

## Employee Management REST API Screenshots

The screenshot shows a SQL IDE window with a script editor containing the following SQL commands:

```
1 • create database emra_db;
2 • use emra_db;
3 • select * from employees;
4 • select * from users;
5 • select * from roles;
6 • select * from users_roles;
7 • describe employees;
8 • describe users;
9 • truncate table employees;
10 • truncate table roles;
```

The 'Result Grid' shows the 'roles' table with columns 'id' and 'name', both containing 'NULL'. The 'Output' pane displays the execution log:

#	Time	Action	Message	Duration / Fetch
1	17:18:40	select * from employees	7 row(s) returned	0.000 sec / 0.000 sec
2	17:20:12	delete from employees where id = 14	1 row(s) affected	0.015 sec
3	17:20:19	select * from employees	6 row(s) returned	0.000 sec / 0.000 sec
4	17:20:36	delete from employees where id = 13	1 row(s) affected	0.016 sec
5	17:20:44	select * from employees	5 row(s) returned	0.000 sec / 0.000 sec
6	17:21:46	select * from roles	0 row(s) returned	0.000 sec / 0.000 sec

Pic: Blank Roles Table Before Role Insertion in DB.

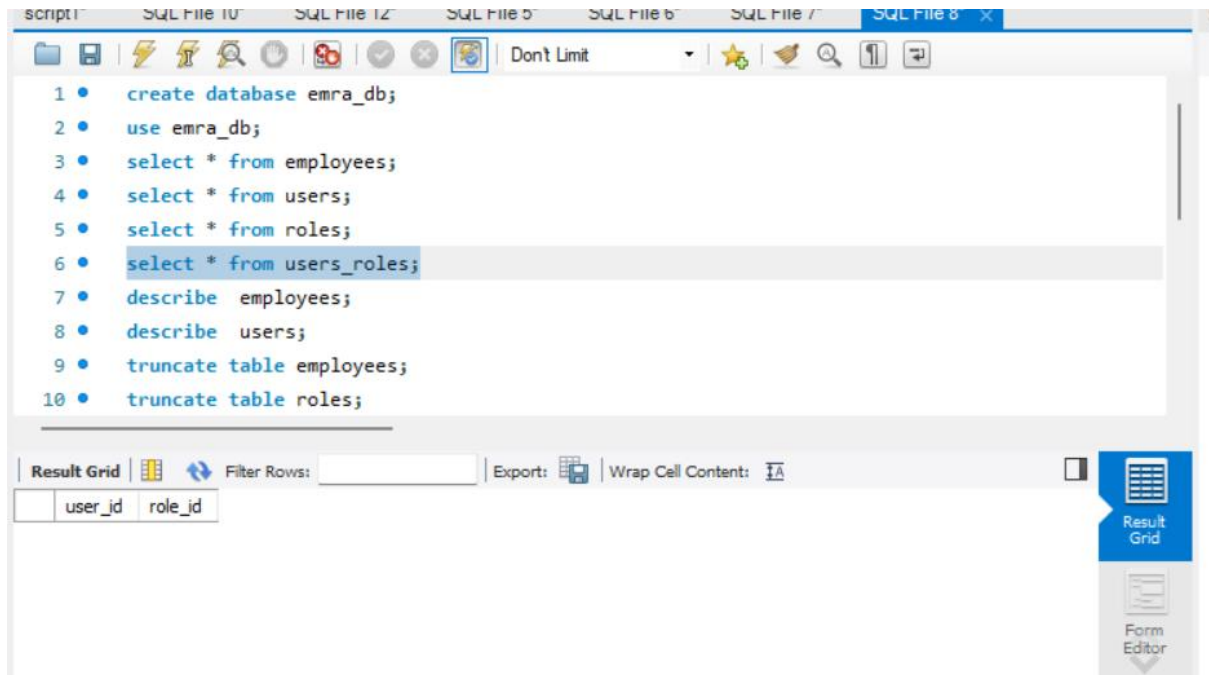
The screenshot shows a SQL IDE window with a script editor containing the following SQL commands:

```
1 • create database emra_db;
2 • use emra_db;
3 • select * from employees;
4 • select * from users;
5 • select * from roles;
6 • select * from users_roles;
7 • describe employees;
8 • describe users;
9 • truncate table employees;
10 • truncate table roles;
```

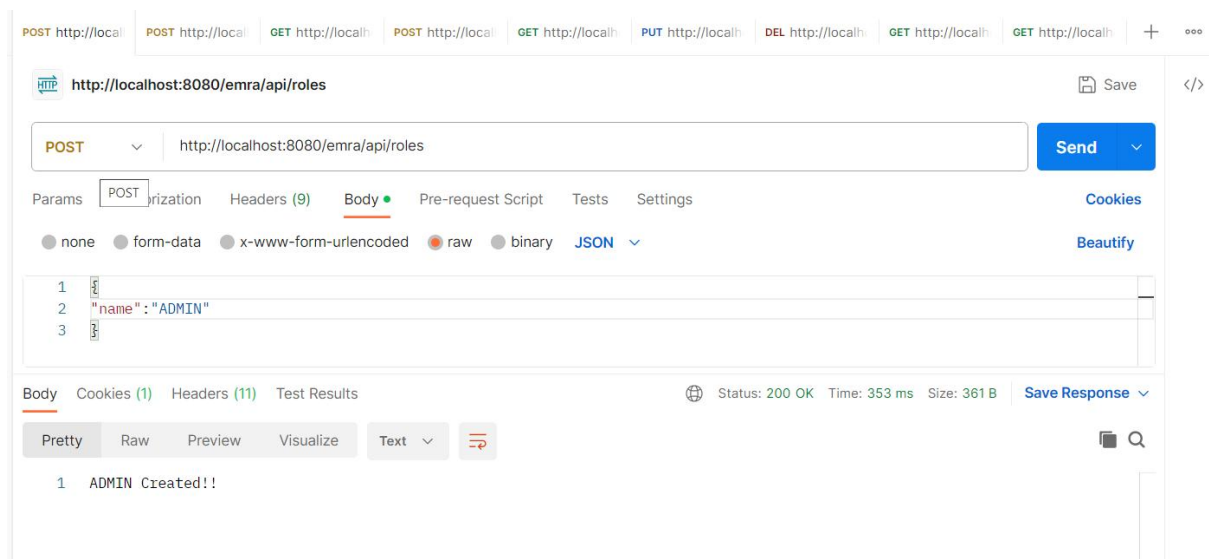
The 'Result Grid' shows the 'users' table with columns 'id', 'password', and 'username', all containing 'NULL'. The 'Output' pane displays the execution log:

#	Time	Action	Message	Duration / Fetch
1	17:18:40	select * from employees	7 row(s) returned	0.000 sec / 0.000 sec
2	17:20:12	delete from employees where id = 14	1 row(s) affected	0.015 sec
3	17:20:19	select * from employees	6 row(s) returned	0.000 sec / 0.000 sec
4	17:20:36	delete from employees where id = 13	1 row(s) affected	0.016 sec
5	17:20:44	select * from employees	5 row(s) returned	0.000 sec / 0.000 sec
6	17:21:46	select * from roles	0 row(s) returned	0.000 sec / 0.000 sec

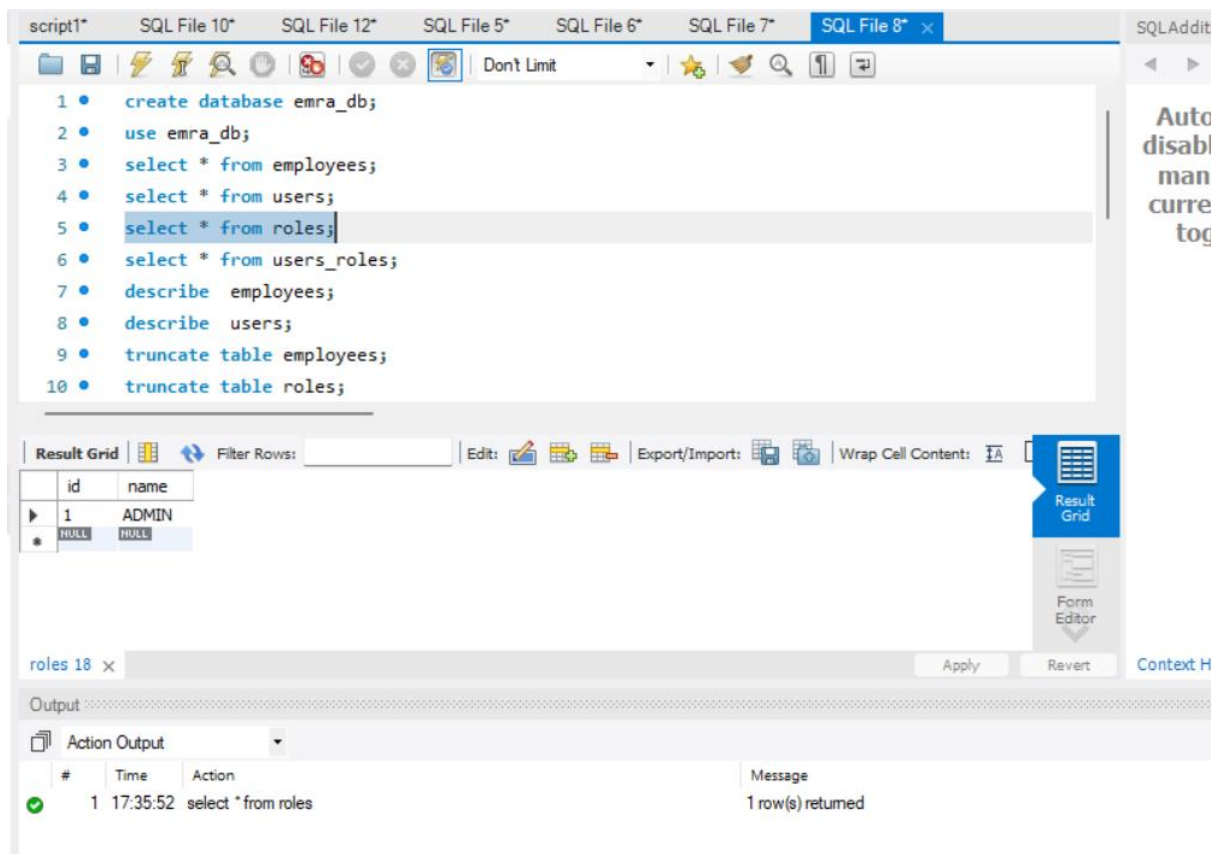
Pic: Blank Users Table Before User Insertion in DB.



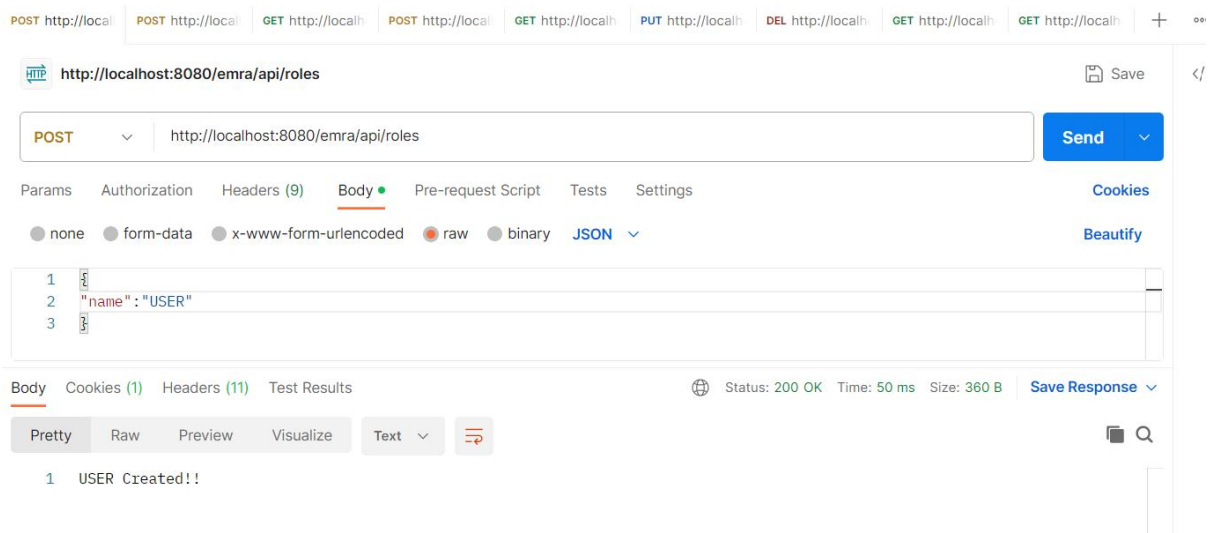
Pic: Blank Users\_Roles Table Before Insertion in DB.



