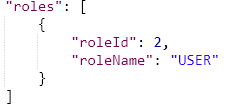
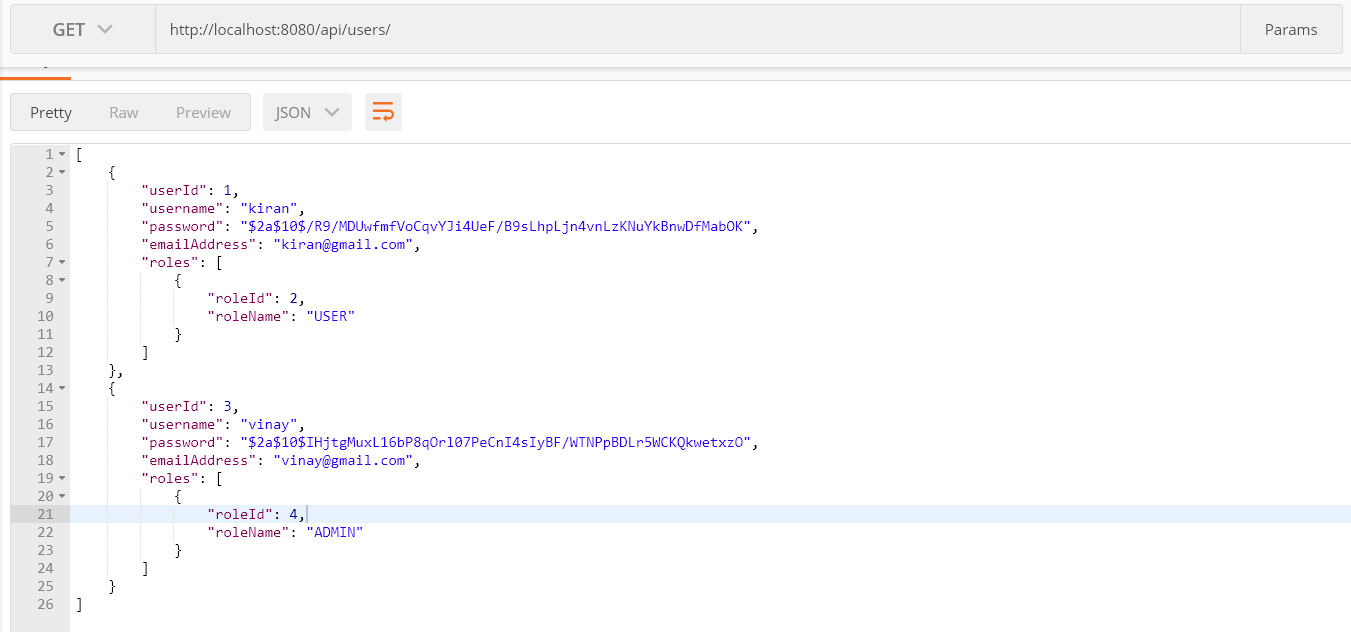
**Employee Management Rest API Based Web Application**

