

9주차 1차시 데이터 검색 1

【학습목표】

1. 데이터 검색의 기본 형식에 대해 설명할 수 있다.
2. 조건을 이용한 데이터 검색을 방법을 설명할 수 있다.

학습내용1 : 데이터 조작어

1. SQL의 데이터 조작 기능

- 데이터 검색, 삽입, 수정, 삭제
 - * 데이터 검색 : SELECT
 - * 데이터 삽입 : INSERT
 - * 데이터 수정 : UPDATE
 - * 데이터 삭제 : DELETE

2. 질의 예제에 사용할 판매 데이터베이스

- Customer 테이블

Customer

<u>account</u>	name	grade	credit	address
bank	홍길동	Gold	30000	서울 종로구
apple	이남이	VIP	5000	경기 용인시
pencil	김돌	Silver	350	경기 수원
eagle	박세째	Gold	2450	부산 남구
bird	구선두	New	0	충남 천안시
king	오나라	Gold	15000	NULL

그림. 실습 질의용 Customer 릴레이션

- Items 테이블

Items

<u>code</u>	name	stocks	price
E01	김	30	1500
E02	단무지	45	3250
E03	햄	15	7000
E04	우유	25	4560
E05	주스	20	2300
E06	라면	30	2560
ZZZ	삭제상품	0	0

그림. 실습 질의용 Items 릴레이션

- Orders 테이블

Orders

<u>num</u>	customer	item	qty	date	saddr
00001	apple	E06	15	2015-06-01	경기 용인시
00002	apple	E05	2	2015-06-01	경기 용인시
00003	eagle	E01	3	2015-06-03	부산 남구
00004	bank	E04	5	2015-06-03	서울 종로구
00005	bird	E01	10	2015-06-04	충남 천안시
00006	bird	E04	1	2015-06-04	충남 천안시

그림. 실습 질의용 Orders 릴레이션

학습내용2 : 데이터 검색(SELECT 문)

1. 데이터 검색 (SELECT 문)의 개요

- . SELECT 키워드와 함께 검색하고 싶은 속성의 이름 나열
- . FROM 키워드와 함께 검색하고 싶은 속성이 있는 테이블의 이름 나열
- . 검색 결과 : 테이블 형태

- 형식

```
SELECT [ALL | DISTINCT] 속성_리스트
FROM 테이블리스트
[WHERE 검색조건]
[GROUP BY 속성이름]
[HAVING 검색조건]
[ORDER BY 속성이름 [ASC | DESC]];
```

2. 기본 검색

- 형식

- . SELECT [ALL | DISTINCT] 속성_리스트
FROM 테이블_리스트;

질의 1

- Customer 테이블에 존재하는 모든 속성을 검색하시오.

- . SELECT account, name, grade, credit, address FROM customer;

```
mysql> select account, name, grade, credit, address from customer;
```

account	name	grade	credit	address
apple	이남이	VIP	5000	경기 용인시
bank	홍길동	Gold	30000	서울 종로구
bird	구선두		0	충남 천안시
eagle	박세재	Gold	2450	부산 남구
king	오나라	Gold	15000	NULL
pencil	김돌	Silver	350	경기 수원

```
6 rows in set (0.00 sec)

mysql>
```

질의 2

- Customer 테이블에 존재하는 모든 속성을 검색하시오.
 - . *는 모든 속성을 나타냄 (속성의 순서가 테이블의 속성순서와 동일)
 - . SELECT * FROM customer

```
mysql> select * from customer;
```

account	name	grade	credit	address
apple	이남이	VIP	5000	경기 용인시
bank	홍길동	Gold	30000	서울 종로구
bird	구선두		0	충남 천안시
eagle	박세재	Gold	2450	부산 남구
king	오나라	Gold	15000	NULL
pencil	김돌	Silver	350	경기 수원

```
6 rows in set (0.00 sec)
```

질의 3

- Orders 테이블에서 주문한 item 만 검색하시오.
 - . SELECT item FROM orders

```
mysql> select item from orders;
```

item
E06
E05
E01
E04
E01
E04

```
6 rows in set (0.00 sec)
```

- 결과 테이블의 투플이 중복되어 있음.
 - . 관계 데이터 모델의 일반 릴레이션은 동일한 투플이 존재할 수 없음
 - . SELECT 문의 수행 결과로 반환되는 테이블에서는 동일한 투플이 중복될 수 있음
 - . 결과 테이블이 중복을 허용할 수 있도록 명시적으로 지시하기 위하여 ALL을 사용

질의 4

- Orders 테이블에서 주문한 item을 검색하되, 명시적으로 결과 튜플의 중복을 허용하시오.

