

Chapter. 02

문법설명

| SQL 문법 설명

FAST CAMPUS
ONLINE
Part4. SQL I

강사. 민경우

I 목차

STEP 1 강의 소개

STEP 2 문법 설명

STEP 3 정리

I 소개

강의 소개

- SQL 문제 풀이 실습을 위해서 기본적으로 알고 있어야 하는 문법들 설명
 - 순위집계
 - 조인
- 기본적인 문법들의 사용방법을 익힘으로써 좀더 쉽게 문제를 풀 수 있도록 구성

I 순위집계

1. RANK

SELECT **RANK()** **OVER**(**PARTITION BY** [그룹할 컬럼들] **ORDER BY** [순위를 매길때 사용할 컬럼들])
FROM [테이블명1]

예) 1등(100점), 2등(90점), **3등(85점)**,**3등(85점)**,**3등(85점)**, **6등(80점)**, 7등(79점)...100등(55점)

2. DENSE_RANK

SELECT **DENSE_RANK()** **OVER**(**PARTITION BY** [그룹할 컬럼들] **ORDER BY** [순위를 매길때 사용할 컬럼들])
FROM [테이블명1]

예) 1등(100점), 2등(90점), **3등(85점)**,**3등(85점)**,**3등(85점)**, **4등(80점)**, 5등(79점)...100등(55점)

3. ROW_NUMBER

SELECT **ROW_NUMBER()** **OVER**(**PARTITION BY** [그룹할 컬럼들] **ORDER BY** [순위를 매길때 사용할 컬럼들])
FROM [테이블명1]

예) 1등(100점), 2등(90점), **3등(85점)**,**4등(85점)**,**5등(85점)**, **6등(80점)**, 7등(79점)...100등(55점)

I 순위집계

tScore

부서명 (Department)	이름 (Name)	점수 (Score)
개발부	홍길동	100
개발부	김갑수	90
개발부	김이수	85
개발부	김사원	85
개발부	이길동	85
개발부	서준수	80
개발부	이기영	79
개발부	이해오	55

쿼리

```
SELECT NAME
      , SCORE
      , RANK() OVER(PARTITION BY department ORDER BY socre DESC)
FROM tScore
```

결과 값

	name character varying(8)	score integer	rank bigint
1	홍길동	100	1
2	김갑수	90	2
3	김이수	85	3
4	김사원	85	3
5	이길동	85	3
6	서준수	80	6
7	이기영	79	7
8	이해오	55	8

I 순위집계

tScore

부서명 (Department)	이름 (Name)	점수 (Score)
개발부	홍길동	100
개발부	김갑수	90
개발부	김이수	85
개발부	김사원	85
개발부	이길동	85
개발부	서준수	80
개발부	이기영	79
개발부	이해오	55

쿼리

```
SELECT NAME
      , SCORE
      , DENSE_RANK() OVER(PARTITION BY department ORDER BY socre DESC)
FROM tScore
```

결과 값

	name character varying(8)	score integer	dense_rank bigint
1	홍길동	100	1
2	김갑수	90	2
3	김이수	85	3
4	김사원	85	3
5	이길동	85	3
6	서준수	80	4
7	이기영	79	5
8	이해오	55	6

I 순위집계

tScore

부서명 (Department)	이름 (Name)	점수 (Score)
개발부	홍길동	100
개발부	김갑수	90
개발부	김이수	85
개발부	김사원	85
개발부	이길동	85
개발부	서준수	80
개발부	이기영	79
개발부	이해오	55

쿼리

```
SELECT NAME
      , SCORE
      , ROW_NUMBER() OVER(PARTITION BY department ORDER BY socre DESC)
FROM tScore
```

