



# Day(BE)120\_함수, 트랜잭션

📅 Date	@2022/12/28
🕒 작성일시	@December 28, 2022 9:11 AM
📌 강의 번호	US101
☰ Skill	Data_SQL
👤 Instructor	조성희
☑ Sum up	☑
☑ Review	☐

GitHub - Kray273/SQLLab

Day23 <https://www.notion.so/Day23-2dac3079bfe044e6ac6a0e905d76b237>

Day24 <https://www.notion.so/Day24-690b518731314b0ba14207dfd37c4a58>

Day25 ...

<https://github.com/Kray273/SQLLab>

Kray273/SQLLab



1

Contributor

0

Issues

0

Stars

0

Forks



Review

함수

함수와 데이터 타입

트랜잭션

## ▼ Review



Review

Sol

## ▼ 함수



키워드를 통해 정해진 기능을 구현

함수 - 주어진 언어내부 연산자 제공되지 않은 기능 구현  
결과1개 =이름(매개변수)  
MARIA DB 내장함수  
집계함수(GROUP)--> 여러개 레코드 모아서 1개  
SCALAR 함수--> 1개 레코드 결과 1개  
문자 날짜 숫자 변환 조건 null처리

## ▼ 함수와 데이터 타입



## MariaDB가 제공하는 함수와 데이터 타입

```
SELECT 1234.5678, ROUND(1234.5678), ROUND(1234.5678,1), ROUND(1234.5678,2)
```

-- 반올림 매개변수는 출력하고 싶은 소수점 자리 지정  
-- 숫자 사용에 자주 나옴.

결과 #1 (1r x 4c)				
1234.5678	ROUND(1234.5678)	ROUND(1234.5678,1)	ROUND(1234.5678,2)	ROUND(1234.5678,3)
1,234.5678	1,235	1,234.6	1,234.57	1,234.568

```
-- 문자
SELECT phone_number, REPLACE(phone_number,'.','_') FROM employees
```

-- replace()는 문자를 변경  
-- 매개변수 1은 변경을 원하는 문자, 매개변수 2는 변경을 하고 싶은 문자.

employees (107r x 2c)	
phone_number	REPLACE(phone_number,'.','_')
515.123.4567	515_123_4567
515.123.4568	515_123_4568

```
-- 성 - last_name, 이름 - first_name
SELECT CONCAT('성 - ', last_name, ', 이름 - ', first_name) FROM employees
```

-- oracle : '성 - ' \ last\_name \ ' , 이름 - ' \ first\_name  
-- java : '성 - ' + last\_name + ' , 이름 - ' + first\_name

-- 자바 : char(' '), string("")  
-- database : char/varchar 문자열 표현('')  
-- mariadb : char/ varchar 256바이트(한글 3바이트), 문자열 표현('','")  
-- mariadb : text 65536문자, 문자열 표현 ('','")

employees (107r x 1c)
CONCAT('성 - ', last_name, ', 이름 - ', first_name)
성 - King, 이름 - Steven
성 - Kochhar, 이름 - Neena

```
-- 숫자 함수
-- 숫자형 데이터 타입
-- 정수 : tinyint(1byte), smallint(2byte), int(4byte), long(8byte)_주로 상품코드 사용
-- 실수 : float(4byte), double(8byte)
-- 사용자 구성 : decimal(10,3) _ 총자리수, 소수점 이하 자리수

-- 평균급여
SELECT AVG(salary), ROUND(AVG(salary),-2), TRUNCATE(AVG(salary),2) FROM employees;
```

-- avg() 평균  
-- round() 반올림 매개변수에 음수는 정수 위치에서 반올림  
-- truncate() 버림, 매개변수는 소수점 자리수

employees (1r x 3c)			
AVG(salary)	ROUND(AVG(salary),-2)	TRUNCATE(AVG(salary),2)	TRUNCATE(AVG(salary),3)
6,461.831776	6,500	6,500	6,461.832

```
-- + - * / 는 있지만 % 나머지 연산자는 없다
-- mod()를 통해 구현
SELECT 1234, MOD(1234, 3), MOD(1234, 7)
```

결과 #1 (1r x 3c)			
1234	MOD(1234, 3)	MOD(1234, 7)	
1,234	1		

```
-- 짝수 사번을 가진 사원의 사번, 이름 조회
SELECT e.employee_id, e.last_name FROM employees e
WHERE MOD(e.employee_id, 2) = 0;
```

employees (54r x 2c)		
employee_id	last_name	
100	King	
102	De Haan	
104	Ernst	

```
SELECT e.employee_id, e.last_name,
       case when MOD(e.employee_id,2) = 0 then '짝수사번'
       ELSE '홀수사번'
       END '사번의 성격'
FROM employees e;
```

employees (107r x 3c)		
employee_id	last_name	사번의 성격
100	King	짝수사번
101	Kochhar	홀수사번
102	De Haan	짝수사번

```
-- 256자리 : char(고정길이) / varchar(가변길이)
-- 65536 : TEXT
SELECT 'abcdef', LENGTH('abcdef'), LENGTH('가나다라마');
-- length() 문자열의 길이 리턴
SELECT "ABCDEF", "가나다라마", CHAR_LENGTH("ABCDEF"), LENGTH("ABCDEF"),
       BIT_LENGTH("ABCDEF"), CHAR_LENGTH("가나다라마"), LENGTH("가나다라마"),
       BIT_LENGTH("가나다라마");
```

