

9주차 3차시 데이터 검색 3

【학습목표】

1. 검색 결과 테이블의 투플을 사용자가 원하는 순서로 출력하는 방법에 대하여 설명할 수 있다.
2. 집계함수를 사용하여 속성 값에 대하여 통계적으로 계산하는 방법을 설명할 수 있다.

학습내용1 : 정렬 검색

- SELECT 문의 검색 결과 테이블의 투플 순서는 DBMS에서 정한 순서로 출력됨
- 결과 테이블의 내용을 사용자가 원하는 순서로 출력하는 방법
==> ORDER BY 키워드 사용
- 기본 형식

```
SELECT [ ALL | DISTINCT ] 속성_리스트  
FROM   테이블_리스트  
[ WHERE 조건 ]  
[ ORDER BY 속성_리스트 [ ASC | DESC ] ];
```

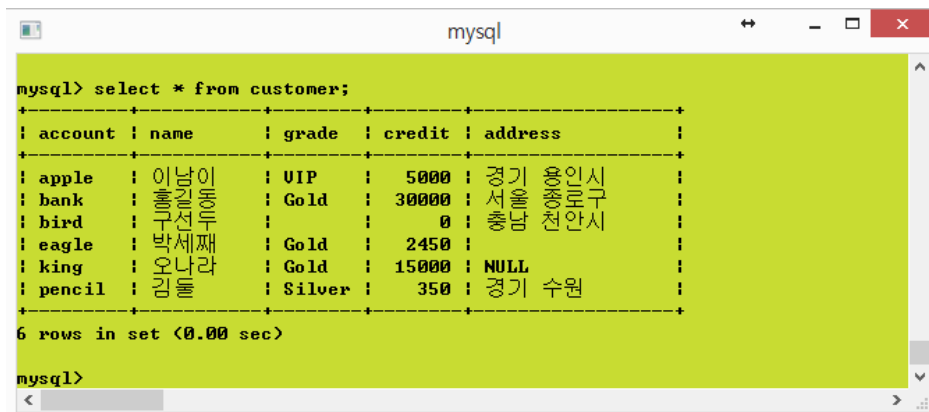
- . ORDER BY 속성_리스트 : 정렬 기준이 되는 속성을 지정
 - * 여러 기준에 따라 정렬하기 위하여 정렬 기준이 되는 속성을 차례로 제시
- . ASC : 오름차순 정렬
- . DESC : 내림차순 정렬
- . 기본 정렬 방식 - 오름차순 정렬
- 정렬
 - . 오름차순 정렬 : 알파벳 순서 / 사전 순서, ASCII 코드의 순서
 - . 내림차순 정렬 : 오름차순의 반대
 - . NULL 값의 출력 위치
 - * 오름차순 - 첫 번째 출력
 - * 내림차순 - 맨 마지막에 출력

1. 질의 1

- Customer 테이블에서 계정(account), 이름(name), 등급(grade)을 검색하되, 계정(account)을 내림차순으로 정렬하시오.

```
SELECT  account, name, grade
FROM    customer
ORDER BY account DESC;
```

* Customer 테이블의 데이터



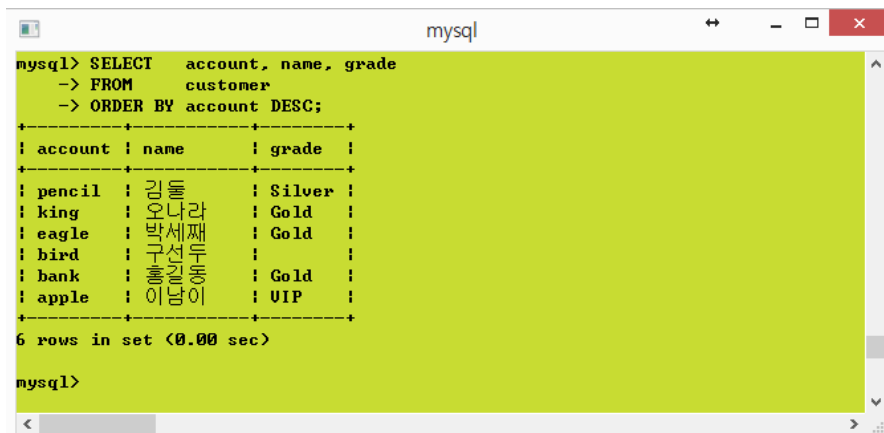
```
mysql> select * from customer;
```

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

```
6 rows in set (0.00 sec)

mysql>
```

