REPORT

과	목 :	데이터베이스이론및실습
제출일자	:	2024.03.19
담당교수	:	권기현
학	과 :	정보통신공학과
학	번 :	201920479
이	름 :	김행복

-- 4-1) EMP 테이블 구성 살펴보기

DESC EMP;

이름	널?	유형
EMPNO	NOT NULL	NUMBER(4)
ENAME		VARCHAR2(10)
JOB		VARCHAR2(9)
MGR		NUMBER(4)
HIREDATE		DATE
SAL		NUMBER(7,2)
COMM		NUMBER(7,2)
DEPTNO		NUMBER(2)

-- 4-2) DEPT 테이블 구성 살펴보기

DESC DEPT;

이름	널?	유형
DEPTNO		NUMBER(2)
DNAME		VARCHAR2(14)
LOC		VARCHAR2(13)

-- 4-3) SALGRADE 테이블 구성 살펴보기

DESC SALGRADE;

이름	널?	유형
GRADE		NUMBER
LOSAL		NUMBER
HISAL		NUMBER

-- 4-4) EMP 테이블 전체 열 조회하기

SELECT * FROM EMP;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
7	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
8	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
9	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
10	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
11	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
12	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
13	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
14	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 4-5) 열을 쉼표로 구분하여 출력하기

SELECT EMPNO, ENAME, DEPTNO FROM EMP;

	EMPNO	ENAME	DEPTNO
1	7369	SMITH	20
2	7499	ALLEN	30
3	7521	WARD	30
4	7566	JONES	20
5	7654	MARTIN	30
6	7698	BLAKE	30
7	7782	CLARK	10
8	7788	SCOTT	20
9	7839	KING	10
10	7844	TURNER	30
11	7876	ADAMS	20
12	7900	JAMES	30
13	7902	FORD	20
14	7934	MILLER	10

-- 4-6) DISTINCT로 열의 중복 제거하기

```
SELECT DISTINCT DEPTNO FROM EMP;
```

	DEPTNO
1	20
2	30
3	10

-- 4-7) 여러 개 열을 명시하여 중복 제거하기

```
SELECT DISTINCT JOB, DEPTNO FROM EMP;
```

	JOB	DEPTNO
1	CLERK	20
2	SALESMAN	30
3	MANAGER	20
4	MANAGER	30
5	MANAGER	10
6	ANALYST	20
7	PRESIDENT	10
8	CLERK	30
9	CLERK	10

-- 4-8) 직책, 부서 번호 출력하기(ALL사용)

```
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

-- 4-9) 열에 연산식을 사용하여 출력하기

SELECT ENAME, SAL, SAL*12+COMM, COMM FROM EMP;

	ENAME	SAL	SAL*12+COMM	COMM
1	SMITH	800	(null)	(null)
2	ALLEN	1600	19500	300
3	WARD	1250	15500	500
4	JONES	2975	(null)	(null)
5	MARTIN	1250	16400	1400
6	BLAKE	2850	(null)	(null)
7	CLARK	2450	(null)	(null)
8	SCOTT	3000	(null)	(null)
9	KING	5000	(null)	(null)
10	TURNER	1500	18000	0
11	ADAMS	1100	(null)	(null)
12	JAMES	950	(null)	(null)
13	FORD	3000	(null)	(null)
14	MILLER	1300	(null)	(null)

-- 4-10) 곱하기를 사용하지 않고 사원의 연간 총 수입 출력하기

SELECT ENAME, SAL, SAL+SAL+SAL+SAL+SAL+SAL+SAL+SAL+SAL+SAL+SAL+SAL+COMM, COMM

FROM EMP;

	ENAME	SAL	SAL+SAL+SAL+SAL+SAL+SAL+SAL+SAL+SAL+SAL+SAL+SAL+COMM	COMM
1	SMITH	800	(null)	(null)
2	ALLEN	1600	19500	300
3	WARD	1250	15500	500
4	JONES	2975	(null)	(null)
5	MARTIN	1250	16400	1400
6	BLAKE	2850	(null)	(null)
7	CLARK	2450	(null)	(null)
8	SCOTT	3000	(null)	(null)
9	KING	5000	(null)	(null)
10	TURNER	1500	18000	0
11	ADAMS	1100	(null)	(null)
12	JAMES	950	(null)	(null)
13	FORD	3000	(null)	(null)
14	MILLER	1300	(null)	(null)

-- 4-11) 별칭을 사용하여 사원의 연간 총 수입 출력하기

SELECT ENAME, SAL, SAL*12+COMM AS ANNSAL, COMM FROM EMP;

	ENAME	SAL	ANNSAL	COMM
1	SMITH	800	(null)	(null)
2	ALLEN	1600	19500	300
3	WARD	1250	15500	500
4	JONES	2975	(null)	(null)
5	MARTIN	1250	16400	1400
6	BLAKE	2850	(null)	(null)
7	CLARK	2450	(null)	(null)
8	SCOTT	3000	(null)	(null)
9	KING	5000	(null)	(null)
10	TURNER	1500	18000	0
11	ADAMS	1100	(null)	(null)
12	JAMES	950	(null)	(null)
13	FORD	3000	(null)	(null)
14	MILLER	1300	(null)	(null)

-- 4-12) EMP 테이블의 모든 열을 급여 기준으로 오름차순 정렬하기

SELECT * FROM EMP ORDER BY SAL;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
3	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
4	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10
7	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
8	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
9	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
10	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
11	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
12	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
13	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
14	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10

-- 4-13) EMP 테이블의 모든 열을 급여 기준으로 내림차순 정렬하기

```
SELECT * FROM EMP ORDER BY SAL DESC;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
2	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
3	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
6	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
7	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
8	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
9	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10
10	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
11	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
12	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
13	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
14	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20

-- 4-14) EMP 테이블의 전체 열을 부서 번호(오름차순)와 급여(내림차순)로 정렬하기

```
SELECT * FROM EMP ORDER BY DEPTNO ASC, SAL DESC;
```

	EM...	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
2	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
3	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10
4	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
5	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
6	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
7	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
8	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
9	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
10	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
11	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
12	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
13	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
14	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30

-- 5-1) EMP 테이블의 모든 열 출력하기

SELECT * FROM EMP;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
7	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
8	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
9	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
10	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
11	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
12	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
13	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
14	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-2) 부서 번호가 30인 데이터만 출력하기

