



Database 프로그래밍을 위한
오라클 명령어

08

SQL 주요 함수

강 사 : 김 진 성



목 차

1

DUAL 테이블과 SQL 함수 분류

2

숫자 함수

3

문자 처리 함수

4

날짜 함수

5

형 변환 함수

01. DUAL 테이블과 SQL 함수 분류



- ❖ DUAL 테이블의 구조를 살펴보기 위해서 DESC 명령어를 사용

예 **DESC DUAL**

- ❖ DUAL 테이블은 DUMMY라는 단 하나의 컬럼으로 구성되어 있다.
- ❖ 이 컬럼의 최대 길이 : 1

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> DESC DUAL
이름                                널?       유형
-----
DUMMY                                YES        VARCHAR2(1)
```

01. DUAL 테이블과 SQL 함수 분류



- ❖ DUMMY 컬럼엔 과연 어떤 값이 저장되어 있는 것일까요?

예

```
SELECT *  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT *  
2 FROM DUAL;  
  
D  
-  
X
```

- ❖ DUAL 테이블은 DUMMY라는 단 하나의 컬럼에 X라는 단 하나의 로우만을 저장하고 있으나 이 값은 아무런 의미가 없다.
- ❖ 쿼리문의 수행 결과가 하나의 로우로 출력되도록 하기 위해서 단 하나의 로우를 구성하고 있을 뿐이다.

02. 숫자 함수



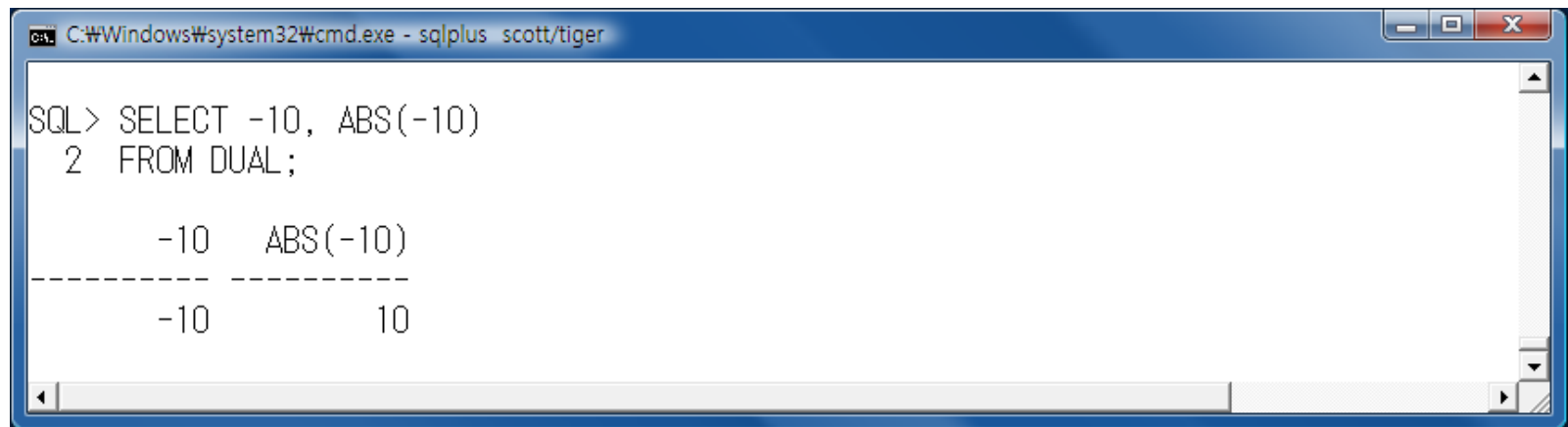
구 분	설 명
ABS	절대값을 구한다.
COS	COSINE 값을 반환한다.
EXP	$e(2.71828183\cdots)$ 의 n 승을 반환한다.
LOG	LOG값을 반환한다.
SIN	SINE값을 반환한다.
TAN	TANGENT값을 반환한다.
ROUND	특정 자릿수에서 반올림한다.
TRUNC	특정 자릿수에서 잘라낸다. (버림)
MOD	입력 받은 수를 나눈 나머지 값을 반환한다.

2.1 절대값 구하는 ABS 함수

- ❖ ABS 함수는 절대값을 구한다. 절대값은 방향은 없고 크기만 있는 것으로서 주어진 데이터가 음수일 경우 양수로 표현
- ❖ -10 에 대한 절대값을 구해 보시다.

예

```
SELECT -10, ABS(-10) FROM DUAL;
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". The prompt is "SQL>". The user has entered the command "SELECT -10, ABS(-10) FROM DUAL;". The output is displayed as follows:

-10	ABS(-10)
-10	10

2.2 소수점 아래를 버리는 FLOOR 함수



- ❖ FLOOR 함수는 소수점 아래를 버린다. 34.5678를 FLOOR 함수에 적용하면 34가 구해진다.
- ❖ -10 에 대한 절대값 구하기

예

```
SELECT 34.5678, FLOOR(34.5678)
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT 34.5678, FLOOR(34.5678)
2  FROM DUAL;

34.5678 FLOOR(34.5678)
-----
34.5678              34
```

2.3 특정 자릿수에서 반올림하는 ROUND 함수

- ❖ 34.5678를 반올림하면 35 이과 같이 반올림한 결과를 구하기 위한 함수로 오라클에서는 ROUND가 제공
- ❖ -10 에 대한 절대값 구하기



예

```
SELECT 34.5678, ROUND(34.5678)
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT 34.5678, ROUND(34.5678)
2  FROM DUAL;

34.5678 ROUND(34.5678)
-----
34.5678                35
```


2.3 특정 자릿수에서 반올림하는 ROUND 함수



- ❖ ROUND 함수는 지정한 자릿수에서 반올림을 할 수 있도록 자릿수를 지정할 수 있다.

형식	ROUND(대상, 자릿수)
----	----------------

- ❖ ROUND 함수의 두 번째 인자 값이 2이면 소수점 이하 세 번째 자리에서 반올림하여 소수점 이하 두 번째 자리까지 표시 즉, ROUND(34.5678, 2)의 결과는 34.57이 된다.

