

GET /getAllEmployees

Parameters

No parameters

Execute Clear

Responses

Curl

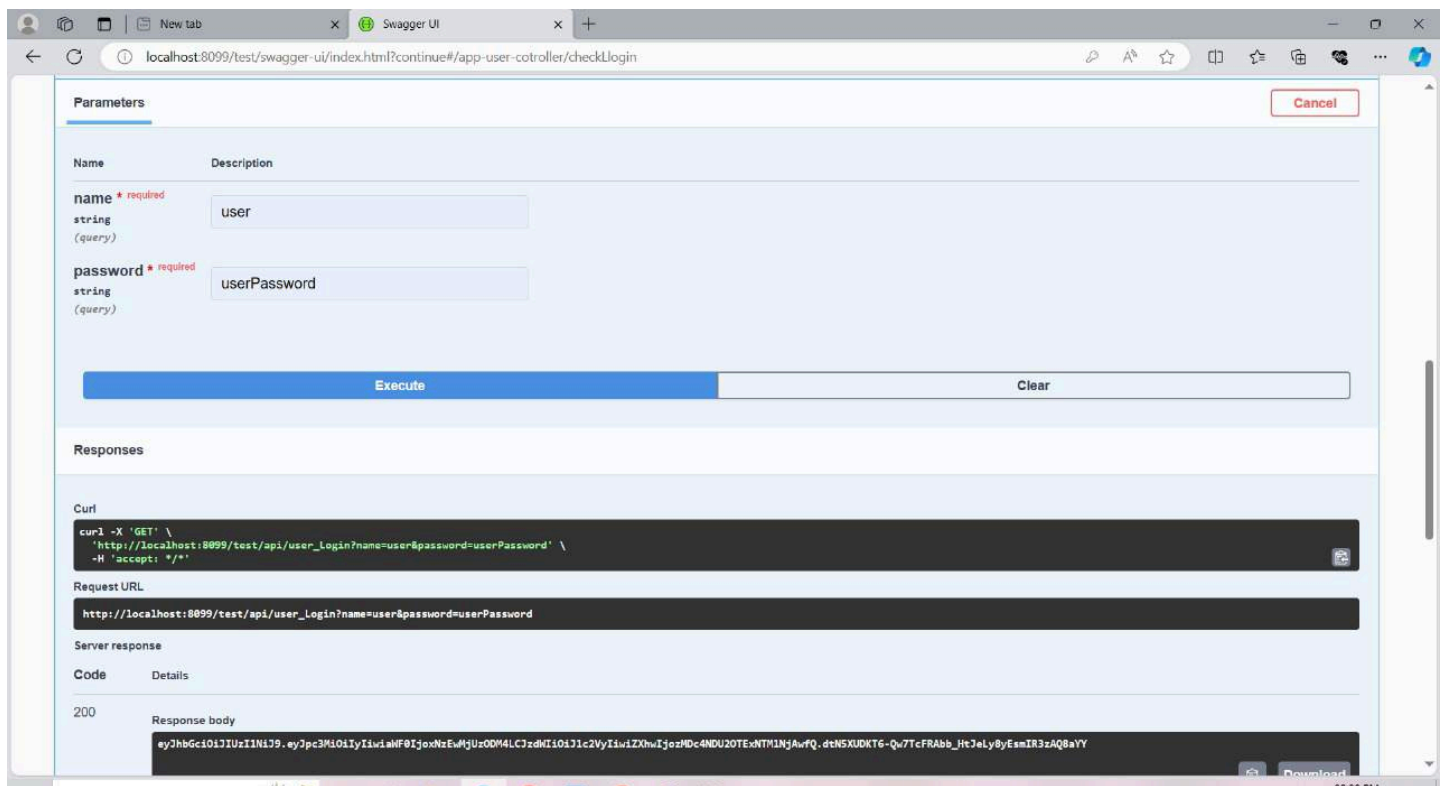
```
curl -X 'GET' \
  'http://localhost:8099/test/getAllEmployees' \
  -H 'accept: */*'
```

Request URL

http://localhost:8099/test/getAllEmployees

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "id": 1, "firstname": "swathi", "lastname": "kn", "email": "swathj@gmail.com" }, { "id": 2, "firstname": "sri", "lastname": "k", "email": "sri@gmail.com" }]</pre>



Swagger UI interface showing a REST client request and response.