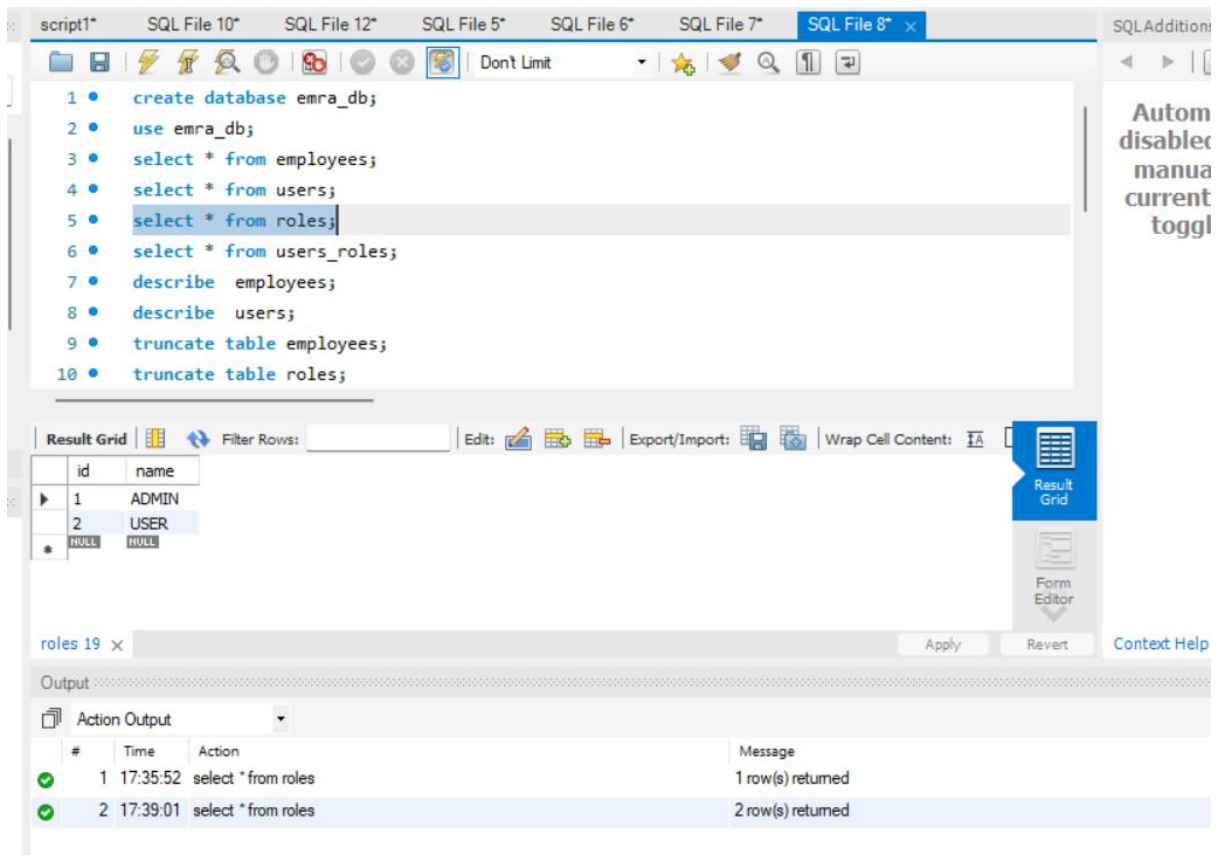
Pic: After the Role (ADMIN) Creation in POSTMAN.



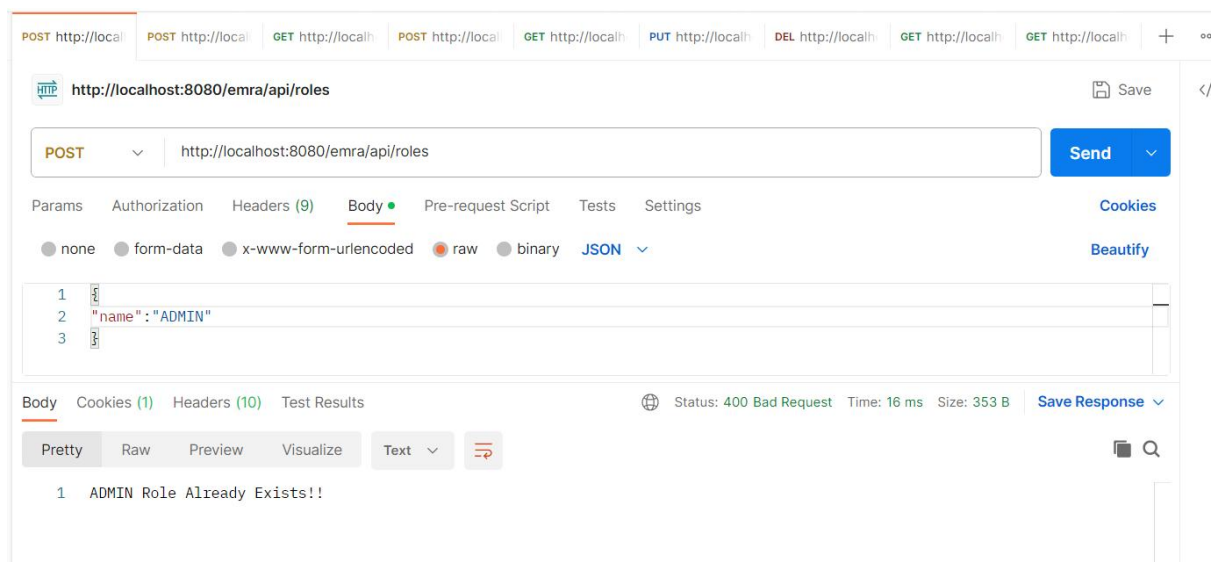
Pic: After the Role (ADMIN) Creation in the DB.



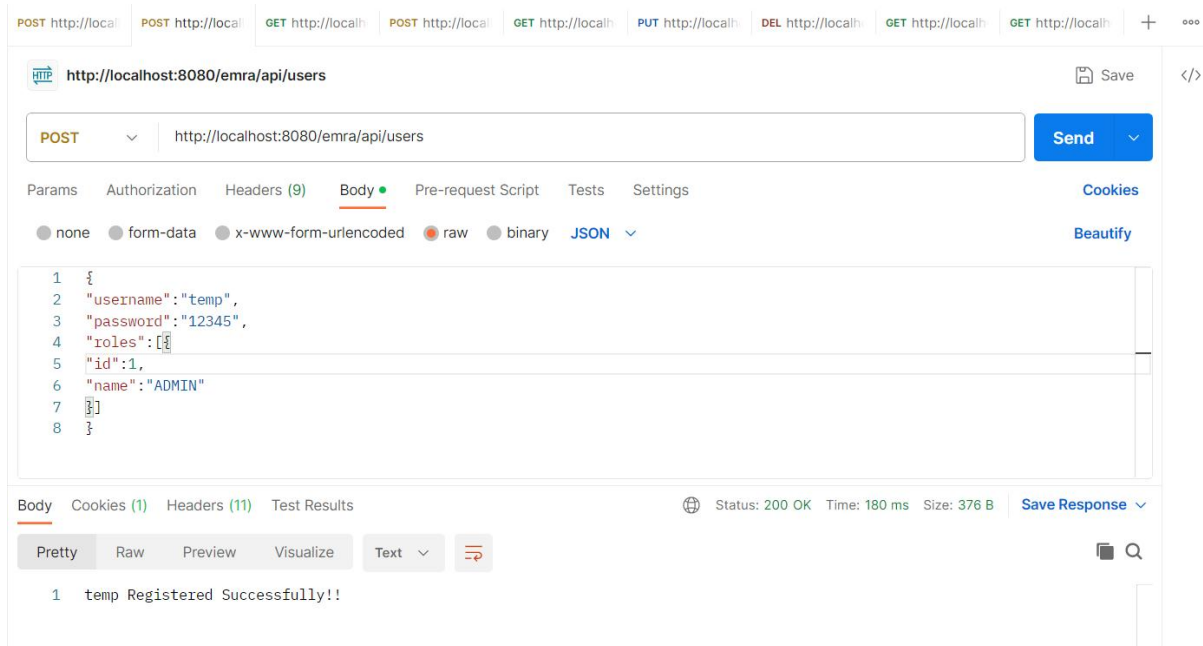
Pic: After the Role (USER) Creation in POSTMAN.



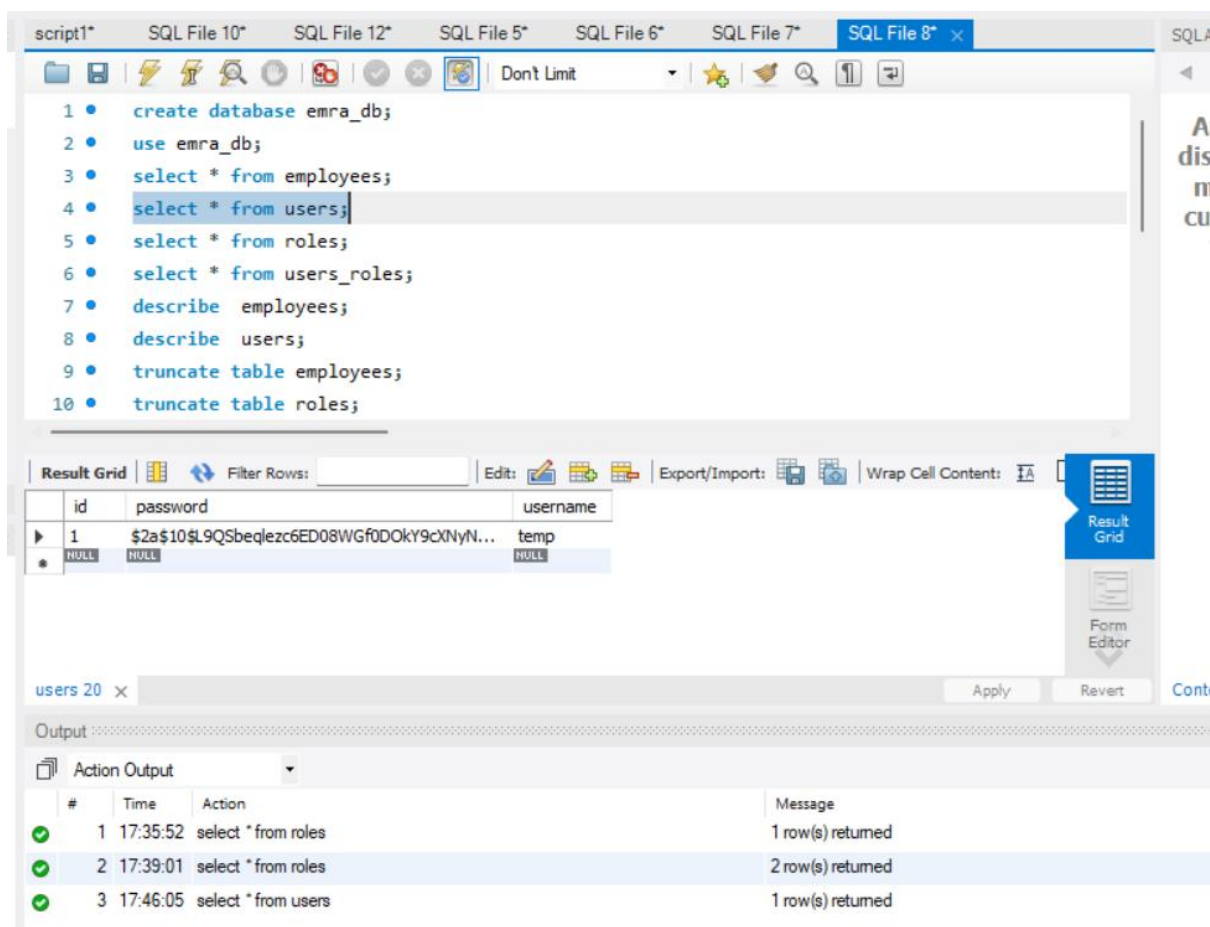
Pic: After the Role (USER) Creation in the DB.



