


[DB] DB구조&설계 - JOIN

👤 생성자	 정희주
🕒 생성 일시	@2023년 12월 13일 오전 2:53
🏷 태그	

조인 종류

INNER JOIN (JOIN만 입력해도 INNER JOIN)

OUTER JOIN

LEFT OUTER JOIN (LEFT JOIN)

RIGHT OUTER JOIN (RIGHT JOIN)

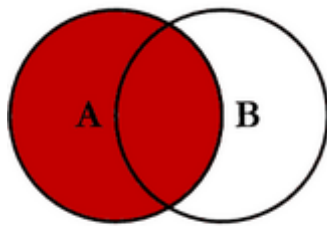
CROSS JOIN



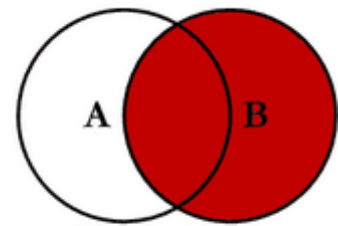
조인은 여러 테이블을 결합하는 것으로, 데이터 조회 시 다른 테이블의 데이터를 함께 조회해야 할 때 이용한다.

조인 종류

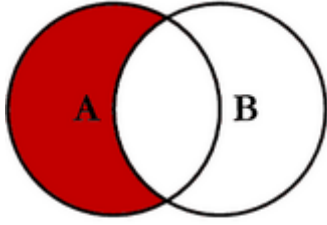
SQL JOINS



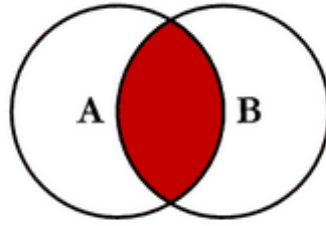
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
```



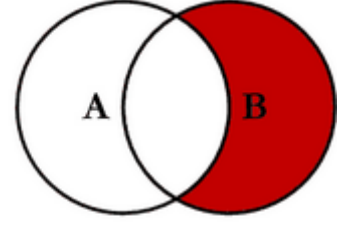
```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
```



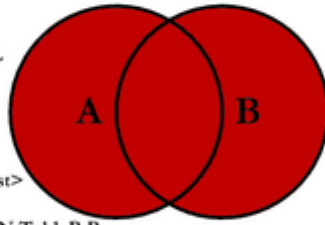
```
SELECT <select_list>
FROM TableA A
LEFT JOIN TableB B
ON A.Key = B.Key
WHERE B.Key IS NULL
```



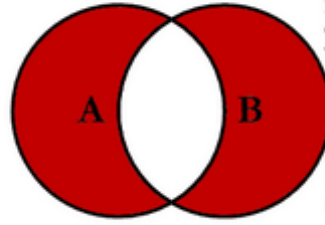
```
SELECT <select_list>
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
RIGHT JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```



```
SELECT <select_list>
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL
OR B.Key IS NULL
```

© C.L. Moffatt, 2008

- Inner Join
- Outer Join(Left Outer Join, Right Outer Join, Full Outer Join)
- Cross Join
- Self Join

다음과 같은 테이블 두 개를 가지고 각각의 조인에 대해 알아보자.

[Star 테이블]

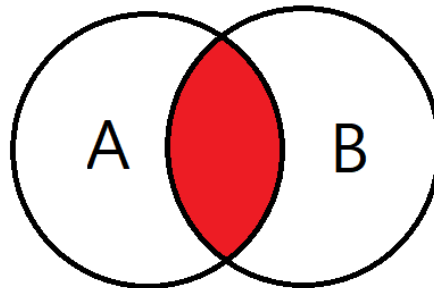
ID	Name	DepNo
1	강호동	10
2	이수근	10
3	유재석	20
4	박명수	20
5	안재현	30

[Dep 테이블]

DepNo	DepName
10	1박2일
20	무한도전
30	신서유기
40	이경규가간다

6	송민호	30
7	이병헌	NULL

INNER JOIN (JOIN만 입력해도 INNER JOIN)