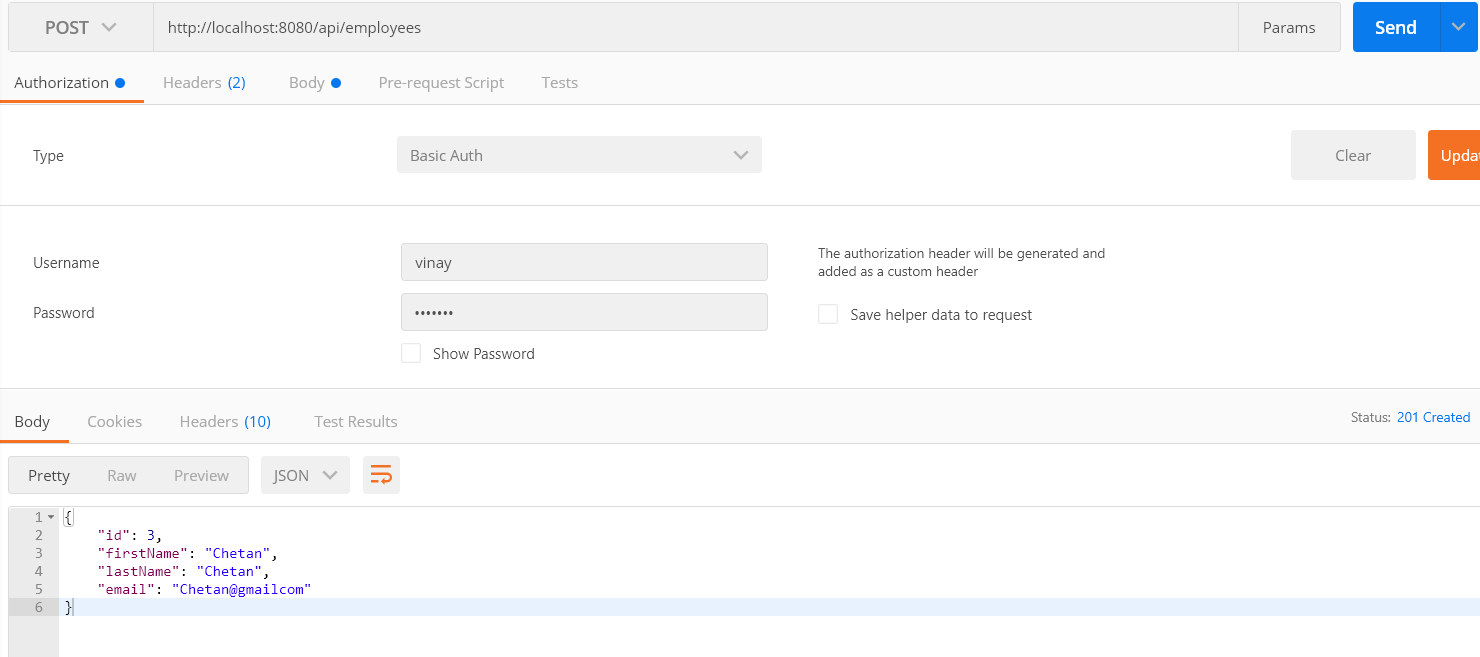
1. Adding roles dynamically in the DB



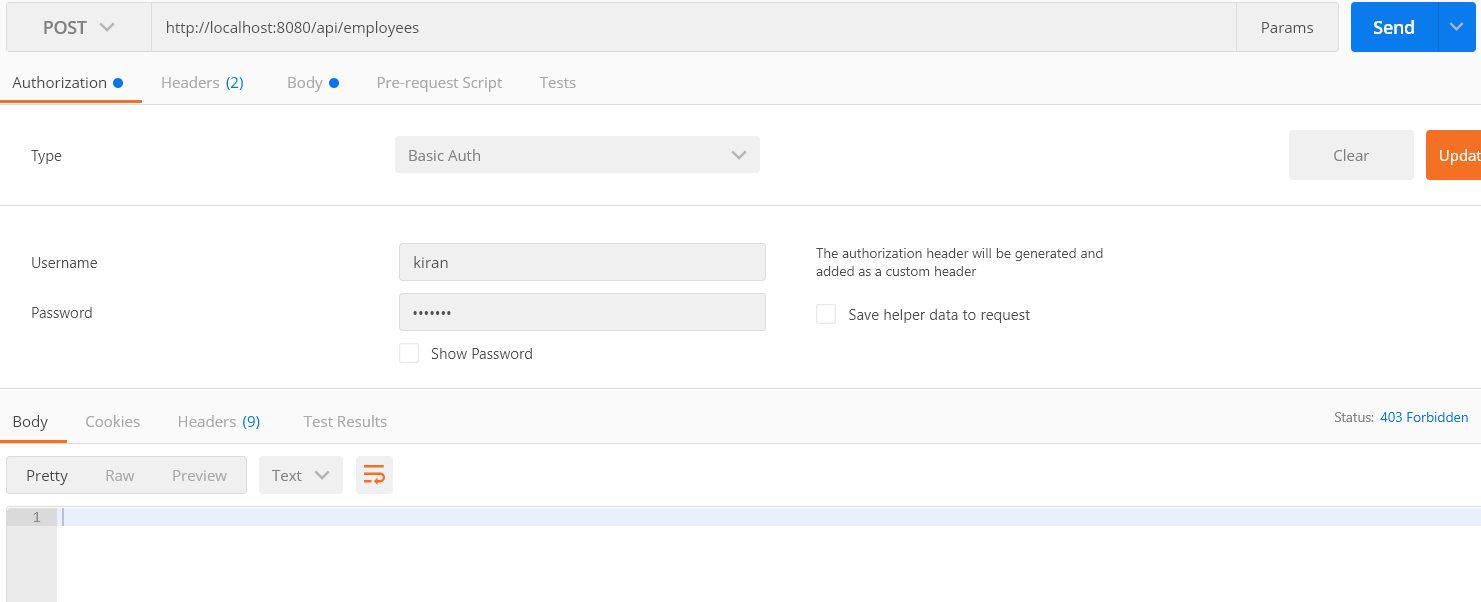
2. Adding the roles-’User(kiran) & Admin(vinay)’ in the DB, which is used for authentication purposes and authorization purposes.



