

# Database

(데이터베이스)

# FROM

---

- 1 FROM의 개념
- 2 JOIN의 의미와 원리
- 3 INNER JOIN / OUTER JOIN
- 4 ANSI 문법



# FROM의 개념

## SQL 문법 실행 순서

5	SELECT	출력하고 싶은 컬럼만 작성하기
1	FROM	데이터를 가져올 테이블 입력
2	WHERE	원하는 튜플만 가져오도록 필터링(조건문)
3	GROUP BY	특정 컬럼을 기준으로 그룹화
4	HAVING	그룹화 상태의 데이터를 필터링
6	ORDER BY	특정 컬럼으로 정렬하기

# FROM의 개념

---

## FROM이란?

~로 부터 라는 뜻으로 우리가 찾고자 하는 데이터가 존재하는  
객체(테이블) 앞에 사용

예제에서 보통 가장 앞에 등장하므로 어렵지 않게 작성 가능

예) 직원테이블에서 직원 이름이 '김민수'인 직원을 찾아주세요!

# FROM의 개념

---

**FROM 뒤에 여러 개의 테이블 사용 가능**

**BUT!!**

**여러 개의 테이블이 올 경우 보통 테이블에 ‘별칭’을 사용**

예) 직원 테이블에서는 직원ID, 직원이 속한 부서명을 부서 테이블에서 꺼내서 출력해주세요

SELECT A.직원ID, B.부서명

FROM 직원 A, 부서 B



**출력 결과는.....?**

Step 2.

★ JOIN ★



# JOIN

**JOIN은 2개 이상의 테이블을 하나의 SQL 문으로 연결한 것**

예) 직원ID 가 A1인 직원의 이름과 휴대폰 번호를 찾아봅시다

A 테이블

직원ID	이름
A1	김민수
B2	이지은
C3	장범준

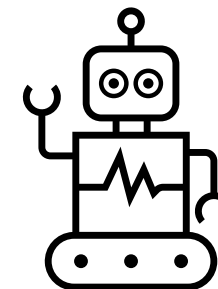
B 테이블

직원ID	구분코드	연락처
A1	휴대폰	010-111-1111
A1	집전화	062-111-1111
B2	집전화	062-222-2222



휴먼

김민수 씨의 휴대폰 번호는  
010-111-1111이겠네!



컴퓨터

## JOIN

## 컴퓨터가 계산하는 방법 (feat.컴퓨터 바보)

A 테이블

직원ID	이름
A1	김민수
B2	이지은
C3	장범준

B 테이블

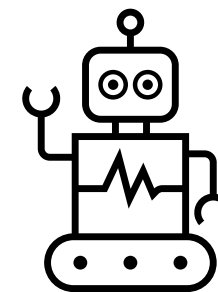
직원ID	구분코드	연락처
A1	휴대폰	010-111-1111
A1	집전화	062-111-1111
B2	집전화	062-222-2222



```
SELECT * FROM A , B ;
```

직원ID	이름	직원ID_1	구분코드	연락처
A1	김민수	A1	휴대폰	010-111-1111
A1	김민수	A1	집전화	062-111-1111
A1	김민수	B2	집전화	062-222-2222
B2	이지은	A1	휴대폰	010-111-1111
B2	이지은	A1	집전화	062-111-1111
B2	이지은	B2	집전화	062-222-2222
C3	장범준	A1	휴대폰	010-111-1111
C3	장범준	A1	집전화	062-111-1111
C3	장범준	B2	집전화	062-222-2222

무식한 방법이지만  
일단 전부 합쳐보자  
→ 카티션 조인



컴퓨터



# JOIN

FROM 뒤에 여러 테이블을 입력만 했다면 **카티션조인** 실행

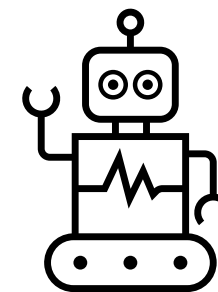
카티션 조인 :

각 테이블의 모든 튜플에 대해  
**모든 경우의 수**를 붙인 것

(컴퓨터에겐 이방법이 최선)

SELECT \* FROM A , B ;

직원ID	이름	직원ID_1	구분코드	연락처
A1	김민수	A1	휴대폰	010-111-1111
A1	김민수	A1	집전화	062-111-1111
A1	김민수	B2	집전화	062-222-2222
B2	이지은	A1	휴대폰	010-111-1111
B2	이지은	A1	집전화	062-111-1111
B2	이지은	B2	집전화	062-222-2222
C3	장범준	A1	휴대폰	010-111-1111
C3	장범준	A1	집전화	062-111-1111
C3	장범준	B2	집전화	062-222-2222