* 질의 결과 테이블



```
mysql> SELECT  account, name, grade
-> FROM    customer
-> ORDER BY account DESC;
```

account	name	grade
pencil	김돌	Silver
king	오나라	Gold
eagle	박세째	Gold
bird	구선두	
bank	홍길동	Gold
apple	이남이	VIP

```
6 rows in set (0.00 sec)

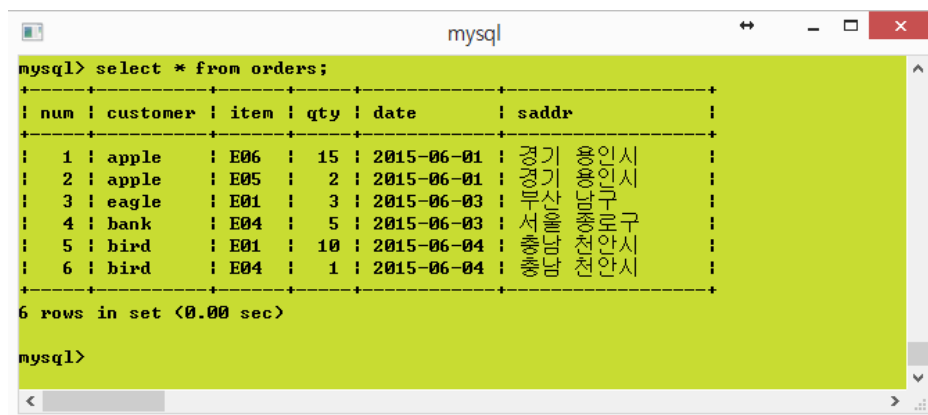
mysql>
```

2. 질의 2

- Orders 테이블에서 주문 수량(qty)이 10개 미만인 주문의 고객계정(customer), 제품코드(item), 수량(qty), 주문일자를 검색하시오. 단, 제품코드를 기준으로 오름차순으로 정렬하고, 동일 제품은 수량을 기준으로 내림차순으로 정렬하시오.

```
SELECT    customer, item, qty, date
FROM      orders
WHERE     qty < 10
ORDER BY  item ASC, qty DESC;
```

* Orders 테이블의 데이터



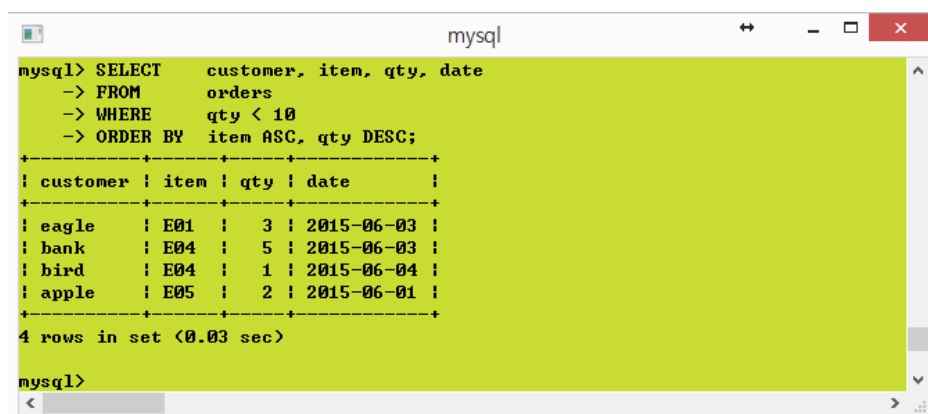
```
mysql> select * from orders;
```

num	customer	item	qty	date	saddr
1	apple	E06	15	2015-06-01	경기 용인시
2	apple	E05	2	2015-06-01	경기 용인시
3	eagle	E01	3	2015-06-03	부산광역시
4	bank	E04	5	2015-06-03	서울특별시
5	bird	E01	10	2015-06-04	충남 천안시
6	bird	E04	1	2015-06-04	충남 천안시

