

# 관계와 조인의 이해 개념서

이 문서는 데이터베이스에서 중요한 개념인 관계와 조인에 대해 상세히 설명합니다. 관계의 기본 개념부터 시작하여 다양한 조인 유형, 조인 전략, 성능 고려사항, 고급 조인 기법, 실전 활용 예제, 그리고 문제 해결 방법까지 포괄적으로 다룹니다. 또한 SQLD 시험을 준비하는 학습자를 위한 팁도 포함되어 있습니다.

## 1. 관계와 조인의 기본 개념

### 1.1 관계의 정의

- 데이터 모델에서 엔터티 간의 업무적 연관성
- 데이터의 논리적 연결 구조
- 물리적으로는 조인으로 구현

### 1.2 조인의 정의

- 두 개 이상의 테이블을 연결하여 데이터를 조회
- 관계를 기반으로 한 데이터 검색 방법
- PRIMARY KEY와 FOREIGN KEY를 활용

## 2. 관계와 조인의 대응

### 2.1 관계와 조인의 매핑

- 1:1 관계**

```
-- 1:1 관계 구현
SELECT e.*, l.사물함번호
FROM 사원 e
JOIN 사물함 l ON e.사원번호 = l.사원번호;
```

- 1:N 관계**

```
-- 1:N 관계 구현
SELECT d.부서명, e.사원명
FROM 부서 d
JOIN 사원 e ON d.부서번호 = e.부서번호;
```

- N:M 관계**

```
-- N:M 관계 구현
SELECT s.학생명, c.과목명
FROM 학생 s
JOIN 수강신청 r ON s.학번 = r.학번
JOIN 과목 c ON r.과목코드 = c.과목코드;
```

## 3. 조인의 종류

### 3.1 INNER JOIN

```
-- 기본 INNER JOIN
SELECT e.사원명, d.부서명
FROM 사원 e
INNER JOIN 부서 d ON e.부서번호 = d.부서번호;

-- 다중 INNER JOIN
SELECT e.사원명, d.부서명, p.프로젝트명
FROM 사원 e
INNER JOIN 부서 d ON e.부서번호 = d.부서번호
INNER JOIN 프로젝트 p ON e.프로젝트번호 = p.프로젝트번호;
```

### 3.2 OUTER JOIN

```
-- LEFT OUTER JOIN
SELECT e.사원명, d.부서명
FROM 사원 e
LEFT OUTER JOIN 부서 d ON e.부서번호 = d.부서번호;

-- RIGHT OUTER JOIN
SELECT e.사원명, d.부서명
FROM 사원 e
RIGHT OUTER JOIN 부서 d ON e.부서번호 = d.부서번호;

-- FULL OUTER JOIN
SELECT e.사원명, d.부서명
FROM 사원 e
FULL OUTER JOIN 부서 d ON e.부서번호 = d.부서번호;
```

## 4. 관계에 따른 조인 전략

### 4.1 식별 관계 조인

```
-- 식별 관계의 조인
SELECT o.주문번호, d.상품코드, d.수량
FROM 주문 o
JOIN 주문상세 d ON o.주문번호 = d.주문번호;
```

### 4.2 비식별 관계 조인

```
-- 비식별 관계의 조인
SELECT e.사원명, d.부서명
FROM 사원 e
LEFT JOIN 부서이력 d ON e.부서번호 = d.부서번호;
```

## 5. 조인의 성능 고려사항

### 5.1 조인 최적화

#### 1. 인덱스 활용

```
-- 조인 컬럼 인덱스
CREATE INDEX idx_dept_no ON 사원(부서번호);
CREATE INDEX idx_emp_no ON 부서(부서번호);
```

#### 2. 조인 순서

```
-- 작은 테이블을 먼저 조인
SELECT /*+ ORDERED */ *
FROM 소규모테이블 a
JOIN 대규모테이블 b ON a.컬럼 = b.컬럼;
```

### 5.2 조인 조건

```
-- 적절한 조인 조건
SELECT e.사원명, d.부서명
FROM 사원 e
JOIN 부서 d ON e.부서번호 = d.부서번호
WHERE e.입사일자 >= '2023-01-01';

-- 부적절한 조인 조건 (카테시안 곱)
SELECT e.사원명, d.부서명
FROM 사원 e, 부서 d;
```

## 6. 고급 조인 기법

### 6.1 SELF JOIN

```
-- 조직도 구현
SELECT e1.사원명 as 사원,
       e2.사원명 as 관리자
FROM 사원 e1
LEFT JOIN 사원 e2 ON e1.관리자번호 = e2.사원번호;
```

### 6.2 NATURAL JOIN

```
-- 자동 조인
SELECT *
FROM 사원
NATURAL JOIN 부서;

-- USING 절 활용
SELECT *
FROM 사원
JOIN 부서 USING (부서번호);
```

## 7. 실전 조인 활용 예제

### 7.1 복합 조인

```
-- 다중 테이블 조인
SELECT c.고객명,
       o.주문일자,
       p.상품명,
       d.주문수량
FROM 고객 c
JOIN 주문 o ON c.고객번호 = o.고객번호
JOIN 주문상세 d ON o.주문번호 = d.주문번호
JOIN 상품 p ON d.상품코드 = p.상품코드;
```

### 7.2 분석용 조인

```
-- 부서별 급여 통계
SELECT d.부서명,
       COUNT(*) as 인원수,
       AVG(e.급여) as 평균급여,
       MAX(e.급여) as 최대급여
FROM 부서 d
LEFT JOIN 사원 e ON d.부서번호 = e.부서번호
GROUP BY d.부서명;
```

## 8. 조인 관련 문제 해결

### 8.1 카테시안 곱 방지

```
-- 잘못된 예
SELECT *
FROM 사원, 부서;

-- 올바른 예
SELECT *
FROM 사원 e
JOIN 부서 d ON e.부서번호 = d.부서번호;
```

### 8.2 NULL 처리

```
-- OUTER JOIN의 NULL 처리
SELECT e.사원명,
       COALESCE(d.부서명, '미배치') as 부서명
FROM 사원 e
LEFT JOIN 부서 d ON e.부서번호 = d.부서번호;
```

## SQLD 시험 대비 TIP

### 주요 출제 포인트

- 관계와 조인의 개념
  - 관계의 종류
  - 조인의 유형
- 조인 구분
  - INNER/OUTER JOIN
  - SELF JOIN
- 성능 고려사항
  - 조인 최적화
  - 인덱스 활용

### 학습 전략

- 관계와 조인의 개념 이해
- 다양한 조인 구분 실습
- 성능 관련 내용 학습
- 실전 예제 연습

### 실전 문제 유형

- 관계에 맞는 조인 선택
- 조인 결과 예측
- 성능 개선 방안
- 복합 조인 작성