

## ■ 데이터 조작용어 (DML)



## ■ 데이터 삽입 : INSERT

```
INSERT INTO 테이블명  
[(속성 리스트)]  
VALUES  
(속성의 값 리스트);
```

- 속성 리스트를 생략하면 테이블을 정의할 때 지정한 속성의 순서대로 값 삽입
- VALUES 와 함께 삽입할 속성의 값을 입력
- INTO 의 속성과 VALUES 속성은 순서대로 1:1 대응되어야 됨

## ■ 데이터 삽입 : INSERT

### ● '부서' 테이블에 데이터 삽입

 부서번호	부서이름
1	인사부
2	연구부
3	홍보부

```
INSERT INTO 부서 (부서번호, 부서이름) VALUES (1, '인사부');  
INSERT INTO 부서 (부서번호, 부서이름) VALUES (2, '연구부');  
INSERT INTO 부서 (부서번호, 부서이름) VALUES (3, '홍보부');
```

### ● '사원' 테이블에 데이터 삽입

 사원번호	사원이름	 소속부서
1,001	AAA	3
1,002	BBB	2
1,003	CCC	1

```
INSERT INTO 사원 (사원번호, 사원이름, 소속부서) VALUES (1001, 'AAA', 3);  
INSERT INTO 사원 (사원번호, 사원이름, 소속부서) VALUES (1002, 'BBB', 2);  
INSERT INTO 사원 (사원번호, 사원이름, 소속부서) VALUES (1003, 'CCC', 1);
```

## ■ 데이터 삽입 : INSERT

### ● '고객' 테이블에 데이터 삽입

👉 고객아이디	고객이름	나이	등급	직업	적립금
apple	정소화	20	gold	학생	1,000
banana	김선우	25	vip	간호사	2,500
carrot	고명석	28	gold	교사	4,500
melon	성원용	35	gold	회사원	5,000
orange	김용욱	22	silver	학생	0
peach	오형준	(NULL)	silver	의사	300
pear	채광주	31	silver	회사원	500

```
INSERT INTO 고객 VALUES ('apple', '정소화', 20, 'gold', '학생', 1000);
INSERT INTO 고객 VALUES ('banana', '김선우', 25, 'vip', '간호사', 2500);
INSERT INTO 고객 VALUES ('carrot', '고명석', 28, 'gold', '교사', 4500);
INSERT INTO 고객 VALUES ('orange', '김용욱', 22, 'silver', '학생', 0);
INSERT INTO 고객 VALUES ('melon', '성원용', 35, 'gold', '회사원', 5000);
INSERT INTO 고객 VALUES ('peach', '오형준', NULL, 'silver', '의사', 300);
INSERT INTO 고객 VALUES ('pear', '채광주', 31, 'silver', '회사원', 500);
```

## ■ 데이터 삽입 : INSERT

### ● '제품' 테이블에 데이터 삽입

👉 제품번호	제품명	재고량	단가	제조업체
p01	그냥만두	5,000	4,500	대한식품
p02	매운짬면	2,500	5,500	민국푸드
p03	콩떡파이	3,600	2,600	한빛제과
p04	맛난초콜렛	1,250	2,500	한빛제과
p05	얼큰라면	2,200	1,200	대한식품
p06	통통우동	1,000	1,550	민국푸드
p07	달콤비스킷	1,650	1,500	한빛제과

```
INSERT INTO 제품 VALUES ('p01', '그냥만두', 5000, 4500, '대한식품');
INSERT INTO 제품 VALUES ('p02', '매운짬면', 2500, 5500, '민국푸드');
INSERT INTO 제품 VALUES ('p03', '콩떡파이', 3600, 2600, '한빛제과');
INSERT INTO 제품 VALUES ('p04', '맛난초콜렛', 1250, 2500, '한빛제과');
INSERT INTO 제품 VALUES ('p05', '얼큰라면', 2200, 1200, '대한식품');
INSERT INTO 제품 VALUES ('p06', '통통우동', 1000, 1550, '민국푸드');
INSERT INTO 제품 VALUES ('p07', '달콤비스킷', 1650, 1500, '한빛제과');
```