결과 #1 (1r x 3c)		
abcdef	LENGTH('abcdef')	LENGTH('가나다라마')
abcdef	6	

결과 #1 (1r x 4c)			
ABCDEF	가나다라마	CHAR_LENGTH('ABCDEF')	LENGTH('ABCDEF')
ABCDEF	가나다라마	6	6

CHAR_LENGTH('가나다라마')		
CHAR_LENGTH('가나다라마')	LENGTH('가나다라마')	BIT_LENGTH('가나다라마')
5	15	

```
-- 문자 찾기(날짜도 문자열 함수 사용)
-- 오라클 + mysql => 함수 + mariadb

SELECT ELT(2, '일이', '둘', '셋');
-- elt()는 해당 인덱스에 위치한 데이터 리턴
-- index 1부터 시작, 자바는 0부터
SELECT FIELD('일이', '일이', '둘', '셋');
-- field()는 해당하는 문자의 인덱스값 리턴
SELECT FIND_IN_SET('일이', '일이,삼사,오육 ');
-- FIND_IN_SET()는 문자열안에서 해당 글자의 위치 리턴
-- 분리 및 구분자 필요
SELECT INSTR('일이삼사오육', '삼');
-- INSTR() 분리자 없이 해당 문자가 문자열의 해당하는 위치 리
SELECT LOCATE('삼', '일이삼사오육');
-- INSTR()와 비슷
SELECT SUBSTR('김상형의 SQL 정복', 5, 4);
-- SUBSTR() 문자열에서 원하는 위치의 문자 리턴
-- substring() 동일 가능
SELECT NOW(); -- now() 현재시간 리턴
```

결과 #1 (1r x 1c)	
ELT(2, '일이', '둘', '셋')	
둘	

결과 #1 (1r x 1c)
FIELD('일이','일이','둘','셋')

결과 #1 (1r x 1c)	
FIND_IN_SET('일이', '일이,삼사,오육')	
	1

결과 #1 (1r x 1c)
INSTR('일이삼사오육', '삼')
3

결과 #1 (1r x 1c)	
LOCATE('삼','일이삼사오육')	
	3

결과 #1 (1r x 1c)
SUBSTR('김상형의 SQL 정복',5,3)
SQL

결과 #1 (1r x 1c)
NOW()
2022-12-28 13:14:19

employees (107r x 3c)		
hire_date	입사년도	SUBSTR(e.hire_date, 6, 2)
2003-06-17 00:00:00	2003	06
2003-09-21 00:00:00	2003	09

```
SELECT e.hire_date 입사시간, SUBSTRING(e.hire_date, 1, 4) 입사년도,
       SUBSTR(e.hire_date, 6, 2)
FROM employees e;
```

```
-- 클라이언트 암호 입력 -> 스프링, 자바 -> db 테이블 저장
-- 다양한 위치에서 암호화 가능
SET @pw = 'abc가나다123';
SELECT @pw, INSERT(@pw, 2, 4, "****");
SELECT repeat('*', CHAR_LENGTH(@pw));
```

결과 #1 (1r x 2c)	
@pw	INSERT(@pw, 2, 4, "****")
abc가나다123	a****다123

```
SET @pw = 'abc가나다12345';
SELECT @pw, INSERT(@pw, 1, CHAR_LENGTH(@pw), repeat('*', CHAR_LENGTH(@pw)))
```

결과 #1 (1r x 1c)
repeat('*', CHAR_LENGTH(@pw))
*****

@pw	INSERT(@pw, 1, CHAR_LENGTH(@pw), repeat('a', 10))
abc7H4G12345	*****

```
SELECT 'abcdefg', LEFT('abcdefg',3), RIGHT('abcdefg',3);
```

abcdefg	LEFT('abcdefg',3)	RIGHT('abcdefg',3)
abcdefg	abc	efg

```
SELECT 'abcSEdSdr', UPPER('abcSEdSdr'), LOWER('abcSEdSdr');
-- mariadb는 대소문자를 구분하지 않기에 문자열을 비교하기 위해서는 자바에서 실시
-- 자바키보드 입력, "Maria".ToUpperCase() - db upper('mArIa')
```