. SELECT ALL item FROM orders;



```
mysql> select all item from orders;
+-----+
| item |
+-----+
| E06  |
| E05  |
| E01  |
| E04  |
| E01  |
| E04  |
+-----+
6 rows in set (0.00 sec)

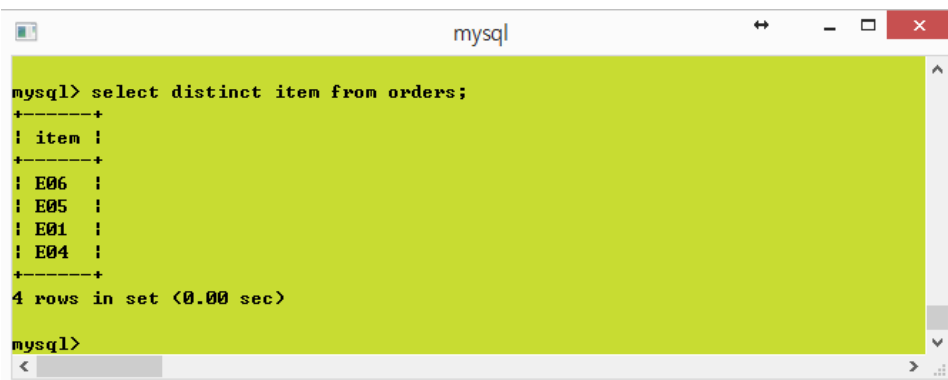
mysql>
```

질의 5

- Orders 테이블에서 주문한 item을 검색하되, 중복을 제거하고 한번씩만 출력되도록 하시오.

. 결과 테이블의 튜플 중복을 제거하기 위하여 DISTINCT를 사용

. SELECT DISTINCT item FROM orders;

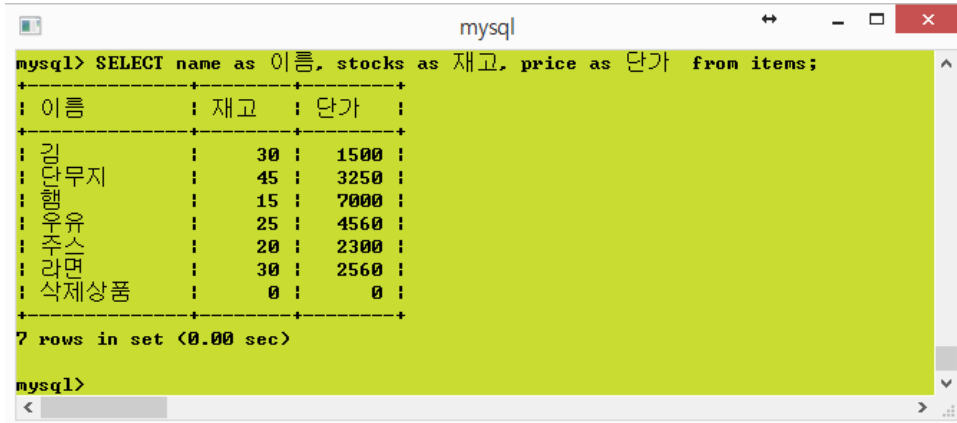


```
mysql> select distinct item from orders;
+-----+
| item |
+-----+
| E06  |
| E05  |
| E01  |
| E04  |
+-----+
4 rows in set (0.00 sec)

mysql>
```

질의 6

- items 테이블에서 name, stocks, price를 검색하되, 출력되는 이름으로 '이름', '재고', '단가'의 이름을 사용하시오.
- . SELECT name as 이름, stocks as 재고, price as 단가 from items;



```
mysql> SELECT name as 이름, stocks as 재고, price as 단가 from items;
```

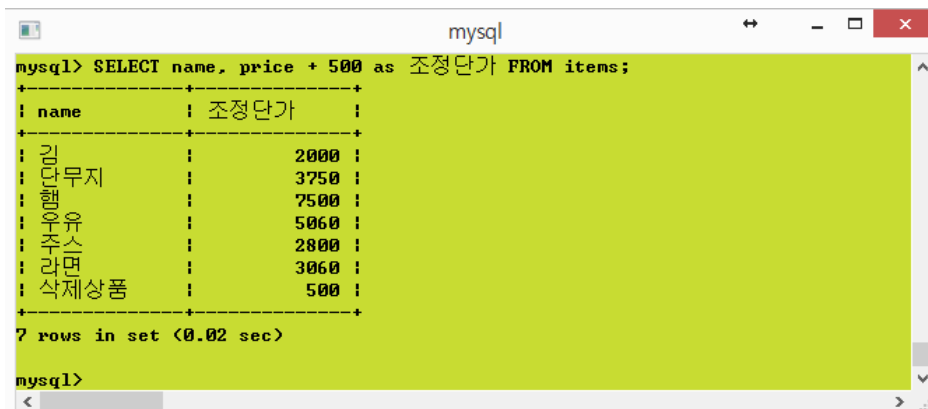
이름	재고	단가
김	30	1500
단무지	45	3250
햄	15	7000
우유	25	4560
주스	20	2300
라면	30	2560
삭제상품	0	0

