Employee Management Rest Api

***Group 2A-S1** -> Sachin Datta, Samdarsh Gaur, Shivani Chippa*

Sohan Desai

Above project uses Spring Rest along with Spring Boot to perform basic CRUD operations on Employee Management System.

The model ***Employee*** has 3 visible as well as 1 hidden attribute.

They are ->

1. *Id* (hidden)
2. *Firstname*
3. *Lastname*
4. *Email*

The application performs following functions ->

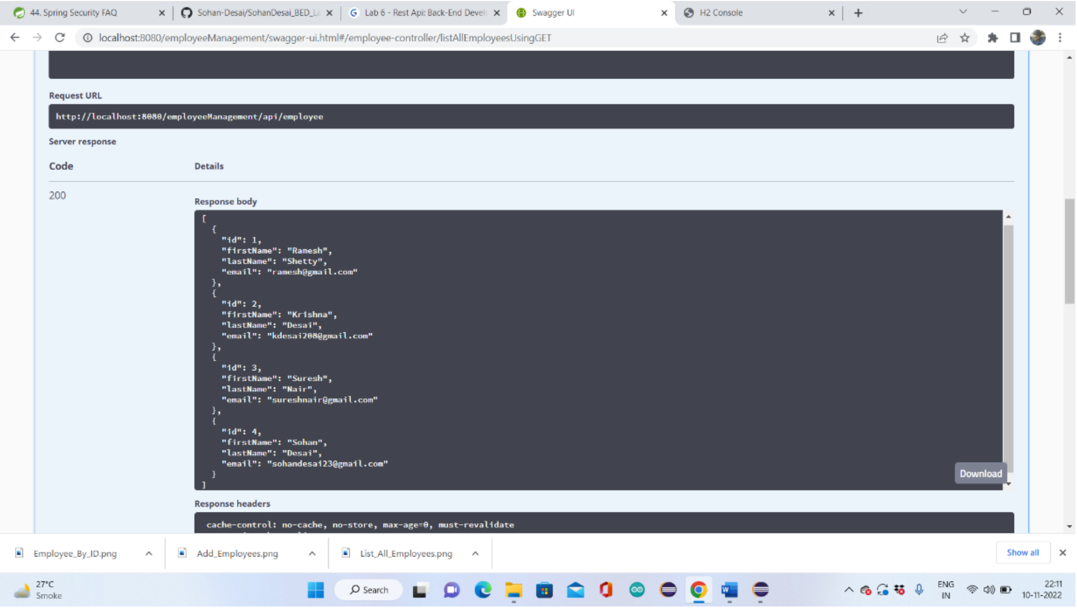
1. **Create** a new Student record
2. **Read** and display existing records
3. **Update** existing records and save any changes
4. **Delete** from existing records

The application also has security with roles -> USER and ADMIN

We shall further see each of these functionalities in greater detail

Logging in as USER

USER has password “secret”. USER can access all the GET mappings.



44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/listAllEmployeesUsingGET
Download
Employee_By_ID.png
Add_Employees.png
List_All_Employees.png
Show all
27°C
Smoke
Search
ENG
IN
22:11
10-11-2022

Request URL

```
http://localhost:8080/employeeManagement/api/employee
```

Server response

Code | Details

200

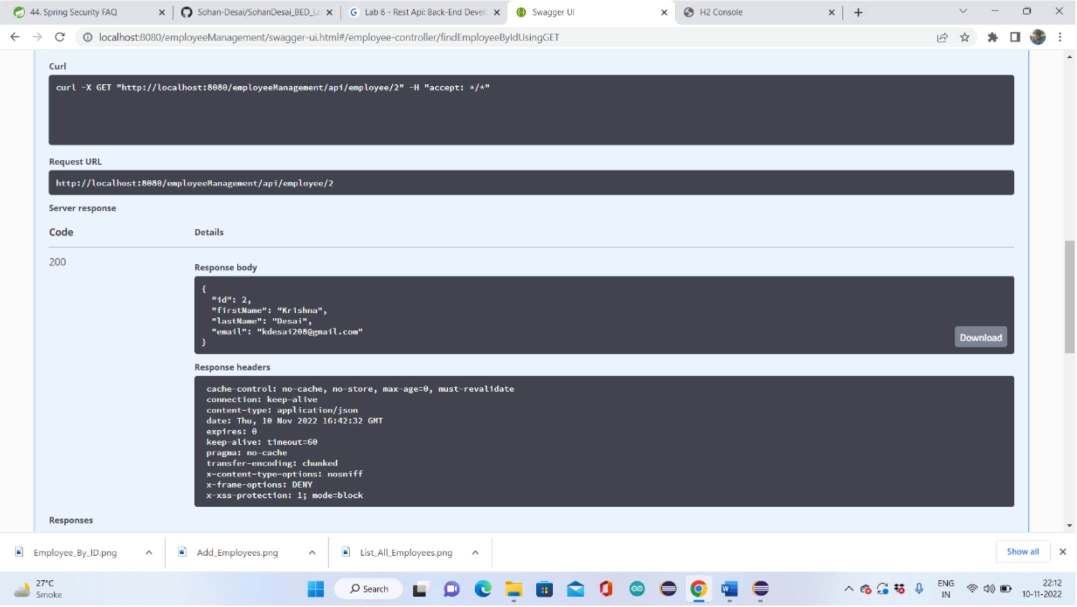
Response body

```
[
  {
    "id": 1,
    "firstName": "Ramesh",
    "lastName": "Shetty",
    "email": "ramesh@gmail.com"
  },
  {
    "id": 2,
    "firstName": "Krishna",
    "lastName": "Desai",
    "email": "kdesai208@gmail.com"
  },
  {
    "id": 3,
    "firstName": "Suresh",
    "lastName": "Nair",
    "email": "sureshnair@gmail.com"
  },
  {
    "id": 4,
    "firstName": "Sohan",
    "lastName": "Desai",
    "email": "sohandesai123@gmail.com"
  }
]
```

Response headers

```
cache-control: no-cache, no-store, max-age=0, must-revalidate
```

List all employees


44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/findEmployeeByIdUsingGET
Download
Employee_By_ID.png
Add_Employees.png
List_All_Employees.png
Show all
27°C
Smoke
Search
ENG
IN
22:12
10-11-2022

Curl

```
curl -X GET "http://localhost:8080/employeeManagement/api/employee/2" -H "accept: */*"
```

Request URL

```
http://localhost:8080/employeeManagement/api/employee/2
```

Server response

Code	Details
200	Response body

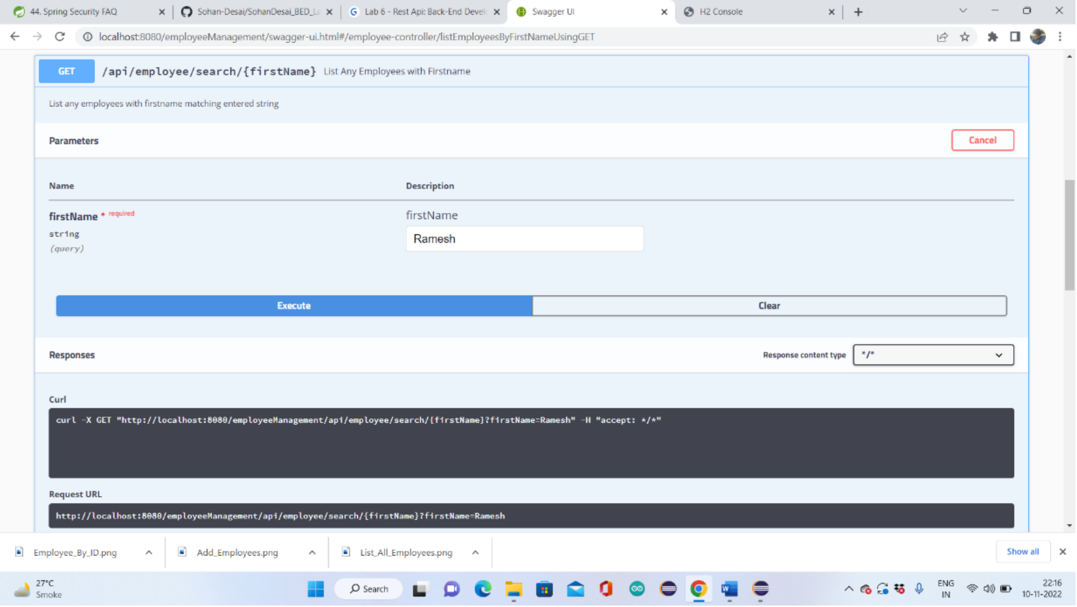
```
{
  "id": 2,
  "firstName": "Krishna",
  "lastName": "Desai",
  "email": "kdesai208@gmail.com"
}
```

Response headers

```
cache-control: no-cache, no-store, max-age=0, must-revalidate
connection: keep-alive
content-type: application/json
date: Thu, 10 Nov 2022 16:42:32 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 1; mode=block
```

Responses

Find employee by id



GET /api/employee/search/{firstName} List Any Employees with Firstname

List any employees with firstname matching entered string

Parameters

Name	Description
firstName * required string (query)	firstName Ramesh

Responses

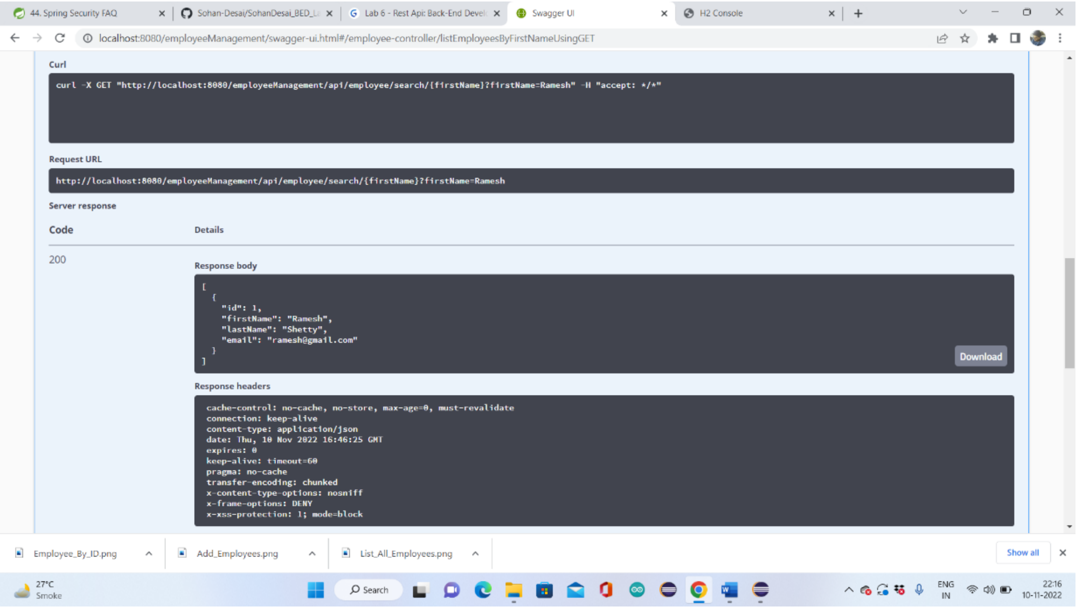
Response content type: */*

Curl

```
curl -X GET "http://localhost:8080/employeeManagement/api/employee/search/{firstName}?firstName=Ramesh" -H "accept: */*"
```

Request URL

```
http://localhost:8080/employeeManagement/api/employee/search/{firstName}?firstName=Ramesh
```


44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/listEmployeesByFirstNameUsingGET
Employee_By_ID.png
Add_Employees.png
List_All_Employees.png
Show all
27°C
Smoke
Search
ENG
IN
22:16
10-11-2022

Curl

```
curl -X GET "http://localhost:8080/employeeManagement/api/employee/search/{firstName}?firstName=Ramesh" -H "accept: */*"
```

