

## 제 4 장

# SQL(Structured Query Language)



**SQL Server**

**ORACLE®**

# 목 차

4.1 데이터 정의어

4.2 데이터 조작어

4.3 뷰(view)

4.4 데이터 무결성

## 4.1 데이터 정의어

- 데이터베이스 언어, 데이터 언어, DBMS 언어

- 데이터 정의어(DDL; Data Definition Language)
- 데이터 조작어(DML; Data Manipulation Language)
- 데이터 제어어(DCL; Data Control Language)

- 데이터 정의어

- 1) 데이터베이스, 테이블, 어트리뷰트를 정의하거나 변경, 삭제한다.
- 2) 테이블 내에서 각 어트리뷰트에 대한 형(type)을 정의한다.
- 3) 새로운 인덱스를 정의하거나, 삭제한다.
- 4) 기본키를 정의한다.
- 5) 새로운 뷰를 정의하거나, 삭제한다.
- 6) 사용자들에게 권한을 부여하며, 데이터 보안에 관여한다.

- CREATE TABLE, CREATE VIEW, CREATE INDEX,  
ALTER TABLE, DROP TABLE, DROP VIEW, DROP INDEX

## 4.1 데이터 정의어

- 베이스 테이블(BASE TABLE)

학번	이름	학과	학년	← 머리부분(heading)
2007010	홍길동	기계과	2	← 몸체부분(body)
2008001	이육사	컴퓨터시스템과	1	
2010002	김기자	정보통신과	1	
2009003	박이사	컴퓨터정보과	3	

## 4.1 데이터 정의어

- 테이블 정의

```
CREATE TABLE base-table name  
    (column-name data-type [NOT NULL], .....);
```

```
CREATE TABLE 학생신상  
    (학번 CHAR(7) NOT NULL,  
     이름 CHAR(10),  
     학과 CHAR(20)  
     학년 INTEGER);
```

P. 107 표 4.1

## 4.1 데이터 정의어

- 인덱스(Index)

- 빠른 접근

- 메모리 사용

```
CREATE INDEX 성명 ON 학생신상(이름);
```

```
DROP INDEX 성명;
```

## 4.2 데이터 조작용어

- 검색(select), 수정(update), 삽입(insert), 삭제(delete)

### 학사관리

#### 수강

과목번호	학번	점수
101	20131001	80
101	20131025	65
102	20153075	66
102	20153088	61
103	20152088	45
104	20131001	56
104	20131025	65
105	20153075	56
105	20153088	78
106	20132003	72
107	20143054	41
108	20151062	81

#### 학생

학번	이름	생년월일	학과	학년	반	성별	신장	체중
20131001	김종현	1992-12-12	컴퓨터정보	3	A	M		72
20131025	옥성우	1992-03-04	컴퓨터정보	3	A	F	172	63
20132003	박희철	1992-07-19	전기전자	3	B	M		63
20141007	진현무	1993-05-12	컴퓨터정보	2	C	M	174	64
20142021	심수정	1993-01-01	전기전자	2	A	F	168	45
20143054	유가인	1993-02-19	기계	2	C	F	154	47
20151062	김인중	1994-06-03	컴퓨터정보	1	B	M	166	67
20152088	조민우	1994-09-01	전기전자	1	C	M	188	90
20153075	옥한빛	1994-08-26	기계	1	C	M	177	80
20153088	이태연	1994-03-22	기계	1	C	F	162	50

#### 과목

과목번호	과목명	교수명	학년	학과
101	컴퓨터개론	강종영	3	컴퓨터정보
102	기계공학작법	김태영	1	기계
103	기초전자실험	김유석	1	전기전자
104	시스템분석설	강석현	3	컴퓨터정보
105	기계요소설계	김명성	1	기계
106	전자회로실험	최영민	3	전기전자
107	CAD응용실습	구봉규	2	기계
108	소프트웨어공	권민성	1	컴퓨터정보
109	자동화설계	박민영	3	기계
110	자동제어	정순정	2	전기전자
111	데이터베이스	이재영	2	컴퓨터정보

## 4.2 데이터 조작용어

### ● 학사관리 데이터베이스

학생		
	필드 이름	데이터 형식
🔑	학번	숫자
	이름	짧은 텍스트
	생년월일	날짜/시간
	학과	짧은 텍스트
	학년	숫자
	반	짧은 텍스트
	성별	짧은 텍스트
	신장	숫자
	체중	숫자

과목		
	필드 이름	데이터 형식
🔑	과목번호	숫자
	과목명	짧은 텍스트
	담당교수	짧은 텍스트
	학년	숫자
	학과	짧은 텍스트

수강		
	필드 이름	데이터 형식
🔑	과목번호	숫자
🔑	학번	숫자
	점수	숫자



## 4.2 데이터 조작용어

- 검색문(SELECT)

```
SELECT [DISTINCT] item-commalist  
FROM table names  
[WHERE predicate]  
[GROUP BY item-commalist [HAVING predicate]]  
[ORDER BY item-commalist];
```

## 4.2 데이터 조작어

### ● 단일 검색

〈질의1〉 학생의 학번, 이름, 학과를 검색하시오.

```
SELECT 학번, 이름, 학과  
FROM 학생;
```

학번	이름	학과
20131001	김종헌	컴퓨터정보
20131025	옥성우	컴퓨터정보
20132003	박희철	전기전자
20141007	진현무	컴퓨터정보
20142021	심수정	전기전자
20143054	유가인	기계
20151062	김인중	컴퓨터정보
20152088	조민우	전기전자
20153075	옥한빛	기계
20153088	이태연	기계

## 4.2 데이터 조작어

### ● 수식 포함 검색

〈질의2〉 학생들의 신장(cm)을 ft(피트)로 구하시오. 단,  $1\text{cm} = 0.032808\text{ft}$

```
SELECT 학번, 이름, 신장 * 0.032808  
FROM 학생;
```

학번	이름	
20131001	김종헌	
20131025	옥성우	5.642976
20132003	박희철	
20141007	진현무	5.708592
20142021	심수정	5.511744
20143054	유가인	5.052432
20151062	김인중	5.446128
20152088	조민우	6.167904
20153075	옥한빛	5.807016
20153088	이태연	5.314896

## 4.2 데이터 조작어

### ● 수식 포함 검색

〈질의3〉 과목 테이블의 모든 데이터를 검색하시오.