예	SELECT 34.5678, ROUND(34.5678, 2) FROM DUAL;
---	---

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT 34.5678, ROUND(34.5678, 2)
2 FROM DUAL;

34.5678 ROUND(34.5678,2)
-----
34.5678          34.57
```

2.3 특정 자릿수에서 반올림하는 ROUND 함수



- ❖ 자릿수에 음수를 지정할 수 있는데 이럴 경우 소수점 이하가 아니라 반대쪽인 일단위, 십단위, 백단위 순으로 거슬러 올라가며, 인자 값이 양수일 때와는 달리 해당 자리에서 반올림

❖ 예 **SELECT 34.5678, ROUND(34.5678, -1)
FROM DUAL;**

- ❖ **ROUND(34.5678, -1)의 결과는 30**

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT 34.5678, ROUND(34.5678, -1)
2  FROM DUAL;

34.5678 ROUND(34.5678, -1)
-----
34.5678                    30
```

2.4 특정 자릿수에서 잘라내는 TRUNC 함수

❖ TRUNC 함수는 지정한 자리 수 이하를 버리는 함수



예

```
SELECT TRUNC(34.5678, 2), TRUNC(34.5678, -1), TRUNC(34.5678)
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT TRUNC(34.5678, 2), TRUNC(34.5678, -1), TRUNC(34.5678)
2  FROM DUAL;

TRUNC(34.5678,2) TRUNC(34.5678,-1) TRUNC(34.5678)
-----
          34.56                30                34
```

2.5 나머지 구하는 MOD 함수

❖ MOD 함수는 나눗셈 연산 후에 나머지를 결과 리턴 함수



예 **SELECT MOD (27, 2), MOD (27, 5), MOD (27, 7)
FROM DUAL;**

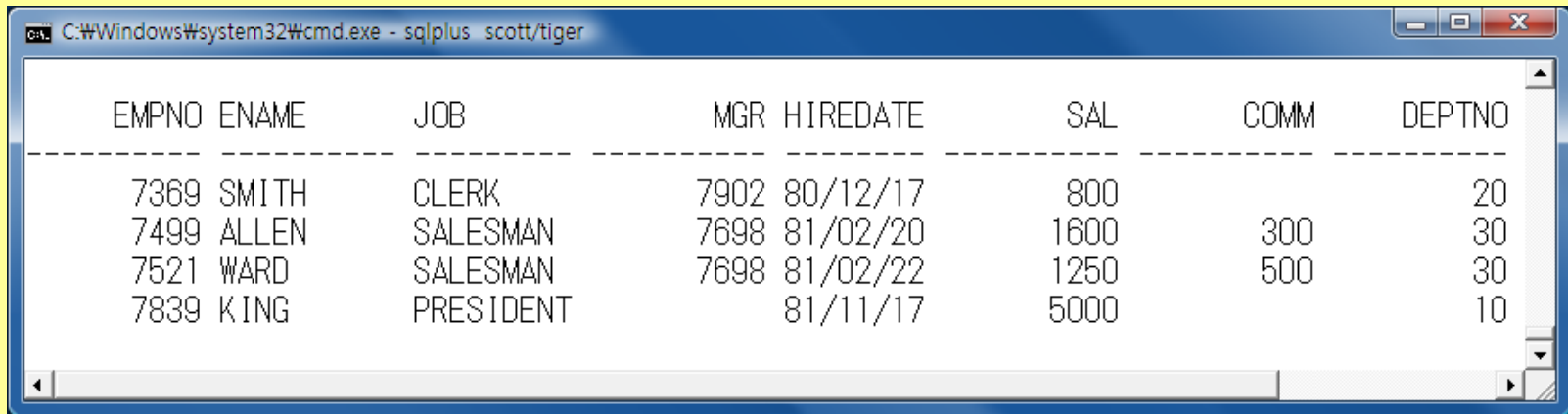
```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT MOD (27, 2), MOD (27, 5), MOD (27, 7)
2 FROM DUAL;

MOD(27,2)  MOD(27,5)  MOD(27,7)
-----
          1           2           6
```

<실습>

1. 사번이 홀수인 사람들을 검색해 보십시오.(EMP 테이블)



EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	80/12/17	800		20
7499	ALLEN	SALESMAN	7698	81/02/20	1600	300	30
7521	WARD	SALESMAN	7698	81/02/22	1250	500	30
7839	KING	PRESIDENT		81/11/17	5000		10

03. 문자 처리 함수



구 분	설 명
LOWER	소문자로 변환한다.
UPPER	대문자로 변환한다.
INITCAP	첫 글자만 대문자로 나머지 글자는 소문자로 변환한다.
CONCAT	문자의 값을 연결한다.
SUBSTR	문자를 잘라 추출한다. (한글 1Byte)
SUBSTRB	문자를 잘라 추출한다. (한글 2Byte)
LENGTH	문자의 길이를 반환한다.(한글 1Byte)
LENGTHB	문자의 길이를 반환한다.(한글 2Byte)

03. 문자 처리 함수



구 분	설 명
LPAD, RPAD	입력 받은 문자열과 기호를 정렬하여 특정 길이의 문자열로 반환한다.
TRIM	잘라내고 남은 문자를 표시한다.
CONVERT	CHAR SET을 변환한다.
CHR	ASCII 코드 값으로 변환한다.
ASCII	ASCII 코드 값을 문자로 변환한다.
REPLACE	문자열에서 특정 문자를 변경한다.

3.1 대문자로 변환하는 UPPER 함수



- ❖ UPPER 함수는 입력한 문자값을 대문자로 변환하는 함수
- ❖ 문자가 변환되는 결과를 명료하게 살펴보기 위해서 특정 테이블에 대한 컬럼에 대해서 함수를 적용하는 것은 나중에 하기로 하고 우선 문자 상수에 대해서 적용해보자.