컴퓨터

# JOIN

테이블 간에 **특정 컬럼으로 연결**하면 서로 연관된 데이터만 출력

**SELECT \***

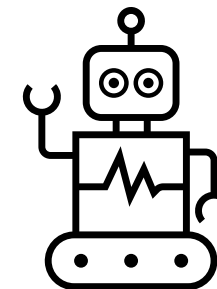
**FROM A, B**

**WHERE A.직원ID = B.직원ID**

A 테이블의 직원ID 컬럼과  
B 테이블의 직원ID가  
똑같은 튜플만 선택하는 조건



A.직원ID		B.직원ID		
직원ID	이름	직원ID_1	구분코드	연락처
A1	김민수	A1	휴대폰	010-111-1111
A1	김민수	A1	집전화	062-111-1111
A1	김민수	B2	집전화	062-222-2222
B2	이지은	A1	휴대폰	010-111-1111
B2	이지은	A1	집전화	062-111-1111
B2	이지은	B2	집전화	062-222-2222
C3	장범준	A1	휴대폰	010-111-1111
C3	장범준	A1	집전화	062-111-1111
C3	장범준	B2	집전화	062-222-2222



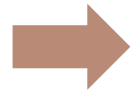
컴퓨터

# JOIN

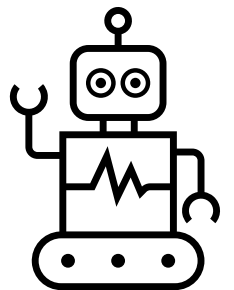
테이블 간에 **특정 컬럼으로 연결**하면 서로 연관된 데이터만 출력

→ **조인조건!**

```
SELECT *  
FROM A , B  
WHERE A.직원ID = B.직원ID  
      AND A.직원ID = 'A1'  
      AND B.구분코드 = '휴대폰';
```



A.직원ID		B.직원ID		
직원ID	이름	직원ID_1	구분코드	연락처
A1	김민수	A1	휴대폰	010-111-1111
A1	김민수	A1	집전화	062-111-1111
B2	이지은	B2	집전화	062-222-2222



컴퓨터

# JOIN

하나의 SQL문으로 두 개 이상의 테이블 결과 값을 연결하여 보고싶을 때 **JOIN** 사용



휴먼

A 테이블

직원ID	이름
A1	김민수
B2	이지은
C3	장범준

B 테이블

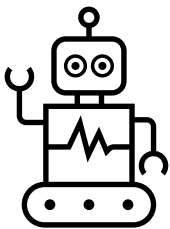
직원ID	구분코드	연락처
A1	휴대폰	010-111-1111
A1	집전화	062-111-1111
B2	집전화	062-222-2222



A.직원ID

B.직원ID

직원ID	이름	직원ID_1	구분코드	연락처
A1	김민수	A1	휴대폰	010-111-1111



컴퓨터

```
SELECT *
FROM A , B
WHERE A.직원ID = B.직원ID
      AND A.직원ID = 'A1'
      AND B.구분코드 = '휴대폰' ;
```



A.직원ID

B.직원ID

직원ID	이름	직원ID_1	구분코드	연락처
A1	김민수	A1	휴대폰	010-111-1111

# JOIN

## JOIN, 꼭 배워야 할까요?

실무에서 굉장히 많이 사용하고, SQLD 및 정보처리기사 시험에서 빠지지 않고 등장하는 개념

```
SELECT *  
FROM A , B, C, ...  
WHERE A.직원ID = B.직원ID  
      AND A.직원ID = 'A1'  
      AND B.구분코드 = '휴대폰' ;  
      AND C.전화번호 = '000-011-111'  
      AND ....
```

### <실무 TIP!>

조인을 사용하는 가장 큰 이유는  
여러 테이블에서 필요한 컬럼들을  
한 번에 검색 할 수 있어  
성능이 증가한다는 장점을  
가졌기 때문

# JOIN

## 실습 문제

문제1) 직원 테이블과 부서 테이블을 부서ID로 연결하여 부서 ID가 서로 같은 모든 정보를 출력하세요

직원ID	패스워드	이름	성별	나이	입사일시	주민등록번호	연봉	부서ID	부서ID_1	부서명	근무지
A0001	12345	김철수	남	25	22/03/21	991212-1566123	2800	D001	D001	인사부	서울
A0002	hello123!	강홍수	남	28	21/09/12	950223-1562867	3000	D002	D002	급여부	서울
A0003	nono132	이현정	여	(null)	22/11/06	000112-4566123	2600	D003	D003	전략기획부	경기
A0004	123123!!	김선미	여	(null)	20/03/11	930722-2766443	4500	D004	D004	SI사업부	경기
A0006	774433	송대주	남	44	15/07/16	790903-1566127	7500	D001	D001	인사부	서울
A0007	pwd123	메이슨	남	40	16/08/19	830629-1676551	6200	D002	D002	급여부	서울
A0008	anjffhgkw1123	송진아	여	47	15/07/16	761212-2508143	7500	D003	D003	전략기획부	경기
A0009	test123	이서연	여	50	13/11/23	730317-259616	9000	D004	D004	SI사업부	경기
A0010	coffeegood!	김홍민	남	52	13/11/23	710513-1572876	9300	D005	D005	사업부	제주

