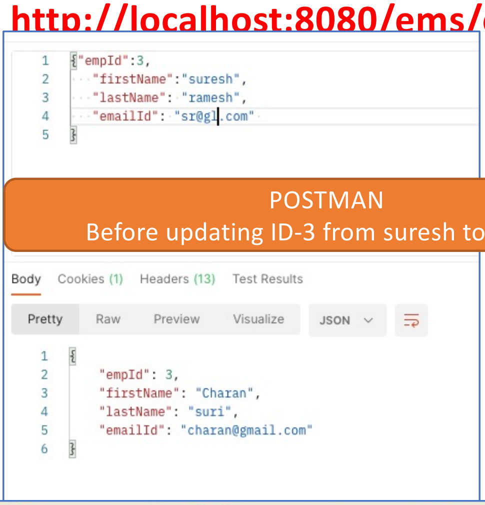
GET http://localhost:8080/ems/employee/3

Status: 200 OK Time

```
{
    "empId": 3,
    "firstName": "Charan",
    "lastName": "suri",
    "emailId": "charan@gmail.com"
}
```

Finding with ID-3

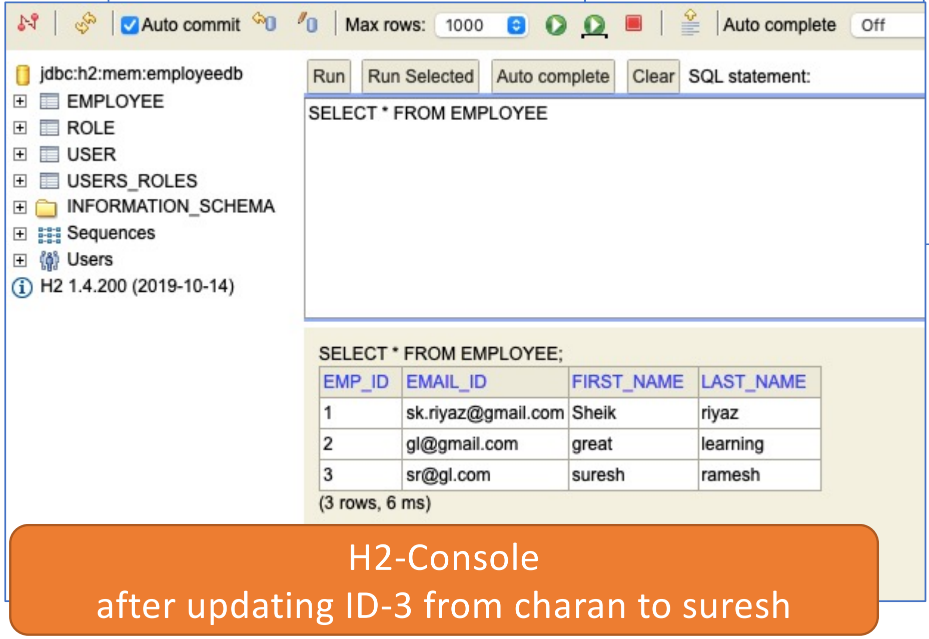
6. Updating Employee details-charan(id-3)
URL: http://localhost:8080/ems/employee



```
{"empId":3,
   "firstName":"suresh",
   "lastName": "ramesh",
   "emailId": "sr@gl.com"
}
```

POSTMAN
Before updating ID-3 from suresh to charan

```
{
    "empId": 3,
    "firstName": "Charan",
    "lastName": "suri",
    "emailId": "charan@gmail.com"
}
```



