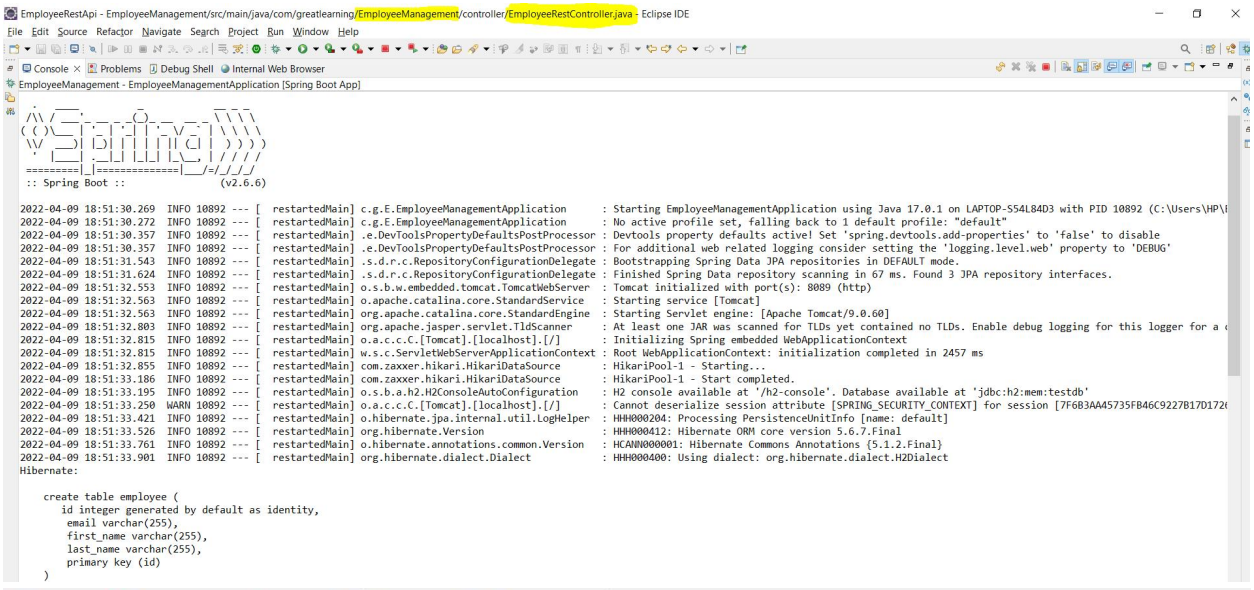
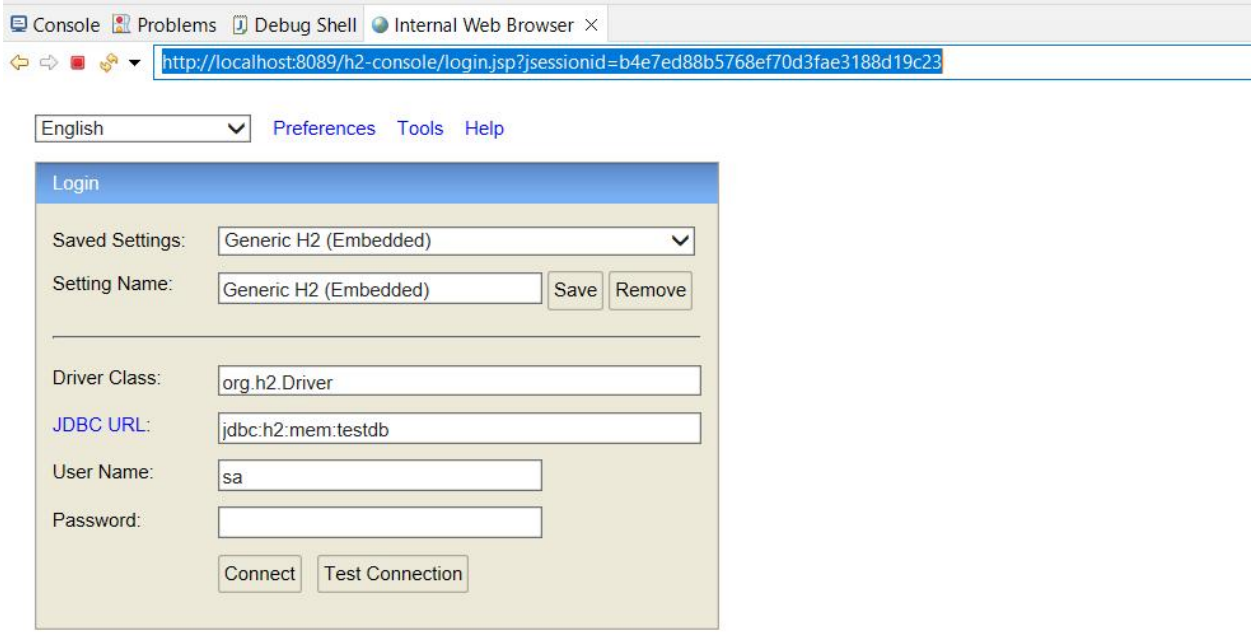
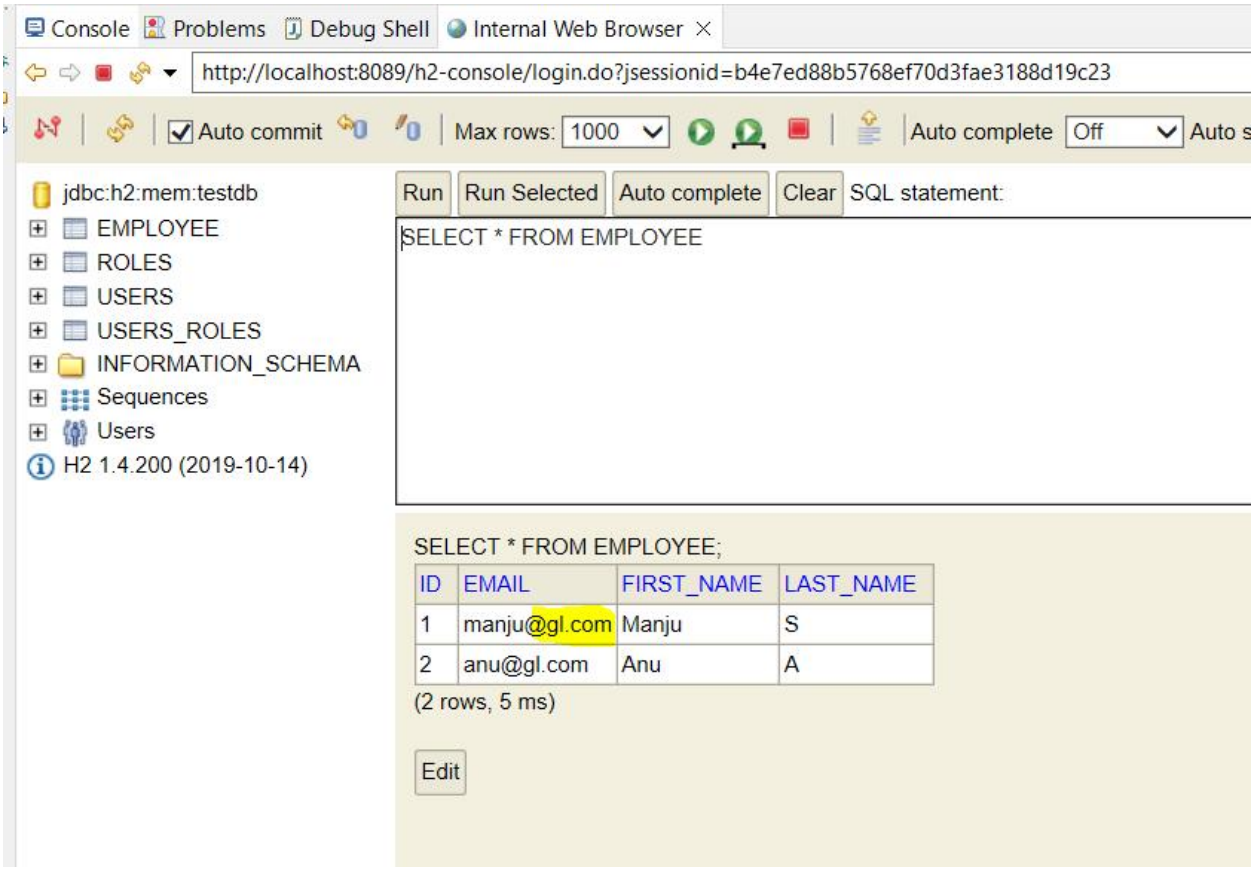
# Graded coding assignment 6

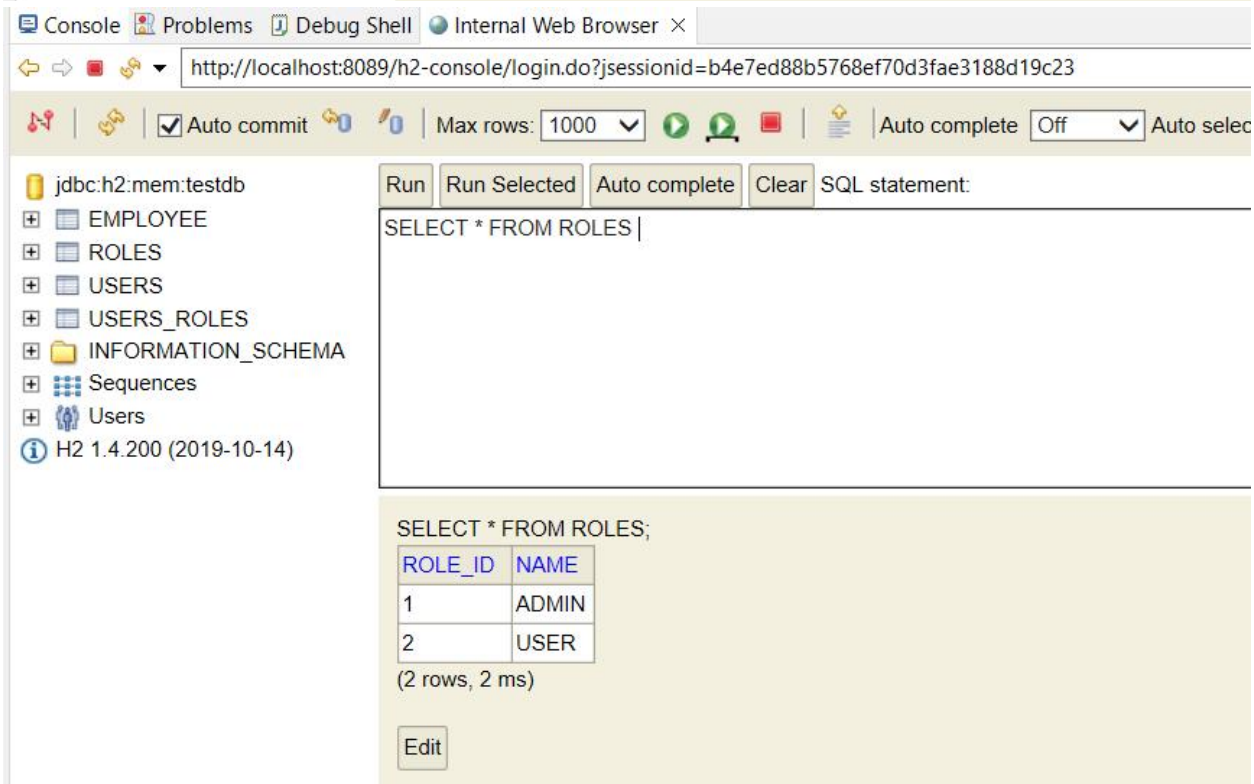
## Employee Management Rest API Assignment