Request:

- URL:** `localhost:8099/test/swagger-ui/index.html?continue#/employee-controller/addEmployee`
- Method:** POST
- Body:** `email=swa@gmail.com`

Response:

- Status:** 200
- Body:** `Employee added`
- Headers:**
 - `cache-control: no-cache, no-store, max-age=0, must-revalidate`
 - `connection: keep-alive`
 - `content-length: 14`
 - `content-type: text/plain; charset=UTF-8`
 - `date: Tue, 12 Mar 2024 14:25:32 GMT`
 - `expires: 0`
 - `keep-alive: timeout=60`

employee-controller

PUT /updateEmployee

POST /addNewEmployee

Parameters

Cancel

Name	Description
id * required integer(\$int32) (query)	1
firstname * required string (query)	swathi
lastname * required string (query)	kn
email * required string (query)	swa@gmail.com

Execute

Responses

Code	Description	Links
------	-------------	-------

Swagger UI interface showing the DELETE endpoint for /deleteEmployee.

DELETE /deleteEmployee

Parameters

Name	Description
id <small>* required</small>	
integer(\$int32)	
(query)	

2

Execute **Clear**

Responses

Curl

```
curl -X 'DELETE' \
  'http://localhost:8099/test/deleteEmployee?id=2' \
  -H 'accept: */*'
```

Request URL

```
http://localhost:8099/test/deleteEmployee?id=2
```

Server response

Code	Details
200	<p>Response body</p> <p>Employee deleted</p>

Download

Swagger UI interface showing the GET /sortemployee endpoint.

Parameters: No parameters. [Cancel](#)

Execute **Clear**

Responses:

Curl:

```
curl -X 'GET' \
  'http://localhost:8099/test/sortemployee' \
  -H 'accept: */*'
```

Request URL: http://localhost:8099/test/sortemployee

Server response:

Code	Details
200	<p>Response body</p> <pre>{ "id": 2, "firstname": "sri", "lastname": "k", "email": "sri@gmail.com" }, { "id": 1, "firstname": "swathi", "lastname": "kn", "email": "swathi@gmail.com" }</pre>

[Download](#)

Swagger UI interface showing the execution of a DELETE API endpoint.

API Endpoint: `localhost:8099/test/swagger-ui/index.html?continue#/app-user-controller/deleteEmployee`

Parameters:

- id** (required): `integer($int32)` (query). Value: `1`
- token** (required): `string` (query). Value: `eyJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJlbiwWF`

Buttons: Execute, Clear

Responses:

Curl:

```
curl -X 'POST' \
'http://localhost:8099/test/api/delete_employee?id=1&token=eyJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJlbiwWF0IjoxNzEwMjUzODM4LjZhdWII01J1c2VyIiwiaXNwIjoxMDc4NDU2OTExNTM1NjAwfQ.dtNSXUDKT6-Qw7TcFRAbb_Ht3eLy8yEsmIR3zAQ' \
-H 'accept: */*' \
-d ''
```

Request URL:

```
http://localhost:8099/test/api/delete_employee?id=1&token=eyJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJlbiwWF0IjoxNzEwMjUzODM4LjZhdWII01J1c2VyIiwiaXNwIjoxMDc4NDU2OTExNTM1NjAwfQ.dtNSXUDKT6-Qw7TcFRAbb_Ht3eLy8yEsmIR3zAQ8aYY
```

Server response:

Code	Details
200	<p>Response body: <code>Employee Deleted</code></p> <p>Download</p>

Swagger UI

localhost:8099/test/swagger-ui/index.html?continue

Swagger

/test/v3/api-docs

Explore

OpenAPI definition v0 OAS3
[/test/v3/api-docs](#)

Servers

http://localhost:8099/test - Generated server url

employee-controller

PUT /updateEmployee

POST /addNewEmployee

GET /sortemployee

GET /searchbyemployee

GET /getEmployeeById

GET /getAllEmployees

Swagger UI interface showing a REST client for the `updateEmployee` endpoint.

