

Chap04. SELECT문의 기본 형식

04-1 실습용 테이블 살펴보기

“ 사원 정보가 들어 있는 **EMP** 테이블

- “ **EMPNO** :사원 번호
- “ **ENAME** :사원 이름
- “ **JOB** :사원 직책
- “ **MGR** :직속 상관의 사원 번호
- “ **HIREDATE** :입사일
- “ **SAL** :급여
- “ **COMM** :급여 외 추가 수당
- “ **DEPTNO** :사원이 속한 부서 번호



04-1 실습용 테이블 살펴보기

“ 회사 부서 정보가 들어있는 **DEPT** 테이블

“ **DEPTNO** : 부서 번호

“ **DNAME** : 부서 이름

“ **LOC** : 부서가 위치한 지역



04-1 실습용 테이블 살펴보기

“ 사원들의 급여 정보가 들어있는 SALGRADE 테이블

“ GRADE : 급여등급

“ LOSAL : 급여등급의 최소 급여액

“ HISAL : 급여등급의 최대 급여액



04-2 데이터를 조회하는 3가지 방법

“ 행 단위로 조회하는 선택션

학번	이름	...	학과 코드	졸업 여부	...
16031055	홍길동		COM		
12071632	성춘향		BN5	졸업	
...		...			
15022655	박문수		MTH		

학생 중 졸업한 학생만 조회

학번	이름	...	학과 코드	졸업 여부	...
12071632	성춘향	...	BN5	졸업	...

04-2 데이터를 조회하는 3가지 방법

“ 열 단위로 조회하는 프로젝트션

학번	이름	...	학과 코드	졸업 여부	...
16031055	홍길동		COM		
12071632	성춘향		BN5	졸업	
...		...			
15022655	박문수		MTH		

전체 학생 데이터에서 학번, 이름, 학과 코드만을 조회

학번	이름	학과 코드
16031055	홍길동	COM
12071632	성춘향	BN5
...
15022655	박문수	MTH

04-2 데이터를 조회하는 3가지 방법

“ 선택선과 프로젝션 함께 사용하기

학번	이름	...	학과 코드	졸업 여부	...
16031055	홍길동		COM		
12071632	성춘향		BN5	졸업	
...		...			
15022655	박문수		MTH		



학번	이름	학과 코드
12071632	성춘향	BN5

04-2 데이터를 조회하는 3가지 방법

“ 두 개 이상의 테이블을 사용하여 조회하는 조인

열 1	열 2	...	열 N	열 1	열 2	...	열 N
행 1				행 1			
행 2				행 2			
...				...			
행 N				행 N			

사이좋게 하나의 테이블인 것처럼 붙여서 사용하면 조인

04-3 SQL의 기본 뼈대, SELECT절과 FROM절

“ SELECT 와 FROM

“ SELECT [전체를 뜻하는 * 또는 조회할 열 이름], ...

“ FROM [테이블 명]



04-3 SQL의 기본 뼈대, SELECT절과 FROM절

“ * 로 테이블 전체 열 출력하기
(EMP테이블 전체 열 출력하기)

“ SELECT *
FROM EMP;

	123 EMPNO	ASC ENAME	ASC JOB	123 MGR	🕒 HIREDATE	123 SAL	123 COMM	123 DEPTNO
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20
2	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
3	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
4	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20
5	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
6	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
7	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
8	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20
9	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10
10	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
11	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20
12	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30
13	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20
14	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10



04-3 SQL의 기본 뼈대, SELECT절과 FROM절

“ 테이블 부분 열 출력하기

(EMP테이블에서 사원번호, 이름, 부서번호 조회)

“ SELECT EMPNO, ENAME, DEPTNO
FROM EMP;

	123 EMPNO ▼	ASC ENAME ▼	123 DEPTNO ▼
1	7,369	SMITH	20
2	7,499	ALLEN	30
3	7,521	WARD	30
4	7,566	JONES	20
5	7,654	MARTIN	30
6	7,698	BLAKE	30
7	7,782	CLARK	10
8	7,788	SCOTT	20
9	7,839	KING	10
10	7,844	TURNER	30
11	7,876	ADAMS	20
12	7,900	JAMES	30
13	7,902	FORD	20
14	7,934	MILLER	10




04-4 중복 데이터를 삭제하는 DISTINCT

“ DISTINCT로 열 중복 제거하기 (열이 한 개인 경우)

“ SELECT **DISTINCT** deptno
FROM emp;