예

```
SELECT 'Welcome to Oracle', UPPER('Welcome to Oracle')  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT 'Welcome to Oracle', UPPER('Welcome to Oracle')  
2 FROM DUAL;  
  
'WELCOMETOORACLE' UPPER('WELCOMETOO  
-----  
Welcome to Oracle WELCOME TO ORACLE
```


3.2 소문자로 변환하는 LOWER 함수

❖ LOWER 함수는 문자열을 모두 소문자로 변경



예

```
SELECT 'Welcome to Oracle', LOWER('Welcome to Oracle')  
FROM DUAL;
```

A screenshot of a Windows command prompt window. The title bar shows 'C:\Windows\system32\cmd.exe - sqlplus scott/tiger'. The command prompt displays the following SQL query and its output:

```
SQL> SELECT 'Welcome to Oracle', LOWER('Welcome to Oracle')  
2 FROM DUAL;  
  
'WELCOMETOORACLE' LOWER('WELCOMETOO  
-----  
Welcome to Oracle welcome to oracle
```

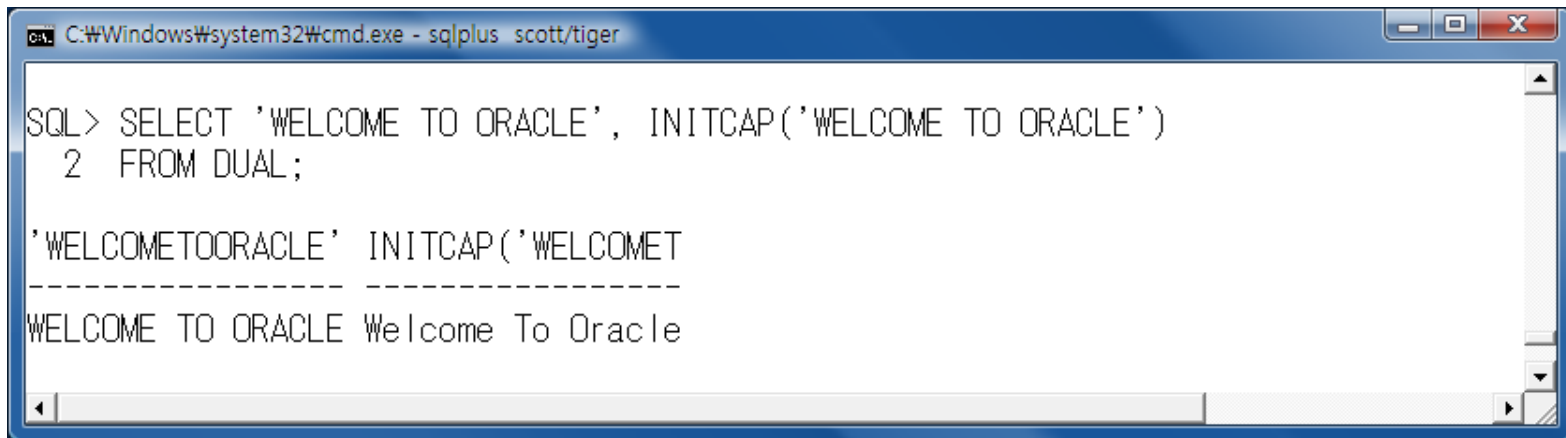
3.3 이니셜만 대문자로 변환하는 INITCAP 함수

❖ INITCAP 함수는 단어 시작만 대문자로 변경



예

```
SELECT 'WELCOME TO ORACLE',  
       INITCAP('WELCOME TO ORACLE')  
FROM DUAL;
```



```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT 'WELCOME TO ORACLE', INITCAP('WELCOME TO ORACLE')
2  FROM DUAL;

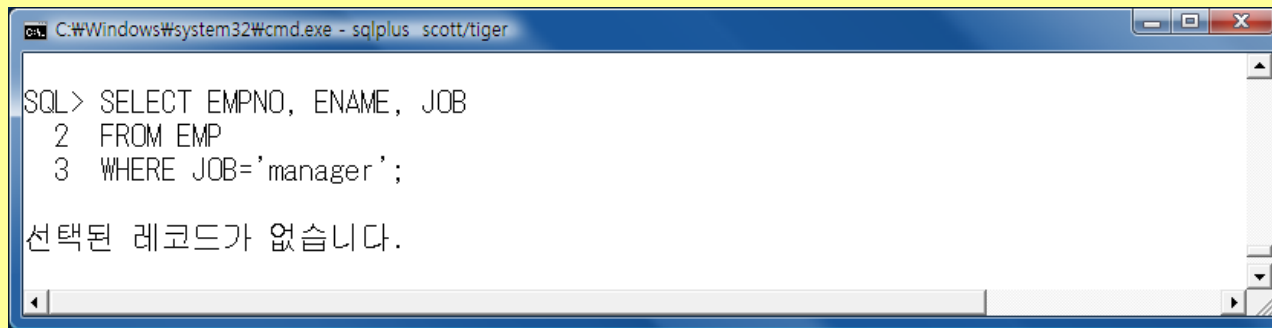
'WELCOMETOORACLE' INITCAP('WELCOMET
-----
WELCOME TO ORACLE Welcome To Oracle
```

실습

2. 다음과 같이 쿼리문을 구성하면 직급이 'manager'인 사원을 검색할까?

예

```
SELECT EMPNO, ENAME, JOB  
FROM EMP  
WHERE JOB='manager';
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". Inside the window, the following SQL query is entered:

```
SQL> SELECT EMPNO, ENAME, JOB  
2 FROM EMP  
3 WHERE JOB='manager';
```

The output of the query is displayed below the prompt:

```
선택된 레코드가 없습니다.
```

결론적으로 위와 같이 쿼리문을 기술하면 검색에 실패합니다.

왜냐하면 직급 칼럼에 저장된 데이터는 모두 대문자 형태이기 때문이다.

3.4 문자 길이를 구하는 LENGTH



- ❖ LENGTH 함수는 컬럼에 저장된 데이터가 몇 개의 문자로 구성되었는지 길이를 알려주는 함수
- ❖ 영문자와 한글의 길이를 구해보자.

예

```
SELECT LENGTH('Oracle'), LENGTH('오라클')  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT LENGTH('Oracle'), LENGTH('오라클')  
2 FROM DUAL;  
  
LENGTH('ORACLE') LENGTH('오라클')  
-----  
6 3
```

3.5 바이트 수를 알려주는 LENGTHB 함수



❖ 이번에 살펴볼 LENGTHB 함수는 바이트 수를 알려주는 함수

예

```
SELECT LENGTHB('Oracle'), LENGTHB('오라클')  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT LENGTHB('Oracle'), LENGTHB('오라클')  
2 FROM DUAL;  
  
LENGTHB('ORACLE') LENGTHB('오라클')  
-----  
6 6  
  
SQL>
```

❖ 한글 1자는 2바이트를 차지한다. 그렇기 때문에 수행 결과를 보면 한글 3자로 구성된 '오라클'의 LENGTHB 함수의 결과는 6이 된다.

3.6 문자열 일부만 추출하는 SUBSTR 함수



- ❖ SUBSTR 과 SUBSTRB 함수는 대상 문자열이나 칼럼의 자료에서 시작위치부터 선택 개수만큼의 문자를 추출
- ❖ SUBSTRB 함수도 같은 형식이지만 명시된 개수만큼의 문자가 아닌 바이트 수를 잘라낸다는 점에서만 차이가 나타난다.

형식 SUBSTR(*대상, 시작위치, 추출할 개수*)

예 SELECT SUBSTR('Welcome to Oracle', 4, 3)
FROM DUAL;

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT SUBSTR('Welcome to Oracle', 4, 3)
2  FROM DUAL;

SUB
---
com

SQL>
```