6 rows in set (0.00 sec)

```
mysql>
```

* 질의 결과 테이블



```
mysql> SELECT    customer, item, qty, date
-> FROM      orders
-> WHERE     qty < 10
-> ORDER BY  item ASC, qty DESC;
```

customer	item	qty	date
eagle	E01	3	2015-06-03
bank	E04	5	2015-06-03
bird	E04	1	2015-06-04
apple	E05	2	2015-06-01

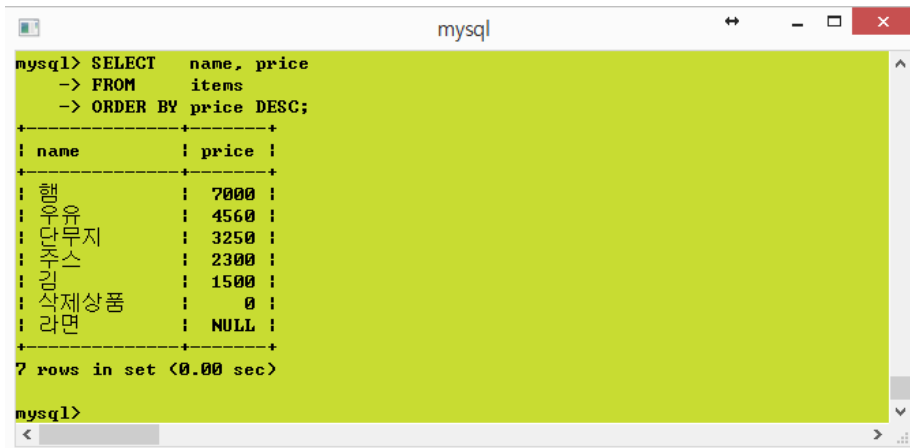
4 rows in set (0.03 sec)

```
mysql>
```

3. 질의 3

- Items 테이블의 품명(name) 과 가격(price)를 검색하되, 가격(price)를 내림차순으로 검색하여 NULL의 위치를 확인하고, 다시 오름차순으로 검색하여 NULL의 위치를 확인하시오.

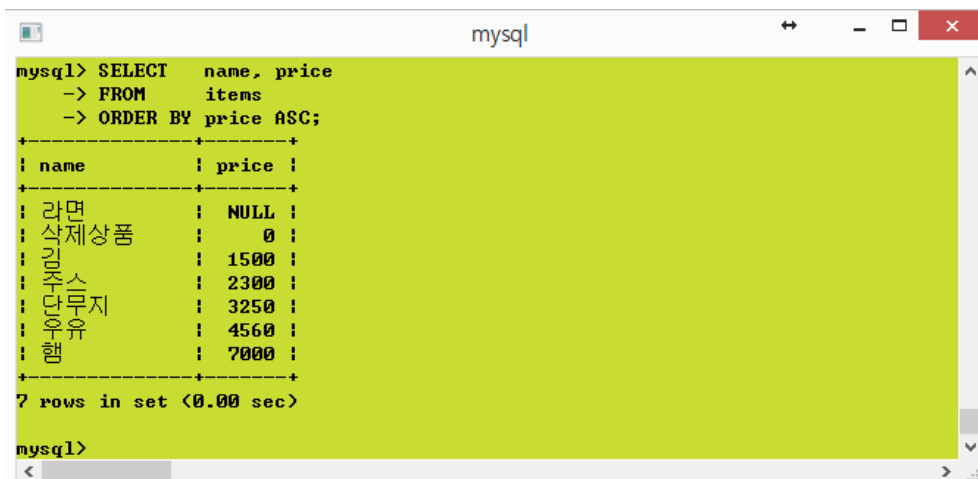
```
SELECT  name, price
FROM    items
ORDER BY price DESC;
```



```
mysql> SELECT  name, price
-> FROM    items
-> ORDER BY price DESC;
+-----+-----+
| name  | price |
+-----+-----+
| 햄    | 7000  |
| 우유  | 4560  |
| 단무지 | 3250  |
| 주스  | 2300  |
| 김    | 1500  |
| 삭제상품 | 0     |
| 라면  | NULL  |
+-----+-----+
7 rows in set (0.00 sec)

mysql>
```

```
SELECT  name, price
FROM    items
ORDER BY price ASC;
```



```
mysql> SELECT  name, price
-> FROM    items
-> ORDER BY price ASC;
+-----+-----+
| name  | price |
+-----+-----+
| 라면  | NULL  |
| 삭제상품 | 0     |
| 김    | 1500  |
| 주스  | 2300  |
| 단무지 | 3250  |
| 우유  | 4560  |
| 햄    | 7000  |
+-----+-----+
7 rows in set (0.00 sec)

mysql>
```