Request URL

```
http://localhost:8080/employeeManagement/api/employee/search/{firstName}?firstName=Ramesh
```

Server response

Code Details

200

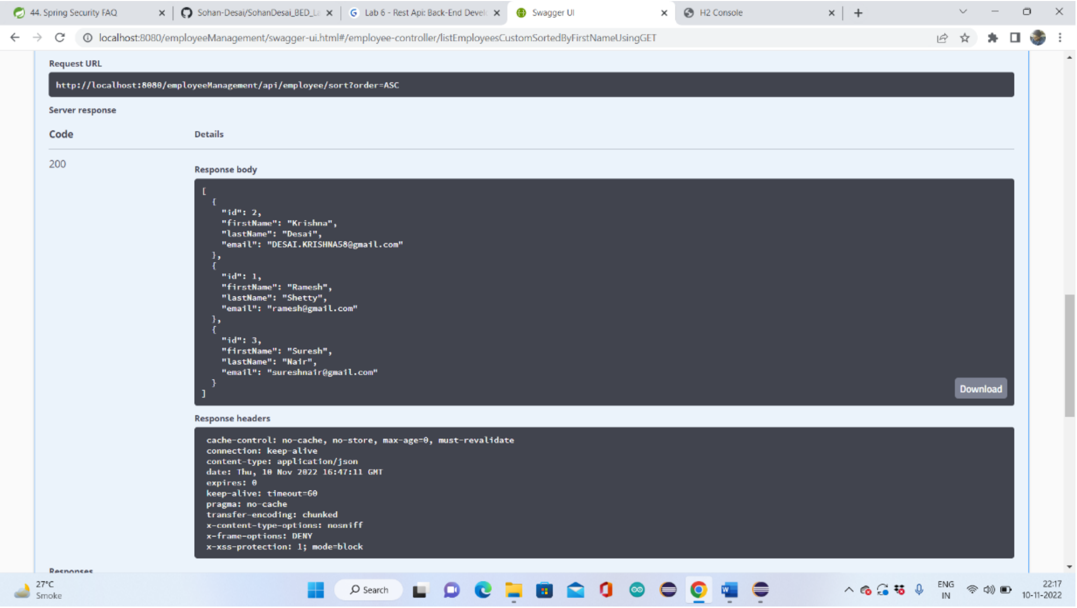
Response body

```
[
  {
    "id": 1,
    "firstName": "Ramesh",
    "lastName": "Shetty",
    "email": "ramesh@gmail.com"
  }
]
```

Response headers

```
 cache-control: no-cache, no-store, max-age=0, must-revalidate
 connection: keep-alive
 content-type: application/json
 date: Thu, 10 Nov 2022 16:46:25 GMT
 expires: 0
 keep-alive: timeout=60
 pragma: no-cache
 transfer-encoding: chunked
 x-content-type-options: nosniff
 x-frame-options: DENY
 x-xss-protection: 1; mode=block
```

Find Employee by firstname


44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/listEmployeesCustomSortedByFirstNameUsingGET
27°C
Smoke
Search
ENG
IN
22:17
10-11-2022

Request URL

```
http://localhost:8080/employeeManagement/api/employee/sort?order=ASC
```

Server response

Code Details

200

Response body

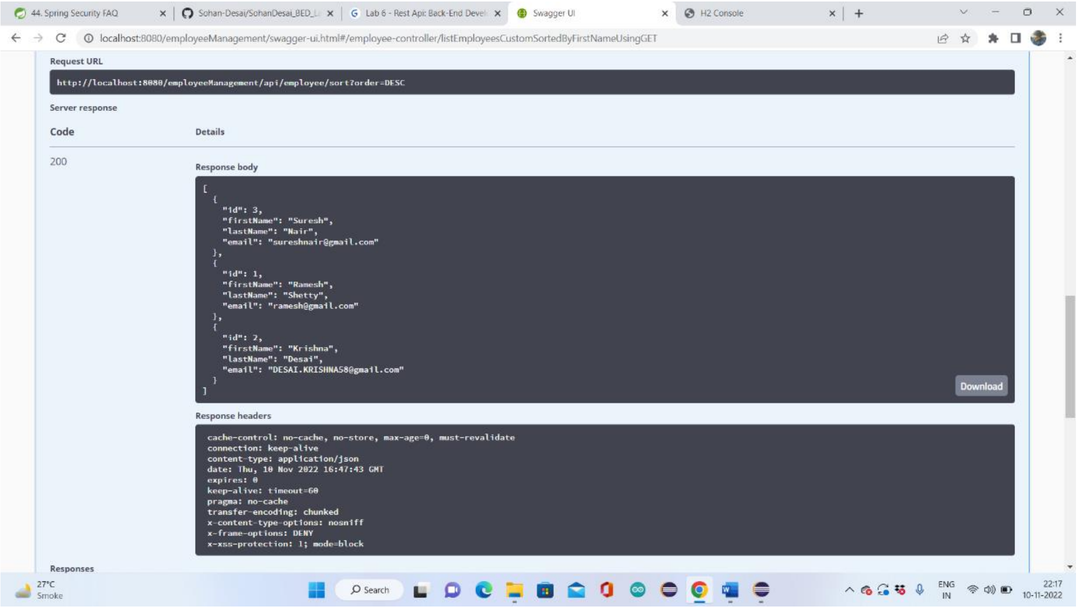
```
[
  {
    "id": 2,
    "firstName": "Krishna",
    "lastName": "Desai",
    "email": "DESAI.KRISHNA58@gmail.com"
  },
  {
    "id": 1,
    "firstName": "Ramesh",
    "lastName": "Shetty",
    "email": "ramesh@gmail.com"
  },
  {
    "id": 3,
    "firstName": "Suresh",
    "lastName": "Nair",
    "email": "sureshnair@gmail.com"
  }
]
```

Response headers

```
 cache-control: no-cache, no-store, max-age=0, must-revalidate
 connection: keep-alive
 content-type: application/json
 date: Thu, 10 Nov 2022 16:47:11 GMT
 expires: 0
 keep-alive: timeout=60
 pragma: no-cache
 transfer-encoding: chunked
 x-content-type-options: nosniff
 x-frame-options: DENY
 x-xss-protection: 1; mode=block
```

Responses

List all employees by firstname, order=ASC



44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/listEmployeesCustomSortedByFirstNameUsingGET

Request URL

```
http://localhost:8080/employeeManagement/api/employee/sort?order=DESC
```

Server response

Code | Details

200

Response body

```
[
  {
    "id": 3,
    "firstName": "Suresh",
    "lastName": "Nair",
    "email": "sureshnair@gmail.com"
  },
  {
    "id": 1,
    "firstName": "Ramesh",
    "lastName": "Shetty",
    "email": "ramesh@gmail.com"
  },
  {
    "id": 2,
    "firstName": "Krishna",
    "lastName": "Desai",
    "email": "DESAI.KRISHNA58@gmail.com"
  }
]
```

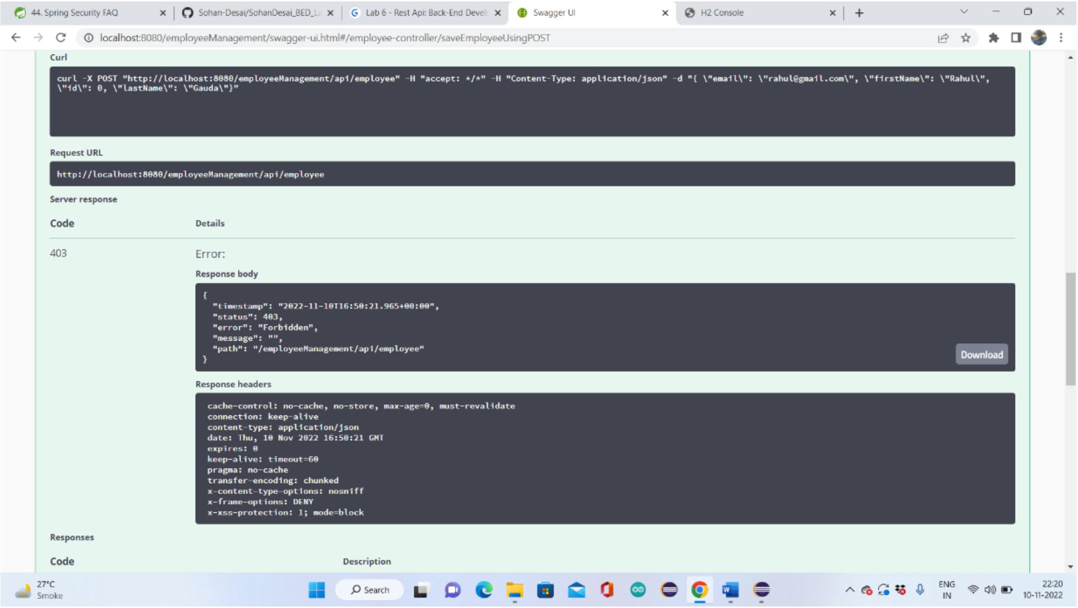
Response headers

```
cache-control: no-cache, no-store, max-age=0, must-revalidate
connection: keep-alive
content-type: application/json
date: Thu, 10 Nov 2022 16:47:43 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 1; mode=block
```

Responses

List all employees by firstname, order=DESC

However USER cannot save, update or delete employee records



44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/saveEmployeeUsingPOST

Curl

```
curl -X POST "http://localhost:8080/employeeManagement/api/employee" -H "accept: */*" -H "Content-Type: application/json" -d "{ \"email\": \"rahul@gmail.com\", \"firstName\": \"Rahul\",
\"id\": 0, \"lastName\": \"Gauda\"}"
```

Request URL

```
http://localhost:8080/employeeManagement/api/employee
```

Server response

Code | Details

403 Error:

Response body

```
{
  "timestamp": "2022-11-10T16:50:21.965+00:00",
  "status": 403,
  "error": "Forbidden",
  "message": "",
  "path": "/employeeManagement/api/employee"
}
```

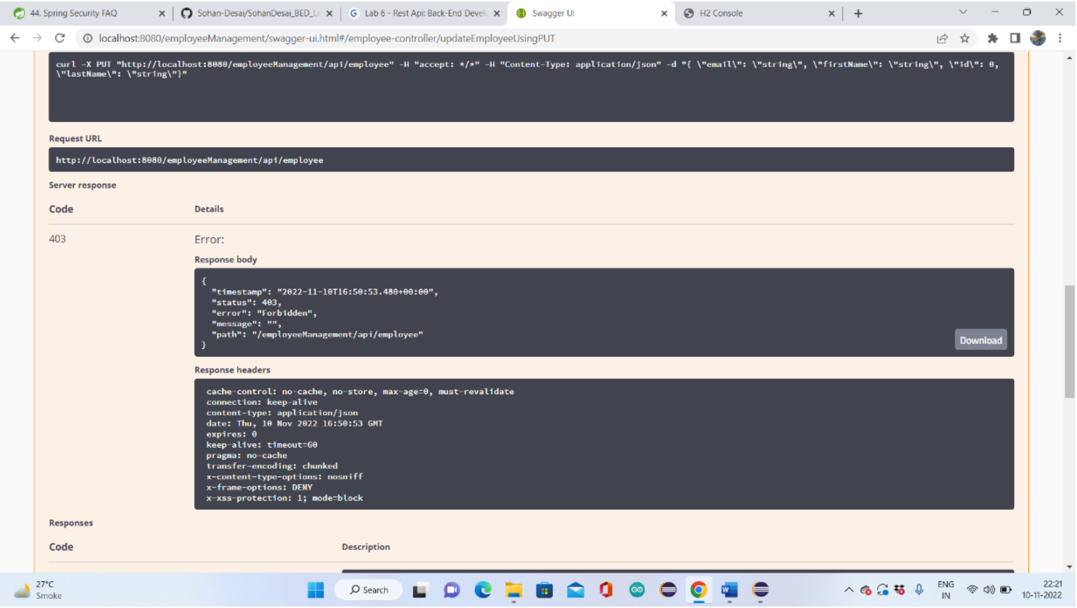
Response headers

```
cache-control: no-cache, no-store, max-age=0, must-revalidate
connection: keep-alive
content-type: application/json
date: Thu, 10 Nov 2022 16:50:21 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 1; mode=block
```