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
W	e	l	c	o	m	e		t	o		O	r	a	c	l	e

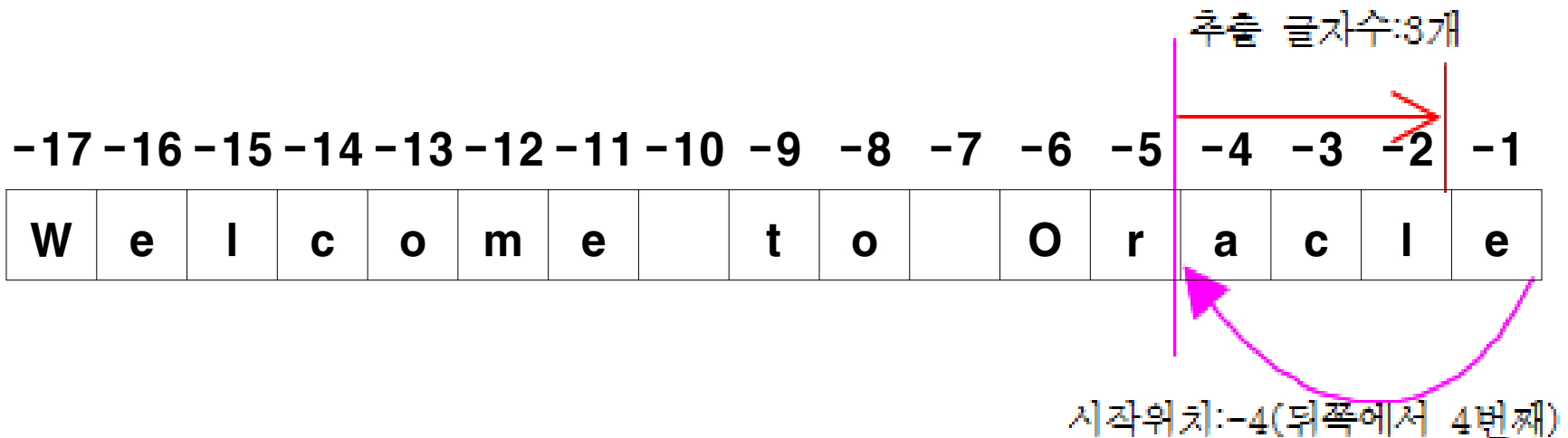
3.6 문자열 일부만 추출하는 SUBSTR 함수



- ❖ 시작위치 인자 값을 음수 값으로 줄 수 있는데 이때는 문자열의 앞쪽이 아닌 뒤 쪽에서부터 카운터 하여 시작위치를 잡는다.

예

```
SELECT SUBSTR('Welcome to Oracle', -4, 3)  
FROM DUAL;
```



3.6 문자열 일부만 추출하는 SUBSTR 함수



- ❖ 직원들의 입사년도만 출력할 경우
- ❖ SUBSTR 함수를 이용해서 입사일을 저장하고 있는 HIREDATE 컬럼에서 첫 글자부터 2개를 추출한다.
- ❖ 입사한 달만 출력하려면 HIREDATE 컬럼에서 네번째 글자부터 2개를 추출한다.

예

```
SELECT SUBSTR(HIREDATE, 1, 2) 년도,  
       SUBSTR(HIREDATE, 4, 2) 달 FROM EMP;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT SUBSTR(HIREDATE, 1, 2) 년도, SUBSTR(HIREDATE, 4, 2) 달  
2 FROM EMP;
```

년도	달
80	12
81	02
81	02
81	04
81	09
81	05
81	06
87	04

실습



❖ 3. 9월에 입사한 사원을 출력해보세요. (EMP 테이블)

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

S


```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7654	MARTIN	SALESMAN	7698	81/09/28	1250	1400	30
7844	TURNER	SALESMAN	7698	81/09/08	1500	0	30

```
SQL>
```



❖ 3. 9월에 입사한 사원을 출력해보세요. (EMP 테이블)

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT *
  2  FROM EMP
  3  WHERE SUBSTR(HIREDATE, 4, 2)='09';
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7654	MARTIN	SALESMAN	7698	81/09/28	1250	1400	30
7844	TURNER	SALESMAN	7698	81/09/08	1500	0	30

```
SQL>
```

3.7 바이트 수를 기준으로 문자열 일부만 추출하는 SUBSTRB 함수



- ❖ 한글 1자는 2바이트를 차지하기 때문에 SUBSTR 함수와 SUBSTRB 함수의 결과가 달라진다.

예 `SELECT SUBSTR('웰컴투오라클', 3, 4), SUBSTRB('웰컴투오라클', 3, 4)
FROM DUAL;`

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT SUBSTR('웰컴투오라클', 3, 4), SUBSTRB('웰컴투오라클', 3, 4)
2 FROM DUAL;

SUBSTR(' SUBS
-----
투오라클 컴투

SQL>
```

3.8 특정 문자의 위치를 구하는 INSTR 함수



- ❖ INSTR 함수는 대상 문자열이나 컬럼에서 특정 문자가 나타나는 위치를 알려준다.
- ❖ 문자열 'Welcome to Oracle'에 'O'가 저장된 위치가 얼마인지 알고 싶을 때에는 INSTR 함수를 사용하여 다음과 같이 쿼리문을 작성할 수 있다.

예

```
SELECT INSTR('WELCOME TO ORACLE', 'O')  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
5  
SQL> SELECT INSTR('WELCOME TO ORACLE', 'O')  
2 FROM DUAL;  
  
INSTR('WELCOMETOORACLE', 'O')  
-----  
5  
  
SQL>
```

3.8 특정 문자의 위치를 구하는 INSTR 함수



- ❖ INSTR 함수의 기본 형식은 다음과 같다.