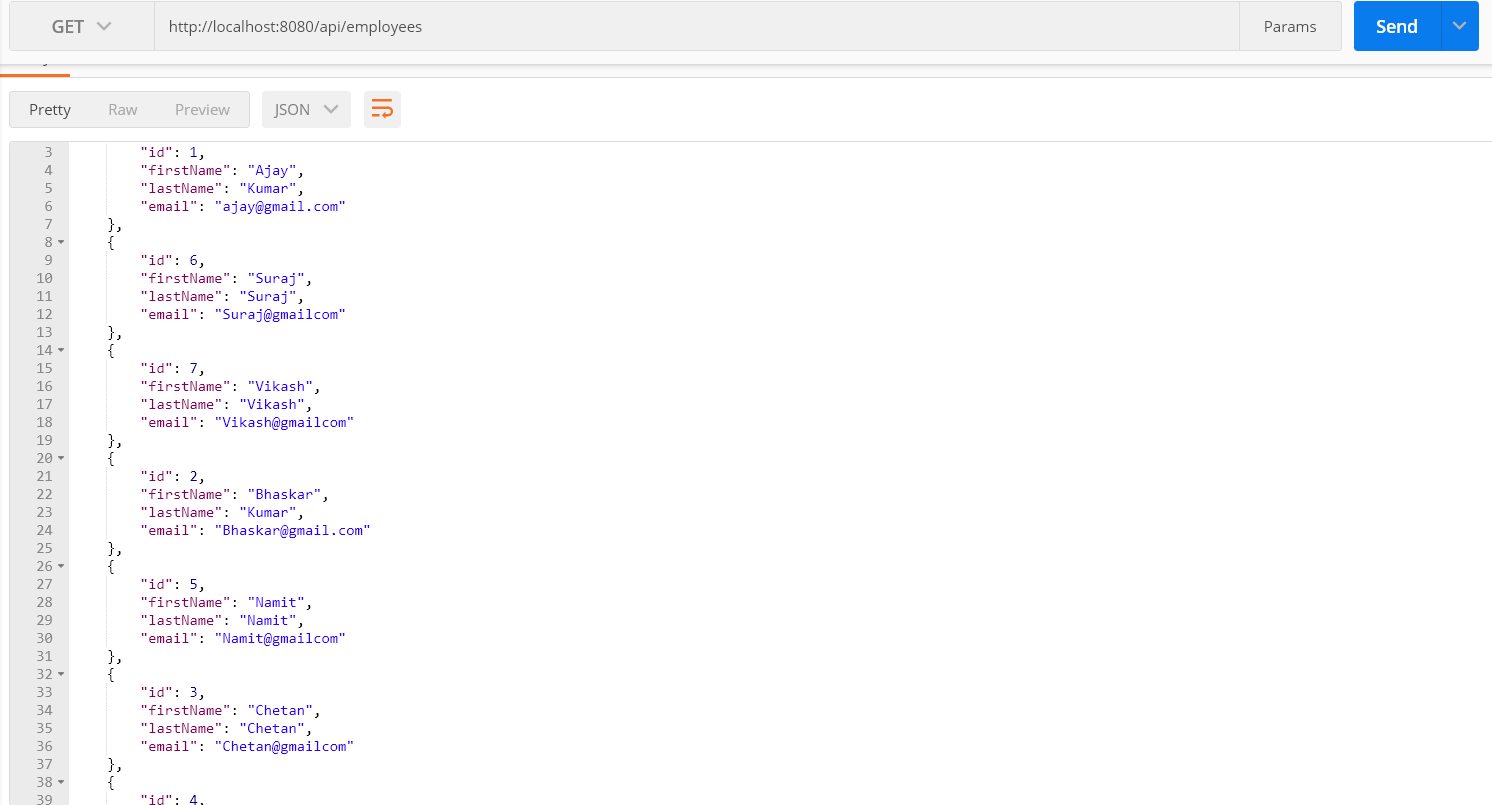
3. Adding the employees in the DB only if the authenticated user is an admin, ‘vinay’ in this case.



