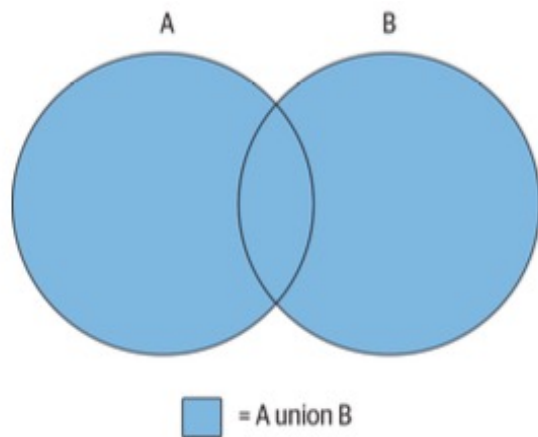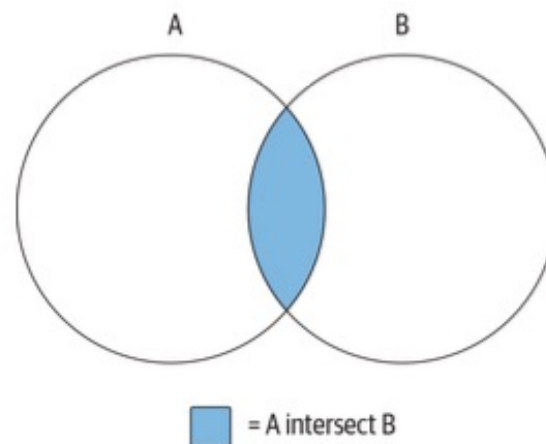# 데이터베이스와 SQL

6장. 집합 연산자

목차

- 6.1 집합 이론

- 6.2 집합 이론 실습

- 6.3 집합 연산자
  - Union 연산자
  - Intersection 연산자
  - Except 연산자

- 6.4 집합 연산 규칙
  - 복합 쿼리의 결과 정렬
  - 집합 연산의 순서

6.1 집합 이론

합집합

= A union B
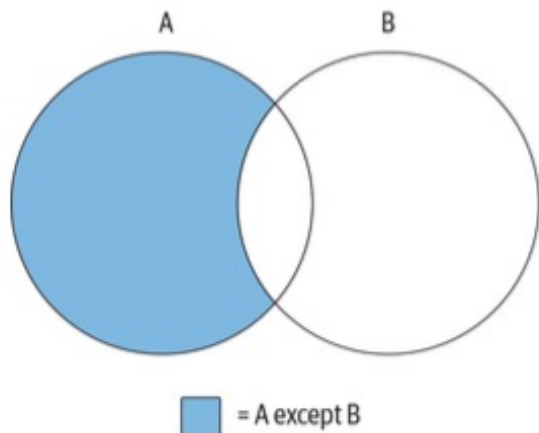
교집합

= A intersect B

차집합

= A except B

?

= ????

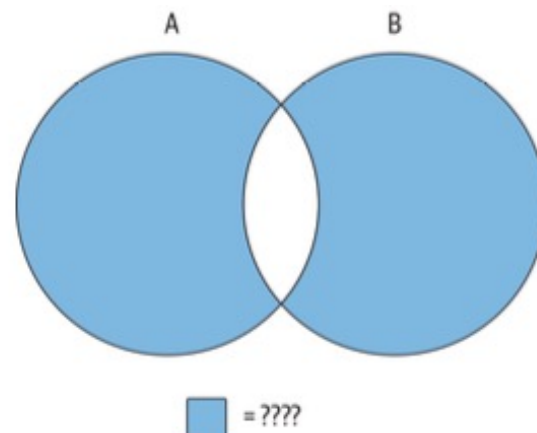(A union B) except (A intersection B)

6.2 집합 이론 실습

- 집합 연산 규칙
  - 두 데이터셋 모두 같은 수의 열(column)을 가져야 됨
  - 두 데이터셋의 각 열의 자료형은 서로 동일해야 됨

```
select 1 as num, 'abc' as str
union
select 9 as num, 'xyz' as str;
```

```
num¦str¦
---+---+
  1¦abc¦
  9¦xyz¦
```

6.3 집합 연산자

- union 연산자
  - union 연산자
    - 결합된 집합을 정렬하고 중복을 제거

  - union all 연산자
    - 최종 데이터셋의 행의 수는 결합되는 집합의 행의 수의 총합과 같음
    - 중복되는 모든 값을 보여줌

6.3 집합 연산자

- 집합 연산을 하기 전에 customer 테이블과 actor 테이블 구성 확인
  - 두 테이블 모두, first_name, last_name이 존재하고 데이터타입도 동일