# JOIN

---

## 실습 문제

문제1 ) 직원 테이블과 부서 테이블을 부서ID로 연결하여 부서 ID가 서로 같은 모든 정보를 출력하세요

```
SELECT *  
FROM 직원 A, 부서 B  
WHERE A.부서ID = B.부서ID;
```

## JOIN

## 실습 문제

문제2 ) 직원 테이블과 직원 연락처 테이블에서 연락처 정보가 있는 모든 직원 정보를 출력하세요

직원ID	패스워드	이름	성별	나이	입사일시	주민등록번호	연봉	부서ID	직원ID_1	구분코드	연락처
A0001	12345	김철수	남	25	22/03/21	991212-1566123	2800	D001	A0001	집전화	062-123-1234
A0001	12345	김철수	남	25	22/03/21	991212-1566123	2800	D001	A0001	휴대폰	010-1231-1234
A0002	hello123!	강홍수	남	28	21/09/12	950223-1562867	3000	D002	A0002	집전화	062-254-6342
A0002	hello123!	강홍수	남	28	21/09/12	950223-1562867	3000	D002	A0002	휴대폰	010-2544-6342
A0003	nono132	이현정	여	(null)	22/11/06	000112-4566123	2600	D003	A0003	집전화	062-776-5231
A0003	nono132	이현정	여	(null)	22/11/06	000112-4566123	2600	D003	A0003	휴대폰	010-7766-5231
A0004	123123!!	김선미	여	(null)	20/03/11	930722-2766443	4500	D004	A0004	집전화	062-443-5522
A0004	123123!!	김선미	여	(null)	20/03/11	930722-2766443	4500	D004	A0004	휴대폰	010-4433-5522
A0005	test123	문현철	남	34	(null)	891231-1786155	5000	(null)	A0005	집전화	062-998-7273
A0005	test123	문현철	남	34	(null)	891231-1786155	5000	(null)	A0005	휴대폰	010-9988-7273
A0006	774433	송대주	남	44	15/07/16	790903-1566127	7500	D001	A0006	휴대폰	010-8373-5511
A0007	pwd123	메이슨	남	40	16/08/19	830629-1676551	6200	D002	A0007	휴대폰	010-2323-1133
A0008	anjffhgkw1123	송진아	여	47	15/07/16	761212-2508143	7500	D003	A0008	휴대폰	010-8877-0087



# JOIN

---

## 실습 문제

문제2 ) 직원 테이블과 직원 연락처 테이블에서 연락처 정보가 있는 모든 직원 정보를 출력하세요

```
SELECT *  
FROM 직원 A, 직원연락처 B  
WHERE A.직원ID = B.직원ID;
```

# JOIN

## 실습 문제

문제3 ) 직원 테이블과 직원주소 테이블에서 주소 정보가 있는 직원의 직원ID와 이름, 주소를 출력하세요

⚡ 직원ID	⚡ 이름	⚡ 주소
A0006	송대주	동구 총장로 예술로 12
A0007	메이슨	서울 중구 423
A0008	송진아	서울 관악구 청룡로 66
A0009	이서연	남구 송원로 812
A0010	김홍민	동구 백서로 53

# JOIN

---

## 실습 문제

문제3 ) 직원 테이블과 직원주소 테이블에서 주소 정보가 있는 직원의 직원ID와 이름, 주소를 출력하세요

```
SELECT A.직원ID, A.이름, B.주소  
FROM 직원 A, 직원주소 B  
WHERE A.직원ID = B.직원ID;
```

Step 3.

INNER JOIN  
OUTER JOIN



# INNER JOIN / OUTER JOIN

아래 조인 문법에 나타나지 않은 튜플은?

SELECT \*

FROM 직원 A, 부서 B

WHERE A.부서ID = B.부서ID;

직원 TABLE

직원ID	패스워드	이름	성별	나이	입사일시	주민등록번호	연봉	부서ID
A0001	12345	김철수	남	25	22/03/21	991212-1566123	2800	D001
A0002	hello123!	강홍수	남	28	21/09/12	950223-1562867	3000	D002
A0003	nono132	이현정	여	(null)	22/11/06	000112-4566123	2600	D003
A0004	123123!!	김선미	여	(null)	20/03/11	930722-2766443	4500	D004
A0005	test123	문현철	남	34	(null)	891231-1786155	5000	(null)
A0006	774433	송대주	남	44	15/07/16	790903-1566127	7500	D001
A0007	pwd123	메이슨	남	40	16/08/19	830629-1676551	6200	D002
A0008	anjffhgkw1123	송진아	여	47	15/07/16	761212-2508143	7500	D003
A0009	test123	이서연	여	50	13/11/23	730317-259616	9000	D004
A0010	coffeegood!	김홍민	남	52	13/11/23	710513-1572876	9300	D005