학습내용2 : 집계 함수를 이용한 검색

- 특정 속성 값을 통계적으로 계산한 결과를 검색
- 집계 함수 (Aggregate Function)를 사용
 - . 열 함수 (Column Function)라고도 함
 - . 개수, 합계, 평균, 최댓값, 최솟값의 계산을 제공
- 집계 함수

함수	의미	사용 가능한 데이터 타입
COUNT	속성 값의 개수	모든 데이터
MAX	속성 값의 최댓값	
MIN	속성 값의 최솟값	
SUM	속성 값의 합계	숫자 데이터
AVG	속성 값의 평균	

[주의 사항]

- NULL 인 속성 값은 제외하고 계산함.
- WHERE 절에서는 사용할 수 없고 SELECT 절과 HAVING 절에서만 사용할 수 있음.

1. 질의 1 : AVG 집계 함수

- Items 테이블에서 모든 제품의 평균 단가를 검색하시오.
- Items 테이블의 데이터

```
mysql> SELECT * FROM items;
```

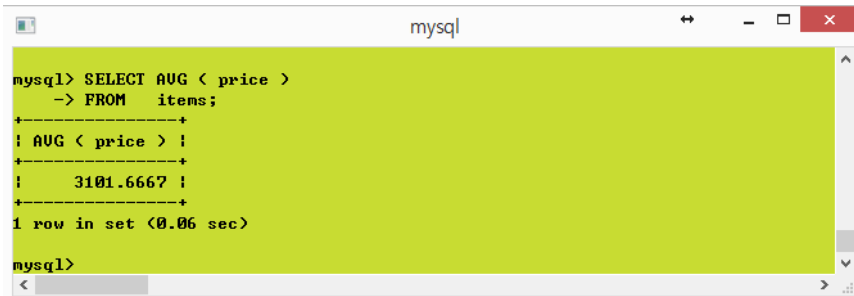
code	name	stocks	price
E01	김	30	1500
E02	단무지	45	3250
E03	햄	15	7000
E04	우유	25	4560
E05	주스	20	2300
E06	라면	30	NULL
ZZZ	삭제상품	0	0

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

```
mysql>
```

- 질의문

```
SELECT AVG ( price )
FROM items;
```



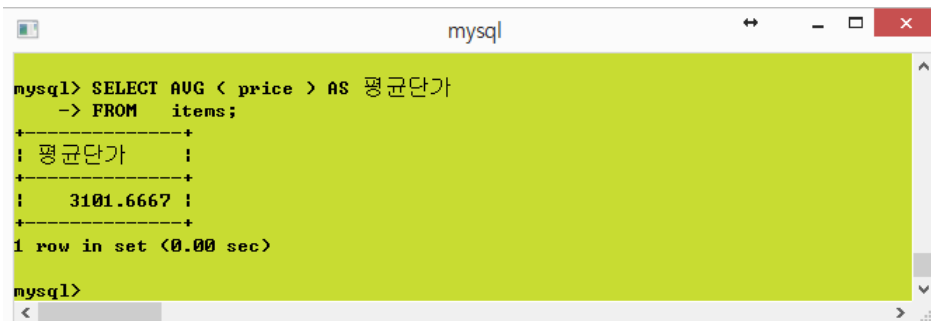
```
mysql> SELECT AVG ( price )
-> FROM items;
+-----+
| AVG ( price ) |
+-----+
| 3101.6667 |
+-----+
1 row in set (0.06 sec)

mysql>
```

- * 속성이름으로 AVG(price)을 사용함.
- * AS 키워드를 사용해 이름을 부여
. 새이름에 공란이 포함된 경우 작은따옴표로 묶는다.

