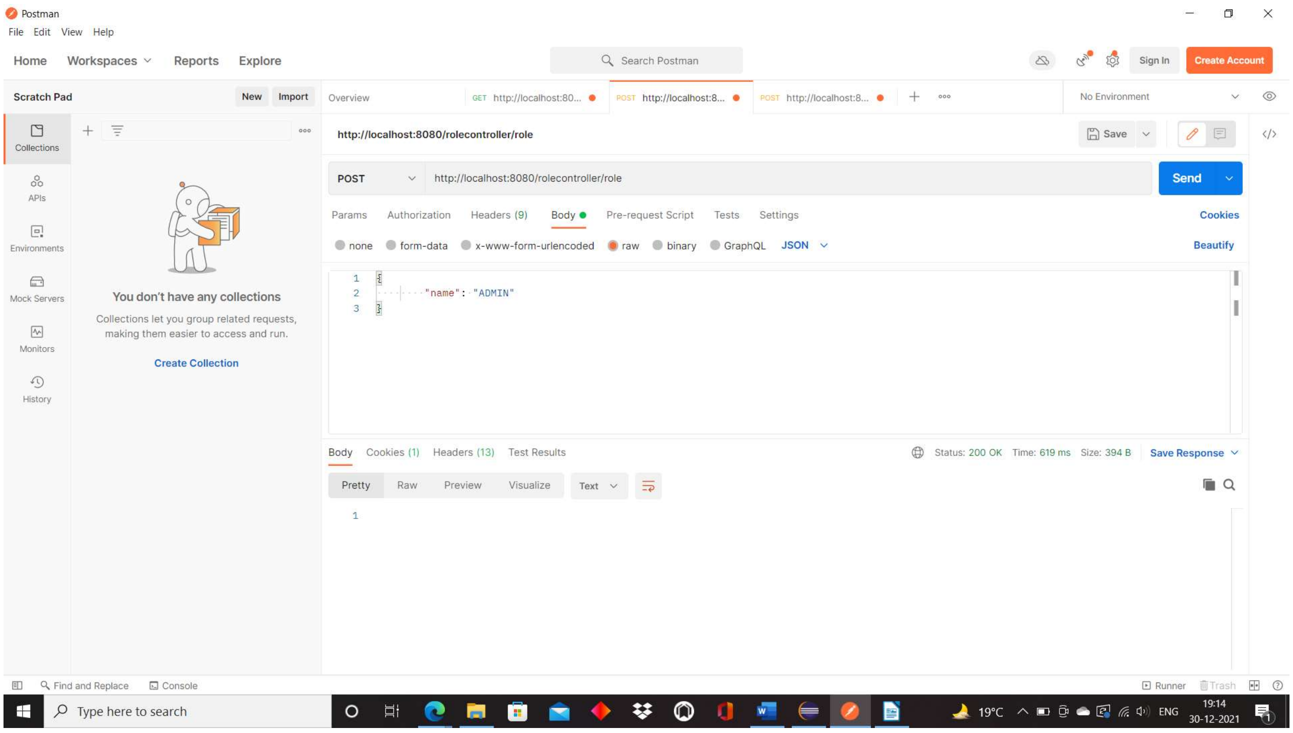
Adding Role in database:



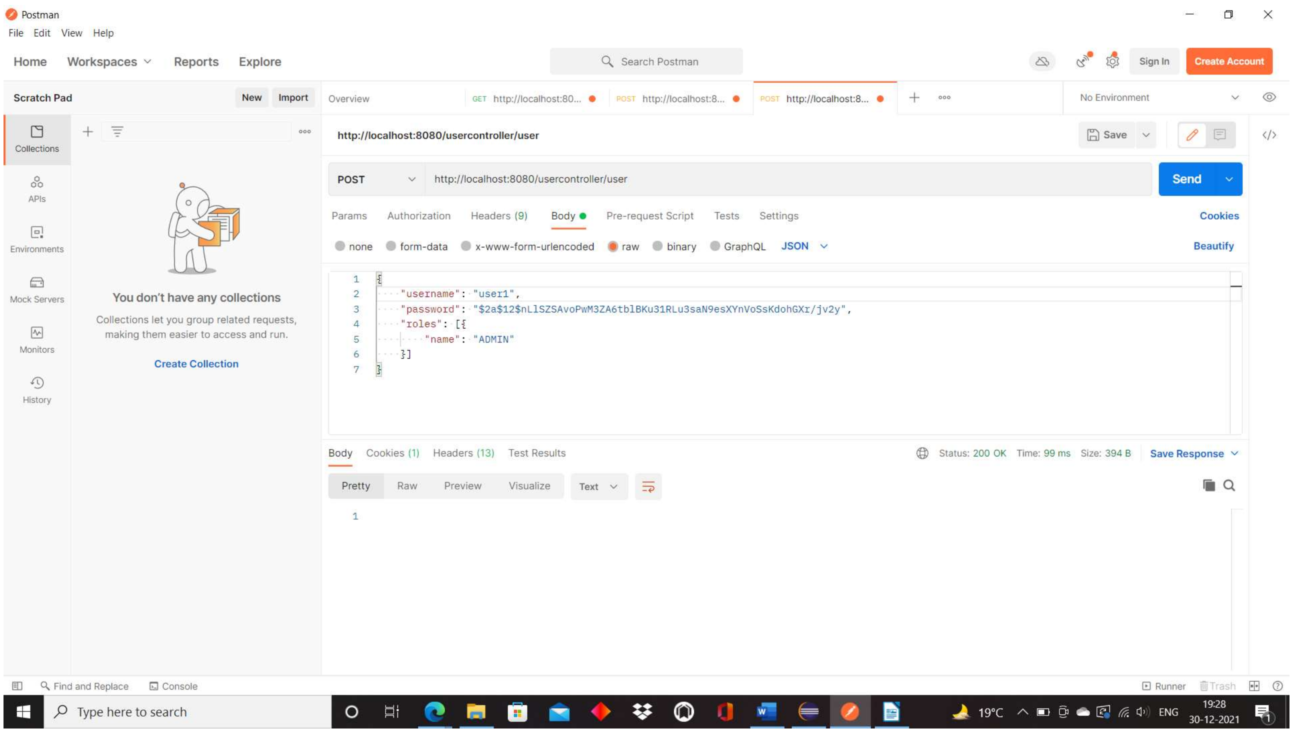
POST http://localhost:8080/rolecontroller/role

Body: raw, JSON

```
{
    "name": "ADMIN"
}
```

Status: 200 OK Time: 619 ms Size: 394 B

Adding User in database with BCrypt encrypted password (12345):



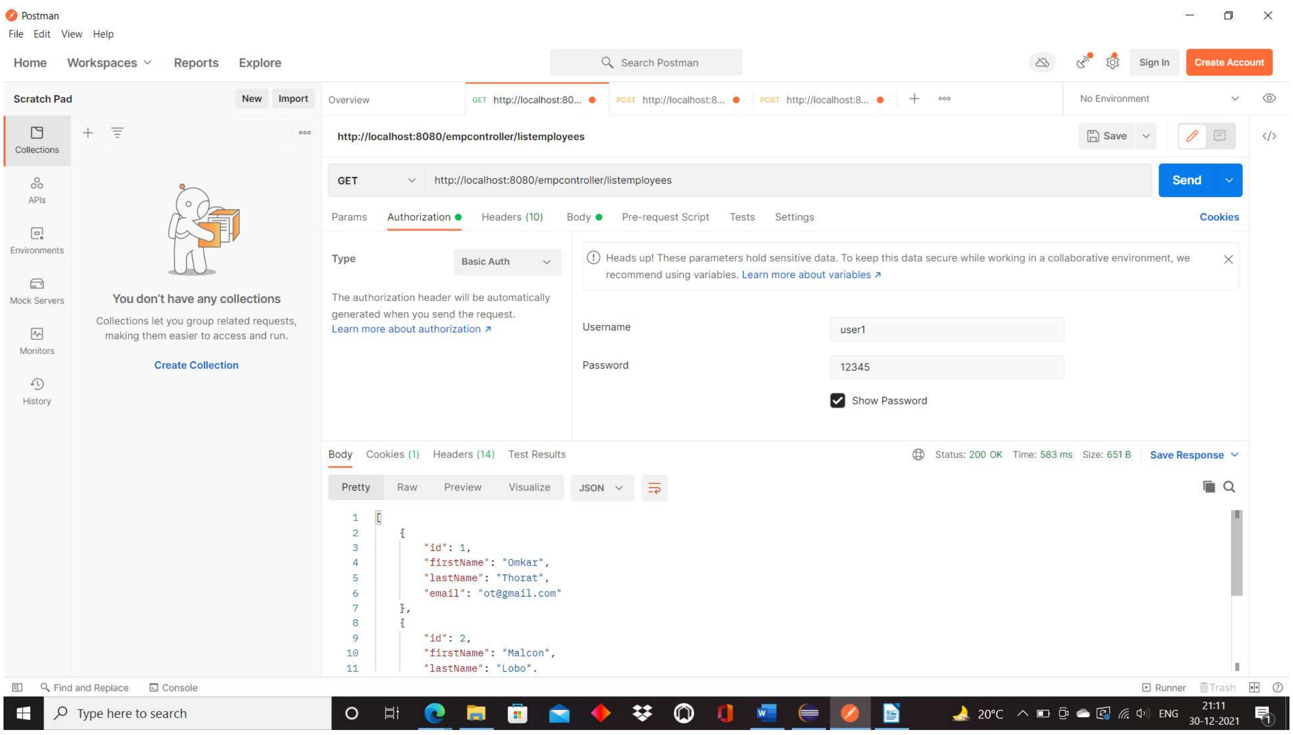
POST http://localhost:8080/usercontroller/user