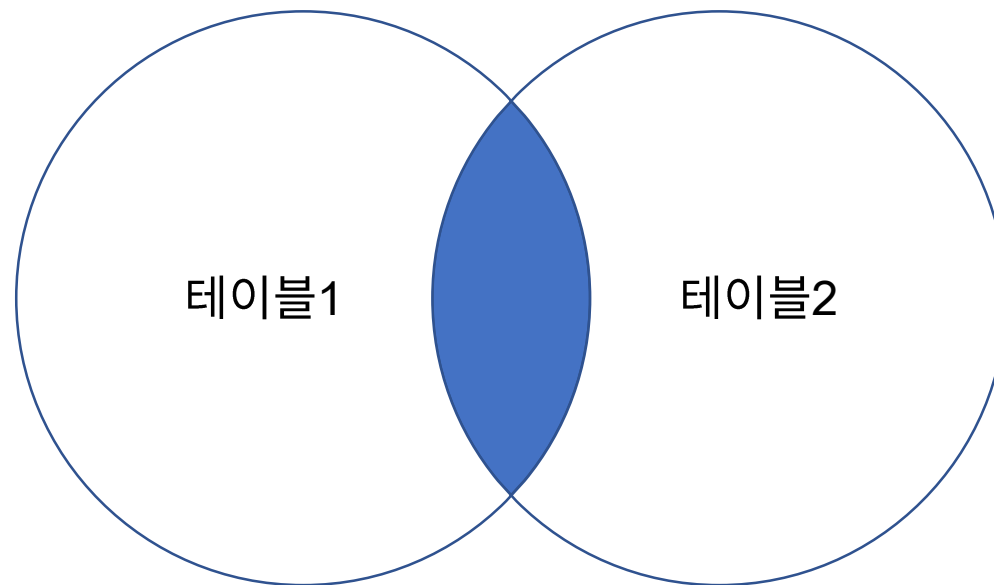
결과 값

	name character varying(8)	score integer	row_number bigint
1	홍길동	100	1
2	김갑수	90	2
3	김이수	85	3
4	김사원	85	4
5	이길동	85	5
6	서준수	80	6
7	이기영	79	7
8	이해오	55	8

I 조인

1. INNER JOIN

```
SELECT *  
FROM [테이블1] AS A  
INNER JOIN [테이블2] AS B  
ON A.KEY = B.KEY
```



I 조인

tDepartment

부서번호 (DNumber)	부서명 (DName)
D1001	문구생산부
D2001	가구생산부
D3001	악세사리생산부
D4001	전자기기생산부
D5001	음료생산부

tEmployee (테이블 내 일부 데이터)

직원번호 (ENumber)	부서번호 (DNumber)	직급번호 (RNumber)	직원명 (ENAME)	주민번호 (ERRN)	주소 (EAddr)	입사일 (StartDate)	퇴사일 (ResignationDate)
E0001		R0001	김이수	850726-1118323	서울시 강남구	2018-01-02 09:00:00+09	
E0002		R0001	김사원	900513-1136345	부산시 북구	2018-01-02 09:00:00+09	
E0003		R0002	이길동	890206-1112632	대전시 서구	2018-01-02 09:00:00+09	
E0004		R0002	서준수	880816-1116234	서울시 동구	2018-01-02 09:00:00+09	
E0005		R0003	이기영	920322-2114233	울산시 동구	2018-01-02 09:00:00+09	
E1001	D1001	R0001	박하나	760312-2127522	서울시 강남구	2016-01-02 09:00:00+09	
E1002	D1001	R0002	김문구	830622-1152643	부산시 남구	2014-03-05 09:00:00+09	
E1003	D1001	R0003	이과장	790312-1117865	부산시 중구	2010-02-03 09:00:00+09	
E1004	D1001	R0003	이해오	800312-1112346	서울시 도봉구	2010-05-06 09:00:00+09	
E1005	D1001	R0006	박총광	670107-1116244	서울시 광진구	2002-02-06 09:00:00+09	
E1006	D1001	R0001	김아해	870721-1113245	부산시 북구	2016-05-02 09:00:00+09	2021-02-25 23:59:59+09

쿼리

```
SELECT tde.DNAME, tem.ENAME
FROM tDepartment as tde
INNER JOIN tEmployee as tem
ON tde.DNUMBER = tem.DNUMBER
ORDER BY tem.ENAME
```

결과 값

	dname character varying(10)	ename character varying(5)
1	문구생산부	권종인
2	가구생산부	김구성
3	음료생산부	김대한
4	문구생산부	김문구
5	음료생산부	김실약