Responses

Code | Description

Access forbidden 403 for POST operation



44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/updateEmployeeUsingPUT

```
curl -X PUT "http://localhost:8080/employeeManagement/api/employee" -H "accept: */*" -H "Content-Type: application/json" -d "{ \"email\": \"string\", \"firstName\": \"string\", \"id\": 0, \"lastName\": \"string\"}"
```

Request URL

```
http://localhost:8080/employeeManagement/api/employee
```

Server response

Code | Details
403 | Error:

Response body

```
{
  "timestamp": "2022-11-10T16:50:53.480+00:00",
  "status": 403,
  "error": "Forbidden",
  "message": "",
  "path": "/employeeManagement/api/employee"
}
```

Response headers

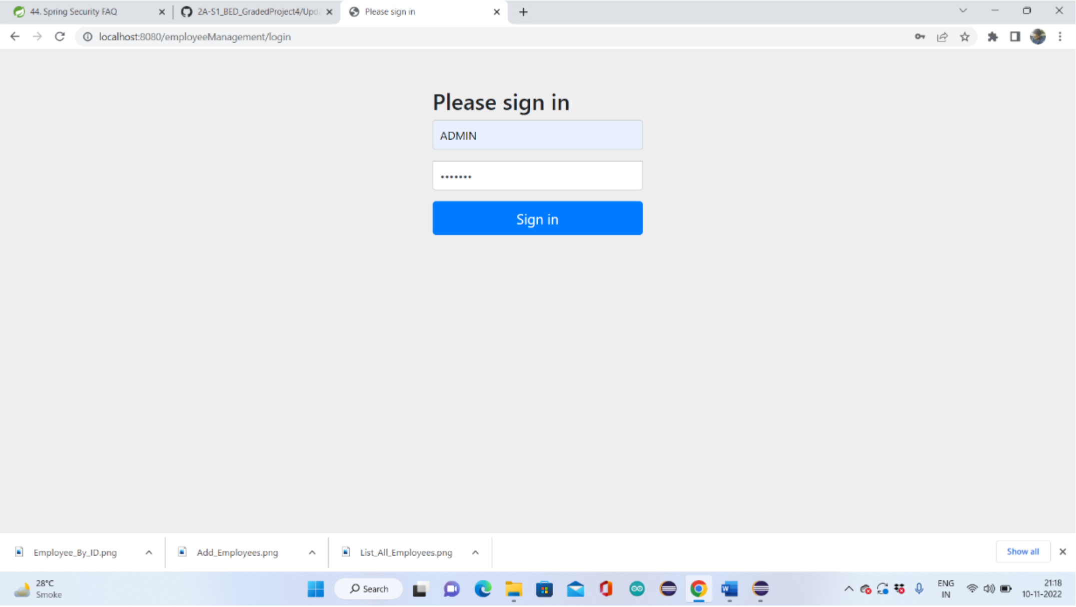
```
cache-control: no-cache, no-store, max-age=0, must-revalidate
connection: keep-alive
content-type: application/json
date: Thu, 10 Nov 2022 16:50:53 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 1; mode=block
```

Responses

Code | Description

Access forbidden 403 for DELETE operation

So let’s logout and log back in as ADMIN



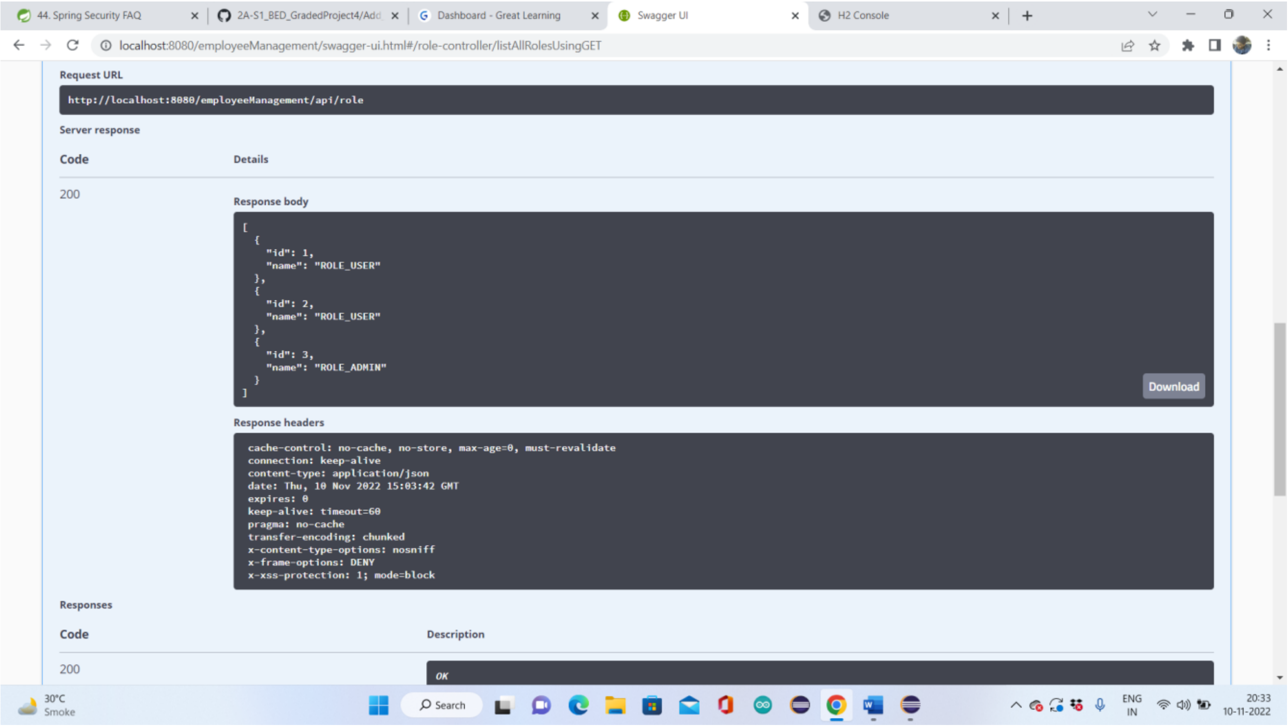
44. Spring Security FAQ
2A-S1_BED_GradedProject4/Upd
Please sign in
localhost:8080/employeeManagement/login

Please sign in

ADMIN

•••••••

ADMIN has password “success”. ADMIN can access ALL the mappings.


44. Spring Security FAQ
2A-S1_BED_GradedProject4/Add
Dashboard - Great Learning
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/role-controller/listAllRolesUsingGET
Download
30°C
Smoke
Search
ENG
IN
20:33
10-11-2022

Request URL

```
http://localhost:8080/employeeManagement/api/role
```

Server response

Code	Details
200	Response body

```
[
  {
    "id": 1,
    "name": "ROLE_USER"
  },
  {
    "id": 2,
    "name": "ROLE_USER"
  },
  {
    "id": 3,
    "name": "ROLE_ADMIN"
  }
]
```

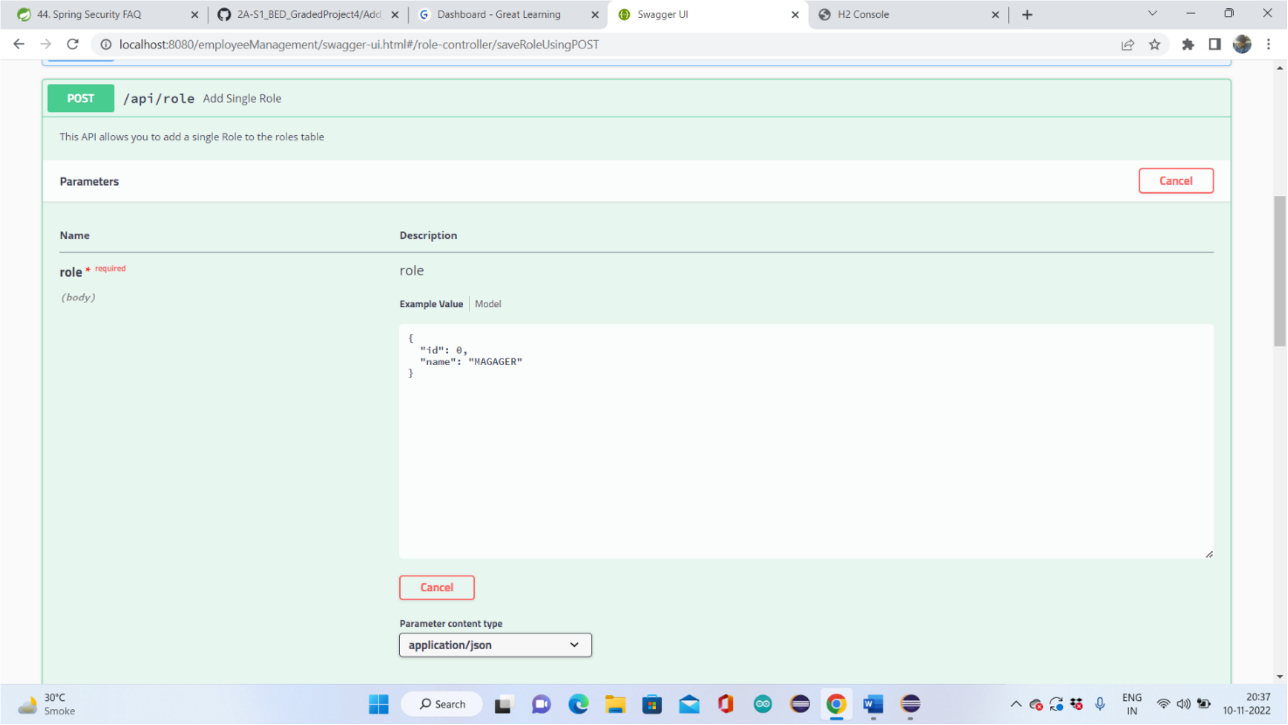
Response headers

```
cache-control: no-cache, no-store, max-age=0, must-revalidate
connection: keep-alive
content-type: application/json
date: Thu, 10 Nov 2022 15:03:42 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 1; mode=block
```

Responses

Code	Description
200	*OK*

List all roles


44. Spring Security FAQ
2A-S1_BED_GradedProject4/Add
Dashboard - Great Learning
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/role-controller/saveRoleUsingPOST
POST
Cancel
Cancel
30°C
Smoke
Search
ENG
IN
20:37
10-11-2022