형식 INSTR(*대상*, *찾을글자*, *시작위치*, *몇_번째_발견*)

- ❖ '시작위치'와 '몇 번째 발견'을 생략한 경우 모두 1로 간주
즉, 시작 위치도 1이고 첫 번째 발견된 위치를 반환

예

```
SELECT INSTR('WELCOME TO ORACLE', 'O', 6, 2)
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT INSTR('WELCOME TO ORACLE', 'O', 6, 2)
2 FROM DUAL;

INSTR('WELCOMETOORACLE','O',6,2)
-----
12

SQL>
```

3.9 왼쪽에서 공백 문자를 삭제하는 LTRIM 함수



❖ LTRIM 함수는 문자열의 왼쪽(앞)의 공백 문자들을 삭제

예

```
SELECT LTRIM(' Oracle ')
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT LTRIM(' Oracle ')
2 FROM DUAL;

LTRIM('OR
-----
Oracle

SQL>
```

3.10 오른쪽에서 공백 문자를 삭제하는 RTRIM 함수

❖ RTRIM 함수 역시 다음과 같이 기술하면 공백 문자를 잘라낸다.



예

```
SELECT RTRIM(' Oracle ')  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT RTRIM(' Oracle ')  
2  FROM DUAL;  
  
RTRIM('O  
-----  
Oracle  
  
SQL>
```

3.11 특정 문자를 잘라내는 TRIM 함수



- ❖ TRIM 함수는 칼럼이나 대상 문자열에서 특정 문자가 첫 번째 글자이거나 마지막 글자이면 잘라내고 남은 문자열만 반환

예

```
SELECT TRIM('a' FROM 'aaaaOracleaaaa')  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT TRIM('a' FROM 'aaaaOracleaaaa')  
2 FROM DUAL;  
  
TRIM(''  
-----  
Oracle  
  
SQL>
```

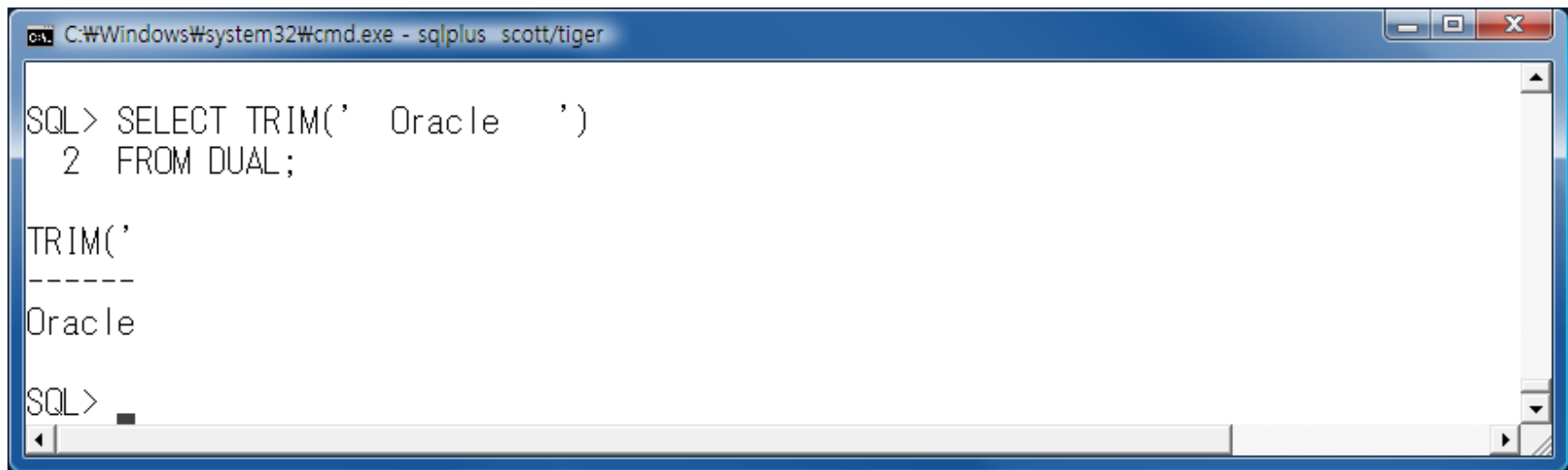

3.11 특정 문자를 잘라내는 TRIM 함수



❖ 다음과 같이 기술하면 앞과 뒤 공백 문자를 제거해준다.

예

```
SELECT TRIM(' Oracle ')
FROM DUAL;
```



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". The prompt is "SQL>". The user has entered the SQL query "SELECT TRIM(' Oracle ')" and pressed Enter. The output shows the result of the query, which is "Oracle". The prompt is now "SQL>".

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT TRIM(' Oracle ')
2  FROM DUAL;

TRIM('
-----
Oracle

SQL>
```

04. 날짜 함수



구 분	설 명
SYSDATE	시스템 저장된 현재 날짜를 반환한다.
MONTHS_BETWEEN	두 날짜 사이가 몇 개월인지를 반환한다.
ADD_MONTHS	특정 날짜에 개월 수를 더한다.
NEXT_DAY	특정 날짜에서 최초로 도래하는 인자로 받은 요일의 날짜를 반환한다.
LAST_DAY	해당 달의 마지막 날짜를 반환한다.
ROUND	인자로 받은 날짜를 특정 기준으로 반올림한다.
TRUNC	인자로 받은 날짜를 특정 기준으로 버린다.

4.1 현재 날짜를 반환하는 SYSDATE 함수



- ❖ SYSDATE 함수는 시스템에 저장된 현재 날짜를 반환하는 함수입니다. 다음은 시스템에서 현재 날짜를 얻어 와서 출력

예

```
SELECT SYSDATE  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT SYSDATE  
2  FROM DUAL;  
  
SYSDATE  
-----  
09/01/17
```

4.2 날짜 연산



- ❖ 날짜 형 데이터에 숫자를 더하면(날짜+숫자) 그 날짜로부터 그 기간만큼 지난 날짜를 계산한다. 날짜 형 데이터에 숫자를 빼면(날짜-숫자) 그 날짜로부터 그 기간만큼 이전 날짜를 구한다.

예

```
SELECT SYSDATE-1 어제, SYSDATE 오늘, SYSDATE+1 내일  
FROM DUAL;
```

```
C:\WINDOWS\system32\cmd.exe - sqlplus scott/tiger  
SQL> SELECT SYSDATE-1 어제, SYSDATE 오늘, SYSDATE+1 내일  
2 FROM DUAL;  
  