교집합으로, 기존 테이블과 join 테이블의 중복된 값을 보여준다.

```
SELECT Star.Name, Dep.DepName
From Star JOIN Dep ON Star.DepNo = Dep.DepNo
```

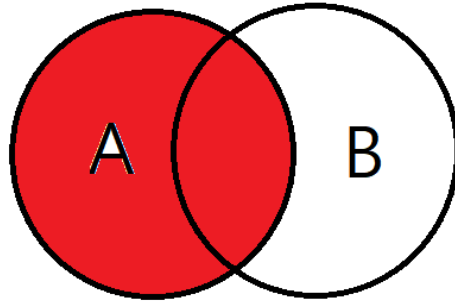
Name	DepName
강호동	1박2일
이수근	1박2일
유재석	무한도전
박명수	무한도전
안재현	신서유기
송민호	신서유기

Star테이블과 Dep테이블 중에서 **Star의 DepNo와 Dep의 DepNo가 일치하는 요소들만** 골라서 출력을 하게된다.

OUTER JOIN

- INNER JOIN은 공통된 부분이 있는 행만 출력을 해주었는데, **공통된 부분이 없는 데이터도** 함께 보고싶은 경우가 있을 것이다. 이럴 때 사용하는 것이 바로 **OUTER JOIN**

LEFT OUTER JOIN (LEFT JOIN)



기준테이블값과 조인테이블과 중복된 값을 보여준다.

왼쪽테이블 기준으로 JOIN을 한다고 생각하면 편하다.

- A테이블에만 있고 B테이블에 없는 값은 NULL로 표시된다.

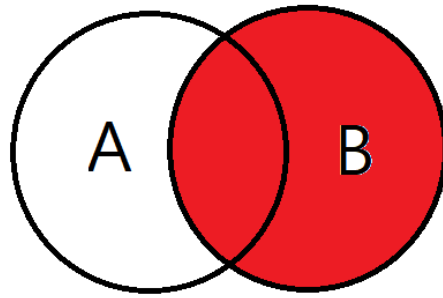
```
SELECT Star.Name, Dep.Name
FROM Star LEFT JOIN Dep
ON Star.DepNo = Dep.DepNo
```

위의 쿼리를 실행한 결과는 다음과 같다.

Name	DepName
강호동	1박2일
이수근	1박2일
유재석	무한도전
박명수	무한도전
안재현	신서유기
송민호	신서유기
이병헌	NULL

공통된 값 + 왼쪽 테이블에만 있는 값이 출력된 것을 알 수 있다. A테이블에만 있고 B테이블에 없는 값은 NULL로 표시된다.

RIGHT OUTER JOIN (RIGHT JOIN)



LEFT OUTER JOIN과는 반대로 오른쪽 테이블 기준으로 JOIN하는 것이다.

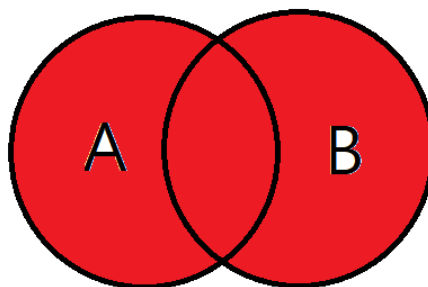
- 값이 없는 데이터는 NULL로 표시가 된다

```
SELECT Star.Name, Dep.Name
FROM Star RIGHT JOIN Dep
ON Star.DepNo = Dep.DepNo
```

위 쿼리를 실행한 결과는 다음과 같다.

Name	DepName
강호동	1박2일
이수근	1박2일
유재석	무한도전
박명수	무한도전
안재현	신서유기
송민호	신서유기
NULL	이경규가간다

- FULL OUTER JOIN



합집합을 말한다. A와 B 테이블의 모든 데이터가 검색된다.