SELECT * FROM EMP WHERE DEPTNO=30;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
2	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
3	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
4	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
5	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
6	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30

-- 5-3) AND 연산자로 여러 개의 조건식 사용하기

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

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
2	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
3	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
4	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30

-- 5-4) OR 연산자로 여러 개의 출력 조건 사용하기

```
SELECT * FROM EMP WHERE DEPTNO=30 OR JOB='CLERK';
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
5	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
6	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
7	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
8	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
9	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-5) 곱셈 산술 연산자를 사용한 예

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

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
2	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20

-- 5-6) 대소 비교 연산자를 사용하여 출력하기

```
SELECT * FROM EMP WHERE SAL>=3000;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
2	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
3	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20

-- 5-7) 문자를 대소 비교 연산자로 비교하기(비교 문자열이 문자 하나일 때)

```
SELECT * FROM EMP WHERE ENAME >= 'F';
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
3	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
4	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
5	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
6	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
7	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
8	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
9	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
10	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-8) 문자열을 대소 비교 연산자로 비교하기(비교 문자열이 문자 여러 개일 때)

```
SELECT * FROM EMP WHERE ENAME <= 'FORZ';
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
2	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
3	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
4	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
5	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20

-- 5-9) 등가 비교 연산자(!=)를 사용하여 출력하기

```
SELECT * FROM EMP WHERE SAL != 3000;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
7	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
8	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
9	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
10	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
11	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
12	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-10) 등가 비교 연산자(<>)를 사용하여 출력하기

SELECT * FROM EMP WHERE SAL <> 3000;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
7	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
8	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
9	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
10	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
11	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
12	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-11) 등가 비교 연산자(^=)를 사용하여 출력하기

SELECT * FROM EMP WHERE SAL ^= 3000;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
7	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
8	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
9	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
10	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
11	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
12	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-12) NOT 연산자를 사용하여 출력하기

```
SELECT * FROM EMP WHERE NOT SAL = 3000;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
7	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
8	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
9	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
10	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
11	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
12	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-13) OR 연산자를 사용하여 여러 개 조건을 만족하는 데이터를 출력하기

```
SELECT * FROM EMP WHERE JOB = 'MANAGER' OR JOB = 'SALESMAN' OR JOB = 'CLERK';
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
7	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
8	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
9	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
10	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
11	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-14) IN 연산자를 사용하여 출력하기

```
SELECT * FROM EMP WHERE JOB IN('MANAGER', 'SALESMAN', 'CLERK');
```

	EM...	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
7	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
8	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
9	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
10	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
11	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-15) 등가 비교 연산자와 AND 연산자를 사용하여 출력하기

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

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
2	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
3	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20

-- 5-16) IN 연산자와 논리 부정 연산자를 사용하여 출력하기

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

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
2	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
3	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20

-- 5-17) 대소 비교 연산자와 AND 연산자를 사용하여 출력하기

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

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
2	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
3	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
4	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
5	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20

-- 5-18) BETWEEN A AND B 연산자를 사용하여 출력하기

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

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
2	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
3	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
4	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
5	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20

-- 5-19) BETWEEN A AND B 연산자와 NOT 연산자를 사용하여 출력하기

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

	EM...	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
5	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
6	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
7	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
8	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
9	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-20) LIKE 연산자 사용하여 출력하기

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

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20

-- 5-21) 사원 이름의 두 번째 글자가 L인 사원만 출력하기

```
SELECT * FROM EMP WHERE ENAME LIKE '_L%';
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
2	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
3	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10

-- 5-22) 사원 이름에 AM이 포함되어 있는 사원 데이터만 출력하기

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

	EM...	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
2	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30

-- 5-23) 사원 이름에 AM이 포함되어 있지 않은 사원 데이터 출력하기

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

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
7	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
8	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
9	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
10	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
11	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
12	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-24) 별칭을 사용하여 열 이름 출력하기

SELECT ENAME, SAL, SAL*12+COMM AS ANNSAL, COMM FROM EMP;

	ENAME	SAL	ANNSAL	COMM
1	SMITH	800	(null)	(null)
2	ALLEN	1600	19500	300
3	WARD	1250	15500	500
4	JONES	2975	(null)	(null)
5	MARTIN	1250	16400	1400
6	BLAKE	2850	(null)	(null)
7	CLARK	2450	(null)	(null)
8	SCOTT	3000	(null)	(null)
9	KING	5000	(null)	(null)
10	TURNER	1500	18000	0
11	ADAMS	1100	(null)	(null)
12	JAMES	950	(null)	(null)
13	FORD	3000	(null)	(null)
14	MILLER	1300	(null)	(null)

-- 5-25) 등가 비교 연산자로 NULL 비교하기

SELECT * FROM EMP WHERE COMM = NULL;

EMPNO	ENAME	JOB	MGR	HIRED...	SAL	COMM	DEPTNO

-- 5-26) IS NULL 연산자를 사용하여 출력하기

SELECT * FROM EMP WHERE COMM IS NULL;

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
3	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
4	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
5	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
6	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
7	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
8	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
9	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
10	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-27) 직속 상관이 있는 사원 데이터만 출력하기

```
SELECT * FROM EMP WHERE MGR IS NOT NULL;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
3	7521	WARD	SALESMAN	7698	1981/02/22	1250	500	30
4	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
5	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
6	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
7	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
8	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
9	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
10	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
11	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
12	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
13	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-28) AND 연산자와 IS NULL 연산자 사용하기

```
SELECT * FROM EMP WHERE SAL > NULL AND COMM IS NULL;
```

EMP...	ENAME	JOB	MGR	HIRED...	SAL	COMM	DEPTNO

-- 5-29) OR 연산자와 IS NULL 연산자 사용하기

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

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
3	7698	BLAKE	MANAGER	7839	1981/05/01	2850	(null)	30
4	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
5	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
6	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
7	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
8	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
9	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
10	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 5-30) 집합 연산자(UNION)를 사용하여 출력하기

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

UNION

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

	EMPNO	ENAME	SAL	DEPTNO
1	7782	CLARK	2450	10
2	7839	KING	5000	10
3	7934	MILLER	1300	10
4	7369	SMITH	800	20
5	7566	JONES	2975	20
6	7788	SCOTT	3000	20
7	7876	ADAMS	1100	20
8	7902	FORD	3000	20

-- 5-31) 집합 연산자(UNION)를 사용하여 출력하기(출력 열 개수가 다를 때)

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

UNION

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