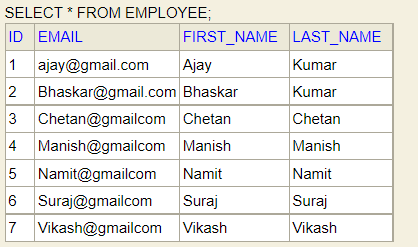
‘kiran’ who has only ‘user’ access is not authorized to add new users to the DB. Getting 403 forbidden status.



4. The endpoint URL-‘http://localhost:8080/api/employees’ gives us the list of all the employees present in the DB.

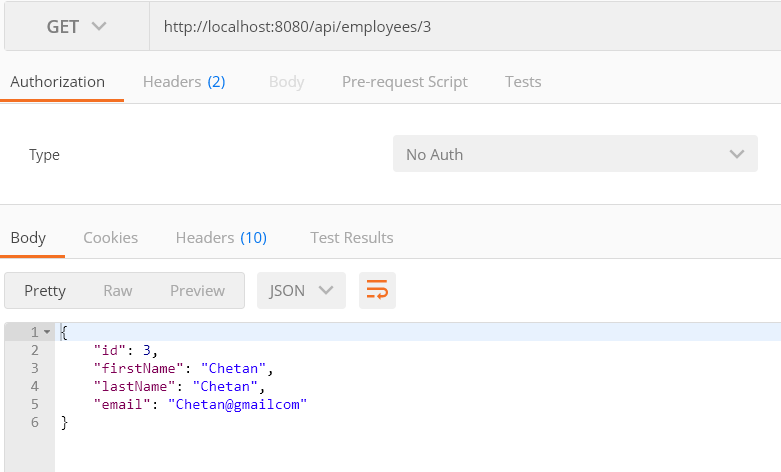


In the DB->



5. Fetch an employee based on their id ->

Url- http://localhost:8080/api/employees/{id}



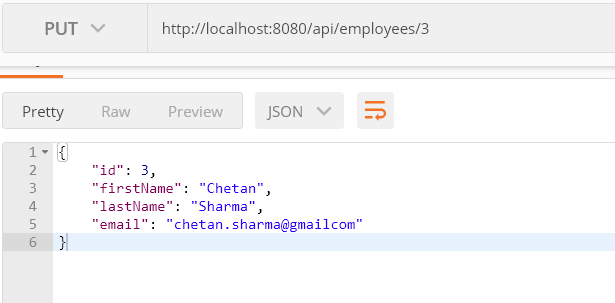
6. An endpoint to update the existing employee record using the employee id->

