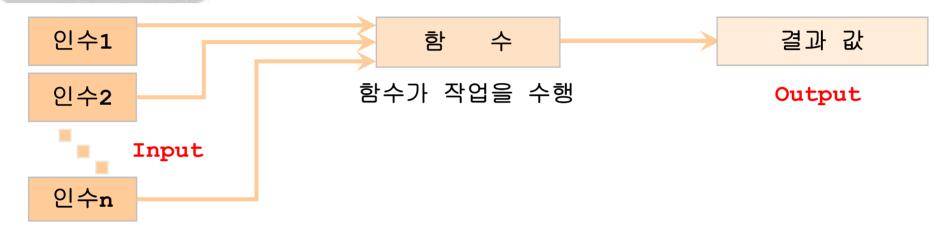


문자 함수 I

# 1. SQL 함수

# 1.1 함수의 정의





### 문자 함수 I

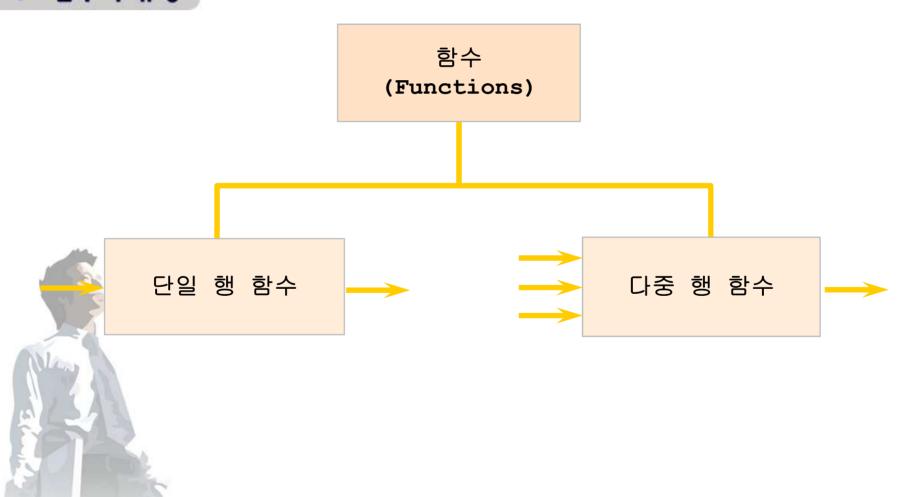
# 1.2 함수의 기능

- ☑ 데이터에 대한 계산 수행
- ☑ 개별 데이터 항목 수정
- ☑ 행 그룹에 대한 출력 결과 조작
- ☑ 표시할 날짜 및 숫자 형식 지정
- ☑ 열 데이터 유형 변환





# 1.3 함수의 유형



#### 문자 함수 I

## 2. DUAL 테이블

- ☑ DUAL 테이블은 사용자 SYS가 소유하며 모든 사용자가 액세스 가능
- ☑ DUMMY라는 column과 x값을 소유
- ☑ 사용자가 데이터가 들어있는 테이블에서 파생된 표현식, 의사열, 상수값 등을 한번만 반환할 때 유용
- ☑ SELECT절과 FROM절은 필수 항목

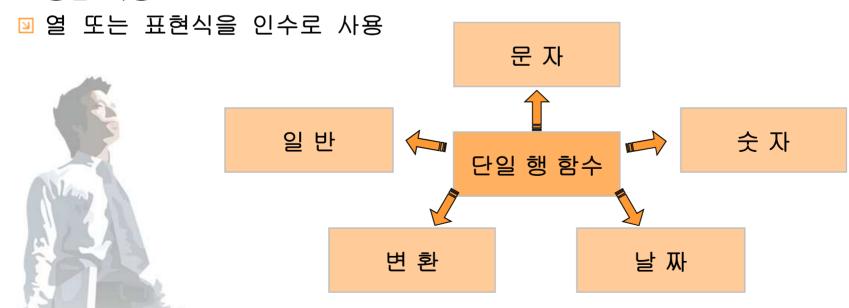
SQL> SELECT SYSDATE
2 FROM DUAL;



#### 문자 함수 I

# 3. 단일 행 함수 정의

- ☑ 데이터 항목 조작
- ☑ 인수를 사용하고 값을 하나 반환
- ☑ 반환되는 각 행에 대해 작업
- ☑ 행 당 하나의 결과 반환
- ☑ 데이터 유형 수정
- ☑ 중첩 가능



# 4. 문자 함수 (Character Functions)

# 4.1 개요

문자 함수

대소문자 조작 함수

문자 조작 함수

LOWER

**UPPER** 

INITCAP

CONCAT

SUBSTR

LENGTH

INSTR

LPAD | RPAD

TRIM

# 4.2 대소문자 조작 함수

함 수	결 과
LOWER('YE DAM') UPPER('ye dam') INITCAP('ye dam')	ye dam YE DAM Ye Dam

```
SQL> SELECT LOWER('YE DAM')
2 FROM dual;

LOWER(
-----
ye dam
```