```
ORA-01789: 질의 블록은 부정확한 수의 결과 열을 가지고 있습니다.
01789. 00000 -  "query block has incorrect number of result columns"
*Cause:
*Action:
```

-- 5-32) 집합 연산자(UNION)를 사용하여 출력하기(출력 열의 자료형이 다를 때)

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

UNION

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

```
ORA-01790: 대응하는 식과 같은 데이터 유형이어야 합니다
01790. 00000 -  "expression must have same datatype as corresponding expression"
*Cause:
*Action:
141행, 8열에서 오류 발생
```

-- 5-33) 집합 연산자(UNION)를 사용하여 출력하기(출력 열 개수와 자료형이 같을 때)

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

UNION

SELECT SAL, JOB, DEPTNO, SAL FROM EMP WHERE DEPTNO = 20;
```

	EMPNO	ENAME	SAL	DEPTNO
1	7782	CLARK	2450	10
2	7839	KING	5000	10
3	7934	MILLER	1300	10
4	800	CLERK	20	800
5	2975	MANAGER	20	2975
6	3000	ANALYST	20	3000
7	1100	CLERK	20	1100

-- 5-34) 집합 연산자(UNION)를 사용하여 출력하기(출력 결과 데이터가 같을 때)

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

UNION

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

	EMPNO	ENAME	SAL	DEPTNO
1	7782	CLARK	2450	10
2	7839	KING	5000	10
3	7934	MILLER	1300	10

-- 5-35) 집합 연산자(UNION ALL)를 사용하여 출력하기(출력 결과 데이터가 같을 때)

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

UNION ALL

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

	EMPNO	EN...	SAL	DEPTNO
1	7782	CLARK	2450	10
2	7839	KING	5000	10
3	7934	MILLER	1300	10
4	7782	CLARK	2450	10
5	7839	KING	5000	10
6	7934	MILLER	1300	10

-- 5-36) 집합 연산자(MINUS)를 사용하여 출력하기

```
SELECT EMPNO, ENAME, SAL, DEPTNO FROM EMP

MINUS

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

	EMPNO	ENAME	SAL	DEPTNO
1	7369	SMITH	800	20
2	7499	ALLEN	1600	30
3	7521	WARD	1250	30
4	7566	JONES	2975	20
5	7654	MARTIN	1250	30
6	7698	BLAKE	2850	30
7	7788	SCOTT	3000	20
8	7844	TURNER	1500	30
9	7876	ADAMS	1100	20
10	7900	JAMES	950	30
11	7902	FORD	3000	20

-- 5-37) 집합 연산자(INTERSECT)를 사용하여 출력하기

```
SELECT EMPNO, ENAME, SAL, DEPTNO FROM EMP

INTERSECT

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

	EMPNO	ENAME	SAL	DEPTNO
1	7782	CLARK	2450	10
2	7839	KING	5000	10
3	7934	MILLER	1300	10

-- 6-1) UPPER, LOWER, INITCAP 함수 사용하기

SELECT ENAME, UPPER(ENAME), LOWER(ENAME), INITCAP(ENAME) FROM EMP;

	ENAME	UPPER(ENAME)	LOWER(ENAME)	INITCAP(ENAME)
1	SMITH	SMITH	smith	Smith
2	ALLEN	ALLEN	allen	Allen
3	WARD	WARD	ward	Ward
4	JONES	JONES	jones	Jones
5	MARTIN	MARTIN	martin	Martin
6	BLAKE	BLAKE	blake	Blake
7	CLARK	CLARK	clark	Clark
8	SCOTT	SCOTT	scott	Scott
9	KING	KING	king	King
10	TURNER	TURNER	turner	Turner
11	ADAMS	ADAMS	adams	Adams
12	JAMES	JAMES	james	James
13	FORD	FORD	ford	Ford
14	MILLER	MILLER	miller	Miller

-- 6-2) UPPER 함수로 문자열 비교하기(사원 이름이 SCOTT인 데이터 찾기)

SELECT * FROM EMP WHERE UPPER(ENAME) = UPPER('scott');

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20

-- 6-3) UPPER 함수로 문자열 비교하기(사원 이름이 SCOTT 단어를 포함 데이터 찾기)

SELECT * FROM EMP WHERE UPPER(ENAME) LIKE UPPER('%scott%');

	EM...	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20

-- 6-4) 선택한 열의 문자열 길이 구하기

```
SELECT ENAME, LENGTH(ENAME) FROM EMP;
```

	ENAME	LENGTH(ENAME)
1	SMITH	5
2	ALLEN	5
3	WARD	4
4	JONES	5
5	MARTIN	6
6	BLAKE	5
7	CLARK	5
8	SCOTT	5
9	KING	4
10	TURNER	6
11	ADAMS	5
12	JAMES	5
13	FORD	4
14	MILLER	6

-- 6-5) 사원 이름의 길이가 5 이상인 행 출력하기

```
SELECT ENAME, LENGTH(ENAME) FROM EMP WHERE LENGTH(ENAME) >= 5;
```

	ENAME	LENGTH(ENAME)
1	SMITH	5
2	ALLEN	5
3	JONES	5
4	MARTIN	6
5	BLAKE	5
6	CLARK	5
7	SCOTT	5
8	TURNER	6
9	ADAMS	5
10	JAMES	5
11	MILLER	6

-- 6-6) LENGTH 함수와 LENGTHB 함수 비교하기

SELECT LENGTH('한글'), LENGTHB('한글') FROM DUAL;

	LENGTH('한글')	LENGTHB('한글')
1	2	6

-- 6-7) SUBSTR 함수를 사용하는 예

SELECT JOB, SUBSTR(JOB, 1, 2), SUBSTR(JOB, 3, 2), SUBSTR(JOB, 5) FROM EMP;