I 조인

2. OUTER JOIN

-- LEFT OUTER JOIN

SELECT *

FROM [테이블1] AS A

LEFT OUTER JOIN [테이블2] AS B

ON A.KEY = B.KEY

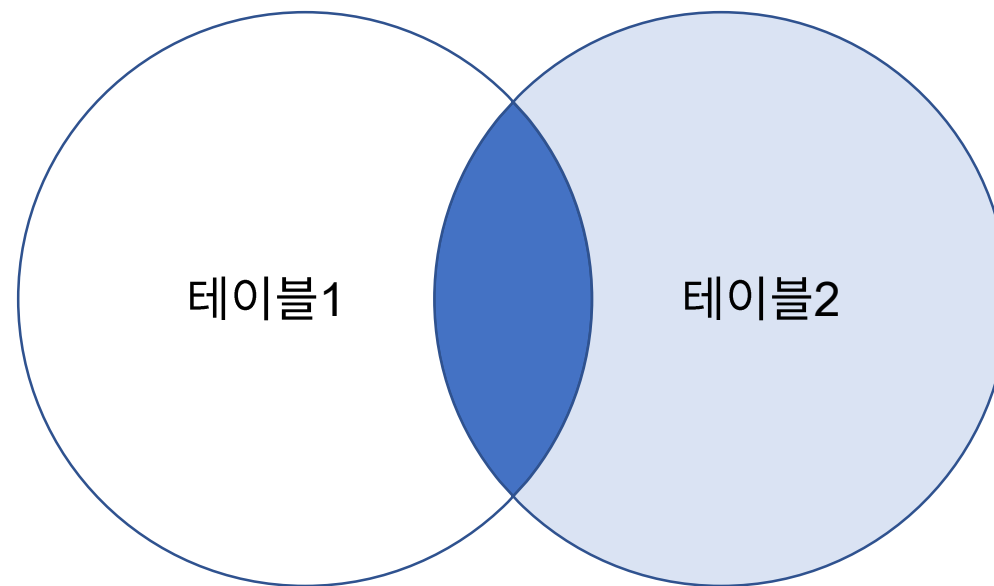
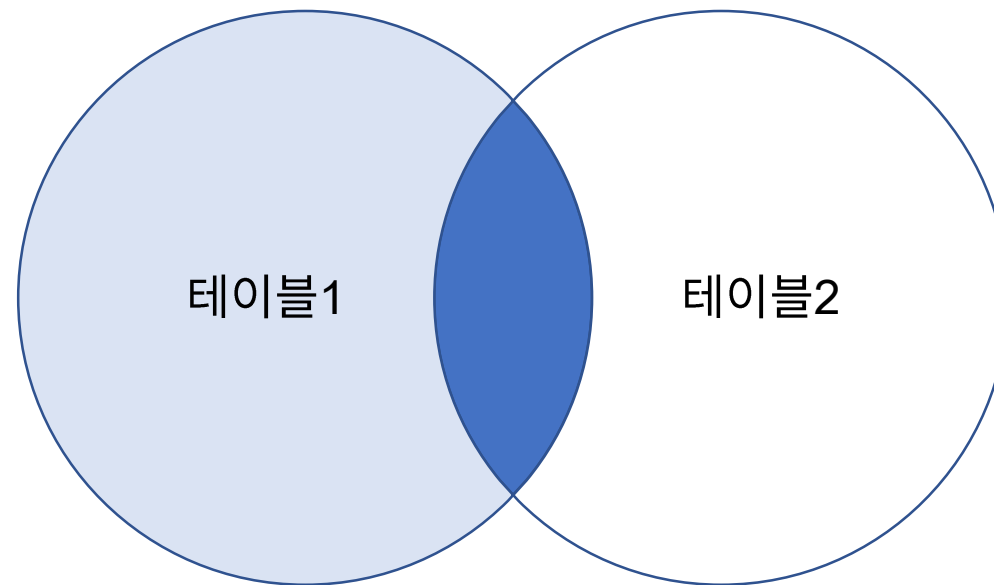
-- RIGHT OUTER JOIN

SELECT *

FROM [테이블1] AS A

RIGHT OUTER JOIN [테이블2] AS B

ON A.KEY = B.KEY



I 조인

tUNI1

아이디 (id)	이름 (name)
1	홍길동
2	김갑수
3	박총괄

tUNI2

아이디 (id)	이름 (name)
1	홍길동
2	김갑수
4	이빛나

쿼리

-- LEFT OUTER JOIN

```
SELECT *
FROM tUNI1 as tu1
LEFT OUTER JOIN tUNI2 as tu2
ON tu1.id = tu2.id
```

결과 값

	id integer	name character varying(8)	id integer	name character varying(8)
1	1	홍길동	1	홍길동
2	2	김갑수	2	김갑수
3	3	박총괄		

-- RIGHT OUTER JOIN

```
SELECT *
FROM tUNI1 as tu1
RIGHT OUTER JOIN tUNI2 as tu2
ON tu1.id = tu2.id
```

	id integer	name character varying(8)	id integer	name character varying(8)
1	1	홍길동	1	홍길동
2	2	김갑수	2	김갑수
3			4	이빛나