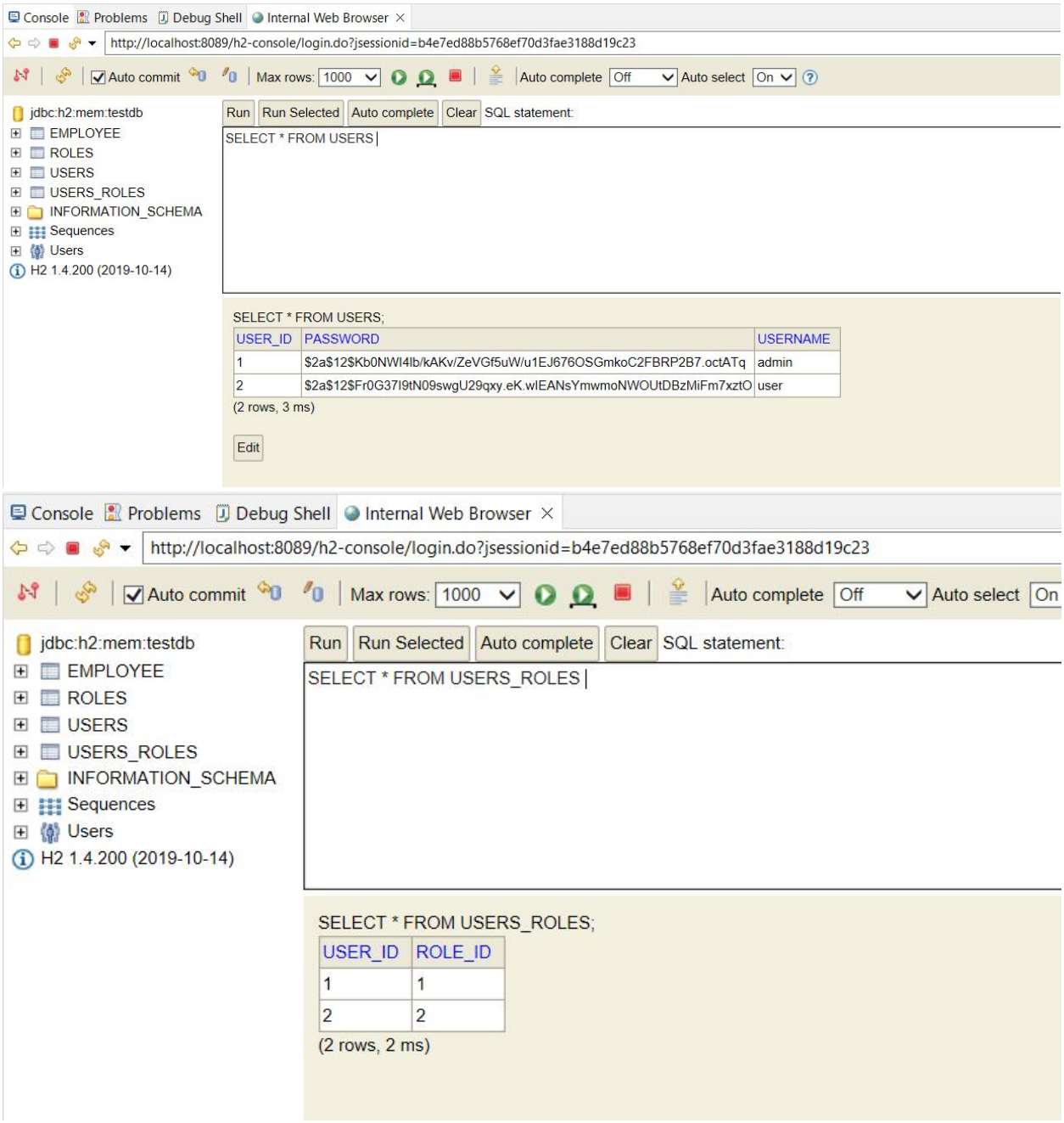
Create a Employee Management Rest Api based Web application, with CRUD(Create,Read,Update and Delete) functionality along with Sorting , searching and some concepts of security.




Console
Problems
Debug Shell
Internal Web Browser
http://localhost:8089/h2-console/login.do?jsessionid=b4e7ed88b5768ef70d3fae3188d19c23
Auto commit
Max rows: 1000
Auto complete Off
jdbc:h2:mem:testdb
EMPLOYEE
ROLES
USERS
USERS_ROLES
INFORMATION_SCHEMA
Sequences
Users
H2 1.4.200 (2019-10-14)
Run
Run Selected
Auto complete
Clear
SQL statement:
SELECT * FROM EMPLOYEE
SELECT * FROM EMPLOYEE;
ID EMAIL FIRST_NAME LAST_NAME
1 manju@gl.com Manju S
2 anu@gl.com Anu A
(2 rows, 5 ms)
Edit


Console
Problems
Debug Shell
Internal Web Browser
http://localhost:8089/h2-console/login.do?jsessionid=b4e7ed88b5768ef70d3fae3188d19c23
Auto commit
Max rows: 1000
Auto complete Off
jdbc:h2:mem:testdb
EMPLOYEE
ROLES
USERS
USERS_ROLES
INFORMATION_SCHEMA
Sequences
Users
H2 1.4.200 (2019-10-14)
Run
Run Selected
Auto complete
Clear
SQL statement:
SELECT * FROM ROLES
SELECT * FROM ROLES;
ROLE_ID NAME
1 ADMIN
2 USER
(2 rows, 2 ms)
Edit




Rest Api is secure, and has different endpoints for different operations as below:

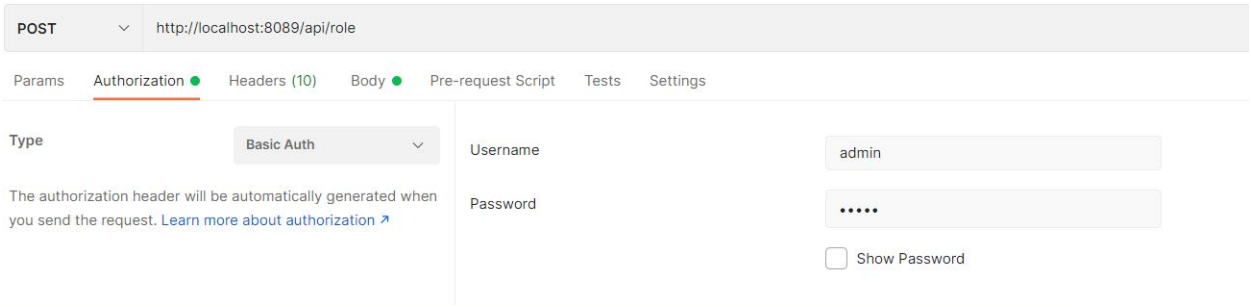
# 1.Add roles in the database dynamically in the db.

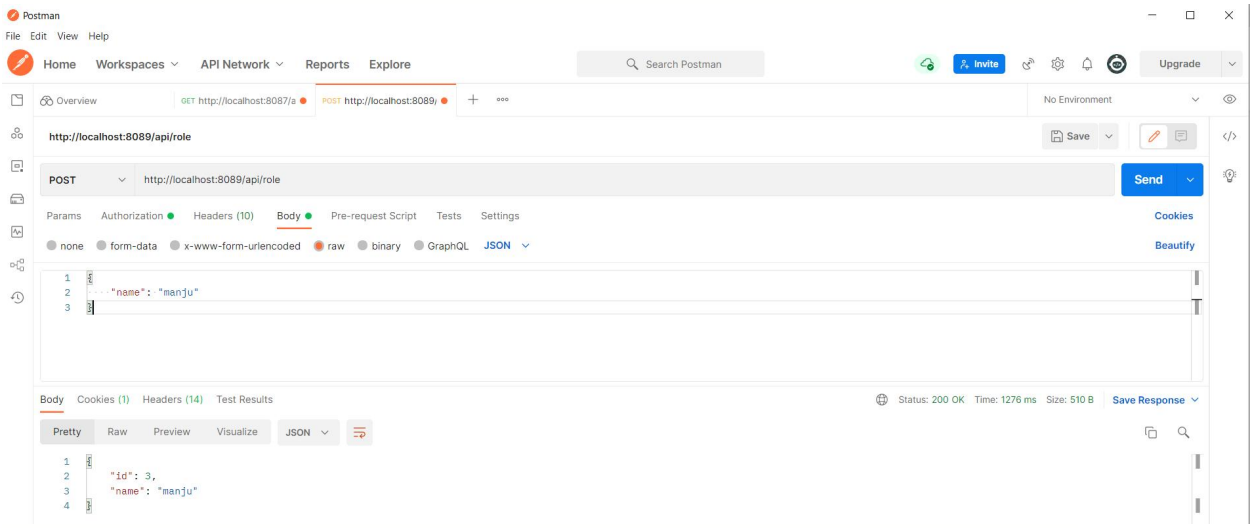
Ex-

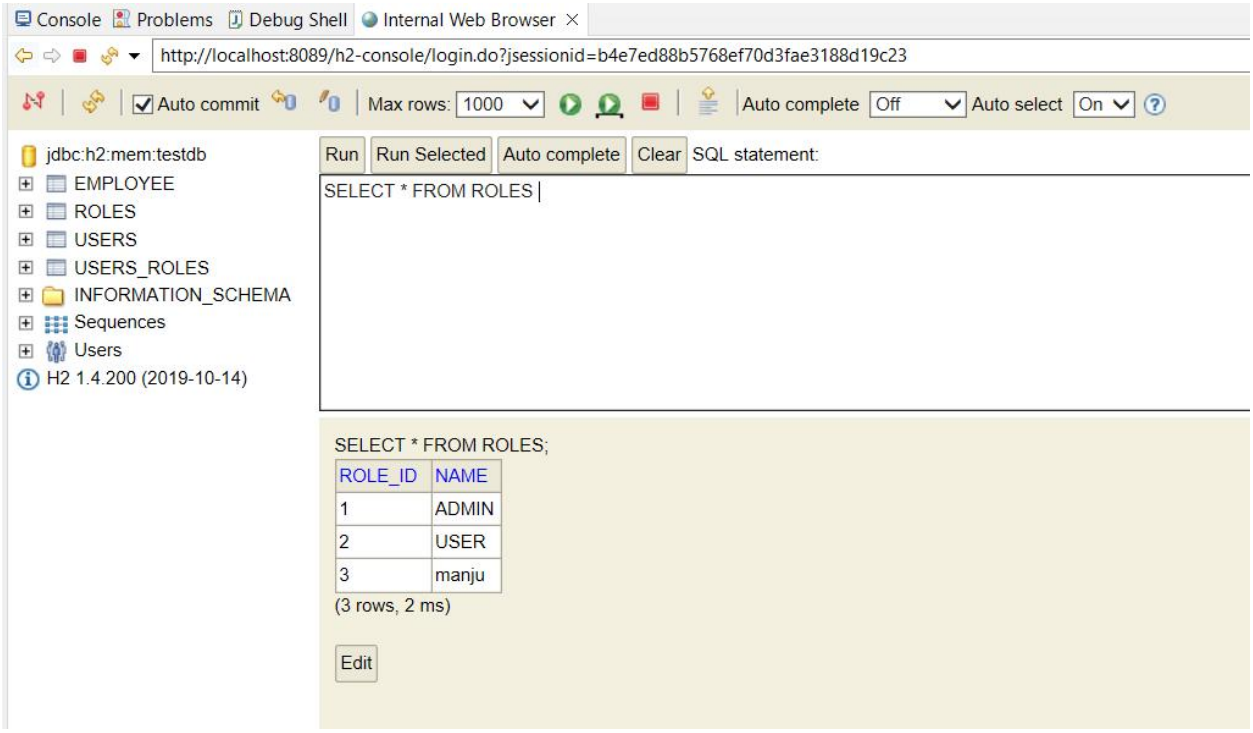
```
{
   "name":"USER"

}
```

Where name specifies a role which can be assigned to a user that will be

used for authentication purposes while interacting with the api.