Body: raw, JSON

```
{
    "username": "user1",
    "password": "$2a$12$nLlSZSAvoPwM3ZA6tblBKu31RLu3saN9esXYnVoSsKdohGXr/jv2y",
    "roles": [{
        "name": "ADMIN"
    }]
}
```

Status: 200 OK Time: 99 ms Size: 394 B

List all employees:


GET http://localhost:8080/empcontroller/listemployees

Authorization: Type Basic Auth

Username: user1

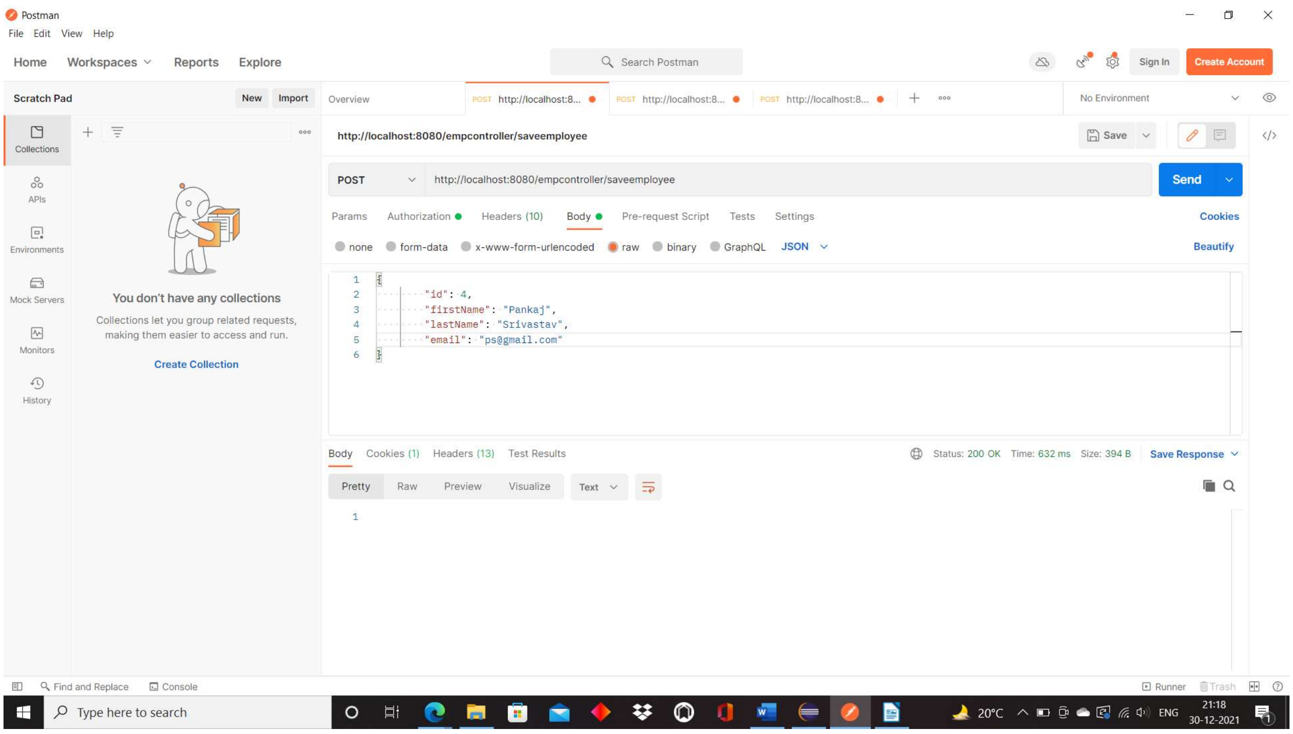
Password: 12345

Show Password

Status: 200 OK Time: 583 ms Size: 651 B

```
[
    {
        "id": 1,
        "firstName": "Omkar",
        "lastName": "Thorat",
        "email": "ot@gmail.com"
    },
    {
        "id": 2,
        "firstName": "Malcon",
        "lastName": "Lobo",
```

