

함수





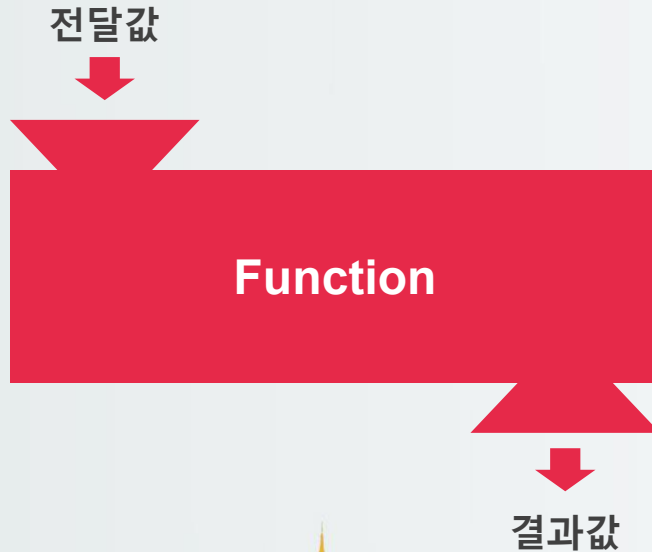
함수



함수

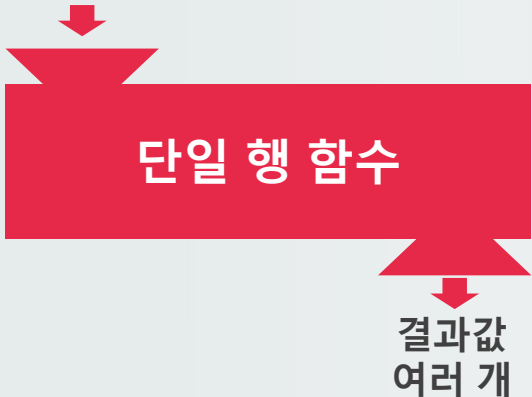

함수

- 하나의 큰 프로그램에서 반복적으로 사용되는 부분들을 분리하여 작성해 놓은 작은 서브프로그램
- 호출하며 값을 전달하면 수행 결과를 리턴 하는 방식으로 사용



함수

함수의 유형

<p>여러 개 값 전달</p>  <p>단일 행 함수</p> <p>결과값 여러 개</p>	<p>여러 개 값 전달</p>  <p>그룹 함수</p> <p>결과값 1개</p>
단일 행 함수	그룹 함수
각 행마다 반복적으로 적용 입력 받은 행 개수 만큼 결과 반환	특정한 행들의 집합으로 그룹을 형성 그룹당 1개의 결과를 반환



문자 처리 함수



문자 처리 함수

구분	입력값 타입	리턴 타입	설명
LENGTH	CHARACTER	NUMBER	문자열의 길이를 반환
LENGTHB			문자열의 바이트 크기를 반환
INSTR			특정 문자의 위치를 반환
INSTRB			특정 문자의 위치 바이트 크기를 반환



문자 처리 함수

구분	입력값 타입	리턴 타입	설명
LPAD	CHARACTER	CHARACTER	문자열을 지정된 숫자만큼의 크기로 설정하고, 지정한 문자를 왼쪽부터 채워서 생성된 문자열 리턴
RPAD			문자열을 지정된 숫자만큼의 크기로 설정하고, 지정한 문자를 오른쪽부터 채워서 생성된 문자열 리턴
LTRIM			왼쪽부터 지정된 문자를 잘라내고 남은 문자를 리턴
RTRIM			오른쪽부터 지정된 문자를 잘라내고 남은 문자를 리턴
TRIM			왼쪽/오른쪽/양쪽부터 지정한 문자를 잘라내고 남은 문자 리턴



문자 처리 함수

구분	입력값 타입	리턴 타입	설명
SUBSTR	CHARACTER	CHARACTER	지정한 위치에서 지정한 길이만큼 문자를 잘라내어 리턴
SUBSTRB			지정한 위치에서 지정한 바이트만큼 문자를 잘라내어 리턴
LOWER			전달받은 문자/문자열을 소문자로 변환하여 리턴
UPPER			전달받은 문자/문자열을 대문자로 변환하여 리턴
INITCAP			전달받은 문자/문자열을 첫글자를 대문자로, 나머지를 소문자로 변환하여 리턴
CONCAT			인자로 전달받은 두 개의 문자/문자열을 합쳐서 리턴
REPLACE			전달받은 문자열 중 지정된 문자를 인자로 받은 문자로 변경하여 리턴



문자 처리 함수

문자 처리 함수 – LENGTH

주어진 컬럼 값/문자열의 길이(문자 개수)를 반환하는 함수

작성법	리턴 타입	파라미터
LENGTH(CHAR STRING)	NUMBER	CHARACTER 타입의 컬럼 또는 문자열

```
SELECT  EMP_NAME,  
        EMAIL,  
        LENGTH(EMAIL)  
FROM EMPLOYEE;
```

EMP_N...	EMAIL	LENGTH(EMAIL)
선동일	sun di@kh.or.kr	15
송송기	song jk@kh.or.kr	16
노호절	no hc@kh.or.kr	14
송은희	song eh@kh.or.kr	16
유재식	yoo js@kh.or.kr	15
정송하	jung jh@kh.or.kr	16
박나라	pack nr@kh.or.kr	16
하이유	ha iy@kh.or.kr	14
김해솔	kim hs@kh.or.kr	15
심보선	sim bs@kh.or.kr	15
윤은혜	youn eh@kh.or.kr	16
전형노	jun hd@kh.or.kr	15
장쯔위	jang zw@kh.or.kr	16
하동운	ha dh@kh.or.kr	14