SELECT 과목번호, 과목명, 교수명, 학년, 학과  
FROM 과목;

SELECT \*  
FROM 과목;

과목번호	과목명	담당교수	학년	학과
101	컴퓨터개론	강종영	3	컴퓨터정보
102	기계공학법	김태영	1	기계
103	기초전자실험	김유석	1	전기전자
104	시스템분석설계	강석현	3	컴퓨터정보
105	기계요소설계	김명성	1	기계
106	전자회로실험	최영민	3	전기전자
107	CAD응용실습	구봉규	2	기계
108	소프트웨어공학	권민성	1	컴퓨터정보
109	자동화설계	박민영	3	기계
110	자동제어	정순정	2	전기전자
111	데이터베이스	이재영	2	컴퓨터정보

## 4.2 데이터 조작어

- 중복 행 제거

〈질의4〉 수강신청된 과목들의 과목번호를 검색하시오.

```
SELECT 과목번호  
FROM 수강;
```

과목번호
101
101
102
102
103
104
104
105
105
106
107
108

```
SELECT DISTINCT 과목번호  
FROM 수강;
```

과목번호
101
102
103
104
105
106
107
108

## 4.2 데이터 조작어

- 조건 검색

〈질의5〉 점수가 80점 이상인 학생의 학번을 검색하시오.

```
SELECT 학번  
FROM 수강  
WHERE 점수 >= 80;
```

학번
20131001
20151062

※ 비교연산자는 =, <>, <, <=, >, >=

## 4.2 데이터 조작어

### ● 조건 검색

〈질의6〉 생년월일이 1994-08-26인 학생의 학번, 이름, 생년월일을 검색해라

```
SELECT 학번, 이름, 생년월일  
FROM 학생  
WHERE 생년월일 = #1994-08-26#;
```

학번	이름	생년월일
20153075	옥성우	1994-08-26

## 4.2 데이터 조작어

- 범위 조건 검색

〈질의7〉 학생 테이블에서 14년도 학번만 검색하시오.

```
SELECT 학번  
FROM 학생  
WHERE 학번 BETWEEN 20140000 AND 20149999;
```

학번
20141007
20142021
20143054



## 4.2 데이터 조작용어

- 유사값 검색(LIKE)

〈질의8〉 성이 ‘김’인 학생의 학번번호와 이름을 검색하시오.

```
SELECT 학번, 이름  
FROM 학생  
WHERE 이름 LIKE “김*”;
```

학번	이름
20131001	김종현
20151062	김인중

### ※ 와일드 카드 문자

표준SQL	Access	의 미
%	*	0개 이상의 문자와 대응된다
_	?	1개의 문자와 대응된다

## 4.2 데이터 조작용어

- <NULL>값 처리

<질의9> 학생 테이블에서 신장이 NULL인 학생의 학번과 이름을 검색하시오.

```
SELECT 학번, 이름  
FROM 학생  
WHERE 신장 IS NULL;
```

학번	이름
20131001	김종헌
20132003	박희철

## 4.2 데이터 조작어

- 복합 조건 검색

〈질의10〉 컴퓨터정보과이면서 2학년인 학생의 이름을 검색하시오.

```
SELECT 이름  
FROM 학생  
WHERE 학과 = “컴퓨터정보” AND 학년 = 2;
```

이름
진현무

※논리연산자는 NOT, AND, OR

## 4.2 데이터 조작어

### ● 검색 결과의 순서화

〈질의11〉 학생의 학번과 이름을 학번의 내림차순으로 검색하시오.

```
SELECT 학번, 이름  
FROM 학생  
ORDER BY 학번 DESC;
```

학번	이름
20153088	이태연
20153075	옥한빛
20152088	조민우
20151062	김인중
20143054	유가인
20142021	심수정
20141007	진현무
20132003	박희철
20131025	옥성우
20131001	김종헌

```
* SELECT 학번, 이름  
FROM 학생  
ORDER BY 2 DESC;
```

\* P.111 〈질의 2〉

## 4.2 데이터 조작용

### ● 내장 함수(Library Function)

내장함수	기 능
SUM()	열 값들의 합계를 구한다.
AVG()	열 값들의 평균값을 구한다.
MAX()	열 값들의 최대값을 구한다.
MIN()	열 값들의 최소값을 구한다.
COUNT()	열의 데이터 값들의 개수를 구한다.
COUNT(*)	튜플(행)들의 개수를 구한다.