	123 DEPTNO
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	



	123 DEPTNO
1	30
2	20
3	10

04-4 중복 데이터를 삭제하는 DISTINCT

“ DISTINCT로 열 중복 제거하기 (열이 여러 개인 경우)

“ SELECT **DISTINCT** job , deptno
FROM emp;

	job	deptno
1		
2		
3		
4	MANAGER	20
5		
6	MANAGER	30
7	MANAGER	10
8		
9	PRESIDENT	10
10		
11		
12	CLERK	30
13		
14	CLERK	10

	job	deptno
1	MANAGER	20
2	PRESIDENT	10
3	CLERK	10
4		
5		
6	MANAGER	30
7	MANAGER	10
8	CLERK	30
9		

04-4 중복 데이터를 삭제하는 DISTINCT

“ ALL로 중복되는 열 제거 없이 그대로 출력하기

(DISTINCT 가 없으면 기본값으로 출력)

“ SELECT **ALL** job, deptno
FROM EMP;

	job	deptno
1	CLERK	20
2	SALESMAN	30
3	SALESMAN	30
4	MANAGER	20
5	SALESMAN	30
6	MANAGER	30
7	MANAGER	10
8	ANALYST	20
9	PRESIDENT	10
10	SALESMAN	30
11	CLERK	20
12	CLERK	30
13	ANALYST	20
14	CLERK	10

04-5 한눈에 보기 좋게 별칭 설정하기

“ 열에 연산식을 추가하기

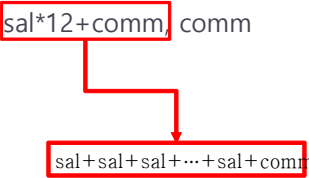
“ SELECT ename, sal, **sal*12+comm**, comm
FROM EMP;

	abc ENAME ▼	123 SAL ▼	123 SAL*12+COMM ▼	123 COMM ▼
1	SMITH	800	[NULL]	[NULL]
2	ALLEN	1,600	19,500	300
3	WARD	1,250	15,500	500
4	JONES	2,975	[NULL]	[NULL]
5	MARTIN	1,250	16,400	1,400
6	BLAKE	2,850	[NULL]	[NULL]
7	CLARK	2,450	[NULL]	[NULL]
8	SCOTT	3,000	[NULL]	[NULL]
9	KING	5,000	[NULL]	[NULL]
10	TURNER	1,500	18,000	0
11	ADAMS	1,100	[NULL]	[NULL]
12	JAMES	950	[NULL]	[NULL]
13	FORD	3,000	[NULL]	[NULL]
14	MILLER	1,300	[NULL]	[NULL]

04-5 한눈에 보기 좋게 별칭 설정하기

“ 열에 연산식을 추가하기

“ SELECT ename, sal, sal*12+comm, comm
FROM EMP;




sal+sal+sal+...+sal+comm

The diagram illustrates the expansion of the SQL expression `sal*12+comm`. A red box highlights `sal*12+comm` in the SQL statement above. A red line descends from this box, turns right, and then descends again with an arrow pointing to a second red box. This second box contains the expression `sal+sal+sal+...+sal+comm`, showing that `sal*12` is equivalent to adding `sal` to itself 12 times.

04-5 한눈에 보기 좋게 별칭 설정하기

“ 열에 별칭을 지정하기

“ SELECT ename, sal, **sal*12+comm AS 연봉**, comm
FROM EMP;



	EMP ENAME	123 SAL	123 연봉	123 COMM
1	SMITH	800	[NULL]	[NULL]
2	ALLEN	1,600	19,500	300
3	WARD	1,250	15,500	500
4	JONES	2,975	[NULL]	[NULL]
5	MARTIN	1,250	16,400	1,400
6	BLAKE	2,850	[NULL]	[NULL]
7	CLARK	2,450	[NULL]	[NULL]
8	SCOTT	3,000	[NULL]	[NULL]
9	KING	5,000	[NULL]	[NULL]
10	TURNER	1,500	18,000	0
11	ADAMS	1,100	[NULL]	[NULL]
12	JAMES	950	[NULL]	[NULL]
13	FORD	3,000	[NULL]	[NULL]
14	MILLER	1,300	[NULL]	[NULL]

04-5 한눈에 보기 좋게 별칭 설정하기

“ 별칭을 지정하는 방식

사용 방법	설명
SAL*12+COMM ANNSAL	연산 및 가공된 문장 이후 한 칸 띄우고 별칭 지정
SAL*12+COMM "ANNSAL"	연산 및 가공된 문장 이후 한 칸 띄우고 별칭을 큰따옴표(" ")로 묶어 지정
SAL*12+COMM AS ANNSAL	연산 및 가공된 문장 이후 한 칸 띄운 후 'AS', 한 칸 뒤에 별칭 지정
SAL*12+COMM AS "ANNSAL"	연산 및 가공된 문장 이후 한 칸 띄운 후 'AS', 한 칸 뒤에 별칭을 큰따옴표(" ")로 묶어 지정