☑ Auto commit

Max rows: 1000

Auto complete Off

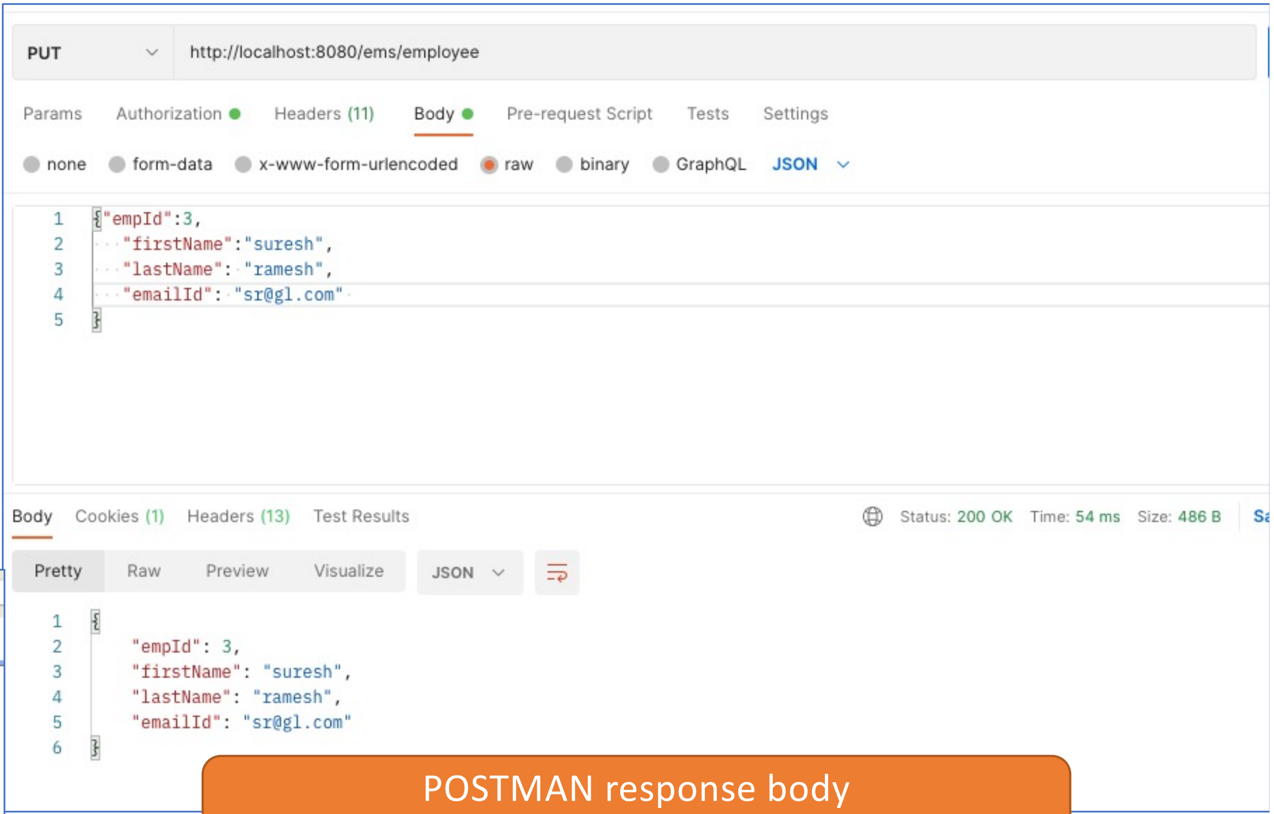
```
SELECT * FROM EMPLOYEE
```

SELECT * FROM EMPLOYEE;

EMP_ID	EMAIL_ID	FIRST_NAME	LAST_NAME
1	sk.riyaz@gmail.com	Sheik	riyaz
2	gl@gmail.com	great	learning
3	sr@gl.com	suresh	ramesh

(3 rows, 6 ms)

H2-Console
after updating ID-3 from charan to suresh



PUT http://localhost:8080/ems/employee

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{"empId":3,
   "firstName":"suresh",
   "lastName": "ramesh",
   "emailId": "sr@gl.com"
}
```

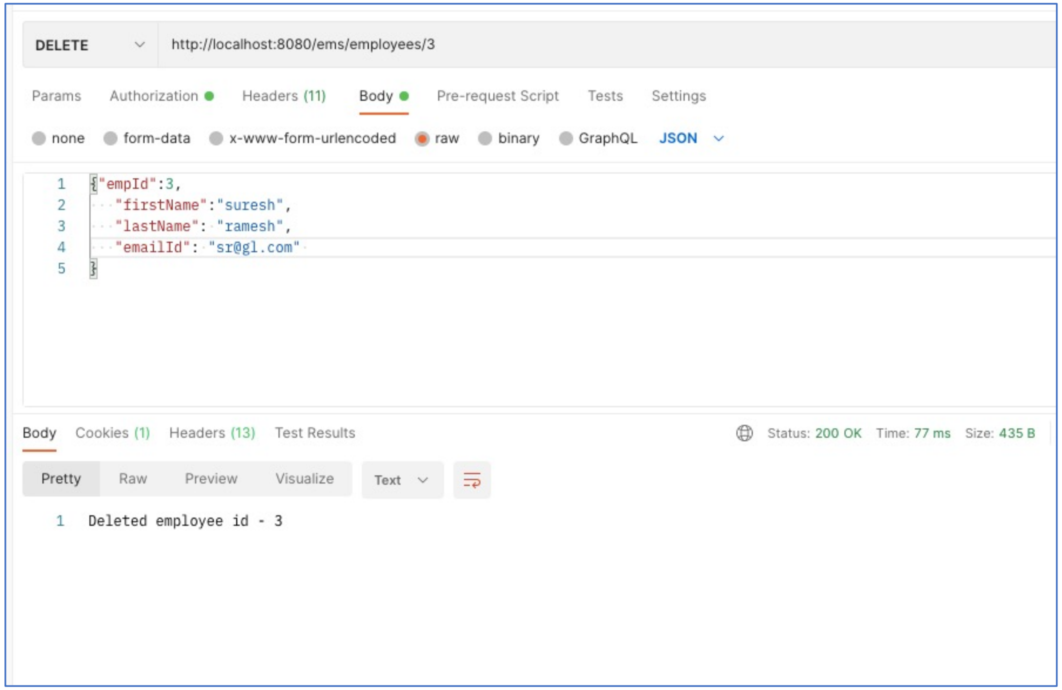
Status: 200 OK Time: 54 ms Size: 486 B

