

## 오라클을 이용한 DB 구축 초급

### 강의 노트

#### 단일 행 함수

## Lesson 1. SQL 함수와 단일 행 함수

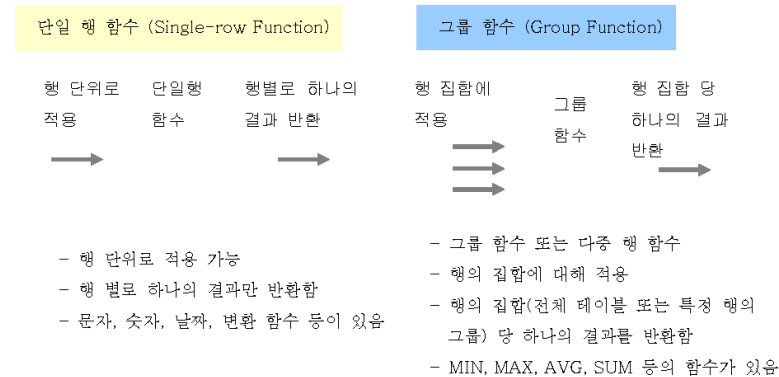
### SQL 함수와 단일 행 함수

1. SQL 함수의 두 가지 유형
2. 단일 행 함수

#### 1. SQL 함수의 두 가지 유형

##### (1) SQL 함수란?

→ 입력된 인수(Argument)를 처리해서 결과 값을 반환하는 함수로, 크게 단일 행 함수와 그룹 함수(또는 다중 행 함수 : Multiple-row Function)로 구분할 수 있다.



#### 2. 단일 행 함수

##### (1) 단일 행 함수의 정의

→ 행 단위로 적용 가능하고, 행별로 하나의 결과만 반환하는 함수이다.

##### (2) 형식

**함수이름 (열\_이름 | 표현식 , [인수1, 인수2, ...])**

- 표현식(Expression) : 문자열이나 산술식을 의미한다.
- 인수(Argument) : 상수 값, 열 이름, 표현식 등이 있다.

### (3) 단일 행 함수의 특징

1. 인수로 전달된 데이터 값을 조작 또는 변환한다.
2. 질의에서 반환되는 각각의 행에 대해 함수가 수행된다.
3. 하나 이상의 인수가 필요하다.
4. 행별로 하나의 결과만 반환한다.
5. SELECT, WHERE, ORDER BY 절에서 사용할 수 있다.
6. 중첩해서 사용 가능하다. 즉, 단일 행 함수가 다른 함수의 인수가 될 수 있다.

### (4) 단일 행 함수의 종류

→ 문자, 숫자, 날짜 관련 함수 및 데이터 형 변환 함수와 일반 함수 등이 있다.

함수 종류	특징
문자 함수	문자를 인수로 입력 받고, 문자나 숫자를 반환한다.
숫자 함수	숫자를 인수로 입력 받고, 숫자를 반환한다.
날짜 함수	MONTHS_BETWEEN 함수만 숫자를 반환하고, 나머지 함수는 모두 날짜 데이터를 반환한다.
데이터 형 변환 함수	어떤 데이터 형의 값을 다른 데이터 형으로 변환한다.
일반 함수	NVL, DECODE 함수 등이 있다.

### Lesson 3. 데이터 형 변환 함수와 일반 함수

데이터 형 변환 함수와 일반 함수

1. 데이터 형 변환
2. 데이터 형 변환 함수
3. 일반 함수
4. 중첩 함수

#### 1. 데이터 형 변환

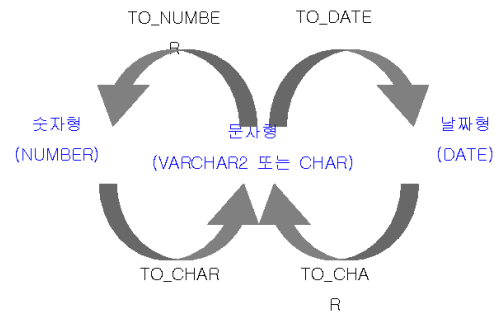
(1) 데이터 형 변환의 구분

##### 암시적 데이터 형 변환

- 열에 값을 할당할 때 할당되는 값의 형에 따라, 오라클이 자동으로 열의 형을 변환하는 것을 의미
- VARCHAR2 또는 CHAR 형의 열에 날짜 값(YY/MM/DDY 형식의)이 할당되면, DATE 형으로 자동 변환됨
- VARCHAR2 또는 CHAR 형의 열에 숫자 값이 할당되면, NUMBER 형으로 자동 형 변환됨
- 암시적 형 변환이 가능한 경우라도, 성능을 위해서 열의 데이터 형과 데이터 값의 형을 일치시키는 것을 권장

