

3-2. SQL 함수2 – 형변환 및 기타 함수

홍형경

chariehong@gmail.com

2020.06

1. 형변환 함수

. 형변환

- 문자,숫자,날짜형 데이터 각각을 다른 데이터 형으로 변환
- 문자형 \longleftrightarrow 숫자형 (예, '123' \longleftrightarrow 123)
- 문자형 \longleftrightarrow 날짜형 (예, '2019-01-01' \longleftrightarrow 2019-01-01)

. 형변환은 묵시적 형변환과 명시적 형변환이 있음

1. 형변환 함수

- **묵시적**(Implicit) 형변환 : 오라클이 자동으로 형변환

```
SELECT *  
  FROM employees  
 WHERE hire_date = '2003-06-17';
```

- 묵시적 형변환 기본 규칙
 - 숫자형 문자와 숫자 비교 시, 문자를 숫자로 변환해 비교
 - 날짜형 문자와 숫자 비교 시, 문자를 날짜로 변환해 비교

1. 형변환 함수

· 명시적 형변환

- 형변환 함수를 사용해 사용자가 직접 형변환
- 가급적 명시적 형변환을 해야 함

· 명시적 형변환을 해야 하는 이유

- 가독성이 좋음
- 묵시적 형변환은 성능 상 좋지 않음
- 묵시적 형변환은 가끔 원치 않는 형태의 값을 반환할 수 있음 (nls_date_format 설정)
- 오라클 버전에 따라 묵시적 형변환 내부 알고리즘이 달라질 가능성이 있음

1. 형변환 함수

함수	기능	사용예	반환값
TO_NUMBER (char)	char을 숫자로 변환	SELECT TO_NUMBER('123') FROM DUAL;	123
TO_CHAR (n, number_format)	숫자인 n을 number_format에 맞게 문자로 변환, number_format은 생략 가능	SELECT TO_CHAR(12345, '99,999') FROM DUAL;	'12,345'
TO_CHAR (date, date_format)	날짜인 date를 date_format에 맞게 문자로 변환, date_format은 생략 가능	SELECT TO_CHAR(SYSDATE, 'YYYY-MM-DD HH24:MI:SS') FROM DUAL;	'2020-06-15 10:52:01'
TO_DATE (char, date_format)	문자 char을 date_format에 맞게 날짜로 변 환, date_format은 생략 가능	SELECT TO_DATE('2020-06- 15 23:52:01', 'YYYY-MM- DD HH24:MI:SS') FROM DUAL;	2020-06-15 23:52:01

1. 형변환 함수

· 숫자 변환 형식 (number_format)

형식	설명	사용예	결과
, (coma)	coma 표시	TO_CHAR(123456, '999,999')	123,456
. (소수점)	소수점 표시	TO_CHAR(123456.4, '999,999.9')	123,456.4
9	한 자리 숫자를 나타내는데, 실제 숫자의 자리 수와 같거나 크게 명시해야 함	TO_CHAR(123456, '999,999')	123,456

1. 형변환 함수

. 날짜 변환 형식

형식	설명	사용예	결과
YYYY YYY, YY, Y	연도 표시	TO_CHAR(SYSDATE, 'YYYY')	2020
MONTH, MON	월 표시	TO_CHAR(SYSDATE, 'MONTH')	6월
MM	월을 01, 02, ..., 12 형태로 표현	TO_CHAR(SYSDATE, 'MM')	06
D	주중 일자를 1~7까지 숫자로 표현, 1(일요일), 7(토요일)	TO_CHAR(SYSDATE, 'D')	2
DAY	주중 일자를 요일로 표현	TO_CHAR(SYSDATE, 'DAY')	월요일
DD	일을 01, 02, ..., 31 형태로 표현	TO_CHAR(SYSDATE, 'DD')	15
DDD	일을 001, 002, ..., 365 형태로 표현	TO_CHAR(SYSDATE, 'DDD')	167