POST `/api/role` Add Single Role

This API allows you to add a single Role to the roles table

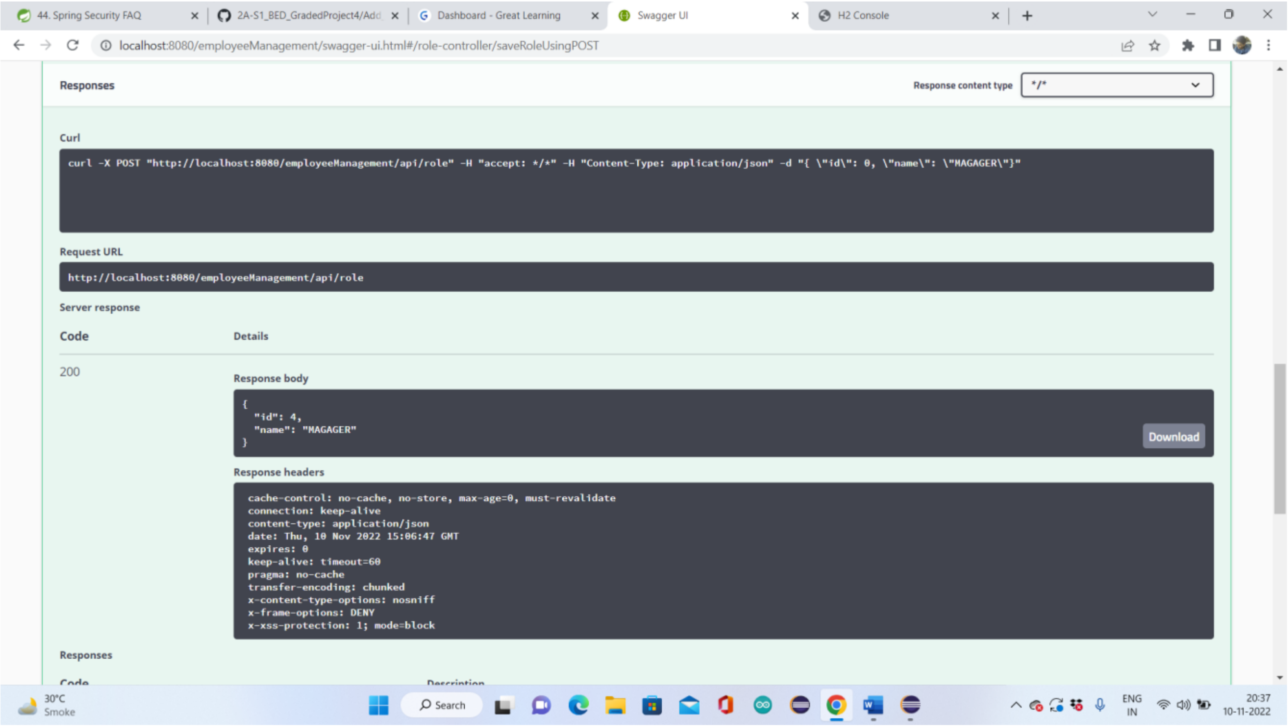
Parameters

Name	Description
role * required *(body)*	role Example Value | Model

```
{
  "id": 0,
  "name": "MAGAGER"
}
```

Parameter content type

application/json


44. Spring Security FAQ
2A-S1_BED_GradedProject4/Add
Dashboard - Great Learning
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/role-controller/saveRoleUsingPOST
Responses
Response content type
/

Curl

```
curl -X POST "http://localhost:8080/employeeManagement/api/role" -H "accept: */*" -H "Content-Type: application/json" -d "{ \"id\": 0, \"name\": \"MAGAGER\"}"
```

Request URL

```
http://localhost:8080/employeeManagement/api/role
```

Server response

Code: 200

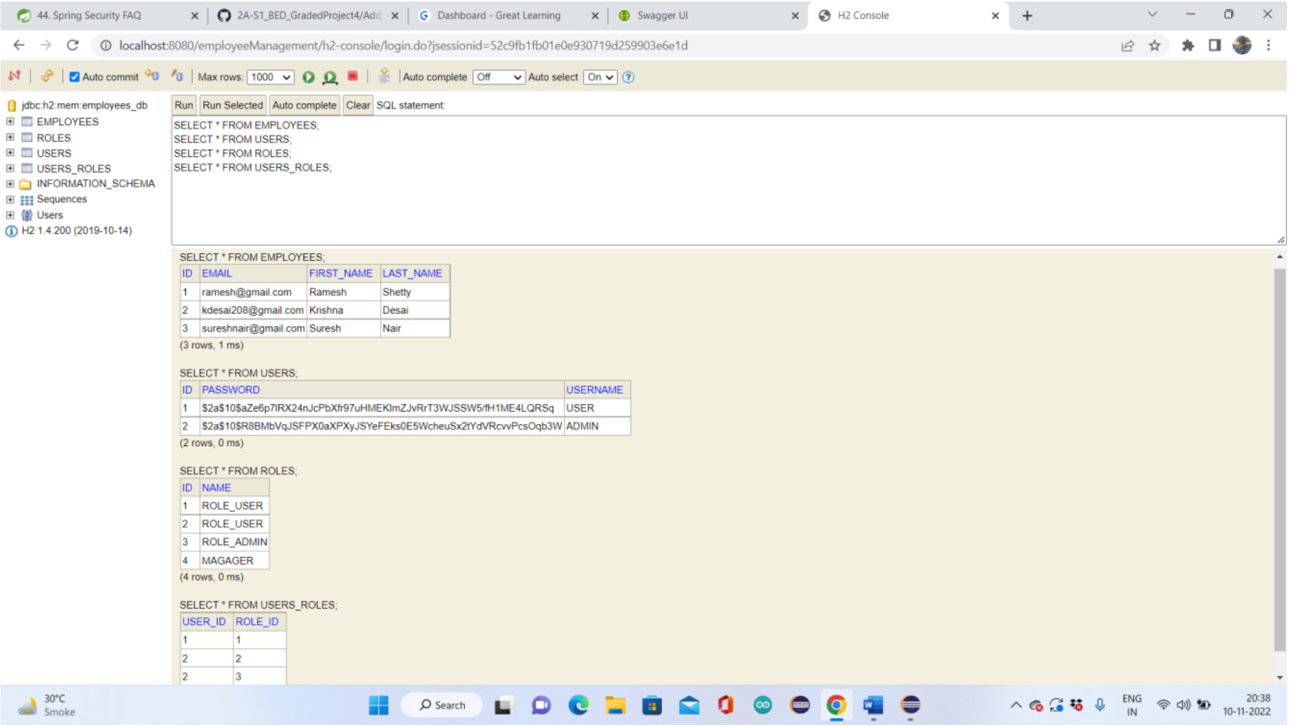
Response body

```
{
  "id": 4,
  "name": "MAGAGER"
}
```

Response headers

```
cache-control: no-cache, no-store, max-age=0, must-revalidate
connection: keep-alive
content-type: application/json
date: Thu, 10 Nov 2022 15:06:47 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 1; mode=block
```

Insert role "MANAGER"



```
SELECT * FROM EMPLOYEES;
SELECT * FROM USERS;
SELECT * FROM ROLES;
SELECT * FROM USERS_ROLES;
```

SELECT * FROM EMPLOYEES;

ID	EMAIL	FIRST_NAME	LAST_NAME
1	ramesh@gmail.com	Ramesh	Shetty
2	kdesai208@gmail.com	Krishna	Desai
3	sureshnair@gmail.com	Suresh	Nair

(3 rows, 1 ms)

SELECT * FROM USERS;

ID	PASSWORD	USERNAME
1	$2a$10$aZe6p7lRX24nJcPbXfr97uHMEKlmZJvRrT3WJSSW5/fH1ME4LQRSq	USER
2	$2a$10$R8BMbVqJSFPX0aXPXyJSYeFEks0E5WcheuSx2tYdVRcvvPcsOqb3W	ADMIN

(2 rows, 0 ms)

SELECT * FROM ROLES;

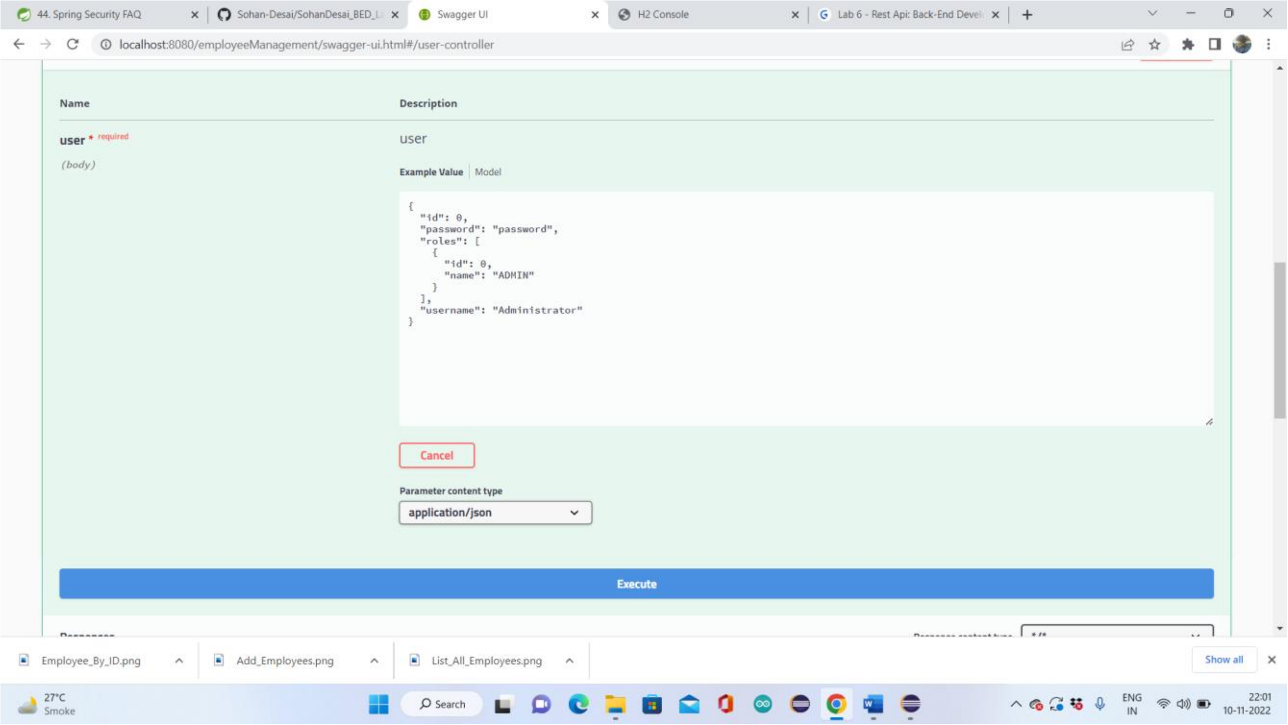
ID	NAME
1	ROLE_USER
2	ROLE_USER
3	ROLE_ADMIN
4	MAGAGER

(4 rows, 0 ms)

SELECT * FROM USERS_ROLES;

USER_ID	ROLE_ID
1	1
2	2
2	3

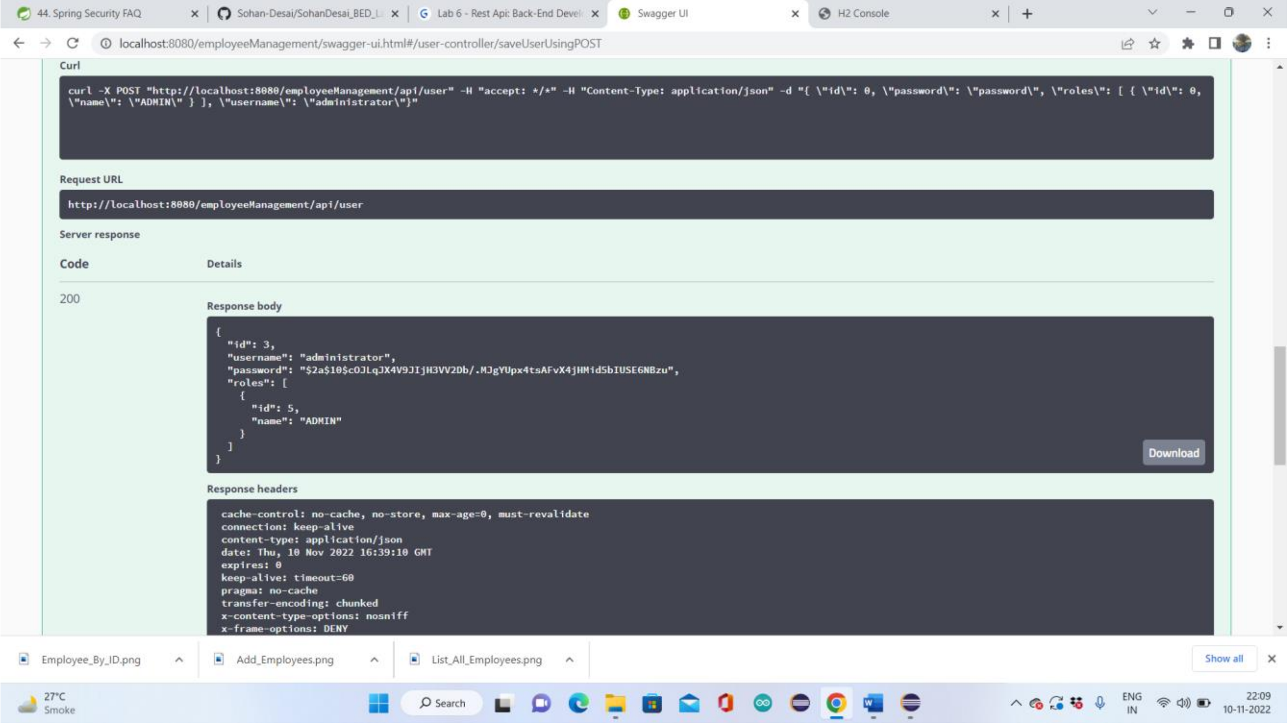
Role MANAGER inserted successfully


44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Swagger UI
H2 Console
Lab 6 - Rest Api: Back-End Devel
localhost:8080/employeeManagement/swagger-ui.html#/user-controller

