

[2-3. SQL 기본 문법]

1. SELECT

1-1. SELECT문 개요

- SELECT문은 SQL에서 가장 기본적이고 자주 사용되는 명령어로, 데이터베이스에서 특정 데이터를 조회하는 데 사용됩니다.
- 다양한 옵션과 함께 사용되어 복잡한 쿼리를 구성할 수 있으며, 필요한 정보를 정확하고 효율적으로 추출할 수 있습니다.

1-2. 기본 구조

```
SELECT 열이름1, 열이름2, ...  
FROM 테이블이름  
WHERE 조건;
```

- SELECT : 조회하고자 하는 열(컬럼)의 이름을 지정합니다. 모든 열을 선택하려면 '*'를 사용합니다.
- FROM : 조회할 데이터가 있는 테이블의 이름을 지정합니다.
- WHERE : 조회할 데이터의 조건을 지정합니다.

사용 예시 1

```
USE room_sales; <- USE 구문 : 먼저 사용할 데이터베이스를 지정  
SELECT *  
FROM Sales  
WHERE sale_date="2024-01-03";
```

사용 예시 2

```
USE room_sales;  
SELECT sale_id, room_number, sale_price  
FROM Sales  
WHERE sale_date="2024-01-03";
```

사용 예시 3

```
SELECT *  
FROM Sales;
```

1-3. ORDER BY, LIMIT 구문

```
SELECT 열이름1, 열이름2, ...  
FROM 테이블이름  
WHERE 조건  
ORDER BY 열이름 [ASC | DESC]  
LIMIT 숫자;
```

- ORDER BY : SELECT 결과 데이터를 특정 열의 값에 따라 정렬합니다. 'ASC'는 오름차순(기본값), 'DESC'는 내림차순을 의미합니다.
- LIMIT : 반환되는 행의 수를 제한합니다. 큰 데이터셋에서 일부 데이터만 빠르게 조회할 때 유용합니다.

사용 예시 1

```
SELECT *  
FROM Sales  
WHERE sale_date="2024-01-03"  
LIMIT 3;
```

사용 예시 2

```
SELECT sale_date, room_number, sale_price  
FROM Sales  
WHERE room_number="101"  
ORDER BY sale_date DESC;
```

사용 예시 3

```
SELECT *  
FROM Sales  
WHERE sale_price >= 110000  
ORDER BY room_number ASC  
LIMIT 10;
```

2. INSERT

2-1. INSERT문 개요

- INSERT문은 데이터를 새로운 행으로 테이블에 추가하는 데 사용됩니다.
- 삽입하려는 테이블의 데이터 구조(열의 데이터 타입, NOT NULL 등의 제약 조건)를 잘 이해하고, 그에 맞춰서 구문을 사용해야 합니다.

2-2. 기본 구조

```
INSERT INTO 테이블명 (열1, 열2, ..., 열N)
VALUES (값1, 값2, ..., 값N);
```

- 테이블명 : 데이터를 삽입할 대상 테이블의 이름
- (열1, 열2, ..., 열N) : 데이터를 삽입할 열(컬럼)의 이름을 지정
- VALUES : 다음에 오는 괄호 안에 삽입할 데이터 값을 순서대로 나열

사용 예시 1

```
INSERT INTO Sales (room_number, sale_date, sale_price)
VALUES ('101', '2024-01-15', 100000);
```

사용 예시 2

```
INSERT INTO Sales (room_number, sale_date, sale_price)
VALUES
('101', '2024-01-15', 100000),
('102', '2024-01-15', 150000),
('103', '2024-01-15', 170000),
('201', '2024-01-15', 110000),
('202', '2024-01-15', 160000),
('203', '2024-01-15', 180000);
```

3. UPDATE

3-1. UPDATE문 개요

- UPDATE문은 데이터베이스 내의 기존 행의 데이터를 수정할 때 사용됩니다.
- 주로 WHERE 절과 함께 사용하여 특정 조건을 만족하는 행만을 업데이트합니다.

3-2. 기본 구조

```
UPDATE 테이블이름  
SET 열이름1 = 값1, 열이름2 = 값2, ...  
WHERE 조건;
```

- 테이블이름 : 수정하고자 하는 데이터가 있는 테이블의 이름
- SET : 변경할 열과 해당 열에 할당할 새로운 값을 지정. 여러 열을 수정할 경우 쉼표(,)로 구분
- WHERE : 수정할 레코드를 지정하는 조건을 정의.

사용 예시 1

```
UPDATE Sales  
SET sale_price = 110000  
WHERE sale_id = 8;
```

사용 예시 2

```
UPDATE Sales  
SET sale_price = 110000  
WHERE  
    room_number = '201'  
    AND sale_date = '2024-01-03';
```

4. DELETE

4-1. DELETE 문 개요

- DELETE 문은 테이블의 하나 이상의 행을 제거하는 데 사용됩니다.
- 주로 WHERE 절과 함께 사용하여 특정 조건을 만족하는 행만 선택적으로 삭제합니다.
- WHERE 절을 생략하면 테이블의 모든 레코드가 삭제됩니다.

4-2. 기본 구조

```
DELETE FROM 테이블이름  
WHERE 조건;
```

- 테이블이름 : 삭제할 데이터가 있는 테이블의 이름
- WHERE : 삭제할 레코드를 지정하는 조건을 정의.

사용 예시 1

```
DELETE FROM Sales  
WHERE sale_id = 10;
```

사용 예시 2

```
DELETE FROM Sales  
WHERE  
    room_number = '103'  
    AND sale_date >= '2024-01-08';
```

데이터 삭제 확인

```
SELECT * FROM Sales WHERE room_number='103' ORDER BY  
sale_date asc;
```