URL- http://localhost:8080/api/employees/{id}

Object to be updated->

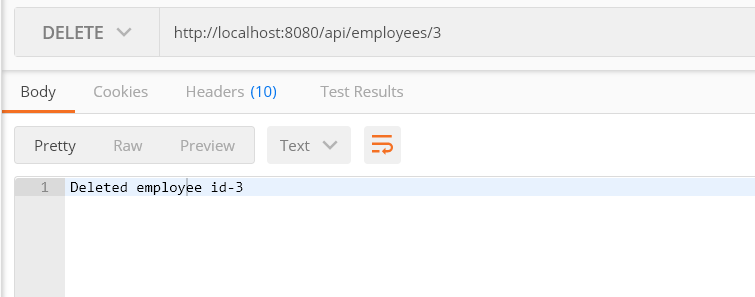


Response body after updating->

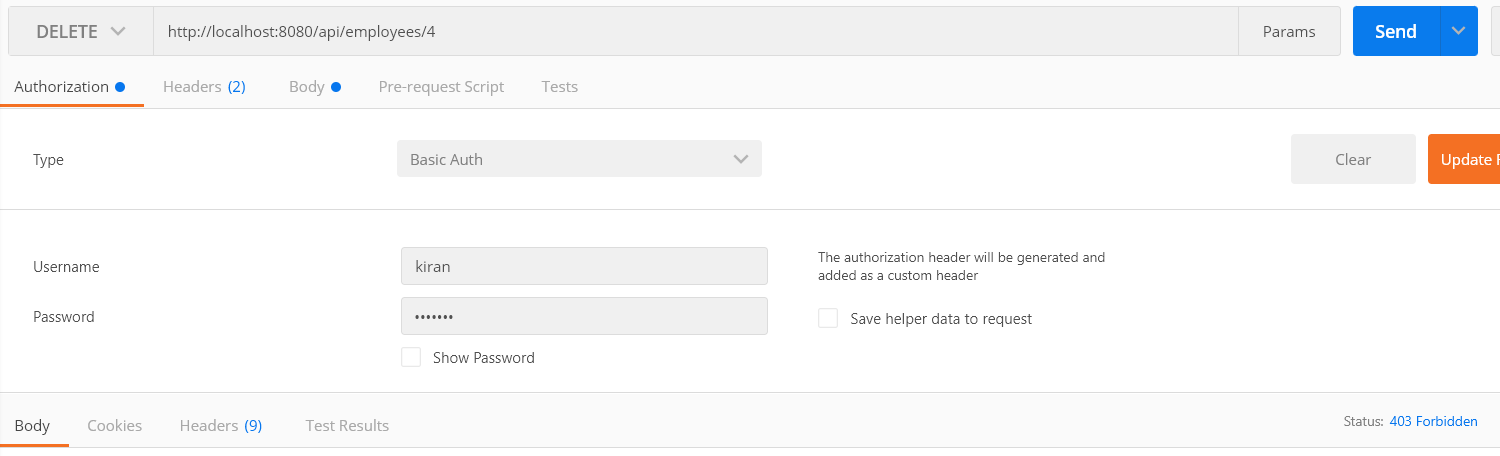


7. An endpoint to delete the existing user based on the id of the employee. Here we are deleting the user with id=3.

URL- http://localhost:8080/api/employees/{id}

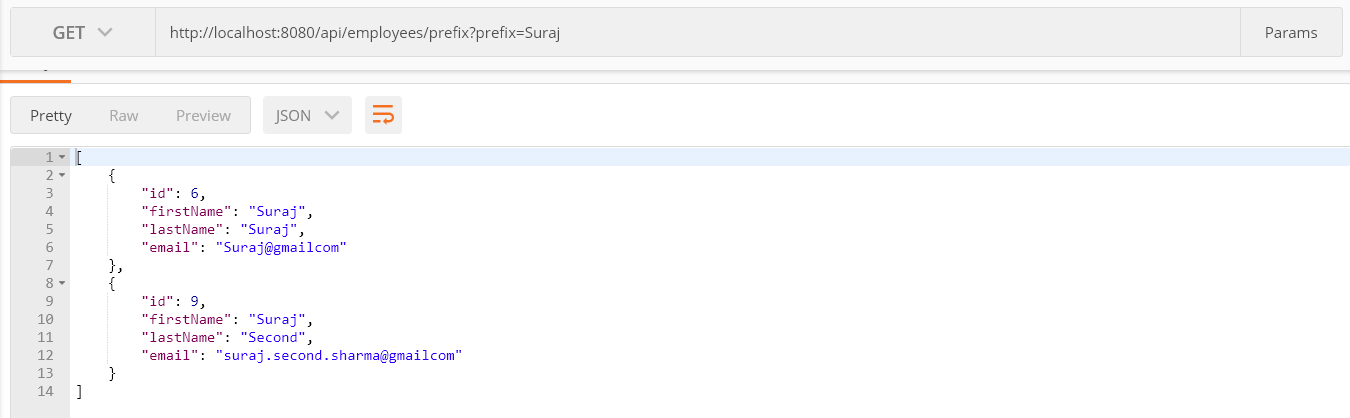


Only the ‘admin’ is allowed to perform delete/put actions. The user ‘kiran’ getting a forbidden 403 response, while trying to perform put/delete operations.