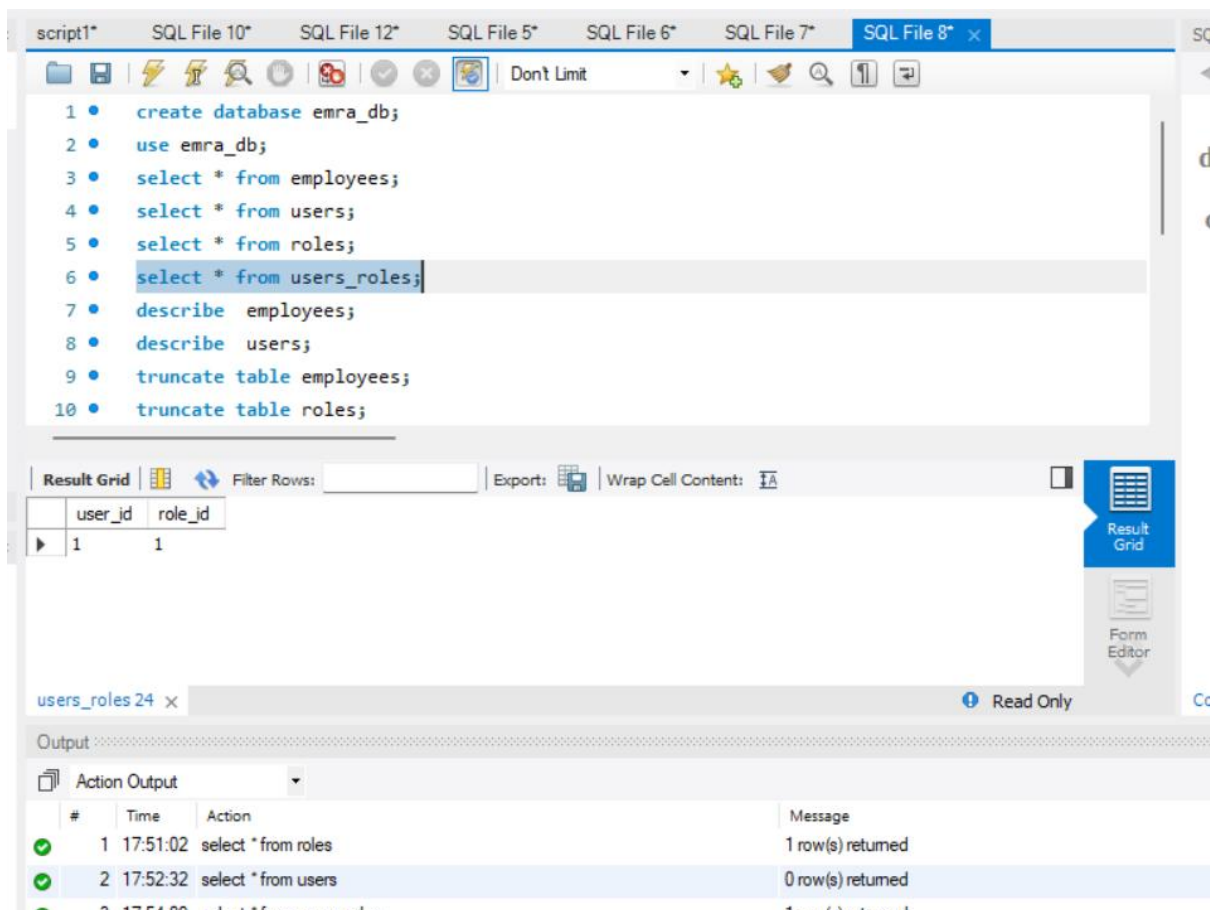
Pic: Restricting from Role Creation with the same role-name in POSTMAN.



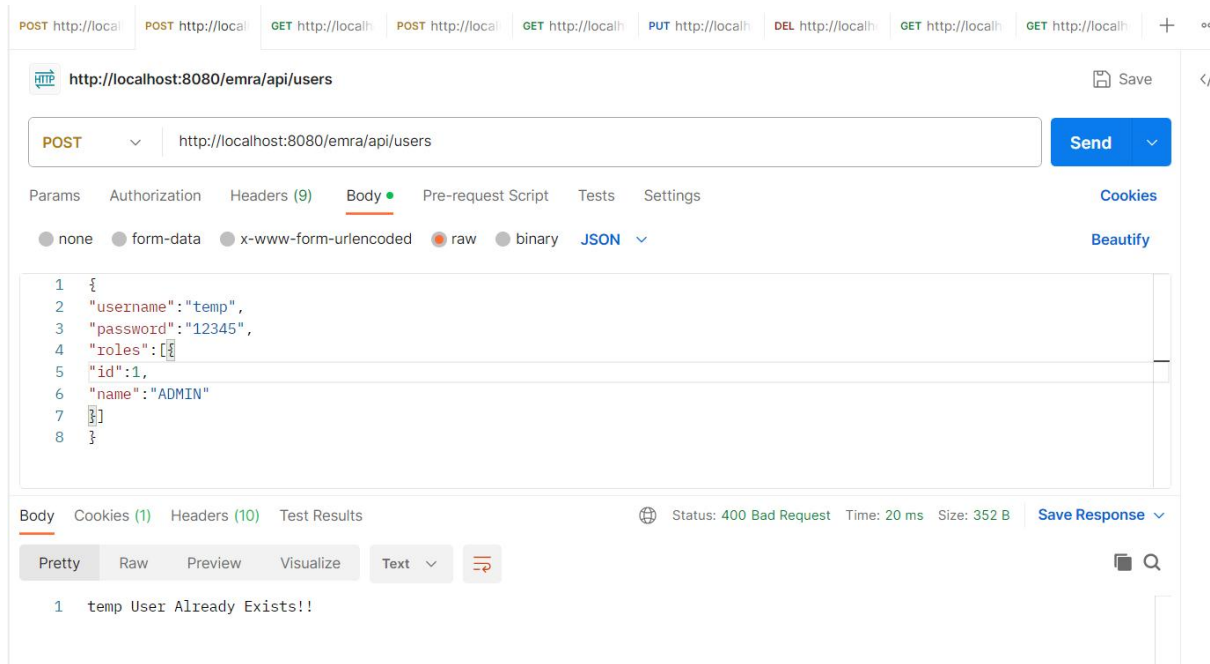
Pic: After the User (temp) Creation in POSTMAN.



Pic: After the User (temp) Creation, the users table, in the DB.

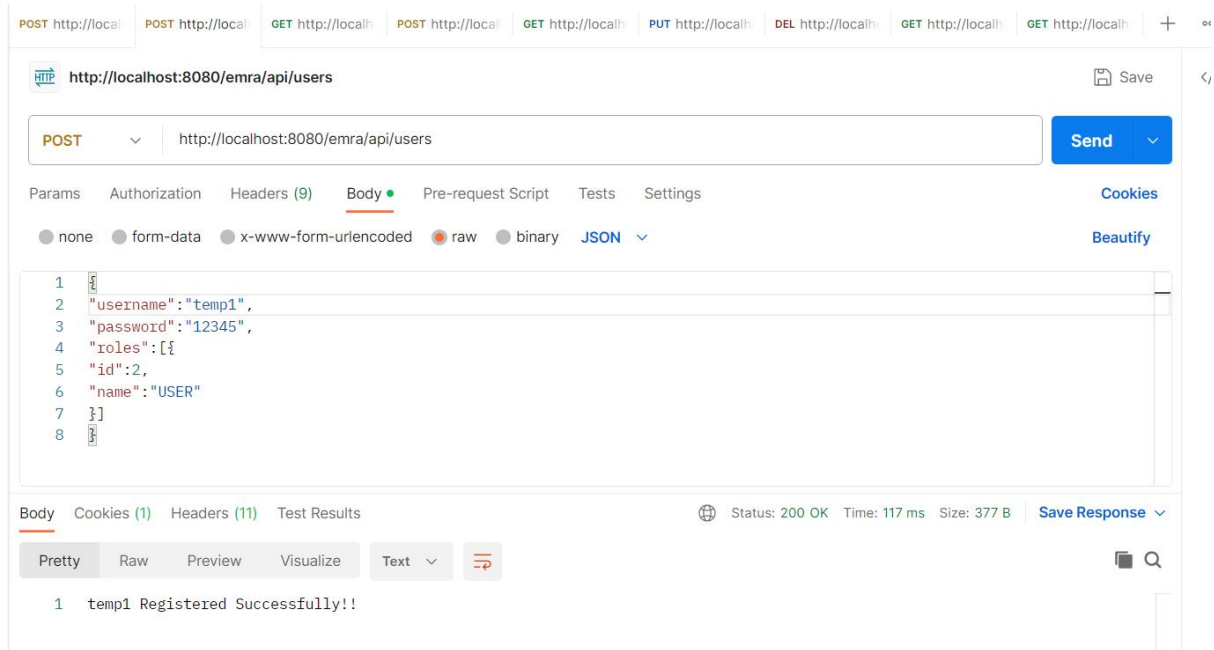


Pic: After the User (temp) Creation, the users\_roles table, in the DB.