1. 형변환 함수

. 날짜 변환 형식

형식	설명	사용예	결과
DL	일을 요일까지 표현	TO_CHAR(SYSDATE, 'DL')	2020년 06월 15일 월요일
HH, HH12	시간을 01, 02, ..., 12 형태로 표현	TO_CHAR(SYSDATE, 'HH')	10
HH24	시간을 24시간 단위로 표현	TO_CHAR(SYSDATE, ' HH24')	22
MI	분을 01, 02, ..., 59 형태로 표현	TO_CHAR(SYSDATE, 'MI')	10
SS	초를 01, 02, ..., 59 형태로 표현	TO_CHAR(SYSDATE, 'SS')	24
WW	주를 01, 02, ..., 53 형태로 표현	TO_CHAR(SYSDATE, 'WW')	02
W	주를 해당월 몇 주차인지 표시	TO_CHAR(SYSDATE, 'W')	2

2. Null 처리 함수

· Null 값 처리를 위한 함수

형식	설명	사용예	결과
NVL (expr1, expr2)	expr1 값이 NULL이면 expr2를 반환	NVL(NULL, 'A')	'A'
NVL2 (expr1, expr2, expr3)	expr1 값이 NULL이면 expr3을, NULL이 아니면 expr2를 반환	NVL2(NULL, 'A', 'B')	'B'
COALESCE (expr1, expr2, ...)	expr1, expr2, expr3, 에서 첫 번째로 NULL이 아닌 값 반환	COALESCE(NULL, NULL, 'B', NULL)	'B'
NULLIF (expr1, expr2)	expr1과 expr2 값이 같으면 NULL을, 같지 않으면 expr1 반환	NULLIF('A', 'B')	'A'

3. 기타 함수

- **DECODE** (expr, val1, result1, val2, result2, ..., default_value)
 - expr값이 value1이면 result1, value2와 같으면 result2를 반환
 - 일치하는 값이 없으면 default_value 반환
 - 이 경우 default_value 생략시 Null 반환
 - 단순형 CASE 표현식과 동작 방식 동일
- **DECODE** 사용
 - CASE 표현식 제공 전에는 DECODE 함수를 많이 사용
 - CASE 표현식 등장 후 사용빈도가 줄어드는 추세
 - DECODE 가 CASE 표현식에 비해 코드 양은 적으나 가독성이 떨어짐

3. 기타 함수

- 집계함수
 - SUM, MAX, MIN, AVG 등 : 4차시 강의에서 설명
- 오라클에서 제공하는 기타 함수
 - 분석 함수
 - LOB, Collection 함수
 - 계층형 함수
 - 데이터 마이닝 함수
 - XML, JSON 함수
 - JSON 함수
- 사용자 정의 함수 : PL/SQL을 사용해 직접 함수 생성 가능

4. SQL 함수 실습

(1) 형변환 함수 – TO_NUMBER

```
SELECT TO_NUMBER('12345.6789'), TO_NUMBER(-12.0)
FROM DUAL;
```

TO_NUMBER('12345.6789')	TO_NUMBER(-12.0)
12345.6789	-12

4. SQL 함수 실습

(1) 형변환 함수 – TO_NUMBER

```
SELECT TO_NUMBER(1234)
FROM DUAL;
```

	 TO_NUMBER(1234)	
1	1234	

```
SELECT TO_NUMBER('ABC')
FROM DUAL;
```

```
ORA-01722: 수치가 부적합합니다
01722, 00000 - "invalid number"
*Cause:   The specified number was invalid.
*Action:  Specify a valid number.
```

4. SQL 함수 실습

(1) 형변환 함수 – TO_CHAR 함수 - 숫자를 문자로 변환

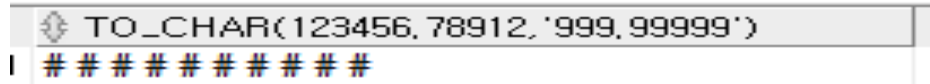
```
SELECT TO_CHAR(123456.78912),  
       TO_CHAR(123456.78912, '999,999.99999')  
FROM DUAL;
```