부서 TABLE

부서ID	부서명	근무지
D001	인사부	서울
D002	급여부	서울
D003	전략기획부	경기
D004	SI사업부	경기
D005	사업부	제주
D006	인프라서비스부	서울

# INNER JOIN / OUTER JOIN

직원테이블의 부서ID가 NULL값인 행은 부서테이블과 **조인 실패**

부서테이블의 부서ID가 D006인 값은 직원테이블에 없으므로 **조인 실패**

직원 TABLE

나이	입사일시	주민등록번호	연봉	부서ID
25	22/03/21	991212-1566123	2800	D001
28	21/09/12	950223-1562867	3000	D002
(null)	22/11/06	000112-4566123	2600	D003
(null)	20/03/11	930722-2766443	4500	D004
34	(null)	891231-1786155	5000	(null)
44	15/07/16	790903-1566127	7500	D001
40	16/08/19	830629-1676551	6200	D002
47	15/07/16	761212-2508143	7500	D003
50	13/11/23	730317-259616	9000	D004
52	13/11/23	710513-1572876	9300	D005

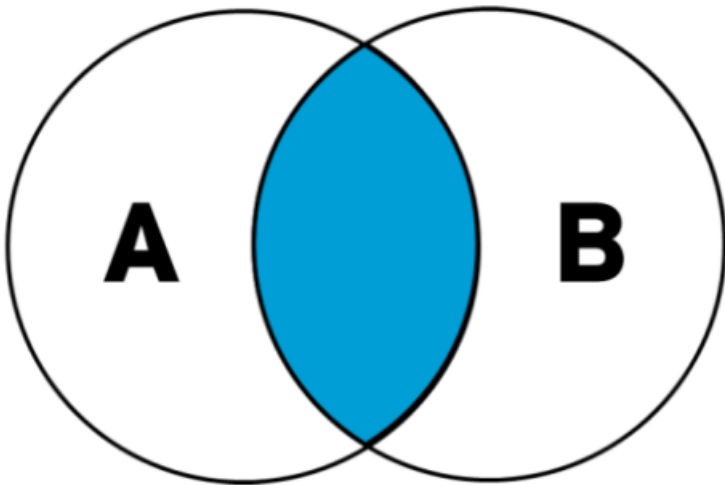
부서 TABLE

부서ID	부서명	근무지
D001	인사부	서울
D002	급여부	서울
D003	전략기획부	경기
D004	SI사업부	경기
D005	사업부	제주
D006	인프라서비스부	서울

# INNER JOIN / OUTER JOIN

INNER 조인은 조건에 해당하는 튜플만 출력(교집합)

## INNER JOIN



SELECT \*

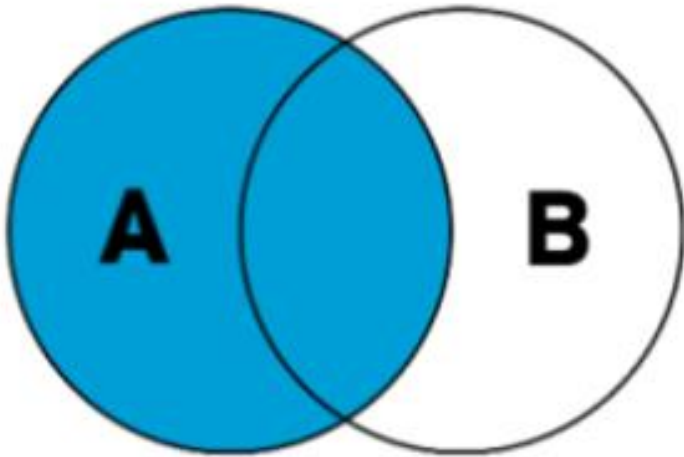
FROM 직원 A, 부서 B

WHERE A.부서ID = B.부서ID;

# INNER JOIN / OUTER JOIN

**OUTER 조인은 조인에 실패한 튜플도 출력**

## OUTER JOIN



**SELECT \***

**FROM 직원 A, 부서 B**

**WHERE A.부서ID = B.부서ID(+);**

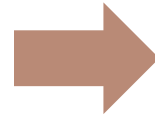


# INNER JOIN / OUTER JOIN

## OUTER 조인은 **조인에 실패한 튜플도 출력**

예) 직원 테이블에서 직원ID와 부서ID, 부서 테이블에서 부서명을 출력하되, 아직 부서를 배정받지 못한 직원의 정보도 출력해주세요

```
SELECT A.직원ID, A.부서ID, B.부서명  
FROM 직원 A, 부서 B  
WHERE A.부서ID = B.부서ID(+);
```



