

```
1 REM Author :
2 REM Date :
3 REM Objective : Chapter 4. Single-Row Function
4 REM Environment : CentOS 6.5, SQLGate 2010 for Oracle, Oracle 11g Enterprise
  Edition 11.2.0
5
6 REM   SQL function
7 1. Single-Row Function - 단일행 함수
8   1) 단일 행만 연산
9   2) 행 당 하나의 결과를 반환
10 2. Aggregate Function - 복수행 함수
11   1) 행 그룹을 조작하여 행 그룹 당 하나의 결과를 반환
12 3. 내부함수(단일행함수, 복수행함수)
13 4. 특징
14   1) 데이터에 계산을 수행한다.
15   2) 개별적인 데이터 항목을 수행할 수 있다.
16   3) 행의 그룹에 대한 결과를 조작할 수 있다.
17   4) 날짜와 숫자 형식을 변환할 수 있다.
18
19 REM 단일행 함수(Single Row function)
20 1. Syntax
21     function_name(column | expression [ arg1, arg2...])
22 2. 종류
23   1) 문자형함수
24   2) 숫자형함수
25   3) 날짜형 함수
26   4) 변환 함수
27   5) 일반 함수
28
29
30 REM 문자 함수
31
32 1. LOWER
33   1) 소문자로 변환
34   2) Syntax
35       LOWER(column | expression)
36
37       SELECT empno, ename
38       FROM emp
39       WHERE LOWER(ename) = 'scott';
40
41 2. UPPER
42   1) 대문자로 변환
43   2) Syntax
44       UPPER (column | expression)
45
46       SELECT empno, ename, deptno
```

```
47      FROM emp
48      WHERE ename = 'blake';
49
50      SELECT empno, ename, deptno
51      FROM emp
52      WHERE ename = UPPER('blake');
53
54 3. INITCAP
55   1) 첫문자를 대문자, 나머지 소문자
56   2) Syntax
57       INITCAP (column | expression)
58
59      SELECT deptno, INITCAP(dname), INITCAP(loc)
60      FROM dept;
61
62 4. CONCAT
63   1) 두개의 문자열을 합성
64   2) 아규먼트는 두개이어야
65   3) Syntax
66       CONCAT (column1 | expression1, column2 | expression2)
67
68      SELECT CONCAT('Oracle', 'Server') AS "연습"
69      FROM dual;
70
71      SELECT ename || ' is a ' || job AS "자기소개"
72      SELECT CONCAT(ename, job)
73      SELECT CONCAT(CONCAT(ename, ' is a '), job)
74      FROM emp
75      WHERE deptno = 10;
76
77 5. SUBSTR
78   1) 지정된 길이만큼의 문자열을 추출
79   2) Syntax
80       SUBSTR (column | expression, m[,n])
81   3) 위치 m에서 n개 지정된 문자를 반환
82   4) m이 음수면 문자 값의 끝부터 세며, n을 생략하면 끝까지 반환
83
84      SELECT SUBSTR(job, 1,6)
85      FROM emp;
86
87      SELECT SUBSTR('Java Programming', 6, 4)
88      FROM dual;
89
90      SELECT empno, ename, hiredate
91      FROM emp
92      WHERE '87' = SUBSTR(hiredate, 1, 2);
93      WHERE hiredate LIKE '87%';
```

```
94         WHERE hiredate >= '87/01/01';
95
96 6. LENGTH
97   1) 문자수 반환
98   2) Syntax
99       LENGTH (column | expression)
100
101       SELECT ename, length(ename)
102       FROM emp;
103
104 7. INSTR
105   1) 지정된 문자열의 위치를 숫자로 반환
106   2) Syntax
107       INSTR(column | expression, 'string', [,m],[n])
108
109       SELECT INSTR('Hello,World', ',W')
110       FROM dual;
111
112       SELECT INSTR('Oracle Programming', 'R')
113       FROM dual;
114
115       SELECT ename, INSTR(ename, 'A')
116       FROM emp
117       WHERE deptno = 20;
118
119 8. LPAD
120   1) 전체 폭이 n이 되도록 문자 값을 오른쪽으로 정렬하고 빈 곳을 지정한 string으로 채운다.
121   2) Syntax
122       LPAD(column | expression, n, 'string')
123
124 9. RPAD
125   1) 전체 폭이 n이 되도록 문자 값을 왼쪽으로 정렬하고 빈 곳을 지정한 string으로 채운다.
126   2) Syntax
127       RPAD(column | expression, n, 'string')
128
129       SELECT LPAD(sal, 5, '*'), RPAD(sal, 6, '*')
130       FROM emp;
131
132       SELECT ename, CONCAT(ename, job),
133       LENGTH(ename), INSTR(ename, 'A'),
134       LPAD(sal, 5, '*'), RPAD(sal, 6, '*')
135       FROM emp
136       WHERE SUBSTR(job, 1, 5) = UPPER('sales');
137
138 10. CHR
139   1) 숫자에 맞는 ASCII문자 출력
140
```

```
141      SELECT CHR(65)
142      FROM dual;
143
144      SELECT empno, ename || CHR(10) || job
145      FROM emp
146      WHERE deptno = 20;
147
148 11. ASCII
149     1) 해당 ASCII로 변환
150
151      SELECT ASCII('a')
152      FROM dual;
153
154 12. LTRIM
155     1) 정의된 문장에서 왼쪽에서 지정된 단어가 나타나면 제거
156
157      SELECT LTRIM('xyxXxyLAST WORD', 'xy')
158      FROM dual;
159
160 13. RTRIM
161     1) 정의된 문장에서 오른쪽에서 지정된 단어가 나타나면 제거
162
163      SELECT RTRIM('TURNERYxXxy', 'xy')
164      FROM dual;
165
166 14. REPLACE
167     1) 해당문자가 발견되면 지정된 문자로 변환
168     2) Syntax
169        REPLACE(column | expression, original, replace)
170
171      SELECT REPLACE('JACK and JUE', 'J', 'BL')
172      FROM dual;
173
174      SELECT job, REPLACE(job, 'A', '$')
175      FROM emp;
176
177 15. GREATEST
178     1) 주어진 단어 중 가장 높은 값
179
180      SELECT GREATEST('HARRY', 'HARIOT', 'HALORD')
181      FROM dual;
182
183 16. LEAST
184     1) 주어진 단어 중 가장 낮은 값
185
186      SELECT LEAST('HARRY', 'HARIOT', 'HALORD')
187      FROM dual;
```