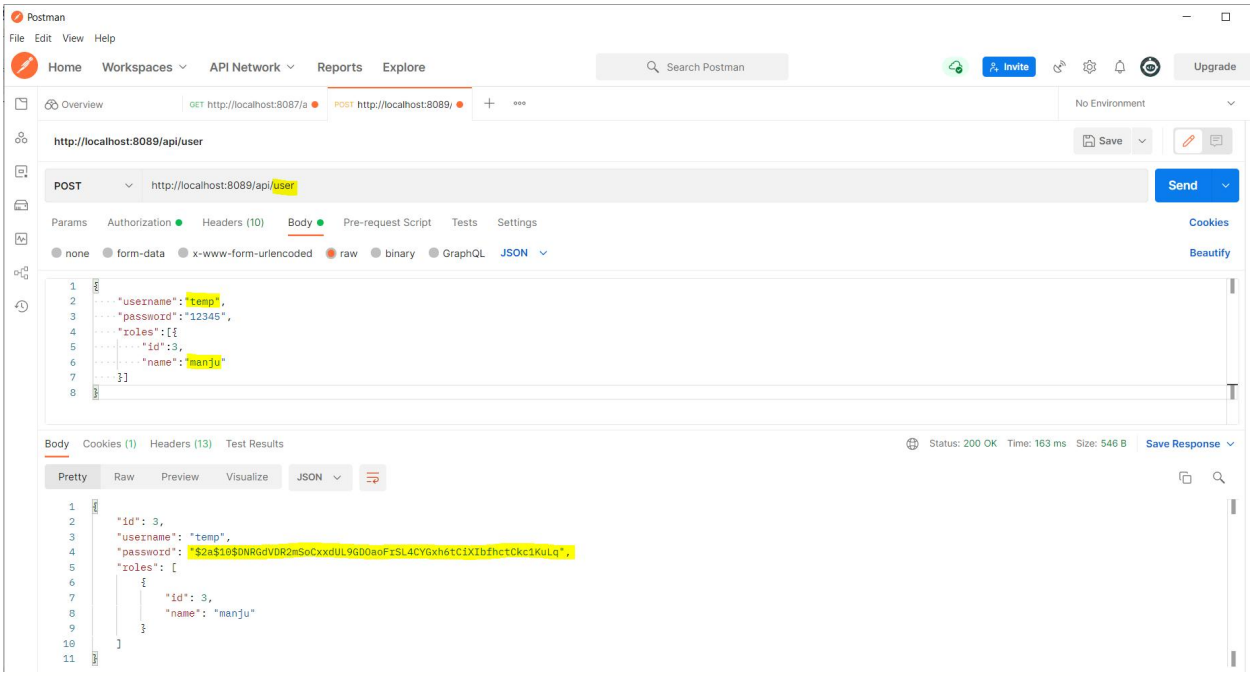



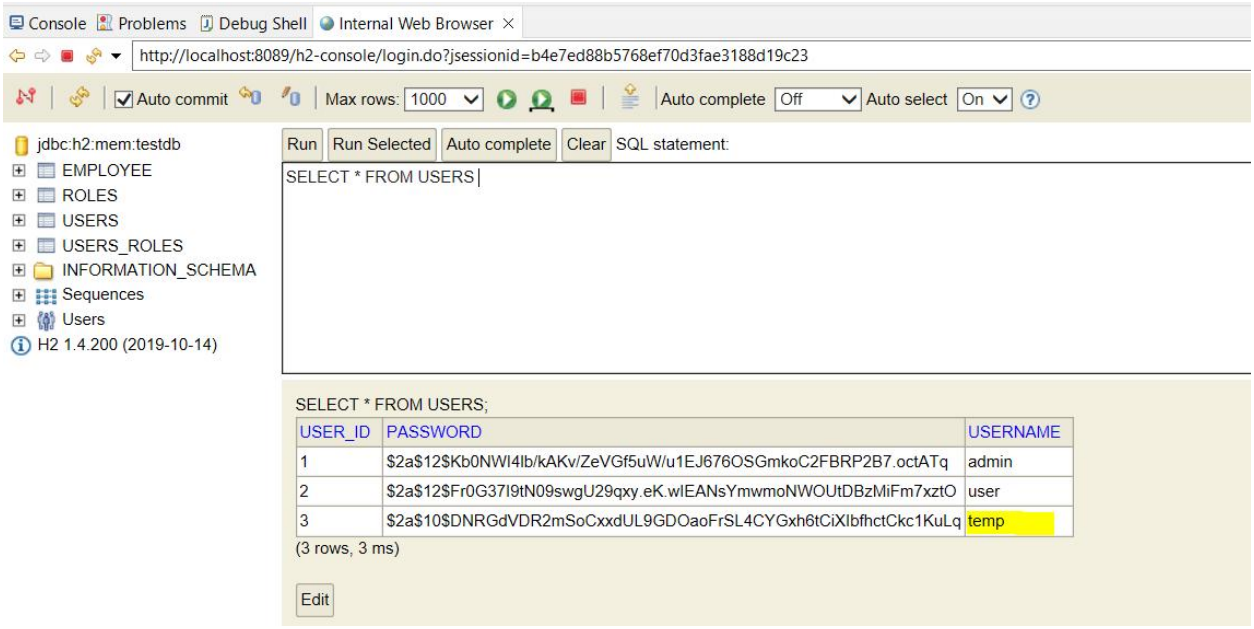



## 2. Add Users in the db which can be used for authentication purposes.

Ex-

```
{
   "username":"temp",
   "password":"12345",
   "roles":[{
      "id":2,
      "name":"USER"
   }]
}
```