직원ID	부서ID	부서명
A0006	D001	인사부
A0001	D001	인사부
A0007	D002	급여부
A0002	D002	급여부
A0008	D003	전략기획부
A0003	D003	전략기획부
A0009	D004	SI사업부
A0004	D004	SI사업부
A0010	D005	사업부
A0005	(null)	(null)

# INNER JOIN / OUTER JOIN

---

## (+) 기호가 있으면 OUTER JOIN

```
SELECT A.직원ID, A.부서ID, B.부서명  
FROM 직원 A, 부서 B  
WHERE A.부서ID = B.부서ID(+);
```

1. (+) 기호는 오라클에서만 사용
2. (+) 기호가 없으면 INNER JOIN
3. (+) 기호가 붙은 쪽의 반대쪽 테이블을 기준

# INNER JOIN / OUTER JOIN

## 실습 문제

문제 1) 직원 테이블과 직원주소 테이블은 둘 다 직원ID 컬럼을 가지고 있습니다. 이 컬럼으로 동등(“=”) 조인을 하면 서로 연관이 있는 튜플만 출력이 됩니다. 아래와 같은 결과가 되도록 조인으로 출력을 해주세요.

[ 직원 테이블 : 직원ID, 성별, 나이 ]

[ 직원주소 테이블 : 직원ID, 구분코드, 주소 ]

직원_직원ID	성별	나이	주소_직원ID	구분코드	주소
A0006	남	44	A0006	집	동구 총장로 예술로 12
A0007	남	40	A0007	집	서울 중구 423
A0008	여	47	A0008	집	서울 관악구 청룡로 66
A0009	여	50	A0009	집	남구 송원로 812
A0010	남	52	A0010	집	동구 백서로 53

# INNER JOIN / OUTER JOIN

## 실습 문제

문제 1) 직원 테이블과 직원주소 테이블은 둘 다 직원ID 컬럼을 가지고 있습니다. 이 컬럼으로 동등(“=”) 조인을 하면 서로 연관이 있는 튜플만 출력이 됩니다. 아래와 같은 결과가 되도록 조인으로 출력을 해주세요.

[ 직원 테이블 : 직원ID , 성별 , 나이 ]

[ 직원주소 테이블 : 직원ID , 구분코드 , 주소 ]

```
SELECT A.직원ID AS 직원_직원ID , A.성별, A. 나이,  
       B.직원ID AS 주소_직원ID, B.구분코드, B.주소  
FROM 직원 A , 직원주소 B  
WHERE A.직원ID = B.직원ID ;
```

# INNER JOIN / OUTER JOIN

## 실습 문제

문제 2) 1번에서 작성한 쿼리 결과에서 추가로 일반 조건을 작성하려고 합니다.  
직원ID 가 A0007 인 직원의 정보만 출력되도록 작성해주세요.

직원_직원ID	성별	나이	주소_직원ID	구분코드	주소
A0007	남	40	A0007	집	서울 중구 423

# INNER JOIN / OUTER JOIN

## 실습 문제

문제 2) 1번에서 작성한 쿼리 결과에서 추가로 일반 조건을 작성하려고 합니다.  
직원ID 가 A0007 인 직원의 정보만 출력되도록 작성해주세요.

직원_직원ID	성별	나이	주소_직원ID	구분코드	주소
A0007	남	40	A0007	집	서울 중구 423

```
SELECT A.직원ID AS 직원_직원ID , A.성별, A. 나이,  
       B.직원ID AS 주소_직원ID, B.구분코드, B.주소  
FROM 직원 A , 직원주소 B  
WHERE A.직원ID = B.직원ID  
AND A.직원ID = 'A0007' ;
```

# INNER JOIN / OUTER JOIN

## 실습 문제

문제 3) 직원 테이블과 직원주소 테이블을 [직원ID] 컬럼으로 조인하려고 합니다. 이 때 직원주소 테이블에는 없는 직원들의 정보도 출력되도록 아우터조인으로 쿼리를 작성해주세요. (힌트: 기준테이블은 직원 테이블입니다)

[ 직원 테이블 : 직원ID , 이름 , 연봉 ]

[ 직원주소 테이블 : 직원ID , 구분코드 , 주소 ]

직원_직원ID	이름	연봉	주소_직원ID	구분코드	주소
A0006	송대주	7500	A0006	집	동구 충장로 예술로 12
A0007	메이슨	6200	A0007	집	서울 중구 423
A0008	송진아	7500	A0008	집	서울 관악구 청룡로 66
A0009	이서연	9000	A0009	집	남구 송원로 812
A0010	김홍민	9300	A0010	집	동구 백서로 53
A0003	이현정	3600	(null)	(null)	(null)
A0002	강홍수	3000	(null)	(null)	(null)
A0005	문현철	6000	(null)	(null)	(null)
A0004	김선미	4500	(null)	(null)	(null)
A0001	김철수	2800	(null)	(null)	(null)

