

13주차 2차시 확장된 조인 검색

【학습목표】

1. 내부 조인의 의미를 이해하고, 실습을 통하여 설명할 수 있다.
2. 오른쪽 조인의 의미를 이해하고, 실습을 통하여 설명할 수 있다.
3. 왼쪽 조인의 의미를 이해하고, 실습을 통하여 설명할 수 있다.
4. 전체 조인의 의미를 이해하고, 실습을 통하여 설명할 수 있다.

학습내용1 : 내부 조인과 오른쪽 조인

- 조인 검색의 종류
 - . 내부조인 (Inner Join)
 - . 오른쪽 조인 (Right Join)
 - . 왼쪽 조인 (Left Join)
 - . 전체 조인 (Full Join)

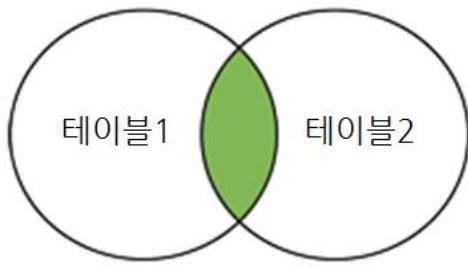
1. 내부 조인 (Inner Join)

- 두 테이블에서 속성이 일치하는 모든 튜플을 반환
- 일반적인 조인과 동일
- 기본 형식

```
SELECT      속성_리스트
FROM        테이블1
INNER JOIN   테이블2
ON          테이블1.속성 = 테이블2.속성
[ ORDER BY 속성 ];
```

또는

```
SELECT      속성_리스트
FROM        테이블1
JOIN        테이블2
ON          테이블1.속성 = 테이블2.속성
[ ORDER BY 속성 ];
```



질의 1 : 주문을 한 고객의 이름(customer.name)과 주문 상품(orders.item)을 검색하시오.

- 필요한 테이블 : customer, orders
- 검색 조건 : customer.account = orders.customer
- 질의문

```
SELECT      customer.name, orders.item
FROM        customer
INNER JOIN  orders
ON          customer.account=orders.customer;
```

- Customer 테이블

MySQL

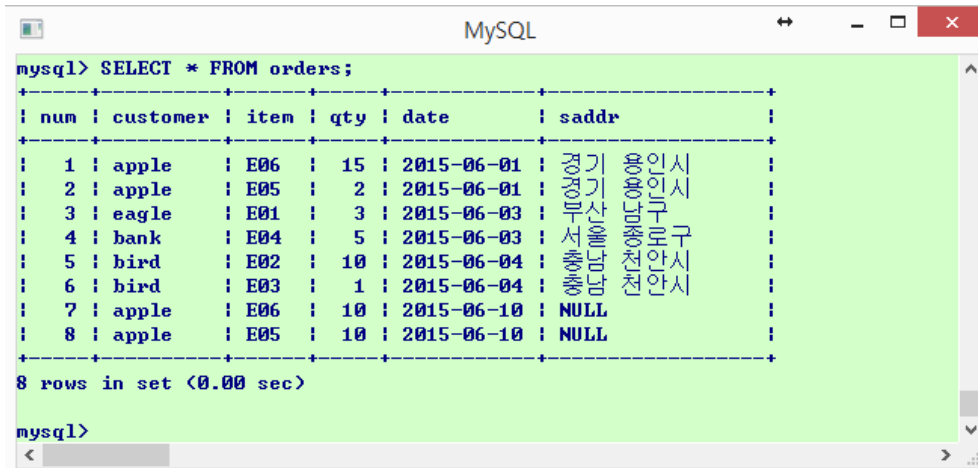
```
mysql> select * from customer;
```

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
moon	한가위	New	1000	서울 용산구
pencil	김돌	Silver	350	경기 수원시
sun	박태양	New	1000	용인 처인구
table	정탁자	New	0	NULL
watch	김정확	New	NULL	NULL