```
{
    "empId": 3,
    "firstName": "suresh",
    "lastName": "ramesh",
    "emailId": "sr@gl.com"
}
```

POSTMAN response body
after updating ID-3 from charan to suresh

7. Deleting Employee -suresh(id-3)

URL: http://localhost:8080/ems/employees/3



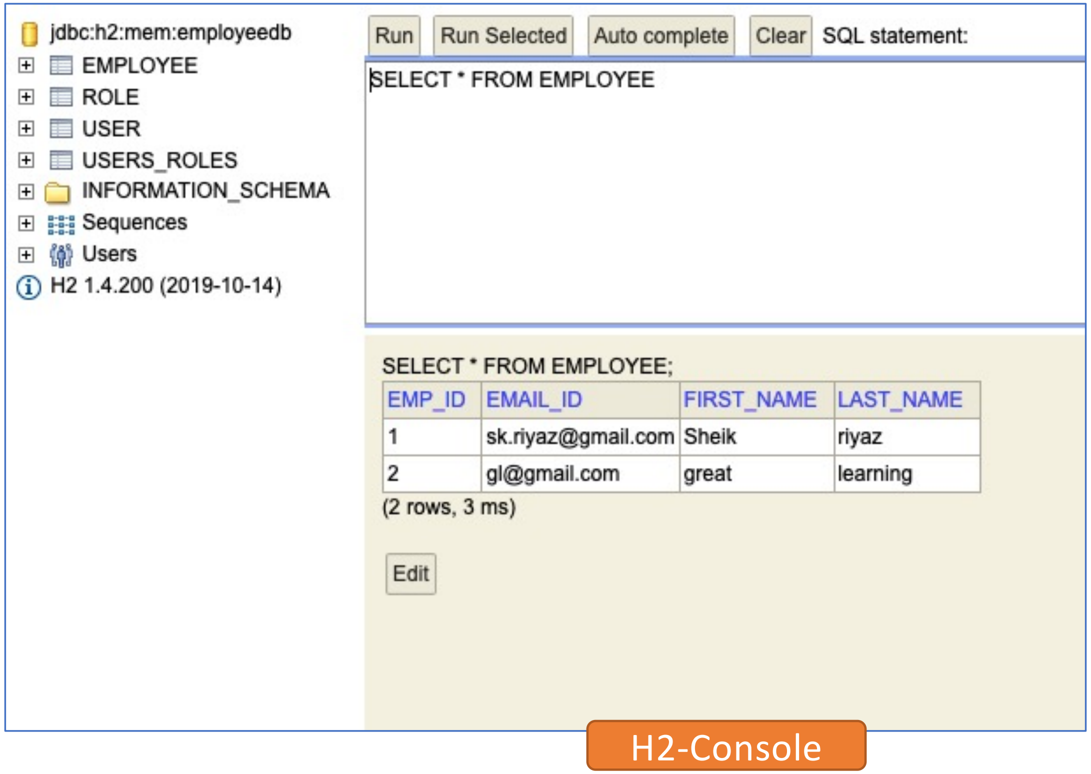
DELETE http://localhost:8080/ems/employees/3

```
{"empId":3,
  "firstName":"suresh",
  "lastName": "ramesh",
  "emailId": "sr@gl.com"
}
```

Status: 200 OK Time: 77 ms Size: 435 B