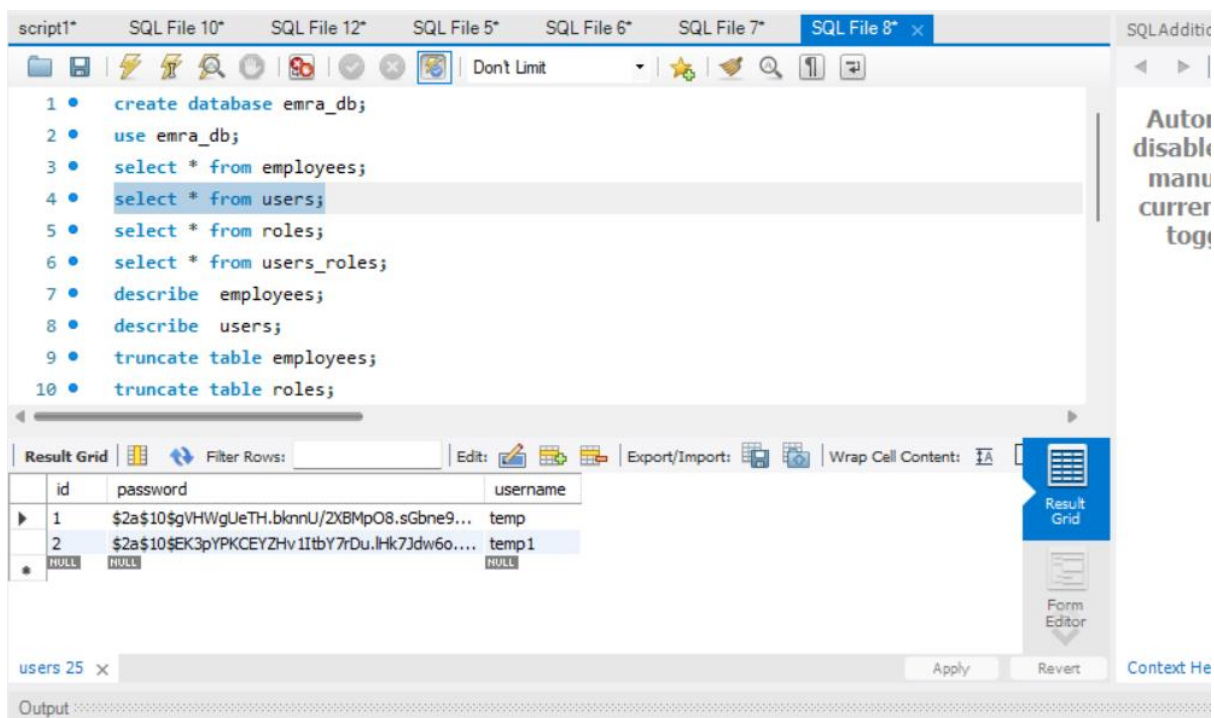


Pic: Restricting from User Creation with the same user-name in POSTMAN

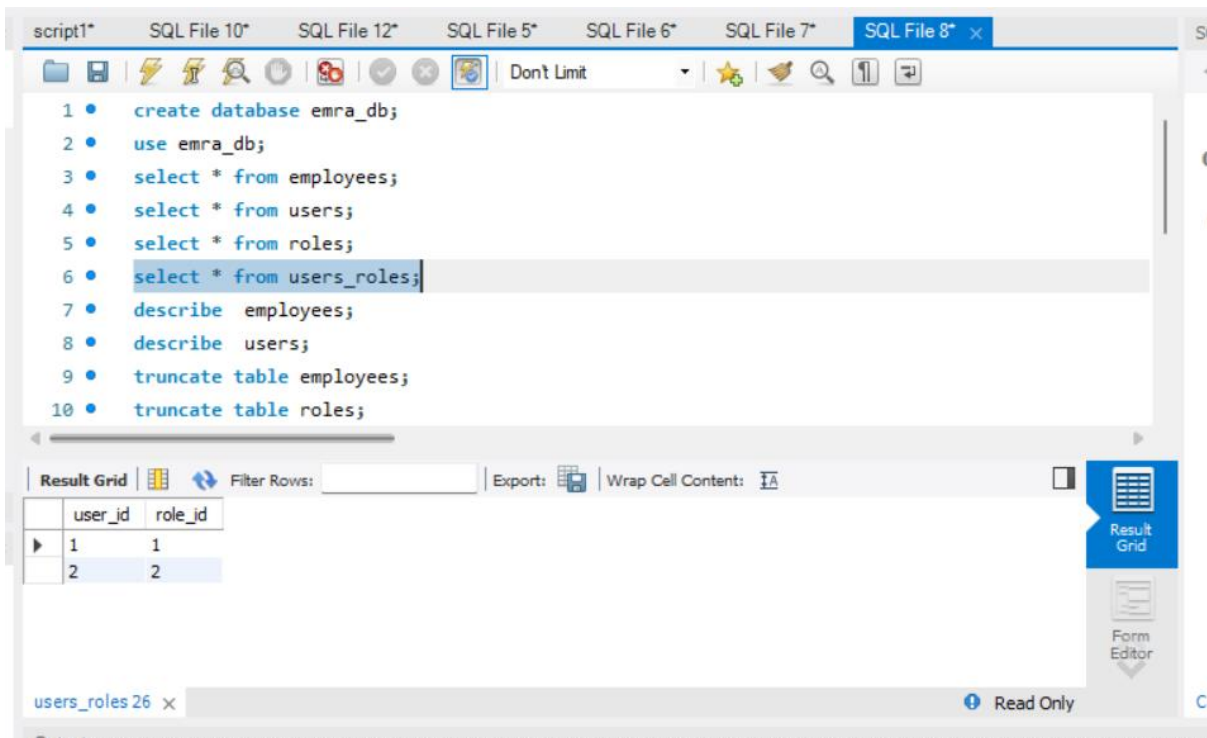




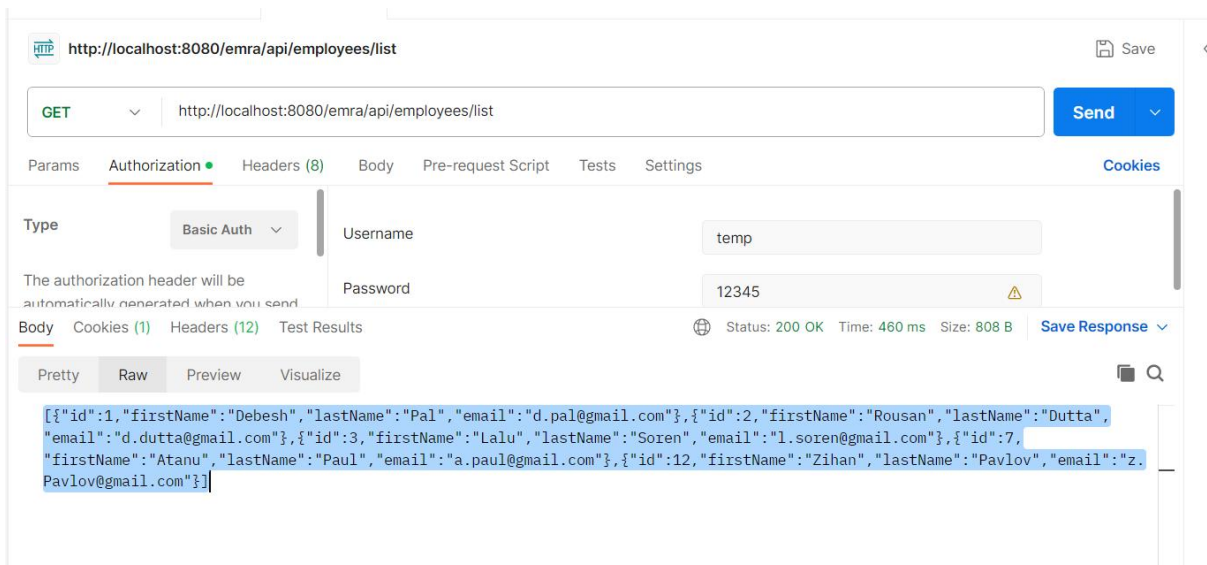
Pic: After the User (temp1) Creation in POSTMAN.



Pic: After the User (temp1) Creation, the users table, in the DB.

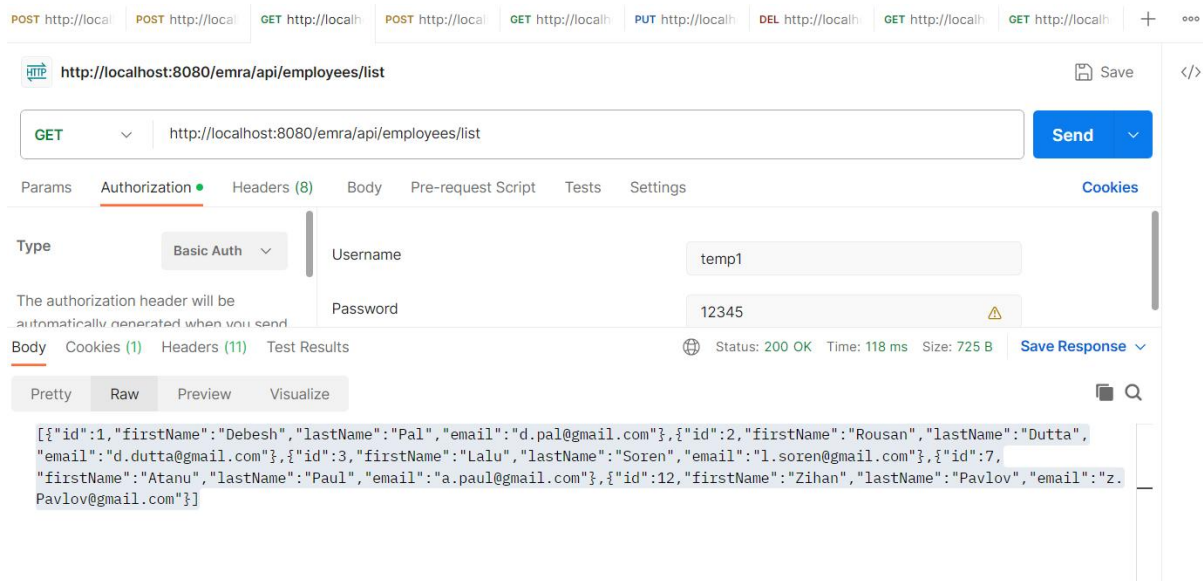


Pic: After the User (temp1) Creation, the users\_roles table, in the DB.

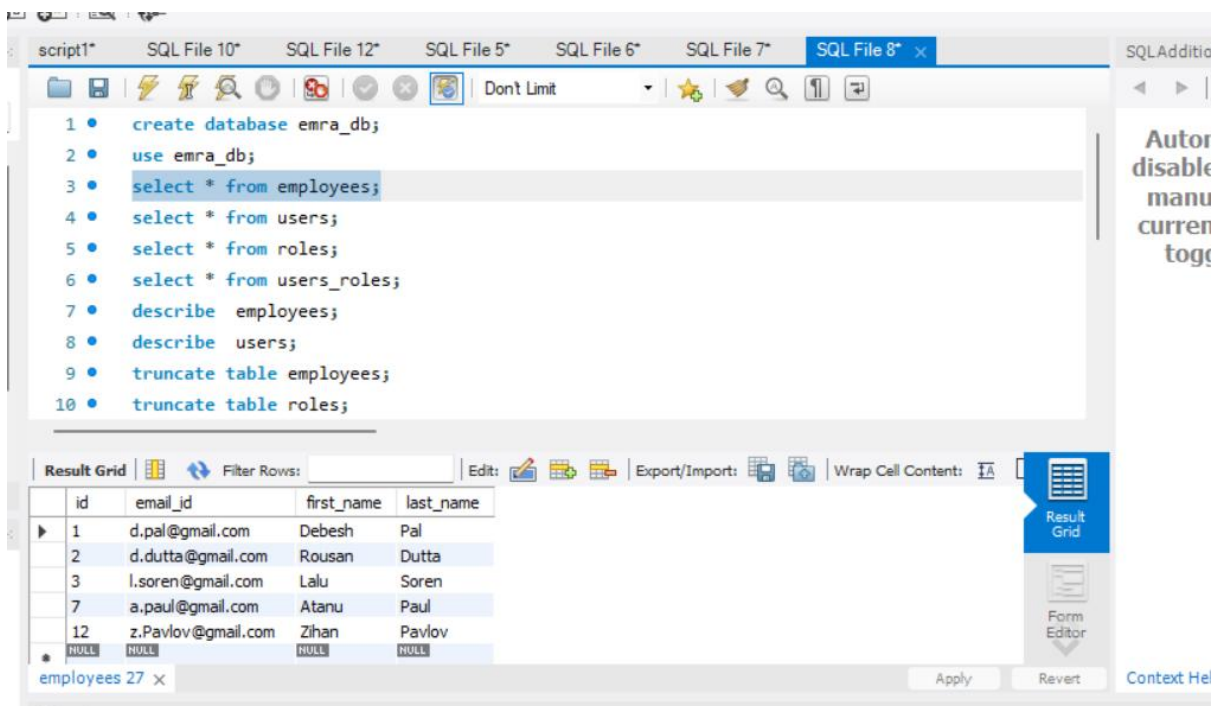


Pic: Listing all the records by the User (temp) Creation in POSTMAN.