## ■ 데이터 삽입 : INSERT

### ● '주문' 테이블에 데이터 삽입

👉 주문번호	👉 주문고객	👉 주문제품	수량	배송지	주문일자
o01	apple	p03	10	서울시 마포구	2013-01-01 00:00:00
o02	melon	p01	5	인천시 계양구	2013-01-10 00:00:00
o03	banana	p06	45	경기도 부천시	2013-01-11 00:00:00
o04	carrot	p02	8	부산시 금정구	2013-02-01 00:00:00
o05	melon	p06	36	경기도 용인시	2013-02-20 00:00:00
o06	banana	p01	19	충청북도 보은군	2013-03-02 00:00:00
o07	apple	p03	22	서울시 영등포구	2013-03-15 00:00:00
o08	pear	p02	50	강원도 춘천시	2013-04-10 00:00:00
o09	banana	p04	15	전라남도 목포시	2013-04-11 00:00:00
o10	carrot	p03	20	경기도 안양시	2013-05-22 00:00:00

```
INSERT INTO 주문 VALUES ('o01', 'apple', 'p03', 10, '서울시 마포구', '2013-01-01');
INSERT INTO 주문 VALUES ('o02', 'melon', 'p01', 5, '인천시 계양구', '2013-01-10');
INSERT INTO 주문 VALUES ('o03', 'banana', 'p06', 45, '경기도 부천시', '2013-01-11');
INSERT INTO 주문 VALUES ('o04', 'carrot', 'p02', 8, '부산시 금정구', '2013-02-01');
INSERT INTO 주문 VALUES ('o05', 'melon', 'p06', 36, '경기도 용인시', '2013-02-20');
INSERT INTO 주문 VALUES ('o06', 'banana', 'p01', 19, '충청북도 보은군', '2013-03-02');
INSERT INTO 주문 VALUES ('o07', 'apple', 'p03', 22, '서울시 영등포구', '2013-03-15');
INSERT INTO 주문 VALUES ('o08', 'pear', 'p02', 50, '강원도 춘천시', '2013-04-10');
INSERT INTO 주문 VALUES ('o09', 'banana', 'p04', 15, '전라남도 목포시', '2013-04-11');
INSERT INTO 주문 VALUES ('o10', 'carrot', 'p03', 20, '경기도 안양시', '2013-05-22');
```

## ■ 데이터 조회 : SELECT

```
SELECT [DISTINCT] 속성 리스트  
FROM 테이블 리스트
```

- 검색하려는 속성의 이름 나열
- 검색하려는 속성이 있는 테이블의 이름 나열
- 검색 결과는 테이블 형태로 반환
- DISTINCT 사용 시 검색 결과 중 중복 튜플은 하나만 출력

## ■ 데이터 조회 : SELECT

- '고객' 테이블에서 고객아이디, 고객이름, 등급 속성 검색

고객 (3x7)		
고객아이디	고객이름	등급
apple	정소화	gold
banana	김선우	vip
carrot	고명석	gold
melon	성원용	gold
orange	김용욱	silver
peach	오형준	silver
pear	채광주	silver

```
SELECT 고객아이디, 고객이름, 등급  
FROM 고객;
```



## ■ 데이터 조회 : SELECT

### ● '고객' 테이블의 모든 속성 검색

고객 (6×7)					
고객아이디	고객이름	나이	등급	직업	적립금
apple	정소화	20	gold	학생	1,000
banana	김선우	25	vip	간호사	2,500
carrot	고명석	28	gold	교사	4,500
melon	성원용	35	gold	회사원	5,000
orange	김용욱	22	silver	학생	0
peach	오형준	(NULL)	silver	의사	300
pear	채광주	31	silver	회사원	500

```
SELECT 고객아이디, 고객이름, 나이, 등급, 직업, 적립금  
FROM 고객;
```

```
SELECT *  
FROM 고객;
```

## ■ 데이터 조회 : SELECT

- '제품' 테이블에서 제조업체 검색

제품 (1×7)
제조업체
대한식품
민국푸드
한빛제과
한빛제과
대한식품
민국푸드
한빛제과

```
SELECT 제조업체  
FROM 제품;
```

- '제품' 테이블에서 제조업체 중복없이 검색

제품 (1×3)
제조업체
대한식품
민국푸드
한빛제과

```
SELECT DISTINCT 제조업체  
FROM 제품;
```

## ■ 데이터 조회 : SELECT

### ● 속성명 변경 출력 : AS

```
SELECT 속성명 [AS] 변경속성  
FROM 테이블
```

- 테이블이 가지고 있는 기존 속성명을 지정한 속성명으로 변경하여 출력
- 변경하는 속성명에 공백이 포함되어 있으면 따옴표로 묶어줌
- AS 키워드 생략 가능