어제      오늘      내일  
-----  
09/01/12  09/01/13  09/01/14
```

4.3 특정 기준으로 반올림하는 ROUND 함수

- ❖ ROUND 함수는 형식을 지정하면 숫자 이외에 날짜에 대해서도 반올림을 할 수 있다.



형식 ROUND (*date, format*)

포맷 모델	단위
CC, SCC	4자리 연도의 끝 두 글자를 기준으로 반올림
SYYY, YYYYY, YEAR SYEAR, YYY, YY, Y	년(7월 1일부터 반올림)
DDD, D, J	일을 기준
HH, HH12, HH24	시를 기준
Q	한 분기의 두 번째 달의 16일을 기준으로 반올림
MONTH, MON, MM, RM	월(16일을 기준으로 반올림)
DAY,DY,D	한주가 시작되는 날짜
MI	분을 기준

4.3 특정 기준으로 반올림하는 ROUND 함수



- ❖ ROUND 함수의 형식으로 MONTH를 지정하였기에 특정 날짜(DATE)를 월(MONTH)을 기준으로 반올림한 날짜를 구한다.
- ❖ 일을 기준으로 16일보다 적으면 이번 달 1일을 크면 다음달 1일을 구한다. 6월 9일 반올림하며 6월 1일이 되고 11월 17일은 반올림하여 12월 1일이 됩니다. 또한 1월 23일을 반올림하면 2월 1일 된다.
- ❖ 다음은 입사일을 달을 기준으로 반올림한 예제

예

```
SELECT HIREDATE, ROUND (HIREDATE, 'MONTH')  
FROM EMP;
```

4.4 특정 기준으로 버리는 TRUNC 함수



- ❖ TRUNC 함수 역시 숫자를 잘라내는 것뿐만 아니라 날짜를 잘라낼 수 있다. ROUND 함수와 마찬가지로 포맷 형식을 주어 다양한 기준으로 날짜를 잘라낼 수 있다.

형식	TRUNC (<i>date, format</i>)
----	-------------------------------

- ❖ 특정 날짜(DATE)를 달(MONTH)을 기준으로 버림한 날짜를 구하기 위해서는 다음과 같이 표현

예	SELECT HIREDATE, TRUNC(HIREDATE, 'MONTH') FROM EMP;
---	--

- ❖ 위 예는 입사일을 월을 기준으로 절삭한 예제

4.5 두 날짜 사이 간격을 MONTHS_BETWEEN 함수



- ❖ MONTHS_BETWEEN 함수는 날짜와 날짜 사이의 개월 수를 구하는 함수

형식 MONTHS_BETWEEN (*date1*, *date2*)

- ❖ 다음은 각 직원들의 근무한 개월 수를 구하는 예제

예

```
SELECT ENAME, SYSDATE, HIREDATE,  
       MONTHS_BETWEEN (SYSDATE, HIREDATE)  
FROM EMP;
```


4.5 두 날짜 사이 간격을 MONTHS_BETWEEN 함수

- ❖ 이 결과 역시 소수점 이하 까지 구해줍니다. 소수점 이하자리는 한 달이 되지 못한 일수를 나타냅니다. 이를 TRUNC 함수를 사용하면 소수점 이하를 절삭할 수 있다.



예

```
SELECT ENAME, HIREDATE,  
       TRUNC( MONTHS_BETWEEN(SYSDATE, HIREDATE))  
FROM EMP;
```

```
C:\Windows\system32\CMD.exe - SQLPLUS scott/tiger

SQL> SELECT ENAME, HIREDATE,  
2          TRUNC( MONTHS_BETWEEN(SYSDATE, HIREDATE))  
3 FROM EMP;
```

ENAME	HIREDATE	TRUNC(MONTHS_BETWEEN(SYSDATE, HIREDATE))
SMITH	80/12/17	337
ALLEN	81/02/20	335
WARD	81/02/22	335
JONES	81/04/02	333
MARTIN	81/09/28	327
BLAKE	81/05/01	332

4.6 개월 수를 더하는 ADD_MONTHS 함수

- ❖ ADD_MONTHS 함수는 특정 개월 수를 더한 날짜를 구하는 함수



형식 ADD_MONTHS (*date*, *number*)

- ❖ 다음은 입사날짜에서 6개월을 추가하는 예제

예 SELECT ENAME, HIREDATE, ADD_MONTHS(HIREDATE, 6)
FROM EMP;

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT ENAME, HIREDATE, ADD_MONTHS(HIREDATE, 6)
2  FROM EMP;

ENAME      HIREDATE  ADD_MONT
-----
SMITH      80/12/17  81/06/17
ALLEN      81/02/20  81/08/20
WARD       81/02/22  81/08/22
JONES      81/04/02  81/10/02
MARTIN     81/09/28  82/03/28
BLAKE      81/05/01  81/11/01
```

4.7 해당 요일의 가장 가까운 날짜를 반환하는 NEXT_DAY 함수

- ❖ NEXT_DAY 함수는 해당 날짜를 기준으로 최초로 돌아오는 요일에 해당되는 날짜를 반환하는 함수



형식 NEXT_DAY (*date*, *요일*)

- ❖ 다음은 오늘을 기준으로 최초로 도래하는 수요일은 언제인지 알아보는 예제

예 SELECT SYSDATE, NEXT_DAY(SYSDATE, '수요일')
FROM DUAL;

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT SYSDATE, NEXT_DAY(SYSDATE, '수요일')
       2  FROM DUAL;

SYSDATE    NEXT_DAY
-----
09/01/17  09/01/21
```

4.8 해당 달의 마지막 날짜를 반환하는 LAST_DAY 함수



- ❖ LAST_DAY 함수는 해당 날짜가 속한 월의 마지막 날짜를 반환하는 함수
- ❖ 다음은 입사한 달의 마지막 날을 구하는 예제

예

```
SELECT HIREDATE, LAST_DAY(HIREDATE)
FROM EMP;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

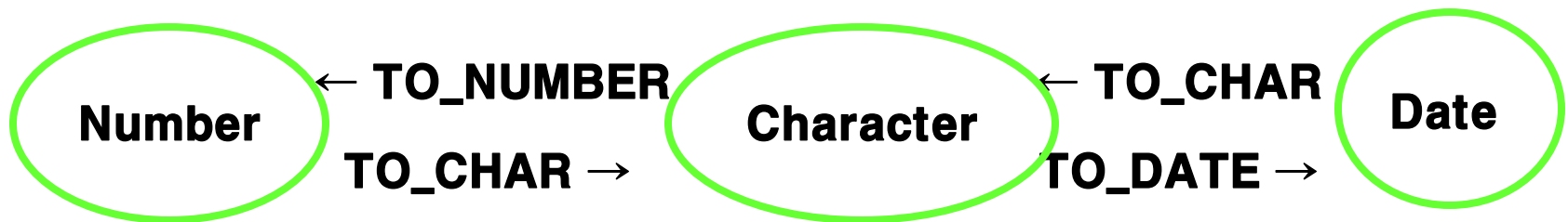
SQL> SELECT HIREDATE, LAST_DAY(HIREDATE)
2 FROM EMP;