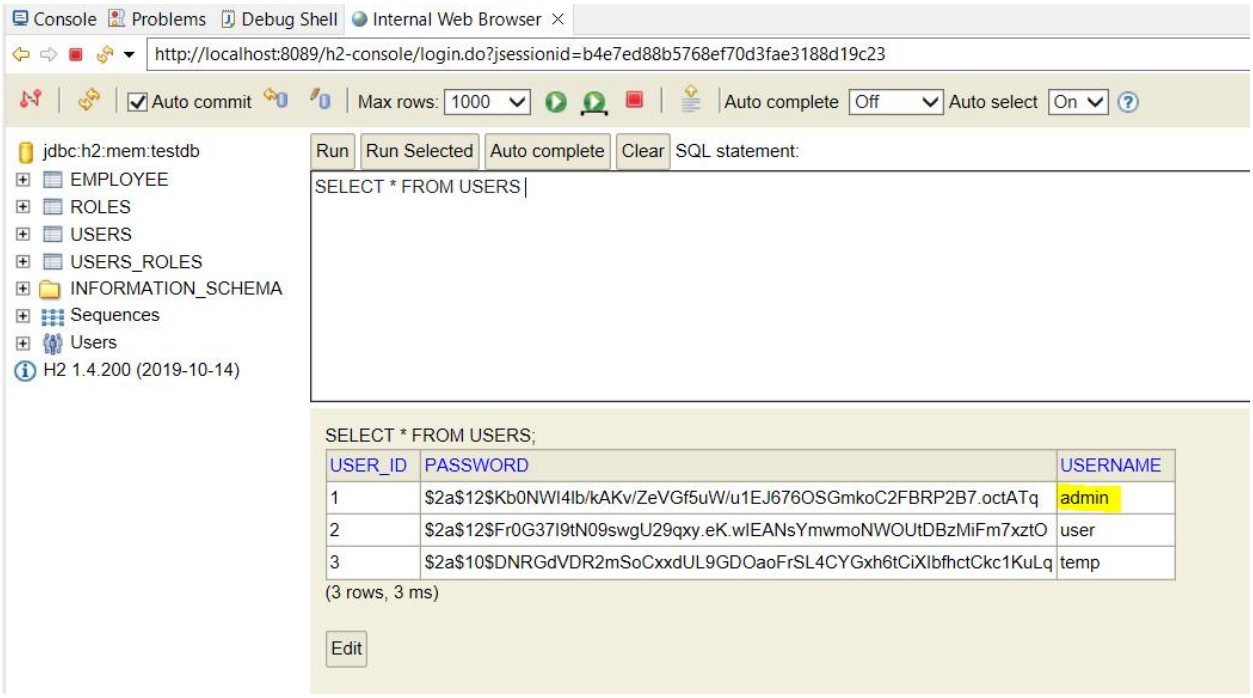


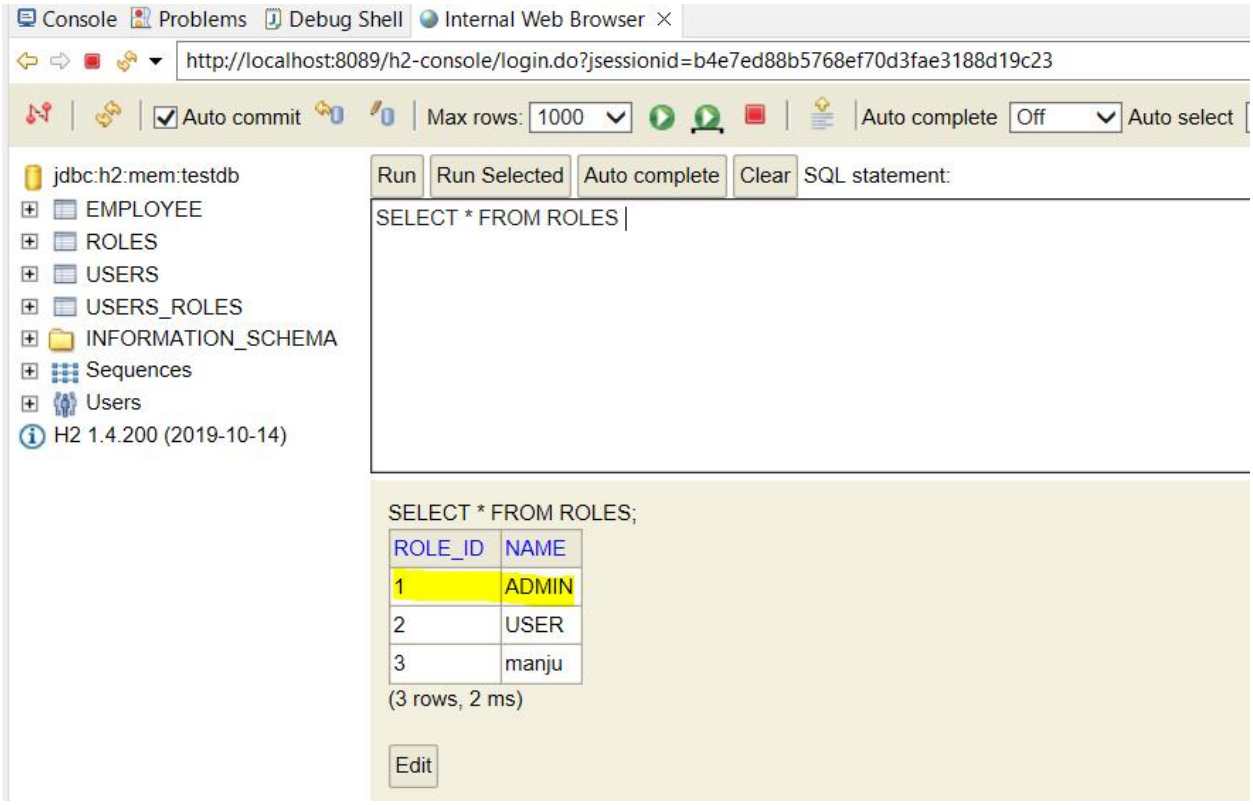


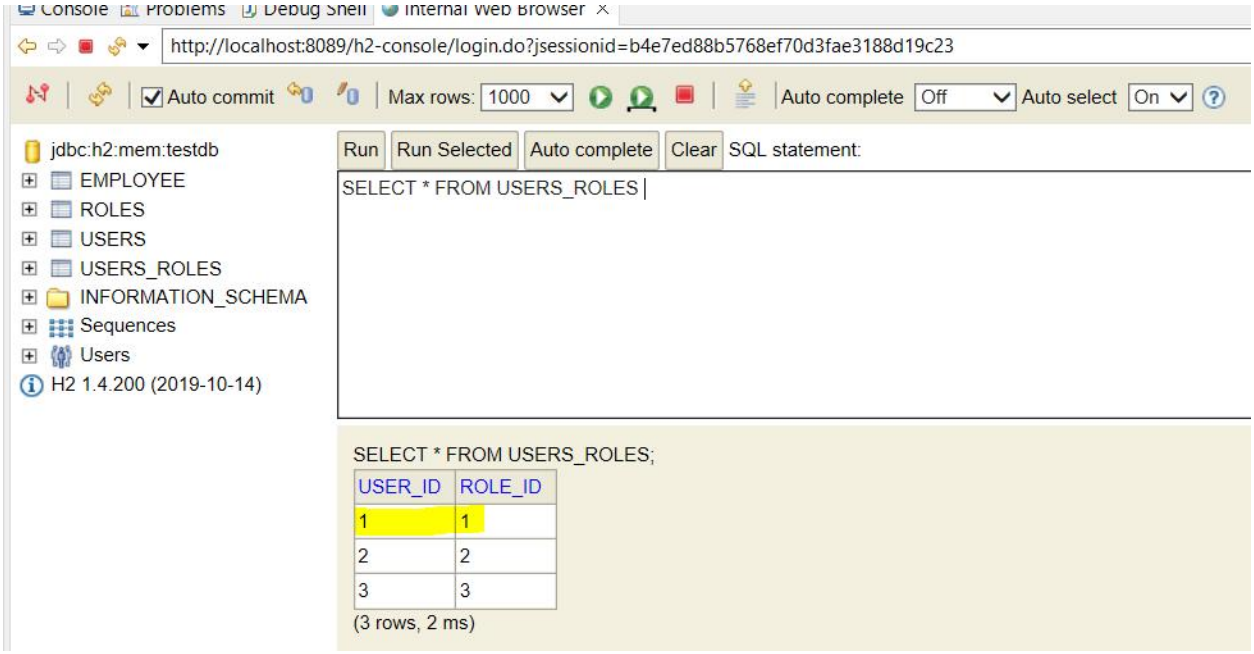

# 3. Add employees data in the db if and only if the authenticated user is ADMIN.

Ex-

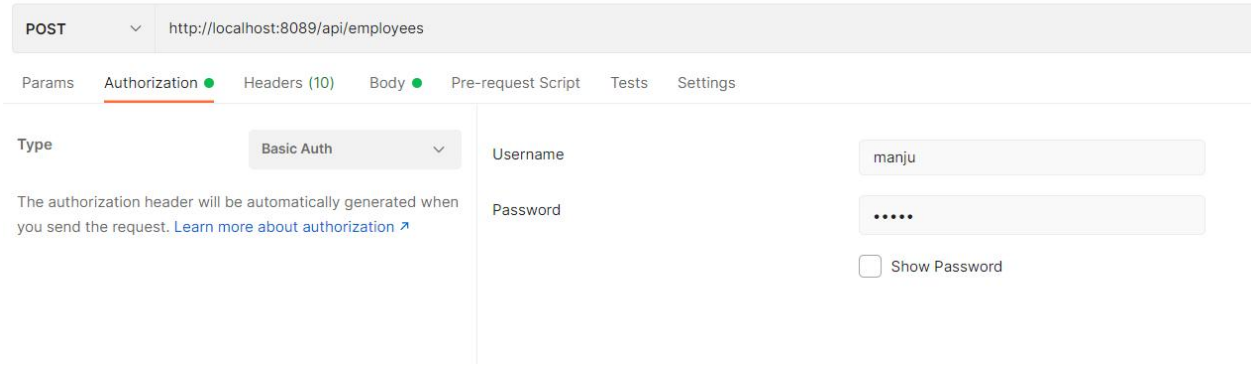
```
{
   "firstName":"gl",
   "lastName":"postman",
   "email":"postman@gamil.com"
}
```

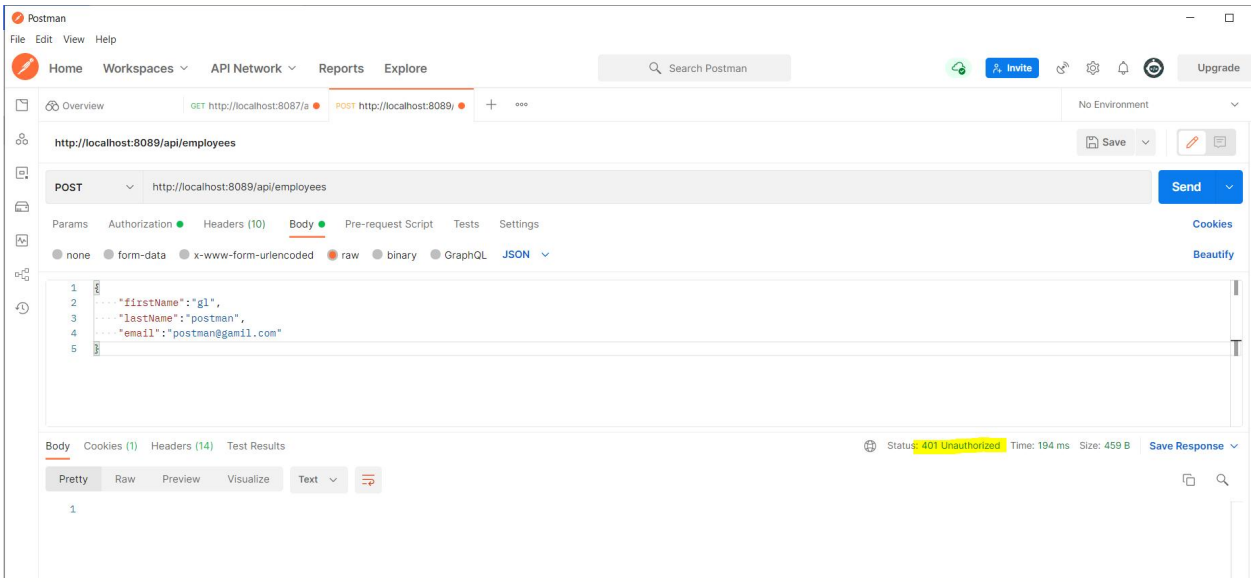

Console
Problems
Debug Shell
Internal Web Browser
http://localhost:8089/h2-console/login.do?jsessionid=b4e7ed88b5768ef70d3fae3188d19c23
Auto commit
Max rows: 1000
Auto complete Off
Auto select On
jdbc:h2:mem:testdb
EMPLOYEE
ROLES
USERS
USERS_ROLES
INFORMATION_SCHEMA
Sequences
Users
H2 1.4.200 (2019-10-14)
Run
Run Selected
Auto complete
Clear
SQL statement:
SELECT * FROM USERS
SELECT * FROM USERS;
USER_ID
PASSWORD
USERNAME
1
$2a$12$Kb0NWI4lb/kAKv/ZeVGf5uW/u1EJ676OSGmkoC2FBRP2B7.octATq
admin
2
$2a$12$Fr0G37I9tN09swgU29qxy.eK.wIEANsYmwmoNWOUtDBzMiFm7xztO
user
3
$2a$10$DNRGdVDR2mSoCxxdUL9GDOaoFrSL4CYGxh6tCiXIbfhctCkc1KuLq
temp
(3 rows, 3 ms)
Edit

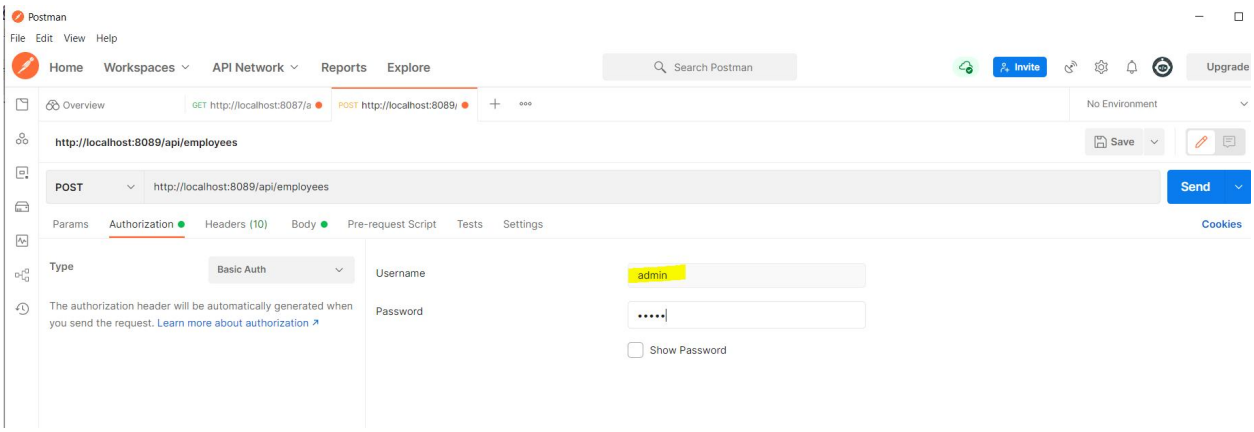

Console
Problems
Debug Shell
Internal Web Browser
http://localhost:8089/h2-console/login.do?jsessionid=b4e7ed88b5768ef70d3fae3188d19c23
Auto commit
Max rows: 1000
Auto complete Off
Auto select
jdbc:h2:mem:testdb
EMPLOYEE
ROLES
USERS
USERS_ROLES
INFORMATION_SCHEMA
Sequences
Users
H2 1.4.200 (2019-10-14)
Run
Run Selected
Auto complete
Clear
SQL statement:
SELECT * FROM ROLES
SELECT * FROM ROLES;
ROLE_ID
NAME
1
ADMIN
2
USER
3
manju
(3 rows, 2 ms)
Edit

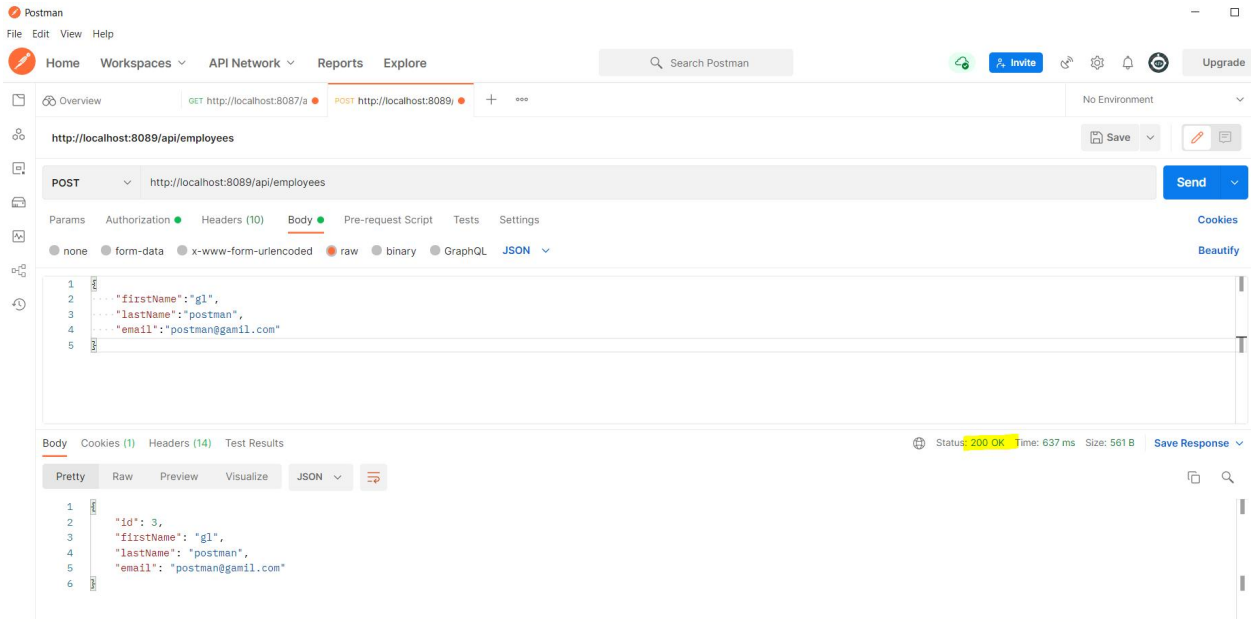


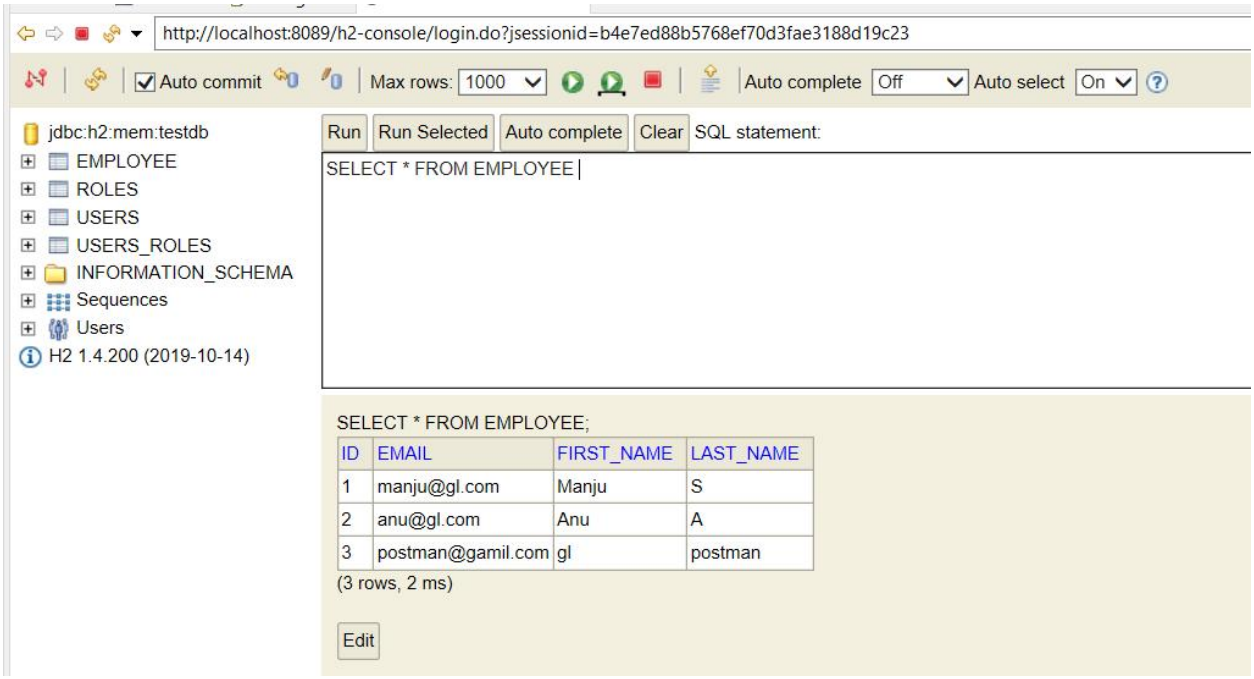

**http://localhost:8089/api/employees**




Postman
File Edit View Help
Home Workspaces API Network Reports Explore
Search Postman
Invite
Upgrade
Overview
GET http://localhost:8087/a
POST http://localhost:8089/
No Environment
http://localhost:8089/api/employees
Save
POST
http://localhost:8089/api/employees
Send
Params Authorization Headers (10) Body Pre-request Script Tests Settings
Cookies
Type
Basic Auth
Username
admin
Password
The authorization header will be automatically generated when you send the request. Learn more about authorization
Show Password