# INNER JOIN / OUTER JOIN

## 실습 문제

문제 3) 직원 테이블과 직원주소 테이블을 [직원ID] 컬럼으로 조인하려고 합니다. 이 때 직원주소 테이블에는 없는 직원들의 정보도 출력되도록 아우터조인으로 쿼리를 작성해주세요. (힌트: 기준테이블은 직원 테이블입니다)

[ 직원 테이블 : 직원ID , 이름 , 연봉 ]

[ 직원주소 테이블 : 직원ID , 구분코드 , 주소 ]

```
SELECT A.직원ID AS 직원_직원ID , A.이름, A. 연봉,  
       B.직원ID AS 주소_직원ID, B.구분코드, B.주소  
FROM 직원 A , 직원주소 B  
WHERE A.직원ID = B.직원ID(+);
```



# INNER JOIN / OUTER JOIN

## 실습 문제

문제 4) 3번에서 작성한 쿼리에서 추가로 일반 조건을 입력하려고 합니다.  
이번에는 주소정보가 없는 직원들만 출력되도록 쿼리를 작성해주세요.

힌트 : IS NULL , IS NOT NULL

⌘ 직원_직원ID	⌘ 성별	⌘ 나이	⌘ 주소_직원ID	⌘ 구분코드	⌘ 주소
A0003	여	(null)	(null)	(null)	(null)
A0002	남	28	(null)	(null)	(null)
A0005	남	34	(null)	(null)	(null)
A0004	여	(null)	(null)	(null)	(null)
A0001	여	25	(null)	(null)	(null)

# INNER JOIN / OUTER JOIN

---

## 실습 문제

문제 4) 3번에서 작성한 쿼리에서 추가로 일반 조건을 입력하려고 합니다.  
이번에는 주소정보가 없는 직원들만 출력되도록 쿼리를 작성해주세요.

힌트 : IS NULL , IS NOT NULL

```
SELECT A.직원ID AS 직원_직원ID , A.성별, A. 나이,  
       B.직원ID AS 주소_직원ID, B.구분코드, B.주소  
FROM 직원 A , 직원주소 B  
WHERE A.직원ID = B.직원ID(+)  
AND B.직원ID IS NULL ;
```

B.구분코드 IS NULL 이나 B.주소 IS NULL 도 정답

# INNER JOIN / OUTER JOIN

## 실습 문제

문제 5) 직원 테이블과 직원주소 테이블, 직원연락처 테이블을 [직원ID] 컬럼으로 '=' 조인하려고 합니다. INNER JOIN으로 위 세 개의 테이블을 조인해보세요.

[ 직원 테이블 : 직원ID, 이름, 나이 ]

[ 직원주소 테이블 : 주소 ]

[ 직원연락처 테이블 : 연락처 ]

직원ID	이름	나이	연락처	주소
A0006	송대주	44	010-8373-5511	동구 총장로 예술로 12
A0007	메이슨	40	010-2323-1133	서울 중구 423
A0008	송진아	47	010-8877-0087	서울 관악구 청룡로 66

# INNER JOIN / OUTER JOIN

---

## 실습 문제

문제 5) 직원 테이블과 직원주소 테이블, 직원연락처 테이블을 [직원ID] 컬럼으로 '=' 조인하려고 합니다. INNER JOIN으로 위 세 개의 테이블을 조인해보세요.

[ 직원 테이블 : 직원ID, 이름, 나이 ]

[ 직원주소 테이블 : 주소 ]

[ 직원연락처 테이블 : 연락처 ]

```
SELECT A.직원ID, A.이름, A. 나이, B.연락처, C.주소  
FROM 직원 A, 직원연락처 B, 직원주소 C  
WHERE A.직원ID = B.직원ID  
AND B.직원ID = C.직원ID ;
```

# INNER JOIN / OUTER JOIN

## 실습 문제

문제 6) 직원ID 가 'A0001' , 'A0002' , 'A0003' 인 직원에 대해 아래 정보를 출력해주세요

직원ID	이름	입사일시	연락처
A0001	김철수	22/03/21	010-1231-1234
A0002	강홍수	21/09/12	010-2544-6342
A0003	이현정	22/11/06	010-7766-5231

# INNER JOIN / OUTER JOIN

## 실습 문제

문제 6) 직원ID 가 'A0001' , 'A0002' , 'A0003' 인 직원에 대해 아래 정보를 출력해주세요

```
SELECT A.직원ID, A.이름, A.입사일시, B.연락처  
FROM 직원 A, 직원연락처 B  
WHERE A.직원ID = B.직원ID  
AND A.직원ID IN ('A0001', 'A0002', 'A0003')  
AND B.구분코드 = '휴대폰' ;
```

직원ID	이름	입사일시	연락처
A0001	김철수	22/03/21	010-1231-1234
A0002	강홍수	21/09/12	010-2544-6342
A0003	이현정	22/11/06	010-7766-5231