	JOB	SUBSTR(JOB,1,2)	SUBSTR(JOB,3,2)	SUBSTR(JOB,5)
1	CLERK	CL	ER	K
2	SALESMAN	SA	LE	SMAN
3	SALESMAN	SA	LE	SMAN
4	MANAGER	MA	NA	GER
5	SALESMAN	SA	LE	SMAN
6	MANAGER	MA	NA	GER
7	MANAGER	MA	NA	GER
8	ANALYST	AN	AL	YST
9	PRESIDENT	PR	ES	IDENT
10	SALESMAN	SA	LE	SMAN
11	CLERK	CL	ER	K
12	CLERK	CL	ER	K
13	ANALYST	AN	AL	YST
14	CLERK	CL	ER	K

-- 6-8) SUBSTR 함수 안에 다른 함수(LENGTH) 함께 사용하기

SELECT JOB, SUBSTR(JOB, -LENGTH(JOB)), SUBSTR(JOB, -LENGTH(JOB), 2), SUBSTR(JOB, -3) FROM EMP;

	JOB	SUBSTR(JOB,-LENGTH(JOB))	SUBSTR(JOB,-LENGTH(JOB),2)	SUBSTR(JOB,-3)
1	CLERK	CLERK	CL	ERK
2	SALESMAN	SALESMAN	SA	MAN
3	SALESMAN	SALESMAN	SA	MAN
4	MANAGER	MANAGER	MA	GER
5	SALESMAN	SALESMAN	SA	MAN
6	MANAGER	MANAGER	MA	GER
7	MANAGER	MANAGER	MA	GER
8	ANALYST	ANALYST	AN	YST
9	PRESIDENT	PRESIDENT	PR	ENT
10	SALESMAN	SALESMAN	SA	MAN
11	CLERK	CLERK	CL	ERK
12	CLERK	CLERK	CL	ERK
13	ANALYST	ANALYST	AN	YST
14	CLERK	CLERK	CL	ERK

-- 6-9) INSTR 함수로 문자열 데이터에서 특정 문자열 찾기

```
SELECT INSTR('HELLO, ORACLE!', 'L') AS INSTR_1,
       INSTR('HELLO, ORACLE!', 'L', 5) AS INSTR_2,
       INSTR('HELLO, ORACLE!', 'L', 2, 2) AS INSTR_3
FROM DUAL;
```

	INSTR_1	INSTR_2	INSTR_3
1	3	12	4

-- 6-10) INSTR 함수로 사원 이름에 문자 S가 있는 행 구하기

```
SELECT * FROM EMP WHERE INSTR(ENAME, 'S') > 0;
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
3	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
4	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
5	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30

-- 6-11) LIKE 연산자로 사원 이름에 문자 S가 있는 행 구하기

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

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7369	SMITH	CLERK	7902	1980/12/17	800	(null)	20
2	7566	JONES	MANAGER	7839	1981/04/02	2975	(null)	20
3	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
4	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
5	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30

-- 6-12) REPLACE 함수로 문자열 안에 있는 특정 문자 바꾸기

```
SELECT '010-1234-5678' AS REPLACE_BEFORE, REPLACE('010-1234-5678', '-', ' ') AS REPLACE_1,
REPLACE('010-1234-5678', '-') AS REPLACE_2 FROM DUAL;
```

	REPLACE_BEFORE	REPLACE_1	REPLACE_2
1	010-1234-5678	010 1234 5678	01012345678

-- 6-13) LPAD, RPAD 함수 사용하여 출력하기

```
SELECT 'Oracle', LPAD('Oracle', 10, '#') AS LPAD_1,
                 RPAD('Oracle', 10, '*') AS RPAD_1,
                 LPAD('Oracle', 10) AS LPAD_2,
                 RPAD('Oracle', 10) AS RPAD_2
FROM DUAL;
```

	'ORACLE'	LPAD_1	RPAD_1	LPAD_2	RPAD_2
1	Oracle	####Oracle	Oracle****	Oracle	Oracle

-- 6-14) RPAD 함수를 사용하여 개인정보 뒷자리 * 표시로 출력하기

```
SELECT RPAD('971225-', 14, '*') AS RPAD_JMNO,
       RPAD('010-1234-',13, '*') AS RPAD_PHONE
FROM DUAL;
```

	RPAD_JMNO	RPAD_PHONE
1	971225-*******	010-1234-****

-- 6-15) 두 열 사이에 콜론(:) 넣고 연결하기

```
SELECT CONCAT(EMPNO, ENAME), CONCAT(EMPNO, CONCAT(' : ', ENAME)) FROM EMP WHERE
ENAME = 'SCOTT';
```

	CONCAT(EMPNO,ENAME)	CONCAT(EMPNO,CONCAT(':',ENAME))
1	7788SCOTT	7788 : SCOTT

-- 6-16) TRIM 함수로 공백 제거하여 출력하기

```
SELECT '[' || TRIM(' _ _Oracle_ _ ') || ']' AS TRIM, '[' || TRIM(LEADING FROM ' _ _Oracle_ _ ') || ']' AS
TRIM_LEADING, '[' || TRIM(TRAILING FROM ' _ _Oracle_ _ ') || ']' AS TRIM_TRAILING,
'[' || TRIM(BOTH FROM ' _ _Oracle_ _ ') || ']' AS TRIM_BOTH FROM DUAL;
```

	TRIM	TRIM_LEADING	TRIM_TRAILING	TRIM_BOTH
1	[_ _Oracle_ _]	[_ _Oracle_ _]	[_ _Oracle_ _]	[_ _Oracle_ _]

-- 6-17) TRIM 함수로 삭제할 문자 _ 삭제 후 출력하기

```
SELECT '[' || TRIM('_' FROM '_ _Oracle_ _') || ']' AS TRIM,
       '[' || TRIM(LEADING '_' FROM '_ _Oracle_ _') || ']' AS TRIM_LEADING,
       '[' || TRIM(TRAILING '_' FROM '_ _Oracle_ _') || ']' AS TRIM_TRAILING,
       '[' || TRIM(BOTH '_' FROM '_ _Oracle_ _') || ']' AS TRIM_BOTH
FROM DUAL;
```

	TRIM	TRIM_LEADING	TRIM_TRAILING	TRIM_BOTH
1	[_Oracle_]	[_Oracle_ _]	[_ _Oracle_]	[_Oracle_]

-- 6-18) TRIM, LTRIM, RTRIM 사용하여 문자열 출력하기