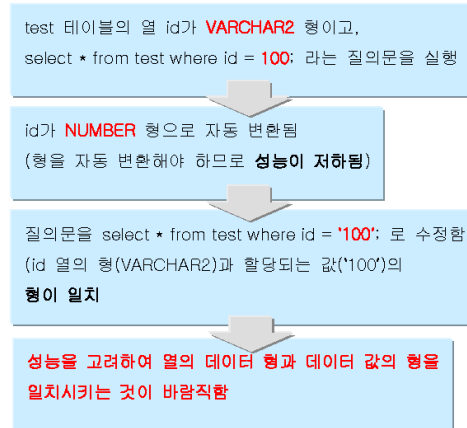
##### 명시적 데이터 형 변환

- 데이터 형 변환 함수를 이용하여 사용자가 명시적으로 형 변환을 지정



### ☞ 참고 : 암시적 데이터형 변환과 성능과의 관계

암시적 데이터형 변환은 성능에 나쁜 영향을 미치게 되므로 열의 원래 데이터 형과 데이터 값의 형을 일치시키는 것이 바람직하다.



## 2. 데이터 형 변환 함수

(1) 데이터 형 변환 함수의 종류와 기능

→ 명시적 형 변환을 위해서 TO\_CHAR, TO\_DATE, TO\_NUMBER 등과 같은 함수가 있다.

TO_CHAR(날짜 숫자[, 'fmt'])	숫자나 날짜를 나타내는 문자 값을 지정된 포맷(fmt)에 따라 VARCHAR2 형의 문자열로 변환함
TO_NUMBER(문자열)	숫자 값을 갖는 문자열을 숫자 형으로 변환함
TO_DATE(문자열[, 'fmt'])	날짜를 나타내는 문자열을 명시된 포맷(fmt)에 따라 날짜형으로 변환함 (포맷을 생략하면 YY/MM/DD 형식)

(2) 날짜 변환 TO\_CHAR 함수

→ 날짜를 명시한 포맷 형태를 갖는 VARCHAR2 형의 문자열로 변환한다.

TO\_CHAR (날짜, '포맷')

포맷 기술 방법

포맷은 단일 인용부호로 묶어서 기술해야 한다.

대소문자를 구분한다. 즉, 포맷 요소를 대문자로 기술하면 대문자로 출력되고, 소문자로 기술하면 소문자로 출력된다.

포맷에 문자를 추가할 때는 이중 인용부호로 묶어서 기술해야 한다. 예) to\_char(hiredate, YY"년" MM"월" DD"일") :

추가된 공백을 제거하거나 앞부분의 무의미한 숫자 0을 제거하기 위해 fm(또는 FM) 요소를 사용한다. 단, fm이 지정된 이후의 모든 포맷 요소에 적용된다.

날짜 형식 포맷 요소

포맷 요소	기능
YYYY	4자리 숫자로 년도 표시
YY	마지막 2자리 숫자로 년도 표시
YEAR	전체 문자로 년도 표시
MM	2자리 숫자로 월 표시
MONTH	최대 9자리 문자로 전체 월 이름 표시
MON	3자리 약자로 월 이름 표시
DY	3자리 약자로 요일 이름 표시
DAY	최대 9자리로 전체 요일 이름 표시
DD	2자리 숫자로 일자 표시

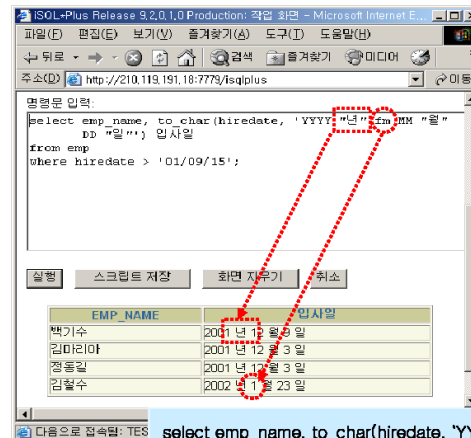
일자 출력용 접미사

- 일을 나타내는 숫자를 문자열로 변환하기 위한 여러 가지 접미사가 있다.
- 접미사 자체는 대소문자를 구분하지 않으며, 일자 포맷 요소(DD 또는 dd)의 대소문자 지정에 따라 대문자 또는 소문자로 표시된다.