04-5 한눈에 보기 좋게 별칭 설정하기

“ 별칭을 지정하는 방식

사용 방법	설명
SAL*12+COMM ANNSAL	연산 및 가공된 문장 이후 한 칸 띄우고 별칭 지정
SAL*12+COMM "ANNSAL"	연산 및 가공된 문장 이후 한 칸 띄우고 별칭을 큰따옴표(" ")로 묶어 지정
SAL*12+COMM AS ANNSAL	연산 및 가공된 문장 이후 한 칸 띄운 후 'AS', 한 칸 뒤에 별칭 지정
SAL*12+COMM AS "ANNSAL"	연산 및 가공된 문장 이후 한 칸 띄운 후 'AS', 한 칸 뒤에 별칭을 큰따옴표(" ")로 묶어 지정

실무에서 주로 사용하는 별칭 지정 방식

> 열이 많아지면 어떤 게 별칭인지 헷갈리기 때문.

04-6 원하는 순서로 출력 데이터를 정렬하는 ORDER BY

“ SELECT문 가장 마지막 절

SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]

FROM [조회할 테이블 이름]

· (그 밖의 절)
·

ORDER BY [정렬하려는 열 이름(여러 열 지정 가능)] [정렬 옵션];



04-6 원하는 순서로 출력 데이터를 정렬하는 ORDER BY

“ SELECT문 가장 마지막 절

SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]

FROM [조회할 테이블 이름]

· (그 밖의 절)
·

desc / asc

desc : 내림차순

asc : 오름차순(기본값)

ORDER BY [정렬하려는 열 이름(여러 열 지정 가능)] [정렬 옵션];

04-6 원하는 순서로 출력 데이터를 정렬하는 ORDER BY

“ ORDER BY 오름차순 정렬

```
“ SELECT *  
FROM EMP  
ORDER BY sal;
```

	123 EMPNO	ABC ENAME	ABC JOB	123 MGR	HIREDATE	123 SAL	123 COMM	123 DEPTNO
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20
2	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30
3	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20
4	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
5	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
6	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10
7	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
8	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
9	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
10	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
11	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20
12	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20
13	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20
14	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10

04-6 원하는 순서로 출력 데이터를 정렬하는 ORDER BY

“ ORDER BY 내림차순 정렬

```
“ SELECT *  
FROM EMP  
ORDER BY sal DESC;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10
2	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20
3	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20
4	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20
5	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
6	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
7	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
8	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
9	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10
10	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
11	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
12	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20
13	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30
14	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20

04-6 원하는 순서로 출력 데이터를 정렬하는 ORDER BY

“ ORDER BY 여러 컬럼 지정하기

“ SELECT *

FROM EMP

ORDER BY deptno, sal;

2순위

1순위

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300		10
2	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450		10
3	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000		10
4	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800		20
5	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100		20
6	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975		20
7	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000		20
8	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000		20
9	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950		30
10	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
11	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
12	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
13	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
14	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850		30

04-6 원하는 순서로 출력 데이터를 정렬하는 ORDER BY

“ ORDER BY 여러 컬럼 내림차순, 오름차순 같이 사용

“ SELECT *

FROM EMP

ORDER BY deptno ASC, sal DESC;

2순위

1순위

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10
2	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
3	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10
4	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20
5	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20
6	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20
7	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20
8	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20
9	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
10	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
11	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
12	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
13	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
14	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30

04-6 원하는 순서로 출력 데이터를 정렬하는 ORDER BY

“ ORDER BY 사용 주의사항

- “ 꼭 필요할 때가 아니면 사용 X
- “ 자원 낭비 多 > 비용 소모 多
- “ 2개 데이터 정렬 / 10개 데이터 정렬 / 100개 데이터 정렬
시간 차이가 극심함

따라서 , ORDER BY 를 막 사용하면, 서비스 응답 시간이 오래 걸림.



Chap05. 더 정확하고 다양하게 결과를 출력하는 WHERE절과 연산자

05-1 필요한 데이터만 쏙 출력하는 WHERE절

“ WHERE 기본형