```
SELECT '[' || TRIM(' _Oracle_ ') || ']' AS TRIM,
       '[' || LTRIM(' _Oracle_ ') || ']' AS LTRIM,
       '[' || LTRIM('<_Oracle_>', '_<') || ']' AS LTRIM_2,
       '[' || RTRIM(' _Oracle_ ') || ']' AS RTRIM,
       '[' || RTRIM('<_Oracle_>', '>_') || ']' AS RTRIM_2
FROM DUAL;
```

	TRIM	LTRIM	LTRIM_2	RTRIM	RTRIM_2
1	[_Oracle_]	[_Oracle_]	[Oracle_>]	[_Oracle_]	[<_Oracle]

-- 6-19) ROUND 함수를 사용하여 반올림된 숫자 출력하기

```
SELECT ROUND(1234.5678) AS ROUND, ROUND(1234.5678, 0) AS ROUND_0,
       ROUND(1234.5678, 1) AS ROUND_1,
       ROUND(1234.5678, 2) AS ROUND_2,
       ROUND(1234.5678, -1) AS ROUND_MINUS1,
       ROUND(1234.5678, -2) AS ROUND_MINUS2 FROM DUAL;
```

	ROUND	ROUND_0	ROUND_1	ROUND_2	ROUND_MINUS1	ROUND_MINUS2
1	1235	1235	1234.6	1234.57	1230	1200

-- 6-20) ROUND 함수를 사용하여 반올림된 숫자 출력하기

```
SELECT TRUNC(1234.5678) AS TRUNC,
        TRUNC(1234.5678, 0) AS TRUNC_0,
        TRUNC(1234.5678, 1) AS TRUNC_1,
        TRUNC(1234.5678, 2) AS TRUNC_2,
        TRUNC(1234.5678, -1) AS TRUNC_MINUS1,
        TRUNC(1234.5678, -2) AS TRUNC_MINUS2
FROM DUAL;
```

	TRUNC	TRUNC_0	TRUNC_1	TRUNC_2	TRUNC_MINUS1	TRUNC_MINUS2
1	1234	1234	1234.5	1234.56	1230	1200

-- 6-21) CEIL, FLOOR 함수로 숫자 출력하기

```
SELECT CEIL(3.14), FLOOR(3.14), CEIL(-3.14), FLOOR(-3.14) FROM DUAL;
```

	CEIL(3.14)	FLOOR(3.14)	CEIL(-3.14)	FLOOR(-3.14)
1	4	3	-3	-4

-- 6-22) MOD 함수를 사용하여 나머지 값 출력하기

```
SELECT MOD(15, 6), MOD(10, 2), MOD(11, 2) FROM DUAL;
```

	MOD(15,6)	MOD(10,2)	MOD(11,2)
1	3	0	1

-- 6-23) SYSDATE 함수를 사용하여 날짜 출력하기

```
SELECT SYSDATE AS NOW, SYSDATE-1 AS YESTERDAY, SYSDATE+1 AS TOMORROW FROM DUAL;
```

	NOW	YESTERDAY	TOMORROW
1	2024/03/19	2024/03/18	2024/03/20

-- 6-24) SYSDATE와 ADD_MONTHS 함수로 3개월 후 날짜 구하기

```
SELECT SYSDATE, ADD_MONTHS(SYSDATE, 3) FROM DUAL;
```

	SYSDATE	ADD_MONTHS(SYSDATE,3)
1	2024/03/19	2024/06/19

-- 6-25) 입사 10주년이 되는 사원들 데이터 출력하

SELECT EMPNO, ENAME, HIREDATE, ADD_MONTHS(HIREDATE, 120) AS WORK10YEAR FROM EMP;

	EMPNO	ENAME	HIREDATE	WORK10YEAR
1	7369	SMITH	1980/12/17	1990/12/17
2	7499	ALLEN	1981/02/20	1991/02/20
3	7521	WARD	1981/02/22	1991/02/22
4	7566	JONES	1981/04/02	1991/04/02
5	7654	MARTIN	1981/09/28	1991/09/28
6	7698	BLAKE	1981/05/01	1991/05/01
7	7782	CLARK	1981/06/09	1991/06/09
8	7788	SCOTT	1982/12/09	1992/12/09
9	7839	KING	1981/11/17	1991/11/17
10	7844	TURNER	1981/09/08	1991/09/08
11	7876	ADAMS	1983/01/12	1993/01/12
12	7900	JAMES	1981/12/03	1991/12/03
13	7902	FORD	1981/12/03	1991/12/03
14	7934	MILLER	1982/01/23	1992/01/23

-- 6-26) 입사 42.5년 미만인 사원 데이터 출력하기

SELECT EMPNO, ENAME, HIREDATE, SYSDATE FROM EMP WHERE ADD_MONTHS(HIREDATE, 510)

> SYSDATE;

	EMPNO	ENAME	HIREDATE	SYSDATE
1	7654	MARTIN	1981/09/28	2024/03/19
2	7788	SCOTT	1982/12/09	2024/03/19
3	7839	KING	1981/11/17	2024/03/19
4	7876	ADAMS	1983/01/12	2024/03/19
5	7900	JAMES	1981/12/03	2024/03/19
6	7902	FORD	1981/12/03	2024/03/19
7	7934	MILLER	1982/01/23	2024/03/19

-- 6-27) HIREDATE와 SYSDATE 사이의 개월 수를 MONTHS_BETWEEN 함수로 출력하기

```
SELECT EMPNO, ENAME, HIREDATE, SYSDATE,
        MONTHS_BETWEEN(HIREDATE, SYSDATE) AS MONTH1,
        MONTHS_BETWEEN(SYSDATE, HIREDATE) AS MONTH2,
        TRUNC(MONTHS_BETWEEN(SYSDATE, HIREDATE)) AS MONTH3
FROM EMP;
```

	EMPNO	ENAME	HIREDATE	SYSDATE	MONTH1	MONTH2	MON...
1	7369	SMITH	1980/12/17	2024/03/19	-519.0840494...	519.084049432...	519
2	7499	ALLEN	1981/02/20	2024/03/19	-516.9872752...	516.987275238...	516
3	7521	WARD	1981/02/22	2024/03/19	-516.9227591...	516.922759109...	516
4	7566	JONES	1981/04/02	2024/03/19	-515.5679204...	515.567920400...	515
5	7654	MARTIN	1981/09/28	2024/03/19	-509.7292107...	509.729210722...	509
6	7698	BLAKE	1981/05/01	2024/03/19	-514.6001784...	514.600178464...	514
7	7782	CLARK	1981/06/09	2024/03/19	-513.3421139...	513.342113948...	513
8	7788	SCOTT	1982/12/09	2024/03/19	-495.3421139...	495.342113948...	495
9	7839	KING	1981/11/17	2024/03/19	-508.0840494...	508.084049432...	508
10	7844	TURNER	1981/09/08	2024/03/19	-510.3743720...	510.374372013...	510
11	7876	ADAMS	1983/01/12	2024/03/19	-494.2453397...	494.245339755...	494
12	7900	JAMES	1981/12/03	2024/03/19	-507.5356623...	507.535662335...	507
13	7902	FORD	1981/12/03	2024/03/19	-507.5356623...	507.535662335...	507
14	7934	MILLER	1982/01/23	2024/03/19	-505.8905010...	505.890501045...	505

-- 6-28) NEXT_DAY, LAST_DAY 함수를 사용하여 출력하기