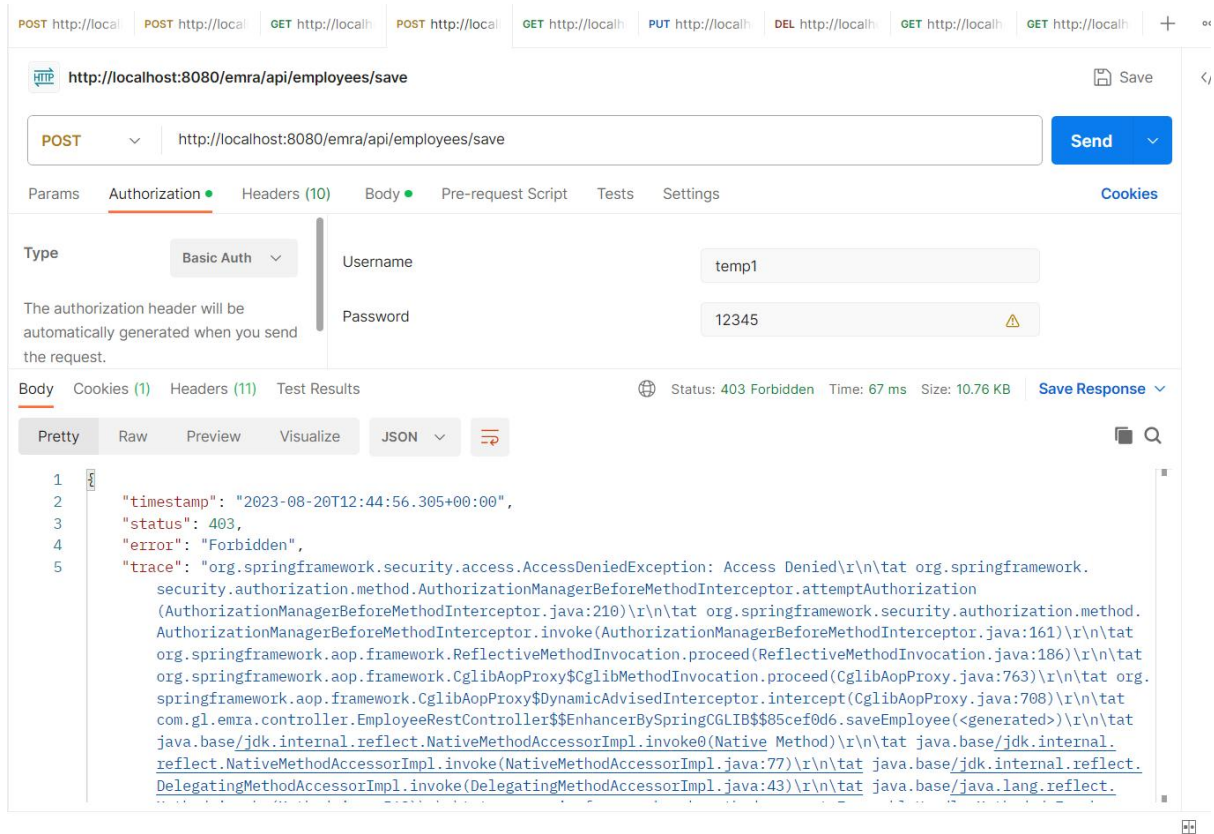




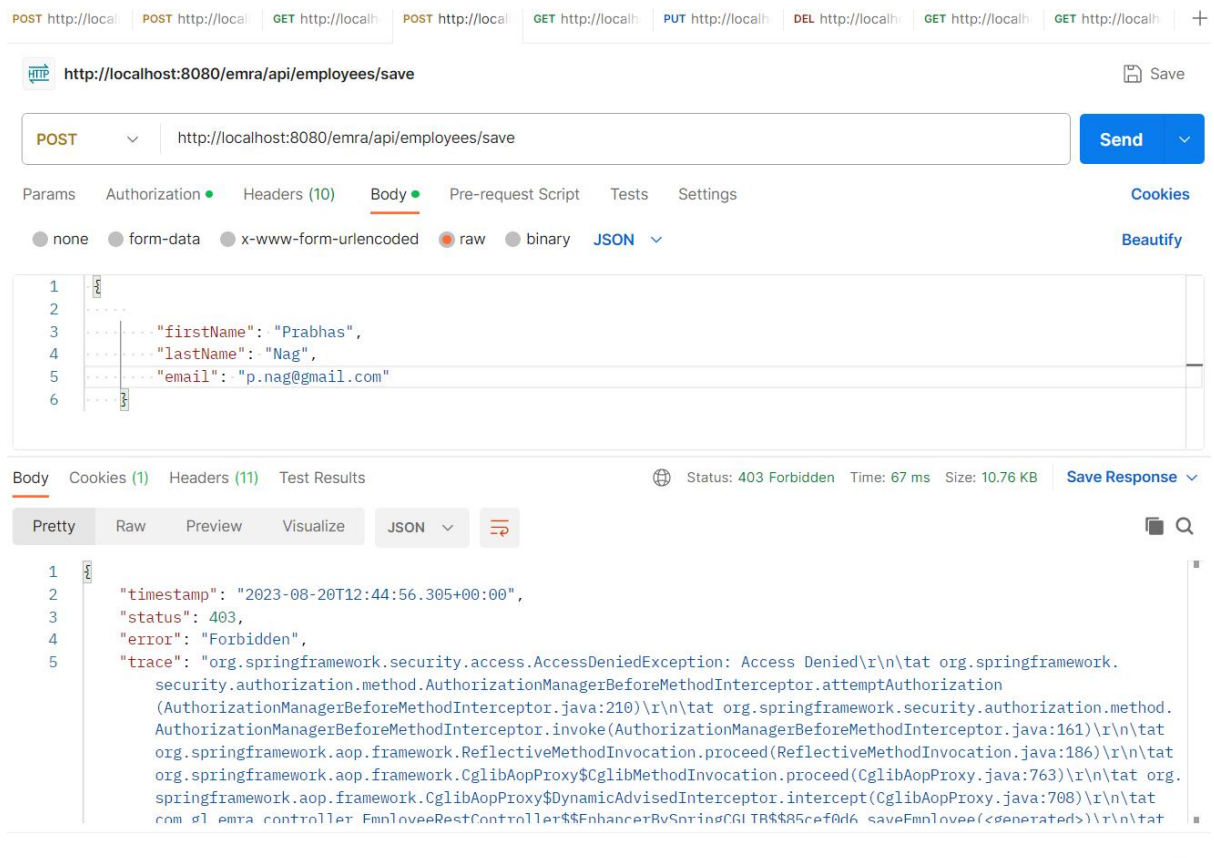
Pic: Listing all the records by the User (temp1) Creation in POSTMAN.



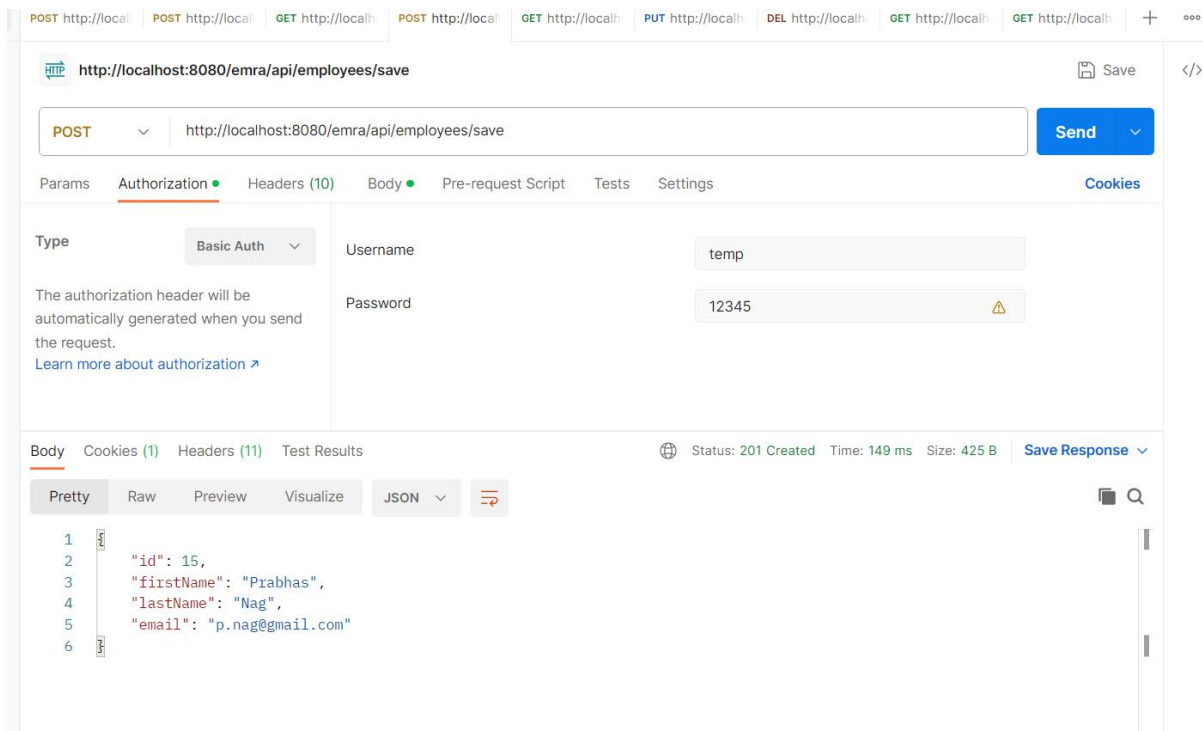
Pic: Before saving of the record of first\_name = Prabhas, the DB picture.



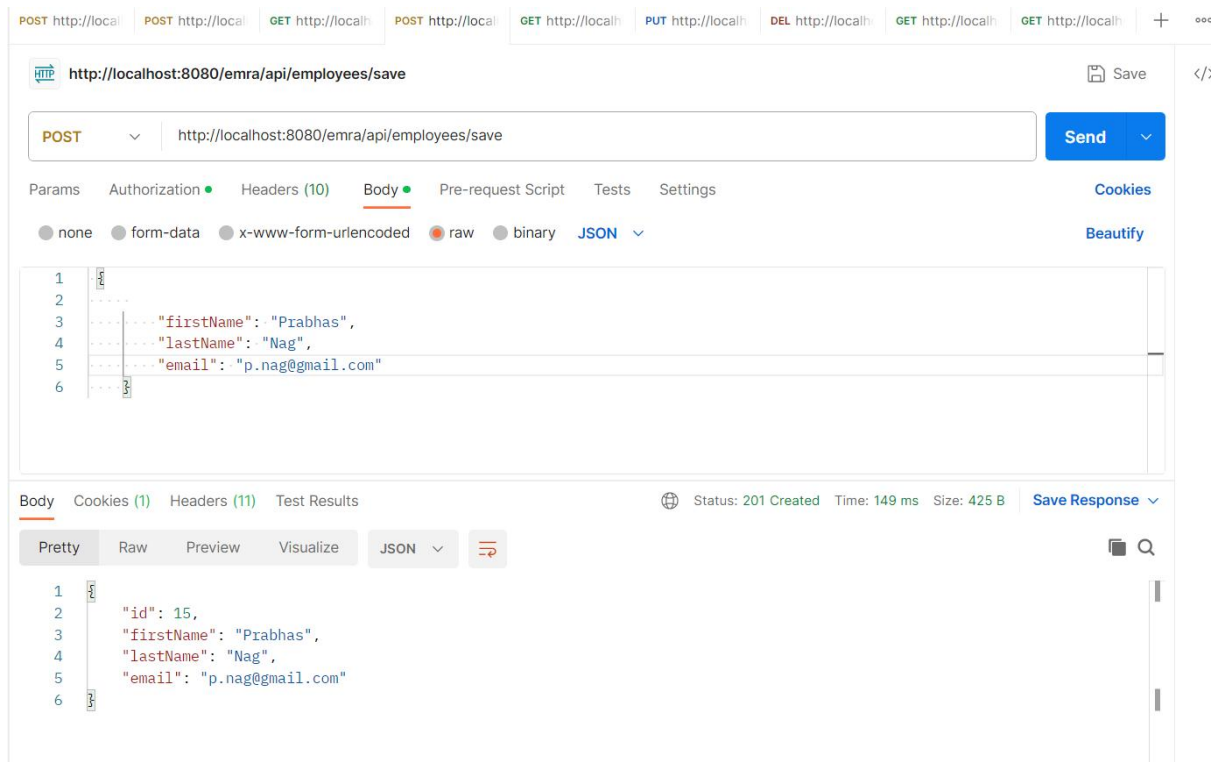
Pic: After the trying to save an employee by the User (temp1, that is normal user) POSTMAN.



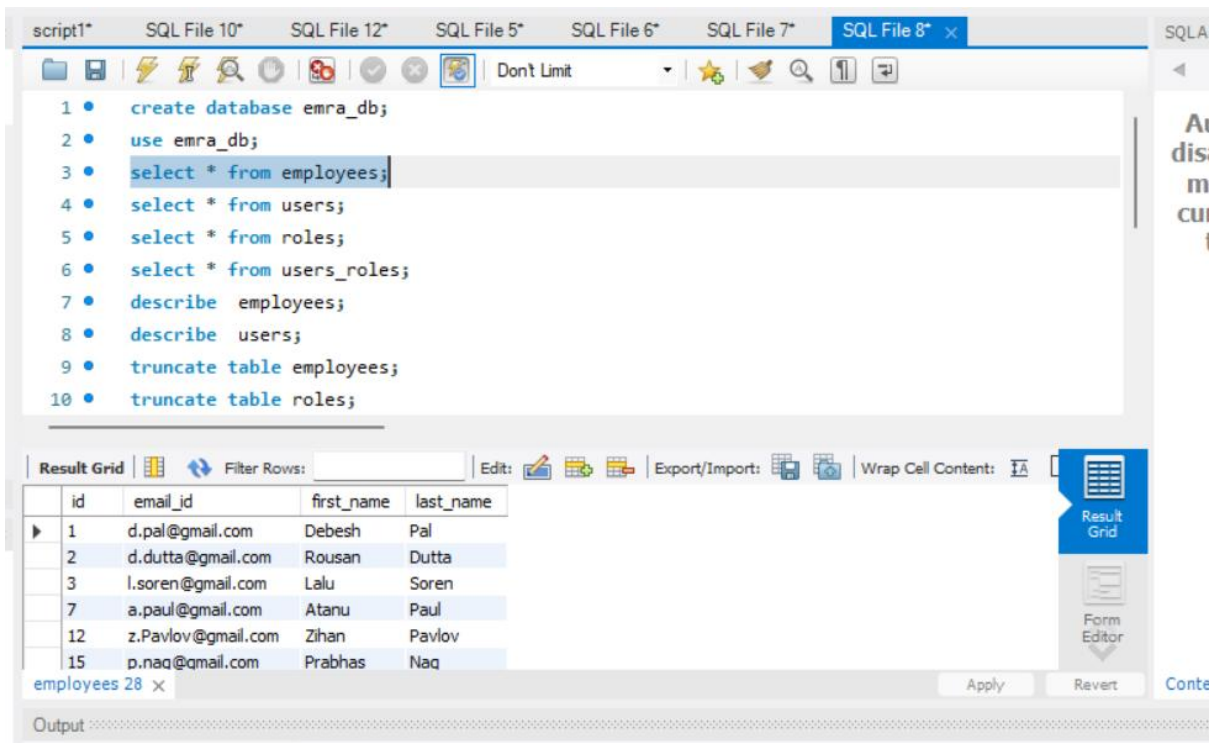
Pic: After the trying to save an employee by the User (temp1, that is normal user) POSTMAN.



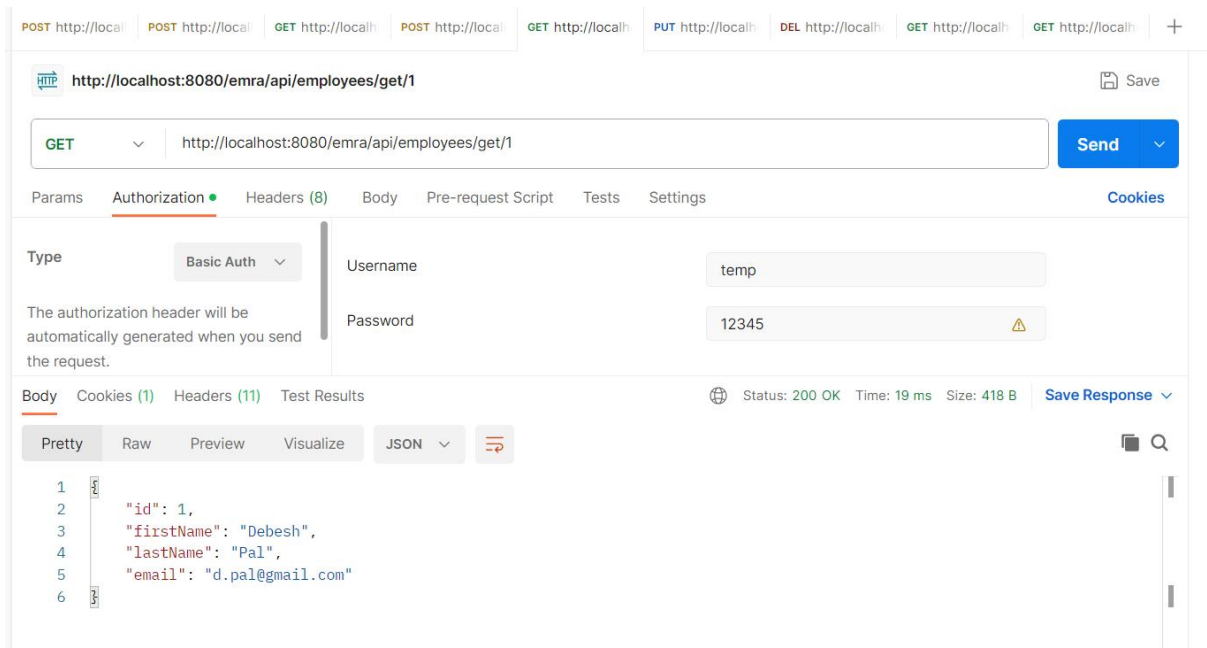
Pic: After the trying to save an employee by the User (temp, that is admin user), POSTMAN.



