



데이터베이스 Database

07

숫자, 날짜, 변환, 일반 함수

숫자
날짜
변환
일반
함수
함수
함수
함수
함수

숫자 함수

- 숫자 함수는 주로 숫자 계산과 추가 처리에 사용
- 숫자 함수 종류

함수	설명	예	결과
CEIL()	정수 올림	CEIL(12.345)	13
FLOOR()	정수 내림	FLOOR(12.345)	12
ROUND()	반올림	ROUND(12.345, 0)	12
TRUNC()	절삭	TRUNC(12.345, 1)	12.3
MOD()	나눈 나머지	MOD(12, 5)	2
SIGN()	양수, 음수, 0 구분	SIGN(12)	1
POWER()	거듭제곱	POWER(3, 3)	27
SQRT()	제곱근	SQRT(4)	2

- CEIL() / FLOOR()
 - 숫자를 정수로 올림하는 CEIL() 함수, 숫자를 정수로 내림하는 FLOOR() 함수

```
SELECT salary, salary/21,
       CEIL(salary/21), FLOOR(salary/21)
FROM employees;
```

- ROUND() / TRUNC()
 - 숫자를 반올림하는 ROUND() 함수, 숫자를 절삭하는 TRUNC() 함수

```
SELECT salary, salary/21,
       ROUND(salary/21), ROUND(salary/21, 2), ROUND(salary/21, -1),
       TRUNC(salary/21), TRUNC(salary/21, 2), TRUNC(salary/21, -1)
FROM employees;
```

숫자 함수

- MOD()

- 숫자를 나눈 후 나머지를 구함

```
SELECT salary, salary/21, MOD(salary, 21)
FROM employees;
```

- SIGN()

- 숫자가 양수일 경우 1, 음수일 경우 -1, 나머지는 0을 반환

```
SELECT SIGN(-123), SIGN(0), SIGN(123)
FROM dual;
```

- POWER() / SQRT()

- 거듭제곱을 출력하는 POWER() 함수, 제곱근을 출력하는 SQRT() 함수

```
SELECT POWER(3, 3), SQRT(4)
FROM dual;
```

날짜 함수

- 데이터 중의 날짜 형식을 가지는 데이터를 계산하기 위해서 사용되는 날짜 함수
 - 날짜(Date) + 숫자(Number) = 날짜에 숫자 이후의 날짜
 - 날짜(Date) - 숫자(Number) = 날짜에 숫자 이전의 날짜
 - 날짜(Date) + 날짜(Date) = 날짜에서 날짜를 더한 날짜
 - 날짜(Date) - 날짜(Date) = 날짜에서 날짜를 뺀 날짜
- SYSDATE
 - 오라클이 설치된 시스템의 현재 날짜를 반환

```
SELECT SYSDATE, SYSDATE + 1, SYSDATE - 1
FROM dual;
```

- 날짜 함수

함수	설명	예	결과
MONTH_BETWEEN()	두 날짜 사이의 개월 수 계산	MONTHS_BETWEEN(SYSDATE, hire_date)	218.683
ADD_MONTHS()	날짜에 지정한 개월 수를 더함	ADD_MONTHS(hire_date, 3)	03/09/17
NEXT_DAY()	돌아오는 요일의 날짜 반환	NEXT_DAY(hire_date, 3)	03/06/24
LAST_DAY()	월의 마지막 날짜를 계산	LAST_DAY(hire_date)	03/06/30
ROUND()	날짜를 가장 가까운 연도나 월로 반올림	ROUND(hire_date, 'YEAR')	03/01/01
TRUNC()	날짜를 가장 가까운 연도 또는 월로 절삭	TRUNC(hire_date, 'MONTH')	03/06/01

날짜 함수

- MONTH_BETWEEN()
 - 날짜와 날짜 사이의 개월 수를 계산

```
SELECT SYSDATE, hire_date, MONTHS_BETWEEN(SYSDATE, hire_date)
FROM employees;
```

