



06장(오라클함수)

SQL문(쿼리문)은 대소문자 구분이 없지만, 데이터는 대소문자 구분이 필수!

함수란?

- 특정한 결과 값을 얻기 위해 데이터를 입력할 수 있는 특수 명령어
- 오라클의 함수 종류
 - 내장함수 → 기본으로 제공
 - 단일행 함수
 - 다중행 함수
 - 정의함수 → 사용자가 필요에 의해 직접 정의

문자함수

- **UPPER()** → 모두 대문자로 변경
 - ▼ **사원이름이 scott인 사원 대소문자 상관없이 찾기**

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

- **LOWER()** → 모두 소문자로 변경
- **INITCAP()** → 첫번째 단어만 대문자로 변경
- **LENGTH()** → 문자열의 길이를 구하는 함수
- **SUBSTR(대상, start, [length])** → 필요한 부분만 추출
- **INSTR(대상, 찾으려는문자, [start], [~th])** → 특정 문자의 위치 찾기

- 위치 값을 음수로 사용하면 오른쪽 끝부터 왼쪽으로 검색
- 찾으려는 데이터가 없으면 0을 반환
- **REPLACE(대상, target, change)** → 문자를 대체하는 함수
- **LPAD, RPAD(대상, length, ['change'])** → 빈공간을 'change'로 채우는 함수(입력안 하면 공백)
- **CONCAT()** → 두 문자를 합치는 함수
- **TRIM, LTRIM, RTRIM(대상, [삭제할문자])** → 특정 문자를 삭제(문자 생략시 공백을 제거)
 - TRIM은 데이터에 혹시나 들어 있을지도 모르는 양쪽끝의 공백을 제거할 때 많이 사용

ex) 유저가 로그인시 실수로 스페이스바를 입력했을 경우

숫자함수

- **ROUND(대상, [location])** → 반올림 함수
 - 양수 : 해당 숫자자리까지 보임
 - 0 : 소수첫째자리에서 반올림
 - 음수 : 해당 숫자의 절댓값 자리에서 반올림
- **TRUNC(대상, [location])** → 버림 함수
- **CEIL()** → 지정 숫자보다 큰 수중 가장 작은 정수
- **FLOOR()** → 지정 숫자보다 작은 수중 가장 큰 정수
- **MOD(대상, 나눌 숫자)** → 지정함수를 나눈 나머지 값

날짜함수

▼ 날짜 포맷 모델

포맷 모델	기준 단위
CC, SCC	네 자리 연도의 끝 두 자리를 기준으로 사용 (2016년이면 2050 이하이므로, 반올림할 경우 2001년으로 처리)
YYYY, YYYY, YEAR, SYEAR, YYY, YY, Y	날짜 데이터의 해당 연·월·일의 7월 1일을 기준 (2016년 7월 1일 일 경우, 2017년으로 처리)
IYYY, IYY, IY, I	ISO 8601에서 제정한 날짜 기준년도 포맷을 기준
Q	각 분기의 두 번째 달의 16일 기준
MONTH, MON, MM, RM	각 달의 16일 기준
WW	해당 연도의 몇 주(1~53번째 주)를 기준
IW	ISO 8601에서 제정한 날짜 기준 해당 연도의 주(week)를 기준
W	해당 월의 주(1~5번째 주)를 기준
DDD, DD, J	해당 일의 정오(12:00:00)를 기준
DAY, DY, D	한 주가 시작되는 날짜를 기준
HH, HH12, HH24	해당일의 시간을 기준
MI	해당일 시간의 분을 기준

- 연산(일수로 계산됨)

- 날짜 + 숫자 : 가능
- 날짜 - 숫자 : 가능
- 날짜 - 날짜 : 두 날짜의 차이를 계산
- 날짜 + 날짜 : 불가능

- **ADD_MONTHS(날짜, 숫자)** → 숫자만큼 더함

- **MONTHS_BETWEEN(날짜, 날짜)** → 두 날짜간의 개월 수 차이를 구함

- **NEXT_DAY(날짜, 요일)** → 입력한 요일에서 돌아오는 요일의 날짜를 출력

- **LAST_DAY(날짜)** → 해당 달의 마지막 날짜를 반환

- **ROUND()**

- 숫자, 위치
- 날짜, 기준

- **TRUNC()**

- 숫자, 위치
- 날짜, 기준

형 변환 함수

- **TO_CHAR**(날짜, '포맷형태',]'NLS_DATE_LANGUAGE = '])
 - 날짜를 문자열로 포맷형태로 변경
 - LANGUAGE 설정시 해당 언어에 맞게 출력됨(7월, JULY, 07, ...)
- **TO_DATE**(문자, '포맷형태')

NULL처리 함수

- **NVL**(대상, 변환할 데이터)
- **NVL2**(대상, NULL아닐시, NULL일시)

조건함수

- **DECODE**()

▼ 예시 코드

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

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

- CASE()

- ▼ 예시 코드

```

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;

```

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

▼ 조건식을 사용한 예시코드

```

SELECT EMPNO, ENANE, COMM,
CASE
  WHEN COMM IS NULL THEN '해당사항없음'
  WHEN COMM=0 THEN '수당없음'
  WHEN COMM>0 THEN '수당:' || COMM
END AS COMM_TEXT
FROM EMP;

```

EMPNO	ENAME	COMM	COMM_TEXT
7369	SMITH		해당사항 없음
7499	ALLEN	300	수당 : 300
7521	WARD	500	수당 : 500
7566	JONES		해당사항 없음
7654	MARTIN	1400	수당 : 1400
7698	BLAKE		해당사항 없음
7782	CLARK		해당사항 없음
7788	SCOTT		해당사항 없음
7839	KING		해당사항 없음
7844	TURNER	0	수당없음
7876	ADAMS		해당사항 없음
7900	JAMES		해당사항 없음
7902	FORD		해당사항 없음
7934	MILLER		해당사항 없음