문자 처리 함수

문자 처리 함수 – LENGTHB

주어진 컬럼 값/문자열의 길이(BYTE)를 반환하는 함수

작성법	리턴 타입	파라미터
LENGTHB(CHAR STRING)	NUMBER	CHARACTER 타입의 컬럼 또는 문자열

```
SELECT EMP_NAME,  
       LENGTH (EMP_NAME),  
       LENGTHB(EMP_NAME)  
FROM EMPLOYEE;
```

EMP_N...	LENGTH(EMP_NAME)	LENGTHB(EMP_NAME)
서인영	3	9
송기철	3	9
송영희	3	9
송재식	3	9
유재하	3	9
정송하	3	9
박나라	3	9
하이유	3	9
김해성	3	9
심봉성	3	9
윤영애	3	9
조현우	3	9
장원우	3	9
하노우	3	9

문자 처리 함수

문자 처리 함수 – INSTR

찾는 문자(열)이 지정한 위치부터 지정한 회수만큼 나타난 시작위치를 반환

작성법	리턴 타입
INSTR(String,STR,[POSITION],[OCCURRENCE]])	NUMBER

파라미터	설명
STRING	문자 타입 컬럼 또는 문자열
STR	찾으려는 문자(열)
POSITION	찾을 위치 시작 값(기본값 1) POSITION > 0 : STRING의 시작부터 끝방향으로 찾음 POSITION < 0 : STRING의 끝부터 시작방향으로 찾음
OCCURRENCE	찾으려는 문자열이 반복될 때, 몇 번째 값인지 지정하는 숫자(기본값 1)



문자 처리 함수

문자 처리 함수 – INSTR

```
SELECT  EMAIL,  
        INSTR(EMAIL, '@',1,1)
```

```
FROM EMPLOYEE;
```

또는

```
SELECT  EMAIL,  
        INSTR(EMAIL, '@',-1,1)
```

```
FROM EMPLOYEE;
```

※ 부호를 통해 찾는 방향을 지정하더라도
앞에서 부터 몇 번째에 있는지가 결과로
나오기 때문에 결과는 동일

EMAIL	INSTR(EMAIL, '@',-1,1)
sun di@kh.or.kr	7
song jk@kh.or.kr	8
no hc@kh.or.kr	6
song eh@kh.or.kr	8
yoo js@kh.or.kr	7
jung jh@kh.or.kr	8
pack nr@kh.or.kr	8
ha iy@kh.or.kr	6
kim hs@kh.or.kr	7
sim bs@kh.or.kr	7
youn eh@kh.or.kr	8
jun hd@kh.or.kr	7
jang zw@kh.or.kr	8
ha dh@kh.or.kr	6

문자 처리 함수

문자 처리 함수 – LPAD/RPAD

주어진 컬럼 문자열에 임의의 문자열을 왼쪽/오른쪽에 덧붙여 길이 N의 문자열을 반환하는 함수

작성법	리턴 타입
LPAD(String,N,[STR])/RPAD(String,N,[STR])	CHARACTER

파라미터	설명
STRING	문자 타입 컬럼 또는 문자열
N	반환한 문자열의 길이(byte) 원래 길이보다 작다면 N만큼 잘라서 표시
STR	덧 붙이려는 문자(열), 생략 시 공백 문자



문자 처리 함수

문자 처리 함수 – LPAD/RPAD

```
SELECT LPAD(EMAIL, 20, '#')  
FROM EMPLOYEE;
```

LPAD(EMAIL,20,'#')
#####sun di@kh.or.kr
#####song jk@kh.or.kr
#####no hc@kh.or.kr
#####song eh@kh.or.kr
#####yoo js@kh.or.kr
#####jung jh@kh.or.kr
#####pack nr@kh.or.kr
#####ha iy@kh.or.kr
#####kim hs@kh.or.kr
#####sim bs@kh.or.kr
#####youn eh@kh.or.kr
#####jun hd@kh.or.kr
#####jang zw@kh.or.kr
#####ha dh@kh.or.kr

```
SELECT RPAD(EMAIL, 20, '#')  
FROM EMPLOYEE;
```

RPAD(EMAIL,20,'#')
sun di@kh.or.kr#####
song jk@kh.or.kr#####
no hc@kh.or.kr#####
song eh@kh.or.kr#####
yoo js@kh.or.kr#####
jung jh@kh.or.kr#####
pack nr@kh.or.kr#####
ha iy@kh.or.kr#####
kim hs@kh.or.kr#####
sim bs@kh.or.kr#####
youn eh@kh.or.kr#####
jun hd@kh.or.kr#####
jang zw@kh.or.kr#####
ha dh@kh.or.kr#####