Request Parameters:

- `lastname` (string, query, required):
- `email` (string, query, required):

Buttons: [Execute](#) [Clear](#)

Responses

Curl

```
curl -X 'PUT' \
  'http://localhost:8099/test/updateEmployee?id=1&firstname=swathi&lastname=kn&email=swathi%40gmail.com' \
  -H 'accept: */*'
```

Request URL

```
http://localhost:8099/test/updateEmployee?id=1&firstname=swathi&lastname=kn&email=swathi%40gmail.com
```

Server response

Code	Details
200	<p>Response body</p> <pre>Employee updated</pre> <p>Download</p> <p>Response headers</p> <pre>cache-control: no-cache, no-store, max-age=0, must-revalidate</pre>

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- car
- chennai01
- chennai_servlet
- e_commerce
- employee
- employees
- example
- gl
- gradedassignment
- Tables
- Views
- Administration
- Schemas

Information: [SQL Developer 12.1.0.2.0 Help](#)

No object selected

Query 1 SQL File 2* SQL File 4* SQL File 5* SQL File 6* SQL File 7*

```
1 create database SpringBoot_Assignment;  
2 use SpringBoot_Assignment;  
3 show tables;  
4 select * from employee;
```

Limit to 50000 rows

SQL Additions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

	id	email	firstname	lastname
1	ova@gmail.com	swathi	kn	

employee 3 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
4	19:49:35	create database SpringBoot_Assignment	1 row(s) affected	0.688 sec
5	19:49:50	use SpringBoot_Assignment	0 row(s) affected	0.016 sec
6	19:49:57	show tables	0 row(s) returned	0.031 sec / 0.000 sec
7	19:55:56	show tables	3 row(s) returned	0.015 sec / 0.000 sec
8	19:56:19	select * from employee LIMIT 0, 50000	1 row(s) returned	0.390 sec / 0.000 sec

Object Info Session

Query Completed

Swagger UI interface showing API endpoints for an application.

Endpoints listed:

- GET /getAllEmployees
- DELETE /deleteEmployee
- app-user-cotroller**
 - POST /api/updat_eemployee
 - POST /api/delete_employee
 - POST /api/add_employee
 - GET /getallemployee
 - GET /api/user_login
 - GET /api/get_id
 - GET /add_form

Schemas:

- Employee >

MySQL Workbench

Local instance MySQL x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- car
- chennai01
- chennai_servlet
- e_commerce
- employee
- employees
- example
- gl
- gradedassignment
- Tables
- Views
- Stored Procedures

Administration Schemas

Information

No object selected

Query 1 SQL File 2* SQL File 4* SQL File 5* SQL File 6* SQL File 7*

```
1 create database SpringBoot_Assignment;
2 use SpringBoot_Assignment;
3 show tables;
4 select * from employee;
```

Limit to 50000 rows

SQL Additions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

id	email	firstname	lastname
1	swathi@gmail.com	swathi	kn

employee 4 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
5	19:49:50	use SpringBoot_Assignment	0 row(s) affected	0.016 sec
6	19:49:57	show tables	0 row(s) returned	0.031 sec / 0.000 sec
7	19:55:56	show tables	3 row(s) returned	0.015 sec / 0.000 sec
8	19:56:19	select * from employee LIMIT 0, 50000	1 row(s) returned	0.390 sec / 0.000 sec
9	19:57:19	select * from employee LIMIT 0, 50000	1 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Query Completed