- 질의문

```
SELECT AVG ( price ) AS 평균단가
FROM items;
```

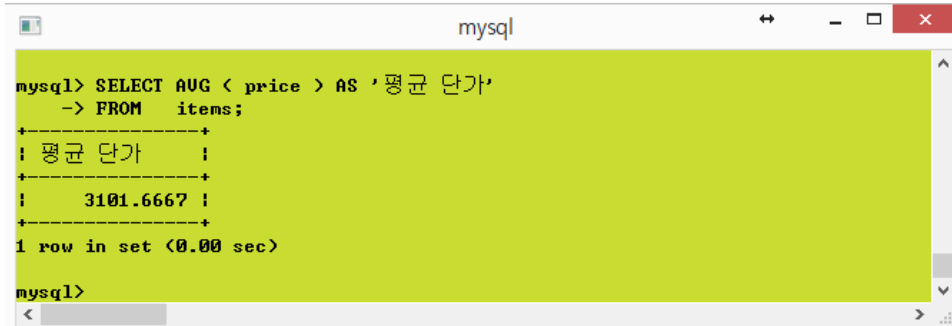


```
mysql> SELECT AVG ( price ) AS 평균단가
-> FROM items;
+-----+
| 평균단가 |
+-----+
| 3101.6667 |
+-----+
1 row in set (0.00 sec)

mysql>
```

- 질의문

```
SELECT AVG ( price ) AS '평균 단가'
FROM items;
```

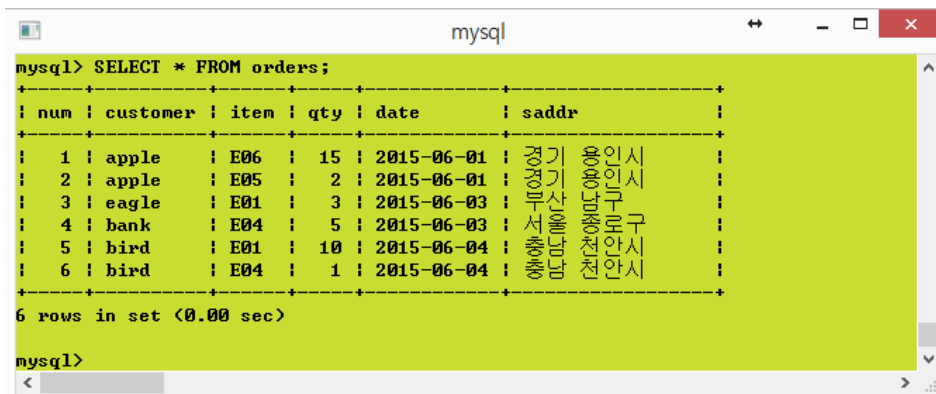


```
mysql> SELECT AVG ( price ) AS '평균 단가'
-> FROM items;
+-----+
| 평균 단가 |
+-----+
| 3101.6667 |
+-----+
1 row in set (0.00 sec)

mysql>
```

2. 질의 2 : SUM 집계 함수

- Orders 테이블에서 2015-06-01 부터 2015-06-02 까지 주문한 제품 개수의 합계를 검색하시오.
- Orders 테이블 데이터



```
mysql> SELECT * FROM orders;
+----+-----+-----+-----+-----+-----+
| num | customer | item | qty | date       | saddr |
+----+-----+-----+-----+-----+-----+
| 1 | apple | E06 | 15 | 2015-06-01 | 경기 용인시 |
| 2 | apple | E05 | 2 | 2015-06-01 | 경기 용인시 |
| 3 | eagle | E01 | 3 | 2015-06-03 | 부산 구로구 |
| 4 | bank | E04 | 5 | 2015-06-03 | 서울 용인시 |
| 5 | bird | E01 | 10 | 2015-06-04 | 충남 천안시 |
| 6 | bird | E04 | 1 | 2015-06-04 | 충남 천안시 |
+----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql>
```