Name	Description
user * required *(body)*	user

Example Value | Model

```
{
  "id": 0,
  "password": "password",
  "roles": [
    {
      "id": 0,
      "name": "ADMIN"
    }
  ],
  "username": "Administrator"
}
```


44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/user-controller/saveUserUsingPOST

Curl

```
curl -X POST "http://localhost:8080/employeeManagement/api/user" -H "accept: */*" -H "Content-Type: application/json" -d "{ \"id\": 0, \"password\": \"password\", \"roles\": [ { \"id\": 0,
\"name\": \"ADMIN\" } ], \"username\": \"administrator\"}"
```

Request URL

```
http://localhost:8080/employeeManagement/api/user
```

Server response

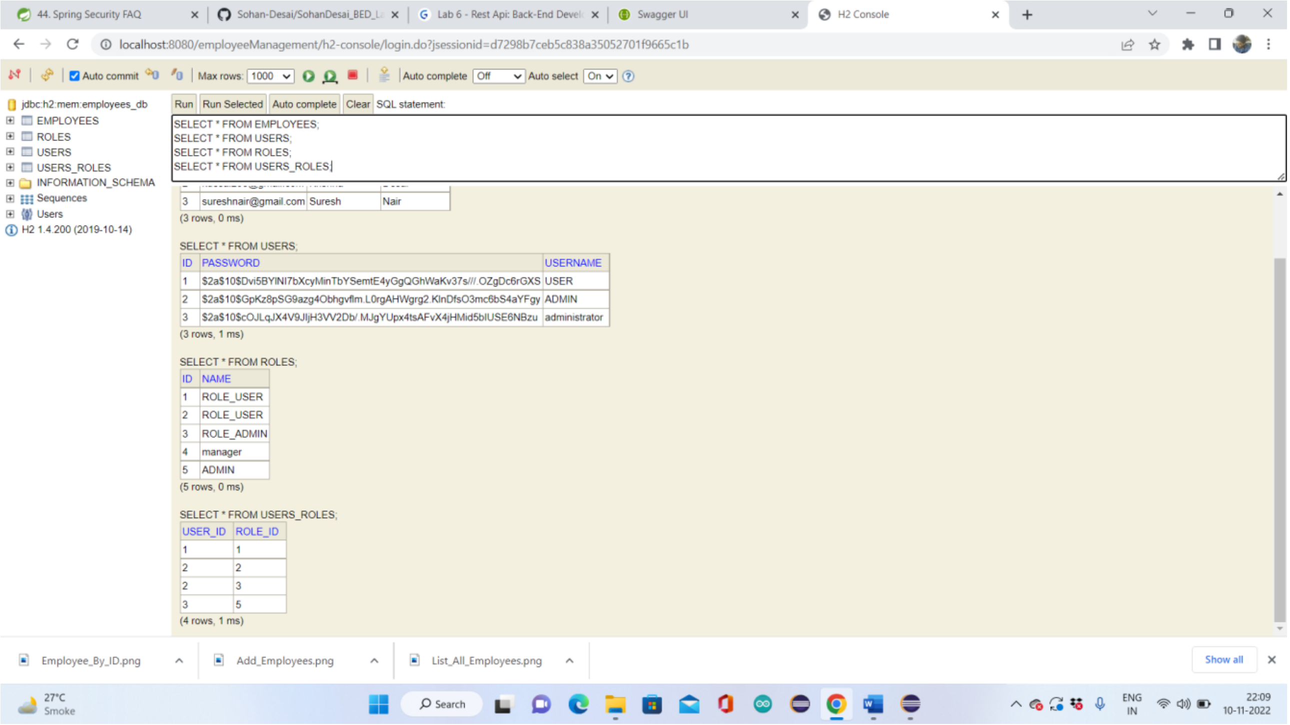
Code	Details
200	Response body

```
{
  "id": 3,
  "username": "administrator",
  "password": "$2a$10$cOJLqJX4V9JIjH3VV2Db/.MJgYUpx4tsAFvX4jHMid5bIUSE6NBzu",
  "roles": [
    {
      "id": 5,
      "name": "ADMIN"
    }
  ]
}
```

Response headers

```
cache-control: no-cache, no-store, max-age=0, must-revalidate
connection: keep-alive
content-type: application/json
date: Thu, 10 Nov 2022 16:39:10 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
```

Insert user “administrator” with password as “password”



localhost:8080/employeeManagement/h2-console/login.do?jsessionid=d7298b7ceb5c838a35052701f9665c1b

jdbc:h2:mem:employees_db
- EMPLOYEES
- ROLES
- USERS
- USERS_ROLES
- INFORMATION_SCHEMA
- Sequences
- Users
- H2 1.4.200 (2019-10-14)

```
SELECT * FROM EMPLOYEES;
SELECT * FROM USERS;
SELECT * FROM ROLES;
SELECT * FROM USERS_ROLES;
```

3	sureshnair@gmail.com	Suresh	Nair

(3 rows, 0 ms)

SELECT * FROM USERS;

ID	PASSWORD	USERNAME
1	$2a$10$Dvi5BYINI7bXcyMinTbYSemtE4yGgQGhWaKv37s///.OZgDc6rGXS	USER
2	$2a$10$GpKz8pSG9azg4Obhgvflm.L0rgAHWgrg2.KlnDfsO3mc6bS4aYFgy	ADMIN
3	$2a$10$cOJLqJX4V9JIjH3VV2Db/.MJgYUpx4tsAFvX4jHMid5bIUSE6NBzu	administrator

(3 rows, 1 ms)

SELECT * FROM ROLES;

ID	NAME
1	ROLE_USER
2	ROLE_USER
3	ROLE_ADMIN
4	manager
5	ADMIN

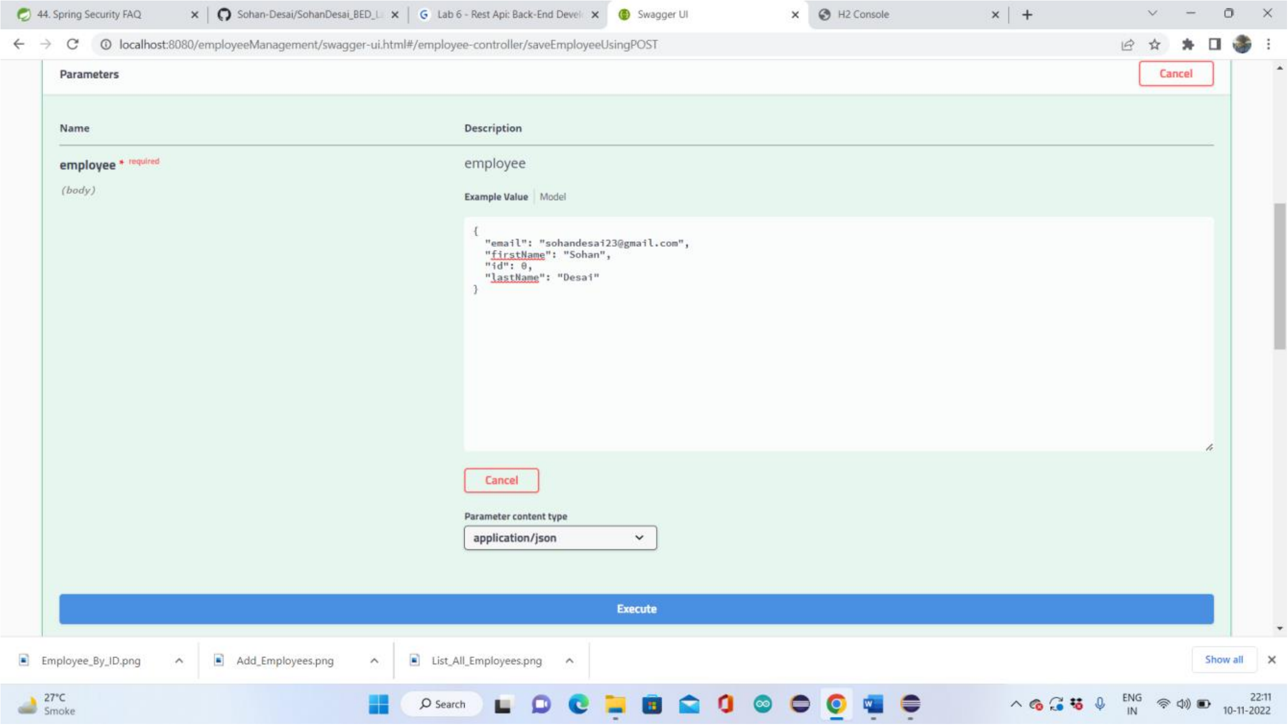
(5 rows, 0 ms)

SELECT * FROM USERS_ROLES;

USER_ID	ROLE_ID
1	1
2	2
2	3
3	5

(4 rows, 1 ms)

User administrator inserted successfully



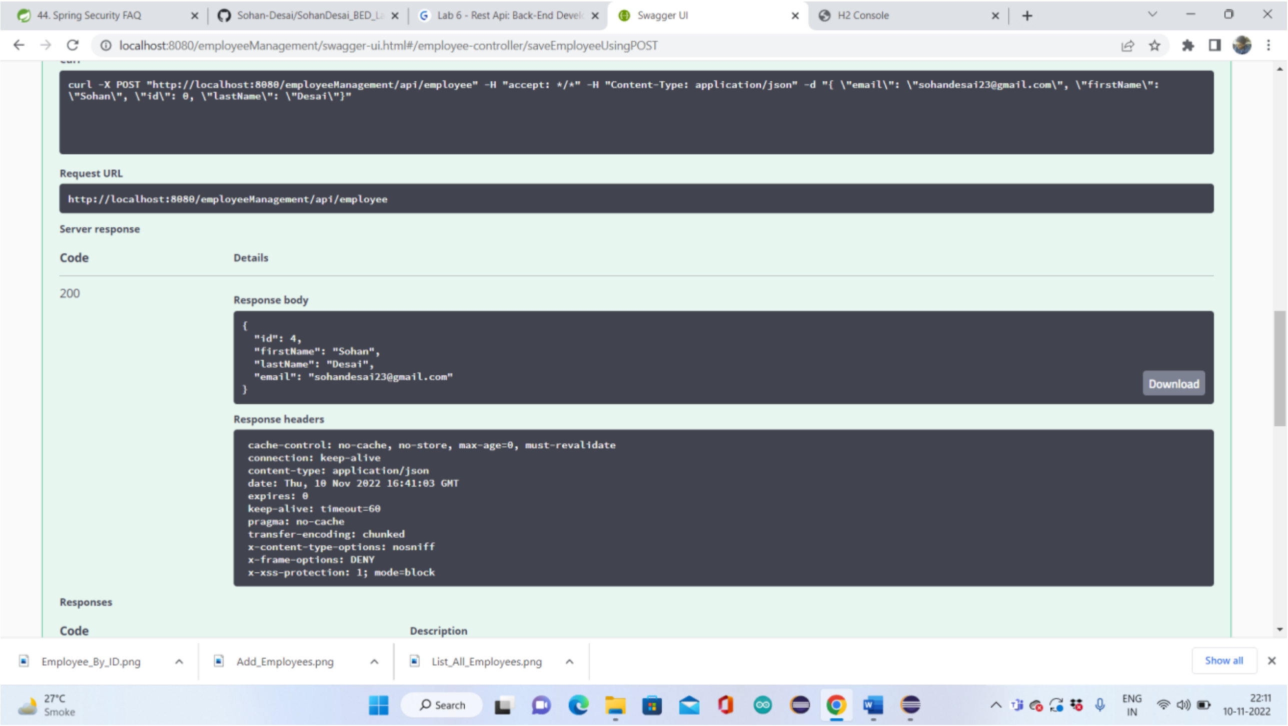
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/saveEmployeeUsingPOST