Postman
File Edit View Help
Home Workspaces API Network Reports Explore
Search Postman
Invite
Upgrade
Overview
GET http://localhost:8087/a
POST http://localhost:8089/
No Environment
http://localhost:8089/api/employees
Save
POST
http://localhost:8089/api/employees
Send
Params Authorization Headers (10) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"firstName":"gl",
"lastName":"postman",
"email":"postman@gamil.com"
}
Body Cookies (1) Headers (14) Test Results
Status: 200 OK Time: 637 ms Size: 561 B
Save Response
Pretty Raw Preview Visualize JSON
{
"id": 3,
"firstName": "gl",
"lastName": "postman",
"email": "postman@gamil.com"
}



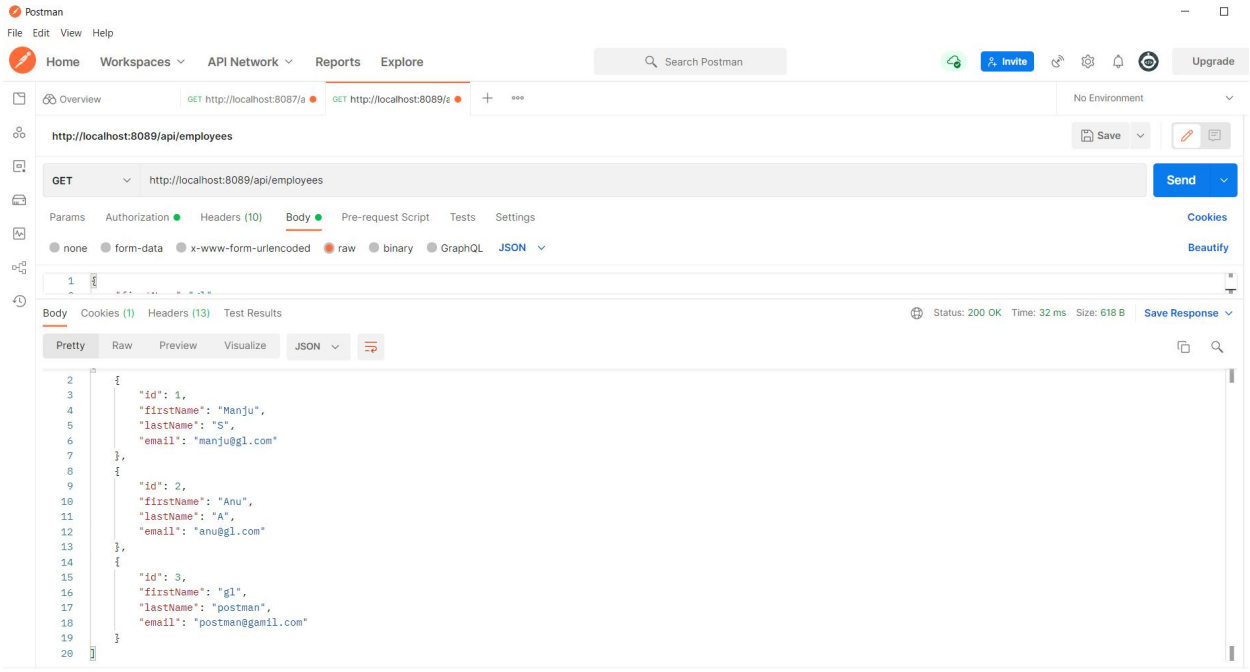

# 4. List all the employees stored in the database.

Ex-

Response Body-


```
[
  {
    "id": 1,
    "firstName": "Ujjawal",
    "lastName": "Sharma",
    "email": "fdfdfd@gmail.com"
  },
  {
    "id": 2,
    "firstName": "temp",
    "lastName": "kaushik",
    "email": "jdfdkfdjj@gmail.com"
  },
```

```
    {
        "id": 3,
        "firstName": "postman",
        "lastName": "postman",
        "email": "postman@gamil.com"
    }
]
```

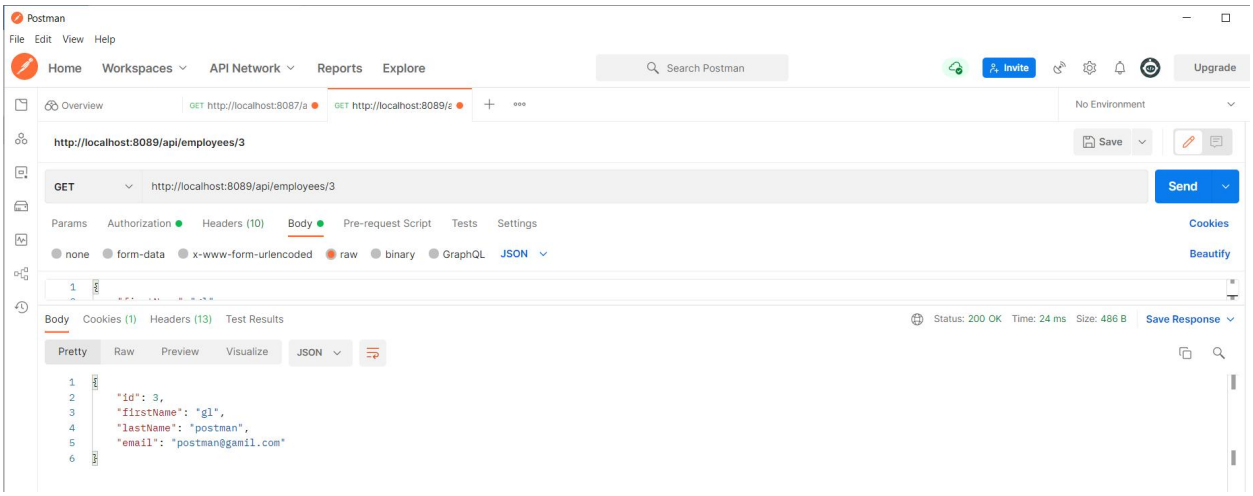




# 5. Fetch or get an employee record specifically based on the id of that employee.

Ex- Url- http://localhost:8080/api/employees/3

Response Body-


```
{
    "id": 3,
    "firstName": "postman",
    "lastName": "postman",
    "email": "postman@gamil.com"
}
```