- ADD_MONTH()
 - 오라클이 설치된 시스템의 현재 날짜를 반환

```
SELECT hire_date,
       ADD_MONTHS(hire_date, 2), ADD_MONTHS(hire_date, -2)
FROM employees;
```

- NEXT_DAY() / LAST_DAY()
 - 지정된 날짜부터 돌아오는 요일 날짜를 출력하는 NEXT_DAY() 함수
 - 월의 마지막 날짜를 계산하는 LAST_DAY() 함수

```
SELECT hire_date,
       NEXT_DAY(hire_date, 3),
       NEXT_DAY(hire_date, '수요일'),
       LAST_DAY(hire_date)
FROM employees;
```

- ROUND() / TRUNC()
 - 날짜를 연도나 월 단위로 반올림하는 ROUND() 함수
 - 날짜를 연도나 월 단위로 절삭하는 TRUNC() 함수

```
SELECT hire_date,
       ROUND(hire_date, 'YEAR'),
       TRUNC(hire_date, 'MONTH')
FROM employees;
```

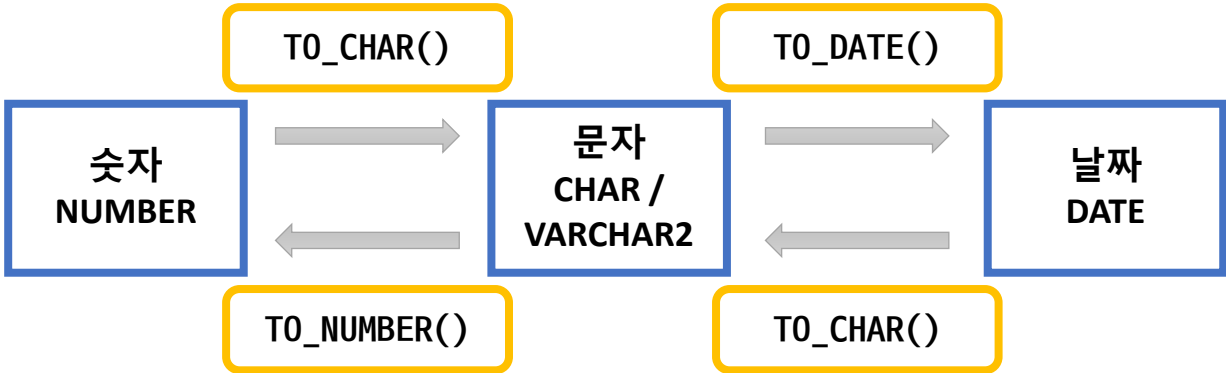
변환 함수

- 오라클에서 제공하는 데이터 타입을 필요에 따라 변환
- 자동(암묵적) 변환



```
SELECT 1 + '2'  
FROM dual;
```

- 수동(명시적) 변환



함수	설명
TO_CHAR()	숫자와 날짜 데이터를 문자 데이터로 변환
TO_NUMBER()	문자 데이터를 숫자 데이터로 변환
TO_DATE()	문자 데이터를 날짜 데이터로 변환

변환 함수

- TO_CHAR(): 숫자와 날짜 데이터를 문자 데이터로 변환하는 함수

```
SELECT TO_CHAR(SYSDATE, 'CC AD Q')
FROM dual;
```

```
SELECT TO_CHAR(SYSDATE, 'YYYY/MM/DD')
FROM dual;
```

```
SELECT TO_CHAR(SYSDATE, 'W DAY')
FROM dual;
```