Parameters

Name	Description
employee * required (body)	employee

```
{
  "email": "sohandesai23@gmail.com",
  "firstName": "Sohan",
  "id": 0,
  "lastName": "Desai"
}
```

Parameter content type: application/json



localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/saveEmployeeUsingPOST

```
curl -X POST "http://localhost:8080/employeeManagement/api/employee" -H "accept: */*" -H "Content-Type: application/json" -d "{ \"email\": \"sohandesai23@gmail.com\", \"firstName\":
\"Sohan\", \"id\": 0, \"lastName\": \"Desai\"}"
```

Request URL

```
http://localhost:8080/employeeManagement/api/employee
```

Server response

Code: 200

Response body

```
{
  "id": 4,
  "firstName": "Sohan",
  "lastName": "Desai",
  "email": "sohandesai23@gmail.com"
}
```

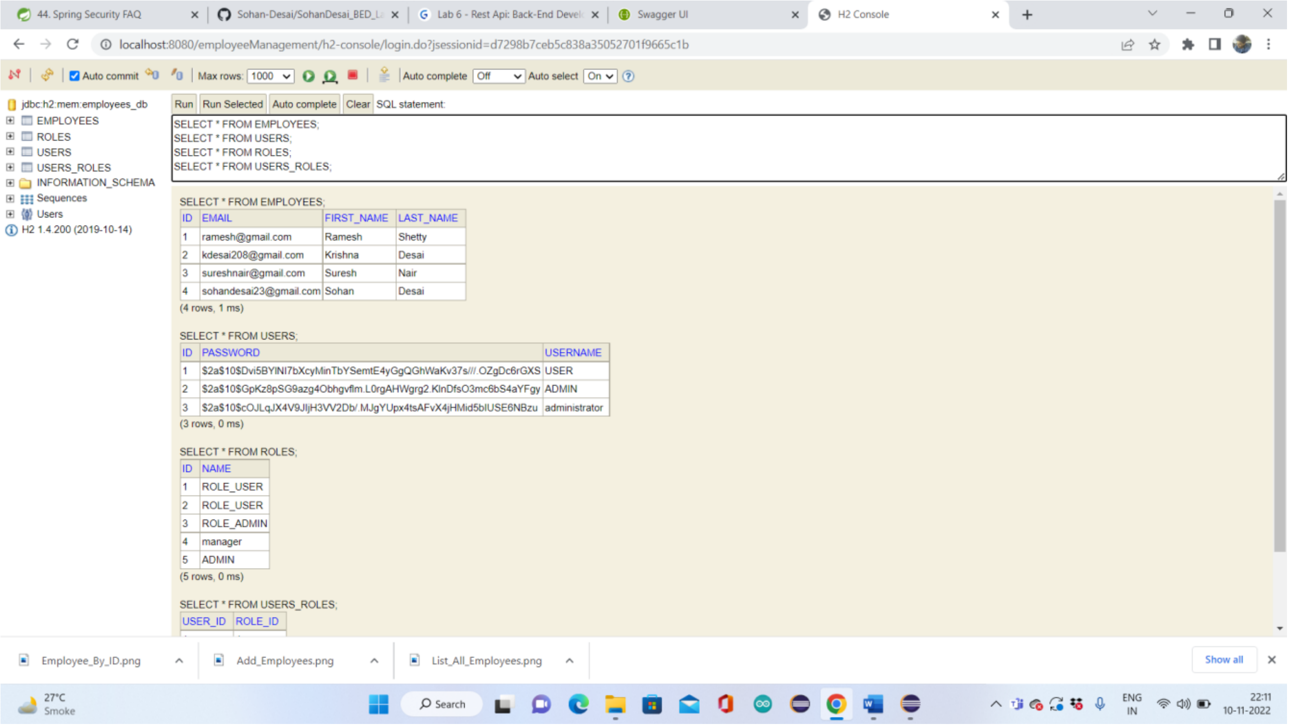
Response headers

```
cache-control: no-cache, no-store, max-age=0, must-revalidate
connection: keep-alive
content-type: application/json
date: Thu, 10 Nov 2022 16:41:03 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 1; mode=block
```

Responses

Code Description

Insert Employee



localhost:8080/employeeManagement/h2-console/login.do?jsessionid=d7298b7ceb5c838a35052701f9665c1b

```
SELECT * FROM EMPLOYEES;
SELECT * FROM USERS;
SELECT * FROM ROLES;
SELECT * FROM USERS_ROLES;
```

SELECT * FROM EMPLOYEES;

ID	EMAIL	FIRST_NAME	LAST_NAME
1	ramesh@gmail.com	Ramesh	Shetty
2	kdesai208@gmail.com	Krishna	Desai
3	sureshnair@gmail.com	Suresh	Nair
4	sohandesai23@gmail.com	Sohan	Desai

(4 rows, 1 ms)

SELECT * FROM USERS;

ID	PASSWORD	USERNAME
1	$2a$10$Dvi5BYINI7bXcyMinTbYSemtE4yGgQGhWaKv37s///.OZgDc6rGXS	USER
2	$2a$10$GpKz8pSG9azg4Obhgvflm.L0rgAHWgrg2.KlnDfsO3mc6bS4aYFgy	ADMIN
3	$2a$10$cOJLqJX4V9JljH3VV2Db/.MJgYUpx4tsAFvX4jHMid5bIUSE6NBzu	administrator

(3 rows, 0 ms)

SELECT * FROM ROLES;

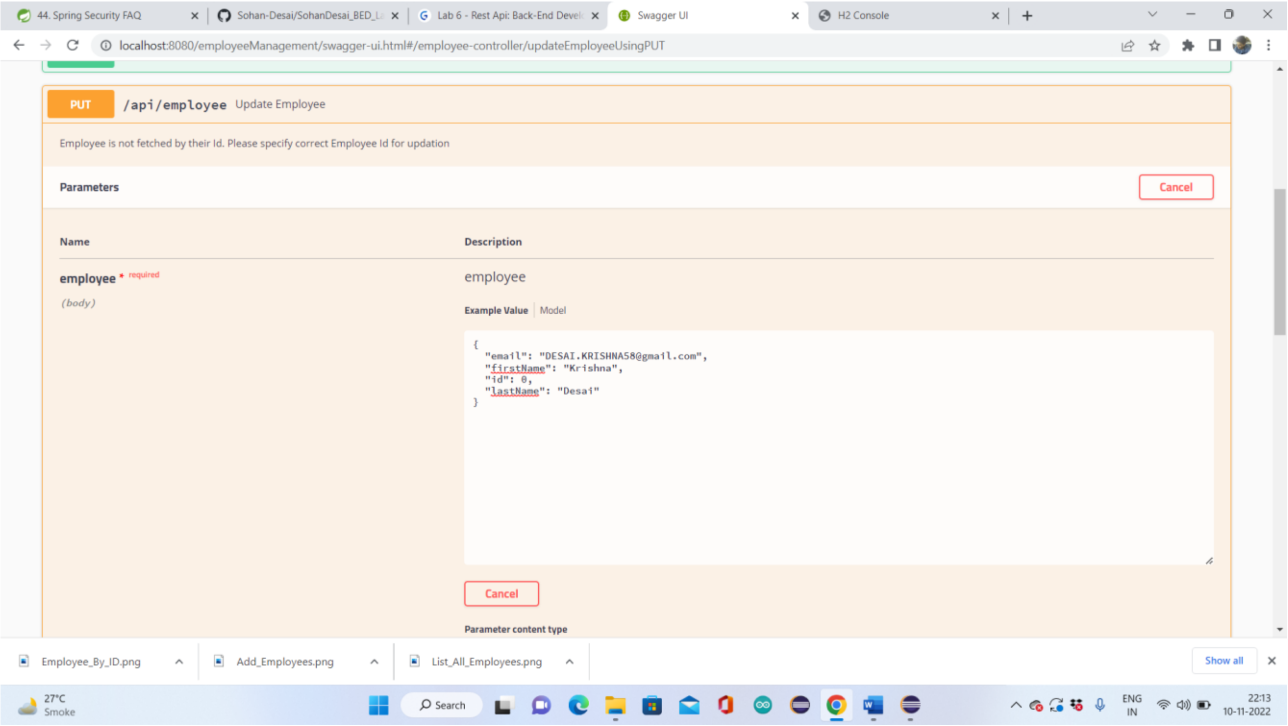
ID	NAME
1	ROLE_USER
2	ROLE_USER
3	ROLE_ADMIN
4	manager
5	ADMIN

(5 rows, 0 ms)

SELECT * FROM USERS_ROLES;

USER_ID	ROLE_ID

Employee inserted successfully


44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/updateEmployeeUsingPUT

PUT /api/employee Update Employee

Employee is not fetched by their Id. Please specify correct Employee id for updation

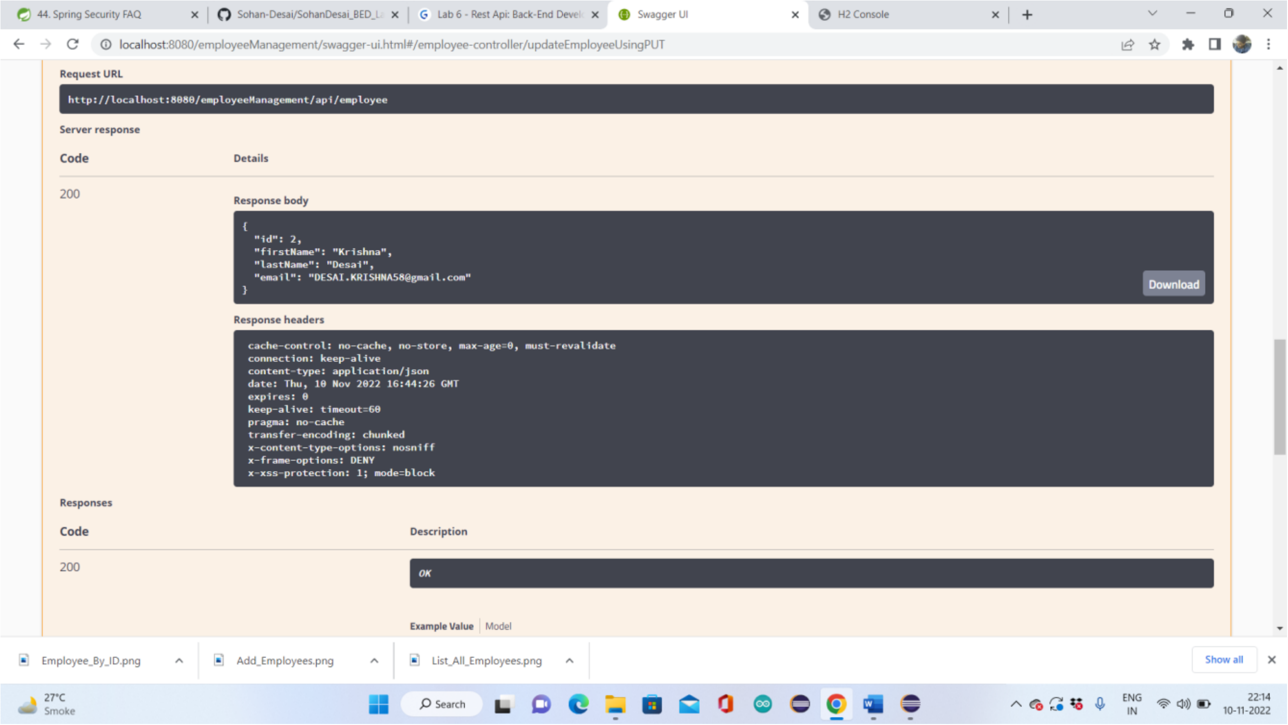
Parameters

Name	Description
employee * required (body)	employee

Example Value | Model

```
{
  "email": "DESAI.KRISHNA58@gmail.com",
  "firstName": "Krishna",
  "id": 0,
  "lastName": "Desai"
}
```