```
SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]  
FROM   [조회할 테이블 이름]  
WHERE  [조회할 행을 선별하기 위한 조건식];
```



05-1 필요한 데이터만 속 출력하는 WHERE절

“ WHERE

“ 부서번호가 30 인 데이터 출력

“ SELECT *
FROM EMP
WHERE DEPTNO = 30;

조건식을 대입하여 결과가 참(true)인
데이터만 출력

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	
7369	SMITH	CLERK	7902	1980/12/17	800		20	
7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30	true
7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30	
7566	JONES	MANAGER	7839	1981/04/02	2975		20	
7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30	
7698	BLAKE	MANAGER	7839	1981/05/01	2850		30	
7782	CLARK	MANAGER	7839	1981/06/09	2450		10	
7788	SCOTT	ANALYST	7566	1987/04/19	3000		20	
7839	KING	PRESIDENT		1981/11/17	5000		10	
7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30	
7876	ADAMS	CLERK	7788	1987/05/23	1100		20	
7900	JAMES	CLERK	7698	1981/12/03	950		30	false
7902	FORD	ANALYST	7566	1981/12/03	3000		20	

05-1 필요한 데이터만 속 출력하는 WHERE절

“ WHERE

“ 부서번호가 30 인 데이터 출력

“ SELECT *
FROM EMP

WHERE DEPTNO = 30;

	123 EMPNO	ASC ENAME	ASC JOB	123 MGR	HIREDATE	123 SAL	123 COMM	123 DEPTNO
1	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
2	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
3	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
4	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
5	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
6	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30

05-2 여러 개 조건식을 사용하는 AND, OR 연산자

“ AND

피연산자 1 \ 피연산자 2	true	false
true	true	false
false	false	false

“ SELECT *
FROM EMP
WHERE DEPTNO = 30
AND JOB = 'SALESMAN' ;

	123 EMPNO ▼	ABC ENAME ▼	ABC JOB ▼	123 MGR ▼	🕒 HIREDATE ▼	123 SAL ▼	123 COMM ▼	123 DEPTNO ▼
1	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30 📄
2	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30 📄
3	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30 📄
4	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30 📄

05-2 여러 개 조건식을 사용하는 AND, OR 연산자

“ OR

피연산자 1 \ 피연산자 2	true	false
true	true	true
false	true	false

“ SELECT *
FROM EMP
WHERE DEPTNO = 30
OR JOB = 'SALESMAN' ;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
2	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
3	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
4	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
5	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
6	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30

05-2 여러 개 조건식을 사용하는 AND, OR 연산자

“ WHERE 절 조건식의 개수

- “ 사실상 제한이 없다.
- “ 조건식 사이에 계속 AND / OR 을 사용하면 가능하다.
- “ AND / OR 같은 경우는 가장 많이 사용한다.





05-3 연산자 종류와 활용 방법 알아보기

“ 산술 연산자 (사칙연산)

} + , - , * , /