10 rows in set (0.00 sec)

```
mysql>
```

- Orders 테이블

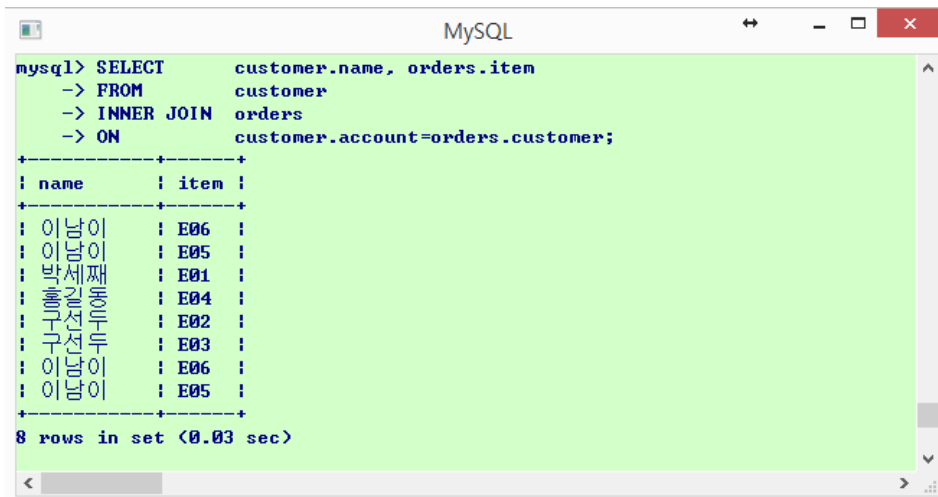


```
mysql> SELECT * FROM orders;
```

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	E02	10	2015-06-04	충청남도
6	bird	E03	1	2015-06-04	충남 천안시
7	apple	E06	10	2015-06-10	NULL
8	apple	E05	10	2015-06-10	NULL

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

- 질의 실행 결과



```
mysql> SELECT customer.name, orders.item
-> FROM customer
-> INNER JOIN orders
-> ON customer.account=orders.customer;
```

name	item
이남이	E06
이남이	E05
박세재	E01
홍길동	E04
구선두	E02
구선두	E03
이남이	E06
이남이	E05

```
8 rows in set (0.03 sec)
```

질의 2 : 주문된 상품(items.name)과 주문 수량(orders.qty)를 검색하시오.

- 필요한 테이블 : Items, Orders
- 검색 조건 : items.code=orders.item
- 질의문

```
SELECT items.name, orders.qty
FROM items
INNER JOIN orders
ON items.code=orders.item;
```

- Items 테이블

MySQL

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

code	name	stocks	price
E01	김	30	1575
E02	단무지	45	3413
E03	햄	15	7350
E04	우유	25	4788
E05	주스	20	2415
E06	라면	30	2688
ZZZ	삭제상품	0	0

7 rows in set (0.00 sec)

```
mysql>
```

- Orders 테이블

MySQL

```
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 동대문구
5	bird	E02	10	2015-06-04	충남 천안시
6	bird	E03	1	2015-06-04	충남 천안시
7	apple	E06	10	2015-06-10	NULL
8	apple	E05	10	2015-06-10	NULL

8 rows in set (0.00 sec)

```
mysql>
```

- 질의 실행 결과

MySQL

```
mysql> SELECT
-> FROM      items
-> INNER JOIN orders
-> ON        items.code=orders.item;
```

name	qty
라면	15
주스	2
김	3
우유	5
단무지	10
햄	1
라면	10
주스	10

8 rows in set (0.00 sec)

2. 오른쪽 조인 (Right Join)

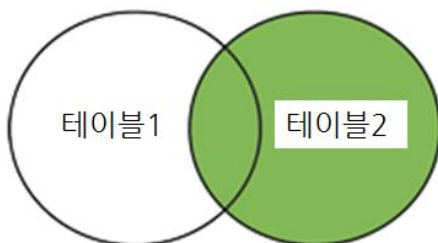
- 왼쪽 테이블(테이블1)과 매칭되는 오른쪽 테이블(테이블2)의 모든 튜플을 반환한다.
- 왼쪽 테이블과 매칭이 되지 않을 경우, NULL을 반환된다.

- 기본 형식

```
SELECT      속성_리스트
FROM        테이블1
RIGHT JOIN   테이블2
ON          테이블1.속성 = 테이블2.속성
[ ORDER BY 속성 ];
```

또는