문자 처리 함수

문자 처리 함수 – LTRIM/RTRIM

주어진 컬럼 문자열의 왼쪽 혹은 오른쪽에서 지정한 STR에 포함된 모든 문자를 제거한 나머지를 반환

작성법	리턴 타입
LTRIM(String,STR)/RTRIM(String,STR)	CHARACTER

파라미터	설명
STRING	문자 타입 컬럼 또는 문자열
STR	제거 하려는 문자(열), 생략 시 공백 문자



문자 처리 함수

문자 처리 함수 – LTRIM

수행 문장	결과
SELECT LTRIM(' KH') FROM DUAL;	KH
SELECT LTRIM(' KH', ' ') FROM DUAL;	KH
SELECT LTRIM('000123456', '0') FROM DUAL;	123456
SELECT LTRIM('123123KH', '123') FROM DUAL;	KH
SELECT LTRIM('123123KH123', '123') FROM DUAL;	KH123
SELECT LTRIM('ACABACCKH', 'ABC') FROM DUAL;	KH
SELECT LTRIM('57812KH', '0123456789') FROM DUAL;	KH



문자 처리 함수

문자 처리 함수 – RTRIM

수행 문장	결과
SELECT RTRIM('KH ') FROM DUAL;	KH
SELECT RTRIM('KH ', ' ') FROM DUAL;	KH
SELECT RTRIM('123456000', '0') FROM DUAL;	123456
SELECT RTRIM('KH123123', '123') FROM DUAL;	KH
SELECT RTRIM('123KH123123', '123') FROM DUAL;	123KH
SELECT RTRIM('KHACABACC', 'ABC') FROM DUAL;	KH
SELECT RTRIM('KH57812', '0123456789') FROM DUAL;	KH



문자 처리 함수

문자 처리 함수 – TRIM

주어진 컬럼 문자열의 앞/뒤/양쪽에 있는 지정한 문자를 제거한 나머지를 반환

작성법	리턴 타입
TRIM(String) TRIM(Char From String) TRIM(Leading Trailing Both [Char] From String)	Character

파라미터	설명
String	문자 타입 컬럼 또는 문자열
Char	제거 하려는 문자, 생략 시 공백 문자
Leading	TRIM할 위치를 지정 앞(Leading) / 뒤(Trailing) / 양쪽(Both) 지정 가능. 기본값 양쪽



문자 처리 함수

문자 처리 함수 – TRIM

수행 문장	결과
SELECT TRIM(' KH ') FROM DUAL;	KH
SELECT TRIM('Z' FROM 'ZZZKHZZZ') FROM DUAL;	KH
SELECT TRIM(LEADING 'Z' FROM 'ZZZ123456') FROM DUAL;	123456



문자 처리 함수

문자 처리 함수 – SUBSTR

컬럼이나 문자열에서 지정한 위치부터 지정한 개수의 문자열을 잘라내어 리턴

작성법	리턴 타입
SUBSTR(String, POSITION,[LENGTH])	CHARACTER

파라미터	설명
STRING	문자 타입 컬럼 또는 문자열
POSITION	문자열을 잘라낼 위치 양수이면 시작방향에서 지정한 수만큼, 음수이면 끝방향에서 지정한 수만큼의 위치를 의미
LENGTH	반환한 문자의 개수를 의미 생략하는 경우 끝까지를 의미함, 음수이면 NULL을 리턴



문자 처리 함수

문자 처리 함수 – SUBSTR

수행 문장	결과
SELECT SUBSTR('SHOWMETHEMONEY',5,2) FROM DUAL;	ME
SELECT SUBSTR('SHOWMETHEMONEY',7) FROM DUAL;	THEMONEY
SELECT SUBSTR('SHOWMETHEMONEY',1,6) FROM DUAL;	SHOWME
SELECT SUBSTR('SHOWMETHEMONEY',-8,3) FROM DUAL;	THE
SELECT SUBSTR('SHOWMETHEMONEY',-10,2) FROM DUAL;	ME
SELECT SUBSTR('쇼우 미 더 머니',2,5) FROM DUAL;	우 미 더



문자 처리 함수

문자 처리 함수 – LOWER/UPPER/INITCAP

컬럼이나 문자열을 대문자 혹은 소문자로 변환하여 출력

작성법	리턴 타입
LOWER(String) / UPPER(String) / LOWER(String)	CHARACTER

파라미터	설명
String	문자 타입 컬럼 또는 문자열



문자 처리 함수

문자 처리 함수 – LOWER/UPPER/INITCAP

수행 문장	결과
SELECT LOWER('Welcome To My World') FROM DUAL;	welcome to my world
SELECT UPPER('Welcome To My World') FROM DUAL;	WELCOME TO MY WORLD
SELECT INITCAP('welcome to my world') FROM DUAL;	Welcome To My World



문자 처리 함수

문자 처리 함수 – CONCAT

컬럼의 문자 혹은 문자열을 두 개 전달 받아 하나로 합친 후 리턴

작성법	리턴 타입	파라미터
CONCAT(String,String)	CHARACTER	문자 타입 컬럼 또는 문자열

수행 문장	결과
SELECT CONCAT('가나다라', 'ABCD') FROM DUAL;	가나다라ABCD
SELECT '가나다라' 'ABCD' FROM DUAL;	가나다라ABCD



문자 처리 함수

문자 처리 함수 – REPLACE

전달받은 문자열 중 지정한 문자를 전달받은 문자로 변환하여 리턴

작성법	리턴 타입
REPLACE(String,STR1,STR2)	CHARACTER

파라미터	설명
String	문자 타입 컬럼 또는 문자열
STR1	변경하려고 하는 문자 혹은 문자열
STR2	변경하고자 하는 문자 혹은 문자열



문자 처리 함수

문자 처리 함수 – REPLACE

수행 문장	결과
SELECT REPLACE('서울시 영등포구 당산동', '당산동', '양평동') FROM DUAL;	서울시 영등포구 양평동
SELECT REPLACE('test@naver.com', '@naver.com', '@gmail.com') FROM DUAL;	test@gmail.com