```
“ SELECT *  
  FROM EMP  
  WHERE SAL * 12 = 36000 ;
```

	123 EMPNO ▼	123 ENAME ▼	123 JOB ▼	123 MGR ▼	123 HIREDATE ▼	123 SAL ▼	123 COMM ▼	123 DEPTNO ▼
1	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20 
2	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20 

05-3 연산자 종류와 활용 방법 알아보기

“ 비교 연산자

“ 대소 비교 연산자

연산자	사용법	설명
>	$A > B$	A 값이 B 값을 초과할 경우 true
>=	$A \geq B$	A 값이 B 값 이상일 경우 true
<	$A < B$	A 값이 B 값 미만일 경우 true
<=	$A \leq B$	A 값이 B 값 이하일 경우 true



05-3 연산자 종류와 활용 방법 알아보기


“ 비교 연산자

“ 대소 비교 연산자

“ SELECT *

FROM EMP

WHERE SAL >= 3000;

	123 EMPNO ▼	ABC ENAME ▼	ABC JOB ▼	123 MGR ▼	🕒 HIREDATE ▼	123 SAL ▼	123 COMM ▼	123 DEPTNO ▼
1	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20 
2	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10 
3	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20 

05-3 연산자 종류와 활용 방법 알아보기

“ 비교 연산자

“ 대소 비교 연산자

“ SELECT *

FROM EMP

WHERE ENAME >= 'F' ;

	123 EMPNO ▼	ABC ENAME ▼	ABC JOB ▼	123 MGR ▼	🕒 HIREDATE ▼	123 SAL ▼	123 COMM ▼	123 DEPTNO ▼
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20 ➡
2	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30 ➡
3	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20 ➡
4	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30 ➡
5	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20 ➡
6	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10 ➡
7	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30 ➡
8	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30 ➡
9	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20 ➡
10	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10 ➡

05-3 연산자 종류와 활용 방법 알아보기

“ 비교 연산자

“ 대소 비교 연산자

“ SELECT *

FROM EMP

WHERE ENAME < 'FORZ' ;

	123 EMPNO ▼	ENC ENAME ▼	ENC JOB ▼	123 MGR ▼	HIREDATE ▼	123 SAL ▼	123 COMM ▼	123 DEPTNO ▼
1	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
2	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
3	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
4	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20
5	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20

05-3 연산자 종류와 활용 방법 알아보기

“ 비교 연산자

“ 등가 비교 연산자

연산자	사용법	의미
=	A = B	A 값이 B 값과 같을 경우 true, 다를 경우 false 반환
!=	A != B	A 값과 B 값이 다를 경우 true, 같을 경우 false 반환
<>	A <> B	
^=	A ^= B	



05-3 연산자 종류와 활용 방법 알아보기

“ 비교 연산자

“ 등가 비교 연산자

“ SELECT *

FROM EMP

WHERE SAL != 3000 ;

WHERE SAL <> 3000;

WHERE SAL ^= 3000;

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	1980-12-17 00:00:00.000	800	[NULL]	20
7499	ALLEN	SALESMAN	7698	1981-02-20 00:00:00.000	1,600	300	30
7521	WARD	SALESMAN	7698	1981-02-22 00:00:00.000	1,250	500	30
7566	JONES	MANAGER	7839	1981-04-02 00:00:00.000	2,975	[NULL]	20
7654	MARTIN	SALESMAN	7698	1981-09-28 00:00:00.000	1,250	1,400	30
7698	BLAKE	MANAGER	7839	1981-05-01 00:00:00.000	2,850	[NULL]	30
7782	CLARK	MANAGER	7839	1981-06-09 00:00:00.000	2,450	[NULL]	10
7839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10
7844	TURNER	SALESMAN	7698	1981-09-08 00:00:00.000	1,500	0	30
7876	ADAMS	CLERK	7788	1987-05-23 00:00:00.000	1,100	[NULL]	20
7900	JAMES	CLERK	7698	1981-12-03 00:00:00.000	950	[NULL]	30
7934	MILLER	CLERK	7782	1982-01-23 00:00:00.000	1,300	[NULL]	10

05-3 연산자 종류와 활용 방법 알아보기

“ 논리 부정 연산자

“ NOT

“ SELECT *

FROM EMP

WHERE NOT SAL = 3000 ;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20
2	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
3	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
4	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20
5	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
6	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
7	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
8	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10
9	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
10	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20
11	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30
12	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10

05-3 연산자 종류와 활용 방법 알아보기

“ IN 연산자

```
SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]  
FROM   [조회할 테이블 이름]  
WHERE  열 이름 IN (데이터1, 데이터2, ... 데이터N);
```



05-3 연산자 종류와 활용 방법 알아보기

“ IN 연산자 설명 전에 OR을 사용하면 ?

“ SELECT *

FROM EMP

WHERE JOB = 'MANAGER'

OR JOB = 'SALESMAN'

OR JOB = 'CLERK' ;

“ SELECT *

FROM EMP

WHERE JOB

IN ('MANAGER' , 'SALESMAN' , 'CLERK') ;

	123 EMPNO	ASC ENAME	ASC JOB	123 MGR	HIREDATE	123 SAL	123 COMM	123 DEPTNO
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20
2	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
3	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
4	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20
5	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
6	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
7	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
8	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
9	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20
10	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30
11	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10

05-3 연산자 종류와 활용 방법 알아보기

“ IN 연산자의 부정은 ?

```
“ SELECT *  
  FROM EMP  
  WHERE JOB != 'MANAGER'  
  AND JOB <> 'SALESMAN'  
  AND JOB ^= 'CLERK';
```

```
“ SELECT *  
  FROM EMP  
  WHERE JOB  
  NOT IN ( 'MANAGER' , 'SALESMAN' , 'CLERK' ) ;
```

	123 EMPNO	ASC ENAME	ASC JOB	123 MGR	HIREDATE	123 SAL	123 COMM	123 DEPTNO
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20
2	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
3	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
4	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20
5	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
6	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
7	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
8	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
9	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20
10	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30
11	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10

05-3 연산자 종류와 활용 방법 알아보기

“ BETWEEN A AND B 연산자

