

4주차 과제 제출-윤주은

1. Employee 클래스 생성, POST메서드

The screenshot shows a REST client interface. The method is POST and the URL is http://localhost:8080/employee. The Content-Type header is set to application/json. The request body is a JSON object:

```
{  "id" : 1,  "name" : "멋사1",  "department" : "백엔드",  "age" : 20}
```

. The response status is 200.

Response

200

2.

The screenshot shows a database client interface. The SQL statement is `SELECT * FROM EMPLOYEE;`. The results are displayed in a table with 4 columns: AGE, ID, DEPARTMENT, and NAME. There are 3 rows of data.

AGE	ID	DEPARTMENT	NAME
20	1	백엔드	멋사1
21	2	프론트엔드	멋사2
50	3	기획디자인	멋사3

(3 rows, 3 ms)

2-1.

The screenshot shows the SQL Developer interface. On the left, the database schema is visible, including tables like EMPLOYEE, STUDENT, and INFORMATION_SCHEMA. The main window displays the following SQL statement:

```
UPDATE employee SET department = '백엔드' WHERE department = '프론트엔드';
```

Below the statement, the result of the query is shown as a table:

AGE	ID	DEPARTMENT	NAME
20	1	백엔드	멋사1
21	2	백엔드	멋사2
50	3	기획디자인	멋사3

The result is summarized as "(3 rows, 1 ms)". An "Edit" button is visible at the bottom.

2-2.

The screenshot shows the SQL Developer interface. The main window displays the following SQL statement:

```
DELETE FROM employee WHERE age = 50;
```

Below the statement, the result of the query is shown as a table:

AGE	ID	DEPARTMENT	NAME
20	1	백엔드	멋사1
21	2	백엔드	멋사2

The result is summarized as "(2 rows, 0 ms)". An "Edit" button is visible at the bottom.

2-3. insert into employee(id, name, department, age) values(4, '멋사4', '프론트', 23);

The screenshot shows the SQL Developer interface. The main window displays the following SQL statement:

```
SELECT * FROM EMPLOYEE;
```

Below the statement, the result of the query is shown as a table:

AGE	ID	DEPARTMENT	NAME
20	1	백엔드	멋사1
21	2	백엔드	멋사2
23	4	프론트	멋사4

The result is summarized as "(3 rows, 0 ms)". An "Edit" button is visible at the bottom.