※ P. 119

## 4.2 데이터 조작용어

- 내장 함수

〈질의12〉 체중과 신장의 합을 구하라.

```
SELECT SUM(신장), SUM(신장)  
FROM 학생;
```

SUM(신장)	SUM(신장)
641	1361

```
SELECT COUNT(신장), SUM(신장), COUNT(신장), SUM(신장)  
FROM 학생;
```

## 4.2 데이터 조작어

- 그룹화 검색

〈질의13〉 학생별 학번과 점수 합계를 구하시오.

```
SELECT 학번, SUM(점수)
FROM 수강
GROUP BY 학번;
```

학번	Expr1001
20131001	136
20131025	130
20132003	72
20143054	41
20151062	81
20152088	45
20153075	122
20153088	139

## 4.2 데이터 조작어

- 그룹화 조건 검색

〈질의14〉 학생별 점수의 합계가 100 이상인 학생의 학번과 점수 합계를 구하시오.

```
SELECT 학번, SUM(점수)
FROM 수강
GROUP BY 학번 HAVING SUM(점수) >= 100;
```

학번	Expr1001
20131001	136
20131025	130
20153075	122
20153088	139

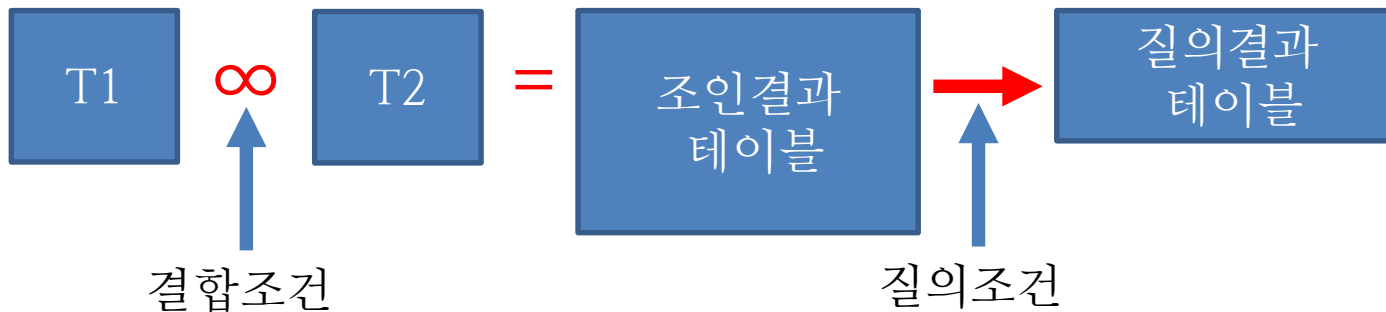


## 4.2 데이터 조작어

- 조인(JOIN) 검색

〈질의15〉 102번 과목을 수강한 학생의 이름과 학과를 검색하시오.

```
SELECT 이름, 학과  
FROM 학생, 수강  
WHERE 학생.학번 = 수강.학번 AND 과목번호 = 102;
```



이름	학과
옥한빛	기계
이태연	기계

## 4.2 데이터 조작어

- 3테이블 조인(JOIN) 검색

〈질의16〉 컴퓨터개론 과목을 수강한 학생의 이름과 학과를 검색하시오.

```
SELECT 이름, 학생.학과  
FROM 학생, 수강, 과목  
WHERE 학생.학번 = 수강.학번  
AND 수강.과목번호 = 과목.과목번호  
AND 과목명 = "컴퓨터개론";
```

이름	학과
김종헌	컴퓨터정보
옥성우	컴퓨터정보

## 4.2 데이터 조작용어

- 부질의(subquery) 다중질의

〈질의18〉 점수가 평균점수 이상인 학생의 학번과 점수를 구하시오.

```
SELECT 학번, 점수
FROM 수강
WHERE 점수 >= (SELECT AVG(점수)
               FROM 수강);
```

학번	점수
20131001	80
20131025	65
20153075	66
20131025	65
20153088	78
20132003	72
20151062	81

## 4.2 데이터 조작용어

- 원소 검사(**IN**)

〈질의19〉 점수가 80 이상인 학생의 이름을 검색하시오.

```
SELECT 이름  
FROM 학생  
WHERE 학번 IN (SELECT 학번  
                FROM 수강  
                WHERE 점수 >= 80);
```

이름

김종헌

김인중

```
SELECT 이름  
FROM 학생  
WHERE 학번 IN (20131001, 20151062)
```

- 모든 부질의 갖는 질의를 조인질의로 바꿀 수 있는 것은 아니다.

## 4.2 데이터 조작어

- 존재 검사( **EXISTS** )

〈질의20〉 한 과목 이상 수강한 학생의 학번을 검색하시오.

```
SELECT 학번, 이름
FROM 학생
WHERE EXISTS (SELECT *
               FROM 수강
               WHERE 학생.학번 = 수강.학번);
```

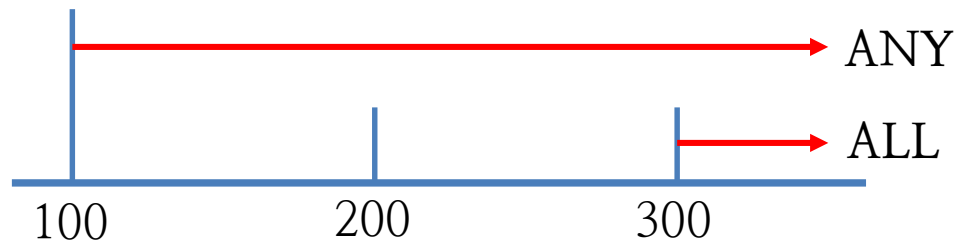
학번	이름
20131001	김종헌
20131025	옥성우
20132003	박희철
20143054	유가인
20151062	김인중
20152088	조민우
20153075	옥한빛
20153088	이태연

\* **NOT EXISTS**

## 4.2 데이터 조작어

- 한정 비교 검사

```
SELECT .....  
FROM .....  
WHERE 검사값 >= {ANY | ALL} ( 100, 200, 300);
```



## 4.2 데이터 조작용어

### ● 다단계 부질의

〈질의21〉 강종영교수의 과목을 수강한 학생의 이름을 검색하시오.

```
SELECT 이름
FROM 학생
WHERE 학번 IN
    (SELECT 학번
     FROM 수강
     WHERE 과목번호 IN
         (SELECT 과목번호
          FROM 과목
          WHERE 담당교수 = “강종영”));
```

