■STEP1. BASIC

Q001

-EMP 테이블에서 ENAME을 대문자, 소문자, 첫글자만 대문자로 조회하시오.

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

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

Q002

-EMP 테이블에서 UPPER를 이용하여 ENAME이 KING인 데이터를 조회하시오.

	ENAME
1	KING

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

Q003

-EMP 테이블에서 UPPER를 이용하여 ENAME에 KING인 포함된 데이터를 조회하시오.

- 대소문자 상관없이 KING인 사람을 조회하는 것이 가능해짐.

	ENAME
1	KING

```sql
SELECT *
  FROM EMP
 WHERE UPPER(ENAME) LIKE UPPER('%king%');
```

Q004

-EMP 테이블에서 LENGTH를 이용하여 ENAME의 문자열 길이를 조회하시오.

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

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

Q005

-EMP 테이블에서 ENAME의 문자열 길이가 5이상인 데이터를 조회하시오.

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

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

Q006 코드확인

- LENGTH('한글'), LENGTHB('한글')
- 문자열길이반환, 문자열 바이트 수 반환환

	LENGTH('한글')	LENGTHB('한글')
1	2	6

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

Q007

- 문자열 일부분을 추출
- SUBSTR(문자열 , 시작위치, 추출길이)

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

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

Q008 코드확인

- -의 의미는?
- C(-5)L(-4)E(-3)R(-2)K(-1)

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

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

Q009 코드확인

- 특정문자위치 찾기
- INSTR(문자열, 찾을거, 시작위치, 몇번째째)

	INSTR_1	INSTR_2	INSTR_3
1	3	12	4

```sql
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;
```

Q010 코드확인

- EMP테이블에서 INSTR 함수로 사원이름에 S가 있는 데이터를 조회하시오

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7566	JONES	MANAGER	7839	81/04/01	2975	(null)	20
2	7900	JAMES	CLERK	7698	81/12/11	950	(null)	30
3	7369	SMITH	CLERK	7902	80/12/09	800	(null)	20
4	7788	SCOTT	ANALYST	7566	82/12/22	3000	(null)	20
5	7876	ADAMS	CLERK	7788	83/01/15	1100	(null)	20

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

Q011

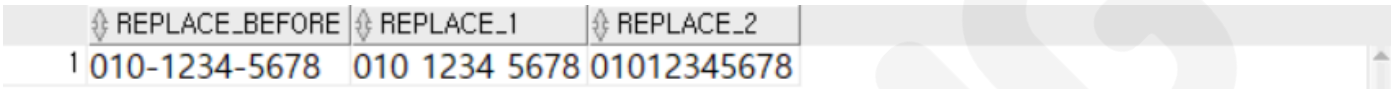
- EMP테이블에서 LIKE를 이용하여 사원이름에 S가 있는 데이터를 조회하시오

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7566	JONES	MANAGER	7839	81/04/01	2975	(null)	20
2	7900	JAMES	CLERK	7698	81/12/11	950	(null)	30
3	7369	SMITH	CLERK	7902	80/12/09	800	(null)	20
4	7788	SCOTT	ANALYST	7566	82/12/22	3000	(null)	20
5	7876	ADAMS	CLERK	7788	83/01/15	1100	(null)	20