```
SELECT      속성_리스트
FROM        테이블1
RIGHT OUTER JOIN   테이블2
ON          테이블1.속성 = 테이블2.속성
[ ORDER BY 속성 ];
```



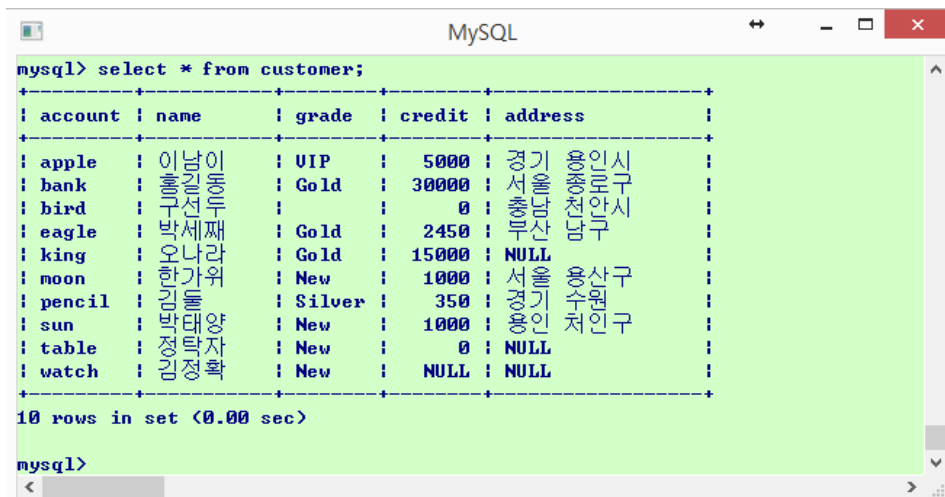
질의 3 : 고객의 구매 현황을 조사하되, 구매하지 않은 고객은 구매 항목에 NULL로 표시하고, 구매한 고객은 구매한 상품코드를 조사하시오. (오른쪽 조인 사용)

- 필요한 테이블 : Customer, Orders
- 고객의 모든 투플이 결과 테이블에 나타나야 하므로, 오른쪽 테이블은 Customer
- 검색조건 : customer.account = orders.customer

- 질의문

```
SELECT      customer.name, orders.item
FROM        orders
RIGHT JOIN   customer
ON          customer.account=orders.customer;
```

- Customer 테이블

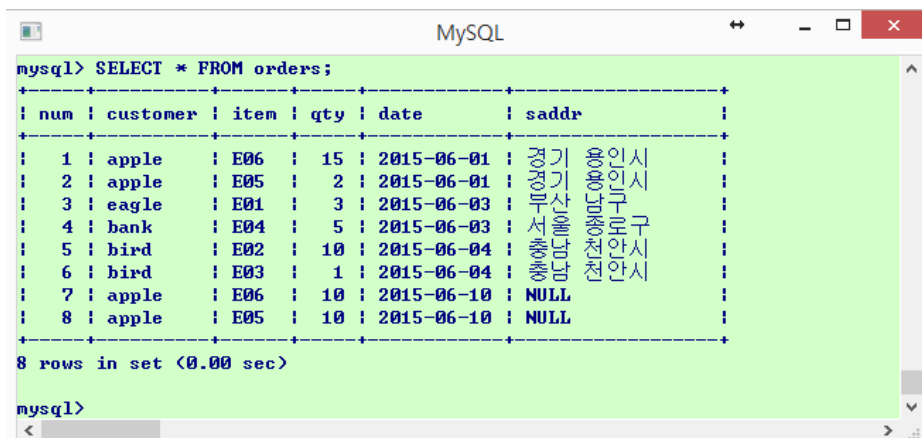


```
mysql> select * from customer;
```

account	name	grade	credit	address
apple	이남이	VIP	5000	경기 용인시
bank	홍길동	Gold	30000	서울 용인구
bird	구선두		0	충남 천안시
eagle	박세재	Gold	2450	부산
king	오나라	Gold	15000	NULL
moon	한가위	New	1000	서울 용인구
pencil	김들	Silver	350	경기 수원시
sun	박태양	New	1000	충인
table	정탁자	New	0	NULL
watch	김정확	New	NULL	NULL

10 rows in set (0.00 sec)

- Orders 테이블



```
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
4	bank	E04	5	2015-06-03	서울 용인구
5	bird	E02	10	2015-06-04	충남 천안시
6	bird	E03	1	2015-06-04	충남 천안시
7	apple	E06	10	2015-06-10	NULL
8	apple	E05	10	2015-06-10	NULL

8 rows in set (0.00 sec)

- 질의 실행 결과

MySQL

```
mysql> SELECT      customer.name, orders.item
-> FROM            orders
-> RIGHT JOIN      customer
-> ON              customer.account=orders.customer;
```

name	item
이남이	E06
이남이	E05
박세재	E01
박세재	E04
김영두	E02
김영두	E03
이남이	E06
이남이	E05
어나라	NULL
한가위	NULL
김태양	NULL
정자화	NULL
김정확	NULL

14 rows in set (0.00 sec)

질의 4 : 상품별 판매 현황을 조사하시오. 판매된 상품의 경우 판매수량을 '판매수량'이라는 이름으로 표시하고, 판매되지 않은 상품은 NULL로 표시하시오. (오른쪽 조인 사용)

- 필요한 테이블 : Items, Orders
- 모든 상품에 대한 조사이므로, 오른쪽 조인 검색의 오른쪽 테이블을 Items으로 함
- 판매 수량의 합계를 구하기 위하여 orders.item을 그룹으로 설정하고 합산
SUM(orders.qty) GROUP BY orders.item
- 조인 검색 조건 : items.code = orders.item