```
SELECT [조회할 열1 이름], [열2 이름], ..., [열N 이름]  
FROM   [조회할 테이블 이름]  
WHERE  열 이름 BETWEEN 최솟값 AND 최댓값;
```



05-3 연산자 종류와 활용 방법 알아보기

“ BETWEEN A AND B 연산자

```
“ SELECT *  
  FROM EMP  
  WHERE SAL >= 2000  
  AND   SAL <= 3000 ;
```

```
“ SELECT *  
  FROM EMP  
  WHERE SAL BETWEEN 2000 AND 3000 ;
```

	123 EMPNO	ABC ENAME	ABC JOB	123 MGR	🕒 HIREDATE	123 SAL	123 COMM	123 DEPTNO
1	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20
2	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
3	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
4	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20
5	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20

05-3 연산자 종류와 활용 방법 알아보기

“ BETWEEN A AND B 연산자의 부정

```
“ SELECT *  
  FROM EMP  
  WHERE SAL <= 2000  
  AND   SAL >= 3000 ;
```

```
“ SELECT *  
  FROM EMP  
  WHERE SAL NOT BETWEEN 2000 AND 3000 ;
```

	123 EMPNO ▼	ABC ENAME ▼	ABC JOB ▼	123 MGR ▼	HIREDATE ▼	123 SAL ▼	123 COMM ▼	123 DEPTNO ▼
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20
2	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
3	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
4	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
5	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10
6	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
7	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20
8	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30
9	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10

05-3 연산자 종류와 활용 방법 알아보기

“ LIKE 연산자

```
“ SELECT *  
  FROM EMP  
  WHERE ENAME LIKE 'S%';
```

	123 EMPNO ▼	ABC ENAME ▼	ABC JOB ▼	123 MGR ▼	HIREDATE ▼	123 SAL ▼	123 COMM ▼	123 DEPTNO ▼
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20
2	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20

05-3 연산자 종류와 활용 방법 알아보기

“ 와일드 카드

종류	의미
_	어떤 값이든 상관없이 한 개의 문자 데이터를 의미
%	길이와 상관없이(문자 없는 경우도 포함) 모든 문자 데이터를 의미



05-3 연산자 종류와 활용 방법 알아보기

“ LIKE 연산자

```
“ SELECT *  
  FROM EMP  
  WHERE ENAME LIKE ‘_L%’ ;
```

	123 EMPNO ▼	ABC ENAME ▼	ABC JOB ▼	123 MGR ▼	HIREDATE ▼	123 SAL ▼	123 COMM ▼	123 DEPTNO ▼
1	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
2	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
3	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10

05-3 연산자 종류와 활용 방법 알아보기

“ LIKE 연산자

```
“ SELECT *  
  FROM EMP  
  WHERE ENAME LIKE ‘_L%’ ;
```

	123 EMPNO ▼	ABC ENAME ▼	ABC JOB ▼	123 MGR ▼	HIREDATE ▼	123 SAL ▼	123 COMM ▼	123 DEPTNO ▼
1	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
2	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
3	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10

05-3 연산자 종류와 활용 방법 알아보기

“ LIKE 연산자

```
“ SELECT *  
  FROM EMP  
  WHERE ENAME LIKE ' %AM% ' ;
```

	123 EMPNO ▼	ABC ENAME ▼	ABC JOB ▼	123 MGR ▼	🕒 HIREDATE ▼	123 SAL ▼	123 COMM ▼	123 DEPTNO ▼
1	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20 📄
2	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30 📄

05-3 연산자 종류와 활용 방법 알아보기

“ LIKE 연산자의 부정

```
“ SELECT *  
  FROM EMP  
  WHERE ENAME NOT LIKE ' %AM% ' ;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20
2	7,499	ALLEN	SALESMAN	7,698	1981-02-20 00:00:00.000	1,600	300	30
3	7,521	WARD	SALESMAN	7,698	1981-02-22 00:00:00.000	1,250	500	30
4	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20
5	7,654	MARTIN	SALESMAN	7,698	1981-09-28 00:00:00.000	1,250	1,400	30
6	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
7	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
8	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20
9	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10
10	7,844	TURNER	SALESMAN	7,698	1981-09-08 00:00:00.000	1,500	0	30
11	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20
12	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10

05-3 연산자 종류와 활용 방법 알아보기

“ 와일드카드 문자가 데이터로 있다면 ?

“ 예를 들어, A_ABCD 라는 데이터가 있다면

LIKE 문을 어떻게 작성해야 할까 ?