```
SELECT SYSDATE, NEXT_DAY(SYSDATE, 'MONDAY'), LAST_DAY(SYSDATE) FROM DUAL;
```

	SYSDATE	NEXT_DAY(SYSDATE,'월요일')	LAST_DAY(SYSDATE)
1	2024/03/19	2024/03/25	2024/03/31

-- 6-29) ROUND 함수 사용하여 날짜 데이터 출력하기

```
SELECT SYSDATE,
        ROUND(SYSDATE, 'CC') AS FORMAT_CC,
        ROUND(SYSDATE, 'YYYY') AS FORMAT_YYYY,
        ROUND(SYSDATE, 'Q') AS FORMAT_Q,
        ROUND(SYSDATE, 'DDD') AS FORMAT_DDD,
        ROUND(SYSDATE, 'HH') AS FORMAT_HH
```

FROM DUAL;

	SYSD...	FORMAT_CC	FORMAT_YYYY	FORMAT_Q	FORMAT_DDD	FORMAT_HH
1	2024/03/19	2001/01/01	2024/01/01	2024/04/01	2024/03/20	2024/03/19

-- 6-30) TRUNC 함수 사용하여 날짜 데이터 출력하기

```
SELECT SYSDATE,
        TRUNC(SYSDATE, 'CC') AS TRUNC_CC,
        TRUNC(SYSDATE, 'YYYY') AS TRUNC_YYYY,
        TRUNC(SYSDATE, 'Q') AS TRUNC_Q,
        TRUNC(SYSDATE, 'DDD') AS TRUNC_DDD,
        TRUNC(SYSDATE, 'HH') AS TRUNC_HH
FROM DUAL;
```

	SYSDATE	TRUNC_CC	TRUNC_YYYY	TRUNC_Q	TRUNC_DDD	TRUNC_HH
1	2024/03/19	2001/01/01	2024/01/01	2024/01/01	2024/03/19	2024/03/19

-- 6-31) 숫자와 문자열(숫자)을 더하여 출력하기

```
SELECT EMPNO, ENAME, EMPNO + '500' FROM EMP WHERE ENAME = 'SCOTT';
```

	EMPNO	ENAME	EMPNO+'500'
1	7788	SCOTT	8288

-- 6-32) 문자열(문자)과 숫자를 더하여 출력하기

```
SELECT 'ABCD' + EMPNO, EMPNO FROM EMP WHERE ENAME = 'SCOTT';
```

```
ORA-01722: 수치가 부적합합니다
01722. 00000 -  "invalid number"
*Cause:    The specified number was invalid.
*Action:   Specify a valid number.
```

-- 6-33) SYSDATE 날짜 형식 지정하여 출력하기

```
SELECT TO_CHAR(SYSDATE, 'YYYY/MM/DD HH24:MI:SS') AS 현재날짜시간 FROM DUAL;
```

	현재날짜시간
1	2024/03/19 14:33:30

-- 6-34) 월과 요일을 다양한 형식으로 출력하기

```
SELECT SYSDATE,
        TO_CHAR(SYSDATE, 'MM') AS MM,
        TO_CHAR(SYSDATE, 'MON') AS MON,
        TO_CHAR(SYSDATE, 'MONTH') AS MONTH,
        TO_CHAR(SYSDATE, 'DD') AS DD,
        TO_CHAR(SYSDATE, 'DY') AS DY,
        TO_CHAR(SYSDATE, 'DAY') AS DAY
FROM DUAL;
```

	SYSDATE	MM	MON	MONTH	DD	DY	DAY
1	2024/03/19	03	3월	3월	19	화	화요일

-- 6-35) 여러 언어로 날짜(월) 출력하기

```
SELECT SYSDATE,
        TO_CHAR(SYSDATE, 'MM') AS MM,
        TO_CHAR(SYSDATE, 'MON', 'NLS_DATE_LANGUAGE = KOREAN') AS MON_KOR,
        TO_CHAR(SYSDATE, 'MON', 'NLS_DATE_LANGUAGE = JAPANESE') AS MON_JPN,
        TO_CHAR(SYSDATE, 'MON', 'NLS_DATE_LANGUAGE = ENGLISH') AS MON_ENG,
        TO_CHAR(SYSDATE, 'MONTH', 'NLS_DATE_LANGUAGE = KOREAN') AS MONTH_KOR,
        TO_CHAR(SYSDATE, 'MONTH', 'NLS_DATE_LANGUAGE = JAPANESE') AS MONTH_JPN,
        TO_CHAR(SYSDATE, 'MONTH', 'NLS_DATE_LANGUAGE = ENGLISH') AS MONTH_ENG
FROM DUAL;
```

	YSDATE	MM	MON_KOR	MON_JPN	MON_ENG	MONTH_KOR	MONTH_JPN	MONTH_ENG
1	4/03/19	03	3월	3月	MAR	3월	3月	MARCH

-- 6-36) 여러 언어로 날짜(요일) 출력하기