188

## 189 17. NVL

190 1) 칼럼이 NULL이면 지정한 값으로 대치

191

```
192     SELECT NVL(sal, 0), NVL(ename, '*'), NVL(hiredate, SYSDATE)
193     FROM emp;
```

194

## 195 18. TRIM

196 1) LTRIM과 RTRIM을 결합

197 2) LEADING == LTRIM

198 TRAILING == RTRIM

199 BOTH == 둘다

200

```
201     SELECT ename, TRIM(LEADING 'A' FROM ename) AS TRIM
202     FROM emp
203     WHERE ename LIKE 'A%';
```

204

```
205     SELECT ename, TRIM(TRAILING 'N' FROM ename) AS TRIM
206     FROM emp
207     WHERE ename LIKE '%N';
```

208

```
209     SELECT ename, TRIM(BOTH 'A' FROM ename) AS TRIM
210     FROM emp
211     WHERE ename LIKE 'A%';
```

212

213

## 214 REM 숫자 함수

215

## 216 1. ROUND

217 1) 반올림, 반내림함수

218 2) Syntax

219 ROUND(column | expression, n)

220 3) 열, 표현식, 값을 소수점 n째 자리로 반올림

221 4) n을 지정하지 않은 경우 소수점 이하 값이 없어짐

222 5) n이 음수이면 소수점 왼쪽 수가 반올림

223

```
224     SELECT ROUND(45.925, 2), ROUND(45.925, 0), ROUND(45.925, -1)
225     FROM DUAL;
```

226

## 227 2. TRUNC

228 1) 명시된 숫자를 절삭하는 함수

229 2) 열, 표현식, 값을 소수점 n째 자리까지 남기고 버린다.