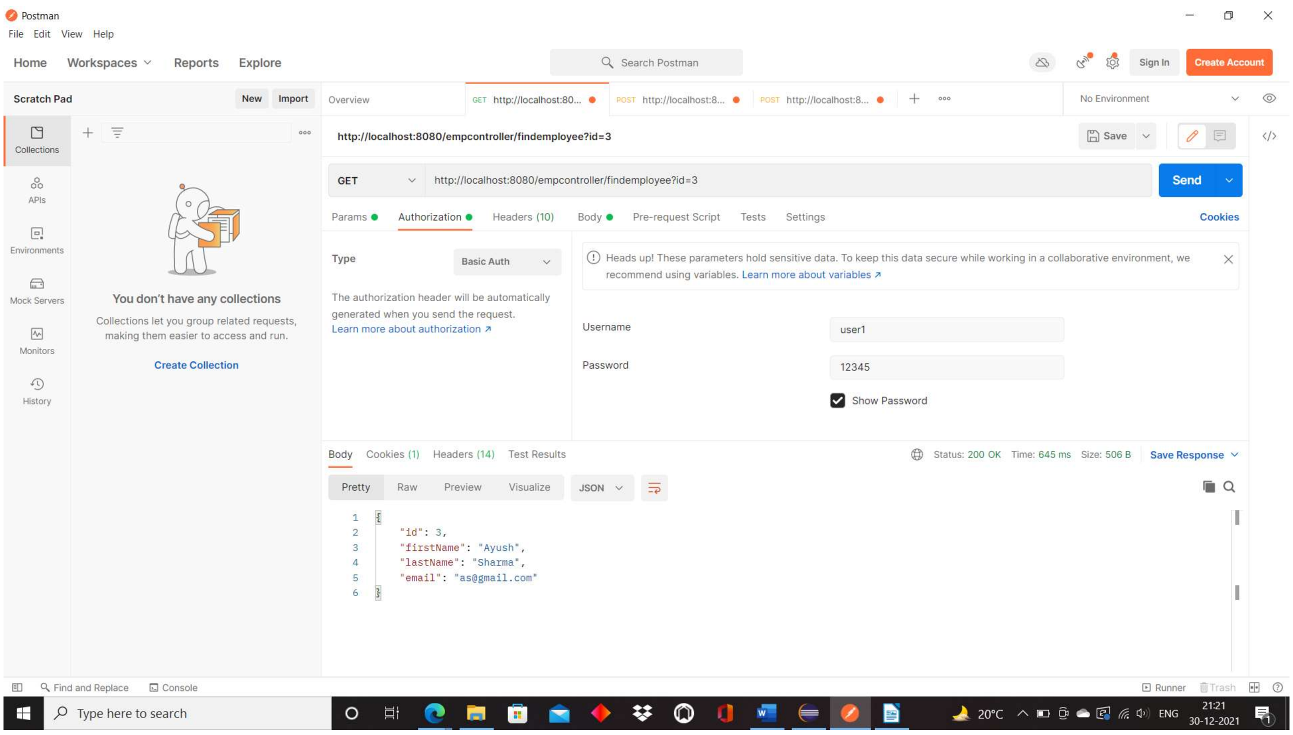
Add employee:


POST http://localhost:8080/empcontroller/saveemployee

```
{
    "id": 4,
    "firstName": "Pankaj",
    "lastName": "Srivastav",
    "email": "ps@gmail.com"
}
```

Status: 200 OK Time: 632 ms Size: 394 B

Find employee by id:



GET http://localhost:8080/empcontroller/findemployee?id=3

Authorization: Type Basic Auth

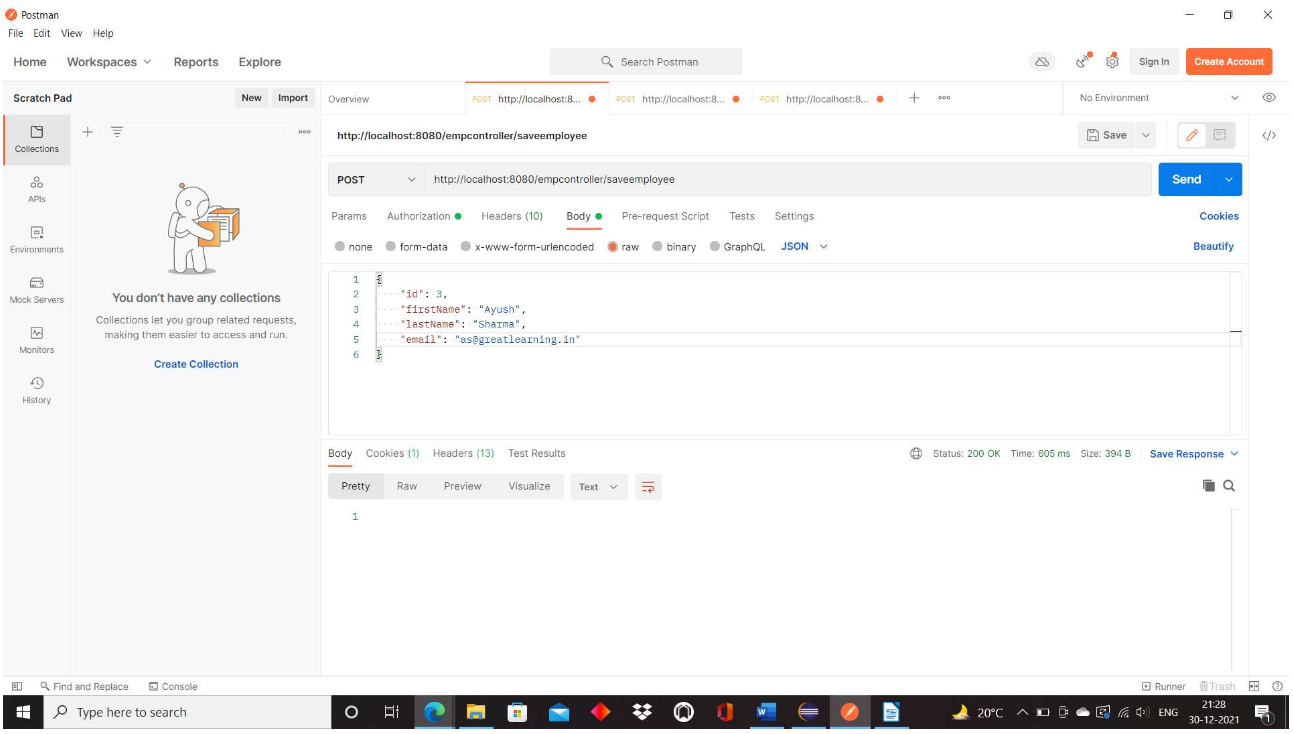
Username: user1

Password: 12345

Status: 200 OK Time: 645 ms Size: 506 B

```
{
    "id": 3,
    "firstName": "Ayush",
    "lastName": "Sharma",
    "email": "as@gmail.com"
}
```

Updating employee details (updated email):

