

본질식별자 vs 인조식별자 개념서

이 문서는 데이터베이스 설계에서 중요한 개념인 식별자, 특히 본질식별자와 인조식별자에 대해 상세히 설명합니다. 식별자의 기본 개념, 특성, 본질식별자와 인조식별자의 정의, 장단점, 선택 기준, 실전 활용 예제, 성능 고려사항, 그리고 식별자 전환 전략을 다룹니다. 또한 SQLD 시험 준비를 위한 팁도 포함하고 있습니다.

1. 식별자의 기본 개념

1.1 식별자의 정의

- 엔터티 내에서 인스턴스를 유일하게 구분할 수 있는 속성 또는 속성의 조합
- 업무적으로나 기술적으로 엔터티를 식별하는 용도로 사용
- 주식별자로 선택되어 기본키가 됨

1.2 식별자의 특성

- 유일성**
 - 인스턴스를 유일하게 구분
 - 중복값 불허용
- 최소성**
 - 최소한의 속성으로 구성
 - 불필요한 속성 제외
- 불변성**
 - 한번 부여된 값은 변경 불가
 - 안정적인 식별 보장

2. 본질식별자(Natural Key)

2.1 정의와 특징

```
-- 업무적으로 만들어지는 자연스러운 식별자
CREATE TABLE 직원 (
  주민등록번호 CHAR(13) PRIMARY KEY,
  이름 VARCHAR2(100),
  입사일자 DATE
);
```

2.2 장점

- 업무적 의미**
 - 실제 업무적 의미 포함
 - 별도 속성 불필요
- 직관성**
 - 데이터의 의미 파악 용이
 - 업무 이해도 향상

2.3 단점

- 변경 가능성**

```
-- 식별자 변경 시 관련 테이블 모두 수정 필요
UPDATE 직원
SET 주민등록번호 = '9901011234567'
WHERE 주민등록번호 = '9901011234566';

UPDATE 급여이력
SET 주민등록번호 = '9901011234567'
WHERE 주민등록번호 = '9901011234566';
```

- 복잡성**

- 복합키 구성 가능성
- 조인 조건 복잡

3. 인조식별자(Surrogate Key)

3.1 정의와 특징

```
-- 인위적으로 만드는 식별자
CREATE TABLE 직원 (
  직원번호 NUMBER PRIMARY KEY, -- 인조식별자
  주민등록번호 CHAR(13) UNIQUE,
  이름 VARCHAR2(100),
  입사일자 DATE
);

-- 시퀀스를 통한 자동 생성
CREATE SEQUENCE 직원_SEQ
START WITH 1
INCREMENT BY 1;
```

3.2 장점

- 단순성**
 - 단순한 구조
 - 관리 용이
- 안정성**
 - 업무 변경에 영향 최소화
 - 변경 가능성 없음

3.3 단점

- 업무적 의미 없음**
 - 추가 속성 필요
 - 직관성 부족
- 추가 용량**
 - 저장 공간 추가 필요
 - 인덱스 크기 증가

4. 식별자 선택 기준

4.1 본질식별자 선택 시

```
-- 변경 가능성이 없는 경우
CREATE TABLE 국가 (
  국가코드 CHAR(3) PRIMARY KEY,
  국가명 VARCHAR2(100)
);

-- 업무적 의미가 중요한 경우
CREATE TABLE 상품코드 (
  상품코드 CHAR(10) PRIMARY KEY, -- 체계적인 분류코드
  상품명 VARCHAR2(200)
);
```

4.2 인조식별자 선택 시

```
-- 식별자가 자주 변경되는 경우
CREATE TABLE 고객 (
  고객번호 NUMBER PRIMARY KEY,
  이메일 VARCHAR2(100) UNIQUE,
  연락처 VARCHAR2(20)
);

-- 복잡기를 피하고 싶은 경우
CREATE TABLE 주문상세 (
  주문상세ID NUMBER PRIMARY KEY, -- 인조식별자
  주문번호 NUMBER,
  상품코드 VARCHAR2(20),
  수량 NUMBER,
  UNIQUE (주문번호, 상품코드)
);
```

5. 실전 활용 예제

5.1 본질식별자 활용

```
-- 코드성 테이블
CREATE TABLE 부서 (
  부서코드 CHAR(3) PRIMARY KEY, -- 본질식별자
  부서명 VARCHAR2(100),
  상위부서코드 CHAR(3),
  FOREIGN KEY (상위부서코드) REFERENCES 부서(부서코드)
);

-- 분류체계 테이블
CREATE TABLE 품목분류 (
  분류코드 CHAR(6) PRIMARY KEY, -- 본질식별자
  분류명 VARCHAR2(100),
  상위분류코드 CHAR(6),
  FOREIGN KEY (상위분류코드) REFERENCES 품목분류(분류코드)
);
```

5.2 인조식별자 활용

```
-- 트랜잭션 테이블
CREATE TABLE 주문 (
  주문ID NUMBER PRIMARY KEY, -- 인조식별자
  주문번호 VARCHAR2(20) UNIQUE, -- 업무식별자
  주문일자 DATE,
  고객ID NUMBER
);

-- 이력 테이블
CREATE TABLE 상품변경이력 (
  이력ID NUMBER PRIMARY KEY, -- 인조식별자
  상품코드 VARCHAR2(20),
  변경일시 TIMESTAMP,
  변경내용 VARCHAR2(4000)
);
```

6. 성능 고려사항

6.1 저장 공간

```
-- 본질식별자 (복합키)
CREATE TABLE 주문상세 (
  주문번호 VARCHAR2(20),
  상품코드 VARCHAR2(20),
  수량 NUMBER,
  PRIMARY KEY (주문번호, 상품코드)
);

-- 인조식별자
CREATE TABLE 주문상세 (
  주문상세ID NUMBER PRIMARY KEY,
  주문번호 VARCHAR2(20),
  상품코드 VARCHAR2(20),
  수량 NUMBER
);
```

6.2 조인 성능

```
-- 복합키 조인 (본질식별자)
SELECT *
FROM 주문상세 d
JOIN 주문 o ON d.주문번호 = o.주문번호
WHERE d.상품코드 = 'P001';

-- 단일키 조인 (인조식별자)
SELECT *
FROM 주문상세 d
JOIN 주문 o ON d.주문ID = o.주문ID
WHERE d.상품코드 = 'P001';
```

7. 식별자 전환 전략

7.1 본질식별자에서 인조식별자로 전환

```
-- 1. 인조식별자 컬럼 추가
ALTER TABLE 테이블
ADD (새식별자 NUMBER);

-- 2. 시퀀스 생성 및 값 할당
CREATE SEQUENCE 테이블_SEQ;
UPDATE 테이블
SET 새식별자 = 테이블_SEQ.NEXTVAL;

-- 3. 기본키 변경
ALTER TABLE 테이블
DROP PRIMARY KEY;

ALTER TABLE 테이블
ADD PRIMARY KEY (새식별자);
```

SQLD 시험 대비 TIP

주요 출제 포인트

- 식별자의 특징**
 - 본질식별자와 인조식별자의 차이
 - 각각의 장단점
- 선택 기준**
 - 적절한 식별자 선택
 - 업무적 특성 고려
- 성능 영향**
 - 저장 공간
 - 조인 성능

학습 전략

- 식별자 개념 이해
- 각 식별자의 장단점 숙지
- 실제 사례 분석
- 성능 관련 내용 학습

실전 문제 유형

- 식별자 선택
- 장단점 비교
- 성능 영향 분석
- 전환 전략 수립