숫자 처리 함수



숫자 처리 함수

구분	입력값 타입	리턴 타입	설명
ABS	NUMBER	NUMBER	절대값을 구하여 리턴
MOD			입력 받은 수를 나눈 나머지 값을 리턴
ROUND			특정 자릿수에서 반올림하여 리턴
FLOOR			소수점 아래를 잘라내고 리턴(버림)
TRUNC			특정 자릿수에서 잘라내고 리턴(버림)
CEIL			지정한 자릿수에서 올림 하여 리턴



숫자 처리 함수

숫자 처리 함수 – ABS

인자로 전달받은 숫자의 절대값을 리턴

작성법	리턴 타입	파라미터
ABS(NUMBER)	NUMBER	숫자 혹은 숫자 데이터 컬럼

수행 문장	결과
SELECT ABS(10) FROM DUAL;	10
SELECT ABS(-10) FROM DUAL;	10



숫자 처리 함수

숫자 처리 함수 – MOD

인자로 전달받은 숫자를 나누어 나머지를 구하는 함수

작성법	리턴 타입	파라미터
MOD(NUMBER,DIVISION)	NUMBER	숫자 혹은 숫자 데이터 컬럼,나눌 수

수행 문장	결과
SELECT MOD(10,3) FROM DUAL;	1
SELECT MOD(10,2) FROM DUAL;	0
SELECT MOD(10,4) FROM DUAL;	2



숫자 처리 함수

숫자 처리 함수 – ROUND

인자로 전달 받은 숫자 혹은 컬럼에서 지정한 위치부터 반올림하여 값을 리턴

작성법	리턴 타입
ROUND(NUMBER) / ROUND(NUMBER,POSITION)	NUMBER

파라미터	설명
NUMBER	숫자 혹은 숫자 데이터 컬럼
POSITION	반올림 할 위치



숫자 처리 함수

숫자 처리 함수 – FLOOR

인자로 전달받은 숫자 혹은 컬럼의 소수점 자리의 수를 버리고 리턴

작성법	리턴 타입
FLOOR(NUMBER)	NUMBER

파라미터	설명
NUMBER	숫자 혹은 숫자 데이터 컬럼



숫자 처리 함수

숫자 처리 함수 – TRUNC

인자로 전달 받은 숫자 혹은 컬럼에서 지정한 위치부터 소수점 자리의 수를 버리는 함수

작성법	리턴 타입
TRUNC(NUMBER,POSITION)	NUMBER

파라미터	설명
NUMBER	숫자 혹은 숫자 데이터 컬럼
POSITION	버림 할 위치



숫자 처리 함수

숫자 처리 함수 – CEIL

인자로 전달받은 숫자 혹은 컬럼의 소수점 자리의 수를 올리고 리턴

작성법	리턴 타입
CEIL(NUMBER)	NUMBER

파라미터	설명
NUMBER	숫자 혹은 숫자 데이터 컬럼



숫자 처리 함수

숫자 처리 함수 – ROUND/FLOOR/TRUNC/CEIL

수행 문장	결과
SELECT ROUND(123.456) FROM DUAL;	123
SELECT ROUND(123.456, 1) FROM DUAL;	123.5
SELECT ROUND(123.456, 2) FROM DUAL;	123.46
SELECT ROUND(123.456, -1) FROM DUAL;	120
SELECT FLOOR(123.456) FROM DUAL;	123
SELECT TRUNC(123.456) FROM DUAL;	123
SELECT TRUNC(123.456, 1) FROM DUAL;	123.4
SELECT TRUNC(123.456, 2) FROM DUAL;	123.45
SELECT TRUNC(123.456, -1) FROM DUAL;	120
SELECT CEIL(123.456) FROM DUAL;	124





날짜 처리 함수



날짜 처리 함수

구분	입력값 타입	리턴 타입	설명
SYSDATE		DATE	시스템에 저장된 현재 날짜를 리턴
MONTHS_BETWEEN	DATE	NUMBER	두 날짜를 전달받아 몇 개월 차이인지 계산하여 리턴
ADD_MONTH	DATE	DATE	특정 날짜에 개월 수를 더하여 리턴
NEXT_DAY			특정 날짜에서 최초로 다가오는 인자로 받은 요일의 날짜 리턴
LAST_DAY			해당 달의 마지막 날짜를 리턴
EXTRACT			년,월,일 정보를 추출하여 리턴



날짜 처리 함수

날짜 처리 함수 – SYSDATE

시스템에 저장되어 있는 현재 날짜를 반환하는 함수

작성법	리턴 타입	파라미터
SYSDATE	DATE	없음



날짜 처리 함수

날짜 처리 함수 – MONTHS_BETWEEN

인자로 날짜 두 개를 전달받아, 개월 수의 차이를 숫자 데이터 형으로 리턴

작성법	리턴 타입
MONTHS_BETWEEN(DATE1, DATE2)	DATE

파라미터	설명
DATE1	기준이 되는 날짜를 입력
DATE2	개월 수를 구하려는 날짜를 입력



날짜 처리 함수

날짜 처리 함수 – MONTHS_BETWEEN

```
SELECT EMP_NAME, HIRE_DATE,  
       MONTHS_BETWEEN(SYSDATE, HIRE_DATE)  
FROM EMPLOYEE;
```