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

Q012 코드확인

- REPLACE를 이용하여 연락처의 -을 공백으로, -을 뺀데이터로 조회하시오오

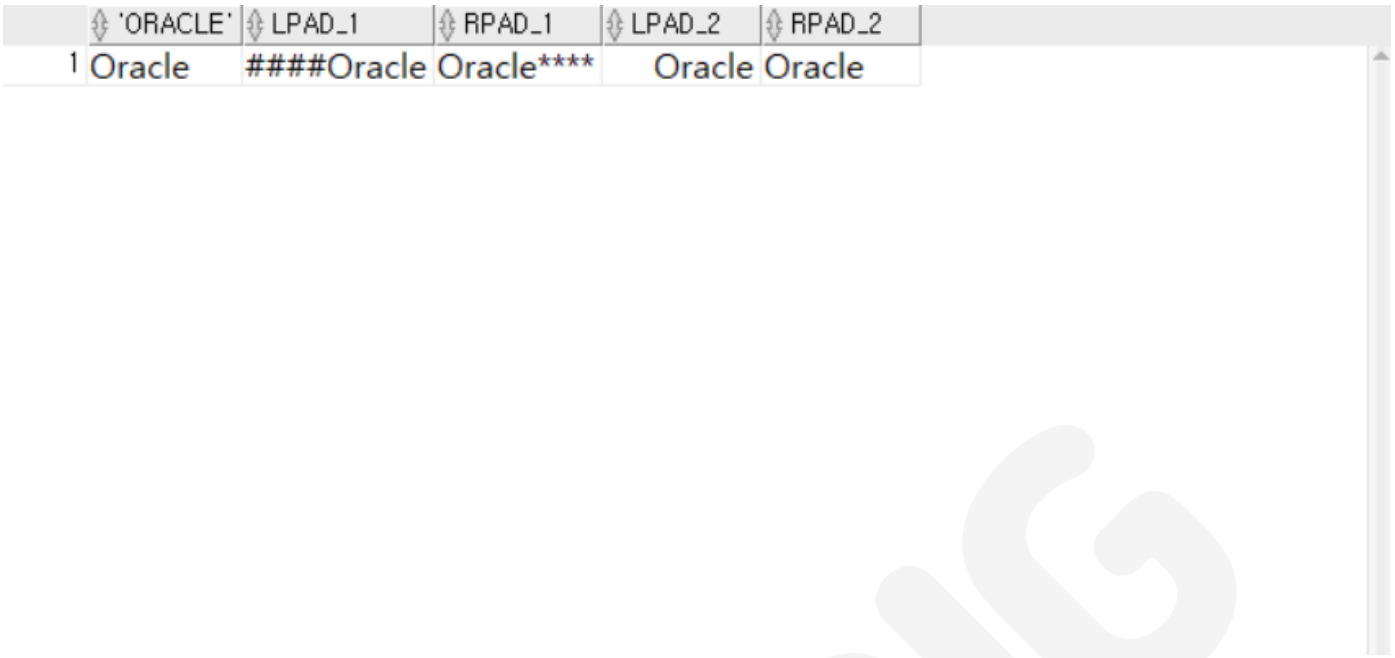


	REPLACE_BEFORE	REPLACE_1	REPLACE_2
1	010-1234-5678	010 1234 5678	01012345678

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

Q013 코드확인

- LPAD, RPAD를 이용하여 다음과 같이 데이터를 출력하시오오



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

```sql
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;
```

Q014

- RPAD를 이용하여 개인정보뒷자리 *로 출력하시오.

	RPAD_JMNO	RPAD_PHONE
1	911225-*******	010-1234-****

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

Q015

- EMP 테이블에서 EMPNO와 ENAME 사이에 :을 넣고 문자열을 연결하시오.

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

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

Q016 코드확인

- TRIM을 이용하여 다음과 같이 공백을 제거하고 출력하시오.

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

```sql
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;
```

Q017 코드확인

- TRIM을 이용하여 삭제할 문자 삭제후 출력가능능

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

```
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;
```

Q018 코드확인

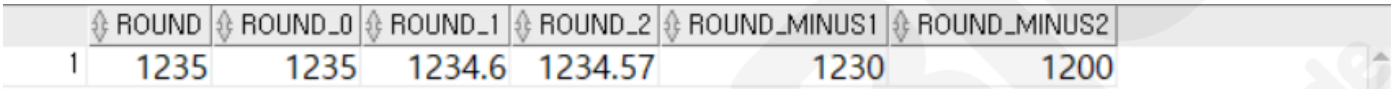
- TRIM, LTRIM, RTRIM 사용하여 문자열 출력하기기

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

```
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;
```

Q019

- ROUND를 이용하여 반올림 된 숫자 출력하기



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

```
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;
```

Q020

- 특정위치에서 버리는 TRUNC 함수수

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

```
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;
```