- SELECT *

FROM TABLE_NAME

WHERE COLUMN_NAME LIKE ' A\A% ' **ESCAPE '\'** ;

※ **ESCAPE** 문자를 지정하는 방식은 자주 사용하지 않으며,
혹여나 사용할 경우를 위해 이런 방식이 있다 라고 참고만 할 것.



05-3 연산자 종류와 활용 방법 알아보기

“ ISNULL 연산자

“ NULL : 데이터 값이 존재하지 않는 상태

“ 일반 연산자의 사용이 무의미 (비교 결과 값이 NULL 이다.)

- $\text{NULL} + 100 = \text{NULL}$
- $\text{NULL} > 100 = \text{NULL}$
- $\infty + 100 = \infty$
- $? > 100 = ?$

“ IS NULL : 대상 데이터가 NULL일 때 true

“ IS NOT NULL : 대상 데이터가 NULL이 아닐 때 true



05-3 연산자 종류와 활용 방법 알아보기

“ ISNULL 연산자

“ SELECT *
FROM EMP
WHERE COMM = NULL ;
(결과 값 없음)

“ SELECT *
FROM EMP
WHERE COMM ISNULL ;



	123 EMPNO	ABC ENAME	ABC JOB	123 MGR	HIREDATE	123 SAL	123 COMM	123 DEPTNO
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20
2	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20
3	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30
4	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10
5	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20
6	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10
7	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20
8	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30
9	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20
10	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10

05-3 연산자 종류와 활용 방법 알아보기

“ NULL과 AND 연산

	TRUE	FALSE	NULL
TRUE	TRUE	FALSE	NULL
FALSE	FALSE	FALSE	FALSE
NULL	NULL	FALSE	NULL

“ SELECT *
FROM EMP
WHERE SAL > NULL
AND COMM ISNULL ;

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO

05-3 연산자 종류와 활용 방법 알아보기

“ NULL과 OR 연산

	TRUE	FALSE	NULL
TRUE	TRUE	TRUE	TRUE
FALSE	TRUE	FALSE	NULL
NULL	TRUE	NULL	NULL

```
“ SELECT *  
FROM EMP  
WHERE SAL > NULL  
OR COMM ISNULL;
```

	123 EMPNO	88C ENAME	88C JOB	123 MGR	HIREDATE	123 SAL	123 COMM	123 DEPTNO
1	7,369	SMITH	CLERK	7,902	1980-12-17 00:00:00.000	800	[NULL]	20 05
2	7,566	JONES	MANAGER	7,839	1981-04-02 00:00:00.000	2,975	[NULL]	20 05
3	7,698	BLAKE	MANAGER	7,839	1981-05-01 00:00:00.000	2,850	[NULL]	30 05
4	7,782	CLARK	MANAGER	7,839	1981-06-09 00:00:00.000	2,450	[NULL]	10 05
5	7,788	SCOTT	ANALYST	7,566	1987-04-19 00:00:00.000	3,000	[NULL]	20 05
6	7,839	KING	PRESIDENT	[NULL]	1981-11-17 00:00:00.000	5,000	[NULL]	10 05
7	7,876	ADAMS	CLERK	7,788	1987-05-23 00:00:00.000	1,100	[NULL]	20 05
8	7,900	JAMES	CLERK	7,698	1981-12-03 00:00:00.000	950	[NULL]	30 05
9	7,902	FORD	ANALYST	7,566	1981-12-03 00:00:00.000	3,000	[NULL]	20 05
10	7,934	MILLER	CLERK	7,782	1982-01-23 00:00:00.000	1,300	[NULL]	10 05

05-3 연산자 종류와 활용 방법 알아보기

“ 집합 연산자

- “ UNION : 중복이 제거되는 합집합
- “ UNION ALL : 중복을 허용하는 합집합
- “ MINUS : 차집합
- “ INTERSECT : 교집합



05-3 연산자 종류와 활용 방법 알아보기

“ 집합 연산자 실습 전 주의점

“ 1. 출력 열(COLUMN)의 개수가 같아야 함.

```
“ SELECT EMPNO , ENAME , SAL , DEPTNO  
   FROM EMP  
   WHERE DEPTNO = 10  
   UNION  
   SELECT EMPNO , ENAME , SAL  
   FROM EMP  
   WHERE DEPTNO = 20 ;
```

“ 에러 : 질의 블록은 부정확한 수의
결과 열을 가지고 있습니다.