EMP_NAME	HIRE_DATE	MONTHS_BETWEEN(SYSDATE, HIRE_DATE)
선동일	90/02/06	360.609290621266427718040621266427718041
송송기	01/09/01	221.770580943847072879330943847072879331
노영철	01/01/01	229.770580943847072879330943847072879331
송은희	96/05/03	285.706064814814814814814814814814814815
유재식	00/12/29	229.867355137395459976105137395459976105
정송하	99/09/09	245.512516427718040621266427718040621266
박나라	08/04/02	142.738322879330943847072879330943847073
하이유	94/07/07	307.577032556750298685782556750298685783
김해술	04/04/30	189.835097072879330943847072879330943847
심봉선	11/11/11	99.4480002986857825567502986857825567503
윤은혜	01/02/03	228.706064814814814814814814814814814815
전형논	12/12/12	86.41574223416965352449223416965352449223
장쯔위	15/06/17	56.25445191158900836320191158900836320191
하동운	99/12/31	241.802839008363201911589008363201911589
방명수	10/04/04	118.673806750298685782556750298685782557



날짜 처리 함수

날짜 처리 함수 – ADD_MONTHS

인자로 전달받은 날짜에 인자로 전달받은 숫자만큼 개월 수를 더하여 특정 날짜를 리턴 하는 함수

작성법	리턴 타입
ADD_MONTHS(DATE, NUMBER)	DATE

파라미터	설명
DATE	기준이 되는 날짜를 입력
NUMBER	더하려는 개월 수를 입력한다.



날짜 처리 함수

날짜 처리 함수 – ADD_MONTHS

```
SELECT  EMP_NAME, HIRE_DATE,  
        ADD_MONTHS (HIRE_DATE,6)  
FROM EMPLOYEE;
```

EMP_NAME	HIRE_DATE	ADD_MONTHS(HIRE_DATE,6)
서인영	90/02/06	90/08/06
송송기	01/09/01	02/03/01
노조절	01/01/01	01/07/01
송희의	96/05/03	96/11/03
유재식	00/12/29	01/06/29
정송하	99/09/09	00/03/09
박나라	08/04/02	08/10/02
하이유	94/07/07	95/01/07
김해솔	04/04/30	04/10/31
심보선	11/11/11	12/05/11
권해	01/02/03	01/08/03
전영호	12/12/12	13/06/12
장쥬위	15/06/17	15/12/17
하노	99/12/31	00/06/30

날짜 처리 함수

날짜 처리 함수 – NEXT_DAY

인자로 전달받은 날짜에 인자로 전달받은 요일의 가장 가까운 날짜를 구하여 리턴 하는 함수

작성법	리턴 타입
NEXT_DAY(DATE, STRING NUMBER)	DATE

파라미터	설명
DATE	기준이 되는 날짜를 입력
STRING NUMBER	구하려는 요일 입력(숫자 1=일요일, ..., 7=토요일)



날짜 처리 함수

날짜 처리 함수 – NEXT_DAY

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

	SYSDATE	NEXT_DAY(SYSDATE, '월요일')
1	17/09/11	17/09/18

```
SELECT SYSDATE,  
NEXT_DAY(SYSDATE, 2)  
FROM DUAL;
```

	SYSDATE	NEXT_DAY(SYSDATE, '월요일')
1	17/09/11	17/09/18

```
SELECT SYSDATE,  
NEXT_DAY(SYSDATE, '월')  
FROM DUAL;
```

	SYSDATE	NEXT_DAY(SYSDATE, '월요일')
1	17/09/11	17/09/18



날짜 처리 함수

날짜 처리 함수 – LAST_DAY

인자로 전달받은 날짜가 속한 달의 마지막 날짜를 구하여 리턴 하는 함수

작성법	리턴 타입
LAST_DAY(DATE)	DATE

파라미터	설명
DATE	기준이 되는 날짜를 입력



날짜 처리 함수

날짜 처리 함수 - LAST_DAY

```
SELECT EMP_NAME,  
       HIRE_DATE,  
       LAST_DAY(HIRE_DATE)  
FROM EMPLOYEE;
```

	EMP_NAME	HIRE_DATE	LAST_DAY(HIRE_DATE)
1	선동일	90/02/06	90/02/28
2	송종기	01/09/01	01/09/30
3	노웅철	01/01/01	01/01/31
4	송은희	96/05/03	96/05/31
5	유재식	00/12/29	00/12/31
6	정중하	99/09/09	99/09/30
7	박나라	08/04/02	08/04/30
8	하미유	94/07/07	94/07/31
9	김해술	04/04/30	04/04/30
10	심봉선	11/11/11	11/11/30
11	윤은해	01/02/03	01/02/28
12	전형돈	12/12/12	12/12/31
13	장프위	15/06/17	15/06/30
14	하동운	99/12/31	99/12/31
15	방명수	10/04/04	10/04/30
16	대복훈	17/06/19	17/06/30
17	차태연	13/03/01	13/03/31
18	전지연	07/03/20	07/03/31