- 질의문

```
SELECT      items.name, SUM( orders.qty )
FROM        orders
RIGHT JOIN  items
ON          items.code=orders.item
GROUP BY   orders.item;
```

- Items 테이블

MySQL

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

code	name	stocks	price
E01	김	30	1575
E02	단무지	45	3413
E03	햄	15	7350
E04	우유	25	4788
E05	주스	20	2415
E06	라면	30	2688
ZZZ	삭제상품	0	0

7 rows in set (0.00 sec)

```
mysql>
```

- Orders 테이블

MySQL

```
mysql> SELECT * FROM orders;
```

num	customer	item	qty	date	saddr
1	apple	E06	15	2015-06-01	경기 고양시
2	apple	E05	2	2015-06-01	경기 고양시
3	eagle	E01	3	2015-06-03	경기 고양시
4	bank	E04	5	2015-06-03	경기 고양시
5	bird	E02	10	2015-06-04	경기 고양시
6	bird	E03	1	2015-06-04	경기 고양시
7	apple	E06	10	2015-06-10	NULL
8	apple	E05	10	2015-06-10	NULL

8 rows in set (0.00 sec)

```
mysql>
```

- 질의 실행 결과

MySQL

```
mysql> SELECT
  -> FROM      items.name, SUM( orders.qty )
  -> FROM      orders
  -> RIGHT JOIN items
  -> ON        items.code=orders.item
  -> GROUP BY  orders.item;
```

name	SUM(orders.qty)
삭제상품	NULL
김	3
단무지	10
햄	1
우유	5
주스	12
라면	25

7 rows in set (0.00 sec)

```
mysql>
```


학습내용2 : 왼쪽 조인과 전체조인

1. 왼쪽 조인 (Left Join)

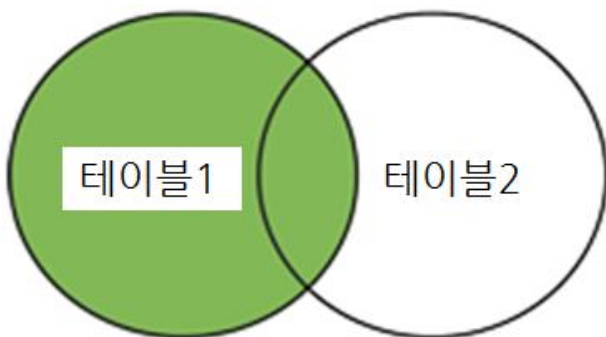
- 오른쪽 테이블(테이블2)과 매칭되는 왼쪽 테이블(테이블1)의 모든 튜플을 반환한다.
- 오른쪽 테이블과 매칭이 되지 않을 경우, NULL을 반환된다.

- 기본 형식

```
SELECT      속성_리스트
FROM        테이블1
LEFT JOIN   테이블2
ON          테이블1.속성 = 테이블2.속성
[ ORDER BY 속성 ];
```

또는

```
SELECT      속성_리스트
FROM        테이블1
LEFT OUTER JOIN 테이블2
ON          테이블1.속성 = 테이블2.속성
[ ORDER BY 속성 ];
```



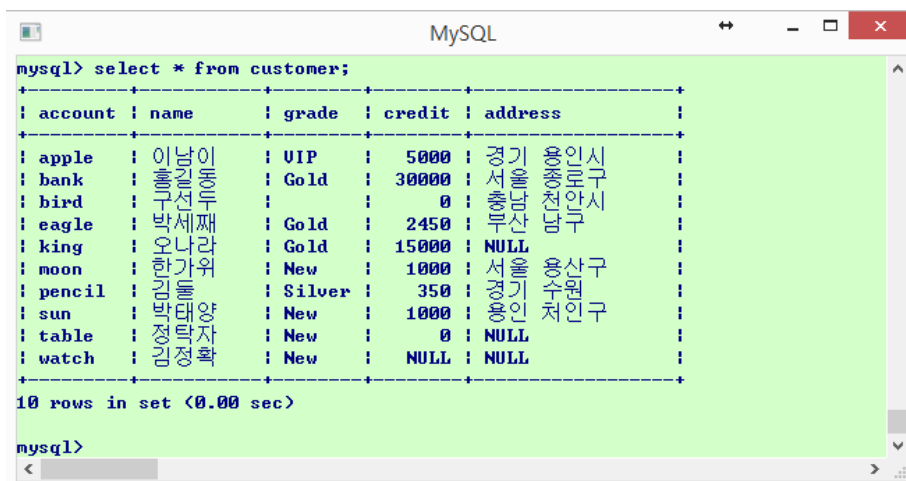
질의 5 : 고객의 구매 현황을 조사하되, 구매하지 않은 고객은 구매 항목에 NULL로 표시하고, 구매한 고객은 구매한 상품코드를 조사하시오. (왼쪽 조인 사용)