TO_CHAR(123456.78912)	TO_CHAR(123456.78912, '999,999.99999')
123456.78912	123,456.78912

4. SQL 함수 실습

(1) 형변환 함수 – TO_CHAR 함수 - 숫자를 문자로 변환

```
SELECT TO_CHAR(123456.78912, '999.99999')  
FROM DUAL;
```

A screenshot of a SQL query execution result. The top row shows the query: TO_CHAR(123456.78912, '999.99999'). The bottom row shows the result: 123456.78912, where the decimal part is formatted with trailing zeros to match the specified format mask.

TO_CHAR(123456.78912, '999.99999')
123456.78912

숫자 포맷을 지정할 때는, 자리수를 넉넉하게

4. SQL 함수 실습

(1) 형변환 함수 – TO_CHAR 함수 - 날짜를 문자로 변환

```
SELECT TO_CHAR(SYSDATE) D1,  
       TO_CHAR(SYSDATE, 'YYYY-MM-DD HH24:MI:SS') D2  
FROM DUAL;
```

D1	D2
2020-06-15 00:00:00	2020-06-15 22:10:05

4. SQL 함수 실습

(1) 형변환 함수 – TO_CHAR 함수 - 날짜를 문자로 변환

```
SELECT TO_CHAR(SYSDATE, 'DD') D1,  
       TO_CHAR(SYSDATE, 'DAY') D2,  
       TO_CHAR(SYSDATE, 'DDD') D3  
FROM DUAL;
```

D1	D2	D3
15	월요일	167

4. SQL 함수 실습

(1) 형변환 함수 – TO_CHAR 함수 - 날짜를 문자로 변환

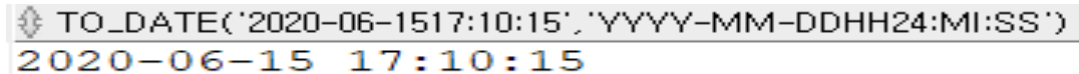
```
SELECT SYSDATE D1,  
       TO_CHAR(SYSDATE, 'W') D2,  
       TO_CHAR(SYSDATE, 'WW') D3,  
       TO_CHAR(TO_DATE('2020-06-15','YYYY-MM-DD'), 'W') D4,  
       TO_CHAR(TO_DATE('2020-06-15','YYYY-MM-DD'), 'WW') D5  
FROM DUAL;
```

D1	D2	D3	D4	D5
2020-06-15 00:00:00	3	24	3	24

4. SQL 함수 실습

(1) 형변환 함수 – TO_DATE 함수 - 문자를 날짜로 변환

```
SELECT TO_DATE('2020-06-15 17:10:15', 'YYYY-MM-DD HH24:MI:SS')  
FROM DUAL;
```



The screenshot shows a SQL query execution result. The query is `TO_DATE('2020-06-15 17:10:15', 'YYYY-MM-DD HH24:MI:SS')`. The result is displayed as `2020-06-15 17:10:15` in a table format.

TO_DATE('2020-06-15 17:10:15', 'YYYY-MM-DD HH24:MI:SS')
2020-06-15 17:10:15

4. SQL 함수 실습

(1) 형변환 함수 – TO_DATE 함수 - 문자를 날짜로 변환

```
SELECT TO_DATE('2020-06-15', 'YYYY-MM-DD HH24:MI:SS')  
FROM DUAL;
```

```
TO_DATE('2020-06-15', 'YYYY-MM-DDHH24:MI:SS')  
2020-06-15 00:00:00
```