이름
김종헌
육성우

## 4.2 데이터 조작어

- 집합연사자를 이용한 검색 연산

〈질의22〉 점수가 80점 이상이거나 수강한 과목이 2개 이상인 학생의 학번을 구하라.

```
SELECT 학번  
FROM 수강  
WHERE 점수 >= 80
```

```
UNION
```

```
SELECT 학번  
FROM 수강  
GROUP BY 학번 HAVING COUNT(*) >= 2
```

학번
20131001
20131025
20151062
20153075
20153088



## 4.2 데이터 조작어

- 변경문(UPDATE)

**UPDATE** table name

**SET** field = expression, [field = expression] ...

[**WHERE** predicate];

## 4.2 데이터 조작어

### ● 단일 튜플 변경

〈질의23〉 학번이 20132003인 학생의 이름이 '황지후' 로 변경되었다.

```
UPDATE 학생  
SET 이름 = “황지후”  
WHERE 학번 = 20132003;
```

학번	이름
20131001	김종헌
20131025	옥성우
20132003	박희철
20141007	진현무
20142021	심수정
20143054	유가인
20151062	김인중
20152088	조민우
20153075	옥한빛
20153088	이태연

학번	이름
20131001	김종헌
20131025	옥성우
20132003	황지후
20141007	진현무
20142021	심수정
20143054	유가인
20151062	김인중
20152088	조민우
20153075	옥한빛
20153088	이태연

## 4.2 데이터 조작어

### ● 다중 튜플 변경

〈질의24〉 과목번호 101의 점수가 10점 올랐다.

UPDATE 수강

SET 점수 = 점수 + 10

WHERE 과목번호 = 101;

과목번호	학번	점수
101	20131001	80
101	20131025	65
102	20153075	66
102	20153088	61
103	20152088	45

과목번호	학번	점수
101	20131001	90
101	20131025	75
102	20153075	66
102	20153088	61
103	20152088	45

## 4.2 데이터 조작용어

- 삽입문(INSERT)

INSERT

INTO table name [(field[, field] ... )]

VALUES (constant[, constant] ... );

또는

INSERT

INTO table name [(field[, field] ... )]

SELECT ... FROM ... WHERE ...;

## 4.2 데이터 조작용어

- 다일 튜플 삽입

〈질의25〉 컴퓨터정보과 1학년 B반에 20161032 김미영이 새로 입학하였다.  
이를 데이터베이스에 반영하시오.

```
INSERT  
  INTO 학생(학번, 이름, 학과, 학년, 반, 성별)  
  VALUES (20161032, “김미영”, “컴퓨터정보“, 1, “B”, “F”);
```

```
INSERT  
  INTO 학생  
  VALUES (20161032, “김미영”, NULL, “컴퓨터정보“, 1, “B”, “F”, NULL, NULL);
```

## 4.2 데이터 조작용어

- 삭제문(DELETE)

DELETE

FROM table name

[WHERE predicate]

## 4.2 데이터 조작어

- 단일 튜플 삭제

〈질의26〉 옥성우 학생이 자퇴하였다. 이를 데이터베이스에 반영하시오.

```
DELETE  
FROM 학생  
WHERE 이름 = “옥성우”;
```

## 4.2 데이터 조작용어

- 다중 튜플 삭제

〈질의27〉 수강 테이블에서 김종현 학생의 모든 자료를 삭제하라.

```
DELETE
FROM 수강
WHERE 학번 =
    (SELECT 학번
     FROM 학생
     WHERE 이름 = “김종현”);
```

〈질의28〉 수강 테이블의 모든 자료를 삭제하라.

```
DELETE
FROM 수강;
```

\*DROP TABLE



## 4.3 뷰(View)

- 가상의 테이블(Virtual table)
- 외부단계(external level)
- 뷰의 장점
  - ① 뷰는 데이터베이스를 재구성하여 논리적 데이터 독립성을 제공한다.
  - ② 뷰를 통하여 사용자와 관련된 데이터만을 되므로 데이터베이스를 보는 관점을 단순화시킬 수 있으며, 원하는 데이터만을 조작할 수 있게 함으로써 데이터의 보완 기능 강화한다.
- 뷰의 정의

```
CREATE VIEW view name
```

```
[(column name[, column name] ... )]
```

```
AS subquery;
```

## 4.3 뷰(View)

- 뷰를 정의하기 위한 규칙

- (1) 뷰에서 새로운 열의 이름을 명시하지 않으면, 그 열의 이름들은 베이스테이블에 있는 열의 이름을 따른다.
- (2) 열의 이름이 다음과 같을 경우에는 반드시 명시해 주어야 한다.
  - ① 뷰의 열이 내장함수, 산술식, 또는 상수로부터 유도되어 베이스테이블의 열 이름을 따를 수 없는 경우, 예를 들면 다음과 같다.