접미사 요소	기능
SPTH 또는 THSP	영문 서수 형태로 숫자를 표시
SP	영문 숫자로 숫자를 표시
TH	숫자 뒤에 TH를 붙여서 서수 형태로 표시

1) TO\_CHAR 함수 활용의 예

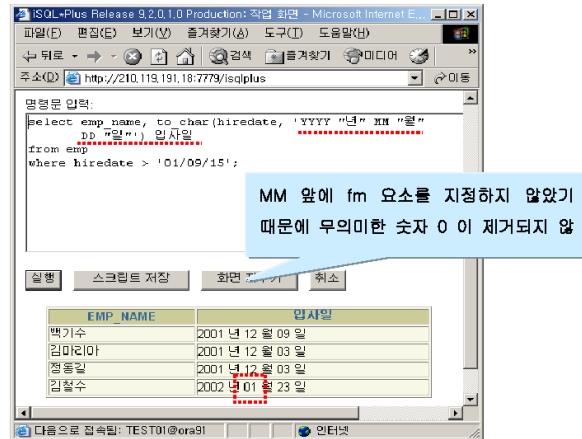
- 2001년 9월 15일 이후에 입사한 모든 사원의 이름과 입사일을 출력한다.



```
select emp_name, to_char(hiredate, 'YYYY "년" fm MM "월"
DD "일"' ) 입사일
from emp
where hiredate > '01/09/15' ;
```

### ☞ 참고 : fm요소의 역할

fm 요소를 지정하지 않은 경우, '1월' 대신에 '01월'이라고 출력된 것을 알 수 있다.  
일반적으로 우리가 이렇게 표기하지 않으므로 검색 결과도 역시 유효 숫자 앞의 무의미한 '0'이 제거될 수 있도록 질의문을 작성하는 것이 좋다.



### (3) 숫자 변환 TO\_CHAR 함수

→ 숫자를 명시한 포맷 형태의 문자열로 변환한다.

TO\_CHAR (숫자, '포맷')

#### 1) 숫자 포맷 요소

→ 숫자를 문자열로 변환하는데 9, 0, \$, L 등 여러 가지 유형의 포맷 요소가 사용된다.

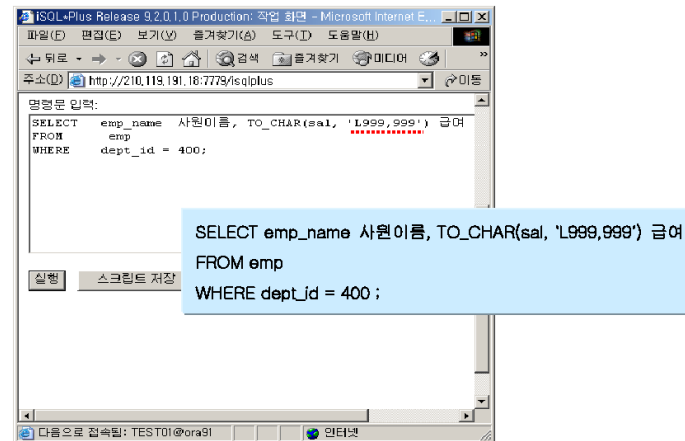
→ 포맷에 지정된 자릿수를 초과하는 '#' 기호를 표시한다. 단, 소수점 이하 값은 반올림 처리한다.

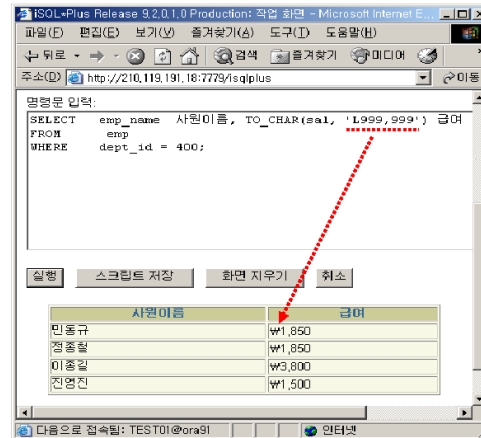


포맷 요소	기능
9	유효 숫자만 표시
0	유효 숫자 앞에 빈 칸을 숫자 0으로 채움
\$	유효 숫자 바로 앞에 달러 기호를 하나 표시
L	유효 숫자 바로 앞에 지역 화폐 단위를 표시
.	명시한 위치에 소수점을 표시
,	명시한 위치에 콤마를 표시

## 2) 숫자 변환 예제

→ 급여 앞에 화폐 단위 (WON)을 출력하고, 세 자리 수마다 콤마를 출력한다.