4. SQL 함수 실습

(1) 형변환 함수 – TO_DATE 함수 - 문자를 날짜로 변환

```
SELECT TO_DATE('20200615', 'YYYY-MM-DD HH24:MI:SS')  
FROM DUAL;
```

```
TO_DATE('20200615', 'YYYY-MM-DDHH24:MI:SS')  
2020-06-15 00:00:00
```

```
SELECT TO_DATE('2020-06-15', 'YYYYMMDD HH24:MI:SS')  
FROM DUAL;
```

```
ORA-01843: 지정한 월이 부적합합니다.  
01843, 00000 - "not a valid month"  
*Cause:  
*Action:
```

4. SQL 함수 실습

(2) NULL 처리 함수 – NVL 함수






```
SELECT NVL(NULL, 'A'), NVL(NULL, 1), NVL(2, 3)
FROM DUAL;
```

NVL(NULL, 'A')	NVL(NULL, 1)	NVL(2, 3)
A	1	2

4. SQL 함수 실습

(2) NULL 처리 함수 – NVL2 함수

```
SELECT NVL2(NULL, 'A', 'B'), NVL2('A', 'B', 'C')  
FROM DUAL;
```

 NVL2(NULL, 'A... 	 NVL2('A', 'B', 'C')
 B	 B

4. SQL 함수 실습

(2) NULL 처리 함수 – COALESCE 함수

```
SELECT COALESCE(NULL, NULL, NULL, 'A', NULL, 'B')  
FROM DUAL;
```

COALESCE(NULL,NULL,NULL,'A',NULL,'B')
A

4. SQL 함수 실습

(2) NULL 처리 함수 – COALESCE 함수

```
SELECT COALESCE(NULL, NULL, NULL, NULL, NULL, NULL)
FROM DUAL;
```

⚡ COALESCE(NULL, NULL, NULL, NULL, NULL, NULL)
1 (null)

4. SQL 함수 실습

(2) NULL 처리 함수 – NULLIF 함수

```
SELECT NULLIF(100, 100), NULLIF(100, 200)
FROM DUAL;
```

⚙ NULLIF(100,100)	⚙ NULLIF(100,200)	
(null)	100	

4. SQL 함수 실습

(3) DECODE 함수

```
SELECT DECODE(1, 2, 3, 4, 5, 1, 7, 9)
       ,DECODE(1, 2, 3, 4, 5, 6, 7, 9)
       ,DECODE(1, 2, 3, 4, 5, 6, 7)
FROM DUAL;
```

DECODE(1,2,3,4,5,1,7,9)	DECODE(1,2,3,4,5,6,7,9)	DECODE(1,2,3,4,5,6,7)
7	9	(null)

학습정리

- 오라클에서는 TO_NUMBER, TO_CHAR, TO_DATE 형변환 함수를 사용해 숫자, 문자, 날짜형 데이터를 명시적으로 형변환 할 수 있다.
- TO_CHAR 함수는 숫자나 날짜를 문자형으로, TO_NUMBER 함수는 문자를 숫자로, TO_DATE 함수는 문자를 날짜형으로 형변환 하는 함수이다.
- 오라클에서 NULL 관련 함수로는 NVL, NVL2, COALESCE, NULLIF 가 있다.
- DECODE 함수는 단순형 CASE 표현식과 같은 동작을 하는 함수이다.

Quiz

1. 2019년 8월 20일을 기준으로 그 달의 마지막 날짜의 요일을 구하는 쿼리를 작성해 보세요.
(to_date, last_day, to_char 함수 사용)

Quiz

2. 다음 문장에서 CASE 표현식 부분을 NVL 함수로 변환해 보세요

```
SELECT employee_id, first_name, last_name, salary, commission_pct  
      ,CASE WHEN commission_pct IS NULL THEN salary  
            ELSE salary + (salary * commission_pct)  
            END real_salary  
FROM employees;
```

Quiz

3. Quiz 2번의 문제에 있는 CASE 표현식을 이번에는 decode 함수를 사용해 동일한 결과를 반환하도록 만들어 보세요.