- 필요한 테이블 : Customer, Orders
- 고객의 모든 투플이 결과 테이블에 나타나야 하므로, 왼쪽 테이블은 Customer
- 검색조건 : customer.account = orders.customer

- 질의문

```
SELECT    customer.name, orders.item
FROM      customer
LEFT JOIN orders
ON        customer.account=orders.customer;
```

- Customer 테이블



MySQL

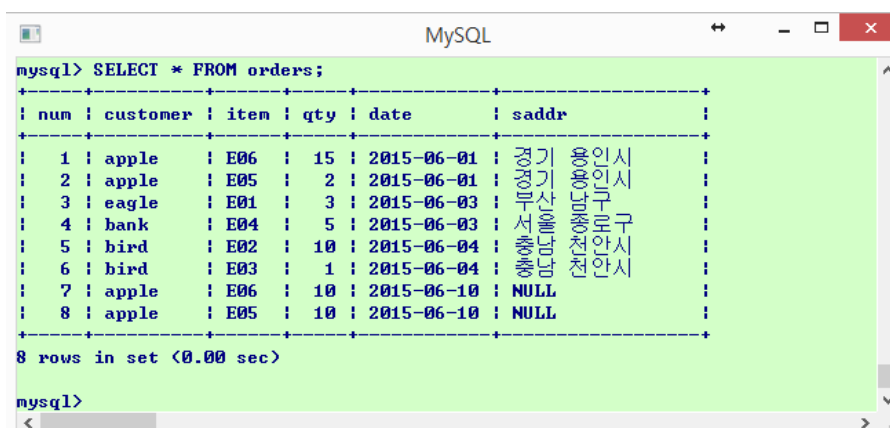
```
mysql> select * from customer;
```

account	name	grade	credit	address
apple	이남이	VIP	5000	경기 용인시
bank	홍길동	Gold	30000	서울 동대문구
bird	구선두		0	충남 천안시
eagle	박세재	Gold	2450	부산 남구
king	오나라	Gold	15000	NULL
moon	한가위	New	1000	서울 용산구
pencil	김돌	Silver	350	경기 수원시
sun	박태양	New	1000	용인 처인구
table	정탁자	New	0	NULL
watch	김정확	New	NULL	NULL

10 rows in set (0.00 sec)

```
mysql>
```

- Orders 테이블



MySQL

```
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 남구
4	bank	E04	5	2015-06-03	서울 동대문구
5	bird	E02	10	2015-06-04	충남 천안시
6	bird	E03	1	2015-06-04	충남 천안시
7	apple	E06	10	2015-06-10	NULL
8	apple	E05	10	2015-06-10	NULL

8 rows in set (0.00 sec)

```
mysql>
```

- 질의 실행 결과

```
mysql> SELECT      customer.name, orders.item
-> FROM      customer
-> LEFT JOIN  orders
-> ON        customer.account=orders.customer;
```

name	item
이남이	E06
이남이	E05
박세재	E01
홍길동	E04
구선두	E02
구선두	E03
이남이	E06
이남이	E05
오나라	NULL
한가위	NULL
김태양	NULL
정택자	NULL
김정확	NULL

14 rows in set (0.00 sec)

질의 6 : 상품별 판매 현황을 조사하시오. 판매된 상품의 경우 판매수량을 표시하고, 판매되지 않은 상품은 NULL로 표시하시오. (왼쪽 조인 사용)

- 필요한 테이블 : Items, Orders
- 모든 상품에 대한 조사이므로, 오른쪽 조인 검색의 왼쪽 테이블은 Items
- 판매 수량의 합계를 구하기 위하여 orders.item을 그룹으로 설정하고 합산
SUM(orders.qty) GROUP BY orders.item
- 조인 검색 조건 : items.code = orders.item

- 질의문

```
SELECT      items.name, SUM( orders.qty )
FROM        items
LEFT JOIN   orders
ON          items.code=orders.item
GROUP BY    orders.item;
```

- Items 테이블

MySQL

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

code	name	stocks	price
E01	김	30	1575
E02	단무지	45	3413
E03	햄	15	7350
E04	우유	25	4788
E05	주스	20	2415
E06	라면	30	2688
ZZZ	삭제상품	0	0

7 rows in set (0.00 sec)