Pic: After the trying to save an employee by the User (temp, that is admin user), POSTMAN.

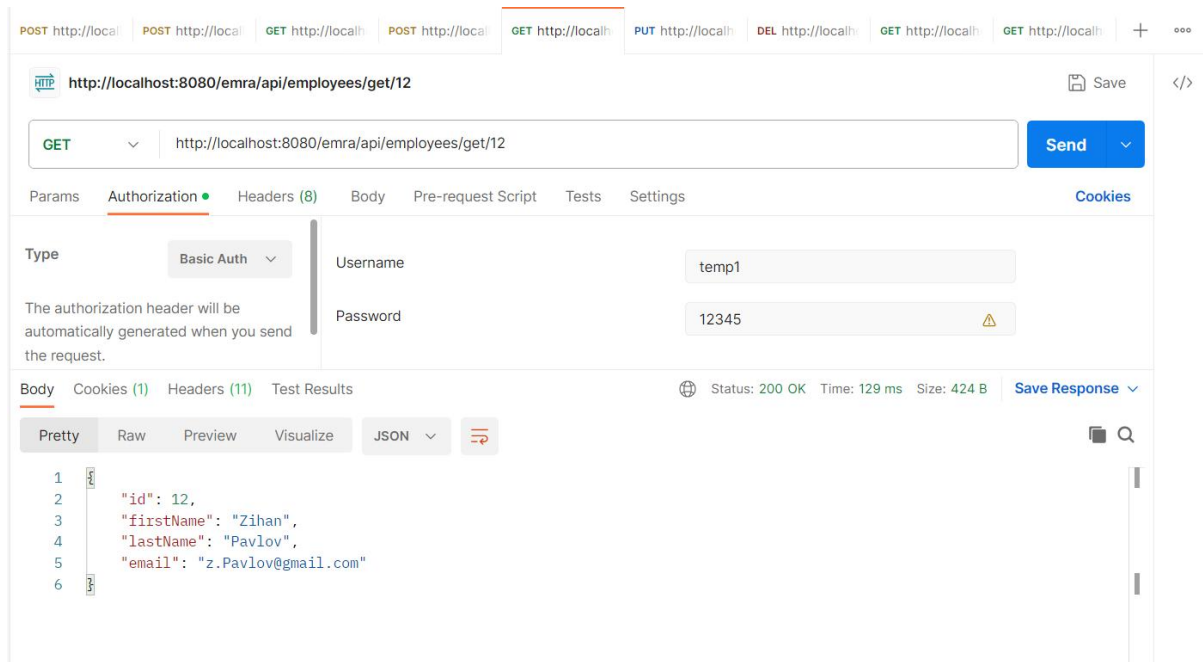


Pic: After the trying to save an employee by the User (temp, that is admin user), the DB.

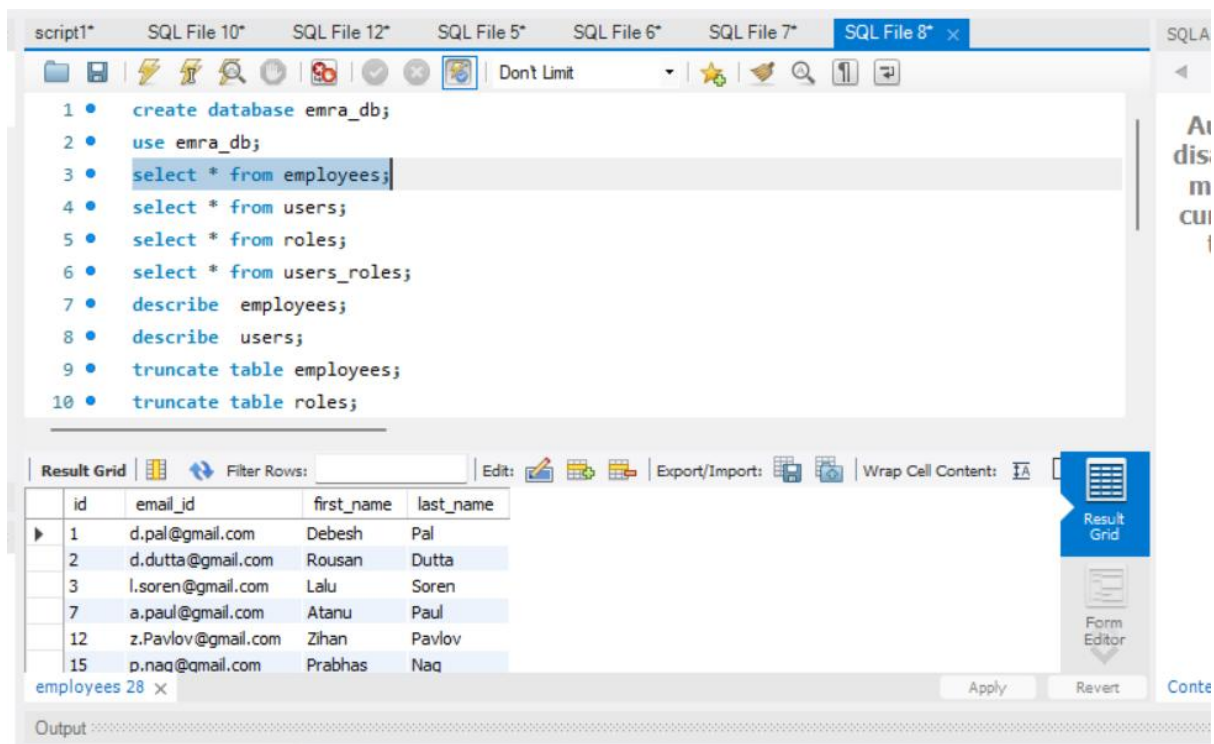


Pic: Get an employee by id the User (temp, that is admin user), the POSTMAN.

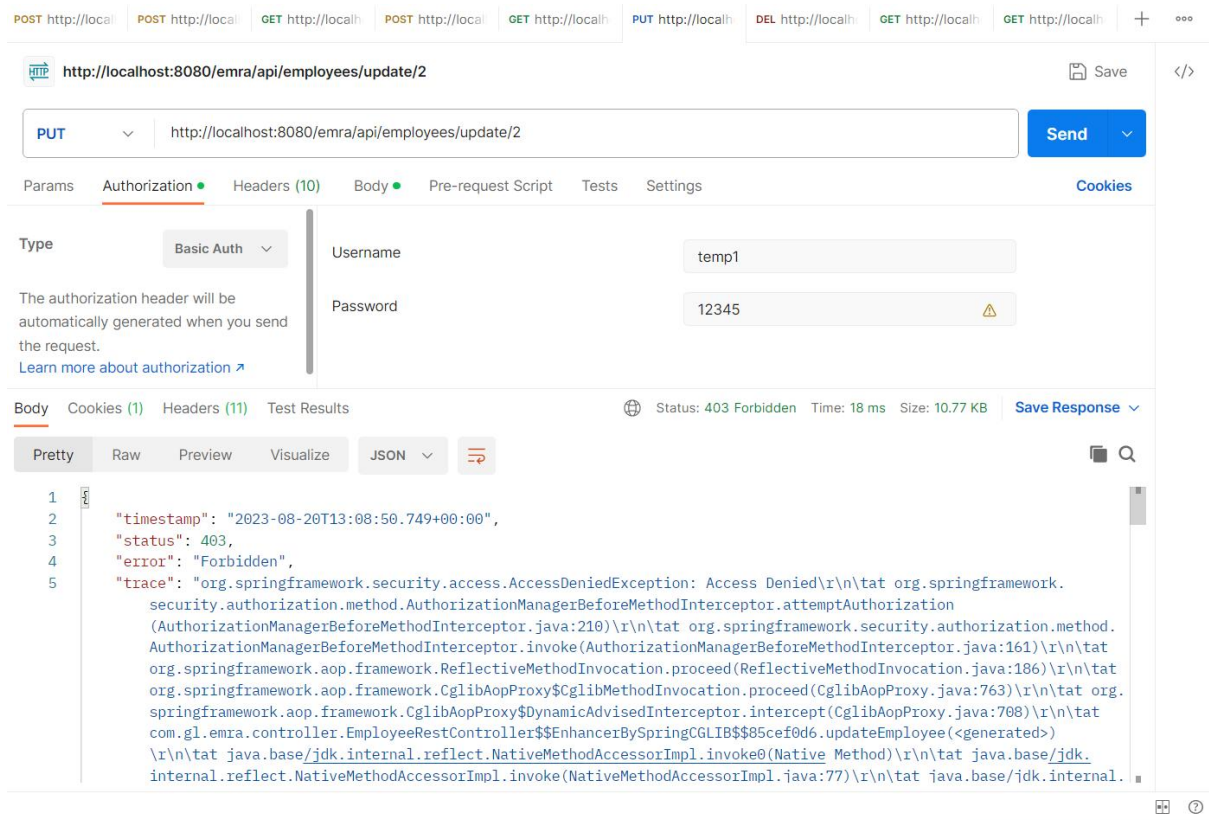




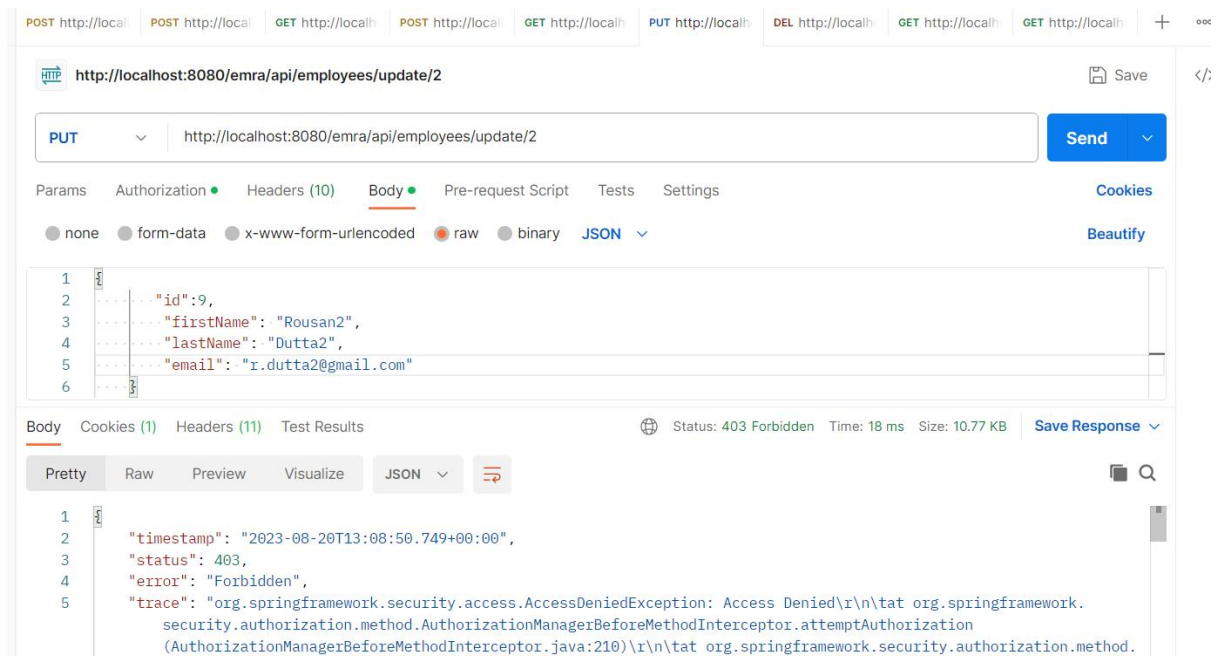
Pic: Get an employee by id the User (temp1, that is normal user), the POSTMAN.



Pic: Before Updation of an employee having id 2 by the User (temp, and temp1), the DB.