날짜 처리 함수

날짜 처리 함수 – EXTRACT

년,월,일 정보를 추출하여 리턴 하는 함수

작성법	리턴 타입
EXTRACT(YEAR FROM DATE) EXTRACT(MONTH FROM DATE) EXTRACT(DAY FROM DATE)	DATE

파라미터	설명
DATE	기준이 되는 날짜를 입력



날짜 처리 함수

날짜 처리 함수 – EXTRACT

```
SELECT EMP_NAME,  
       EXTRACT(YEAR FROM HIRE_DATE),  
       EXTRACT(MONTH FROM HIRE_DATE),  
       EXTRACT(DAY FROM HIRE_DATE)  
FROM EMPLOYEE
```

EMP_NAME	EXTRACT(YEARFROMHIRE_DATE)	EXTRACT(MONTHFROMHIRE_DATE)	EXTRACT(DAYFROMHIRE_DATE)
1 선동일	1990	2	6
2 송종기	2001	9	1
3 노응철	2001	1	1
4 송은희	1996	5	3
5 유재식	2000	12	29
6 정중하	1999	9	9
7 박나라	2008	4	2
8 하미유	1994	7	7
9 김해술	2004	4	30
10 심봉선	2011	11	11
11 윤은혜	2001	2	3
12 전형돈	2012	12	12
13 장프위	2015	6	17



형변환 함수



형변환 함수

구분	입력값 타입	리턴 타입	설명
TO_CHAR	DATE NUMBER	CHARACTER	날짜형 혹은 숫자를 문자형으로 변환
TO_DATE	CHARACTER	DATE	숫자 혹은 문자형을 날짜형으로 변환
TO_NUMBER	CHARACTER	NUMBER	문자형을 숫자형으로 변환



형변환 함수

형변환 함수 – TO_CHAR

날짜형 데이터 또는 숫자데이터를 문자형 데이터로 변환하여 리턴

작성법	리턴 타입
TO_CHAR(DATE,[FORMAT]) TO_CHAR(NUMBER,[FORMAT])	CHARACTER

파라미터	설명
DATE	문자형으로 변환하려는 날짜형 데이터
NUMBER	문자형으로 변환하려는 숫자형 데이터
FORMAT	문자형으로 변환 시 지정할 출력 형식



형변환 함수

형변환 함수 – TO_CHAR – 1 날짜변환 FORMAT

형식	설명	형식	설명
YYYY	년도표현 4자리	YY	년도표현 2자리
RR	년도 표현 2자리	MONTH	월 표시
MM	월을 숫자로 표현	MON	월을 알파벳으로
DD	날짜 표현	DAY	요일 표현
DY	요일을 약어로	D	요일을 숫자로
HH,HH12	시간(12시간)	HH24	시간(24시간)
MI	분	SS	초
AM,PM	오전,오후 표기	FM	앞자리 0을 제거

※ RR의 경우 50보다 작은경우 20XX로 변환하고 50보다 큰경우 19XX로 변환
20 → 2020 / 95 → 1995



형변환 함수

형변환 함수 – TO_CHAR – 1 날짜변환

```
SELECT EMP_NAME,  
       TO_CHAR(HIRE_DATE, 'YYYY-MM-DD'),  
       TO_CHAR(HIRE_DATE, 'YY/MON,DAY,DY')  
FROM EMPLOYEE
```

EMP_NAME	EXTRACT(YEARFROMHIRE_DATE)	EXTRACT(MONTHFROMHIRE_DATE)	EXTRACT(DAYFROMHIRE_DATE)
1 선동일	1990	2	6
2 송종기	2001	9	1
3 노용철	2001	1	1
4 송은희	1996	5	3
5 유재식	2000	12	29
6 정중하	1999	9	9
7 박나라	2008	4	2
8 하미유	1994	7	7
9 김해술	2004	4	30
10 심봉선	2011	11	11
11 윤은혜	2001	2	3
12 전형돈	2012	12	12
13 장프위	2015	6	17

형변환 함수

형변환 함수 – TO_CHAR – 2 숫자 FORMAT

형식	예시	설명
,(comma)	9,999	coma 형식으로 변환
.(period)	99.99	소수점 형식으로 변환
0	0999	왼쪽에 0을 삽입
\$	\$9999	\$ 통화로 표시
L	L9999	Local 통화로 표시(한국 \)
XXXX	XXXX	16진수로 표시

※ 숫자표시 단위는 충분한 크기를 주어야 하며, 변환될 숫자보다 format이 작을 경우 오류 발생

※ 9와 0으로 최대 개수를 표현해야함



형변환 함수

형변환 함수 – TO_CHAR – 2 숫자변환

```
SELECT EMP_NAME,  
       TO_CHAR(SALARY, 'L999,999,999'),  
       TO_CHAR(SALARY, '000,000,000')  
FROM EMPLOYEE;
```

	EMP_NAME	TO_CHAR(SALARY, 'L999,999,999')	TO_CHAR(SALARY, '000,000,000')
1	선동일	₩8,000,000	008,000,000
2	송중기	₩6,000,000	006,000,000
3	노웅철	₩3,700,000	003,700,000
4	송은희	₩2,800,000	002,800,000
5	유재식	₩3,400,000	003,400,000
6	정중하	₩3,900,000	003,900,000
7	박나라	₩1,800,000	001,800,000
8	하미유	₩2,200,000	002,200,000
9	김해솔	₩2,500,000	002,500,000
10	심봉선	₩3,500,000	003,500,000
11	윤은혜	₩2,000,000	002,000,000
12	전형돈	₩2,000,000	002,000,000
13	장프위	₩2,550,000	002,550,000
14	하동운	₩2,320,000	002,320,000
15	반면스	₩1,380,000	001,380,000