```
mysql>
```

- Orders 테이블

MySQL

```
mysql> SELECT * FROM orders;
```

num	customer	item	qty	date	saddr
1	apple	E06	15	2015-06-01	경기 고양시
2	apple	E05	2	2015-06-01	경기 고양시
3	eagle	E01	3	2015-06-03	경기 고양시
4	bank	E04	5	2015-06-03	경기 고양시
5	bird	E02	10	2015-06-04	경기 고양시
6	bird	E03	1	2015-06-04	경기 고양시
7	apple	E06	10	2015-06-10	NULL
8	apple	E05	10	2015-06-10	NULL

8 rows in set (0.00 sec)

```
mysql>
```

- 질의 실행 결과

MySQL

```
mysql> SELECT
  -> FROM      items
  -> LEFT JOIN orders
  -> ON        items.code=orders.item
  -> GROUP BY  orders.item;
```

name	SUM(orders.qty)
삭제상품	NULL
김	3
단무지	10
햄	1
우유	5
주스	12
라면	25

7 rows in set (0.00 sec)

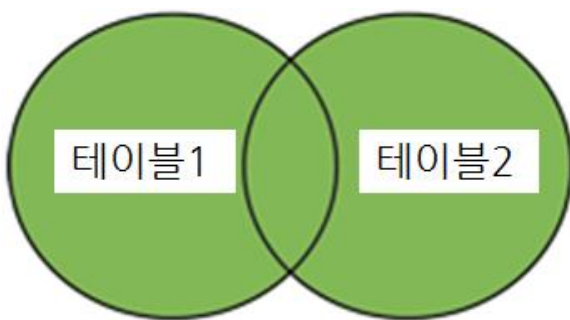
2. 전체 조인 (Full Join)

- 테이블1과 테이블2의 모든 튜플을 반환한다.
- 왼쪽 조인과 오른쪽 조인을 결합한 결과

- 기본 형식

```

SELECT      속성_리스트
FROM        테이블1
FULL OUTER JOIN  테이블2
ON          테이블1.속성 = 테이블2.속성
[ ORDER BY 속성 ];
  
```



- MySQL은 전체조인 지원에 문제가 있으므로, 왼쪽 조인과 오른쪽 조인의 합집으로 해결
- 유니온 (Union)
 - . 두 질의문 결과의 합집합

```

( SELECT 속성리스트 FROM 테이블1 )
UNION
( SELECT 속성리스트 FROM 테이블2 )
  
```

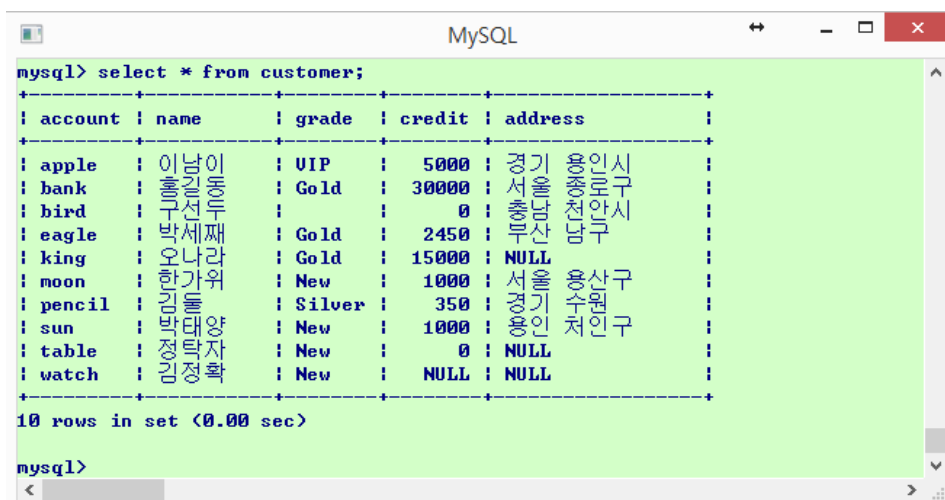
질의 7 : 고객의 구매 현황을 조사하되, 구매하지 않은 고객은 구매 항목에 NULL로 표시하고, 구매한 고객은 구매한 상품코드를 조사하시오. 또한, 비회원(계정이 없는 고객)에 대하여 판매된 상품을 조사하시오 (전체 조인 사용)

- 필요한 테이블 : Customer, Orders
- 검색조건 : customer.account = orders.customer

- 질의문