7 rows in set (0.00 sec)

```
mysql>
```

질의 7

- SELECT 다음의 속성_리스트에 산술식을 제시할 수 있다.
 - 산술식 : 속성이름 +, -, *, / 등의 산술 연산자와 상수
 - items 테이블에서 name과 price 속성을 검색하되, price에 500원을 더해 조정단가라는 이름으로 출력하시오.
- . SELECT name, price + 500 as 조정단가 FROM items;



```
mysql> SELECT name, price + 500 as 조정단가 FROM items;
```

name	조정단가
김	2000
단무지	3750
햄	7500
우유	5060
주스	2800
라면	3060
삭제상품	500

7 rows in set (0.02 sec)

```
mysql>
```

3. 조건 검색

- 조건을 만족하는 데이터만 검색
- 형식

```
. SELECT [ ALL | DISTINCT ] 속성_리스트
  FROM 테이블_리스트
  [ WHERE 조건 ] ;
```

- 조건에 사용하는 비교 연산자

비교연산자	의미
=	같다
◇	다르다
<	작다
>	크다
<=	작거나 같다
>=	크거나 같다

- 조건에 사용하는 논리 연산자

논리연산자	의미
AND	모든 조건을 만족해야 검색한다.
OR	여러 조건 중 하나만 만족하면 검색한다.
NOT	조건을 만족하지 않는 것만 검색한다.

질의 8

- items 테이블에서 재고량 (stocks)이 20이상인 제품의 code, name, stocks를 검색하시오.

```
SELECT code, name, stocks
FROM items
WHERE stocks >= 20;
```

```
mysql> SELECT code, name, stocks
-> FROM items
-> WHERE stocks >= 20;

+----+-----+-----+
| code | name  | stocks |
+----+-----+-----+
| E01  | 김다무지 | 30    |
| E02  | 유유스  | 45    |
| E04  | 추스  | 25    |
| E05  | 라면  | 20    |
| E06  | 라면  | 30    |
+----+-----+-----+
5 rows in set (0.01 sec)

mysql>
```

질의 9

- Orders 테이블에서 apple 계정의 고객의 10 개 이상 주문한 item 과 qty, date를 검색하시오.

```
SELECT item, qty, date
FROM orders
WHERE customer = 'apple' AND qty >= 10 ;
```

```
mysql> SELECT item, qty, date
-> FROM orders
-> WHERE customer='apple' AND qty >= 10 ;

+----+-----+-----+
| item | qty | date       |
+----+-----+-----+
| E06  | 15 | 2015-06-01 |
+----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

질의 10

- Orders 테이블에서 apple 고객이 주문했거나 10 개 이상 주문된 customer, item, qty, date를 검색하시오.

```
SELECT customer, item, qty, date
FROM orders
WHERE customer = 'apple' OR qty >= 10 ;
```

```
mysql> SELECT customer, item, qty, date
-> FROM orders
-> WHERE customer='apple' OR qty >= 10 ;

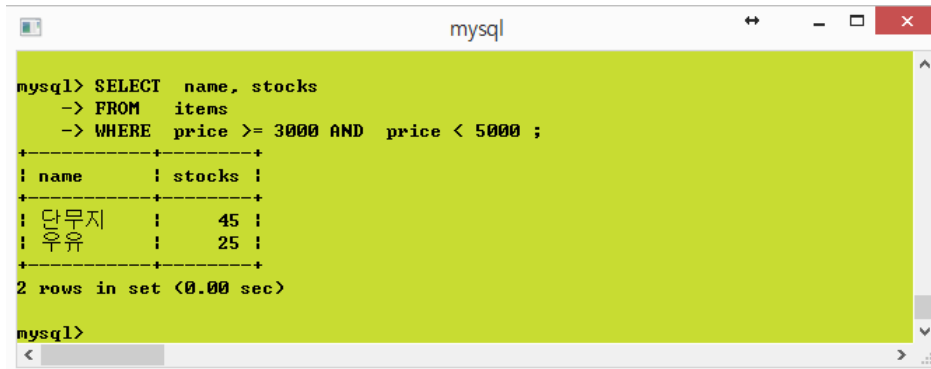
+-----+-----+-----+-----+
| customer | item | qty | date       |
+-----+-----+-----+-----+
| apple    | E06  | 15 | 2015-06-01 |
| apple    | E05  | 2  | 2015-06-01 |
| bird     | E01  | 10 | 2015-06-04 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```


질의 11

- Items 테이블에서 단가(price)가 3000원 이상이고 5000원 미만인 제품의 이름(name)과, 재고량(stocks)를 검색하시오.

```
SELECT name, stocks
FROM items
WHERE price >= 3000 AND price < 5000 ;
```



```
mysql> SELECT name, stocks
-> FROM items
-> WHERE price >= 3000 AND price < 5000 ;
+-----+-----+
| name  | stocks |
+-----+-----+
| 단무지 | 45     |
| 우유   | 25     |
+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

【학습정리】

1. 데이터베이스에서 데이터를 검색하기 위하여 SELECT 문을 사용한다.
2. 조건을 사용하여 검색하고자 할 경우, “SELECT 속성_리스트 FROM 테이블명 WHERE 조건”을 사용한다.