- 질의문

```
SELECT SUM( qty ) AS '총 주문 수량'
FROM orders
WHERE date >= '2015-06-01' AND date <= '2015-06-02';
```



```
mysql> SELECT SUM( qty ) AS '총 주문 수량'
-> FROM orders
-> WHERE date >= '2015-06-01' AND date <= '2015-06-02';
+-----+
| 총 주문 수량 |
+-----+
|          17 |
+-----+
1 row in set (0.00 sec)

mysql>
```

3. 질의 3 : COUNT 집계 함수

- Customer 테이블에 등록되어 있는 고객의 수를 검색하시오.
- COUNT 함수
 - . 테이블의 모든 속성에 적용하여 개수를 계산할 수 있음.
 - . NULL 값은 제외 시킴
 - . 중복된 값도 모두 별도 계산함
 - . 기본키 속성이나 * 를 이용해 개수를 계산함
- Customer 테이블의 데이터



```
mysql> SELECT * FROM customer;
+-----+-----+-----+-----+-----+
| account | name   | grade | credit | address |
+-----+-----+-----+-----+-----+
| apple   | 이남이 | VIP   | 5000   | 경기 용인시 |
| bank    | 홍길동 | Gold  | 30000  | 서울 충로구 |
| bird    | 구선두 | NULL  | 0      | 충남 천안시 |
| eagle   | 박세재 | Gold  | 2450   |             |
| king    | 오나라 | Gold  | 15000  | NULL      |
| pencil  | 김돌   | Silver | 350    | 경기 수원   |
+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql>
```


- 계정(account) 속성을 이용해 계산하는 경우

```
SELECT COUNT( account ) AS 고객수
FROM customer;
```

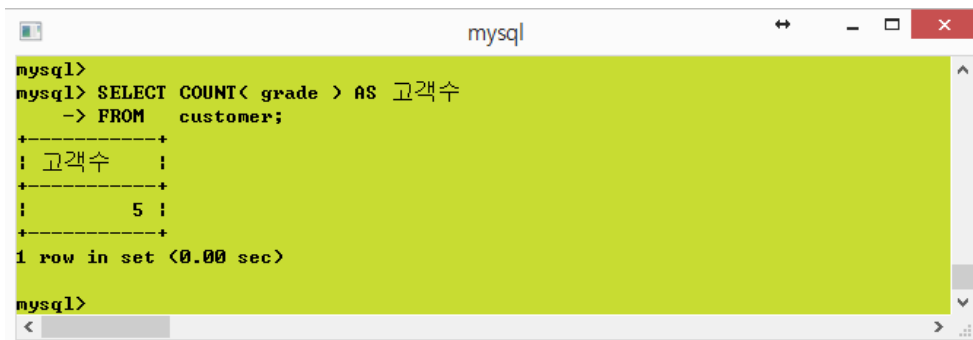


```
mysql> SELECT COUNT( account ) AS 고객수
-> FROM customer;
+-----+
| 고객수 |
+-----+
|      6 |
+-----+
1 row in set (0.00 sec)

mysql>
```