I 조인

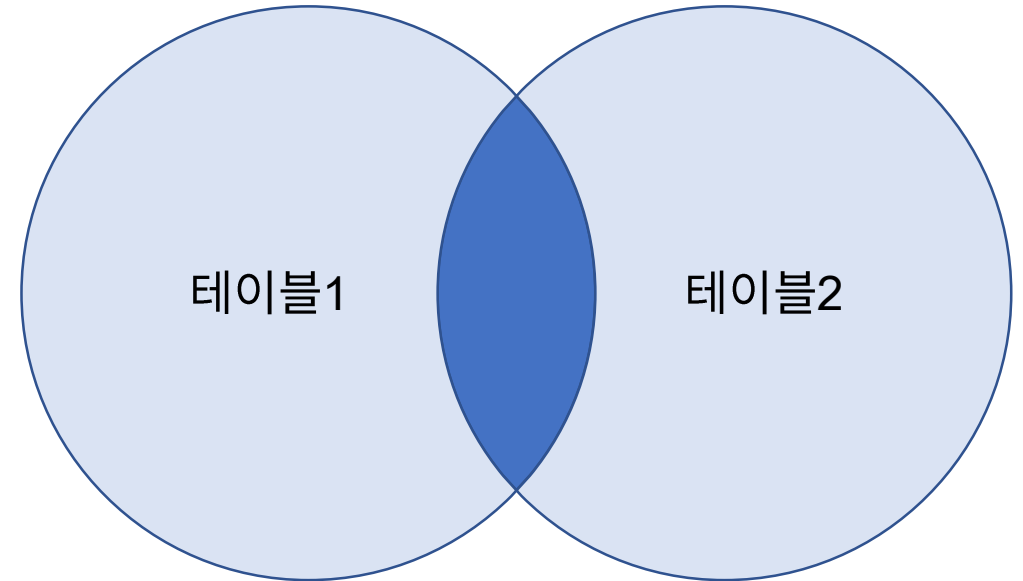
3. FULL OUTER JOIN

SELECT *

FROM [테이블1] AS A

FULL OUTER JOIN [테이블2] AS B

ON A.KEY = B.KEY



I 조인

tUNI1

아이디 (id)	이름 (name)
1	홍길동
2	김갑수
3	박총괄

tUNI2

아이디 (id)	이름 (name)
1	홍길동
2	김갑수
4	이빛나

쿼리

```
SELECT *
FROM tUNI1 as tu1
FULL OUTER JOIN tUNI2 as tu2
ON tu1.id = tu2.id
```

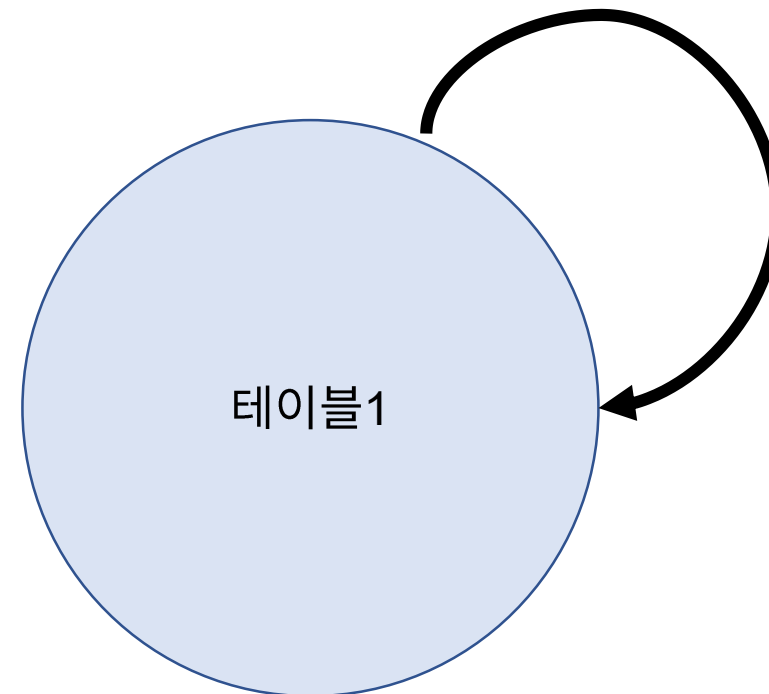
결과 값

	id integer	name character varying(8)	id integer	name character varying(8)
1	1	홍길동	1	홍길동
2	2	김갑수	2	김갑수
3	3	박총괄		
4			4	이빛나