```
SELECT SYSDATE,
        TO_CHAR(SYSDATE, 'MM'),
        TO_CHAR(SYSDATE, 'DD'),
        TO_CHAR(SYSDATE, 'DY', 'NLS_DATE_LANGUAGE = KOREAN') AS DY_KOR,
        TO_CHAR(SYSDATE, 'DY', 'NLS_DATE_LANGUAGE = JAPANESE') AS DY_JPN,
        TO_CHAR(SYSDATE, 'DY', 'NLS_DATE_LANGUAGE = ENGLISH') AS DY_ENG,
        TO_CHAR(SYSDATE, 'DAY', 'NLS_DATE_LANGUAGE = KOREAN') AS DAY_KOR,
        TO_CHAR(SYSDATE, 'DAY', 'NLS_DATE_LANGUAGE = JAPANESE') AS DAY_JPN,
        TO_CHAR(SYSDATE, 'DAY', 'NLS_DATE_LANGUAGE = ENGLISH') AS DAY_ENG
FROM DUAL;
```

	SYSDATE	TO_CHAR(SYSDATE,'MM')	TO_CHAR(SYSDATE,'DD')	DY_KOR	DY_JPN	DY_ENG	DAY_KOR	DAY_JPN	DAY_ENG
1	2024/03/19	03	19	화	火	TUE	화요일	火曜日	TUESDAY

-- 6-37) SYSDATE 시간 형식 지정하여 출력하기

```
SELECT SYSDATE,
        TO_CHAR(SYSDATE, 'HH24:MI:SS') AS HH24MISS,
        TO_CHAR(SYSDATE, 'HH12:MI:SS AM') AS HHMISS_AM,
        TO_CHAR(SYSDATE, 'HH:MI:SS P.M.') AS HHMISS_PM
FROM DUAL;
```

	SYSDATE	HH24MISS	HHMISS_AM	HHMISS_PM
1	2024/03/19	14:34:09	02:34:09 오후	02:34:09 오후

-- 6-38) 여러 가지 숫자 형식을 사용하여 급여 출력하기

```
SELECT SAL,
        TO_CHAR(SAL, '$999,999') AS SAL_$,
        TO_CHAR(SAL, 'L999,999') AS SAL_L,
```

```
       TO_CHAR(SAL, '999,999.00') AS SAL_1,
       TO_CHAR(SAL, '000,999,999.00') AS SAL_2,
       TO_CHAR(SAL, '000999999.99') AS SAL_3,
       TO_CHAR(SAL, '999,999,00') AS SAL_4
FROM EMP;
```

	SAL	SAL_$	SAL_L	SAL_1	SAL_2	SAL_3	SAL_4
1	800	$800	₩800	800.00	000,000,800.00	000000800.00	8,00
2	1600	$1,600	₩1,600	1,600.00	000,001,600.00	000001600.00	16,00
3	1250	$1,250	₩1,250	1,250.00	000,001,250.00	000001250.00	12,50
4	2975	$2,975	₩2,975	2,975.00	000,002,975.00	000002975.00	29,75
5	1250	$1,250	₩1,250	1,250.00	000,001,250.00	000001250.00	12,50
6	2850	$2,850	₩2,850	2,850.00	000,002,850.00	000002850.00	28,50
7	2450	$2,450	₩2,450	2,450.00	000,002,450.00	000002450.00	24,50
8	3000	$3,000	₩3,000	3,000.00	000,003,000.00	000003000.00	30,00
9	5000	$5,000	₩5,000	5,000.00	000,005,000.00	000005000.00	50,00
10	1500	$1,500	₩1,500	1,500.00	000,001,500.00	000001500.00	15,00
11	1100	$1,100	₩1,100	1,100.00	000,001,100.00	000001100.00	11,00
12	950	$950	₩950	950.00	000,000,950.00	000000950.00	9,50
13	3000	$3,000	₩3,000	3,000.00	000,003,000.00	000003000.00	30,00
14	1300	$1,300	₩1,300	1,300.00	000,001,300.00	000001300.00	13,00

-- 6-39) 문자 데이터와 숫자 데이터를 연산하여 출력하기

```
SELECT 1300 - '1500', '1300' + 1500 FROM DUAL;
```

	1300-'1500'	'1300'+1500
1	-200	2800

-- 6-40) 문자 데이터끼리 연산하여 출력하기

```
SELECT '1,300' - '1,500' FROM DUAL;
```

```
ORA-01722: 수치가 부적합합니다
01722. 00000 -  "invalid number"
*Cause:    The specified number was invalid.
*Action:   Specify a valid number.
```

-- 6-41) TO_NUMBER 함수로 연산하여 출력하기

```
SELECT TO_NUMBER('1,300', '999,999') - TO_NUMBER('1,500', '999,999') FROM DUAL;
```

	TO_NUMBER('1,300','999,999')-TO_NUMBER('1,500','999,999')
1	-200

-- 6-42) TO_DATE 함수로 문자 데이터를 날짜 데이터 변환하기

```
SELECT TO_DATE('2018-07-14', 'YYYY-MM-DD') AS TODATE1,
       TO_DATE('20180714', 'YYYY_MM_DD') AS TODATE2
FROM DUAL;
```

	TODATE1	TODATE2
1	2018/07/14	2018/07/14

-- 6-43) 1981년 6월 1일 이후에 입사한 사원 정보 출력하기

```
SELECT * FROM EMP WHERE HIREDATE > TO_DATE('1981/06/01', 'YYYY/MM/DD');
```

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7654	MARTIN	SALESMAN	7698	1981/09/28	1250	1400	30
2	7782	CLARK	MANAGER	7839	1981/06/09	2450	(null)	10
3	7788	SCOTT	ANALYST	7566	1982/12/09	3000	(null)	20
4	7839	KING	PRESIDENT	(null)	1981/11/17	5000	(null)	10
5	7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
6	7876	ADAMS	CLERK	7788	1983/01/12	1100	(null)	20
7	7900	JAMES	CLERK	7698	1981/12/03	950	(null)	30
8	7902	FORD	ANALYST	7566	1981/12/03	3000	(null)	20
9	7934	MILLER	CLERK	7782	1982/01/23	1300	(null)	10

-- 6-44) 여러 각지 형식으로 날짜 데이터 출력하기

```
SELECT TO_DATE('49/12/10', 'YY/MM/DD') AS YY_YEAR_49,
       TO_DATE('49/12/10', 'RR/MM/DD') AS RR_YEAR_49,
       TO_DATE('50/12/10', 'YY/MM/DD') AS YY_YEAR_50,
       TO_DATE('50/12/10', 'RR/MM/DD') AS RR_YEAR_50,
       TO_DATE('51/12/10', 'YY/MM/DD') AS YY_YEAR_51,
       TO_DATE('51/12/10', 'RR/MM/DD') AS RR_YEAR_51
```

FROM DUAL;

	YY_YEAR_49	RR_YEAR_49	YY_YEAR_50	RR_YEAR_50	YY_YEAR_51	RR_YEAR_51
1	2049/12/10	2049/12/10	2050/12/10	1950/12/10	2051/12/10	1951/12/10

-- 6-45) NVL 함수를 사용하여 출력하기