- 등급(grade) 속성을 이용해 계산하는 경우

```
SELECT COUNT( grade ) AS 고객수
FROM customer;
```

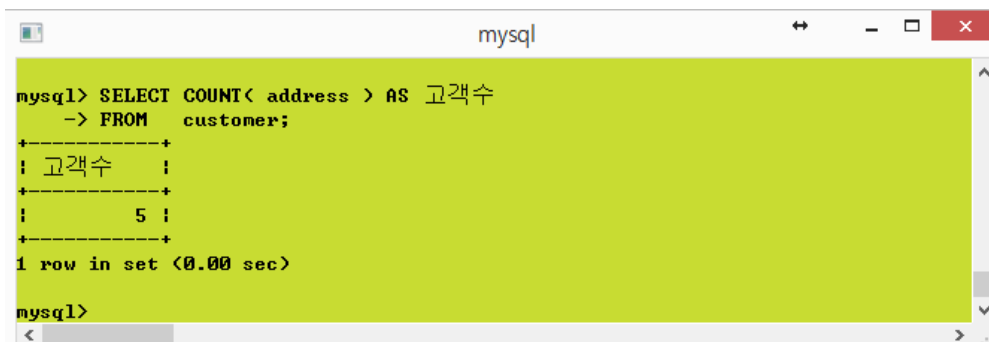


```
mysql> SELECT COUNT( grade ) AS 고객수
-> FROM customer;
+-----+
| 고객수 |
+-----+
|      5 |
+-----+
1 row in set (0.00 sec)

mysql>
```

- 주소(address) 속성을 이용해 계산하는 경우

```
SELECT COUNT( address ) AS 고객수
FROM customer;
```

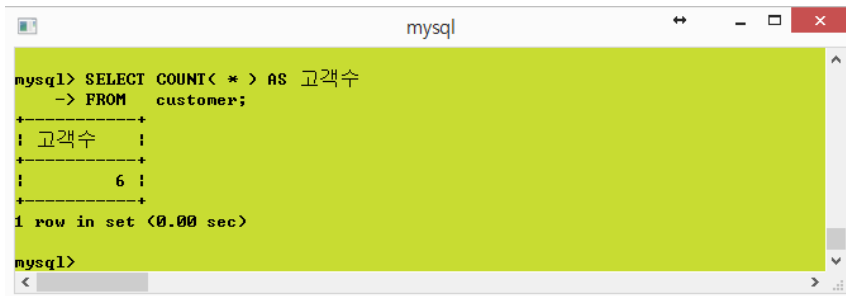


```
mysql> SELECT COUNT( address ) AS 고객수
-> FROM customer;
+-----+
| 고객수 |
+-----+
|      5 |
+-----+
1 row in set (0.00 sec)

mysql>
```

- * 를 이용해 계산하는 경우
 - . * 는 모든 속성을 의미하는 기호
 - . 모든 속성 값으로 구성된 튜플을 대상으로 개수를 계산
- 질의문

```
SELECT COUNT( * ) AS 고객수
FROM customer;
```

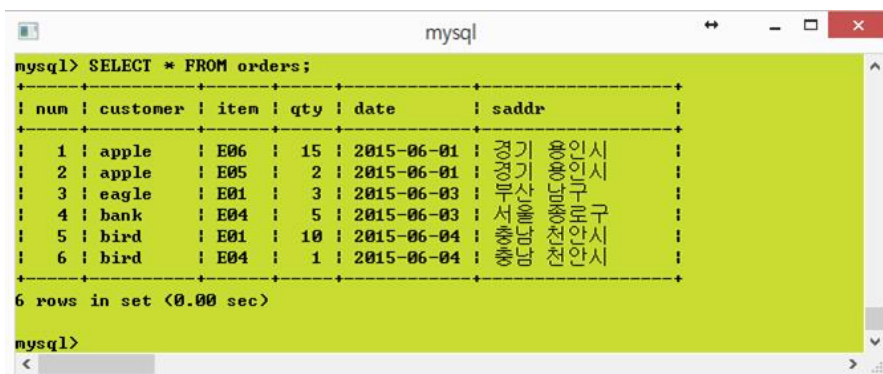


```
mysql> SELECT COUNT( * ) AS 고객수
-> FROM customer;
+-----+
| 고객수 |
+-----+
|      6 |
+-----+
1 row in set (0.00 sec)

mysql>
```