230 3) Syntax

231 TRUNC (column | expression, n)

232

```
233     SELECT TRUNC(45.925, 2), TRUNC(45.925), TRUNC(45.925, -1)
234     FROM dual;
```

```
235
236      SELECT ROUND(456.789, 2), TRUNC(456.789, 2)
237      FROM dual;
238
239 3. MOD(m, n)
240    1) 숫자의 나머지를 구하는 함수
241    2) 값n을 값m로 나눈 나머지를 반환
242
243      SELECT ename, sal, comm, MOD(sal, comm)
244      FROM emp
245      WHERE job = 'SALESMAN';
246
247      SELECT 10 / 3, MOD(10, 3)
248      FROM dual;
249
250      SELECT sal, MOD(sal, 30)
251      FROM emp;
252
253 4. ABS
254    1) 숫자 값을 절대값으로 바꾼다.
255
256      SELECT ABS(-15)
257      FROM dual;
258
259 5. CEIL
260
261      SELECT CEIL(15.7)
262      FROM dual;
263
264 6. FLOOR
265
266      SELECT FLOOR(15.7)
267      FROM dual;
268
269 7. POWER
270
271      SELECT POWER(3,2)
272      FROM dual;
273
274 8. SIGN
275    1) 주어진 수가 양수이면 1, 0이면 0, 음수이면 -1
276
277      SELECT SIGN(-15)
278      FROM dual;
279
280 9. NVL2
281    1) 해당 column이 NULL이면 express2의 값을, NULL이 아니면 express1의 값을 리턴해줌
```

```

282 2) Syntax
283     NVL2(column, express1, express2)
284
285     SELECT empno, ename, NVL(comm, 0), NVL2(comm, comm * 1.1, 0)
286     FROM emp;
287
288 10. NULLIF
289 1) express1과 express2이 값을 비교하여 값이 같으면 NULL, 다르면 express1을 리턴
290 2) Syntax
291     NULLIF(express1, express2)
292
293     SELECT empno, ename, NULLIF(comm, 0)
294     FROM emp;
295
296 11. COALESCE
297 1) express1의 값이 NULL값이 아니면 express1의 값을 출력하고 NULL 값이면 express2의
298    값을 출력하고, express1의 값과 express2의 값 모두 NULL이면 expressN의 값을 출력
299 2) NULL이 아닌 첫번째 칼럼의 값을 돌려줌
300 3) Syntax
301     COALESCE(express1, , , , expressN)
302
303     SELECT comm, sal, COALESCE(comm, sal)
304     FROM emp;
305
306     SELECT COALESCE(comm, 100)
307     FROM emp;
308
309 12. DECODE
310 1) Syntax
311     DECODE(검색칼럼, 조건1, 결과값1, 조건2, 결과값2, ....., 기본값);
312 2) IF ~ THEN ~ ELSEIF ~ ELSEIF ~ END 와 비슷
313
314     SELECT job, sal, DECODE(job, 'ANALYST', sal * 1.1,
315                                'CLERK', sal *
316                                1.15,
317                                'MANAGER', sal *
318                                1.2, sal) AS
319     "SALARY"
320
321     FROM emp;
322
323     SELECT COUNT(DECODE(TO_CHAR(hiredate, 'MM'), '01', 1)) AS "1월",
324                COUNT(DECODE(TO_CHAR(hiredate, 'MM'), '02', 1)) AS "2월",
325                COUNT(DECODE(TO_CHAR(hiredate, 'MM'), '03', 1)) AS "3월",
326                COUNT(DECODE(TO_CHAR(hiredate, 'MM'), '04', 1)) AS "4월",
327                COUNT(DECODE(TO_CHAR(hiredate, 'MM'), '05', 1)) AS "5월",
328                COUNT(DECODE(TO_CHAR(hiredate, 'MM'), '06', 1)) AS "6월",
329                COUNT(*) AS "전체"

```

```

325      FROM emp
326      WHERE TO_CHAR(hiredate, 'MM') >= '01' AND
327             TO_CHAR(hiredate, 'MM') <= '06';
328
329 13. CASE
330     1) 결과는 DECODE 문과 동일
331     2) DECODE는 실행될 때 내부적으로 SORTING을 함으로써 성능 저하 현상이 발생, 이 문제를 해
        결가능
332     3) Oracle 8 부터 추가됐슴
333         SELECT job, sal,
334                CASE WHEN job = 'ANALYST' THEN sal * 1.1
335                      WHEN job = 'CLERK'   THEN sal * 1.15
336                      WHEN job = 'MANAGER' THEN sal * 1.2
337                      ELSE sal
338                END AS "SALARY"
339         FROM emp;
340
341     SELECT COUNT(CASE WHEN TO_CHAR(hiredate, 'MM') = '01' THEN
342     COUNT(*) END) AS "1월",
343     COUNT(CASE WHEN TO_CHAR(hiredate, 'MM') = '02' THEN
344     COUNT(*) END) AS "2월",
345     COUNT(CASE WHEN TO_CHAR(hiredate, 'MM') = '03' THEN
346     COUNT(*) END) AS "3월",
347     COUNT(CASE WHEN TO_CHAR(hiredate, 'MM') = '04' THEN
348     COUNT(*) END) AS "4월",
349     COUNT(CASE WHEN TO_CHAR(hiredate, 'MM') = '05' THEN
350     COUNT(*) END) AS "5월",
351     COUNT(CASE WHEN TO_CHAR(hiredate, 'MM') = '06' THEN
352     COUNT(*) END) AS "6월",
353     COUNT(*) AS "전체"
354     FROM emp
355     GROUP BY hiredate
356     HAVING TO_CHAR(hiredate, 'MM') >= '01' AND
357            TO_CHAR(hiredate, 'MM') <= '06';
358
359 REM 날짜 함수
360
361 1. 날짜데이터
362     1) Oracle에서 유효한 날짜는 B.C. 4712/01/01 ~ A.D. 9999/12/31
363     2) 오라클은 세기, 년, 월, 일, 시, 분, 초를 내부 숫자로 저장한다.
364     3) 형식은 NLS_DATE_FORMAT 에 따라 달라진다.
365
366 2. SYSDATE
367     1) 현재 날짜 및 시간
368
369     SELECT SYSDATE