```
(SELECT      customer.name, orders.item
FROM        customer
LEFT JOIN   orders
ON          customer.account=orders.customer)
UNION
(SELECT      customer.name, orders.item
FROM        customer
RIGHT JOIN  orders
ON          customer.account=orders.customer);
```

- Customer 테이블



MySQL

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

account	name	grade	credit	address
apple	이남이	VIP	5000	경기 용인시
bank	홍길동	Gold	30000	서울 중로구
bird	구선두		0	충남 천안시
eagle	박세재	Gold	2450	부산 남구
king	오나라	Gold	15000	NULL
moon	한가위	New	1000	서울 용산구
pencil	김돌	Silver	350	경기 수원시
sun	박태양	New	1000	용인 처인구
table	정탁자	New	0	NULL
watch	김정확	New	NULL	NULL

10 rows in set (0.00 sec)

```
mysql>
```

- Orders 테이블 (결과를 확인하기 위하여 데이터 변경)

```
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	E02	10	2015-06-04	충청남도
6	bird	E03	1	2015-06-04	충청남도
7		E06	10	2015-06-10	NULL
8	apple		10	2015-06-10	NULL

```
8 rows in set (0.00 sec)

mysql>
```

- 질의 실행 결과

```
mysql> <SELECT customer.name, orders.item
-> FROM customer
-> LEFT JOIN orders
-> ON customer.account=orders.customer>
UNION
<SELECT customer.name, orders.item
-> FROM customer
-> RIGHT JOIN orders
-> ON customer.account=orders.customer>;
```

name	item
이남이	E06
이남이	E05
박세재	E01
홍길동	E04
구선두	E02
구선두	E03
이남이	NULL
오나라	NULL
한가위	NULL
김민철	NULL
박태양	NULL
정택자	NULL
김정호	NULL
NULL	E06

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

- 전체 조인의 모든 속성을 확인하기 위한 질의

```
(SELECT
FROM
LEFT JOIN
ON
UNION
(SELECT
FROM
RIGHT JOIN
ON
customer.account=orders.customer);
```

- 질의 실행 결과

MySQL

```
mysql> (SELECT
-> FROM
-> LEFT JOIN
-> ON
-> UNION
-> (SELECT
-> FROM
-> RIGHT JOIN
-> ON
customer.account=orders.customer);
```

account	name	grade	credit	address	num	customer	item	qty	date	saddr
apple	이대익	VIP	5000	경기 수원시	1	apple	E06	15	2015-06-01	경기 수원시
apple	이대익	VIP	5000	경기 수원시	2	apple	E05	2	2015-06-01	경기 수원시
eagle	이대익	Gold	2450	경기 수원시	3	eagle	E01	3	2015-06-03	경기 수원시
bank	이대익	Gold	30000	경기 수원시	4	bank	E04	5	2015-06-03	경기 수원시
bird	이대익		0	경기 수원시	5	bird	E02	10	2015-06-04	경기 수원시
bird	이대익		0	경기 수원시	6	bird	E03	1	2015-06-04	경기 수원시
apple	이대익	VIP	5000	경기 수원시	8	apple		10	2015-06-10	NULL
king	이대익	Gold	15000	경기 수원시	NULL	NULL	NULL	NULL	NULL	NULL
moon	이대익	New	1000	경기 수원시	NULL	NULL	NULL	NULL	NULL	NULL
pencil	이대익	Silver	350	경기 수원시	NULL	NULL	NULL	NULL	NULL	NULL
sun	이대익	New	1000	경기 수원시	NULL	NULL	NULL	NULL	NULL	NULL
table	이대익	New	0	경기 수원시	NULL	NULL	NULL	NULL	NULL	NULL
watch	이대익	New	NULL	경기 수원시	NULL	NULL	NULL	NULL	NULL	NULL
NULL	이대익	NULL	NULL	경기 수원시	7		E06	10	2015-06-10	NULL