## ■ 데이터 조회 : SELECT

- '제품' 테이블에서 제품명과 단가를 검색, 단가를 가격으로 바꾸어 출력

제품 (2x7)	
제품명	가격
그냥만두	4,500
매운짬면	5,500
콩떡파이	2,600
맛난초콜렛	2,500
얼큰라면	1,200
통통우동	1,550
달콤비스켓	1,500

```
SELECT 제품명, 단가 AS '가격'  
FROM 제품;
```

## ■ 데이터 조회 : SELECT

### ● 산술식을 이용한 검색

```
SELECT 속성명 + 숫자, 속성명 * 숫자, ...  
FROM 테이블
```

- 속성명과 +, -, \*, / 등의 산술 연산자와 상수로 구성
- 실제 속성의 값은 변하지 않고 결과 테이블에서만 계산된 값이 출력

## ■ 데이터 조회 : SELECT

- '제품' 테이블에서 제품명과 단가를 검색,

단가에 500원을 더한 후 조정단가라는 속성명으로 바꾸어 출력

제품 (2x7)	
제품명	조정단가
그냥만두	5,000
매운짬면	6,000
콩떡파이	3,100
맛난초콜렛	3,000
얼큰라면	1,700
통통우동	2,050
달콤비스킷	2,000

```
SELECT 제품명, 단가 + 500 AS '조정단가'
FROM 제품;
```

## ■ 데이터 조회 : SELECT

### ● 조건에 해당하는 데이터 검색 : WHERE

```
SELECT [DISTINCT] 속성 리스트  
FROM 테이블 리스트  
[WHERE 조건];
```

- 비교 / 논리 연산자 등을 이용하여 검색 조건 작성
- 숫자, 문자, 날짜 모두 조건으로 사용 가능
- 문자 또는 날짜 값 사용시 따옴표로 묶어서 표현

종류	연산자	의미	연산자	의미	연산자	의미
비교	=	같다	<	작다	<=	작거나 같다
	<> 또는 !=	다르다	>	크다	>=	크거나 같다
논리	AND	모두 참	OR	1개 이상 참	NOT	거짓일때 참

## ■ 데이터 조회 : SELECT

- '제품' 테이블에서 한빛제과가 제조한 제품의 제품명, 재고량, 단가 검색

제품 (3×3)		
제품명	재고량	단가
쿵떡파이	3,600	2,600
맛난초콜렛	1,250	2,500
달콤비스킷	1,650	1,500

```
SELECT 제품명, 재고량, 단가  
FROM 제품  
WHERE 제조업체 = '한빛제과';
```



## ■ 데이터 조회 : SELECT

- '주문' 테이블에서 apple 고객아이디로 15개 이상 주문한  
주문제품, 수량, 주문일자 검색

주문 (3×1)		
주문제품	수량	주문일자
p03	22	2013-03-15 00:00:00

```
SELECT 주문제품, 수량, 주문일자  
FROM 주문  
WHERE 주문고객 = 'apple' AND 수량 >= 15;
```

## ■ 데이터 조회 : SELECT

- '주문' 테이블에서 apple 고객아이디로 주문했거나 15개 이상 주문된 제품의 주문제품, 수량, 주문일자, 주문고객 검색

주문 (3x8)		
주문제품	수량	주문일자
p03	10	2013-01-01 00:00:00
p06	45	2013-01-11 00:00:00
p06	36	2013-02-20 00:00:00
p01	19	2013-03-02 00:00:00
p03	22	2013-03-15 00:00:00
p02	50	2013-04-10 00:00:00
p04	15	2013-04-11 00:00:00
p03	20	2013-05-22 00:00:00

```
SELECT 주문제품, 수량, 주문일자  
FROM 주문  
WHERE 주문고객 = 'apple' OR 수량 >= 15;
```

## ■ 데이터 조회 : SELECT

- '제품' 테이블에서 단가가 2000원 이상이고 3000원 이하 인 제품의  
제품명, 단가, 제조업체 검색

제품 (3×2)		
제품명	단가	제조업체
쿵떡파이	2,600	한빛제과
맛난초콜렛	2,500	한빛제과

```
SELECT 제품명, 단가, 제조업체  
FROM 제품  
WHERE 단가 >= 2000 AND 단가 <= 3000;
```

```
SELECT 제품명, 단가, 제조업체  
FROM 제품  
WHERE 단가 BETWEEN 2000 AND 3000;
```

