

# *Constraints*

---

---

# 제약 조건이란

- 제약 조건은 테이블 레벨로 교칙을 적용
- 제약 조건은 종속된 테이블의 삭제를 방지
- 유효한 제약 조건 유형

제약 조건	설명
NOT NULL	열이 널 값을 포함하지 못하도록 지정
UNIQUE	테이블의 모든 행에서 고유한 값을 갖는 열 또는 열 부 여 참고조합을 지정
PRIMARY KEY	테이블의 각 행을 고유하게 식별
FOREIGN KEY	한 열과 참조된 테이블의 열 간에 외래 키 관계를 설정 하고 시행
CHECK	참(TRUE)이어야 하는 조건을 지정

# 제약 조건 지침

---

- 제약 조건에 이름을 지정하지 않으면 Oracle Server 가 SYS\_Cn(n은 정수)형식의 이름을 생성
- 제약 조건 생성 시기
  - 테이블이 생성될 때
  - 테이블이 생성된 후
- 열 레벨 또는 테이블 레벨로 제약 조건을 정의
- 데이터 디렉터리에서 제약 조건을 봄
  - USER\_CONSTRAINTS

# 제약 조건 정의

---

- CREATE TABLE [schema.]table  
    (column datatype [DEFAULT expr]  
    [column\_constraint], ...  
    [table\_constraint] [, ...]);
- CREATE TABLE "EMP"(  
    "ID" NUMBER(6,0),  
    "NAME" VARCHAR(20), ...  
    "JOB\_ID" VARCHAR(10) NOT NULL,  
    CONSTRAINT "EMP\_ID\_PK" PRIMARY KEY ("ID"));

# 제약 조건 정의

## ■ 열 제약 조건 레벨

- Column [CONSTRAINT constraint\_name] constraint\_type,

## ■ 테이블 제약 조건 레벨

- Column,

...

[CONSTRAINT constraint\_name] constraint\_type  
(column, ...),

제약 조건 레벨	설명
열	단일 열을 참조하며 해당 제약 조건을 소유하는 열의 명세 부분에서 정의. 모든 무결성 제약 유형을 정의 가능
테이블	하나 이상의 열을 참조하며 테이블의 열 정의와는 별도로 정의. NOT NULL 을 제외한 모든 제약 조건을 정의 가능

# Constraint – NOT NULL

- 해당 열에 널 값을 사용할 수 없도록 함
- 열 레벨만 정의 가능

```
CREATE TABLE employee(  
    employee_id        NUMBER(6),  
    last_name          VARCHAR(25) NOT NULL,  
    salary              NUMBER(8,2),  
    commission_pct     NUMBER(2,2),  
    hire_date           DATE CONSTRAINT  
                        "EMP_HIRE_DATE_NN" NOT NULL,  
    ... )
```

시스템이 이름 지정



사용자가 이름 지정



# Constraint – UNIQUE

---

- 모든 키 값이 고유하게 유지되도록 함
- NOT NULL 을 정의하지 않으면 널 값 허용
  - 널 값은 어떠한 값과도 동일한 값으로 취급되지 않음
  - 테이블 레벨, 열 레벨로 정의

```
CREATE TABLE employee(  
    employee_id      NUMBER(6),  
    last_name        VARCHAR2(25) NOT NULL,  
    email            VARCHAR2(25),  
    salary            NUMBER(8,2)  
    commission_pct   NUMBER(2,2),  
    hire_date        DATE NOT NULL, ...  
    CONSTRAINT "EMP_EMAIL_UK" UNIQUE(email));
```

# Constraint – PRIMARY KEY

---

## ■ 테이블의 기본 키 생성

- 테이블 당 하나만 생성 가능
- 각 행을 고유하게 식별하는 열 또는 열 집합
- 널 값을 가질 수 없음
- 테이블 레벨 또는 열 레벨 정의

```
CREATE TABLE departments(  
    department_id      NUMBER(4),  
    department_name    VARCHAR2(30) NOT NULL,  
    manager_id         NUMBER(6),  
    location_id        NUMBER(8),  
    CONSTRAINT "DEPT_ID_PK" PRIMARY KEY(department_id);
```



# Constraint – FOREIGN KEY

---

- 동일한 테이블 또는 다른 테이블의 기본 키 또는 고유 키와의 참조 관계
- 부모 테이블(참조되는)의 기존 값과 일치하거나 NULL
- 테이블 레벨, 열 레벨 정의
  - 조합 외래 키는 테이블 레벨 정의여야 함
- CREATE TABLE employees(  
    employee\_id        NUMBER(6),  
    ...  
    CONSTRAINT “FK” FOREIGN KEY(department\_id)  
        REFERENCES departments(department\_id),  
    ...;

# Constraint – FOREIGN KEY

---

## ■ FOREIGN KEY

- 테이블 제약 조건 레벨로 자식 테이블의 열을 정의

## ■ REFERENCES

- 부모 테이블 및 부모 테이블에서의 해당 열을 식별

## ■ ON DELETE CASCADE

- 부모 테이블의 행이 삭제되는 경우 자식 테이블의 종속 행을 삭제

## ■ ON DELETE SET NULL

- 종속 외래 키 값을 널로 변환

# Constraint – CHECK

---

- 각 행이 만족시켜야 하는 조건을 정의
  - 다음 표현식은 허용되지 않음
    - CURRVAL, NEXTVAL, LEVEL, ROWNUM 의사 열 참조
    - SYSDATE, UID, USER, USERENV 함수 호출
    - 다른 행의 다른 값을 참조하는 질의
  - 단일 열이 자신을 참조하는 CHECK 제약 조건을 열 정의에 여러 개 포함 가능
    - 열에 정의할 수 있는 CHECK 제약 조건의 수는 무제한
- ```
CREATE TABLE employees( ...,  
    salary NUMBER(8,2) CONSTRAINT emp_salary_min  
    CHECK (salary > 0),  
    ... )
```