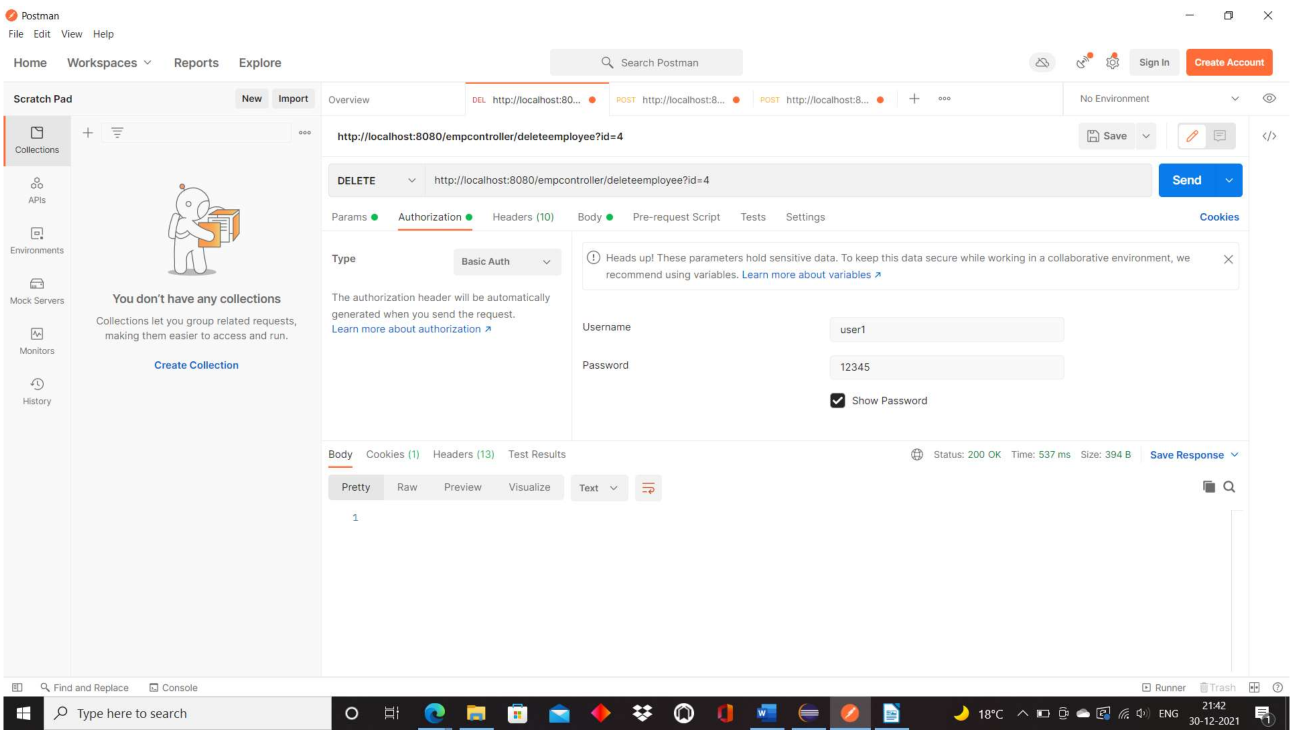


POST http://localhost:8080/empcontroller/saveemployee

```
{
    "id": 3,
    "firstName": "Ayush",
    "lastName": "Sharma",
    "email": "as@greatlearning.in"
}
```

Status: 200 OK Time: 605 ms Size: 394 B

Delete employee by id:



http://localhost:8080/empcontroller/deleteemployee?id=4

DELETE http://localhost:8080/empcontroller/deleteemployee?id=4

Authorization — Type: Basic Auth

Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. Learn more about variables ↗

The authorization header will be automatically generated when you send the request. Learn more about authorization ↗