Parameter content type


44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/updateEmployeeUsingPUT

Request URL

```
http://localhost:8080/employeeManagement/api/employee
```

Server response

Code | Details

200

Response body

```
{
  "id": 2,
  "firstName": "Krishna",
  "lastName": "Desai",
  "email": "DESAI.KRISHNA58@gmail.com"
}
```

Response headers

```
 cache-control: no-cache, no-store, max-age=0, must-revalidate
 connection: keep-alive
 content-type: application/json
 date: Thu, 10 Nov 2022 16:44:26 GMT
 expires: 0
 keep-alive: timeout=60
 pragma: no-cache
 transfer-encoding: chunked
 x-content-type-options: nosniff
 x-frame-options: DENY
 x-xss-protection: 1; mode=block
```

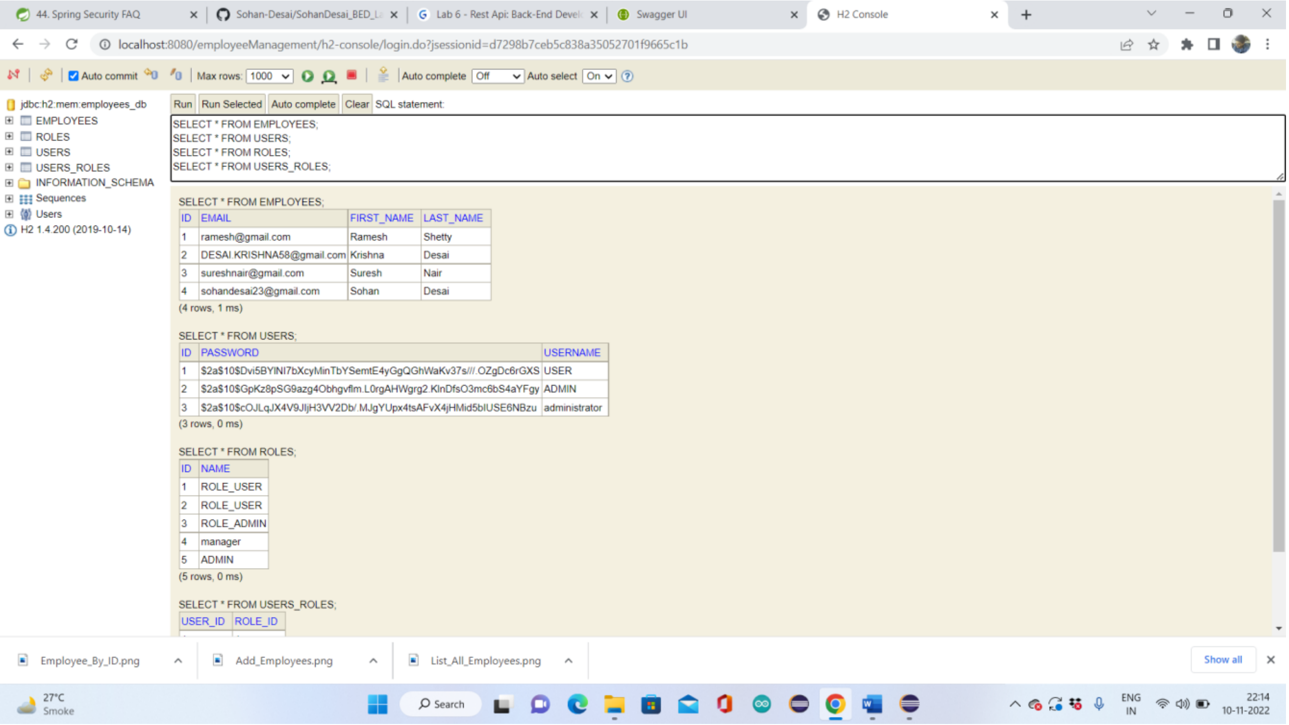
Responses

Code	Description
200	OK

Example Value | Model

Update Employee details

#Note-> As per problem statement, records are not updated on basis of id. User needs to specify correct employee id.



localhost:8080/employeeManagement/h2-console/login.do?jsessionid=d7298b7ceb5c838a35052701f9665c1b

☑ Auto commit | Max rows: 1000 | Auto complete: Off | Auto select: On

jdbc:h2:mem:employees_db

SQL statement:

```
SELECT * FROM EMPLOYEES;
SELECT * FROM USERS;
SELECT * FROM ROLES;
SELECT * FROM USERS_ROLES;
```

SELECT * FROM EMPLOYEES;

ID	EMAIL	FIRST_NAME	LAST_NAME
1	ramesh@gmail.com	Ramesh	Shetty
2	DESAI.KRISHNA58@gmail.com	Krishna	Desai
3	sureshnair@gmail.com	Suresh	Nair
4	sohandesai23@gmail.com	Sohan	Desai

(4 rows, 1 ms)

SELECT * FROM USERS;

ID	PASSWORD	USERNAME
1	$2a$10$Dvi5BYINI7bXcyMinTbYSemtE4yGgQGhWaKv37s///.OZgDc6rGXS	USER
2	$2a$10$GpKz8pSG9azg4Obhgvflm.L0rgAHWgrg2.KlnDfsO3mc6bS4aYFgy	ADMIN
3	$2a$10$cOJLqJX4V9JljH3VV2Db/.MJgYUpx4tsAFvX4jHMid5bIUSE6NBzu	administrator

(3 rows, 0 ms)

SELECT * FROM ROLES;

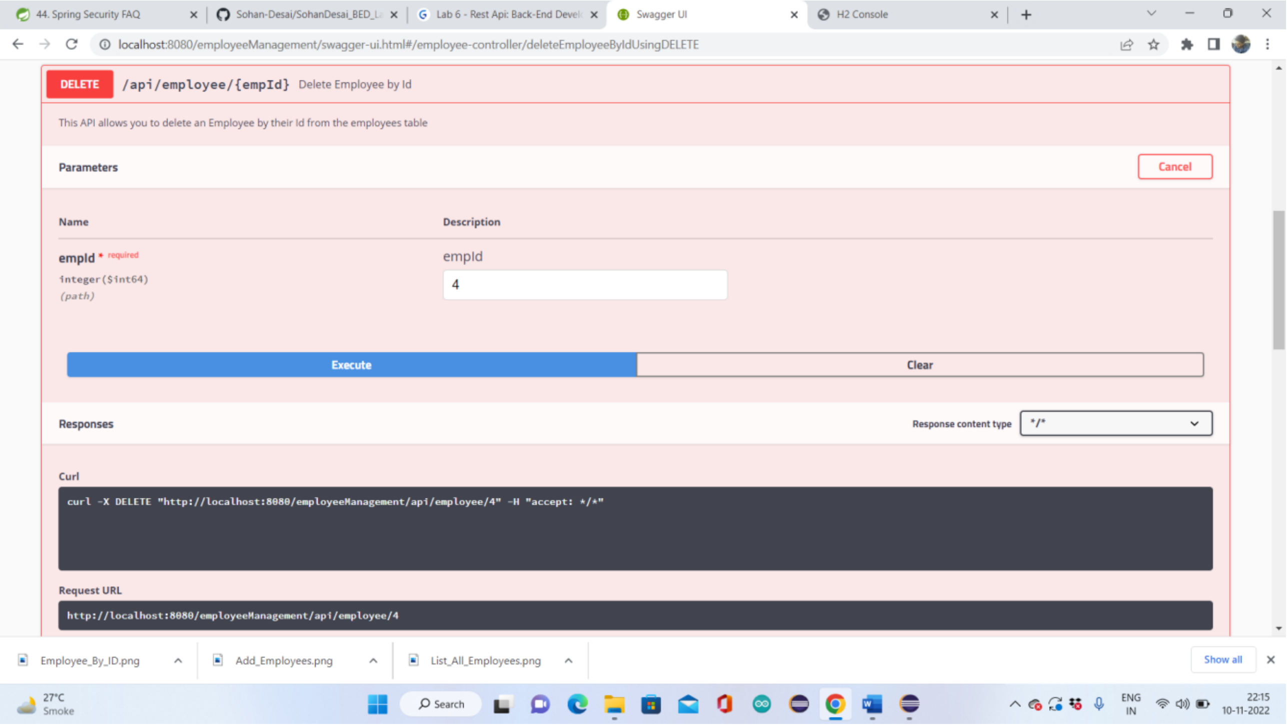
ID	NAME
1	ROLE_USER
2	ROLE_USER
3	ROLE_ADMIN
4	manager
5	ADMIN

(5 rows, 0 ms)

SELECT * FROM USERS_ROLES;

USER_ID	ROLE_ID

Employee details updated



localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/deleteEmployeeByIdUsingDELETE

DELETE /api/employee/{empId} Delete Employee by Id

This API allows you to delete an Employee by their Id from the employees table

Parameters

Name	Description
empId * required integer($int64) (path)	empId 4

Responses

Response content type: */*

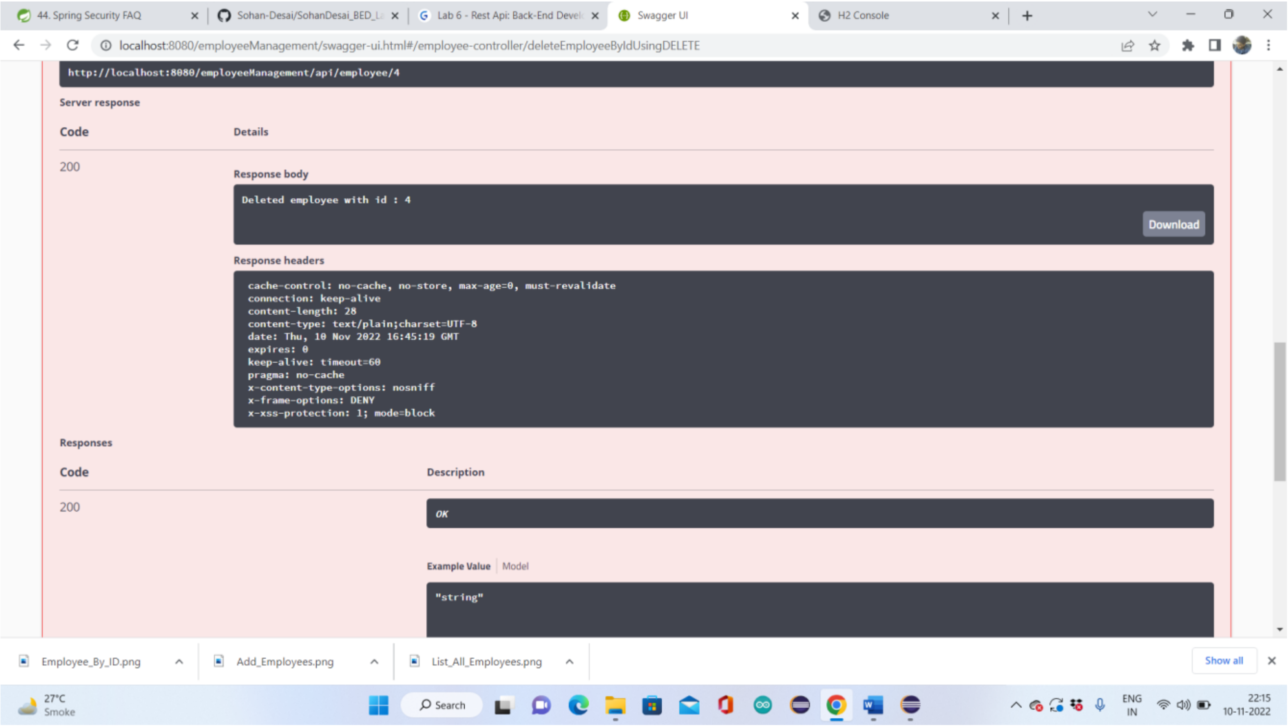
Curl

```
curl -X DELETE "http://localhost:8080/employeeManagement/api/employee/4" -H "accept: */*"
```