Q021

- CEIL : 가장 가까운 큰 정수,
- FLOOR : 가장 가까운 작은 정수 반환

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

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

Q022

- MOD : 특정 숫자를 나누고 그 나머지 출력력

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

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

Q023

- SYSDATE 함수를 사용하여 날짜 출력
- 하루이전, 이후후

	NOW	YESTERDAY	TOMORROW
1	24/12/21	24/12/20	24/12/22

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

Q024

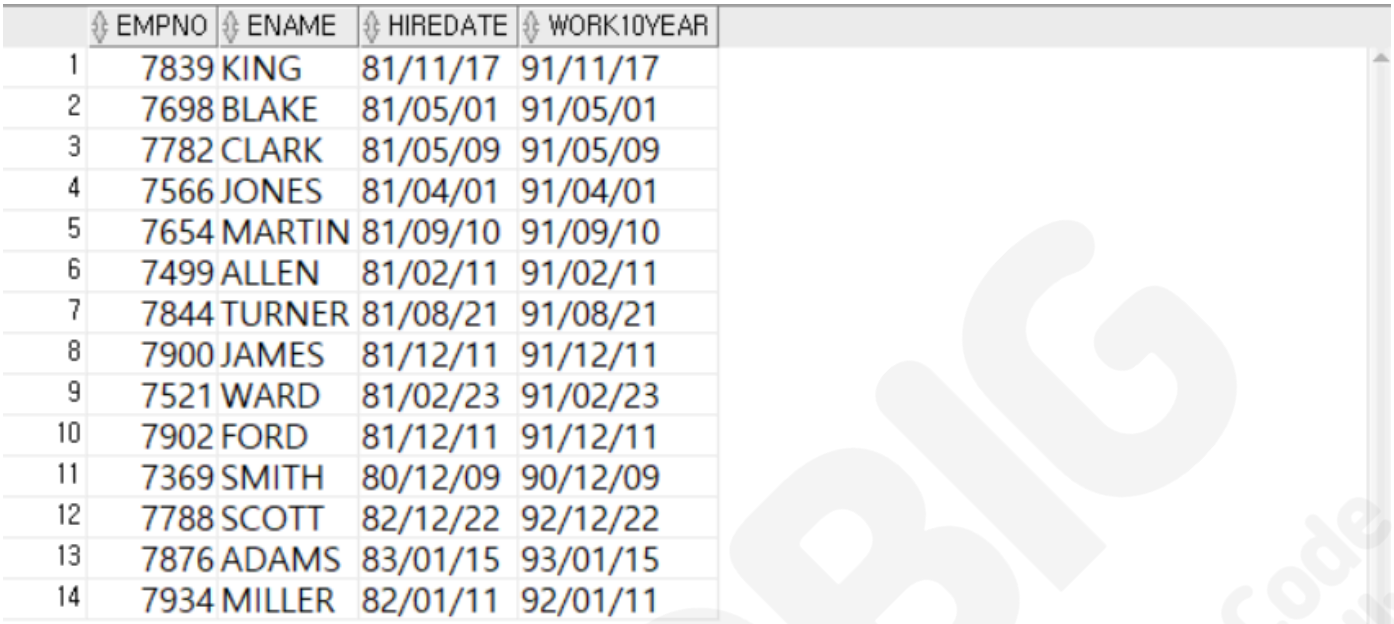
- ADD_MONTHS 3개월 후 날짜짜

	SYSDATE	ADD_MONTHS(SYSDATE,3)
1	24/12/21	25/03/21

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

Q025

- EMP 테이블에서 입사 10주년이 되는 사원들의 데이터를 출력하시오.



	EMPNO	ENAME	HIREDATE	WORK10YEAR
1	7839	KING	81/11/17	91/11/17
2	7698	BLAKE	81/05/01	91/05/01
3	7782	CLARK	81/05/09	91/05/09
4	7566	JONES	81/04/01	91/04/01
5	7654	MARTIN	81/09/10	91/09/10
6	7499	ALLEN	81/02/11	91/02/11
7	7844	TURNER	81/08/21	91/08/21
8	7900	JAMES	81/12/11	91/12/11
9	7521	WARD	81/02/23	91/02/23
10	7902	FORD	81/12/11	91/12/11
11	7369	SMITH	80/12/09	90/12/09
12	7788	SCOTT	82/12/22	92/12/22
13	7876	ADAMS	83/01/15	93/01/15
14	7934	MILLER	82/01/11	92/01/11

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

Q026

- EMP 테이블에서 입사 42년 미만인 사원데이터를 출력하시오.
- 12*42 = 504
- 42년이 지나 안나올 수 도 있음. 안나오면 개월수 늘려서 테스트해볼것