abcSEdSdr	UPPER('abcSEdSdr')	LOWER('abcSEdSdr')
abcSEdSdr	ABCSEdSDR	abcseDSdr

```
-- PAD() 글자추가
SELECT 'abc' , LPAD('abc',10," .") , RPAD ('abc',10,'.');
```

abc	LPAD('abc',10,".")	RPAD ('abc',10,'.')
abc	...abc	abc.....

```
-- trim 문자열 잘라내기
SET @pw = '김상형의 SQL 정복';
SELECT CHAR_LENGTH(@pw), CHAR_LENGTH(LTRIM(@pw)), CHAR_LENGTH(RTRIM(@pw));
SET @kk = 'ㅋㅋㅋㅋ지워줘ㅋㅋㅋㅋㅋㅋ';
SELECT CHAR_LENGTH(@kk), TRIM(LEADING 'ㅋ' FROM @kk),
TRIM(TRAILING 'ㅋ' FROM @kk), TRIM( BOTH 'ㅋ' FROM @kk);
```

CHAR_LENGTH(@pw)	CHAR_LENGTH(LTRIM(@pw))	CHAR_LENGTH(RTRIM(@pw))
26	16	16
TRIM(LEADING 'ㅋ' FROM @kk)	TRIM(TRAILING 'ㅋ' FROM @kk)	TRIM( BOTH 'ㅋ' FROM @kk)
12	12	12

```
-- CREATE TABLE / ALTER TABLE
-- 문자 CHAR / VARCHAR
-- 숫자 INT / DOUBLE / DECIMAL(10,0)
-- 날짜 DATE / TIME / DATETIME

-- 날짜 문자 숫자 입력
-- 날짜 <--> 문자자동형변환 <--> 숫자
SELECT MIN(e.employee_id), MAX(e.employee_id) FROM employees e;
INSERT INTO emp_copy VALUES(300, '갈종', '취', 10000, NOW(), 200);
INSERT INTO emp_copy VALUES('301', '갈종', '취', 10000, NOW(), 200);
INSERT INTO emp_copy VALUES('301', '갈종', '취', '10000a', NOW(), 200); -- error
INSERT INTO emp_copy VALUES(302, '갈종', '취', '10000', CURDATE(), 200);
INSERT INTO emp_copy VALUES(303, '대리', '겸', '10000', '2022-12-25 00:00:00', 200);
INSERT INTO emp_copy VALUES(304, '대리', '곰', '10000', DATE_SUB(NOW(), INTERVAL 4 YEAR), 200);
INSERT INTO emp_copy VALUES(305, '대리', '곰', '10000', '20221212', 200);
INSERT INTO emp_copy VALUES(306, '대리', '곰', '10000', 20201111, 200);

SELECT e.employee_id, e.first_name, e.hire_date FROM emp_copy e ORDER BY e.employee_id DESC;
```

employee_id	first_name	hire_date
306	대리	2020-11-11 00:00:00
305	대리	2022-12-12 00:00:00
304	대리	2018-12-28 14:38:19
303	대리	2022-12-25 00:00:00
302	갈종	2022-12-28 00:00:00
301	갈종	2022-12-28 14:29:53
300	갈종	2022-12-28 14:29:06

```
SELECT 100 + 200, '100' + '200', CONCAT('100' , '200');
```

100 + 200	'100' + '200'	CONCAT('100' , '200')
300	300	100200

```
-- 명시적 형변환
-- 숫자 ROUND(123.444) TRUNCATE(123.444,0)
-- CAST(expr AS type) , CONVERT(expr,TYPE), FORMAT(num, decimal_position[, locale])
SELECT AVG(e.salary), CAST(AVG(e.salary) AS SIGNED INTEGER),
       CONVERT(AVG(e.salary) , SIGNED INTEGER),
       FORMAT(AVG(e.salary), 0)
FROM employees e;
```

AVG(e.salary)	CAST(AVG(e.salary) AS SIGNED INTEGER)
6,461.831776	

  

CONVERT(AVG(e.salary) , SIGNED INTEGER)	FORMAT(AVG(e.salary), 0)
6,462	6,462

```
-- ifnull , nullif
IFNULL(칼럼명 , NULL시 대체값);
NULLIF(값1,값2); -- 값1 과 2과 같으면 NULL, 다르면 표현식 1 리턴

IF(T/F, T, F);

SELECT if(20>10 , '크다', '작다');

-- 사원들 commission_pct null 사원들, 그렇지 않은 사람들
SELECT e.first_name 이름, e.commission_pct,
       if(e.commission_pct IS NULL, '못받는다.', '받는다') 보너스유무
FROM employees e;
```

if(20>10 , '크다', '작다')