Request URL

```
http://localhost:8080/employeeManagement/api/employee/4
```

Delete employee by id



44. Spring Security FAQ
Sohan-Desai/SohanDesai_BED_L
Lab 6 - Rest Api: Back-End Devel
Swagger UI
H2 Console
localhost:8080/employeeManagement/swagger-ui.html#/employee-controller/deleteEmployeeByIdUsingDELETE

```
http://localhost:8080/employeeManagement/api/employee/4
```

Server response

Code	Details
200	Response body

```
Deleted employee with id : 4
```

Response headers

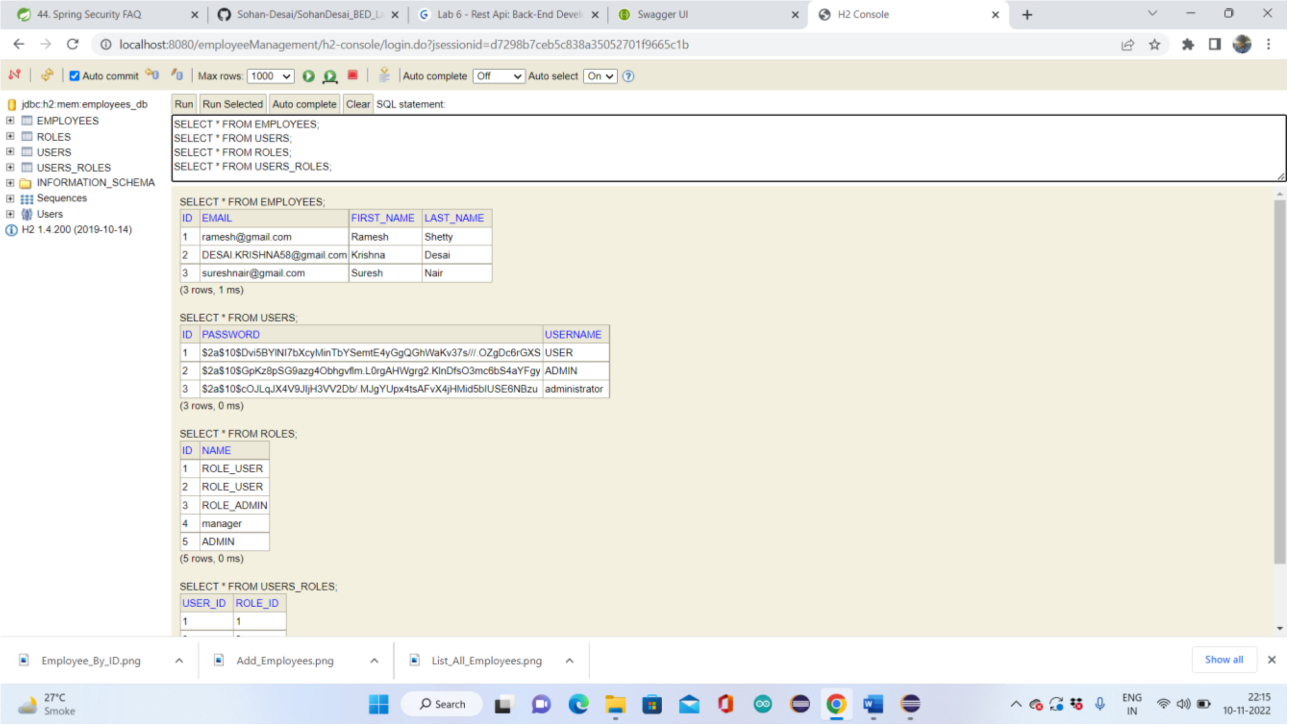
```
cache-control: no-cache, no-store, max-age=0, must-revalidate
connection: keep-alive
content-length: 28
content-type: text/plain;charset=UTF-8
date: Thu, 10 Nov 2022 16:45:19 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 1; mode=block
```

Responses

Code	Description
200	OK

Example Value | Model

```
"string"
```



```
SELECT * FROM EMPLOYEES;
SELECT * FROM USERS;
SELECT * FROM ROLES;
SELECT * FROM USERS_ROLES;
```

SELECT * FROM EMPLOYEES;

ID	EMAIL	FIRST_NAME	LAST_NAME
1	ramesh@gmail.com	Ramesh	Shetty
2	DESAI.KRISHNA58@gmail.com	Krishna	Desai
3	sureshnair@gmail.com	Suresh	Nair

(3 rows, 1 ms)

SELECT * FROM USERS;

ID	PASSWORD	USERNAME
1	$2a$10$Dvi5BYINI7bXcyMinTbYSemtE4yGgQGhWaKv37s///.OZgDc6rGXS	USER
2	$2a$10$GpKz8pSG9azg4Obhgvflm.L0rgAHWgrg2.KlnDfsO3mc6bS4aYFgy	ADMIN
3	$2a$10$cOJLqJX4V9JljH3VV2Db/.MJgYUpx4tsAFvX4jHMid5bIUSE6NBzu	administrator

(3 rows, 0 ms)

SELECT * FROM ROLES;

ID	NAME
1	ROLE_USER
2	ROLE_USER
3	ROLE_ADMIN
4	manager
5	ADMIN

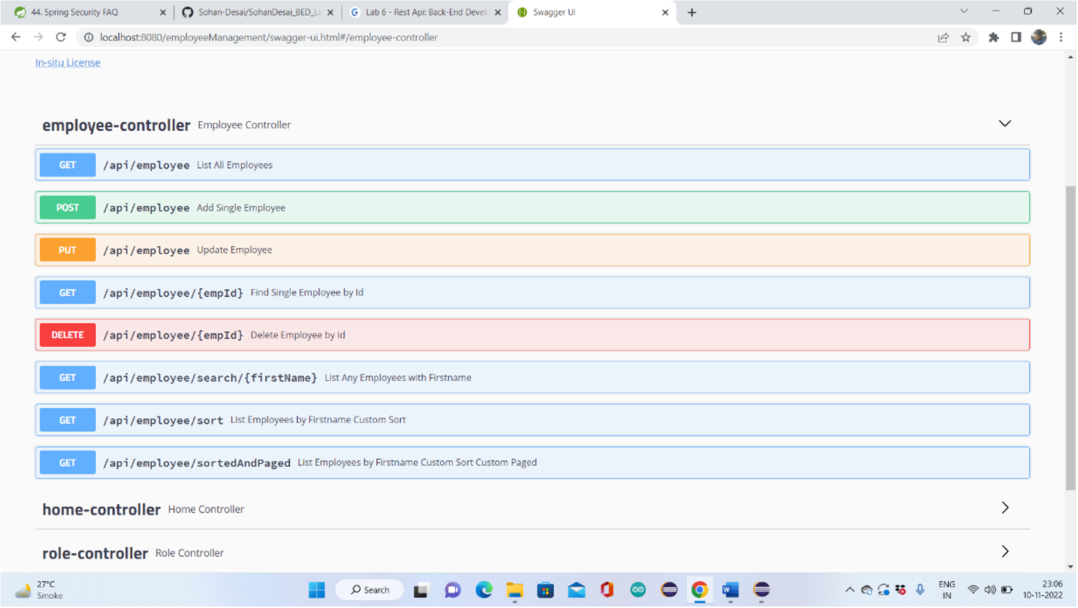
(5 rows, 0 ms)

SELECT * FROM USERS_ROLES;

USER_ID	ROLE_ID
1	1

Employee with id=4 deleted successfully

Finally let’s see the complete list of all APIs



localhost:8080/employeeManagement/swagger-ui.html#/employee-controller

In-situ License

employee-controller Employee Controller

GET /api/employee List All Employees

POST /api/employee Add Single Employee

PUT /api/employee Update Employee

GET /api/employee/{empId} Find Single Employee by Id

DELETE /api/employee/{empId} Delete Employee by Id

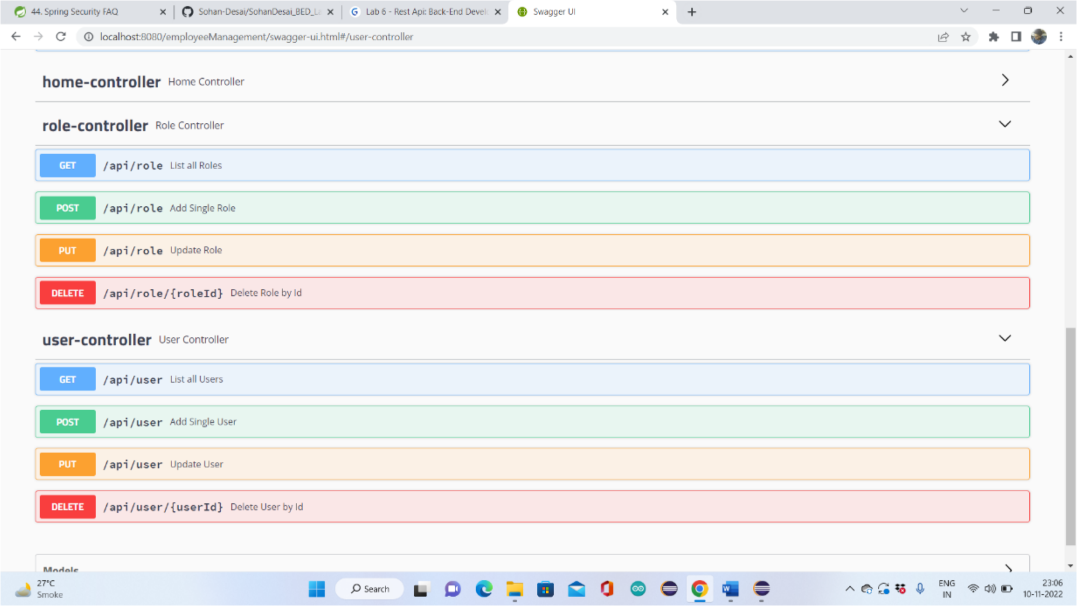
GET /api/employee/search/{firstName} List Any Employees with Firstname

GET /api/employee/sort List Employees by Firstname Custom Sort

GET /api/employee/sortedAndPaged List Employees by Firstname Custom Sort Custom Paged

home-controller Home Controller

role-controller Role Controller



localhost:8080/employeeManagement/swagger-ui.html#/user-controller

home-controller Home Controller

role-controller Role Controller

GET /api/role List all Roles

POST /api/role Add Single Role

PUT /api/role Update Role

DELETE /api/role/{roleId} Delete Role by Id

user-controller User Controller

GET /api/user List all Users

POST /api/user Add Single User

PUT /api/user Update User

DELETE /api/user/{userId} Delete User by Id

Models

Thus we’ve successfully demonstrated all the CRUD operations with Spring Security.

Also you will find a video recording of this project on our google drive. Link to which is shared below.

https://drive.google.com/drive/folders/1suDKkNk5ys9AtyH-Q4sYf1E_LgGtxx25?usp=sharing

Thanks again!