# 6. Update an existing employee record with the given updated json object.

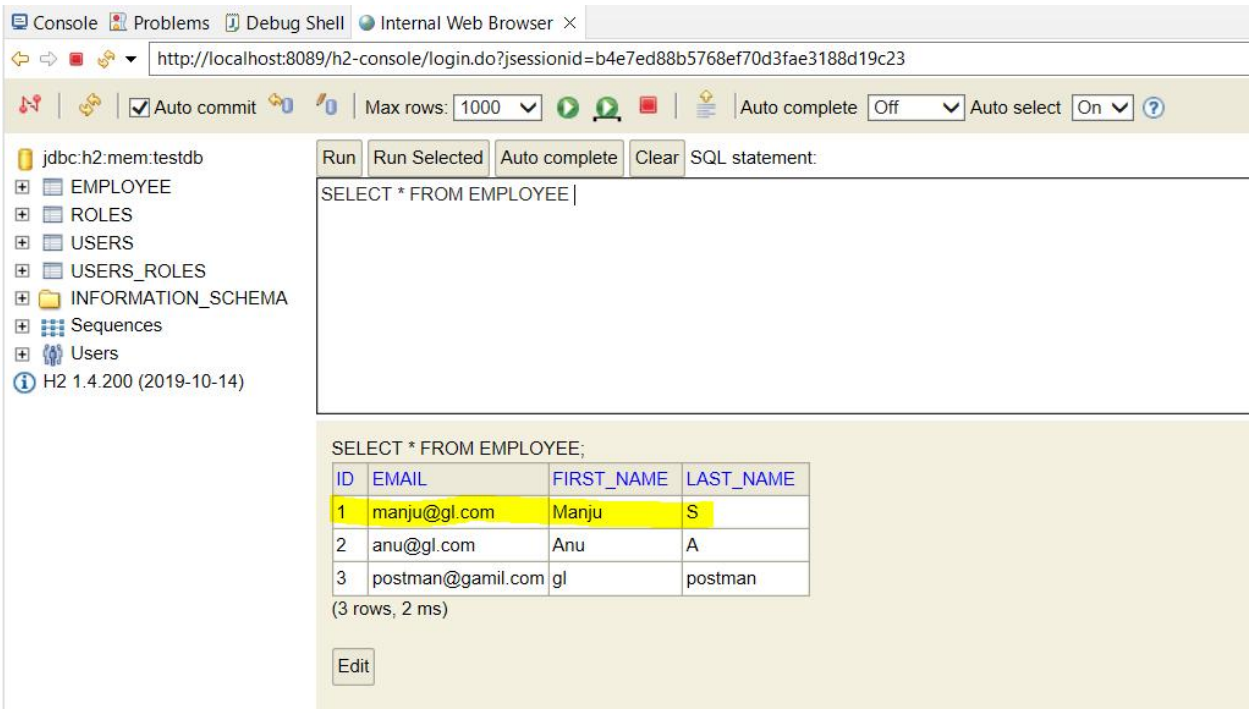
Ex-

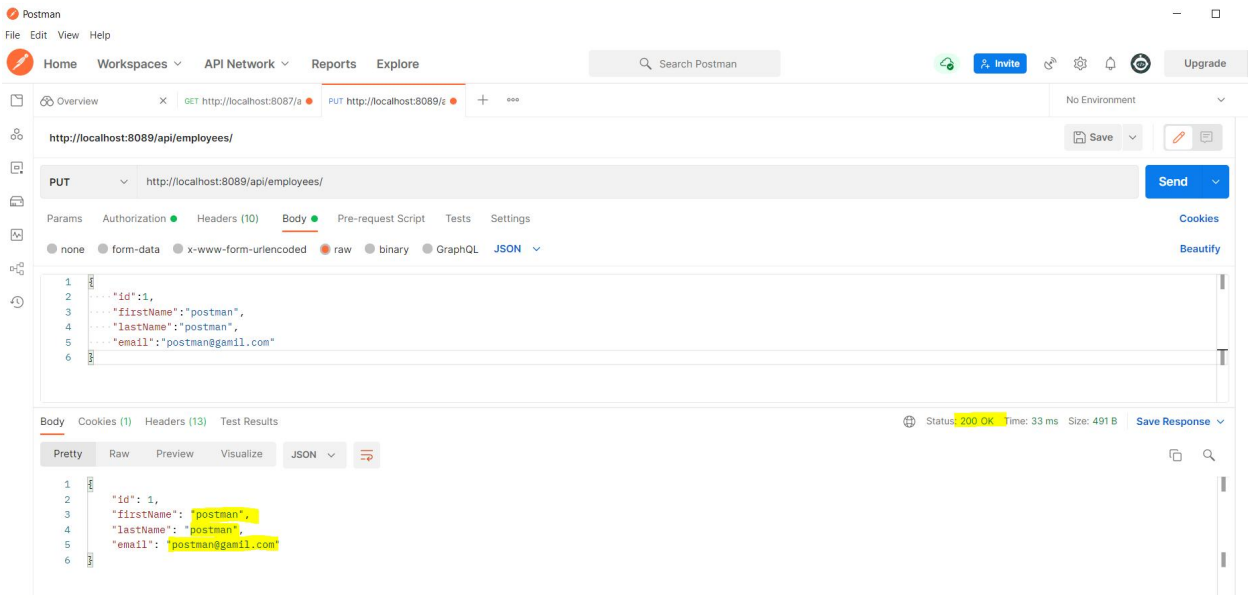
Object to be updated(raw->Json)-

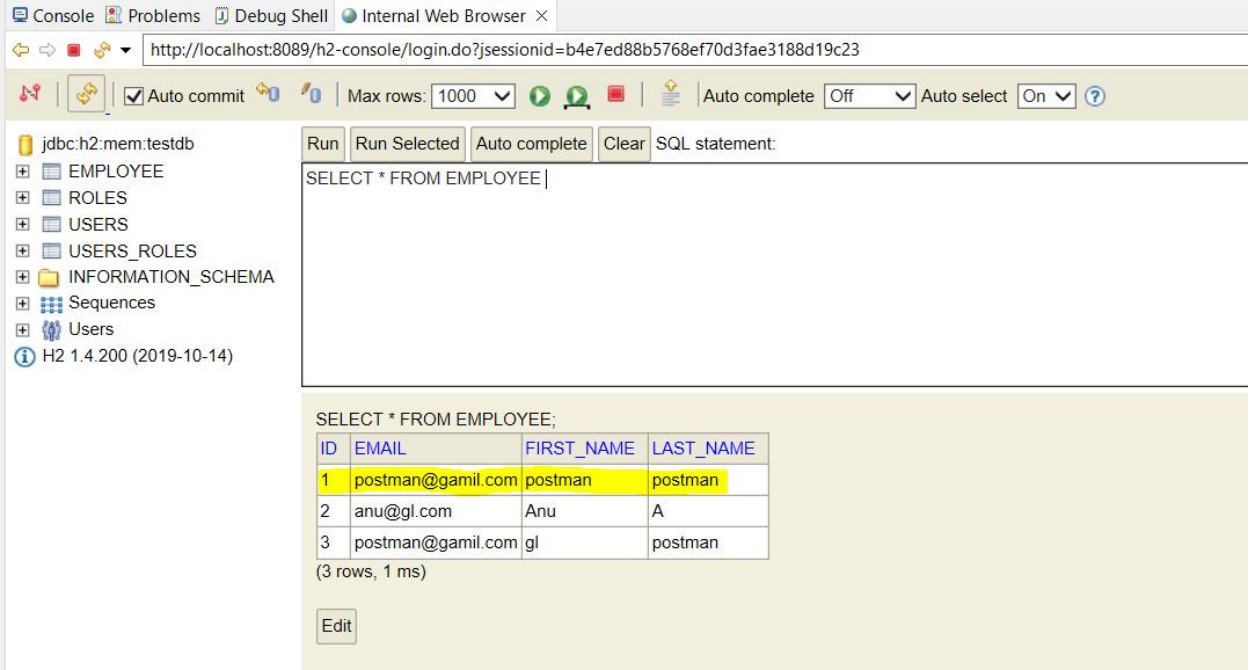
```
{
   "id":1,
   "firstName":"postman",
   "lastName":"postman",
   "email":"postman@gamil.com"
}
```

Response Body after updation-

```
{
   "id": 1,
   "firstName": "postman",
   "lastName": "postman",
   "email": "postman@gamil.com"
}
```


Console
Problems
Debug Shell
Internal Web Browser
http://localhost:8089/h2-console/login.do?jsessionid=b4e7ed88b5768ef70d3fae3188d19c23
Auto commit
Max rows: 1000
Auto complete Off
Auto select On
jdbc:h2:mem:testdb
EMPLOYEE
ROLES
USERS
USERS_ROLES
INFORMATION_SCHEMA
Sequences
Users
H2 1.4.200 (2019-10-14)
Run
Run Selected
Auto complete
Clear
SQL statement:
SELECT * FROM EMPLOYEE
SELECT * FROM EMPLOYEE;
ID EMAIL FIRST_NAME LAST_NAME
1 manju@gl.com Manju S
2 anu@gl.com Anu A
3 postman@gamil.com gl postman
(3 rows, 2 ms)
Edit


Postman
File Edit View Help
Home
Workspaces
API Network
Reports
Explore
Search Postman
Invite
Upgrade
No Environment
http://localhost:8089/api/employees/
Save
PUT
http://localhost:8089/api/employees/
Send
Params
Authorization
Headers (10)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"id":1,
"firstName":"postman",
"lastName":"postman",
"email":"postman@gamil.com"
}
Body
Cookies (1)
Headers (13)
Test Results
Status: 200 OK
Time: 33 ms
Size: 491 B
Save Response
Pretty
Raw
Preview
Visualize
JSON
{
"id": 1,
"firstName": "postman",
"lastName": "postman",
"email": "postman@gamil.com"
}