```



```
365         FROM dual;
366
367 3. 날짜 계산
368     1) 데이터베이스는 날짜를 내부 숫자로 저장한다.
369     2) 산술 연산자를 사용하여 계산 가능
370
371         SELECT ename, (SYSDATE - hiredate) / 7 WEEKS
372         FROM emp
373         WHERE deptno = 10;
374
375         SELECT SYSDATE + 5
376         FROM dual;
377
378 4. 날짜함수
379     1) 모든 날짜 함수는 DATE 데이터 유형 값 반환
380     2) MONTHS_BETWEEN()만 숫자 값 반환
381
382 5. MONTHS_BETWEEN
383     1) 두 날짜 간 달 수
384     2) 결과는 음수가 나올 수 있다.
385     3) Syntax
386         MONTHS_BETWEEN(date1, date2)
387
388         SELECT MONTHS_BETWEEN(SYSDATE, SYSDATE + (5 * 31))
389         FROM dual;
390
391         SELECT hiredate, MONTHS_BETWEEN(SYSDATE, hiredate)
392         FROM emp
393         WHERE empno = 7782;
394
395         SELECT ename, hiredate,
396                TRUNC(MONTHS_BETWEEN(SYSDATE, hiredate))
397         FROM emp
398         WHERE deptno = 20;
399
400 6. ADD_MONTHS
401     1) 날짜에 달 수 더하기
402
403         SELECT ADD_MONTHS(SYSDATE, 5)
404         FROM dual;
405
406         SELECT hiredate, ADD_MONTHS(hiredate, 1)
407         FROM emp
408         WHERE empno = 7782;
409
410 7. NEXT_DAY
411     1) 지정한 날짜의 다음 날
```

```
412 2) 명시된 요일의 돌아오는 날짜를 계산
413 3) 요일을 숫자로 기술가능(1:SUNDAY, 2: MONDAY...)
414 4) 만일 NLS_LANG이 한글이면 한글요일도 가능
415 5) Syntax
416     NEXT_DAY(date, 'string')
417
418     SELECT NEXT_DAY(SYSDATE, '금요일')
419     FROM dual;
420
421     SELECT SYSDATE, NEXT_DAY(SYSDATE, 'MONDAY'),
422     SELECT SYSDATE, NEXT_DAY(SYSDATE, 2), NEXT_DAY(SYSDATE, '월요일')
423     FROM dual;
424
425 8. LAST_DAY
426 1) 해당 달의 마지막 날
427
428     SELECT LAST_DAY(SYSDATE)
429     FROM dual;
430
431     SELECT hiredate, LAST_DAY(hiredate)
432     FROM emp
433     WHERE empno = 7782;
434
435 9. ROUND
436 1) 날짜 반올림
437
438     SELECT ROUND(TO_DATE('07/07/14' , 'YY/MM/DD'), 'YEAR')
439     FROM dual;
440
441 10. TRUNC
442 1) 날짜 버림
443
444     SELECT TRUNC(TO_DATE('07/07/14' , 'YY/MM/DD'), 'YEAR')
445     FROM dual;
446
447     SELECT empno AS "사원 번호" , hiredate AS "입사날짜",
448     MONTHS_BETWEEN(SYSDATE, hiredate) AS "근무한 달 수",
449     ADD_MONTHS(hiredate, 6) AS "6개월 검토일",
450     NEXT_DAY(hiredate, '금요일') AS "입사후 첫 금요일",
451     LAST_DAY(hiredate) AS "입사한 달 마지막 날"
452     FROM emp;
453
454
455 REM 변환함수
456 1. TO_CHAR : 숫자, 날짜 타입의 Data를 VARCHAR2 타입으로 변환
457 2. TO_NUMBER : 숫자를 포함하는 문자 String을 Number타입으로 변환
458 3. TO_DATE : 문자 String 을 날짜 타입으로 변환
```

```

459
460 4. 날짜 형식 요소
461     1) MM : 달 수 (ex : 10)
462     2) MON : 월 이름을 3자리 문자로 표현 (ex : JAN)
463     3) MONTH : 월 이름(ex : JANUARY)
464     4) DD : 날짜(ex : 14)
465     5) D : 주의 일(ex : 3)
466     6) DY : 요일 이름을 3자리 문자로 표현 (ex : SUN)
467     7) DAY : 요일 이름(ex : SUNDAY)
468     8) YYYY : 년도 4자리 수(ex : 2007)
469     9) YY : 년도 마지막 2자리 수(ex : 07)
470
471 5. REM 시간 형식 요소
472     1) HH or HH12 : 시간을 12시간 단위로 표현(ex : 1~ 12)
473     2) HH24 : 시간을 24시간 단위로 표현 (ex : 1~ 24)
474     3) MI : 분 (ex : 1~ 59)
475     4) SS : 초 (ex : 1~ 59)
476     5) AM or PM : 정오 지시자
477
478 6. 숫자 형식 요소
479     1) 9 : 숫자 (ex : 9999 => 1534)
480     2) 0 : 자리 수가 비면 0으로 채움(ex : 09999 => 01534)
481     3) $ : 금액에 $를 표시해줌 (ex : $9999 => $1534)
482     4) . : 명시한 위치에 소수점을 표시함(ex : 99999.99 => 1534.00)
483     5) , : 명시한 위치에 콤마를 표시함 (ex : 999,999 => 1,534)
484
485 7. 문자 변환 함수 TO_CHAR
486     1) 날짜, 숫자, 문자열을 VARCHAR2로 변환
487     2) Syntax
488         TO_CHAR(date, '출력형식')
489
490         SELECT SYSDATE, TO_CHAR(SYSDATE, 'D')
491         FROM dual;
492
493         SELECT ename, TO_CHAR(hiredate, 'DD-MM-YY') AS HIREDATE
494         FROM emp;
495
496         SELECT ename, TO_CHAR(hiredate, 'fmDD-MM-YY') AS HIREDATE
497         FROM emp;
498
499         --사원테이블에서 10번 부서의 사원들 중 입사일사를 "1 May 1981"과 "1981년 5월
500         1일"의 형태로 출력하시오.
501         SELECT deptno, hiredate,
502                TO_CHAR(hiredate, 'DD MONTH YYYY') AS t_eng,
503                TO_CHAR(hiredate, 'YYYY "년" MM "월" DD "일" ') AS
504                t_hiredate
505         FROM emp