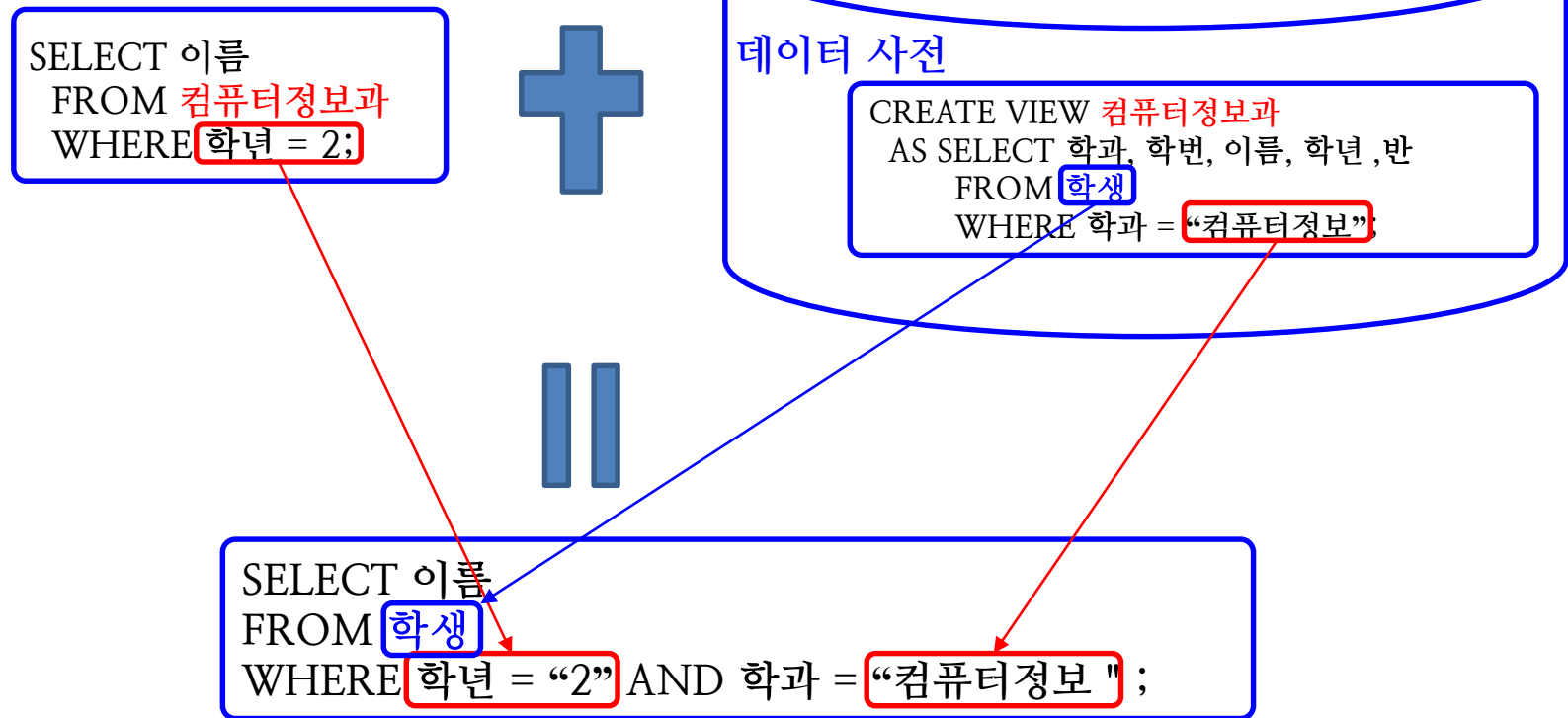
```
CREATE VIEW 컴퓨터정보과(학과, 학번, 이름, 학년, 반)
AS SELECT 학과, 학번, 이름, 학년, 반
FROM 학생
WHERE 학과 = “컴퓨터정보”;
```

- ② 2개 이상의 열들이 동일한 이름을 갖고 있는 경우
- (3) 뷰의 정의에는 UNION이나 ORDER BY를 사용할 수 없다.
- (4) 정의되어 있는 뷰로부터 새로운 뷰를 정의할 수 있다.

## 4.3 뷰(View)

- 뷰에 대한 검색

질의 변형(query modification)



## 4.3 뷰(View)

- 뷰에 대한 갱신 연산은 많은 문제점이 있음.
- 갱신 가능한 뷰의 제약 조건
  - (1) 뷰가 베이스테이블의 모든 기본 키를 포함하고 있지 않다면, 그 뷰는 갱신 이 불가능하다.
  - (2) 뷰의 열이 연산식, 상수, 내장함수로부터 유도되었다면 그 뷰는 갱신 불가능하다.
  - (3) 뷰의 정의에서 GROUP BY절을 포함하고 있다면 그 뷰는 갱신이 불가능하다.
  - (4) 뷰의 정의에서 FROM절에 여러 개의 테이블이 명시되었다면 이것은 조인뷰(join view)로 갱신이 불가능하다.
  - (5) 새로이 생성된 뷰가 갱신이 불가능한 뷰에서 유도되었다면 이 뷰 또한 갱신이 불가능하다.
- WITH CHECK OPTION

## 4.4 데이터 무결성

### ● 제약 조건

- 데이터의 정확성과 안전성
- 관계형 데이터 베이스 상태(인스턴스)가 항상 만족해야 할 기본 규칙
- 데이터 베이스 인스턴스가 삽입, 삭제, 수정 등의 연산을 통해서 변하더라도 지속적으로 만족해야 할 조건

### ❖ 도메인 제약조건(domain constraint)

- 각 어트리뷰트 값은 반드시 원자값이어야 함.