```
Deleted employee id - 3
```

POSTMAN



```
SELECT * FROM EMPLOYEE
```

SELECT * FROM EMPLOYEE;

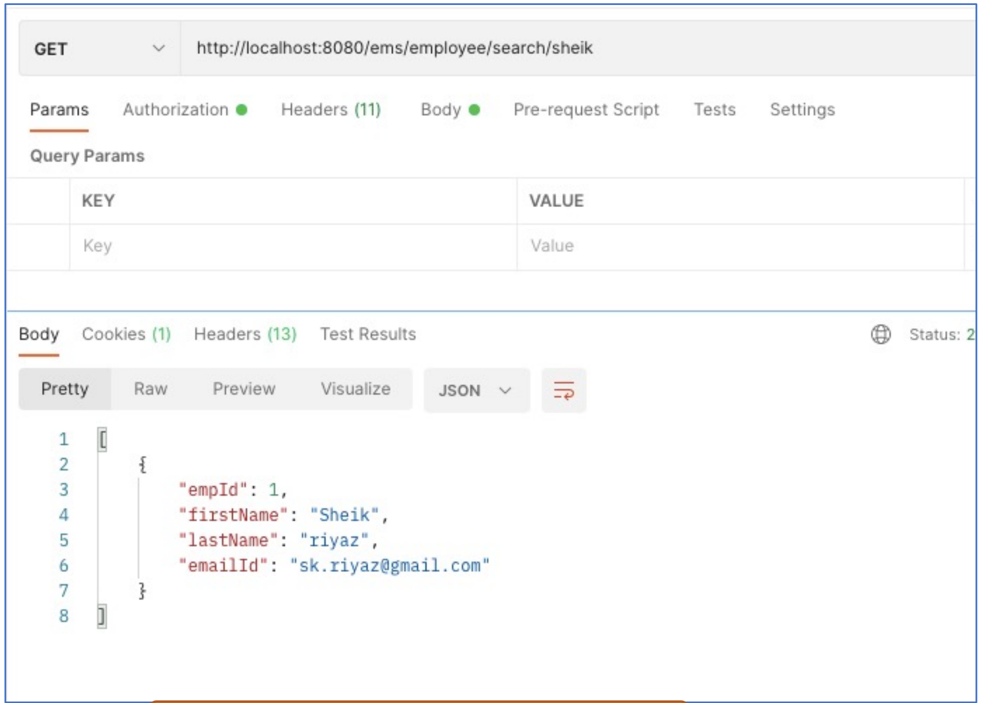
EMP_ID	EMAIL_ID	FIRST_NAME	LAST_NAME
1	sk.riyaz@gmail.com	Sheik	riyaz
2	gl@gmail.com	great	learning

(2 rows, 3 ms)

H2-Console

8. Searching Employee with first name

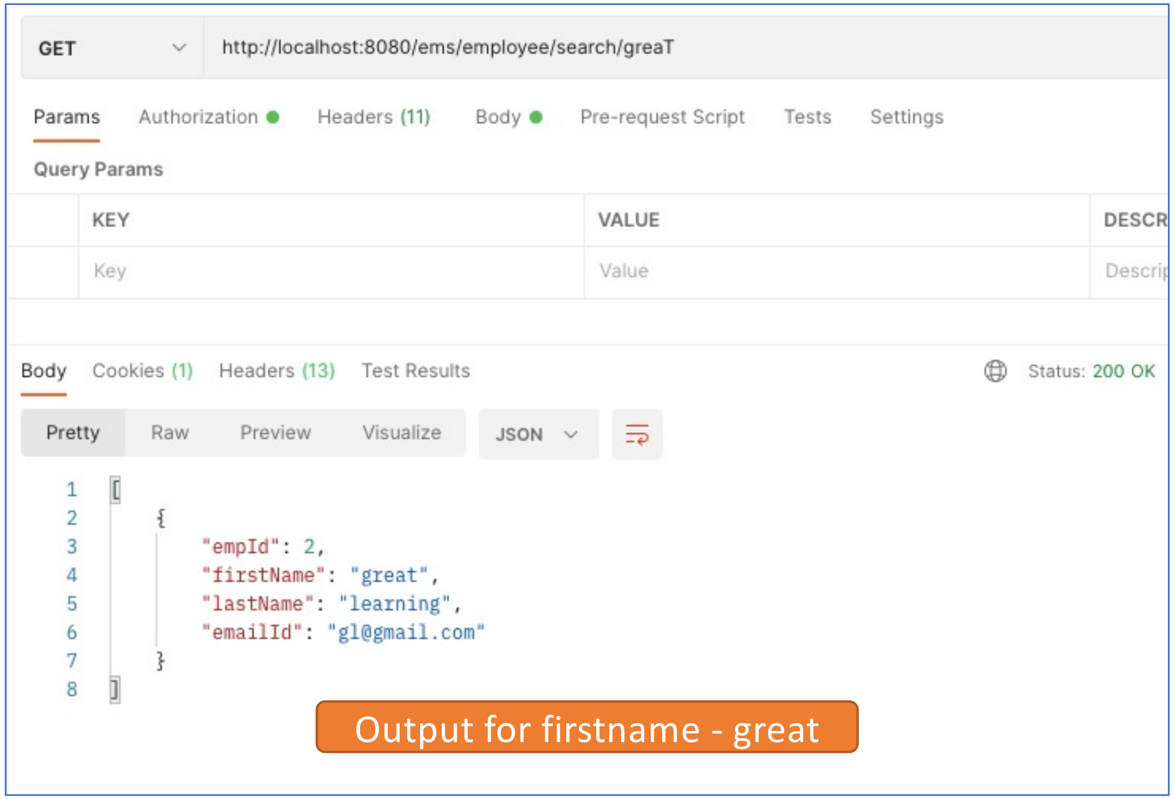
URL: http://localhost:8080/ems/employee/search/sheik & http://localhost:8080/ems/employee/search/greaT



GET http://localhost:8080/ems/employee/search/sheik