형변환 함수

형변환 함수 – TO_DATE

날짜형 데이터 또는 문자형 데이터를 날짜형 데이터로 변환하여 리턴

작성법	리턴 타입
TO_DATE(Character,[format]) TO_DATE(Number,[format])	DATE

파라미터	설명
CHARACTER	날짜형으로 변환하려는 문자형 데이터
NUMBER	날짜형으로 변환하려는 숫자형 데이터
FORMAT	날짜형으로 변환 시 지정할 출력 형식



형변환 함수

형변환 함수 – TO_DATE

```
SELECT EMP_NO,  
       EMP_NAME,  
       HIRE_DATE  
FROM EMPLOYEE  
WHERE HIRE_DATE > TO_DATE(20000101, 'YYYYMMDD');
```

EMP_NO	EMP_NAME	HIRE_DATE
631156-1548654	송송기	01/09/01
861015-1356452	노옹철	01/01/01
660508-1342154	유재식	00/12/29
630709-2054321	박나라	08/04/02
870927-1313564	김해솔	04/04/30
750206-1325546	심봉선	11/11/11
650505-2356985	윤은해	01/02/03
830807-1121321	전형논	12/12/12
780923-2234542	장쯔위	15/06/17
856795-1313513	방명수	10/04/04
881130-1050911	대북훈	17/06/19
770808-1364897	차태연	13/03/01
770808-2665412	전지연	07/03/20
870427-2232123	이오리	16/11/28
770823-1113111	이승석	14/09/18

형변환 함수

형변환 함수 – TO_NUMBER

문자형 데이터를 숫자형 데이터로 변환하여 리턴

작성법	리턴 타입
TO_NUMBER(Character,[format])	NUMBER

파라미터	설명
CHARACTER	숫자형으로 변환하려는 문자형 데이터
FORMAT	숫자형으로 변환 시 지정할 출력 형식



형변환 함수

형변환 함수 – TO_NUMBER

```
SELECT  
TO_NUMBER('1,000,000', '99,999,999') - TO_NUMBER('550,000', '999,999')  
FROM DUAL;
```

	 TO_NUMBER('1,000,000', '99,999,999') - TO_NUMBER('550,000', '999,999')
1	450000

※ 주의사항

1. TO_CHAR과 마찬가지로 값을 담을 수 있을 정도로 충분히 큰 포맷을 지정해야 한다.
2. 유효한 숫자를 입력해야 한다.('123a'와 같이 숫자가 아닌 것은 변환 시 에러 발생)
3. 오라클에서도 자동형변환이 일어나지만 +가 산술연산으로만 사용됨

'1000'+'100' → 1100

'1000a'+'100' → 에러발생





기타 함수



기타 함수

NULL 처리 함수 – NVL

NULL로 되어 있는 컬럼의 값을 지정한 숫자 혹은 문자로 변경하여 리턴

작성법	리턴 타입
NVL(P1,P2)	NUMBER CHARACTER

파라미터	설명
P1	NULL 데이터를 처리할 컬럼명 혹은 값
P2	NULL값을 변경하고자 하는 값



기타 함수

NULL 처리 함수 – NVL

```
SELECT EMP_NAME,  
       SALARY,  
       NVL(BONUS, 0),  
       (SALARY * 12 + (SALARY * 12 * NVL(BONUS, 0)))  
FROM EMPLOYEE;
```

※ BONUS가 NULL인 경우
0으로 계산

EMP_NAME	SALARY	NVL(BONUS,0)	(SALARY*12+(SALARY*12*NVL(BONUS,0)))
김영일	8000000	0.3	124800000
송기철	6000000	0	72000000
노영희	3700000	0	44400000
송이재	2800000	0	33600000
유재식	3400000	0.2	48960000
정송하	3900000	0	46800000
박나라	1800000	0	21600000
하이유	2200000	0.1	29040000
김해솔	2500000	0	30000000
김희선	3500000	0.15	48300000
김민재	2000000	0	24000000
정민호	2000000	0	24000000
장지현	2550000	0.25	38250000
하영호	2320000	0.1	30624000

기타 함수

선택 함수 – DECODE

여러 가지 경우에 선택을 할 수 있는 기능을 제공(일치하는 값)

작성법	리턴 타입
DECODE(표현식,조건1,결과1,조건2,결과2.....)	결과

파라미터	설명
표현식	값에 따라 선택을 다르게 할 컬럼 혹은 값 입력
조건	해당 값이 참인지 거짓인지 여부를 판단
결과	해당 조건과 일치하는 경우 결과를 리턴
DEFAULT	모든 조건이 불일치 시 리턴 할 값



기타 함수

선택 함수 – DECODE

```
SELECT EMP_ID,  
       EMP_NAME,  
       EMP_NO,  
       DECODE(SUBSTR(EMP_NO, 8, 1), '1', '남', '2', '여') AS 성별  
FROM EMPLOYEE;
```

EMP_ID	EMP_NAME	EMP_NO	성별
1 200	선동일	621235-1985634	남
2 201	송종기	631156-1548654	남
3 202	노용철	861015-1356452	남
4 203	송은희	631010-2653546	여
5 204	유재식	660508-1342154	남
6 205	정중하	770102-1357951	남
7 206	박나라	630709-2054321	여
8 207	하미유	690402-2040612	여
9 208	김해솔	870927-1313564	남
10 209	심봉선	750206-1325546	남
11 210	윤은혜	650505-2356985	여
12 211	전형돈	830807-1121321	남
13 212	장프위	780923-2234542	여
14 213	하동운	621111-1785463	남
15 214	방명수	856795-1313513	남
16 215	대불호	881130-1050911	남