14 rows in set (0.00 sec)

```
mysql>
```

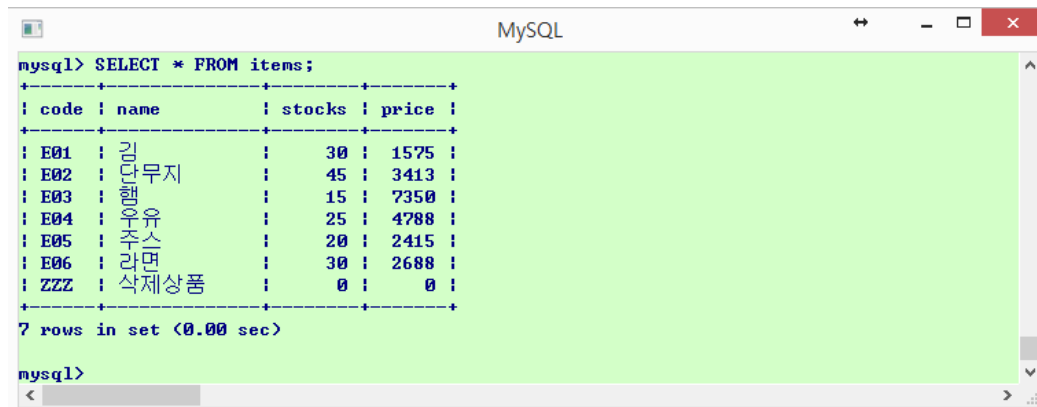

질의 8 : 상품별 판매 현황을 조사하시오. 판매된 상품의 경우 판매수량을 표시하고, 판매되지 않은 상품은 NULL로 표시하시오. 또한, 모든 주문에 대한 현황을 조사 하시오.(전체 조인 사용)

- 왼쪽 테이블 : 상품(items) 테이블
- 오른쪽 테이블 : 주문(Orders) 테이블
- 상품별 판매 현황 : 왼쪽 조인
- 모든 주문에 대한 현황 : 오른쪽 조인

- 질의문

```
( SELECT      items.name, SUM( orders.qty )
  FROM        items
 LEFT JOIN    orders
  ON          items.code=orders.item
 GROUP BY    orders.item )
UNION
( SELECT      items.name, SUM( orders.qty )
  FROM        items
 RIGHT JOIN   orders
  ON          items.code=orders.item
 GROUP BY    orders.item);
```

- Items 테이블



The image shows a MySQL command window with the following content:

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

code	name	stocks	price
E01	김	30	1575
E02	단무지	45	3413
E03	햄	15	7350
E04	우유	25	4788
E05	주스	20	2415
E06	라면	30	2688
ZZZ	삭제상품	0	0

7 rows in set (0.00 sec)

```
mysql>
```

- Orders 테이블

MySQL

```
mysql> SELECT * FROM orders;
```

num	customer	item	qty	date	saddr
1	apple	E06	15	2015-06-01	경기도 성남시
2	apple	E05	2	2015-06-01	경기 양주시
3	eagle	E01	3	2015-06-03	부산광역시
4	bank	E04	5	2015-06-03	서울특별시
5	bird	E02	10	2015-06-04	충청남도 천안시
6	bird	E03	1	2015-06-04	충청남도 천안시
7		E06	10	2015-06-10	NULL
8	apple		10	2015-06-10	NULL

8 rows in set (0.00 sec)

- 상품별 판매 현황 : 왼쪽 조인

MySQL

```
mysql> SELECT items.name, SUM(orders.qty)
-> FROM items
-> LEFT JOIN orders
-> ON items.code=orders.item
-> GROUP BY orders.item;
```

name	SUM(orders.qty)
삭제상품	NULL
김	3
단무지	10
햄	1
우유	5
주스	2
라면	25

7 rows in set (0.00 sec)

mysql>

- 모든 주문에 대한 현황 : 오른쪽 조인

MySQL

```
mysql> SELECT items.name, SUM(orders.qty)
-> FROM items
-> RIGHT JOIN orders
-> ON items.code=orders.item
-> GROUP BY orders.item;
```

name	SUM(orders.qty)
NULL	10
김	3
단무지	10
햄	1
우유	5
주스	2
라면	25

7 rows in set (0.00 sec)

mysql>

- 전체 질의 실행 결과

```

mysql> < SELECT      items.name, SUM( orders.qty )
-> FROM            items
-> LEFT JOIN       orders
-> ON              items.code=orders.item
-> GROUP BY       orders.item >
-> UNION
-> < SELECT      items.name, SUM( orders.qty )
-> FROM            items
-> RIGHT JOIN      orders
-> ON              items.code=orders.item
-> GROUP BY       orders.item>;
+-----+-----+
| name      | SUM( orders.qty ) |
+-----+-----+
| 삭제상품  | NULL              |
| 김        | 3                 |
| 단무지    | 10                |
| 햄        | 1                 |
| 우유      | 5                 |
| 주스      | 2                 |
| 라면      | 25                |
| NULL     | 10                |
+-----+-----+
8 rows in set (0.00 sec)

mysql>

```

【학습정리】

1. 내부 조인은 두 테이블을 조인 검색하고 두 테이블에 모두 속하는 튜플들만 결과로 반환하는 검색이다.
2. 왼쪽 조인과 오른쪽 조인은 두 테이블에 대하여 조인 검색하고 결과를 만들고, 결과로 보고자 하는 테이블을 지정한다.
3. 유니온은 두 질의문의 결과를 결합하여 하나의 결과를 만들 때 사용한다.