HIREDATE LAST_DAY
-----
80/12/17 80/12/31
81/02/20 81/02/28
81/02/22 81/02/28
81/04/02 81/04/30
81/09/28 81/09/30
81/05/01 81/05/31
81/06/09 81/06/30
87/04/19 87/04/30
81/11/17 81/11/30
```

05. 형 변환 함수



- ❖ 숫자, 문자, 날짜의 데이터 형을 다른 데이터형으로 변환
- ❖ 이럴 때 사용하는 함수가 형 변환 함수이다. 형 변환 함수로는 TO_NUMBER, TO_CHAR, TO_DATE 가 있다.



구 분	설 명
TO_CHAR	날짜형 혹은 숫자형을 문자형으로 변환한다.
TO_DATE	문자형을 날짜형으로 변환한다.
TO_NUMBER	문자형을 숫자형으로 변환한다.

5.1 문자형으로 변환하는 TO_CHAR 함수

날짜형을 문자형으로 변환하기

- ❖ DATE 형태의 데이터를 지정한 양식에 의해 VARCHAR2 형의 문자로 변환한다.



형식 TO_CHAR (날짜 데이터, '출력형식')

- ❖ 다음은 날짜 출력 형식의 종류를 나열한 표이다.

종류	의미
YYYY	년도 표현(4자리)
YY	년도 표현(2자리)
MM	월을 숫자로 표현
MON	월을 알파벳으로 표현
DAY	요일 표현(수요일)
DY	요일을 약어로 표현(수)

5.1 문자형으로 변환하는 TO_CHAR 함수 날짜형을 문자형으로 변환하기

❖ 다음은 현재 날짜를 기본 형식과 다른 형태로 출력한 예



예

```
SELECT SYSDATE, TO_CHAR(SYSDATE, 'YYYY-MM-DD')  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT SYSDATE, TO_CHAR(SYSDATE, 'YYYY-MM-DD')  
2 FROM DUAL;  
  
SYSDATE    TO_CHAR(SY  
-----  
09/01/17   2009-01-17  
  
SQL>
```

5.1 문자형으로 변환하는 TO_CHAR 함수 날짜형을 문자형으로 변환하기

❖ 직원들의 입사일을 출력하되 요일까지 함께 출력해보자.



예

```
SELECT HIREDATE, TO_CHAR (HIREDATE, 'YYYY/MM/DD DAY')  
FROM EMP;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT HIREDATE, TO_CHAR (HIREDATE, 'YYYY/MM/DD DAY')  
2 FROM EMP;  
  
HIREDATE TO_CHAR(HIREDATE, 'YY  
-----  
80/12/17 1980/12/17 수요일  
81/02/20 1981/02/20 금요일  
81/02/22 1981/02/22 일요일  
81/04/02 1981/04/02 목요일  
81/09/28 1981/09/28 월요일  
81/05/01 1981/05/01 금요일  
81/06/09 1981/06/09 화요일
```


5.1 문자형으로 변환하는 TO_CHAR 함수 날짜형을 문자형으로 변환하기

- ❖ 년도를 2자리(YY)로 출력하고 월은 문자(MON)로 표시하고 요일을 약어(DY)로 표시한 예



예

```
SELECT HIREDATE, TO_CHAR (HIREDATE, 'YY/MON/DD DY')  
FROM EMP;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT HIREDATE, TO_CHAR (HIREDATE, 'YY/MON/DD DY')  
2 FROM EMP;  
  
HIREDATE TO_CHAR(HIREDATE, 'Y  
-----  
80/12/17 80/12월 /17 수  
81/02/20 81/2월 /20 금  
81/02/22 81/2월 /22 일  
81/04/02 81/4월 /02 목  
81/09/28 81/9월 /28 월  
81/05/01 81/5월 /01 금  
81/06/09 81/6월 /09 화
```

5.1 문자형으로 변환하는 TO_CHAR 함수 날짜형을 문자형으로 변환하기



❖ 다음은 시간 출력 형식의 종류를 나열한 표이다.

종류	의미
AM 또는 PM	오전(AM), 오후(PM) 시각 표시
A.M 또는 P.M	오전(A.M), 오후(P.M) 시각 표시
HH또는 HH12	시간(1~12)
HH24	24시간으로 표현(0~23)
MI	분 표현
SS	초 표현

5.1 문자형으로 변환하는 TO_CHAR 함수 날짜형을 문자형으로 변환하기

❖ 다음은 현재 날짜와 시간을 출력하는 예제



예

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

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT TO_CHAR(SYSDATE, 'YYYY/MM/DD, HH24:MI:SS')  
      2  FROM DUAL;  
  
TO_CHAR(SYSDATE, 'YYYY/MM/DD, HH24:MI:SS')  
-----  
2009/01/17, 17:07:13  
  
SQL>
```

5.1 문자형으로 변환하는 TO_CHAR 함수 숫자형을 문자형으로 변환하기

❖ 1230000이란 숫자를 문자 형태로 출력하는 예



예

```
SELECT TO_CHAR (1230000)
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT TO_CHAR (1230000)
2  FROM DUAL;

TO_CHAR
-----
1230000

SQL>
```

5.1 문자형으로 변환하는 TO_CHAR 함수 숫자형을 문자형으로 변환하기

❖ 다음 표는 숫자 출력 형식을 나열한 표



구 분	설 명
0	자릿수를 나타내며 자릿수가 맞지 않을 경우 0으로 채운다.
9	자릿수를 나타내며 자릿수가 맞지 않아도 채우지 않는다.
L	각 지역별 통화 기호를 앞에 표시한다.
.	소수점
,	천 단위 자리 구분

5.1 문자형으로 변환하는 TO_CHAR 함수 숫자형을 문자형으로 변환하기

- ❖ 각 지역별 통화 기호를 앞에 붙이고 천 단위마다 콤마를 붙여서 출력(예 : \1,230,000) 하려면 어떻게 해야 할까요? 위 표를 참조하여 다음과 같이 표현해야 한다.



