

정규화(Normalization) 개념서

이 문서는 SQLD 시험을 위한 데이터베이스 정규화 개념에 대해 상세히 설명합니다. 정규화의 기본 개념, 단계별 정규화 과정, 장단점, 이상 현상, 반정규화, 그리고 실전 예제를 다룹니다. SQLD 시험 준비를 위한 주요 포인트와 학습 전략도 제공합니다.

1. 정규화의 기본 개념

1.1 정의

- 데이터의 중복을 최소화하고 데이터 일관성과 무결성을 보장하기 위한 데이터베이스 설계 기법
- 하나의 릴레이션을 여러 개의 릴레이션으로 분해하는 과정
- 각 정규형의 규칙을 따라 단계적으로 진행

1.2 목적

- 데이터 중복 제거**
 - 저장 공간 절약
 - 데이터 불일치 방지
- 데이터 무결성 확보**
 - 갱신 이상 방지
 - 삭제 이상 방지
 - 삽입 이상 방지

2. 정규화 단계

2.1 제1정규형(1NF)

- 모든 속성의 값이 원자값을 가짐
- 반복되는 속성이나 그룹을 제거

```
-- 제1정규화 전
CREATE TABLE 학생 (
  학번 NUMBER,
  이름 VARCHAR2(50),
  전화번호 VARCHAR2(100) -- '010-1234-5678, 02-123-4567'
);

-- 제1정규화 후
CREATE TABLE 학생 (
  학번 NUMBER,
  이름 VARCHAR2(50)
);

CREATE TABLE 학생연락처 (
  학번 NUMBER,
  전화번호 VARCHAR2(20)
);
```

2.2 제2정규형(2NF)

- 제2정규형을 만족
- 부분 함수적 종속 제거
- 모든 비주식별자가 주식별자에 완전 함수적 종속

```
-- 제2정규화 전
CREATE TABLE 수강신청 (
  학번 NUMBER,
  과목코드 NUMBER,
  성적 NUMBER,
  지도교수 VARCHAR2(50) -- 학번에만 종속
);

-- 제2정규화 후
CREATE TABLE 수강신청 (
  학번 NUMBER,
  과목코드 NUMBER,
  성적 NUMBER
);

CREATE TABLE 학생정보 (
  학번 NUMBER,
  지도교수 VARCHAR2(50)
);
```

2.3 제3정규형(3NF)

- 제2정규형을 만족
- 이행적 종속 제거
- 비주식별자가 다른 비주식별자에 의존하지 않음

```
-- 제3정규화 전
CREATE TABLE 주문 (
  주문번호 NUMBER,
  고객번호 NUMBER,
  고객등급 VARCHAR2(10) -- 고객번호에 종속
);

-- 제3정규화 후
CREATE TABLE 주문 (
  주문번호 NUMBER,
  고객번호 NUMBER
);

CREATE TABLE 고객 (
  고객번호 NUMBER,
  고객등급 VARCHAR2(10)
);
```

2.4 BCNF(Boyce-Codd Normal Form)

- 제3정규형을 만족
- 모든 결정자가 후보키가 되도록 분해

```
-- BCNF 전
CREATE TABLE 강의 (
  교수번호 NUMBER,
  과목코드 NUMBER,
  교수이름 VARCHAR2(50),
  PRIMARY KEY (교수번호, 과목코드)
);

-- BCNF 후
CREATE TABLE 교수 (
  교수번호 NUMBER PRIMARY KEY,
  교수이름 VARCHAR2(50)
);

CREATE TABLE 강의 (
  교수번호 NUMBER,
  과목코드 NUMBER,
  PRIMARY KEY (교수번호, 과목코드)
);
```

3. 정규화의 장단점

3.1 장점

- 데이터 중복 최소화**
 - 저장 공간 효율화
 - 데이터 일관성 유지
- 데이터 무결성 보장**
 - 갱신 이상 방지
 - 삭제 이상 방지
 - 삽입 이상 방지
- 데이터 구조의 안정성**
 - 확장성 용이
 - 유지보수 편리