#### (4) TO\_DATE 함수

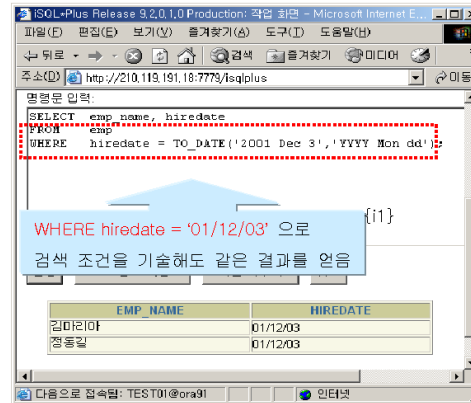
→ 문자열을 명시된 포맷 형태를 참조하여 디폴트 형식(YY/MM/DD)의 날짜로 변환한다.

##### 1) 유의사항

1. 명시된 포맷 형태의 날짜로 변환하는 것이 아니라, 명시된 포맷을 참조하여 문자열이 어떤 날짜 값인지를 인식한 다음, 그 값을 디폴트 형식의 날짜 값으로 변환한다.
2. 즉, 명시된 포맷은 문자열이 어떤 형식인지를 오라클 DBMS에게 알려주는 역할을 한다.
3. 변환되는 디폴트 형식은 YY/MM/DD이다.
4. 포맷 지정에 사용되는 포맷 요소는 TO\_CHAR(날짜, 'fmt') 함수의 포맷 요소 동일하다
5. 오라클에서 날짜 값을 입력할 때는 반드시 TO\_DATE 함수를 사용해야 한다.

##### 2) 날짜 변환 예제

→ 2001년 12월 3일에 입사한 모든 사원의 이름과 입사일을 표시한다.



#### (5) TO\_NUMBER 함수

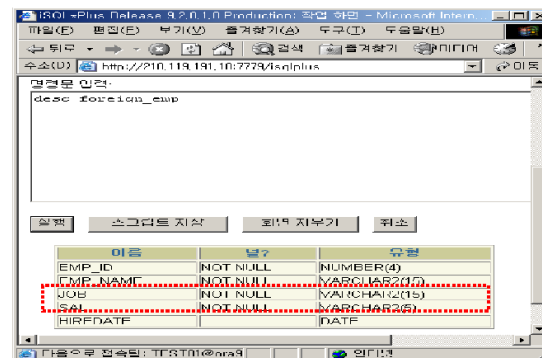
##### 1) TO\_NUMBER 함수란?

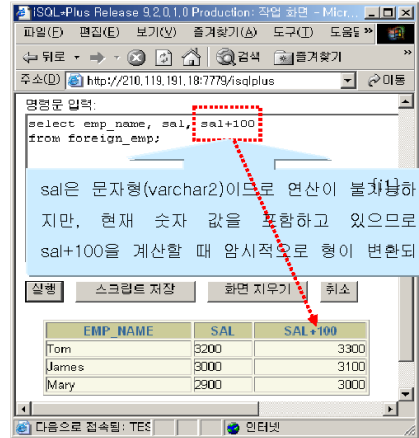
→ 숫자 값을 갖고 있는 문자열 인수를 연산이 가능한 숫자 형으로 변환한다.

TO\_NUMBER (문자열)

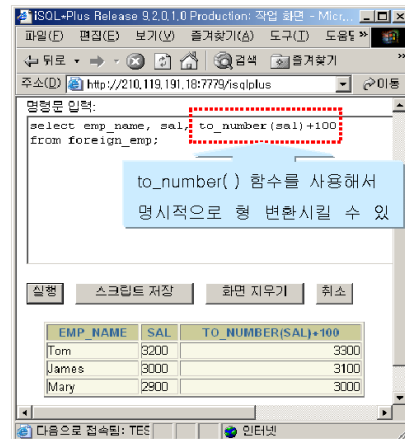
##### 2) TO\_NUMBER 함수 예제

→ foreign\_emp 테이블의 모든 사원의 이름과 급여와 급여에 100을 더한 값을 출력한다. 확인





[단계 2] → 사원이름과 급여 및 급여에 100을 더한 값을 출력



[단계 3]

→ sal에 100을 더한 값을 계산할 때 명시적으로 형이 변환되도록 작성

### 3. 일반 함수

#### (1) NVL( ) 함수

→ 널(Null) 값을 다른 값으로 변환시키기 위해 사용할 수 있게끔 하는 함수이다.