예

```
SELECT ENAME, SAL, TO_CHAR (SAL, 'L999,999')  
FROM EMP;
```

ENAME	SAL	TO_CHAR(SAL, 'L999,999')
SMITH	800	₩800
ALLEN	1600	₩1,600
WARD	1250	₩1,250
JONES	2975	₩2,975
MARTIN	1250	₩1,250
BLAKE	2850	₩2,850
CLARK	2450	₩2,450
SCOTT	3000	₩3,000
KING	5000	₩5,000
TURNER	1500	₩1,500

5.1 문자형으로 변환하는 TO_CHAR 함수 숫자형을 문자형으로 변환하기

- ❖ 9는 자릿수를 나타내며 자릿수가 맞지 않으면 채우지 않습니다. 하지만 0은 자릿수를 나타내며 자릿수가 맞지 않을 경우 0으로 채운다.



예

```
SELECT TO_CHAR (123456, '000000000'),  
       TO_CHAR (123456, '999,999,999')  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT TO_CHAR (123456, '000000000'), TO_CHAR (123456, '999,999,999')
       2  FROM DUAL;

TO_CHAR(12  TO_CHAR(1234
-----
000123456      123,456

SQL>
```

5.2 날짜형으로 변환하는 TO_DATE 함수



- ❖ 날짜 형으로 변환하기에 앞서 날짜 형에 대해서 살펴보도록 한다. 날짜 형은 세기, 년도, 월, 일. 시간, 분. 초와 같이 날짜와 시간에 대한 정보를 저장한다.
- ❖ 오라클에서 기본 날짜 형식은 'YY/MM/DD' 형식으로 '년/월/일' 예를 들면 '06/03/08' 식으로 나타낸다.
- ❖ 만일 년도를 4자리로 출력하려면 'YYYY/MM/DD' 형식으로 지정한다.
- ❖ TO_DATE 함수는 문자열을 날짜 형으로 변환한다.

형식	TO_DATE('문자' , 'format')
----	----------------------------

5.2 날짜형으로 변환하는 TO_DATE 함수

- ❖ 숫자형태인 19810220 을 TO_DATE 함수를 사용해서 날짜형으로 변환합니다. 이때 두 번째 인자는 출력할 형식을 지정해 주어야 한다.



예

```
SELECT ENAME, HIREDATE FROM EMP  
WHERE HIREDATE=TO_DATE(19810220,'YYYYMMDD')
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT ENAME, HIREDATE FROM EMP  
2  WHERE HIREDATE=TO_DATE(19810220,'YYYYMMDD');  
  
ENAME      HIREDATE  
-----  
ALLEN      81/02/20  
  
SQL>
```

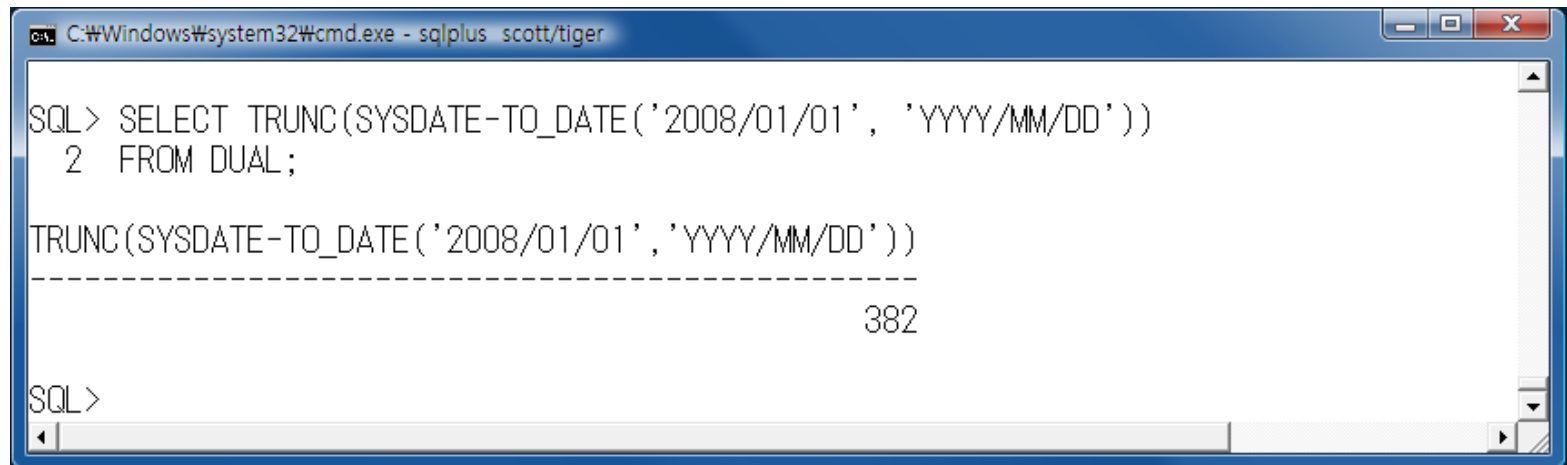
5.2 날짜형으로 변환하는 TO_DATE 함수

- ❖ 다음은 올해 며칠이 지났는지 현재 날짜에서 2008/01/01을 뺀 결과를 출력하는 예제이다.



예

```
SELECT TRUNC(SYSDATE-TO_DATE('2008/01/01', 'YYYY/MM/DD'))  
FROM DUAL;
```



```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT TRUNC(SYSDATE-TO_DATE('2008/01/01', 'YYYY/MM/DD'))  
2 FROM DUAL;  
  
TRUNC(SYSDATE-TO_DATE('2008/01/01', 'YYYY/MM/DD'))  
-----  
382  
  
SQL>
```

5.3 숫자형으로 변환하는 TO_NUMBER 함수



- ❖ TO_NUMBER 함수는 특정 데이터를 숫자형으로 변환
- ❖ 다음과 같이 '20,000'을 '10,000'의 차이를 알아보기 위해서 빼기를 해 보자. 산술 연산을 하려면 문자형을 숫자형으로 변환한 후에 실행해야 한다.

예

```
SELECT TO_NUMBER('20,000', '99,999') - TO_NUMBER('10,000', '99,999')  
FROM DUAL;
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger  
  
SQL> SELECT TO_NUMBER('20,000', '99,999') - TO_NUMBER('10,000', '99,999')  
2 FROM DUAL;  
  
TO_NUMBER('20,000', '99,999')-TO_NUMBER('10,000', '99,999')  
-----  
10000  
  
SQL>
```