### ❖ 키 제약조건(key constraint)

- 릴레이션의 모든 키는 유일한 값이어야 함

### ❖ 엔티티 무결성 제약조건(entity integrity constraint)

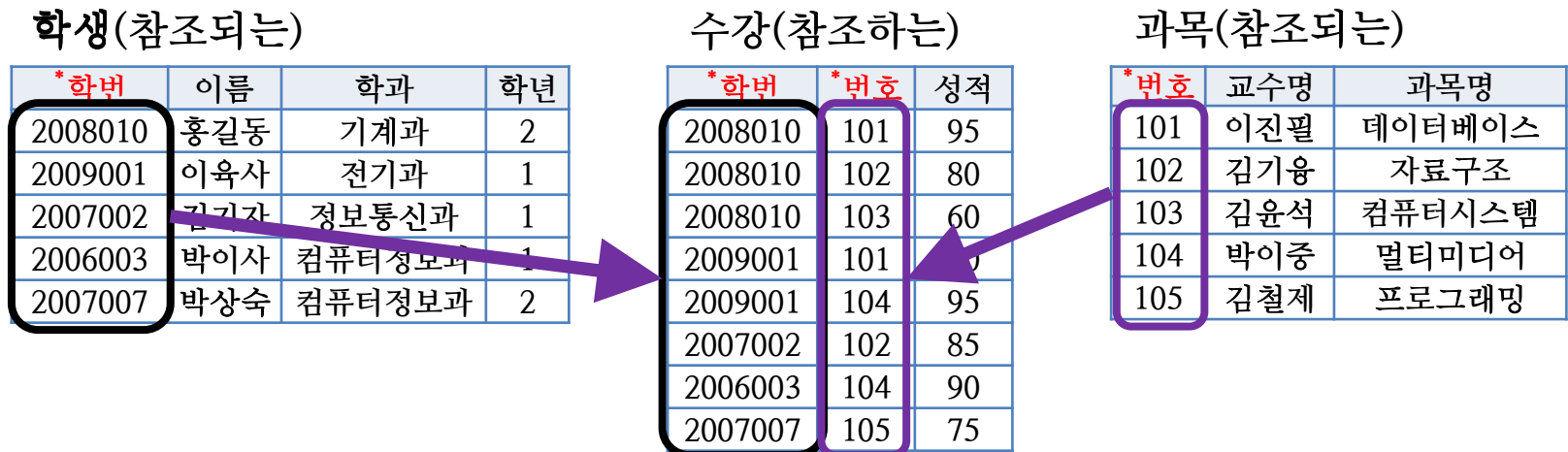
- 릴레이션의 기본키는 어떠한 경우에도 널값(null value)을 가질 수 없음

### ❖ 참조 무결성 제약조건(referential integrity constraint)

- 한 테이블에 있는 튜플이 다른 테이블에 있는 튜플을 참조하려면 반드시 참조되는 튜플이 그 테이블 내에 존재해야함.

## 4.4 데이터 무결성

- 참조 무결성 문제(고아 데이터가 발생할 때)



## 4.4 데이터 무결성

- 참조 무결성 문제(고아 데이터가 발생할 때)

① 새로운 튜플을 참조하는 테이블에 삽입할 때

학생(참조되는)

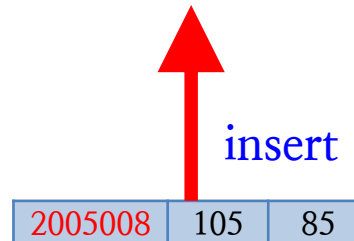
*학번	이름	학과	학년
2008010	홍길동	기계과	2
2009001	이육사	전기과	1
2007002	김기자	정보통신과	1
2006003	박이사	컴퓨터정보과	1
2007007	박상숙	컴퓨터정보과	2

수강(참조하는)

*학번	*번호	성적
2008010	101	95
2008010	102	80
2008010	103	60
2009001	101	90
2009001	104	95
2007002	102	85
2006003	104	90
2007007	105	75

과목(참조되는)

*번호	교수명	과목명
101	이진필	데이터베이스
102	김기용	자료구조
103	김윤석	컴퓨터시스템
104	박이중	멀티미디어
105	김철제	프로그래밍



## 4.4 데이터 무결성

### ● 참조 무결성 문제

② 참조하는 테이블의 외래키값을 변경할 때

학생(참조되는)

*학번	이름	학과	학년
2008010	홍길동	기계과	2
2009001	이육사	전기과	1
2007002	김기자	정보통신과	1
2006003	박이사	컴퓨터정보과	1
2007007	박상숙	컴퓨터정보과	2

수강(참조하는)

*학번	*번호	성적
2008010	101	95
2008010	102	80
2008010	103	60
2009001	101	80
2009001	104	95
2007002	102	85
2006003	104	90
2007007	105	75

과목(참조되는)

*번호	교수명	과목명
101	이진필	데이터베이스
102	김기용	자료구조
103	김윤석	컴퓨터시스템
104	박이중	멀티미디어
105	김철제	프로그래밍

update

2005007	101	95
---------	-----	----



## 4.4 데이터 무결성

### ● 참조 무결성 문제

③ 참조되는 테이블의 튜플을 삭제할 때

학생(참조되는)

*학번	이름	학과	학년
2008010	홍길동	기계과	2
<del>2009001</del>	<del>이옥사</del>	<del>전기과</del>	<del>1</del>
2007002	김기자	정보통신과	1
2006003	박이사	컴퓨터정보과	1
2007007	박상숙	컴퓨터정보과	2

delete

수강(참조하는)

*학번	*번호	성적
2008010	101	95
2008010	102	80
2008010	103	60
2009001	101	80
2009001	104	95
2007002	102	85
2006003	104	90
2007007	105	75

과목(참조되는)

*번호	교수명	과목명
101	이진필	데이터베이스
102	김기용	자료구조
103	김윤석	컴퓨터시스템
104	박이중	멀티미디어
105	김철제	프로그래밍

## 4.4 데이터 무결성

### ● 참조 무결성 문제

④ 참조되는 테이블의 기본키 값을 변경할 때

학생(참조되는)

*학번	이름	학과	학년
2008010	홍길동	기계과	2
2009001	이육사	전기과	1
2007002	김기자	정보통신과	1
2006003	박이사	컴퓨터정보과	1
2007007	박상숙	컴퓨터정보과	2

update

2009002	이육사	전기과	1
---------	-----	-----	---

수강(참조하는)

*학번	*번호	성적
2008010	101	95
2008010	102	80
2008010	103	60
2009001	101	80
2009001	104	95
2007002	102	85
2006003	104	90
2007007	105	75

과목(참조되는)

*번호	교수명	과목명
101	이진필	데이터베이스
102	김기용	자료구조
103	김윤석	컴퓨터시스템
104	박이중	멀티미디어
105	김철제	프로그래밍

## 4.4 데이터 무결성

### ● 참조 무결성 문제

- ① 새로운 튜플을 참조하는 테이블에 삽입할 때
- ② 참조하는 테이블의 외래키값을 변경할 때

작업 거절

- ③ 참조되는 테이블의 튜플을 삭제할 때
- ④ 참조되는 테이블의 기본키 값을 변경할 때

?