# INNER JOIN / OUTER JOIN

## 실습 문제

문제 7) 직원들의 정보를 출력하되, 소속된 부서ID에 해당하는 부서명도 함께 출력해주세요. (힌트 : 두 테이블 모두 부서ID 컬럼이 있음)

직원ID	이름	부서ID	부서명
A0001	김철수	D001	인사부
A0002	강홍수	D002	급여부
A0003	이현정	D003	전략기획부
A0004	김선미	D004	SI사업부
A0006	송대주	D001	인사부
A0007	메이슨	D002	급여부
A0008	송진아	D003	전략기획부
A0009	이서연	D004	SI사업부
A0010	김홍민	D005	사업부

# INNER JOIN / OUTER JOIN

---

## 실습 문제

**문제 7) 직원들의 정보를 출력하되, 소속된 부서ID에 해당하는 부서명도 함께 출력해주세요. (힌트 : 두 테이블 모두 부서ID 컬럼이 있음)**

```
SELECT A.직원ID, A.이름, A.부서ID, B.부서명  
FROM 직원 A, 부서 B  
WHERE A.부서ID = B.부서ID ;
```



# Step 4.


## ANSI 문법



# ANSI 문법

---

## (+) 기호가 있으면 OUTER JOIN



그럼 다른  
DB에서  
사용하려면?

1. (+) 기호는 오라클에서만 사용
2. (+) 기호가 없으면 INNER JOIN
3. (+) 기호가 붙은 쪽의 반대쪽 테이블을 기준

# ANSI 문법

## ANSI(미국국립표준협회) 문법은 모든 DBMS 에서 사용가능한 조인 문법

```
SELECT ...  
  FROM A테이블 INNER JOIN B테이블 -- 두 테이블을 이너조인  
    ON ( 조인조건 )  
  
SELECT ...  
  FROM A테이블 LEFT OUTER JOIN B테이블 --A테이블을 기준으로 아우터조인  
    ON ( 조인조건 )  
  
SELECT ...  
  FROM A테이블 RIGHT OUTER JOIN B테이블 --B테이블을 기준으로 아우터조인  
    ON ( 조인조건 )  
  
SELECT ...  
  FROM A테이블 FULL OUTER JOIN B테이블 --두 테이블 모두 아우터조인  
    ON ( 조인조건 )
```

# ANSI 문법

## 오라클 조인과 ANSI의 차이점

```
SELECT A.직원ID, A.부서ID, B.부서명  
FROM 직원 A, 부서 B  
WHERE A.부서ID = B.부서ID(+);
```

**WHERE** 부분에 조인조건 사용

FROM 이 깔끔하지만 WHERE 가 복잡

아우터 조인은 (+) 기호를 이용해서 표시

LEFT JOIN 과 RIGHT JOIN 사이를 빠르게 전환 가능

```
SELECT A.직원ID, A.부서ID, B.부서명  
FROM 직원 A LEFT OUTER JOIN 부서 B  
ON (A.부서ID = B.부서ID)
```

**FROM** 뒤의 **ON** 부분에 조인조건을 사용

WHERE절 없이 구현 가능하지만 FROM절이 복잡

모든 DBMS에서 사용 가능

# ANSI 문법

따로 (+) 조건이 없다면 INNER JOIN 으로 연결하고  
FROM .. ON 뒤에 조인조건을 입력,  
WHERE 에 일반조건을 입력하는 방식으로 변환

```
SELECT A.직원ID, A.이름, B.주소  
FROM 직원 A , 직원주소 B  
WHERE A.직원ID = B.직원ID  
AND A.직원ID = 'A0006' ;
```

ANSI로 변경



```
SELECT A.직원ID, A.이름, B.주소  
FROM 직원 A INNER JOIN 직원주소 B  
ON (A.직원ID = B.직원ID)  
WHERE A.직원ID = 'A0006' ;
```

# ANSI 문법

(+) 이 붙은 반대쪽 테이블의 방향으로

LEFT 혹은 RIGHT 를 결정하고 OUTER JOIN 으로 작성

ON 에 조인조건 , WHERE 에 일반조건을 입력하는 방식으로 변환

```
SELECT A.직원ID , A.이름 , B.주소  
FROM 직원 A , 직원주소 B  
WHERE A.직원ID = B.직원ID(+)  
AND A.직원ID BETWEEN 'A0005' AND 'A0008' ;
```

ANSI로 변경



```
SELECT A.직원ID , A.이름 , B.주소  
FROM 직원 A LEFT OUTER JOIN 직원주소 B  
ON ( A.직원ID = B.직원ID )  
WHERE A.직원ID BETWEEN 'A0005' AND 'A0008' ;
```

# ANSI 문법

양쪽 테이블 모두 OUTER JOIN을 하고 싶다면  
FULL OUTER JOIN을 이용 (오라클 문법에는 존재x)

```
SELECT A.직원ID, A.부서ID, B.부서명  
FROM 직원 A FULL OUTER JOIN 부서 B  
ON (A.부서ID = B.부서ID);
```