### 4.2 대소문자 조작 함수 (계속)

- ☑ 스크립트 파일 (skjtest.sql) 다운로드
- ☑ C:\ 디렉토리에 복사
- ☑ 스크립트 실행

```
SQL> @ C:\skjtest.sql
```

```
SQL> SELECT *
2 FROM ss
3 WHERE LOWER(name) = 'apple';

S_ID NAME PRICE

1 APPLE 100
```

## 4.2 대소문자 조작 함수 (계속)

```
SQL> SELECT UPPER('ye dam')
 2 FROM dual;
UPPER (
YE DAM
SQL> SELECT *
 2 FROM ss
 3 WHERE name = UPPER('apple');
    S ID NAME PRICE
        1 APPLE
                       100
```

### 문자 함수 I



# 4.2 대소문자 조작 함수 (계속)

```
SQL> SELECT INITCAP('ye dam')
2 FROM dual;
```

INITCA

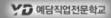
Ye Dam



### 4.2 대소문자 조작 함수 (계속)

```
SQL> SELECT s_id || '번 과일의 이름은: ' || INITCAP(name)
2 || '이며, ' || price || '원 입니다.'
3 AS "과일 정보"
3 FROM ss;
```

```
1번 과일의 이름은: Apple이며, 100원 입니다.2번 과일의 이름은: Orange이며, 200원 입니다.3번 과일의 이름은: Banana이며, 300원 입니다.4번 과일의 이름은: Grape이며, 500원 입니다.
```





# 1. 문자 함수 (Character Functions)

# 1.1 개요

문자 함수

대소문자 조작 함수

문자 조작 함수

LOWER

**UPPER** 

INITCAP

CONCAT

SUBSTR

LENGTH

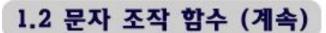
INSTR

LPAD | RPAD

TRIM

# 1.2 문자 조작 함수

	함 수	결 과
	CONCAT ( 'Ye', 'Dam')	YeDam
	SUBSTR('yedam',3,2)	da
	LENGTH ( 'yedam')	5
	<pre>INSTR('yedam', 'd')</pre>	3
	LPAD ('yedam', 10, '*')	****yedam
	RPAD ('yedam', 10, '*')	yedam****
3	TRIM('d' FROM 'ddyedamd')	yedam



```
SQL> SELECT CONCAT('Ye', 'Dam')
2 FROM dual;
```

CONCA

\_\_\_\_

YeDam



## 1.2 문자 조작 함수 (계속)

```
SQL> SELECT CONCAT(job, ename)
2 FROM emp;
```

CONCAT (JOB, ENAME)

------

PRESIDENTKING

**MANAGERJONES** 

**MANAGERBLAKE** 

MANAGERCLARK

ANALYSTFORD

ANALYSTSCOTT

SALESMANALLEN

SALESMANWARD

SALESMANMARTIN

SALESMANTURNER

**CLERKJAMES** 

CLERKMILLER

• • •

### 문자 함수 II



# 1.2 문자 조작 함수 (계속)

```
SQL> SELECT SUBSTR('yedam', 3, 2)
2 FROM dual;
```

SU

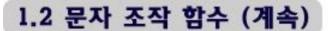
--

da



# 1.2 문자 조작 함수 (계속)

```
SQL> SELECT SUBSTR (ename, 3, 2)
  2 FROM emp;
SUBSTR (ENAME, 3, 2
NG
NE
AK
AR
RD
OT
LE
RD
RT
RN
ME
...
```



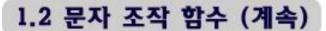
```
SQL> SELECT LENGTH('yedam')
2 FROM dual;

LENGTH('yedam')
-----
5
```



# 1.2 문자 조작 함수 (계속)

```
SQL> SELECT LENGTH(ename)
  2 FROM emp;
LENGTH (ENAME)
```

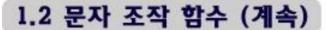


```
SQL> SELECT INSTR('yedam', 'd')
2 FROM dual;
INSTR('YEDAM', 'D'
```



# 1.2 문자 조작 함수 (계속)

```
SQL> SELECT INSTR(ename, 'I')
  2 FROM
          emp;
INSTR(ENAME, 'I')
```

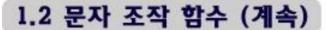


```
LPAD('YEDA RPAD('YEDA
-----
****yedam yedam$$$$$
```



# 1.2 문자 조작 함수 (계속)

```
SQL> SELECT LPAD (ename, 10, '*')
  2 FROM
             emp;
LPAD (ENAME, 10, '*')
*****KING
****JONES
****BLAKE
*****CLARK
*****FORD
****SCOTT
****ALLEN
*****WARD
****MARTIN
****TURNER
****JAMES
```



```
SQL> SELECT TRIM('d' from 'ddyedamd')
2 FROM dual;

TRIM(
-----
yedam
```



## 1.2 문자 조작 함수 (계속)

```
SQL> SELECT TRIM('S' from ename), ename
2 FROM emp;
```

```
TRIM('S' FROM ENAME) ENAME
KING
                      KING
JONE
                      JONES
BLAKE
                      BLAKE
CLARK
                     CLARK
FORD
                      FORD
COTT
                      SCOTT
ALLEN
                      ALLEN
WARD
                      WARD
MARTIN
                      MARTIN
TURNER
                      TURNER
                      JAMES
JAME
...
```