기타 함수

선택 함수 – CASE

여러 가지 경우에 선택을 할 수 있는 기능을 제공(범위 값도 가능)

작성법	리턴 타입
CASE WHEN 조건1 THEN 결과1 WHEN 조건2 THEN 결과2 WHEN 조건3 THEN 결과3 ELSE 결과 END	결과

파라미터	설명
조건	해당 값이 참인지 거짓인지 여부를 판단
결과	해당 조건과 일치하는 경우 결과를 리턴
ELSE	모든 조건이 불일치 시 리턴 할 값



기타 함수

선택 함수 – CASE

```
SELECT EMP_ID,  
       EMP_NAME,  
       EMP_NO,  
       CASE WHEN SUBSTR(EMP_NO, 8, 1) = 1 THEN '남'  
            ELSE '여' END AS 성별  
FROM EMPLOYEE;
```

EMP_ID	EMP_NAME	EMP_NO	성별
1 200	선동일	621235-1985634	남
2 201	송종기	631156-1548654	남
3 202	노용철	861015-1356452	남
4 203	송은희	631010-2653546	여
5 204	유재식	660508-1342154	남
6 205	정중하	770102-1357951	남
7 206	박나라	630709-2054321	여
8 207	하미유	690402-2040612	여
9 208	김해솔	870927-1313564	남
10 209	심봉선	750206-1325546	남
11 210	윤은혜	650505-2356985	여
12 211	전형돈	830807-1121321	남
13 212	장프위	780923-2234542	여
14 213	하동운	621111-1785463	남
15 214	방명수	856795-1313513	남
16 215	대북호	881130-1050911	남



그룹 함수



그룹 함수

그룹 함수

하나 이상의 행을 그룹으로 묶어 연산하여 총합, 평균 등을 하나의 컬럼으로 리턴 하는 함수

구분	설명
SUM	그룹의 누적 합계를 리턴
AVG	그룹의 평균을 리턴
COUNT	그룹의 총 개수를 리턴
MAX	그룹의 최대값을 리턴
MIN	그룹의 최소값을 리턴



그룹 함수

그룹 함수 – SUM

해당 컬럼 값들의 총 합을 구하는 함수

[EMPLOYEE 테이블에서 남자사원의 급여 총합]

```
SELECT SUM(SALARY) FROM EMPLOYEE  
WHERE SUBSTR(EMP_NO,8,1)=1;
```

	SUM(SALARY)
1	49760000

[EMPLOYEE 테이블에서 부서코드가 D5인 직원의 보너스 포함 연봉의 총합]

```
SELECT SUM((SALARY + (SALARY * NVL(BONUS, 0)) * 12))  
FROM EMPLOYEE  
WHERE DEPT_CODE = 'D5';
```

	SUM((SALARY+(SALARY*NVL(BONUS,0))*12))
1	24700000



그룹 함수

그룹 함수 – AVG

해당 컬럼 값들의 평균을 구하는 함수

[EMPLOYEE 테이블에서 전사원의 보너스의 평균을 구해서 소수 2자리에서 반올림]

```
SELECT ROUND(AVG(BONUS), 2)  
FROM EMPLOYEE;
```

	ROUND(AVG(BONUS),2)
1	0.22

※ 그룹함수 연산 시 NULL인 경우 연산제외



[EMPLOYEE 테이블에서 전사원의 보너스의 평균을 구해서 소수 2자리에서 반올림]

```
SELECT ROUND(AVG(NVL(BONUS, 0)), 2)  
FROM EMPLOYEE;
```

	ROUND(AVG(NVL(BONUS,0)),2)
1	0.08



그룹 함수

그룹 함수 – COUNT

테이블에서 조건을 만족하는 행의 개수를 반환하는 함수

[EMPLOYEE 테이블에서 전체 사원의 수 조회]

```
SELECT COUNT(*) FROM EMPLOYEE;
```

	COUNT(*)
1	23

[EMPLOYEE 테이블에서 D5 부서에 속해 있는 전체 사원의 수 조회]

```
SELECT COUNT(*) FROM EMPLOYEE  
WHERE DEPT_CODE='D5';
```

	COUNT(*)
	6

[EMPLOYEE 테이블에서 직원들이 속해 있는 부서의 수를 조회]

```
SELECT COUNT(DISTINCT DEPT_CODE)  
FROM EMPLOYEE;
```

	COUNT(DISTINCTDEPT_CODE)
1	6



그룹 함수

그룹 함수 – MAX / MIN

그룹의 최대값과 최소값을 구하여 리턴 하는 함수

[EMPLOYEE 테이블에서 사원 중 가장 높은 급여와 가장 낮은 급여를 조회]

```
SELECT MAX(SALARY), MIN(SALARY)  
FROM EMPLOYEE;
```

	MAX(SALARY)	MIN(SALARY)
1	8000000	1380000

[EMPLOYEE 테이블에서 가장 오래된 입사일과 가장 최근 입사일을 조회]

```
SELECT MAX(HIRE_DATE), MIN(HIRE_DATE)  
FROM EMPLOYEE;
```

	MAX(HIRE_DATE)	MIN(HIRE_DATE)
1	17/06/19	90/02/06