직원ID	부서ID	부서명
A0001	D001	인사부
A0002	D002	급여부
A0003	D003	전략기획부
A0004	D004	SI사업부
A0005	(null)	(null)
A0006	D001	인사부
A0007	D002	급여부
A0008	D003	전략기획부
A0009	D004	SI사업부
A0010	D005	사업부
(null)	(null)	인프라서비스부

# ANSI 문법

---

## 실습 문제

아래 오라클 방식의 쿼리를 ANSI 방식의 조인문법으로 변경하기

### 1번 문제

```
SELECT A.직원ID, A.이름, B.주소  
FROM 직원 A, 직원주소 B  
WHERE A.직원ID = B.직원ID(+)  
AND A.직원ID IN ('A0005', 'A0008') ;
```



# ANSI 문법

---

## 실습 문제

아래 오라클 방식의 쿼리를 ANSI 방식의 조인문법으로 변경하기

### 1번 문제

```
SELECT A.직원ID, A.이름, B.주소  
FROM 직원 A, 직원주소 B  
WHERE A.직원ID = B.직원ID(+)  
AND A.직원ID IN ('A0005', 'A0008') ;
```

```
SELECT A.직원ID, A.이름, B.주소  
FROM 직원 A LEFT OUTER JOIN 직원주소 B  
ON ( A.직원ID = B.직원ID )  
WHERE A.직원ID IN ('A0005', 'A0008') ;
```

# ANSI 문법

---

## 실습 문제

아래 오라클 방식의 쿼리를 ANSI 방식의 조인문법으로 변경하기

2번 문제

```
SELECT B.직원ID, B.이름, A.주소  
FROM 직원주소 A, 직원 B  
WHERE A.직원ID(+) = B.직원ID ;
```

# ANSI 문법

---

## 실습 문제

아래 오라클 방식의 쿼리를 ANSI 방식의 조인문법으로 변경하기

### 2번 문제

```
SELECT B.직원ID, B.이름, A.주소  
FROM 직원주소 A, 직원 B  
WHERE A.직원ID(+) = B.직원ID ;
```

```
SELECT B.직원ID, B.이름, A.주소  
FROM 직원주소 A RIGHT OUTER JOIN 직원 B  
ON ( A.직원ID = B.직원ID );
```

# ANSI 문법

---

## 실습 문제

아래 오라클 방식의 쿼리를 ANSI 방식의 조인문법으로 변경하기

### 3번 문제

```
SELECT A.직원ID, A.이름, A.나이, B.연락처  
FROM 직원 A, 직원연락처 B  
WHERE A.직원ID = B.직원ID ;
```

# ANSI 문법

---

## 실습 문제

아래 오라클 방식의 쿼리를 ANSI 방식의 조인문법으로 변경하기

### 3번 문제

```
SELECT A.직원ID, A.이름, A.나이, B.연락처  
FROM 직원 A, 직원연락처 B  
WHERE A.직원ID = B.직원ID ;
```

```
SELECT A.직원ID, A.이름, A.나이, B.연락처  
FROM 직원 A INNER JOIN 직원연락처 B  
ON ( A.직원ID = B.직원ID );
```

# ANSI 문법

## <TIP> 실습 문제

직원 테이블과 직원주소 테이블, 직원연락처 테이블을 [직원ID] 컬럼으로 “=” 조인하려고 합니다. INNER JOIN으로 위 세 개의 테이블을 조인해보세요.  
(ANSI 문법 적용)

[ 직원 테이블 : 직원ID, 이름, 나이 ]

[ 직원주소 테이블 : 주소 ]

[ 직원연락처 테이블 : 연락처 ]

⌘ 직원ID	⌘ 이름	⌘ 나이	⌘ 연락처	⌘ 주소
A0006	송대주	44	010-8373-5511	동구 총장로 예술로 12
A0007	메이슨	40	010-2323-1133	서울 중구 423
A0008	송진아	47	010-8877-0087	서울 관악구 청룡로 66

# ANSI 문법

---

## <TIP> 실습 문제

직원 테이블과 직원주소 테이블, 직원연락처 테이블을 [직원ID] 컬럼으로 “=” 조인하려고 합니다. INNER JOIN으로 위 세 개의 테이블을 조인해보세요.  
(ANSI 문법 적용)

[ 직원 테이블 : 직원ID, 이름, 나이 ]

[ 직원주소 테이블 : 주소 ]

[ 직원연락처 테이블 : 연락처 ]

```
SELECT A.직원ID, A.이름, A. 나이, B.연락처, C.주소  
FROM 직원 A INNER JOIN 직원연락처 B  
ON(A.직원ID = B.직원ID) INNER JOIN 직원주소 C  
ON(B.직원ID = C.직원ID);
```