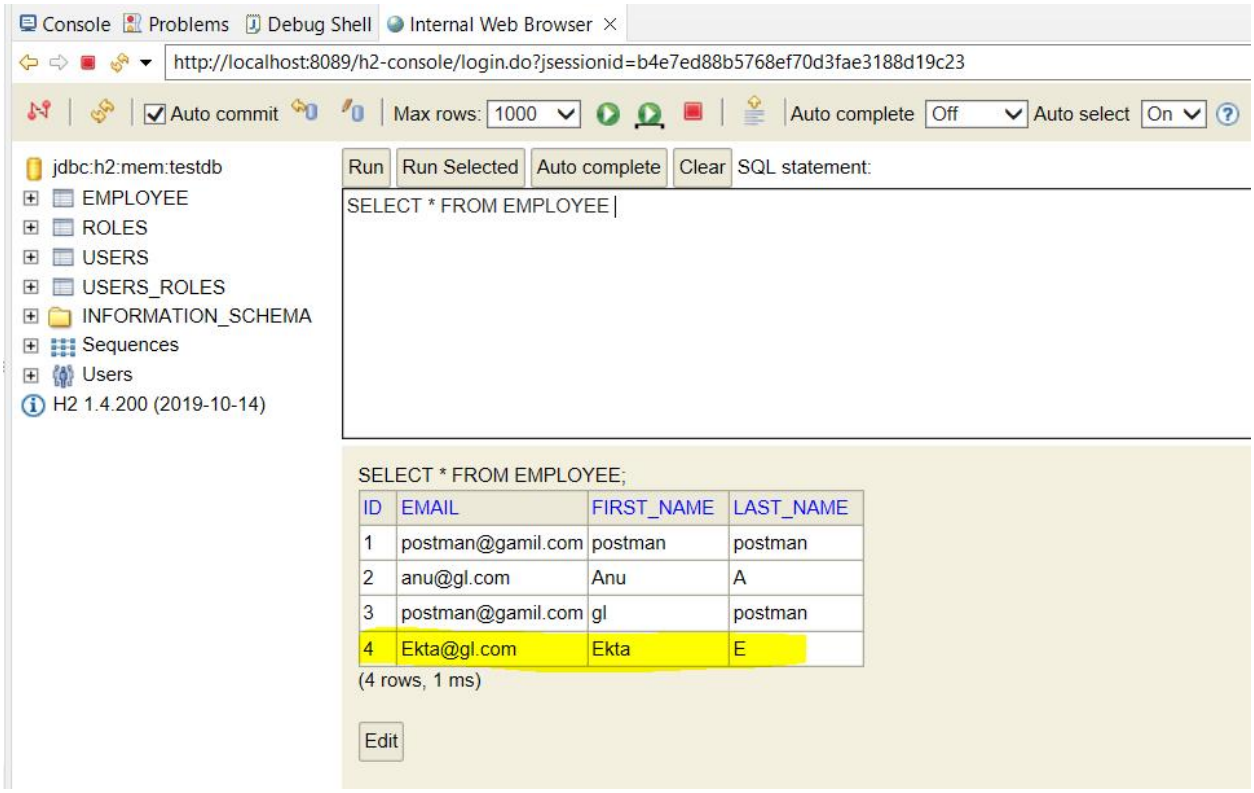

# 7. Delete an existing employee record based on the id of the employee.

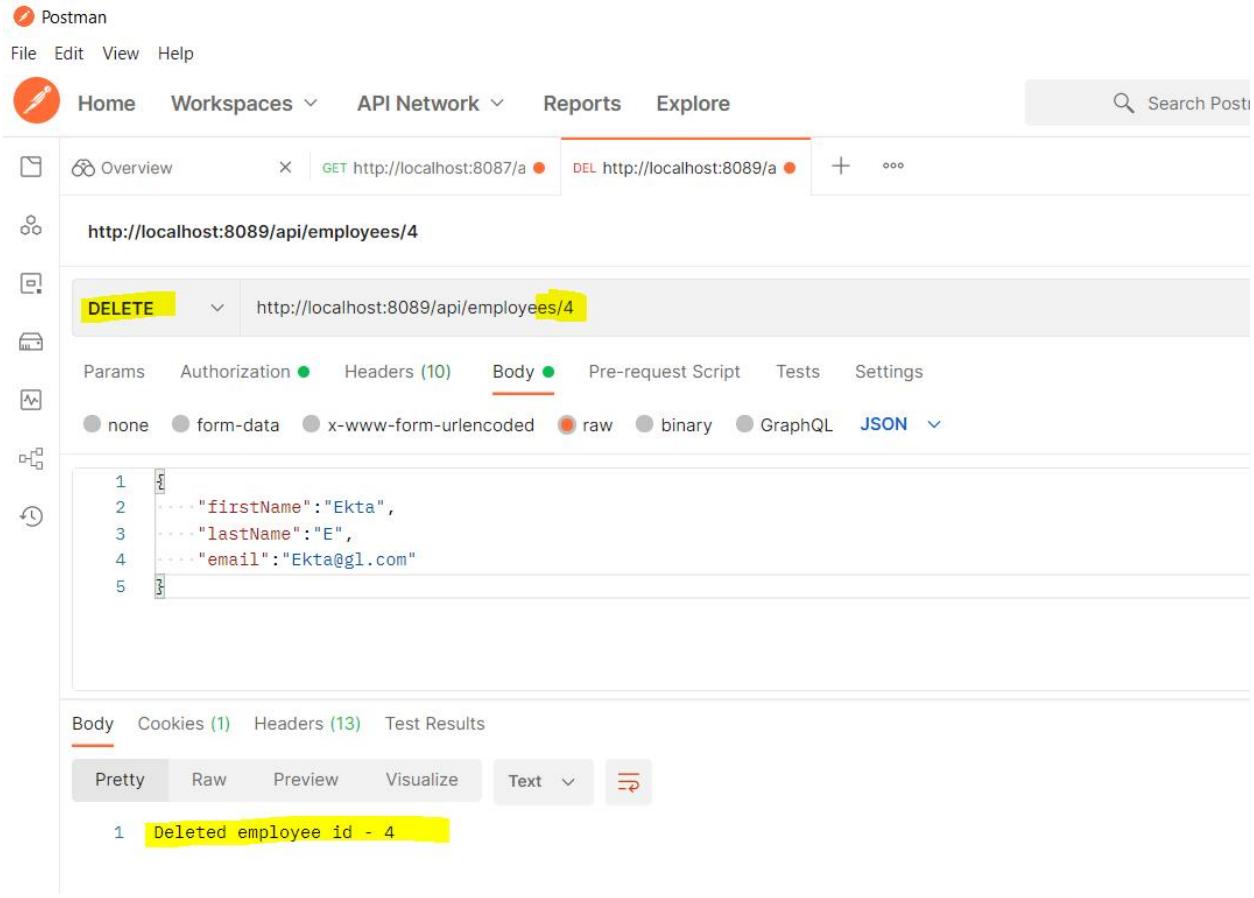
Ex-

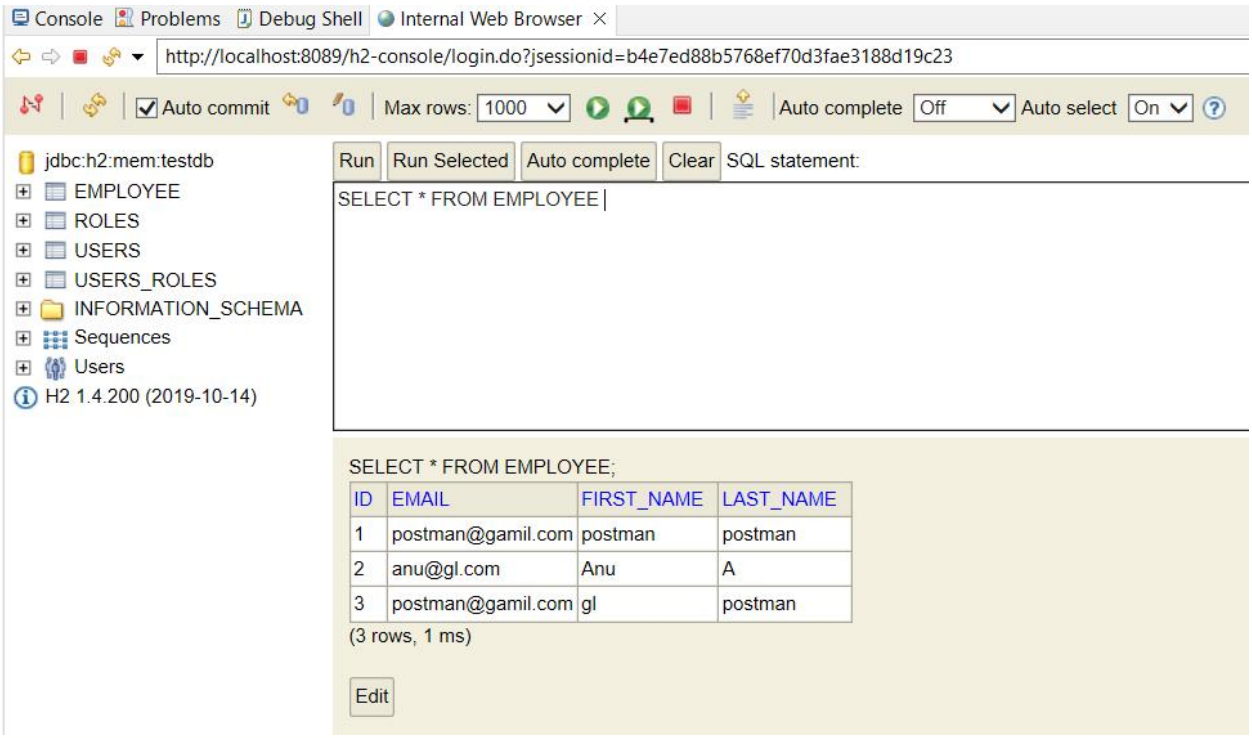
Url- http://localhost:8080/api/employees/4

Response Body-

"Deleted employee id - 4”


Console
Problems
Debug Shell
Internal Web Browser
http://localhost:8089/h2-console/login.do?jsessionid=b4e7ed88b5768ef70d3fae3188d19c23
Auto commit
Max rows: 1000
Auto complete Off
Auto select On
jdbc:h2:mem:testdb
EMPLOYEE
ROLES
USERS
USERS_ROLES
INFORMATION_SCHEMA
Sequences
Users
H2 1.4.200 (2019-10-14)
Run
Run Selected
Auto complete
Clear
SQL statement:
SELECT * FROM EMPLOYEE
SELECT * FROM EMPLOYEE;
ID EMAIL FIRST_NAME LAST_NAME
1 postman@gamil.com postman postman
2 anu@gl.com Anu A
3 postman@gamil.com gl postman
4 Ekta@gl.com Ekta E
(4 rows, 1 ms)
Edit


Postman
File Edit View Help
Home
Workspaces
API Network
Reports
Explore
Search Postman
Overview
GET http://localhost:8087/a
DEL http://localhost:8089/a
http://localhost:8089/api/employees/4
DELETE
http://localhost:8089/api/employees/4
Params
Authorization
Headers (10)
Body
Pre-request Script
Tests
Settings
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
{
"firstName":"Ekta",
"lastName":"E",
"email":"Ekta@gl.com"
}
Body
Cookies (1)
Headers (13)
Test Results
Pretty
Raw
Preview
Visualize
Text
Deleted employee id - 4