## 4.4 데이터 무결성

### ● 삭제 규칙

- ① 제한 삭제 규칙(restrict)
- ② 다단계 삭제 규칙(cascade)
- ③ NULL값 설정 삭제 규칙(set NULL)
- ④ 기본값 설정 삭제 규칙(set default)

### ● 변경 규칙

- ① 제한 변경 규칙(restrict)
- ② 다단계 변경 규칙(cascade)
- ③ NULL값 설정 변경 규칙(set NULL)
- ④ 기본값 설정 변경 규칙(set default)

\* set NULL의 규칙을 사용하여 외래키값이 NULL로 존재해도 참조 무결성 제약조건은 만족한다고 가정함.

## 4.4 데이터 무결성(무관계)

- 참조 무결성 문제(고아 데이터가 발생할 때)

① 새로운 튜플을 참조하는 테이블에 삽입할 때

학생(참조되는)

*학번	이름	학과	학년
20143054	유가인	기계	2
20151062	김인중	컴퓨터정보	1
20152088	조민우	전기전자	1
20153075	옥한빛	기계	1
20153088	이태연	기계	1

수강(참조하는)

*학번	*번호	성적
20152088	102	61
20152088	103	45

과목(참조되는)

*번호	교수명	과목명
101	김종영	컴퓨터개론
102	김태영	기계공학법
103	김유석	시스템분석
104	김명성	전자회로실험
105	최영민	CAD응용실습

insert  
into(수강)  
values(2015008, 105, 85);

2015008	105	85
---------	-----	----

insert

## 4.4 데이터 무결성(무관계)

### ● 참조 무결성 문제

② 참조하는 테이블의 외래키값을 변경할 때

학생(참조되는)

*학번	이름	학과	학년
20143054	유가인	기계	2
20151062	김인중	컴퓨터정보	1
20152088	조민우	전기전자	1
20153075	옥한빛	기계	1
20153088	이태연	기계	1

수강(참조하는)

*학번	*번호	성적
20152088	102	61
20152088	103	45

과목(참조되는)

*번호	교수명	과목명
101	김종영	컴퓨터개론
102	김태영	기계공학법
103	김유석	시스템분석
104	김명성	전자회로실험
105	최영민	CAD응용실습

update 수강  
set 학번 = 20151007  
where 학번 = 20152088;

update

20151007	101	95
----------	-----	----

## 4.4 데이터 무결성(무관계)

### ● 참조 무결성 문제

③ 참조되는 테이블의 튜플을 삭제할 때

학생(참조되는)

*학번	이름	학과	학년
20143054	유가인	기계	2
20151062	김인중	컴퓨터정보	1
<del>20152088</del>	<del>조민우</del>	<del>전기전자</del>	<del>1</del>
20153075	옥한빛	기계	1
20153088	이태연	기계	1

delete

수강(참조하는)

*학번	*번호	성적
20152088	102	61
20152088	103	45

과목(참조되는)

*번호	교수명	과목명
101	김종영	컴퓨터개론
102	김태영	기계공학법
103	김유석	시스템분석
104	김명성	전자회로실험
105	최영민	CAD응용실습

delete  
from 학생  
where 학번 = 20152088;

## 4.4 데이터 무결성(무관계)

### ● 참조 무결성 문제

④ 참조되는 테이블의 기본키 값을 변경할 때

학생(참조되는)

*학번	이름	학과	학년
20143054	유가인	기계	2
20151062	김인중	컴퓨터정보	1
20152088	조민우	전기전자	1
20153075	옥한빛	기계	1
20153088	이태연	기계	1

update

20151002	이육사	전기과	1
----------	-----	-----	---

수강(참조하는)

*학번	*번호	성적
20152088	102	61
20152088	103	45

과목(참조되는)

*번호	교수명	과목명
101	김종영	컴퓨터개론
102	김태영	기계공학법
103	김유석	시스템분석
104	김명성	전자회로실험
105	최영민	CAD응용실습