# Constraints managing

---

- 제약 조건을 추가 또는 삭제할 수 있지만 구조를 수정할 수는 없음
- 제약 조건을 활성화 또는 비활성화
- NOT NULL 제약 조건 추가는 ALTER TABLE ... MODIFY 를 이용
  - 테이블이 비어 있거나 해당 열의 모든 행에 값이 있는 경우
  - ALTER TABLE employees MODIFY (salary NOT NULL);

# Add constraint

---

- ALTER TABLE table\_name  
ADD [CONSTRAINT constraint\_name] type (column);
- ALTER TABLE employees  
ADD CONSTRAINT emp\_manager\_fk  
FOREIGN KEY(manager\_id)  
REFERENCES employees(employee\_id);

# Drop constraint

---

- CASCADE 옵션

- 종속 제약 조건 모두 삭제

- ALTER TABLE table\_name

DROP PRIMARY KEY | UNIQUE (column) |  
CONSTRAINT constraint [CASCADE];

- ALTER TABLE            employee  
      DROP CONSTRAINT    emp\_manager\_fk;
  - ALTER TABLE            departments  
      DROP PRIMARY KEY CASCADE;

# Disable constraint

---

- CREATE TABLE 에도 사용 가능
- 고유 또는 기본 키 제약 조건을 비활성화하면 고유 인덱스가 제거됨
- ALTER TABLE table\_name  
DISABLE CONSTRAINT constraint\_name [CASCADE];

```
ALTER TABLE employees  
DISABLE CONSTRAINT emp_emp_id_pk CASCADE;
```

# Enable constraint

---

- CREATE TABLE 에서 사용 가능
- 활성화 시 해당 제약 조건이 테이블에 있는 모든 데이터에 적용, 테이블의 모든 데이터는 제약 조건을 만족해야 함
- UNIQUE or PRIMARY KEY 를 활성화하면 UNIQUE or PRIMARY KEY 인덱스가 자동으로 생성
- CASCADE 옵션으로 비활성화했던 기본 키 제약 조건을 다시 활성화해도 종속된 외래 키는 활성화 되지 않음

ALTER TABLE employees

ENABLE CONSTRAINT emp\_emp\_id\_pk;



# 제약 조건 연쇄화

---

- CASCADE CONSTRAINTS 절은 DROP COLUMN 절과 함께 사용
- CASCADE CONSTRAINTS 절을 사용하면 삭제되는 열에 정의된 기본 키 및 고유 키를 참조하는 모든 참조 무결성 제약 조건이 삭제됨
- CASCADE CONSTRAINTS 절은 삭제되는 열에 정의된 다중 열 제약 조건도 모두 삭제함

# 제약 조건 연쇄화

---

```
CREATE TABLE test1( pk NUMBER PRIMARY KEY,  
    fk NUMBER,  
    col1 NUMBER,  
    col2 NUMBER,  
    CONSTRAINT fk_constraint FOREIGN KEY (fk) REFERENCES test1,  
    CONSTRAINT ck1 CHECK (pk > 0 and col1 > 0),  
    CONSTRAINT ck2 CHECK (col2 > 0));
```

ALTER TABLE test1 DROP (pk); – 에러

ALTER TABLE test1 DROP (col1); – 에러

ALTER TABLE test1 DROP (pk) CASCADE CONSTRAINTS;

ALTER TABLE test1 DROP (pk, fk, col1);

## 제약 조건 보기

---

- `SELECT constraint_name, constraint_type,  
search_condition FROM user_constraints WHERE  
table_name = 'EMPLOYEES';`
  - C ; CHECK
    - NOT NULL 포함
  - P ; PRIMARY KEY
  - R ; FOREIGN KEY REFERENCES
  - U ; UNIQUE

## 제약 조건 보기

---

- `SELECT constraint_name, column_name FROM user_cons_columns WHERE table_name = 'EMPLOYEES';`

1. EMP 테이블의 ID 열에 테이블 레벨의 PRIMARY KEY 제약 조건을 추가. 제약 조건 이름은 my\_emp\_id\_pk로 지정하시오.
2. ID 열을 사용하여 DEPT 테이블에 PRIMARY KEY 제약 조건을 생성. 제약 조건 이름은 my\_dept\_id\_pk로 지정하시오.
3. EMP 테이블에 DEPT\_ID 열을 추가. 존재하지 않는 부서에 사원이 배정되지 않도록 외래 키 참조를 EMP 테이블에 추가. 제약 조건 이름은 my\_emp\_dept\_id\_fk로 지정하시오.



4. USER\_CONSTRAINTS 뷰를 질의하여 제약 조건이 추가되었는지 확인하시오. (EMP, DEPT 테이블만 질의)
5. EMP 테이블을 수정하여 십진 자릿수 2, 소수점 이하 자릿수 2인 NUMBER 데이터 유형의 COMMISSION 열을 추가하시오. 커미션 값이 0보다 크도록 커미션 열에 제약 조건을 추가하시오.