

SQL 핵심 정리 - SELECT문부터 JOIN까지

1. 기본 SELECT 문법

SELECT 기본 구조

```
sql  
  
SELECT  
  컬럼명1,  
  컬럼명2,  
  컬럼명3  
FROM  
  테이블명  
WHERE  
  조건식  
ORDER BY  
  정렬기준;
```

실행 순서 ★

1. FROM → 2. WHERE → 3. GROUP BY → 4. HAVING → 5. SELECT → 6. ORDER BY

주요 연산자

- 비교연산자: `>`, `<`, `>=`, `<=`, `=`, `!=`, `<>`
- 논리연산자: `AND`, `OR`, `NOT`
- 범위: `BETWEEN A AND B`
- 패턴매칭: `LIKE '%패턴%'` (% = 0글자 이상, _ = 1글자)
- NULL 체크: `IS NULL`, `IS NOT NULL`
- 포함: `IN ('값1', '값2', '값3')`

정렬

```
sql  
  
ORDER BY 컬럼명 ASC/DESC NULLS FIRST/LAST
```

2. 단일행 함수

문자열 함수

- `LENGTH(str)`: 글자수 반환

- `LENGTHB(str)`: 바이트수 반환
- `INSTR(str, '찾을문자')`: 위치값 반환
- `SUBSTR(str, 시작위치, 길이)`: 문자열 추출
- `LPAD/RPAD(str, 길이, '채울문자')`: 좌/우측 패딩
- `TRIM/LTRIM/RTRIM`: 공백 제거
- `UPPER/LOWER/INITCAP`: 대/소문자 변환
- `REPLACE(str, '찾을문자', '바꿀문자')`: 문자 치환

숫자 함수

- `ROUND(숫자, 자릿수)`: 반올림
- `FLOOR(숫자)`: 내림
- `CEIL(숫자)`: 올림
- `TRUNC(숫자, 자릿수)`: 절삭
- `MOD(숫자1, 숫자2)`: 나머지

날짜 함수

- `SYSDATE`: 현재 날짜
- `MONTHS_BETWEEN(날짜1, 날짜2)`: 개월수 차이
- `ADD_MONTHS(날짜, 개월수)`: 개월 더하기
- `NEXT_DAY(날짜, '요일')`: 다음 해당 요일
- `LAST_DAY(날짜)`: 해당 월의 마지막 날
- `EXTRACT(YEAR/MONTH/DAY FROM 날짜)`: 년/월/일 추출

형변환 함수

- `TO_CHAR(숫자/날짜, '포맷')`: 문자로 변환
- `TO_DATE(문자, '포맷')`: 날짜로 변환
- `TO_NUMBER(문자, '포맷')`: 숫자로 변환

NULL 처리 함수

- `NVL(컬럼, 대체값)`: NULL이면 대체값 반환
- `NVL2(컬럼, 값1, 값2)`: NULL이 아니면 값1, NULL이면 값2

조건 함수

- `DECODE(컬럼, 값1, 결과1, 값2, 결과2, 기본값)`: switch문과 유사
 - `CASE WHEN 조건1 THEN 결과1 WHEN 조건2 THEN 결과2 ELSE 기본값 END`: if문과 유사
-

3. 그룹 함수 ★

5대 그룹 함수 (필수 암기!)

1. SUM(컬럼): 합계
2. AVG(컬럼): 평균
3. MAX(컬럼): 최대값
4. MIN(컬럼): 최소값
5. COUNT(컬럼/* /DISTINCT 컬럼): 개수

주의사항

- 단일행 함수와 그룹함수는 함께 사용 불가 (결과 행수가 다름)
- GROUP BY절과 함께 사용

4. GROUP BY & HAVING

기본 문법

```
sql

SELECT
  컬럼명,
  그룹함수
FROM
  테이블명
WHERE
  조건식      -- 그룹화 전 조건
GROUP BY
  컬럼명
HAVING
  그룹함수조건  -- 그룹화 후 조건
ORDER BY
  정렬기준;
```

핵심 규칙

- GROUP BY절에 명시된 컬럼만 SELECT 가능
 - HAVING절은 그룹함수에 대한 조건 처리
-

5. JOIN ★

오라클 전용 구문

```
sql

SELECT
    컬럼들
FROM
    테이블1, 테이블2
WHERE
    테이블1.컬럼 = 테이블2.컬럼;
```

ANSI 표준 구문 (권장)

```
sql

-- 컬럼명이 다를 때 (ON 사용)
SELECT
    컬럼들
FROM
    테이블1
INNER JOIN
    테이블2 ON (테이블1.컬럼 = 테이블2.컬럼);

-- 컬럼명이 같을 때 (USING 사용)
SELECT
    컬럼들
FROM
    테이블1
INNER JOIN
    테이블2 USING(공통컬럼명);
```

JOIN 종류

- **INNER JOIN**: 일치하는 데이터만 조회
- **LEFT OUTER JOIN**: 왼쪽 테이블 기준으로 조회
- **RIGHT OUTER JOIN**: 오른쪽 테이블 기준으로 조회
- **FULL OUTER JOIN**: 모든 데이터 조회
- **CROSS JOIN**: 카테시안 곱

6. 실무 팁 💡

1. 성능을 위해 **SELECT *** 사용 금지

2. 테이블 별칭 활용으로 가독성 향상
3. JOIN시 테이블명 또는 별칭으로 컬럼 명시
4. WHERE절로 먼저 필터링 후 GROUP BY
5. ANSI 구문 사용 권장 (표준이며 가독성 좋음)

7. 자주 사용하는 패턴

sql

-- 부서별 평균 급여가 300만원 이상인 부서 조회

```
SELECT
    DEPT_CODE,
    AVG(SALARY)
FROM
    EMPLOYEE
WHERE
    DEPT_CODE IS NOT NULL
GROUP BY
    DEPT_CODE
HAVING
    AVG(SALARY) >= 3000000;
```

-- 사원명, 부서명, 직급명 조회

```
SELECT
    E.EMP_NAME,
    D.DEPT_TITLE,
    J.JOB_NAME
FROM
    EMPLOYEE E
JOIN
    DEPARTMENT D ON (E.DEPT_CODE = D.DEPT_ID)
JOIN
    JOB J USING(JOB_CODE);
```