Username: user1

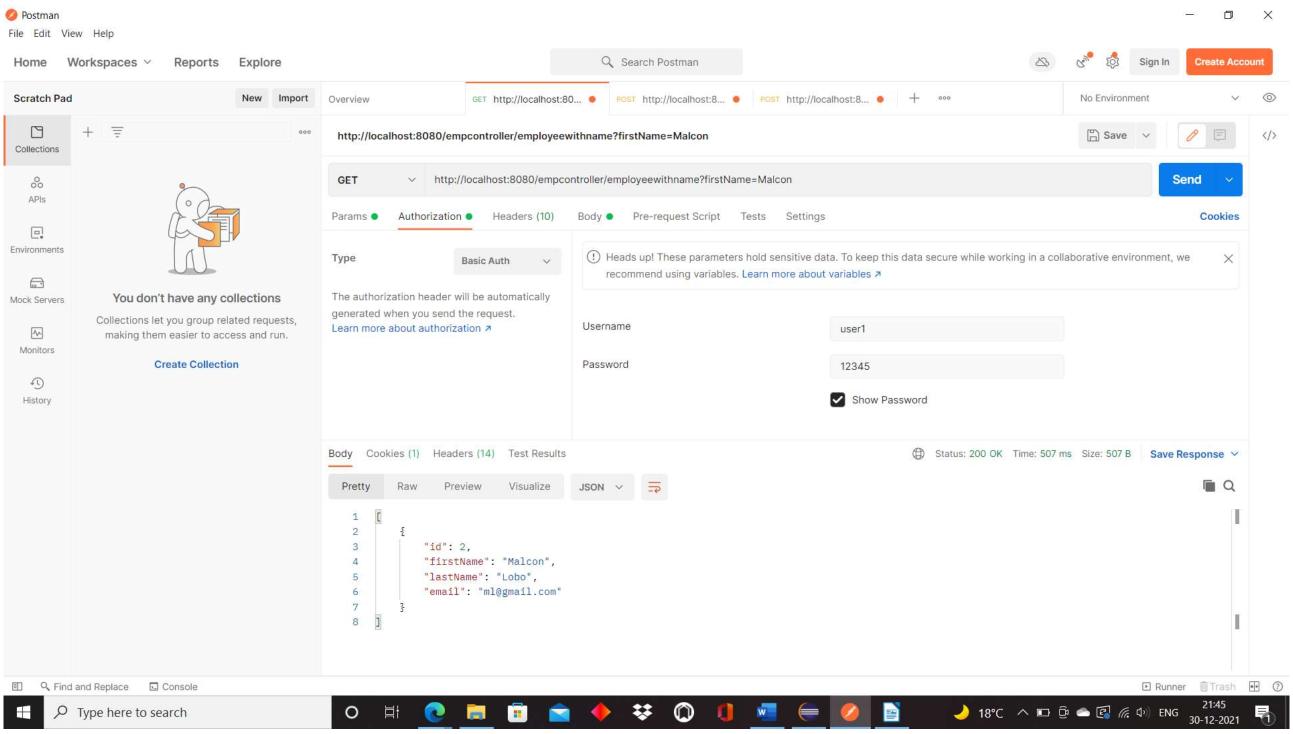
Password: 12345

Show Password

Status: 200 OK Time: 537 ms Size: 394 B

```
1
```

Fetch employee by first name:



http://localhost:8080/empcontroller/employeewithname?firstName=Malcon

GET http://localhost:8080/empcontroller/employeewithname?firstName=Malcon

Authorization — Type: Basic Auth

Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. Learn more about variables ↗

The authorization header will be automatically generated when you send the request. Learn more about authorization ↗

Username: user1

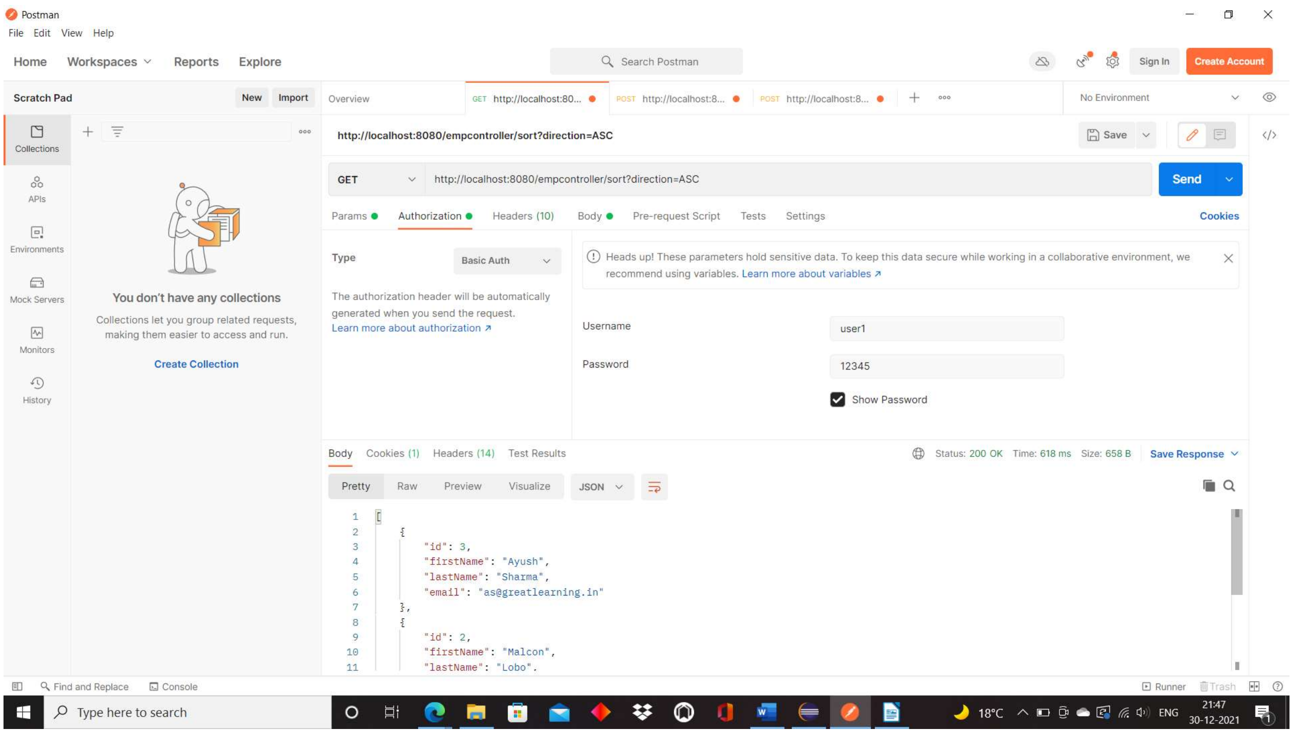
Password: 12345

Show Password

Status: 200 OK Time: 507 ms Size: 507 B

```
[
    {
        "id": 2,
        "firstName": "Malcon",
        "lastName": "Lobo",
        "email": "ml@gmail.com"
    }
]
```