```
[
    {
        "empId": 1,
        "firstName": "Sheik",
        "lastName": "riyaz",
        "emailId": "sk.riyaz@gmail.com"
    }
]
```

Output for firstname - Sheik



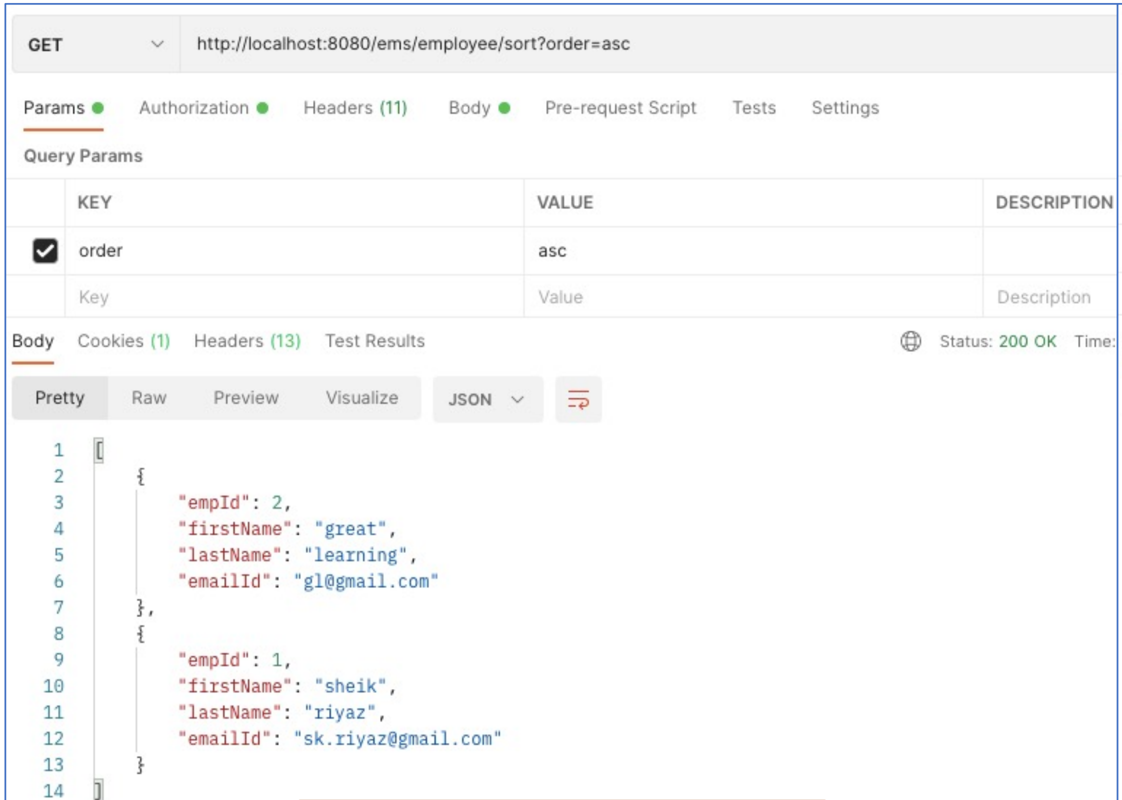
GET http://localhost:8080/ems/employee/search/greaT