##### 1) 형식

→ expr1 에 포함된 널 값을 실제 값 expr2로 변환한다.

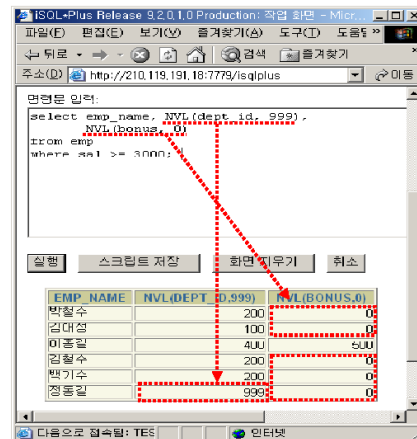
→ expr1과 expr2의 데이터 형이 반드시 일치해야 한다

##### 2) 사용 가능한 데이터 형은 숫자, 날짜, 문자 형이다.

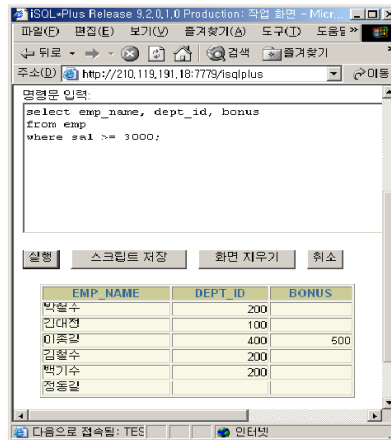
데이터 형
NUMBER
DATE
CHAR 또는 VARCHAR2

#### 3) 널 값 변환 예

→ 급여가 3000 이상인 모든 사원의 이름과 부서번호와 보너스를 출력하  
되, 부서번호가 없는 경우 999로 표시하고, 보너스가 없는 경우 0으로 출  
력한다.

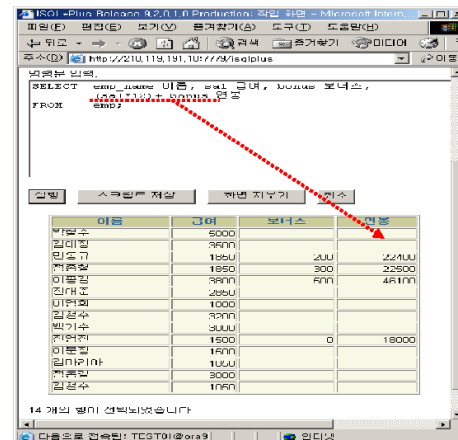


- ☞ 참고 : NVL 함수를 사용하지 않은 경우  
 NVL 함수를 사용하지 않으면 널 값을 갖는 경우 모두 공백으로 출력하게 된다. 따라서 용도에 맞게 널 값을 변환하도록 NVL 함수를 잘 활용할 필요가 있다.



4) 널 값을 포함한 연산 예

- 모든 사원의 이름과 급여와 보너스 및 연봉을 출력한다.  
 (단, 연봉은 (급여\*12)+ 보너스로 계산)



## ☞ 참고 : NVL 함수를 활용한 경우

보너스 값이 널인 경우에도 연봉이 제대로 계산될 수 있도록, NVL 함수를 사용해서 bonus가 널 값을 갖는 경우 숫자 영으로 변환하도록 산술식을 수정할 필요가 있다.

SELECT emp\_name 이름, sal 급여, bonus 보너스,  
(sal\*12) + NVL(bonus, 0) 연봉  
FROM emp;

이름	급여	보너스	연봉
박정수	5000		60000
김미영	3600		43200
박영규	1000	200	12200
전영철	1850	300	22500
이영철	3800	500	46100
전영철	2000		24000
민영희	1000		12000
김철수	3200		38400
박영규	1000		12000
전영철	1500	0	18000
이영철	1600		19200
김미영	1000		12000
전영철	2000		24000
김철수	1050		12600

14 개의 행이 선택되었습니다.

## (2) DECODE 함수

→ 조건의 만족 여부에 따라 다른 값을 출력하는 함수로, 프로그래밍 언어의 IF문이나 CASE문과 유사하게 조건적 검색을 가능하게 한다.