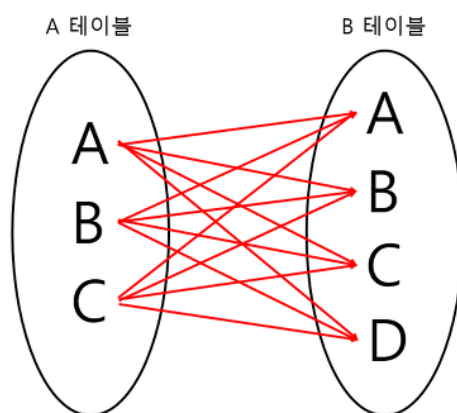
```
SELECT Star.Name, Dep.Name
FROM Star FULL OUTER JOIN Dep
ON Star.DepNo = Dep.DepNo
```

위 쿼리를 실행한 결과는 다음과 같다.

Name	DepName
강호동	1박2일
이수근	1박2일
유재석	무한도전
박명수	무한도전
안재현	신서유기
송민호	신서유기
이병헌	NULL
NULL	이경규가간다

즉, LEFT OUTER JOIN과 RIGHT OUTER JOIN의 결과값을 합친 것이라고 볼 수 있다.

CROSS JOIN



두 테이블 간의 가능한 모든 경우의 수를 전부 표현해주는 방식이다. 즉, 카디널리티 곱을 한 것

```
SELECT Star.Name, Dep.DepName
FROM Star CROSS JOIN Dep
```

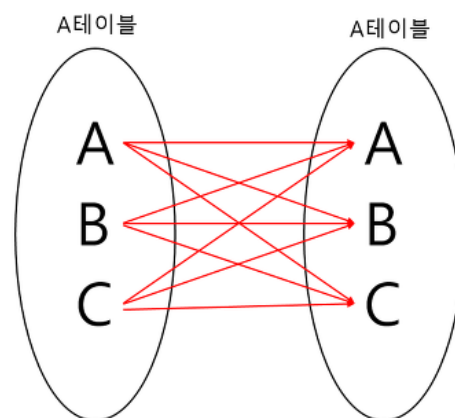
위 쿼리를 실행한 결과는 다음과 같다.

Name	DepName
강호동	1박2일
이수근	1박2일
유재석	1박2일
박명수	1박2일
안재현	1박2일
송민호	1박2일
이병헌	1박2일
강호동	무한도전
이수근	무한도전
...	...
송민호	이경규가간다
이병헌	이경규가간다

위의 결과 처럼, 두 테이블의 **모든 행들을 서로 교차하여 곱한다고** 생각하면 된다.

그래서 Star테이블(7행) x Dep테이블(4행) = 28 행이 탄생하게 된다.

- SELF JOIN



자기 자신과 자기 자신을 조인하는 것이다.

하나의 테이블을 여러번 복사해서 조인한다고 생각하면 편하다.

자신이 갖고 있는 칼럼을 다양하게 변형시켜 활용할 때 자주 사용한다.

```
SELECT A.ID, A.Name, A.Partner, B.Partner PartName
FROM Star A JOIN Star B
ON A.Partner = B.ID
```

SELF JOIN이 필요한 상황 예시

[A 테이블]

ID	Name	Partner
1	강호동	3
2	유재석	4
3	나영석	5
4	김태호	6
5	이명한	1
6	박명수	2

위의 A 테이블을 보면 각각의 스타의 ID와 이름, Partner의 번호가 부여되어 있다.

Partner의 번호 대신 **이름을 알고 싶다면?** → SELF JOIN

위 쿼리를 실행한 결과는 다음과 같다.

ID	Name	Partner	PartName
1	강호동	3	나영석
2	유재석	4	김태호
3	나영석	5	이명한
4	김태호	6	박명수
5	이명한	1	강호동
6	박명수	2	유재석

이렇게 하나의 테이블을 가지고 SELF JOIN을 이용하여 원하는 값을 출력할 수 있다.