3.2 단점

- 테이블 증가**
 - JOIN 연산 증가
 - 조회 성능 저하 가능
- 복잡도 증가**
 - 구조 복잡
 - 구현 난이도 상승

4. 정규화 이상 현상

4.1 삽입 이상

```
-- 삽입 이상 예시
CREATE TABLE 수강정보 (
  학번 NUMBER,
  과목명 VARCHAR2(50),
  교수명 VARCHAR2(50),
  성적 NUMBER,
  PRIMARY KEY (학번, 과목명)
);

-- 새로운 과목 추가 시 반드시 수강 학생이 필요
```

4.2 갱신 이상

```
-- 갱신 이상 예시
CREATE TABLE 직원정보 (
  사번 NUMBER,
  이름 VARCHAR2(50),
  부서명 VARCHAR2(50),
  부서연락처 VARCHAR2(20)
);

-- 부서연락처 변경 시 모든 직원 레코드 수정 필요
```

4.3 삭제 이상

```
-- 삭제 이상 예시
CREATE TABLE 교수강의 (
  교수번호 NUMBER,
  교수명 VARCHAR2(50),
  과목코드 NUMBER,
  강의실 VARCHAR2(20)
);

-- 특정 과목 수강생이 없을 경우 교수 정보도 함께 삭제
```

5. 반정규화(Denormalization)

5.1 반정규화 대상

- 테이블 반정규화**

```
-- 테이블 병합
CREATE TABLE 고객주문통합 (
  고객번호 NUMBER,
  고객명 VARCHAR2(50),
  주문번호 NUMBER,
  주문금액 NUMBER
);
```

- 속성 반정규화**

```
-- 계산된 속성 추가
CREATE TABLE 주문 (
  주문번호 NUMBER,
  주문금액 NUMBER,
  할인금액 NUMBER,
  최종금액 NUMBER -- 계산된 속성
);
```

5.2 반정규화 절차

- 대상 조사**
 - 성능 저하 조사
 - 품질 저하 조사
- 다른 방법 검토**
 - 뷰 테이블
 - 인덱스 조정
 - 응용 프로그램 로직 조정
- 반정규화 실행**
 - 테이블 통합
 - 속성 추가
 - 관계 조정

6. 실전 정규화 예제

6.1 주문 시스템

```
-- 정규화 전
CREATE TABLE 주문통합 (
  주문번호 NUMBER,
  고객번호 NUMBER,
  고객명 VARCHAR2(50),
  고객등급 VARCHAR2(10),
  상품코드 NUMBER,
  상품명 VARCHAR2(100),
  주문수량 NUMBER,
  단가 NUMBER
);

-- 정규화 후
CREATE TABLE 고객 (
  고객번호 NUMBER PRIMARY KEY,
  고객명 VARCHAR2(50),
  고객등급 VARCHAR2(10)
);

CREATE TABLE 상품 (
  상품코드 NUMBER PRIMARY KEY,
  상품명 VARCHAR2(100),
  단가 NUMBER
);

CREATE TABLE 주문 (
  주문번호 NUMBER PRIMARY KEY,
  고객번호 NUMBER REFERENCES 고객(고객번호),
  주문일자 DATE
);

CREATE TABLE 주문상세 (
  주문번호 NUMBER REFERENCES 주문(주문번호),
  상품코드 NUMBER REFERENCES 상품(상품코드),
  주문수량 NUMBER,
  PRIMARY KEY (주문번호, 상품코드)
);
```

SQLD 시험 대비 TIP

주요 출제 포인트

- 정규화 개념**
 - 각 정규형의 특징
 - 정규화 목적
- 이상 현상**
 - 삽입/갱신/삭제 이상
 - 해결 방안
- 반정규화**
 - 반정규화 대상
 - 적용 시기

학습 전략

- 각 정규형의 개념 이해
- 이상 현상 사례 학습
- 정규화 과정 실습
- 반정규화 조건 파악

실전 문제 유형

- 정규화 단계 식별
- 이상 현상 찾기
- 정규화 수행
- 반정규화 판단