	EMPNO	ENAME	HIREDATE	SYSDATE
1	7788	SCOTT	82/12/22	24/12/21
2	7876	ADAMS	83/01/15	24/12/21

```
SELECT EMPNO, ENAME, HIREDATE, SYSDATE
  FROM EMP
 WHERE ADD_MONTHS(HIREDATE, 504) > SYSDATE;
```

Q027

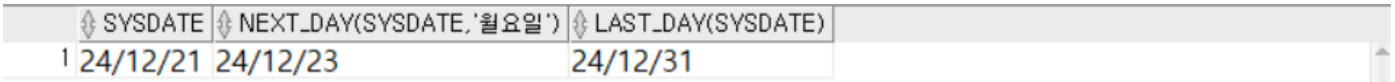
- EMP 테이블에서 HIREDATE와 SYSDATE사이의 개월수를 출력하시오.

	EMPNO	ENAME	HIREDATE	SYSDATE	MONTHS1	MONTHS2	MONTHS3
1	7839	KING	81/11/17	24/12/21	-517.156825716845878136200716845878136201	517.156825716845878136200716845878136201	517
2	7698	BLAKE	81/05/01	24/12/21	-523.672954749103942652329749103942652 33	523.672954749103942652329749103942652 33	523
3	7782	CLARK	81/05/09	24/12/21	-523.414890232974910394265232974910394265	523.414890232974910394265232974910394265	523
4	7566	JONES	81/04/01	24/12/21	-524.672954749103942652329749103942652 33	524.672954749103942652329749103942652 33	524
5	7654	MARTIN	81/09/10	24/12/21	-519.382632168458781362007168458781362007	519.382632168458781362007168458781362007	519
6	7499	ALLEN	81/02/11	24/12/21	-526.350374103942652329749103942652329749	526.350374103942652329749103942652329749	526
7	7844	TURNER	81/08/21	24/12/21	-520	520	520
8	7900	JAMES	81/12/11	24/12/21	-516.350374103942652329749103942652329749	516.350374103942652329749103942652329749	516
9	7521	WARD	81/02/23	24/12/21	-525.963277329749103942652329749103942652	525.963277329749103942652329749103942652	525
10	7902	FORD	81/12/11	24/12/21	-516.350374103942652329749103942652329749	516.350374103942652329749103942652329749	516
11	7369	SMITH	80/12/09	24/12/21	-528.414890232974910394265232974910394265	528.414890232974910394265232974910394265	528
12	7788	SCOTT	82/12/22	24/12/21	-503.995535394265232974910394265232974 91	503.995535394265232974910394265232974 91	503
13	7876	ADAMS	83/01/15	24/12/21	-503.221341845878136200716845878136200717	503.221341845878136200716845878136200717	503
14	7934	MILLER	82/01/11	24/12/21	-515.350374103942652329749103942652329749	515.350374103942652329749103942652329749	515

```
SELECT EMPNO, ENAME, HIREDATE, SYSDATE,
       MONTHS_BETWEEN(HIREDATE, SYSDATE) AS MONTHS1,
       MONTHS_BETWEEN(SYSDATE, HIREDATE) AS MONTHS2,
       TRUNC(MONTHS_BETWEEN(SYSDATE, HIREDATE)) AS MONTHS3
FROM EMP;
```

Q028

- 돌아오는 요일에 해당하는 날짜와 달의 마지막 날짜를 자도으로 계산산



	SYSDATE	NEXT_DAY(SYSDATE,'월요일')	LAST_DAY(SYSDATE)
1	24/12/21	24/12/23	24/12/31

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

Q029

- ROUND를 사용하여 날짜 데이터를 출력하시오.

	SYSDATE	FORMAT_CC	FORMAT_YYYY	FORMAT_Q	FORMAT_DDD	FORMAT_HH
1	24/12/21	01/01/01	25/01/01	25/01/01	24/12/22	24/12/21

```sql
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;
```

Q030

- TRUNC 함수를 사용하여 날짜 데이터를 출력하시오.