4. 질의 4

- Orders 테이블에서 주문된 제품(item)의 수를 검색하시오.
- Orders 테이블 데이터

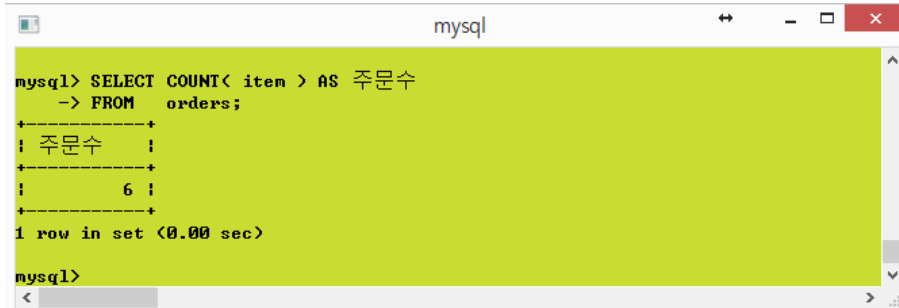


```
mysql> SELECT * FROM orders;
+----+-----+-----+----+-----+-----+
| num | customer | item | qty | date       | saddr |
+----+-----+-----+----+-----+-----+
| 1   | apple    | E06  | 15  | 2015-06-01 | 경기 용인시 |
| 2   | apple    | E05  | 2   | 2015-06-01 | 경기 용인시 |
| 3   | eagle    | E01  | 3   | 2015-06-03 | 부산 강서구 |
| 4   | bank     | E04  | 5   | 2015-06-03 | 서울 중랑구 |
| 5   | bird     | E01  | 10  | 2015-06-04 | 충남 천안시 |
| 6   | bird     | E04  | 1   | 2015-06-04 | 해운대 천안시 |
+----+-----+-----+----+-----+-----+
6 rows in set (0.00 sec)

mysql>
```

- 질의문

```
SELECT COUNT( item ) AS 주문수
FROM orders;
```

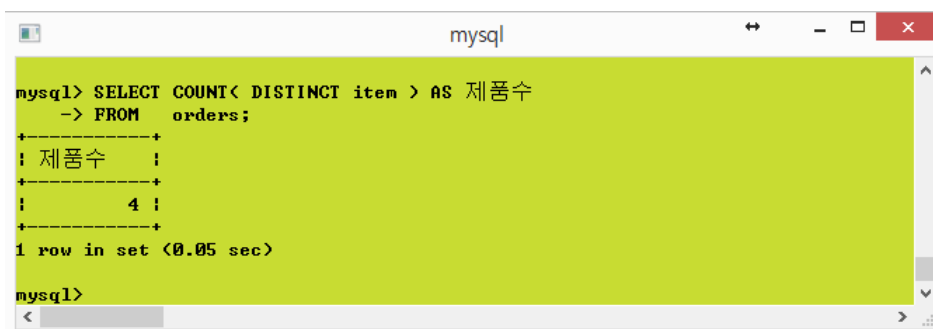


```
mysql> SELECT COUNT( item ) AS 주문수
-> FROM orders;
+-----+
| 주문수 |
+-----+
|      6 |
+-----+
1 row in set (0.00 sec)

mysql>
```

- 질의문

```
SELECT COUNT( DISTINCT item ) AS 제품수
FROM orders;
```



```
mysql> SELECT COUNT( DISTINCT item ) AS 제품수
-> FROM orders;
+-----+
| 제품수 |
+-----+
|      4 |
+-----+
1 row in set (0.05 sec)

mysql>
```

학습내용3 : 정렬 검색과 집계함수 정리

- 정렬 검색 기본 형식

```
SELECT [ ALL | DISTINCT ] 속성_리스트
FROM 테이블_리스트
[ WHERE 조건 ]
[ ORDER BY 속성_리스트 [ ASC | DESC ] ];
```

- 집계 함수를 이용한 검색

함수	의미	사용 가능한 데이터 타입
COUNT	속성 값의 개수	모든 데이터
MAX	속성 값의 최댓값	
MIN	속성 값의 최솟값	
SUM	속성 값의 합계	숫자 데이터
AVG	속성 값의 평균	

- . NULL 인 속성 값은 제외하고 계산함.
- . WHERE 절에서는 사용할 수 없고 SELECT 결과 HAVING 절에서만 사용할 수 있음.

【학습정리】

1. ORDER BY 키워드는 결과 테이블의 내용을 사용자가 원하는 순서로 출력하기 위하여 사용한다.
2. 집계함수는 속성 값의 개수, 합계, 평균, 최댓값, 최솟값을 계산하기 위하여 사용한다.