- 날짜 지정 형식

형식	설명	함수 사용 예	결과
CC	세기	TO_CHAR(SYSDATE, 'CC')	21
AD	서기	TO_CHAR(SYSDATE, 'AD')	서기
Q	분기	TO_CHAR(SYSDATE, 'Q')	3
YYYY	연도(4자리)	TO_CHAR(SYSDATE, 'YYYY')	2021
YYY	연도(3자리)	TO_CHAR(SYSDATE, 'YYY')	021
YY	연도(2자리)	TO_CHAR(SYSDATE, 'YY')	21
Y	연도(1자리)	TO_CHAR(SYSDATE, 'Y')	1
YEAR	연도(문자)	TO_CHAR(SYSDATE, 'YEAR')	TWENTY
MM	월(2자리)	TO_CHAR(SYSDATE, 'MM')	09
MON	월 이름	TO_CHAR(SYSDATE, 'MON')	9월
RM	월(로마 숫자)	TO_CHAR(SYSDATE, 'RM')	IX
WW	주(연 기준)	TO_CHAR(SYSDATE, 'WW')	36
W	주(월 기준)	TO_CHAR(SYSDATE, 'W')	2
DDD	일(연 기준)	TO_CHAR(SYSDATE, 'DDD')	252
DD	일(월 기준)	TO_CHAR(SYSDATE, 'DD')	09
D	일(주 기준)	TO_CHAR(SYSDATE, 'D')	5
DAY	요일	TO_CHAR(SYSDATE, 'DAY')	목요일
DY	요일(약어)	TO_CHAR(SYSDATE, 'DY')	목

변환 함수

- TO_CHAR(): 숫자와 날짜 데이터를 문자 데이터로 변환하는 함수

```
SELECT TO_CHAR(SYSDATE, 'AM HH:MI:SS')
FROM dual;
```

```
SELECT TO_CHAR(SYSDATE, 'HH24:MI:SS PM')
FROM dual;
```

```
SELECT TO_CHAR(SYSDATE, 'YY-MM-DD')
FROM dual;
```

```
SELECT TO_CHAR(SYSDATE, 'MM"월" DD"일"')
FROM dual;
```

- 시간 지정 형식

형식	설명	함수 사용 예	결과
AM / PM	오전 / 오후	TO_CHAR(SYSDATE, 'AM')	오전
HH / HH12 / HH24	시간(1~12시) 시간(0~23시)	TO_CHAR(SYSDATE, 'HH24')	22
MI	분(0~59)	TO_CHAR(SYSDATE, 'MI')	15
SS	초(0~59)	TO_CHAR(SYSDATE, 'SS')	10

- 기타 형식

형식	설명	함수 사용 예	결과
/ : -	출력 결과 표현 문자	TO_CHAR(SYSDATE, 'YY-MM-DD')	21-09-09
“텍스트”	출력 결과 텍스트	TO_CHAR(SYSDATE, 'HH"월" DD"일"')	09월 09일

변환 함수

- TO_CHAR(): 숫자와 날짜 데이터를 문자 데이터로 변환하는 함수

```
SELECT TO_CHAR(salary, '9999999')
FROM employees;
```

```
SELECT TO_CHAR(salary, '0999999')
FROM employees;
```

```
SELECT TO_CHAR(salary, '$999999')
FROM employees;
```

```
SELECT TO_CHAR(salary, 'L999999')
FROM employees;
```

```
SELECT TO_CHAR(salary, '99999.99')
FROM employees;
```

```
SELECT TO_CHAR(salary, '9,999,999')
FROM employees;
```

- 숫자 지정 형식

형식	설명	함수 사용 예	결과
9	출력 자릿수	TO_CHAR(salary, '9999999')	24000
0	자릿수만큼 0 출력	TO_CHAR(salary, '0999999')	0024000
\$	달러 기호	TO_CHAR(salary, '\$999999')	\$24000
L	지역 화폐 기호	TO_CHAR(salary, 'L999999')	₩24000
.	소수점 표시	TO_CHAR(salary, '99999.99')	24000.00
,	쉼표 표시	TO_CHAR(salary, '9,999,999')	24,000

변환 함수

- TO_NUMBER(): 숫자로 된 문자열을 숫자 타입으로 변환하는 함수

```
SELECT TO_NUMBER('123')  
FROM dual;
```

```
SELECT TO_NUMBER('123.123')  
FROM dual;
```

- TO_DATE(): 숫자로 된 문자열을 날짜 타입으로 변환하는 함수

```
SELECT TO_DATE('20210909', 'YYMMDD')  
FROM dual;
```