update 학생

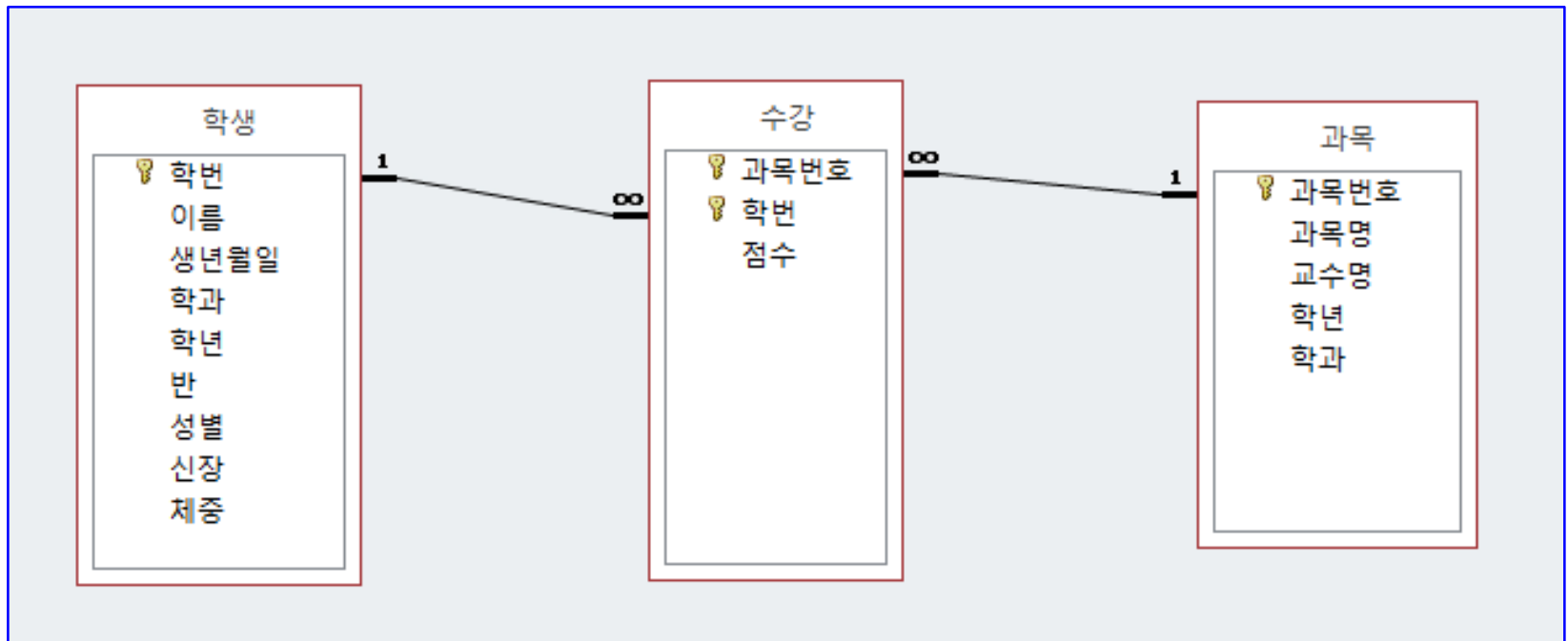
set 학번 = 20152088

where 학번 = 20151002;



## 4.4 데이터 무결성(참조무결성관계)

### ● 참조 무결성 관계 설정



## 4.4 데이터 무결성(참조무결성관계)

- 참조 무결성 문제(고아 데이터가 발생할 때)

① 새로운 튜플을 참조하는 테이블에 삽입할 때

학생(참조되는)

*학번	이름	학과	학년
20143054	유가인	기계	2
20151062	김인중	컴퓨터정보	1
20152088	조민우	전기전자	1
20153075	옥한빛	기계	1
20153088	이태연	기계	1

수강(참조하는)

*학번	*번호	성적
20152088	102	61
20152088	103	45

과목(참조되는)

*번호	교수명	과목명
101	김종영	컴퓨터개론
102	김태영	기계공학법
103	김유석	시스템분석
104	김명성	전자회로실험
105	최영민	CAD응용실습

insert  
into 수강  
values(105, 20151008, 85);

20151008	105	85
----------	-----	----

insert

## 4.4 데이터 무결성(참조무결성관계)

### ● 참조 무결성 문제

② 참조하는 테이블의 외래키값을 변경할 때

학생(참조되는)

*학번	이름	학과	학년
20143054	유가인	기계	2
20151062	김인중	컴퓨터정보	1
20152088	조민우	전기전자	1
20153075	옥한빛	기계	1
20153088	이태연	기계	1

수강(참조하는)

*학번	*번호	성적
20152088	102	61
20152088	103	45

과목(참조되는)

*번호	교수명	과목명
101	김종영	컴퓨터개론
102	김태영	기계공학법
103	김유석	시스템분석
104	김명성	전자회로실험
105	최영민	CAD응용실습

update 수강  
set 학번 = 20151007  
where 학번 = 20152088;

update

20151007	101	95
----------	-----	----

## 4.4 데이터 무결성(참조무결성관계)

### ● 참조 무결성 문제

③ 참조되는 테이블의 튜플을 삭제할 때

학생(참조되는)

*학번	이름	학과	학년
20143054	유가인	기계	2
20151062	김인중	컴퓨터정보	1
<del>20152088</del>	<del>조민우</del>	<del>전기전자</del>	<del>1</del>
20153075	옥한빛	기계	1
20153088	이태연	기계	1

delete

수강(참조하는)

*학번	*번호	성적
20152088	102	61
20152088	103	45

과목(참조되는)

*번호	교수명	과목명
101	김종영	컴퓨터개론
102	김태영	기계공학법
103	김유석	시스템분석
104	김명성	전자회로실험
105	최영민	CAD응용실습

delete  
from 학생  
where 학번 = 20152088;

## 4.4 데이터 무결성(참조무결성관계)

### ● 참조 무결성 문제

④ 참조되는 테이블의 기본키 값을 변경할 때

학생(참조되는)

*학번	이름	학과	학년
20143054	유가인	기계	2
20151062	김인중	컴퓨터정보	1
20152088	조민우	전기전자	1
20153075	옥한빛	기계	1
20153088	이태연	기계	1

update

20151007	이육사	전기과	1
----------	-----	-----	---

수강(참조하는)

*학번	*번호	성적
20152088	102	61
20152088	103	45

과목(참조되는)

*번호	교수명	과목명
101	김종영	컴퓨터개론
102	김태영	기계공학법
103	김유석	시스템분석
104	김명성	전자회로실험
105	최영민	CAD응용실습

update 학생

set 학번 = 20151007

where 학번 = 20152088;

4장을 마치며.....

질의 & 응답