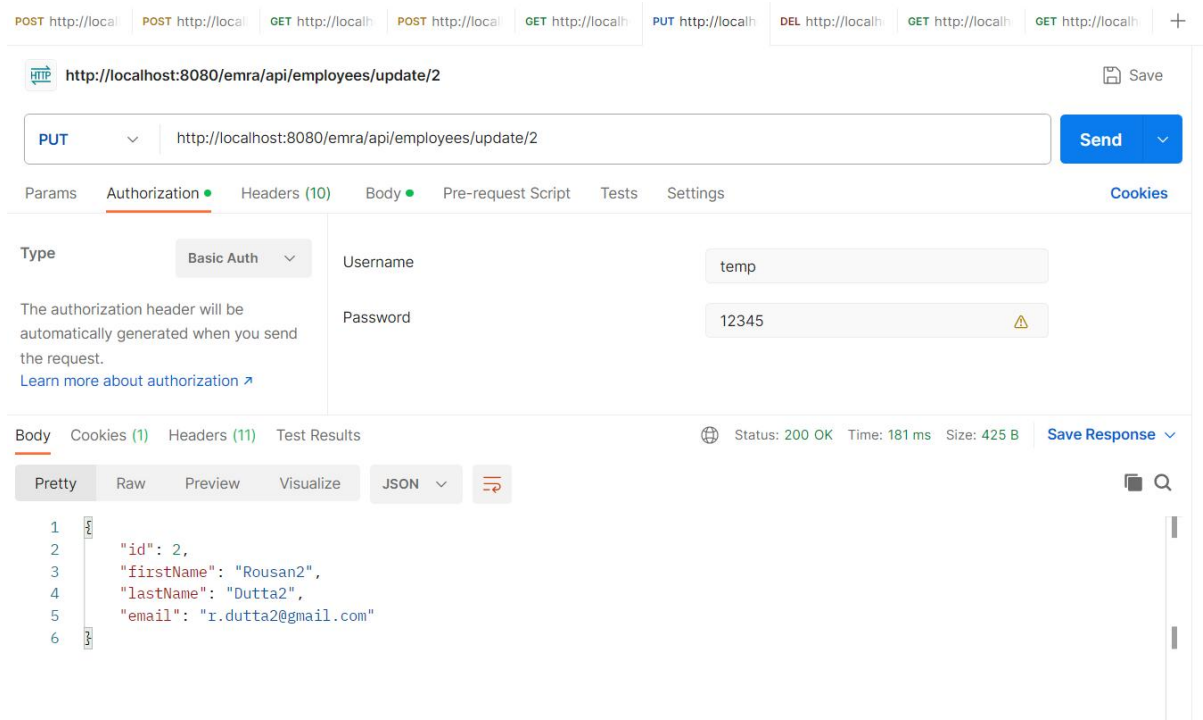


Pic: After the trying to update an employee having id=2, by the User (temp1, normal user) **POSTMAN.**

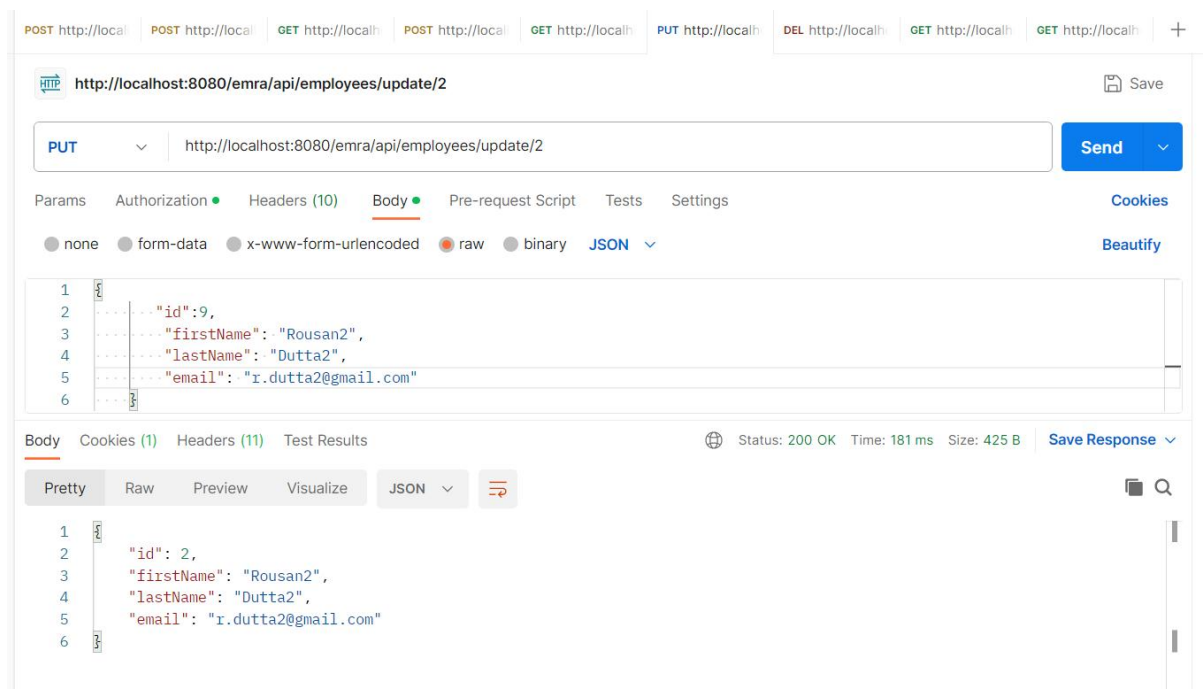


Pic: After the trying to update an employee having id=2, by the User (temp1, normal user) **POSTMAN.**





Pic: After the trying to update an employee having id=2, by the User (temp, admin user)  
POSTMAN.



Pic: After the trying to update an employee having id=2, by the User (temp, admin user)  
POSTMAN.

The screenshot shows the SQL Developer interface with a query window titled 'SQL File 8\*'. The query contains the following SQL commands:

```

1 • create database emra_db;
2 • use emra_db;
3 • select * from employees;
4 • select * from users;
5 • select * from roles;
6 • select * from users_roles;
7 • describe employees;
8 • describe users;
9 • truncate table employees;
10 • truncate table roles;

```

Below the query window, the 'Result Grid' is displayed, showing the output of the 'select \* from employees;' query. The grid has columns for 'id', 'email\_id', 'first\_name', and 'last\_name'. The data is as follows:

id	email_id	first_name	last_name
1	d.pal@gmail.com	Debesh	Pal
2	r.dutta2@gmail.com	Rousan2	Dutta2
3	l.soren@gmail.com	Lalu	Soren
7	a.paul@gmail.com	Atanu	Paul
12	z.Pavlov@gmail.com	Zihan	Pavlov
15	p.naq@gmail.com	Prabhas	Naq

The status bar at the bottom indicates 'employees 29 x' and includes 'Apply' and 'Revert' buttons.

Pic: After the trying to update an employee having id=2, by the User (temp, admin user) the DB.

The screenshot shows the SQL Developer interface with a query window titled 'SQL File 8\*'. The query contains the following SQL commands:

```

1 • create database emra_db;
2 • use emra_db;
3 • select * from employees;
4 • select * from users;
5 • select * from roles;
6 • select * from users_roles;
7 • describe employees;
8 • describe users;
9 • truncate table employees;
10 • truncate table roles;

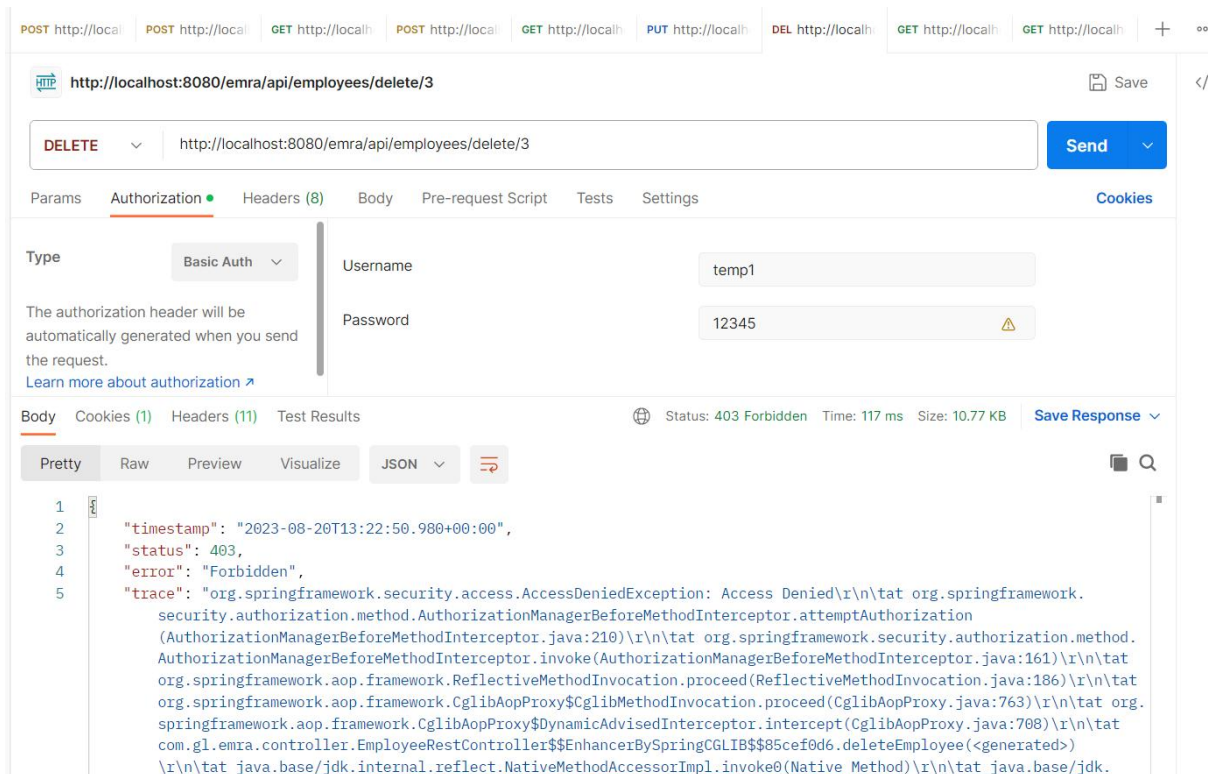
```

Below the query window, the 'Result Grid' is displayed, showing the output of the 'select \* from employees;' query. The grid has columns for 'id', 'email\_id', 'first\_name', and 'last\_name'. The data is as follows:

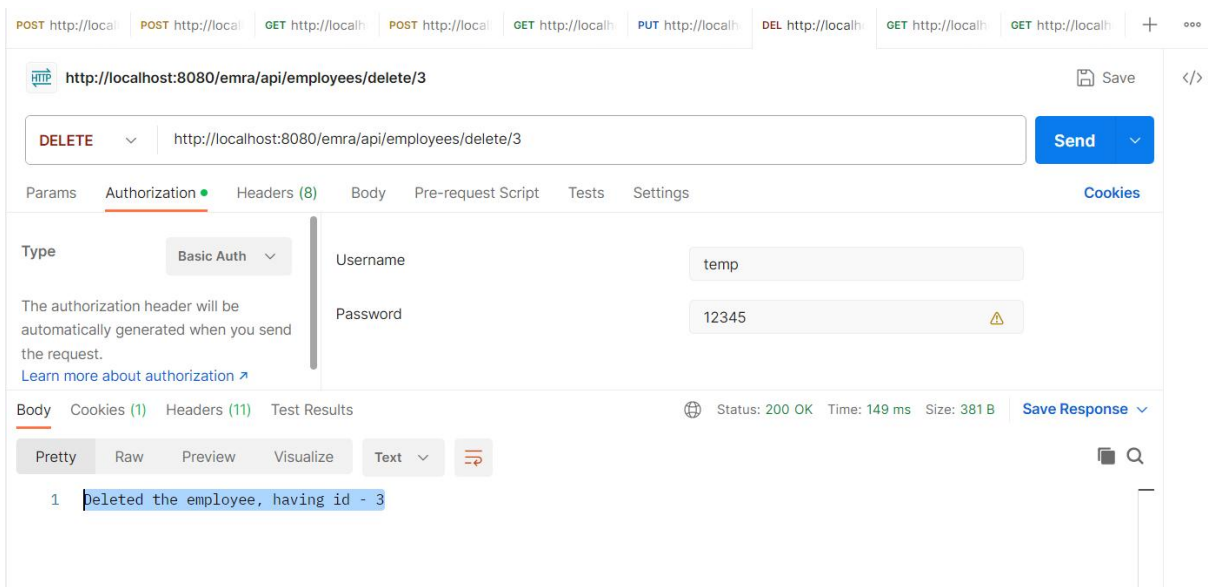
id	email_id	first_name	last_name
1	d.pal@gmail.com	Debesh	Pal
2	r.dutta2@gmail.com	Rousan2	Dutta2
3	l.soren@gmail.com	Lalu	Soren
7	a.paul@gmail.com	Atanu	Paul
12	z.Pavlov@gmail.com	Zihan	Pavlov
15	p.naq@gmail.com	Prabhas	Naq

The status bar at the bottom indicates 'employees 29 x' and includes 'Apply' and 'Revert' buttons. An 'Output' window is visible at the bottom of the interface.