| 조인

4. SELF JOIN

```
SELECT *  
FROM [테이블1] AS A  
JOIN [테이블1] AS B  
ON A.NAME = B.MANAGER
```



I 조인

tSample

아이디 (ID)	직원이름 (NAME)	관리자이름 (MANAGER)
1	홍길동	김갑수
2	김갑수	서이원
3	서이원	
4	김원태	예원희
5	예원희	

쿼리

```
SELECT tsa2.NAME as “부하직원”
      , tsa1.NAME as “관리자”
FROM tSample as tsa1
JOIN tSample as tsa2
ON tsa1.NAME = tsa2.MANAGER
WHERE tsa1.NAME = ‘김갑수’
```

결과 값

	부하직원 character varying(8)	관리자 character varying(8)
1	홍길동	김갑수

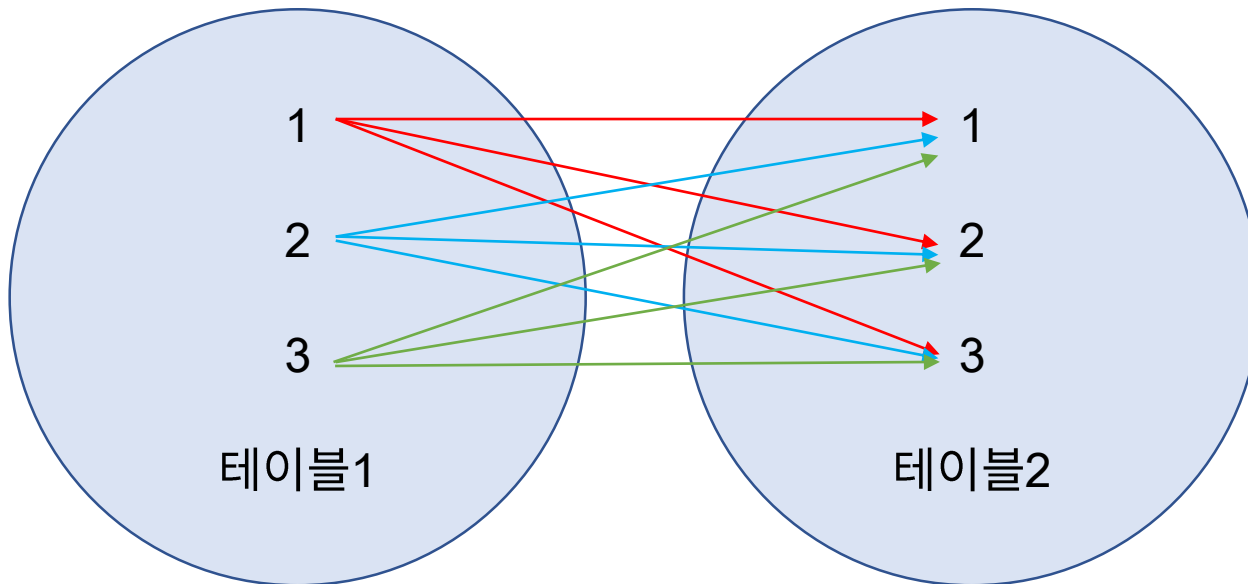
I 조인

5. CROSS JOIN

SELECT *

FROM [테이블1] AS A

CORSS JOIN [테이블2] AS B



I 조인

tUNI1

아이디 (id)	이름 (name)
1	홍길동
2	김갑수
3	박총괄

tUNI2

아이디 (id)	이름 (name)
1	홍길동
2	김갑수
4	이빛나

쿼리

```
SELECT *
FROM tUNI1 as tu1
CROSS JOIN tUNI2 as tu2
```

결과 값

	id integer	name character varying(8)	id integer	name character varying(8)
1	1	홍길동	1	홍길동
2	1	홍길동	2	김갑수
3	1	홍길동	4	이빛나
4	2	김갑수	1	홍길동
5	2	김갑수	2	김갑수
6	2	김갑수	4	이빛나
7	3	박총괄	1	홍길동
8	3	박총괄	2	김갑수
9	3	박총괄	4	이빛나

I 요약

정리

- SQL 문제를 해결하는데 있어서 필요한 기본 문법을 SELECT부터 JOIN 까지를 통하여 익힘
- 기본 문법에 대한 설명과 실제 예시를 통해서 SQL Query에 대한 감각을 익힘