크다

employees (107r × 3c)		
이름	commission_pct	보너스유무
Curtis	(NULL)	못받는다
Randall	(NULL)	못받는다
Peter	(NULL)	못받는다
John	0.40	받는다
Karen	0.30	받는다

```
-- 급여정보 연말 보너스 지급
-- 20000 이상이면 5000증가
-- 15000 이상 20000 미만 10000증가
-- 10000 이상 15000 미만 20000증가
-- 나머지 30000증가

SELECT e.first_name, e.salary,
       case when e.salary >= 20000 then e.salary + 5000
            when e.salary >= 15000 then e.salary + 10000
            when e.salary >= 10000 then e.salary + 20000
            ELSE e.salary + 30000
       END 연말보너스
FROM employees e
ORDER BY e.salary DESC;
```

first_name	salary	연말보너스
Steven	24,000.00	29,000.00
Neena	17,000.00	27,000.00
Lex	17,000.00	27,000.00
John	14,000.00	34,000.00
Karen	13,500.00	33,500.00

```
-- 날짜 함수 형식변환 date_format(쉽다)
-- 자바 : 'yyyy-MM-dd HH:mm:ss'
SELECT NOW() , DATE_FORMAT(NOW(), '%Y/%m/%d'), DATE_FORMAT(NOW(), '%Y-%m-%d %H:%i:%s');
SELECT DATE_FORMAT('2023-01-01 00:00:00', '%Y/%C/%e');
SELECT DATE_FORMAT(NOW(), '%Y/%m/%d %W %H:%i:%s');

/*
%Y, %y => 4/2 자리 년도
%m, %M, %c => 2자리/ 영문월/ 1~2자리 월
%d, %e => 2/1~2자리 일
%W => 영문요일
%a => 영문 3글자 요일

%H, %h => 24/12시간
%i => 분
```

NOW()	DATE_FORMAT(NOW(), '%Y/%m/%d')	DATE_FORMAT(NOW(), '%Y-%m-%d %H:%i:%s')
2023-12-08 16:13:20	2023/12/08	2023-12-08 16:13:20

DATE\_FORMAT('2023-01-01 00:00:00', '%Y/%C/%e')

2023/1/1

DATE\_FORMAT(NOW(), '%Y/%m/%d %W %H:%i:%s')

2022/12/28 Wednesday 16:13:20

```
%S => 초
* /
```

```
-- 연도추출 함수
```

```
SELECT YEAR(NOW()), MONTH(NOW()), DAY(NOW()), HOUR(NOW()), MINUTE(NOW()), SECOND(NOW());
```

YEAR(NOW())	MONTH(NOW())	DAY(NOW())	HOUR(NOW())	MINUTE(NOW())	SECOND(NOW())
2022	12	28	16	28	

```
SELECT DATE_FORMAT(20221204, '%Y/%m/%d-%W'), WEEKDAY(20221204);
```

DATE_FORMAT(20221204, '%Y/%m/%d-%W')	WEEKDAY(20221204)
2022/12/04-Sunday	

```
-- 두 날짜 사이의 계산
```

```
SELECT CURDATE() 오늘날짜,
SUBDATE( CURDATE(), INTERVAL 1 DAY) 어제날짜,
ADDDATE(CURDATE(), INTERVAL 1 DAY) 내일날짜,
ADDDATE(CURDATE(), INTERVAL 1 MONTH ) 한달후날짜,
ADDDATE(CURDATE(), INTERVAL 1 YEAR ) 1년후날짜;
```

오늘날짜	어제날짜	내일날짜	한달후날짜	1년후날짜
2022-12-28	2022-12-27	2022-12-29	2023-01-28	2023-12-28

```
-- NOW() HIRE_date --> 입사한지 얼마나 경과주수 계산
```

```
SELECT DATEDIFF(NOW(), hire_date) 경과일수,
TRUNCATE( DATEDIFF(NOW(), hire_date) /7, 0) 경과주수,
truncate(DATEDIFF(NOW(), hire_date)/365, 0) 경과년수,
PERIOD_DIFF(date_format(NOW(), "%Y%m"), date_format(hire_date, "%Y%m"))
FROM employees;
```

employees (107r x 4c)			
경과일수	경과주수	경과년수	경과개월수
7,134	1,019	19	
6,307	901	17	
8,019	1,145	21	
6,203	886	16	

## ▼ 트랜잭션



### 트랜잭션 - 작업 논리 결함 대체 - ALL or Nothing

- 여러개 SQL 실행 → 계좌이체 1개 작업
  - atm 이체 중 a 계좌 출금 → b 계좌 입금 X 라면 ?
    - 원상태로 되돌려야 한다. → 트랜잭션(Commit or Rollback)
- TCL 을 통해 구현 \_ Transaction Control Lang.
  - commit : 완료
  - rollback : 없던 일로

```
-- 마리아 db 변수 확인
SHOW variables LIKE '%auto%';
-- autocommit 설정 변경
SET autocommit = FALSE;
```