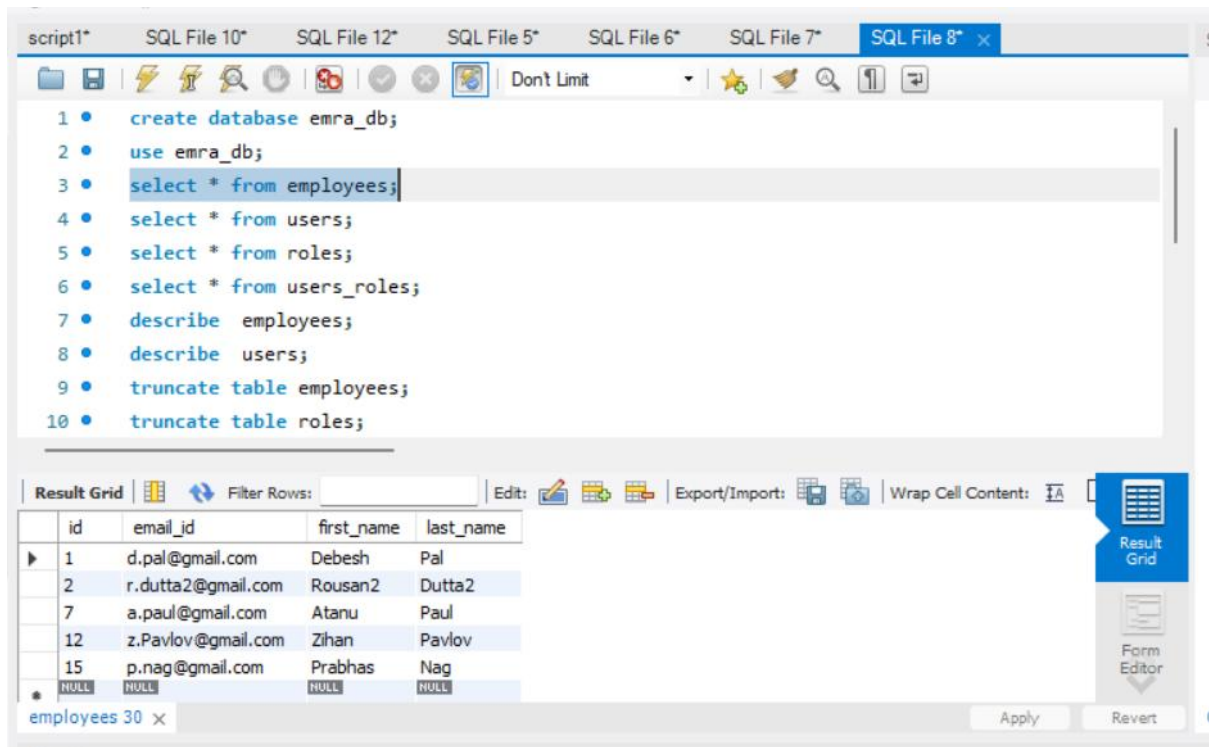
Pic: Before Deletion of an employee having id 3 by the User (temp, and temp1), the DB.



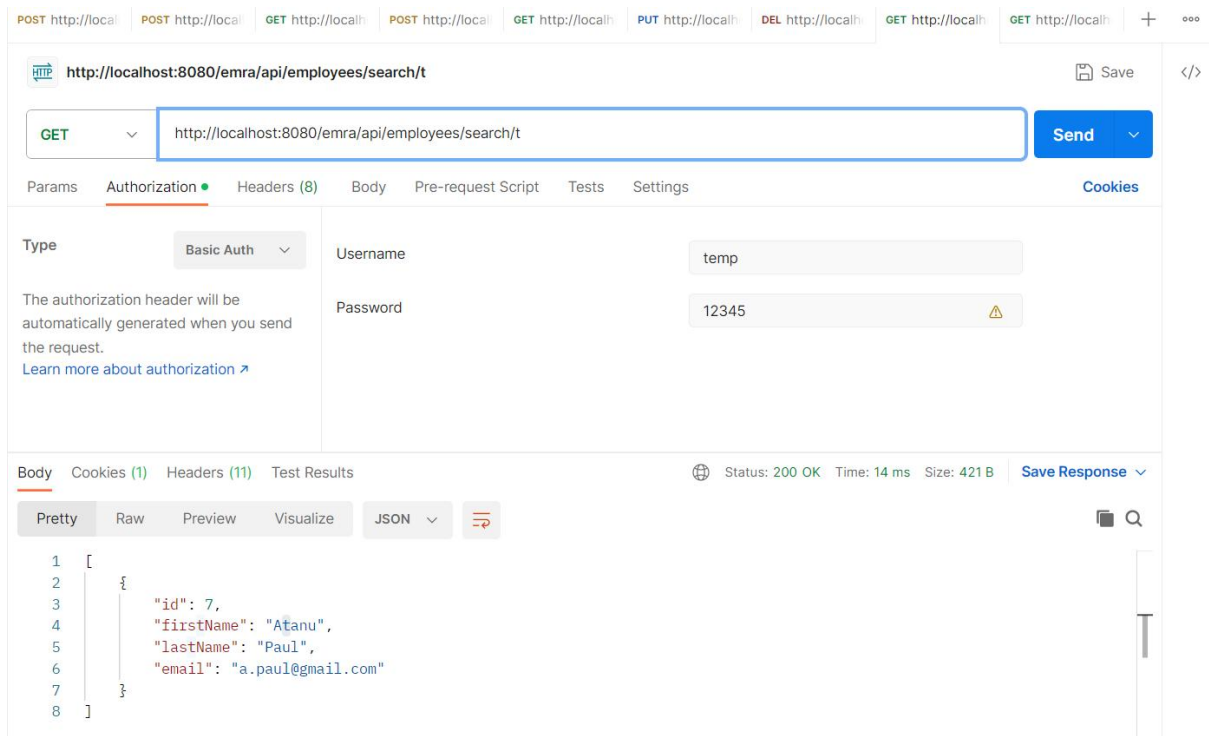
Pic: After the trying to delete an employee having id=3, by the User (temp1, normal user) POSTMAN.



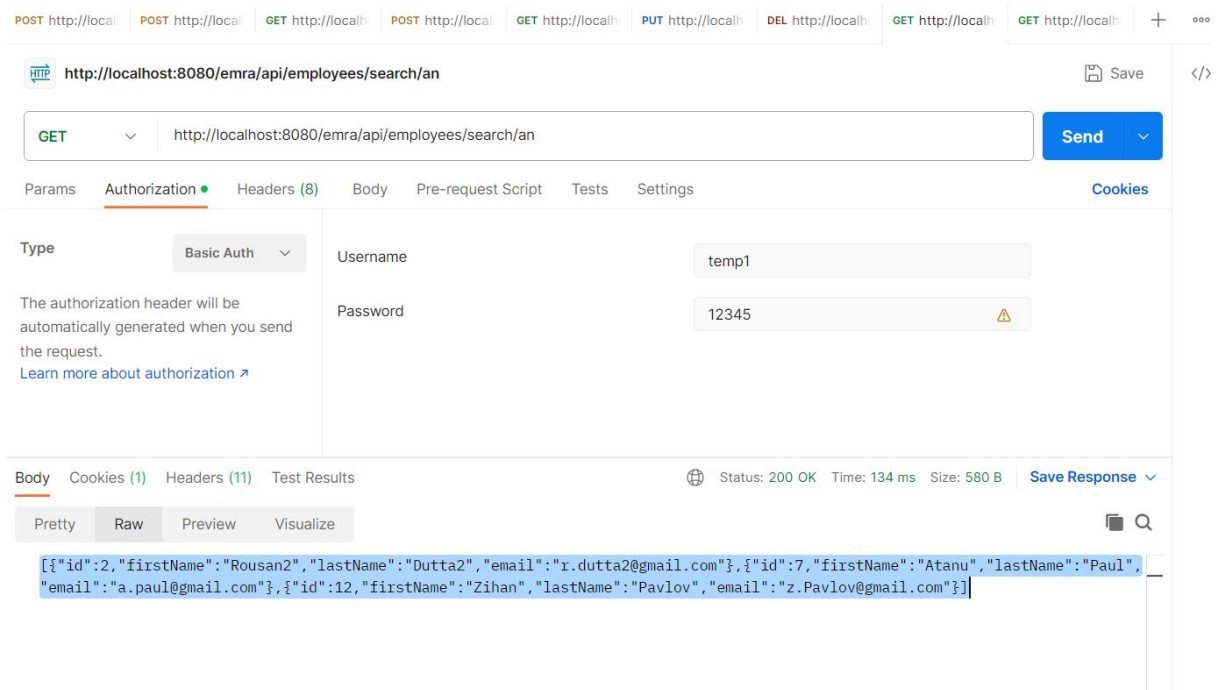
Pic: After the trying to delete an employee having id=3, by the User (temp, admin user) POSTMAN.



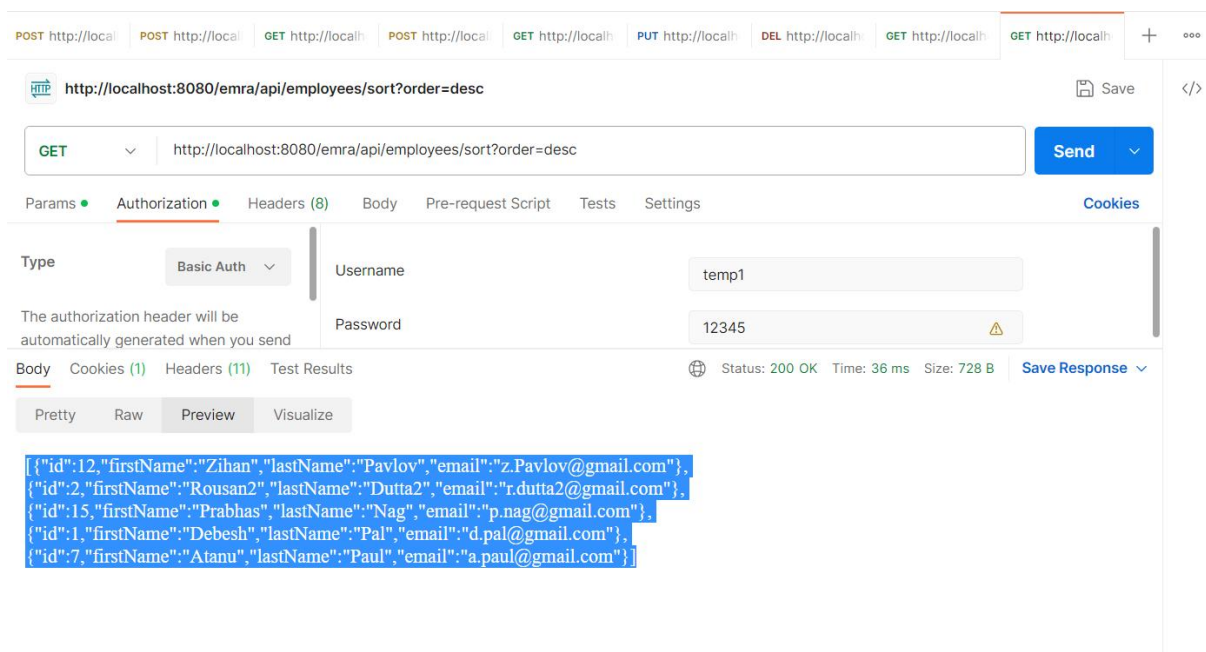
Pic: After the trying to delete an employee having id=3, by the User (temp, admin user) the DB.



Pic: After the trying to search an employee having first\_name with letter 't', by the User (temp, admin user) POSTMAN.



Pic: After the trying to search an employee having first\_name with phrase 'an', by the User (temp1, normal user) POSTMAN.



Pic: After the trying to sort in Descending Order on first\_name by the User (temp1, normal user) POSTMAN.



POST http://localhost:8080/emra/api/employees/sort?order=asc

GET http://localhost:8080/emra/api/employees/sort?order=asc

Authorization: Basic Auth

Username: temp

Password: 12345

Status: 200 OK Time: 118 ms Size: 728 B

```
[{"id":7,"firstName":"Atanu","lastName":"Paul","email":"a.paul@gmail.com"}, {"id":1,"firstName":"Debesh","lastName":"Pal","email":"d.pal@gmail.com"}, {"id":15,"firstName":"Prabhas","lastName":"Nag","email":"p.nag@gmail.com"}, {"id":2,"firstName":"Rousan2","lastName":"Dutta2","email":"r.dutta2@gmail.com"}, {"id":12,"firstName":"Zihan","lastName":"Pavlov","email":"z.Pavlov@gmail.com"}]
```

Pic: After the trying to sort in Ascending Order on first\_name by the User (temp, admin user) POSTMAN.