desc customer**;**

```
Field       ¦Type             ¦Null¦Key¦Default          ¦Extra                                        ¦
------------+-----------------+----+---+-----------------+---------------------------------------------+
customer_id¦smallint unsigned¦NO  ¦PRI¦                 ¦auto_increment                               ¦
store_id   ¦tinyint unsigned ¦NO  ¦MUL¦                 ¦                                             ¦
first_name ¦varchar(45)      ¦NO  ¦   ¦                 ¦                                             ¦
last_name  ¦varchar(45)      ¦NO  ¦MUL¦                 ¦                                             ¦
email      ¦varchar(50)      ¦YES ¦   ¦                 ¦                                             ¦
address_id ¦smallint unsigned¦NO  ¦MUL¦                 ¦                                             ¦
active     ¦tinyint(1)       ¦NO  ¦   ¦1                ¦                                             ¦
create_date¦datetime         ¦NO  ¦   ¦                 ¦                                             ¦
last_update¦timestamp        ¦YES ¦   ¦CURRENT_TIMESTAMP¦DEFAULT_GENERATED on update CURRENT_TIMESTAMP¦
```

desc actor**;**

```
Field       ¦Type             ¦Null¦Key¦Default          ¦Extra                                        ¦
------------+-----------------+----+---+-----------------+---------------------------------------------+
actor_id   ¦smallint unsigned¦NO  ¦PRI¦                 ¦auto_increment                               ¦
first_name ¦varchar(45)      ¦NO  ¦   ¦                 ¦                                             ¦
last_name  ¦varchar(45)      ¦NO  ¦MUL¦                 ¦                                             ¦
last_update¦timestamp        ¦NO  ¦   ¦CURRENT_TIMESTAMP¦DEFAULT_GENERATED on update CURRENT_TIMESTAMP¦
```

6.3 집합 연산자

- customer 테이블과 actor 테이블 union all 연산 수행

```
select 'CUST' as type1, c.first_name, c.last_name
from customer as c
union all
select 'ACTR' as type1, a.first_name, a.last_name
from actor as a;
```

```
type1|first_name|last_name   |
-----+----------+------------+
CUST |MARY      |SMITH       |
CUST |PATRICIA  |JOHNSON     |
CUST |LINDA     |WILLIAMS    |
CUST |BARBARA   |JONES       |
CUST |ELIZABETH |BROWN       |
CUST |JENNIFER  |DAVIS       |
CUST |MARIA     |MILLER      |
         . . .
ACTR |PENELOPE  |GUINESS     |
ACTR |NICK      |WAHLBERG    |
ACTR |ED        |CHASE       |
ACTR |JENNIFER  |DAVIS       |
ACTR |JOHNNY    |LOLLOBRIGIDA|
ACTR |BETTE     |NICHOLSON   |
ACTR |GRACE     |MOSTEL      |
ACTR |MATTHEW   |JOHANSSON   |
ACTR |JOE       |SWANK       |
```

```
select count(first_name) from customer;
```

```
count(first_name)|
-----------------+
              599|
```

```
select count(first_name) from actor;
```

```
count(first_name)|
-----------------+
              200|
```

총 799개의 이름 반환, 599개는 customer 테이블에서, 200개는 actor 테이블에서 가져옴

6.3 집합 연산자

- actor 테이블에 `union_all` 연산 수행
  - 중복 항목 제거 안함
  - 총 데이터수가 400개로 늘어남

```sql
select 'ACTR' as typ, a.first_name, a.last_name
from actor as a
union all
select 'ACTR' as typ, a.first_name, a.last_name
from actor as a;
```

- customer 테이블과 actor 테이블에서
  - 이름이 'J'로 시작하고 성은 'D'로 시작하는 사람들의 합집합: `union all` (중복)

```sql
select c.first_name, c.last_name
from customer as c
where c.first_name like 'J%' and c.last_name like 'D%'
union all
select a.first_name, a.last_name
from actor as a
where a.first_name like 'J%' and a.last_name like 'D%';
```

```
first_name¦last_name¦
----------+---------+
JENNIFER  ¦DAVIS    ¦
JENNIFER  ¦DAVIS    ¦
JUDY      ¦DEAN     ¦
JODIE     ¦DEGENERES¦
JULIANNE  ¦DENCH    ¦
```

6.3 집합 연산자

- union
  - 중복 데이터 제거

```sql
select c.first_name, c.last_name
from customer as c
where c.first_name like 'J%' and c.last_name like 'D%'
union
select a.first_name, a.last_name
from actor as a
where a.first_name like 'J%' and a.last_name like 'D%';
```

```
first_name¦last_name¦
----------+---------+
JENNIFER  ¦DAVIS    ¦
JUDY      ¦DEAN     ¦
JODIE     ¦DEGENERES¦
JULIANNE  ¦DENCH    ¦
```

6.3 집합 연산자

- intersect 연산자
  - MySQL 8.0 버전에서 지원 안함
  - inner join으로 동일한 결과를 얻을 수 있음

```sql
select c.first_name, c.last_name
from customer as c
    inner join actor as a
    on (c.first_name = a.first_name)
    and (c.last_name = a.last_name);
```

```
first_name¦last_name¦
----------+---------+
JENNIFER  ¦DAVIS    ¦
```

```sql
select c.first_name, c.last_name
from customer as c
    inner join actor as a
    on (c.first_name = a.first_name)
    and (c.last_name = a.last_name)
where a.first_name like 'J%' and a.last_name like 'D%';
```

```
first_name¦last_name¦
----------+---------+
JENNIFER  ¦DAVIS    ¦
```