List all employees with ascending/descending order of first names:



http://localhost:8080/empcontroller/sort?direction=ASC

GET http://localhost:8080/empcontroller/sort?direction=ASC

Authorization — Type: Basic Auth

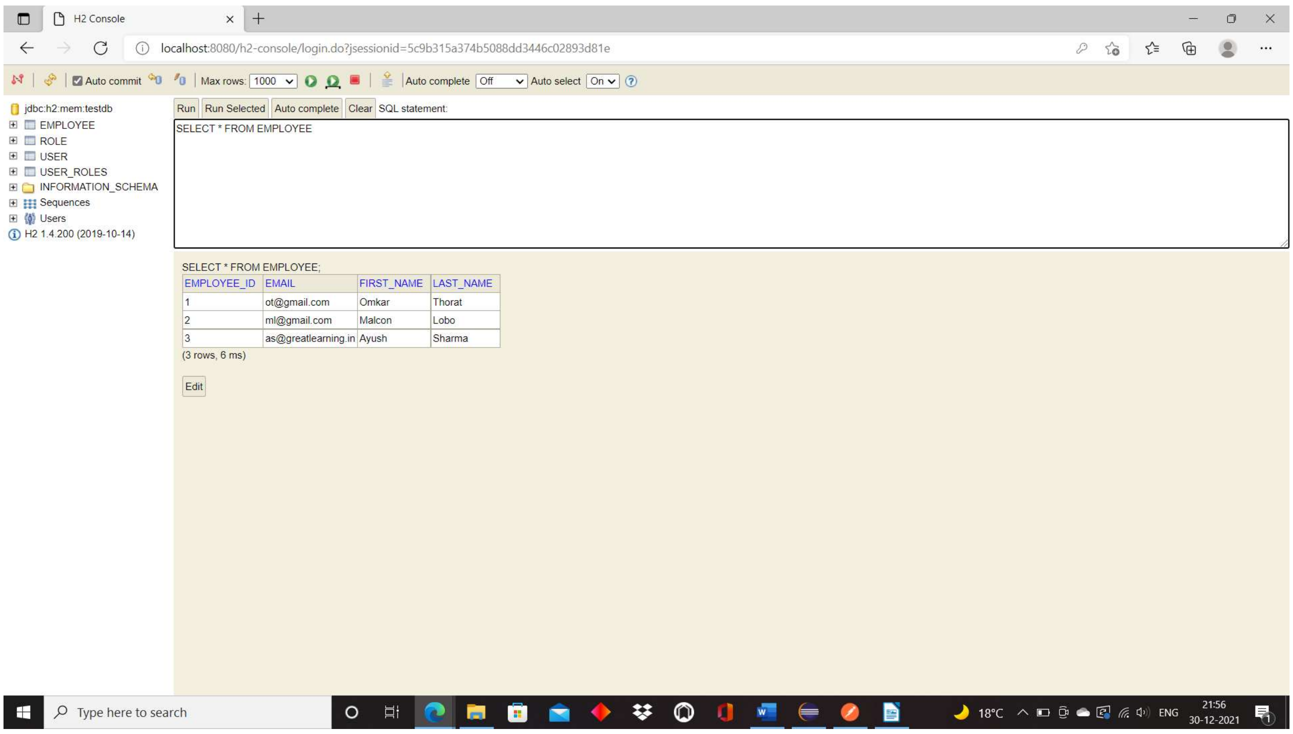
Username: user1

Password: 12345

Status: 200 OK Time: 618 ms Size: 658 B

```
[
    {
        "id": 3,
        "firstName": "Ayush",
        "lastName": "Sharma",
        "email": "as@greatlearning.in"
    },
    {
        "id": 2,
        "firstName": "Malcon",
        "lastName": "Lobo",
```

Database after all operations:



localhost:8080/h2-console/login.do?jsessionid=5c9b315a374b5088dd3446c02893d81e

```
SELECT * FROM EMPLOYEE
```

SELECT * FROM EMPLOYEE;

EMPLOYEE_ID	EMAIL	FIRST_NAME	LAST_NAME
1	ot@gmail.com	Omkar	Thorat
2	ml@gmail.com	Malcon	Lobo
3	as@greatlearning.in	Ayush	Sharma

(3 rows, 6 ms)