```
SELECT EMPNO, ENAME, SAL, COMM, SAL+COMM,
        NVL(COMM, 0),
        SAL+NVL(COMM,0)
FROM EMP;
```

	EMPNO	ENAME	...	COMM	SAL+COMM	NVL(COMM,0)	SAL+NVL(COMM,0)
1	7369	SMITH	800	(null)	(null)	0	800
2	7499	ALLEN	1600	300	1900	300	1900
3	7521	WARD	1250	500	1750	500	1750
4	7566	JONES	2975	(null)	(null)	0	2975
5	7654	MARTIN	1250	1400	2650	1400	2650
6	7698	BLAKE	2850	(null)	(null)	0	2850
7	7782	CLARK	2450	(null)	(null)	0	2450
8	7788	SCOTT	3000	(null)	(null)	0	3000
9	7839	KING	5000	(null)	(null)	0	5000
10	7844	TURNER	1500	0	1500	0	1500
11	7876	ADAMS	1100	(null)	(null)	0	1100
12	7900	JAMES	950	(null)	(null)	0	950
13	7902	FORD	3000	(null)	(null)	0	3000
14	7934	MILLER	1300	(null)	(null)	0	1300

-- 6-46) NVL2 함수를 사용하여 출력하기

```
SELECT EMPNO, ENAME, COMM,
        NVL2(COMM, 'O', 'X'),
        NVL2(COMM, SAL*12+COMM, SAL*12) AS ANNSAL
FROM EMP;
```

	EMPNO	ENAME	COMM	NVL2(COMM,'O','X')	ANNSAL
1	7369	SMITH	(null)	X	9600
2	7499	ALLEN	300	O	19500
3	7521	WARD	500	O	15500
4	7566	JONES	(null)	X	35700
5	7654	MARTIN	1400	O	16400
6	7698	BLAKE	(null)	X	34200
7	7782	CLARK	(null)	X	29400
8	7788	SCOTT	(null)	X	36000
9	7839	KING	(null)	X	60000
10	7844	TURNER	0	O	18000
11	7876	ADAMS	(null)	X	13200
12	7900	JAMES	(null)	X	11400
13	7902	FORD	(null)	X	36000
14	7934	MILLER	(null)	X	15600

```
-- 6-47) DECODE 함수를 사용하여 출력하기

SELECT EMPNO, ENAME, JOB, SAL,

        DECODE(JOB,

                'MANAGER', SAL*1.1,

                'SALESMAN', SAL*1.05,

                'ANALYST', SAL,

                SAL*1.03) AS UPSAL

FROM EMP;
```

	EMPNO	ENAME	JOB	SAL	UPSAL
1	7369	SMITH	CLERK	800	824
2	7499	ALLEN	SALESMAN	1600	1680
3	7521	WARD	SALESMAN	1250	1312.5
4	7566	JONES	MANAGER	2975	3272.5
5	7654	MARTIN	SALESMAN	1250	1312.5
6	7698	BLAKE	MANAGER	2850	3135
7	7782	CLARK	MANAGER	2450	2695
8	7788	SCOTT	ANALYST	3000	3000
9	7839	KING	PRESIDENT	5000	5150
10	7844	TURNER	SALESMAN	1500	1575
11	7876	ADAMS	CLERK	1100	1133
12	7900	JAMES	CLERK	950	978.5
13	7902	FORD	ANALYST	3000	3000
14	7934	MILLER	CLERK	1300	1339

```
-- 6-48) CASE 문을 사용하여 출력하기

SELECT EMPNO, ENAME, JOB, SAL,
      CASE JOB
           WHEN 'MANAGER' THEN SAL*1.1
           WHEN 'SALESMAN' THEN SAL*1.05
           WHEN 'ANALYST' THEN SAL
           ELSE SAL*1.03
      END AS UPSAL
FROM EMP;
```

	EMPNO	ENAME	JOB	SAL	UPSAL
1	7369	SMITH	CLERK	800	824
2	7499	ALLEN	SALESMAN	1600	1680
3	7521	WARD	SALESMAN	1250	1312.5
4	7566	JONES	MANAGER	2975	3272.5
5	7654	MARTIN	SALESMAN	1250	1312.5
6	7698	BLAKE	MANAGER	2850	3135
7	7782	CLARK	MANAGER	2450	2695
8	7788	SCOTT	ANALYST	3000	3000
9	7839	KING	PRESIDENT	5000	5150
10	7844	TURNER	SALESMAN	1500	1575
11	7876	ADAMS	CLERK	1100	1133
12	7900	JAMES	CLERK	950	978.5
13	7902	FORD	ANALYST	3000	3000
14	7934	MILLER	CLERK	1300	1339

```
-- 6-49) 열 값에 따라서 출력 값이 달라지는 CASE 문

SELECT EMPNO, ENAME, COMM,
    CASE
        WHEN COMM IS NULL THEN '해당사항 없음'
        WHEN COMM = 0 THEN '수당없음'
        WHEN COMM > 0 THEN ' 수당 : ' || COMM
    END AS COMM_TEXT
FROM EMP;
```

	EMPNO	ENAME	COMM	COMM_TEXT
1	7369	SMITH	(null)	해당사항 없음
2	7499	ALLEN	300	수당 : 300
3	7521	WARD	500	수당 : 500
4	7566	JONES	(null)	해당사항 없음
5	7654	MARTIN	1400	수당 : 1400
6	7698	BLAKE	(null)	해당사항 없음
7	7782	CLARK	(null)	해당사항 없음
8	7788	SCOTT	(null)	해당사항 없음
9	7839	KING	(null)	해당사항 없음
10	7844	TURNER	0	수당없음
11	7876	ADAMS	(null)	해당사항 없음
12	7900	JAMES	(null)	해당사항 없음
13	7902	FORD	(null)	해당사항 없음
14	7934	MILLER	(null)	해당사항 없음