6.4 집합 연산 규칙

■ 복합 쿼리의 결과 정렬
  ▪ order by 절을 쿼리 마지막에 추가
    • 열 이름 정의는 복합 쿼리의 첫 번째 쿼리에 있는 열의 이름을 사용해야 됨

```sql
select a.first_name as fname, a.last_name as lname
from actor as a
where a.first_name like 'J%' and a.last_name like 'D%'
union all
select c.first_name, c.last_name
from customer as c
where c.first_name like 'J%' and c.last_name like 'D%'
order by lname, fname;
```

```
fname    ¦lname     ¦
--------+---------+
JENNIFER¦DAVIS     ¦
JENNIFER¦DAVIS     ¦
JUDY    ¦DEAN      ¦
JODIE   ¦DEGENERES ¦
JULIANNE¦DENCH     ¦
```

```sql
select a.first_name as fname, a.last_name as lname
from actor as a
where a.first_name like 'J%' and a.last_name like 'D%'
union all
select c.first_name, c.last_name
from customer as c
where c.first_name like 'J%' and c.last_name like 'D%'
order by last_name, first_name;
```

SQL Error [1054] [42S22]: Unknown column 'last_name' in 'order clause'

6.4 집합 연산 규칙

- 집합 연산의 순서
  - 복합 쿼리는 위에서 아래의 순서대로 실행
  - 예외:
    - intersect 연산자가 다른 집합 연산자보다 우선 순위가 높음

```sql
select a.first_name, a.last_name
from actor as a
where a.first_name like 'J%' and a.last_name like 'D%'
union all
select a.first_name, a.last_name
from actor as a
where a.first_name like 'M%' and a.last_name like 'T%'
union
select c.first_name, c.last_name
from customer as c
where c.first_name like 'J%' and c.last_name like 'D%';
```

	first_name	last_name
1	JENNIFER	DAVIS
2	JUDY	DEAN
3	JODIE	DEGENERES
4	JULIANNE	DENCH
5	MARY	TANDY
6	MENA	TEMPLE

6.5 학습 점검

- 실습 6-2
  - 성이 L로 시작하는 모든 배우와 고객의 이름과 성을 찾는 복합 쿼리 작성

```
select first_name, last_name
from actor
where last_name like 'L%'
union
select first_name, last_name
from customer
where last_name like 'L%';
```

```
first_name¦last_name   ¦
----------+------------+
MATTHEW   ¦LEIGH       ¦
JOHNNY    ¦LOLLOBRIGIDA¦
MISTY     ¦LAMBERT     ¦
JACOB     ¦LANCE       ¦
RENEE     ¦LANE        ¦
HEIDI     ¦LARSON      ¦
DARYL     ¦LARUE       ¦
LAURIE    ¦LAWRENCE    ¦
JEANNE    ¦LAWSON      ¦
LAWRENCE  ¦LAWTON      ¦
KIMBERLY  ¦LEE         ¦
LOUIS     ¦LEONE       ¦
SARAH     ¦LEWIS       ¦
GEORGE    ¦LINTON      ¦
MAUREEN   ¦LITTLE      ¦
DWIGHT    ¦LOMBARDI    ¦
JACQUELINE¦LONG        ¦
AMY       ¦LOPEZ       ¦
BARRY     ¦LOVELACE    ¦
PRISCILLA ¦LOWE        ¦
VELMA     ¦LUCAS       ¦
WILLARD   ¦LUMPKIN     ¦
LEWIS     ¦LYMAN       ¦
JACKIE    ¦LYNCH       ¦
```

6.5 학습 점검

- 실습 6-3
  - last_name 열을 기준으로 실습 6-2의 결과를 오름 차순 정렬하시오.

```sql
select first_name, last_name
from actor
where last_name like 'L%'
union
select first_name, last_name
from customer
where last_name like 'L%'
order by last_name;
```

```
first_name¦last_name   ¦
----------+------------+
MISTY     ¦LAMBERT     ¦
JACOB     ¦LANCE       ¦
RENEE     ¦LANE        ¦
HEIDI     ¦LARSON      ¦
DARYL     ¦LARUE       ¦
LAURIE    ¦LAWRENCE    ¦
JEANNE    ¦LAWSON      ¦
LAWRENCE  ¦LAWTON      ¦
KIMBERLY  ¦LEE         ¦
MATTHEW   ¦LEIGH       ¦
LOUIS     ¦LEONE       ¦
SARAH     ¦LEWIS       ¦
GEORGE    ¦LINTON      ¦
MAUREEN   ¦LITTLE      ¦
JOHNNY    ¦LOLLOBRIGIDA¦
DWIGHT    ¦LOMBARDI    ¦
JACQUELINE¦LONG        ¦
AMY       ¦LOPEZ       ¦
BARRY     ¦LOVELACE    ¦
PRISCILLA ¦LOWE        ¦
VELMA     ¦LUCAS       ¦
WILLARD   ¦LUMPKIN     ¦
LEWIS     ¦LYMAN       ¦
JACKIE    ¦LYNCH       ¦
```

Questions?