```
[
    {
        "empId": 2,
        "firstName": "great",
        "lastName": "learning",
        "emailId": "gl@gmail.com"
    }
]
```

Output for firstname - great

7. Sorting(Asc & Desc) Employees with first name

URL: http://localhost:8080/ems/employee/sort?order=asc & http://localhost:8080/ems/employee/sort?order=desc



GET http://localhost:8080/ems/employee/sort?order=asc

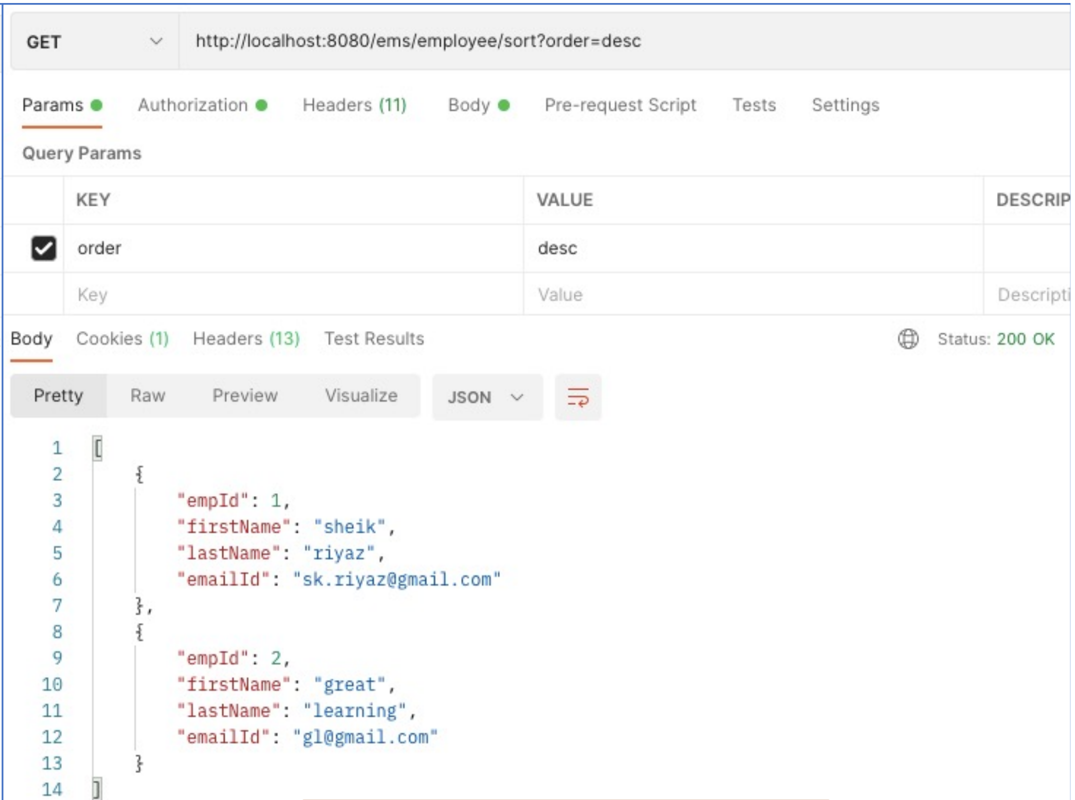
Query Params

	KEY	VALUE	DESCRIPTION
☑	order	asc	
	Key	Value	Description

Status: 200 OK Time:

```
[
    {
        "empId": 2,
        "firstName": "great",
        "lastName": "learning",
        "emailId": "gl@gmail.com"
    },
    {
        "empId": 1,
        "firstName": "sheik",
        "lastName": "riyaz",
        "emailId": "sk.riyaz@gmail.com"
    }
]
```

Response in Ascending



GET http://localhost:8080/ems/employee/sort?order=desc

Query Params

	KEY	VALUE	DESCRIP
☑	order	desc	
	Key	Value	Descripti

Status: 200 OK

```
[
    {
        "empId": 1,
        "firstName": "sheik",
        "lastName": "riyaz",
        "emailId": "sk.riyaz@gmail.com"
    },
    {
        "empId": 2,
        "firstName": "great",
        "lastName": "learning",
        "emailId": "gl@gmail.com"
    }
]
```

Response in descending