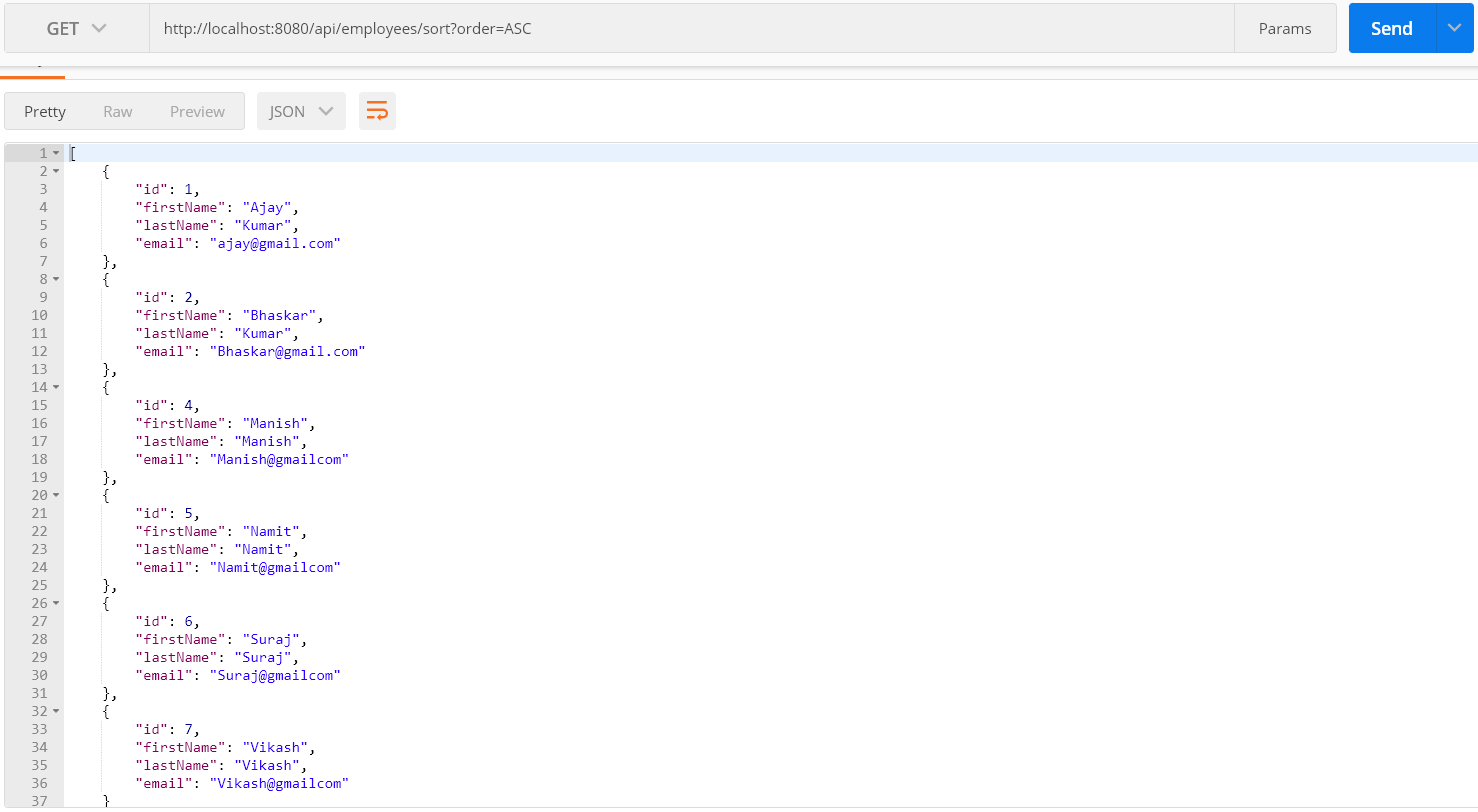
8. The endpoint to fetch an employee or list of employees using their first name. Here we are fetching the employees with the first name ‘Suraj’

URL- http://localhost:8080/api/employees/prefix?prefix=Suraj



9. The endpoint to sort the list of all employees either in ascending or descending order.

Ascending order->URL- http://localhost:8080/api/employees/sort?order=ASC



Descending order->URL- http://localhost:8080/api/employees/sort?order=DESC