일반 함수

- NVL(): NULL 값을 특정한 값으로 치환하는 함수

```
SELECT department_name, NVL(manager_id, 100)
FROM departments;
```

```
SELECT NVL(state_province, 'None')
FROM locations
ORDER BY state_province;
```

- NVL2(): NULL 값인 경우와 아닌 경우를 구분하여 특정한 값으로 치환하는 함수

```
SELECT department_name,
       NVL2(manager_id, '관리자 있음', '관리자 없음')
FROM departments;
```

```
SELECT city, NVL2(state_province, '주소 있음', '주소 없음')
FROM locations
ORDER BY state_province;
```

- DECODE(): 데이터가 조건 값과 일치하면 치환 값을 출력하고, 일치하지 않으면 기본값을 출력하는 조건 논리 처리 함수

```
SELECT job_title, min_salary,
       DECODE(min_salary, 2500, min_salary * 1.1, min_salary)
FROM jobs;
```

```
SELECT job_title, max_salary,
       DECODE(max_salary, 40000, max_salary * 0.9, max_salary)
FROM jobs;
```

일반 함수

- CASE(): 복잡한 논리 조건 처리 함수

```
SELECT job_title, min_salary,  
       CASE  
         WHEN min_salary < 4000 THEN min_salary * 1.2  
         WHEN min_salary BETWEEN 4000 AND 6000 THEN min_salary * 1.1  
         ELSE min_salary  
       END AS 최소급여변경  
FROM jobs;
```

- RANK(), DENSE_RANK(), ROW_NUMBER() 함수
 - RANK(): 공통 순위는 건너뛰어 다음 순위를 출력하는 순위 함수
 - DENSE_RANK(): 공통 순위를 건너뛰지 않고, 다음 순위를 출력하는 순위 함수
 - ROW_NUMBER(): 공통 순위 없이 순위를 출력하는 순위 함수

```
SELECT first_name, salary,  
       RANK() OVER(ORDER BY salary DESC) RANK,  
       DENSE_RANK() OVER(ORDER BY salary DESC) DENSE_RANK,  
       ROW_NUMBER() OVER(ORDER BY salary DESC) ROW_NUMBER  
FROM employees;
```

[실습] 숫자, 날짜 함수

- jobs 테이블에서 min_salary 값을 30으로 나눈 값의 올림값과 내림값을 조회

- jobs 테이블에서 max_salary 값을 30으로 나눈 값을 소수점 둘째 자리에서 반올림한 값과 정수 첫째 자리에서 반올림한 값을 조회

- jobs 테이블에서 max_salary 값을 30으로 나눈 값을 소수점 셋째 자리에서 절삭한 값과 정수 둘째 자리에서 절삭한 값을 조회

- 현재 날짜와 현재 날짜에서 한 달 뒤의 날짜를 조회

- 현재 날짜와 현재 날짜에서 돌아오는 월요일과 금요일의 날짜를 조회

[실습] 변환 함수

- 현재 날짜의 월/일과 요일을 조회

- 현재 시간의 오전 또는 오후 그리고 시:분을 조회

- 2021년 01월 01일의 요일을 조회

[실습] 일반 함수

- employees 테이블에서 salary, commission_pct, 그리고 salary에 commission_pct를 곱한 값을 salary에 반영하고, commission_pct 정렬 순으로 조회
(commission_pct가 null인 경우에는 salary 그대로 반영)

- employees 테이블에서 first_name, last_name, department_id, salary 그리고 department_id가 50인 경우 salary를 10% 증가시킨 값을 '급여인상', department_id가 100인 경우 salary를 10% 감소시킨 값을 '급여감소'로 조회

- jobs 테이블에서 job_title에 Manager가 들어간 데이터 중에 job_title, max_salary, 그리고 max_salary가 20000 이상인 것은 '상위', 10000에서 20000 사이인 것은 '중위', 그 밖에는 '하위'로 표기하는 '급여등급'을 조회



이수안 컴퓨터 연구소

suan computer laboratory