	SYSDATE	FORMAT_CC	FORMAT_YYYY	FORMAT_Q	FORMAT_DDD	FORMAT_HH
1	24/12/21	01/01/01	24/01/01	24/10/01	24/12/21	24/12/21

```sql
SELECT SYSDATE,
       TRUNC(SYSDATE, 'CC') AS FORMAT_CC,
       TRUNC(SYSDATE, 'YYYY') AS FORMAT_YYYY,
       TRUNC(SYSDATE, 'Q') AS FORMAT_Q,
       TRUNC(SYSDATE, 'DDD') AS FORMAT_DDD,
       TRUNC(SYSDATE, 'HH') AS FORMAT_HH
  FROM DUAL;
```

Q031

- 숫자와 문자열을 더하여 출력하시오

	EMPNO	ENAME	EMPNO+'500'
1	7788	SCOTT	8288

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

Q032

- 문자열과 숫자를 더하여 출력하시오. (에러발생!)

```
ORA-01722: invalid number

https://docs.oracle.com/error-help/db/ora-01722/01722, 00000 - "unable to convert string value containing %s to a number: %s"
*Document: YES
*Cause:    The attempted conversion of a character string for column or
           expression to a number failed because the character string is not
           a valid numeric literal. Only numeric fields or character fields
           containing numeric data can be used in arithmetic functions or
           expressions. Only numeric fields can be added to or subtracted from
           dates. If "UNISTR" appears in the error message, the value is
           not compatible with the national character set and cannot be
           represented directly.
*Action:   Use the LIKE expression to identify the problematic value. Ensure
           that it contains only digits, a sign, a decimal separator, and
           the character "E" or "e", and retry the operation.
*Params:   1) invalid_char
           the character string that is being converted to a number or
           UNISTR (character string) if the character is not compatible
           with the database character set.
           2) column_or_expression: The column or expression from where the
```

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

Q033

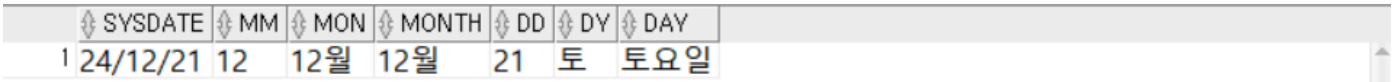
- SYSDATE 날짜 형식지정하여 출력하시오.

	현재날짜시간
1	2024/12/21 20:44:00

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

Q034

- 월과 요일을 다양한 형식으로 출력하시오.



	SYSDATE	MM	MON	MONTH	DD	DY	DAY
1	24/12/21	12	12월	12월	21	토	토요일

```
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;
```

Q035

- 여러 언어로 날짜(월) 출력하시오

	SYSDATE	MM	MON_KOR	MON_JPN	MON_ENG	MONTH_KOR	MONTH_JPN	MONTH_ENG
1	24/12/21	12	12월	12月	DEC	12월	12月	DECEMBER

```sql
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;
```

Q036

- 여러 언어로 날짜(요일일) 출력하시오

	SYSDATE	MM	DD	DY_KOR	DY_JPN	DY_ENG	DAY_KOR	DAY_JPN	DAY_ENG
1	24/12/21	12	21	토	土	SAT	토요일	土曜日	SATURDAY

```sql
SELECT SYSDATE,
       TO_CHAR(SYSDATE, 'MM') AS MM,
       TO_CHAR(SYSDATE, 'DD') AS 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;
```

Q037

- SYSDATE 시간형식 지정하여 출력하시오.

	SYSDATE	HH24MISS	HHMISS_AM	HHMISS_PM
1	24/12/21	20:44:54	08:44:54 오후	08:44:54 오후

```sql
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;
```

Q038

- 여러가지 숫자형식을 사용하여 급여를 출력하시오.

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

```
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;
```

Q039

- 문자데이터와 숫자데이터를 연산하여 출력하시오.

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

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

Q040

- 문자데이터끼지 연산하여 출력하시오 (에러러)

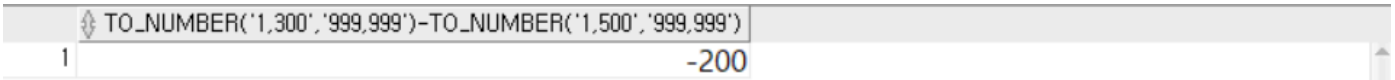
```
ORA-01722: invalid number

https://docs.oracle.com/error-help/db/ora-01722/01722, 00000 - "unable to convert string value containing %s to a number: %s"
*Document: YES
*Cause:    The attempted conversion of a character string for column or
           expression to a number failed because the character string is not
           a valid numeric literal. Only numeric fields or character fields
           containing numeric data can be used in arithmetic functions or
           expressions. Only numeric fields can be added to or subtracted from
           dates. If "UNISTR" appears in the error message, the value is
           not compatible with the national character set and cannot be
           represented directly.
*Action:   Use the LIKE expression to identify the problematic value. Ensure
           that it contains only digits, a sign, a decimal separator, and
           the character "E" or "e", and retry the operation.
*Params:   1) invalid_char
           the character string that is being converted to a number or
           UNISTR (character string) if the character is not compatible
           with the database character set.
           2) column_or_expression: The column or expression from where the
           invalid character comes
```

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

Q041

- TO_NUMBER 함수로 연산하여 출력하시오.