## ■ 데이터 조회 : SELECT

### ● 부분적으로 일치하는 데이터 검색 : LIKE

```
SELECT 속성 리스트  
FROM 테이블  
WHERE 속성명 LIKE '검색 데이터';
```

- 문자열을 이용하는 조건에만 적용 가능

기호	설명
%	0개 이상의 문자 (문자 내용과 개수 상관없이 모두 검색)
_ (underscore)	_ 개수 만큼의 문자 (문자 내용과 상관없이 모두 검색)

사용 예	설명
LIKE '데이터%'	데이터로 시작하는 문자열
LIKE '%데이터'	데이터로 끝나는 문자열
LIKE '%데이터%'	데이터가 포함된 문자열
LIKE '데이터__'	데이터로 시작하는 6자 길이의 문자열
LIKE '__한%'	세 번째 글자가 '한'인 문자열

## ■ 데이터 조회 : SELECT

- '고객' 테이블에서 성이 '김'인 고객의 고객이름, 나이, 등급, 적립금 검색

고객 (4x2)			
고객이름	나이	등급	적립금
김선우	25	vip	2,500
김용욱	22	silver	0

```
SELECT 고객이름, 나이, 등급, 적립금  
FROM 고객  
WHERE 고객이름 LIKE '김%';
```

## ■ 데이터 조회 : SELECT

- '고객' 테이블에서 고객아이디가 5자인 고객의 고객아이디, 고객이름, 등급 검색

고객 (3×3)		
고객아이디	고객이름	등급
apple	정소화	gold
melon	성원용	gold
peach	오형준	silver

```
SELECT 고객아이디, 고객이름, 등급  
FROM 고객  
WHERE 고객아이디 LIKE '_____';
```

## ■ 데이터 조회 : SELECT

### ● NULL 데이터 검색 : IS NULL / IS NOT NULL

```
SELECT 속성 리스트  
FROM 테이블  
WHERE 속성명 IS NULL;
```

- 특정 속성의 값이 NULL 값인지 비교
- IS 키워드를 사용하지 않고 비교연산자 이용 시 항상 거짓으로 출력

## ■ 데이터 조회 : SELECT

- '고객' 테이블에서 나이가 입력되지 않은 (NULL) 고객의 고객이름 검색

고객 (1×1)	
고객이름	
오형준	

```
SELECT 고객이름  
FROM 고객  
WHERE 나이 IS NULL;
```



## ■ 데이터 조회 : SELECT

- '고객' 테이블에서 나이가 입력되어 있는 고객의 고객이름 검색

고객 (1×6)
고객이름
정소화
김선우
고명석
성원용
김용욱
채광주

```
SELECT 고객이름  
FROM 고객  
WHERE 나이 IS NOT NULL;
```

## ■ 데이터 조회 : SELECT

### ● 데이터 검색 후 정렬 : ORDER BY

```
SELECT [DISTINCT] 속성 리스트  
FROM 테이블  
[WHERE 조건]  
[ORDER BY 속성 리스트 [ASC | DESC]];
```

- 검색 결과를 오름 / 내림차순으로 정렬하여 출력
- 오름차순 : ASC, 내림차순 : DESC
- 속성명을 여러 개 사용하는 경우 왼쪽 속성을 우선으로 정렬

## ■ 데이터 조회 : SELECT

- '고객' 테이블에서 고객이름, 등급, 나이를 나이의 내림차순으로 검색

고객 (3×7)		
고객이름	등급	나이
성원용	gold	35
채광주	silver	31
고명석	gold	28
김선우	vip	25
김용욱	silver	22
정소화	gold	20
오형준	silver	(NULL)

```
SELECT 고객이름, 등급, 나이  
FROM 고객  
ORDER BY 나이 DESC;
```

## ■ 데이터 조회 : SELECT

- '주문' 테이블에서 수량이 10개 이상인 주문의 주문고객, 수량, 주문일자를 검색  
단, 주문제품을 기준으로 오름차순 정렬하고 주문제품이 같은 경우  
수량을 기준으로 내림차순 정렬

주문 (4×8)			
주문고객	주문제품	수량	주문일자
banana	p01	19	2013-03-02 00:00:00
pear	p02	50	2013-04-10 00:00:00
apple	p03	22	2013-03-15 00:00:00
carrot	p03	20	2013-05-22 00:00:00
apple	p03	10	2013-01-01 00:00:00
banana	p04	15	2013-04-11 00:00:00
banana	p06	45	2013-01-11 00:00:00
melon	p06	36	2013-02-20 00:00:00

```
SELECT 주문고객, 주문제품, 수량, 주문일자  
FROM 주문  
WHERE 수량 >= 10  
ORDER BY 주문제품 ASC, 수량 DESC;
```