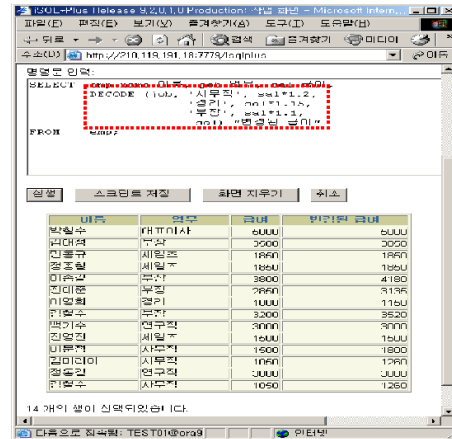
### 1) 형식

- 열 이름이나 표현식이 비교 1 과 같으면 반환하고, 비교 2 와 같으면 결과 2 를 반환하고, 어떤 비교와도 같지 않으면 디폴트 값을 반환한다.
- 디폴트 값을 생략하면, 널 값을 반환한다.

DECODE(열\_이름|표현식, 비교1, 결과1  
[, 비교2, 결과2, ...]  
[, 디폴트 값])

### 2) 조건적 검색 예

→ 사원이름과 업무, 급여를 출력하되, 급여는 다음과 같이 각 사원의 업무에 따라 차등화 된 인상률에 의해 계산되도록 질의문을 작성한다.



#### 4. 중첩 함수

##### (1) 중첩 함수

###### 1) 중첩 함수란?

- 단일 행 함수는 여러 레벤에 걸쳐서 중첩해서 사용할 수 있다.
- 중첩 함수는 가장 내부 함수에서 가장 외부 함수 순으로 계산된다.

**F3 (F2 (F1 (월\_이름, 인수1), 인수2), 인수 3)**

###### 2) 함수 중첩 사용 예

- 모든 사원의 이름과 자신의 부서장의 사원번호를 출력하되, 부서장이 없는 사원의 경우, 부서장 열에 "해당 없음"이라고 출력한다.

1) NUMBER 형인 mgr\_id 열을 "해당 없음"과 형을 일치시키기 위해서 TO\_CHAR 함수 사용

2) mgr\_id가 널인 경우 "해당 없음"으로 변환하기 위해서 NVL 함수 사용



## &lt;학습 정리&gt;

## ◆ 단일 행 함수의 종류와 특징

→ 문자, 숫자, 날짜 관련 함수 및 데이터형 변환 함수와 일반 함수 등이 있다.

함수 종류	특징
문자 함수	문자를 인수로 입력 받고, 문자나 숫자를 반환한다.
숫자 함수	숫자를 인수로 입력 받고, 숫자를 반환한다.
날짜 함수	MONTHS_BETWEEN 함수만 숫자를 반환하고, 나머지 함수는 모두 날짜 데이터를 반환한다.
데이터형 변환 함수	어떤 데이터 형의 값을 다른 데이터 형으로 변환한다.
일반 함수	NVL, DECODE 함수 등이 있다.

## ◆ 문자, 숫자, 날짜 값 조작 단일 행 함수들

→ 다음과 같이 문자와 숫자, 날짜 값을 조작하는 여러 가지 단일 행 함수가 있다.

함수 분류	함수 종류
문자 조작 함수	CONCAT(열이름1 표현식1, 열이름2 표현식2)
	SUBSTR(열이름 표현식, m[, n])
	LENGTH(열이름 표현식)
	INSTR(열이름 표현식, 문자)
	LPAD((열이름 표현식, n, ' 문자')

함수 분류	함수 종류
숫자 함수	ROUND(열이름 표현식, n)
	TRUNC(열이름 표현식, n)
	MOD(m, n)

함수 분류	함수 종류
날짜 함수	SYSDATE
	MONTHS_BETWEEN(날짜1, 날짜2)
	ADD_MONTHS(날짜, n)
	NEXT_DAY(날짜, '문자')
	LAST_DAY(날짜)

◆ 데이터 형 변환 함수들

날짜를 문자열로 변환하기 : TO_CHAR 함수	정의	날짜를 명시한 포맷 형태를 갖는 VARCHAR2 형의 문자열로 변환한다.
	형식	TO_CHAR(날짜, '포맷')
문자열을 날짜로 변환하기 : TO_DATE 함수	정의	문자열을 명시된 포맷 형태를 참조하여 디폴트 형식(YY/MM/DD)의 날짜로 변환한다.
	형식	TO_DATE(문자열, '포맷')

◆ NVL 함수

NVL 함수	정의	다음과 같은 형식으로, <code>expr1</code> 에 포함된 널 값을 실제 값 <code>expr2</code> 로 변환한다.
	형식	<code>NVL (expr1, expr2)</code>