05-3 연산자 종류와 활용 방법 알아보기

“ 집합 연산자 실습 전 주의점

“ 2. 출력하는 열의 자료형이 같아야 함.

```
“ SELECT EMPNO , ENAME , SAL , DEPTNO  
   FROM EMP  
   WHERE DEPTNO = 10  
   UNION  
   SELECT ENAME , EMPNO , DEPTNO , SAL  
   FROM EMP  
   WHERE DEPTNO = 20 ;
```

“ 에러 : 대응하는 식과 같은 데이터
유형이어야 합니다.



05-3 연산자 종류와 활용 방법 알아보기

“ 집합 연산자 실습 전 주의점

“ 3. 출력 열 개수와 자료형만 같다면 ?

```
“ SELECT EMPNO , ENAME , SAL , DEPTNO  
   FROM EMP  
   WHERE DEPTNO = 10  
   UNION  
   SELECT SAL , JOB , DEPTNO , S  
   FROM EMP  
   WHERE DEPTNO = 20 ;
```

	123 EMPNO	ABC ENAME	123 SAL	123 DEPTNO
1	800	CLERK	20	800
2	1,100	CLERK	20	1,100
3	2,975	MANAGER	20	2,975
4	3,000	ANALYST	20	3,000
5	7,782	CLARK	2,450	10
6	7,839	KING	5,000	10
7	7,934	MILLER	1,300	10

05-3 연산자 종류와 활용 방법 알아보기

“ 집합 연산자

“ UNION : 중복이 제거되는 합집합

“ SELECT EMPNO , ENAME , SAL , DEPTNO
FROM EMP
WHERE DEPTNO = 10

UNION

SELECT EMPNO , ENAME , SAL , DEPTNO
FROM EMP
WHERE DEPTNO = 10;

	123 EMPNO	106 ENAME	123 SAL	123 DEPTNO
1	7,782	CLARK	2,450	10
2	7,839	KING	5,000	10
3	7,934	MILLER	1,300	10

05-3 연산자 종류와 활용 방법 알아보기

“ 집합 연산자

“ UNION ALL : 중복을 허용하는 합집합

“ SELECT EMPNO , ENAME , SAL , DEPTNO
FROM EMP
WHERE DEPTNO = 10

UNION ALL

SELECT EMPNO , ENAME , SAL , DEPTNO
FROM EMP
WHERE DEPTNO = 10;

	123 EMPNO	ABC ENAME	123 SAL	123 DEPTNO
1	7,782	CLARK	2,450	10
2	7,839	KING	5,000	10
3	7,934	MILLER	1,300	10
4	7,782	CLARK	2,450	10
5	7,839	KING	5,000	10
6	7,934	MILLER	1,300	10

05-3 연산자 종류와 활용 방법 알아보기

“ 집합 연산자

“ INTERSECT : 교집합

“ SELECT EMPNO , ENAME , SAL , DEPTNO
FROM EMP

INTERSECT

SELECT EMPNO , ENAME , SAL , DEPTNO
FROM EMP

WHERE DEPTNO = 10;

	123 EMPNO	ABC ENAME	123 SAL	123 DEPTNO
1	7,782	CLARK	2,450	10
2	7,839	KING	5,000	10
3	7,934	MILLER	1,300	10

05-3 연산자 종류와 활용 방법 알아보기

“ 집합 연산자

“ MINUS: 차집합

“ SELECT EMPNO , ENAME , SAL , DEPTNO
FROM EMP

MINUS

SELECT EMPNO , ENAME , SAL , DEPTNO
FROM EMP
WHERE DEPTNO = 10;

	123 EMPNO ▼	ABC ENAME ▼	123 SAL ▼	123 DEPTNO ▼
1	7,369	SMITH	800	20
2	7,499	ALLEN	1,600	30
3	7,521	WARD	1,250	30
4	7,566	JONES	2,975	20
5	7,654	MARTIN	1,250	30
6	7,698	BLAKE	2,850	30
7	7,788	SCOTT	3,000	20
8	7,844	TURNER	1,500	30
9	7,876	ADAMS	1,100	20
10	7,900	JAMES	950	30
11	7,902	FORD	3,000	20

05-3 연산자 종류와 활용 방법 알아보기

“ 연산자들의 우선 순위

우선 순위	연산자	설명
높음(↑)	*, /	산술 연산자 곱하기 나누기
	+, -	산술 연산자 더하기 빼기
	=, !=, ^=, <>, >, >=, <, <=	대소 비교연산자
	IS (NOT) NULL, (NOT) LIKE, (NOT) IN	(그 외) 비교연산자
	BETWEEN A AND B	BETWEEN 연산자
	NOT	논리 부정 연산자 NOT
	AND	논리 연산자 AND
낮음(↓)	OR	논리 연산자 OR