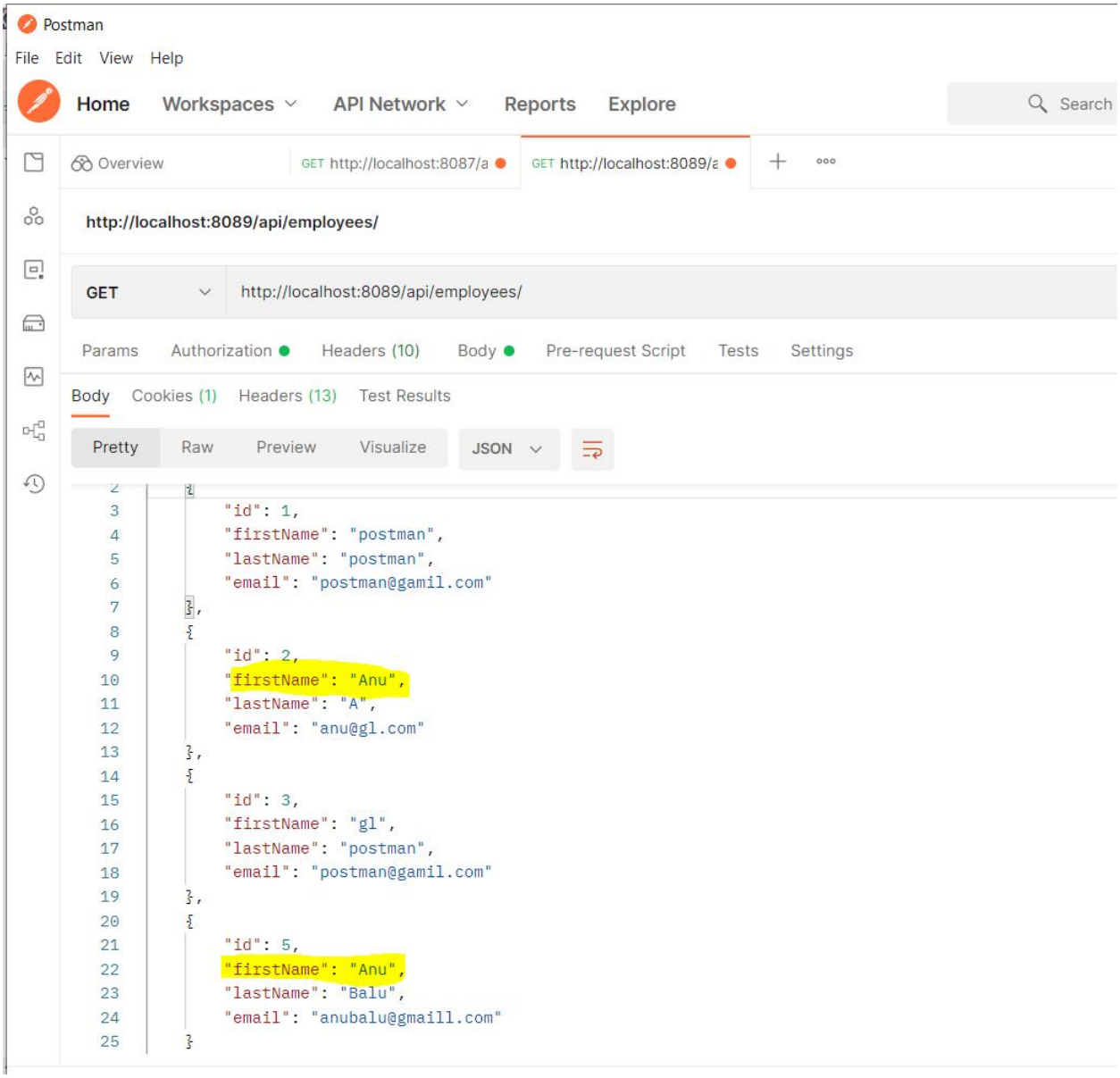

# 8. Fetch an employee by his/her first name and if found more than one record then list them all-

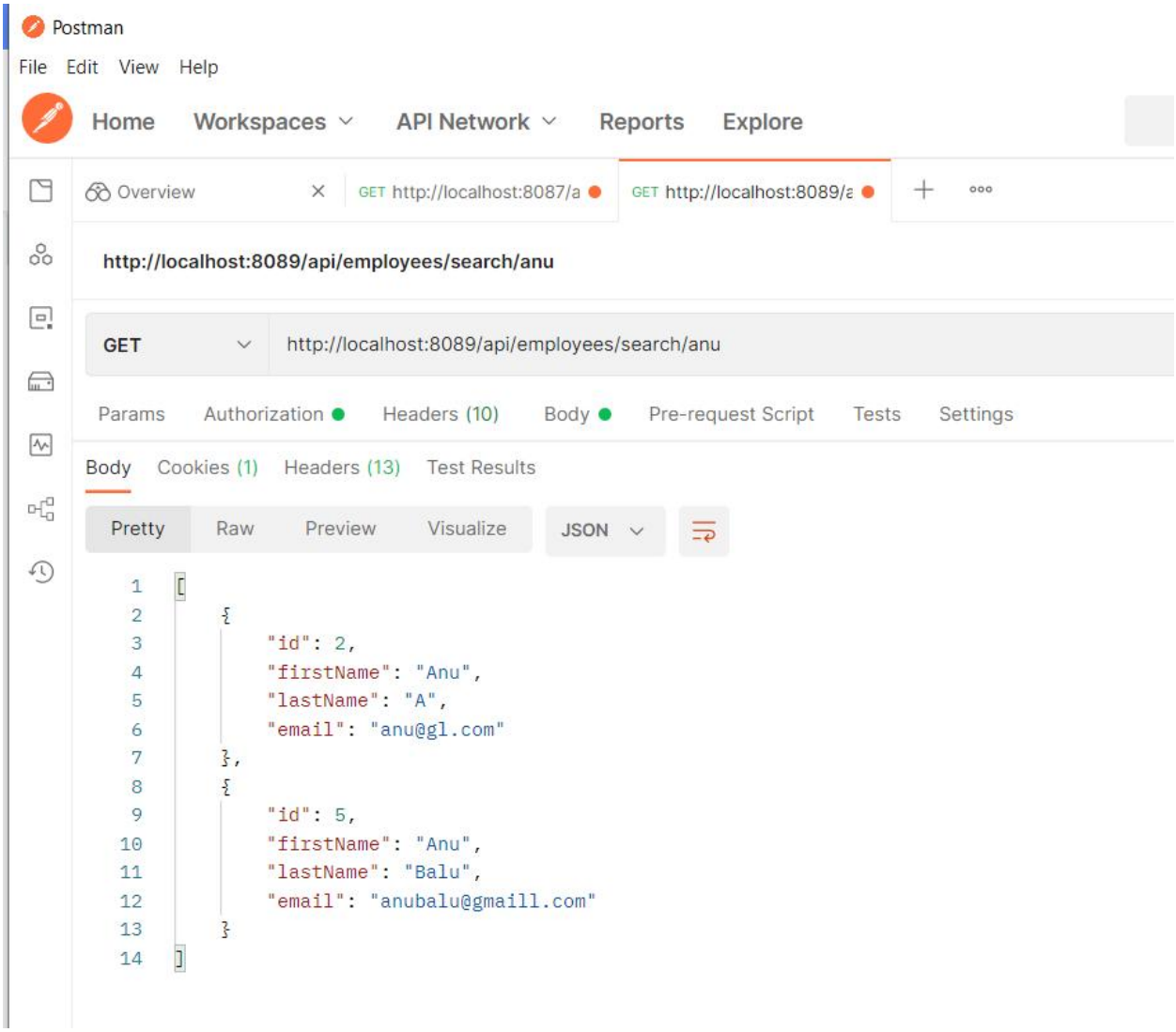
Ex-

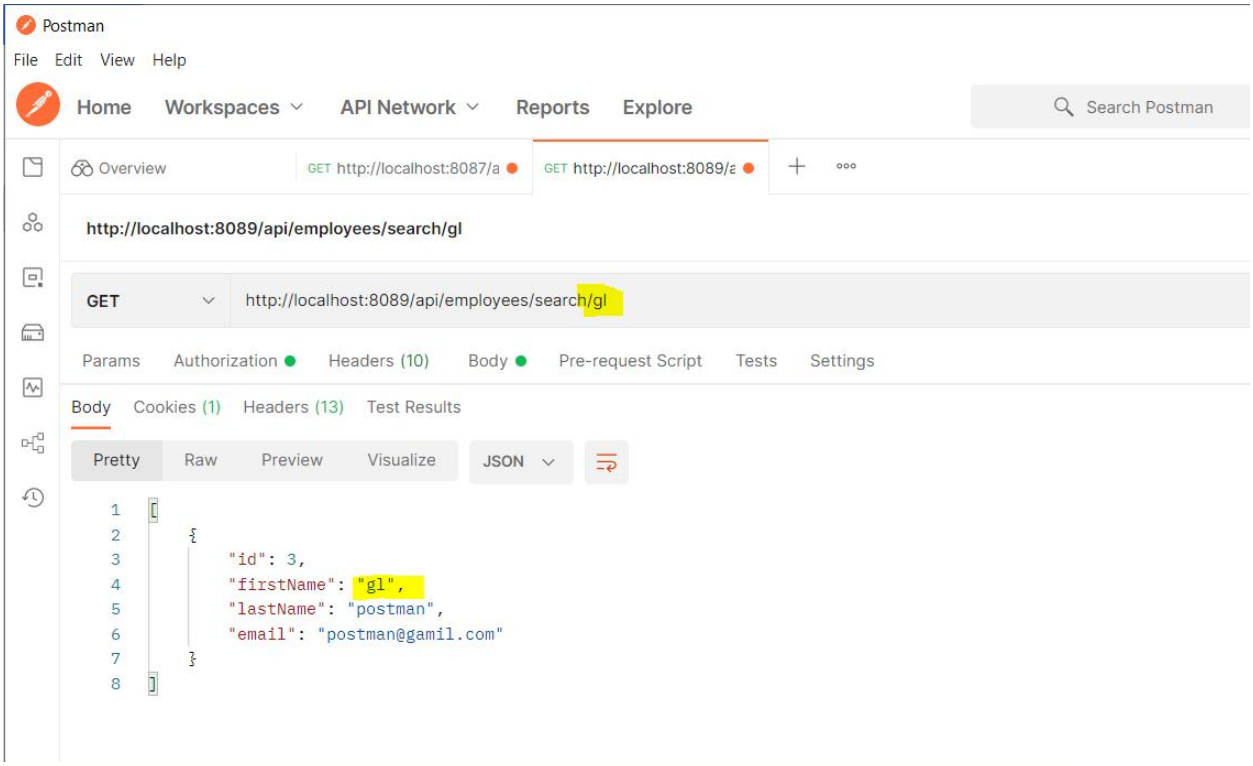
Url- http://localhost:8080/api/employees/search/gl

Response Body-


```
[
  {
    "id": 11,
    "firstName": "gl",
    "lastName": "postman",
    "email": "postman@gamil.com"
  }
]
```


Postman
File Edit View Help
Home Workspaces API Network Reports Explore
Search
Overview
GET http://localhost:8087/a
GET http://localhost:8089/a
http://localhost:8089/api/employees/
GET
http://localhost:8089/api/employees/
Params Authorization Headers (10) Body Pre-request Script Tests Settings
Body Cookies (1) Headers (13) Test Results
Pretty Raw Preview Visualize JSON
{
"id": 1,
"firstName": "postman",
"lastName": "postman",
"email": "postman@gamil.com"
},
{
"id": 2,
"firstName": "Anu",
"lastName": "A",
"email": "anu@gl.com"
},
{
"id": 3,
"firstName": "gl",
"lastName": "postman",
"email": "postman@gamil.com"
},
{
"id": 5,
"firstName": "Anu",
"lastName": "Balu",
"email": "anubalu@gmaill.com"
}

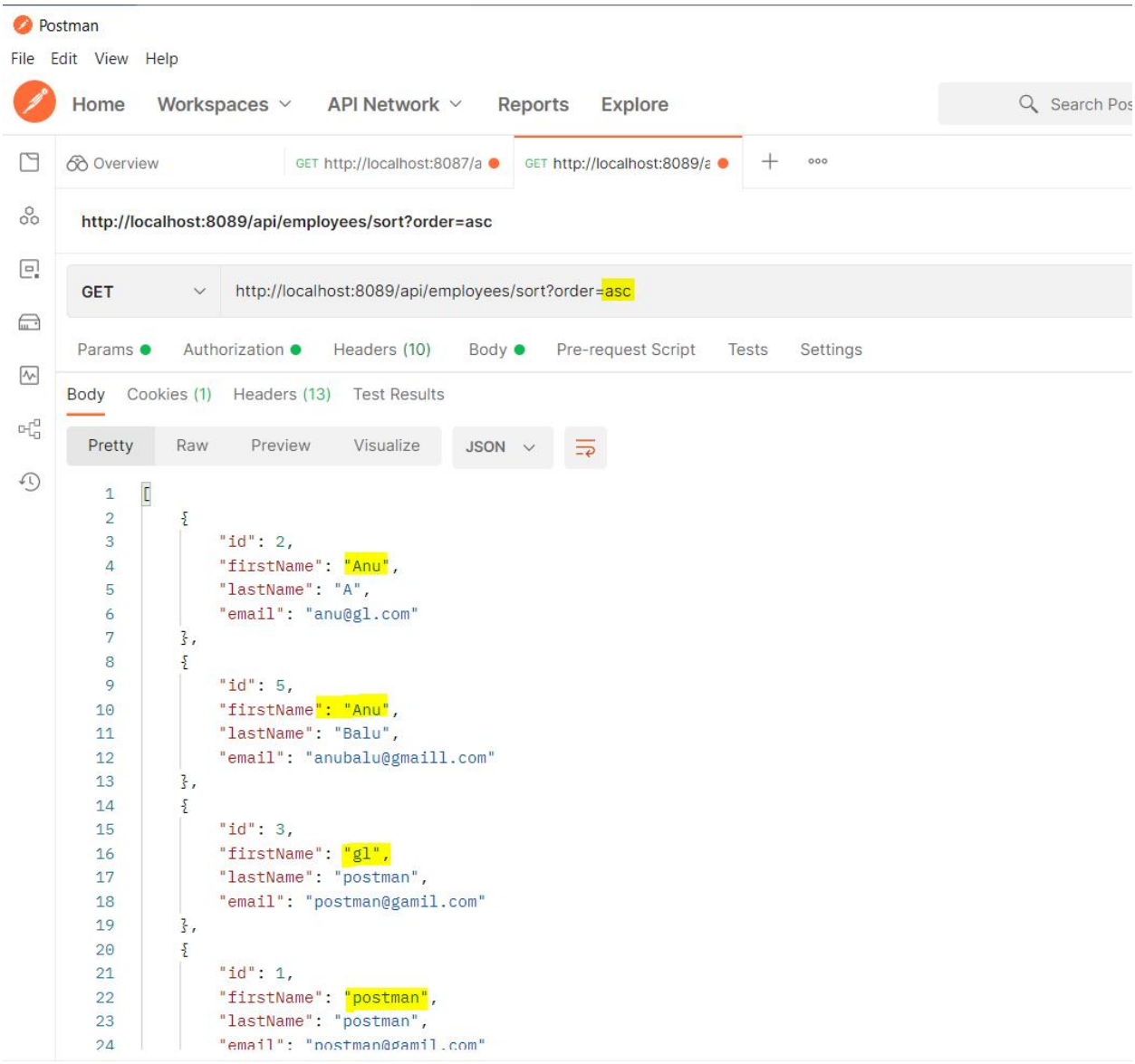

Postman
File Edit View Help
Home Workspaces API Network Reports Explore
Overview
GET http://localhost:8087/a
GET http://localhost:8089/a
http://localhost:8089/api/employees/search/anu
GET
http://localhost:8089/api/employees/search/anu
Params Authorization Headers (10) Body Pre-request Script Tests Settings
Body Cookies (1) Headers (13) Test Results
Pretty Raw Preview Visualize JSON
[
{
"id": 2,
"firstName": "Anu",
"lastName": "A",
"email": "anu@gl.com"
},
{
"id": 5,
"firstName": "Anu",
"lastName": "Balu",
"email": "anubalu@gmaill.com"
}
]



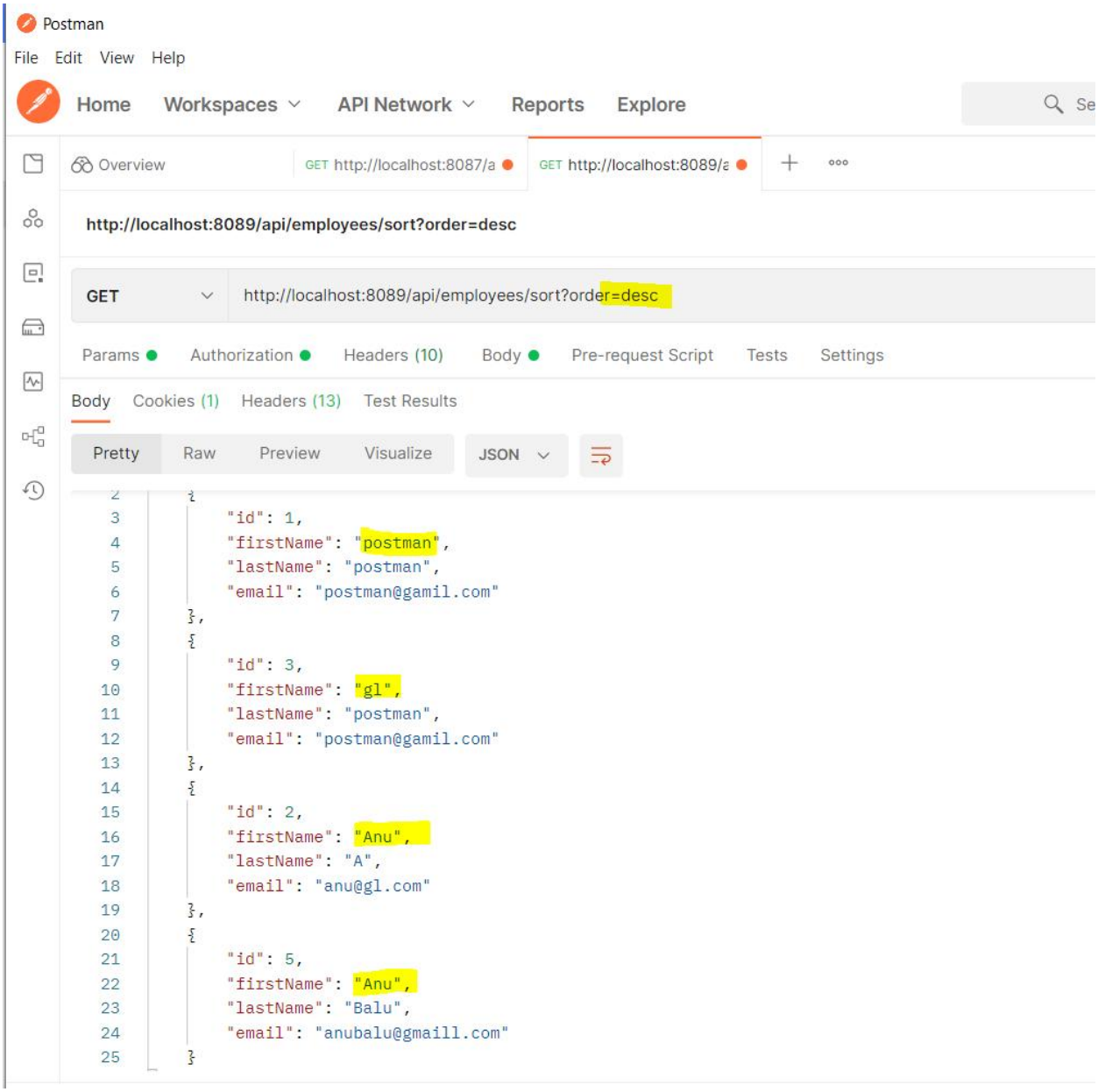

# 9. List all employee records sorted on their first name in either ascending order or descending order .

Ex-

Url- http://localhost:8080/api/employees/sort?order=”asc”




Url- http://localhost:8080/api/employees/sort?order=”desc”


Postman
File Edit View Help
Home Workspaces API Network Reports Explore
Overview
GET http://localhost:8087/a
GET http://localhost:8089/a
http://localhost:8089/api/employees/sort?order=desc
GET
http://localhost:8089/api/employees/sort?order=desc
Params Authorization Headers (10) Body Pre-request Script Tests Settings
Body Cookies (1) Headers (13) Test Results
Pretty Raw Preview Visualize JSON
{
"id": 1,
"firstName": "postman",
"lastName": "postman",
"email": "postman@gamil.com"
},
{
"id": 3,
"firstName": "gl",
"lastName": "postman",
"email": "postman@gamil.com"
},
{
"id": 2,
"firstName": "Anu",
"lastName": "A",
"email": "anu@gl.com"
},
{
"id": 5,
"firstName": "Anu",
"lastName": "Balu",
"email": "anubalu@gmaill.com"
}