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

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

Q042

- TO_DATE를 이용하여 문자 데이터를 날짜 데이터로 변환하시오.

	TODATE1	TODATE2
1	18/07/14	18/07/14

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

Q043

- EMP 테이블에서 1981년 6월 1일 이후에 입사한 사원정보를 출력하시오.

	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
1	7839	KING	PRESIDENT	(null)	81/11/17	5000	(null)	10
2	7654	MARTIN	SALESMAN	7698	81/09/10	1250	1400	30
3	7844	TURNER	SALESMAN	7698	81/08/21	1500	0	30
4	7900	JAMES	CLERK	7698	81/12/11	950	(null)	30
5	7902	FORD	ANALYST	7566	81/12/11	3000	(null)	20
6	7788	SCOTT	ANALYST	7566	82/12/22	3000	(null)	20
7	7876	ADAMS	CLERK	7788	83/01/15	1100	(null)	20
8	7934	MILLER	CLERK	7782	82/01/11	1300	(null)	10

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

Q044

- 여러가지 형식으로 날짜 데이터를 출력하시오.

	YY_YEAR_49	RR_YEAR_49	YY_YEAR_50	RR_YEAR_50	YY_YEAR_51	RR_YEAR_51
1	49/12/10	49/12/10	50/12/10	50/12/10	51/12/10	51/12/10

```
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;
```


Q045

- EMP테이블에서 NVL 함수를 사용하여 다음과 같이 출력하시오.

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

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

Q046

- EMP테이블에서 NVL2 함수를 사용하여 다음과 같이 출력하시오.

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

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

Q047

- EMP테이블에서 DECODE 함수를 사용하여 다음과 같이 출력하시오.
- JOB이 MANAGER 는 급여의 10% 인상한 급여
  SALESMAN 는 급여의 5% 인상한 급여
  ANALYST 는 그대로
  나머지는 3% 인상된 급여

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

```sql
SELECT EMPNO, ENAME, JOB, SAL,
       DECODE(JOB,
              'MANAGER' , SAL*1.1,
              'SALESMAN', SAL*1.05,
              'ANALYST' , SAL,
              SAL*1.03) AS UPSAL
  FROM EMP;
```


Q048

- EMP테이블에서 CASE 함수를 사용하여 다음과 같이 출력하시오.
- JOB이 MANAGER 는 급여의 10% 인상한 급여
  SALESMAN 는 급여의 5% 인상한 급여
  ANALYST 는 그대로
  나머지는 3% 인상된 급여

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

```sql
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;
```


Q049

- 기준데이터 없이 조건식으로만 CASE 사용가능
- COMM 값이 NULL 이면 해당사항 없음
  0 이면 수당없음
  0 초과시 초과한 수당을 출력력

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

```
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;
```

■STEP2.

EX001

- EMP 테이블에서 다음과 같은 결과가 나오도록 SQL문을 작성하시오.

1. EMP 테이블에서 ENAME이 다섯글자 이상이며 여섯글자 미만인 사원을 조회하시오.
2. MASKING_EMPNO 는 EMPNO 앞두자리외 뒷자리를 *로 출력
3. MASKING_ENAME 는 사원이름의 첫글자만 보여주고 나머지는 *로 출력

※ 앞자리 추출 - SUBSTR(문자열, 어디에서, 몇개)
※ RPAD - RPAD(문자열, 몇자리, 채울값)

	EMPNO	MASKING_EMPNO	ENAME	MASKING_ENAME
1	7698	76**	BLAKE	B****
2	7782	77**	CLARK	C****
3	7566	75**	JONES	J****
4	7499	74**	ALLEN	A****
5	7900	79**	JAMES	J****
6	7369	73**	SMITH	S****
7	7788	77**	SCOTT	S****
8	7876	78**	ADAMS	A****

EX002

- EMP 테이블에서 다음과 같은 결과가 나오도록 SQL문을 작성하시오.