```

```

504         WHERE deptno = 10;
505
506         SELECT TO_CHAR(SYSDATE, 'HH24 : MI : SS')
507         FROM dual;
508
509         SELECT TO_CHAR(SYSDATE, 'AM HH : MI' )
510         FROM dual;
511
512         SELECT TO_DATE('01-JUL-99', 'DD-MM-YY')
513         FROM dual;
514
515         SELECT TO_DATE('1:30', 'HH24:MI')
516         FROM dual;
517
518     3) "of" : 인용부호 내의 스트링을 화면에 표시
519         SELECT SYSDATE, TO_CHAR(SYSDATE, 'DD "of" MONTH')
520         FROM dual;
521
522     4) SP : 명시한 수를 문자로 표현(ex : ONE, TWO, THREE)
523     5) TH : 서수로 표현 (ex : 3TH)
524     6) SPTH : 명시한 서수 (ex : FIRST, SECOND)
525         SELECT SYSDATE, TO_CHAR(SYSDATE, 'DDSPTH')
526         FROM dual;
527     7) NUMBER --> VARCHAR2
528         SELECT TO_CHAR(12506, '$999,999')
529         FROM dual;
530
531         SELECT deptno, ename, sal, TO_CHAR(sal, '$999,999')
532         FROM emp
533         WHERE deptno = 20;
534
535     8. 숫자 변환 함수 (TO_NUMBER)
536         SELECT TO_NUMBER('100')
537         FROM dual;
538
539         SELECT TO_NUMBER('45') - TO_NUMBER('32')
540         FROM dual;
541
542     9. TO_DATE
543     1) 숫자형, 문자형을 날짜형으로 변환하는 함수
544
545         SELECT TO_CHAR(TO_DATE(20120906123400, 'YYYYMMDDHHMISS'),
546                        'YYYY "년" MM "월" DD "일", HH "시" MI "분",
547                        SS "초"')
548         FROM dual;
549
550         SELECT empno, ename, hiredate

```

```
550      FROM emp
551      WHERE hiredate = TO_DATE(19810222, 'YYYYMMDD');
```