1. EMP 테이블에서 사원들의 월 평균 근무일 수는 21.5일
   2 하루 근무시간을 8시간으로 보았을때 사원들의 하루급여(DAY_PAY) 와 시급(TIME_PAY)을 계산하여 결과를 조회하시오.
   ※ 하루급여는 소수점 세번째 자리에서 버리고(TRUNC), 시급은 두번째 소수점에서 반올림(ROUND)하시오

	EMPNO	ENAME	SAL	DAY_PAY	TIME_PAY
1	7839	KING	5000	232.55	29.1
2	7698	BLAKE	2850	132.55	16.6
3	7782	CLARK	2450	113.95	14.2
4	7566	JONES	2975	138.37	17.3
5	7654	MARTIN	1250	58.13	7.3
6	7499	ALLEN	1600	74.41	9.3
7	7844	TURNER	1500	69.76	8.7
8	7900	JAMES	950	44.18	5.5
9	7521	WARD	1250	58.13	7.3
10	7902	FORD	3000	139.53	17.4
11	7369	SMITH	800	37.2	4.7
12	7788	SCOTT	3000	139.53	17.4
13	7876	ADAMS	1100	51.16	6.4
14	7934	MILLER	1300	60.46	7.6

EX003

- EMP 테이블에서 다음과 같은 결과가 나오도록 SQL문을 작성하시오.

1. EMP테이블에서 사원들은 입사일(HIREDATE)을 기준으로 3개월이 지난 후 첫 월요일에 정직원이 됨
2. 사원들이 정직원이 되는 날짜(R_JOB)를 YYYY-MM-DD 형식으로 오른쪽과 같이 출력하시오.
3. 추가 수당(COMM)이 없는 사원들의 추가수당은 N/A로 출력하시오.

	EMPNO	ENAME	HIREDATE	R_JOB	COMM
1	7839	KING	81/11/17	1982-02-22	N/A
2	7698	BLAKE	81/05/01	1981-08-03	N/A
3	7782	CLARK	81/05/09	1981-08-10	N/A
4	7566	JONES	81/04/01	1981-07-06	N/A
5	7654	MARTIN	81/09/10	1981-12-14	1400
6	7499	ALLEN	81/02/11	1981-05-18	300
7	7844	TURNER	81/08/21	1981-11-23	0
8	7900	JAMES	81/12/11	1982-03-15	N/A
9	7521	WARD	81/02/23	1981-05-25	500
10	7902	FORD	81/12/11	1982-03-15	N/A
11	7369	SMITH	80/12/09	1981-03-16	N/A
12	7788	SCOTT	82/12/22	1983-03-28	N/A
13	7876	ADAMS	83/01/15	1983-04-18	N/A
14	7934	MILLER	82/01/11	1982-04-12	N/A

EX004

- EMP 테이블에서 다음과 같은 결과가 나오도록 SQL문을 작성하시오.

1. 직속상관의 사원번호(MGR)를 다음과 같은 조건을 기준으로 변환해서 CHG_MGR열에 출력하시오

```
1) 직속상관의 사원번호가 존재하지 않을경우 : 00000
2) 직속상관의 사원번호 앞 두자리가 75일 경우 : 5555
3) 직속상관의 사원번호 앞 두자리가 76일 경우 : 6666
4) 직속상관의 사원번호 앞 두자리가 77일 경우 : 7777
5) 직속상관의 사원번호 앞 두자리가 78일 경우 : 8888
6) 그 외 직속상관 사원번호의 경우 : 본래 직속상관의 사원번호 그대로 출력
```

	EMPNO	ENAME	MGR	CHG_MGR
1	7839	KING	(null)	0000
2	7698	BLAKE	7839	8888
3	7782	CLARK	7839	8888
4	7566	JONES	7839	8888
5	7654	MARTIN	7698	6666
6	7499	ALLEN	7698	6666
7	7844	TURNER	7698	6666
8	7900	JAMES	7698	6666
9	7521	WARD	7698	6666
10	7902	FORD	7566	5555
11	7369	SMITH	7902	7902
12	7788	SCOTT	7566	5555
13	7876	ADAMS	7788	7777
14	7934	MILLER	7782	7777