

MySQL 내장 함수 정리

1. 수학 함수

1) ABS, MOD, POW, SQRT

- ① 절대값 : `SELECT ABS(-100);`
- ② 나머지 : `SELECT MOD(14, 3), 14 % 3, 14 MOD 3;`
- ③ 제곱 & 제곱근 : `SELECT POW(3, 2), SQRT(16);`

2) CEILING(올림), FLOOR(버림), ROUND(반올림)

- ① `SELECT CEILING(3.7), FLOOR(3.7), ROUND(3.7);`
- ② `SELECT CEILING(3.789, 2), FLOOR(3.789, 2), ROUND(3.789, 2);`

3) RAND, TRUNCATE

- ① 랜덤한 수(0~1) 발생 : `SELECT RAND();`
- ② `SELECT TRUNCATE(1234.6789, 2), TRUNCATE(1234.6789, -2);`

문제1> 1에서 10까지의 랜덤 한 수를 출력하세요.

문제2> 로또번호 6개를 다음의 형식으로 출력하세요.

```
+-----+-----+-----+-----+-----+-----+
| 1st | 2st | 3st | 4st | 5st | 6st |
+-----+-----+-----+-----+-----+-----+
| 15  | 34  | 31  | 11  | 3   | 30  |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

2. 문자열 함수

1) BIT_LENGTH, CHAR_LENGTH, LENGTH (문자열 길이)

- UTF8 에서 한글은 3바이트입니다.
- CHAR_LENGTH 는 문자의 갯수, LENGTH 는 바이트 수를 나타냅니다.

- ① SELECT BIT_LENGTH('abc'), CHAR_LENGTH('abc'), LENGTH('abc');
- ② SELECT BIT_LENGTH('가나다'), CHAR_LENGTH('가나다'), LENGTH('가나다');

2) CONCAT, CONCAT_WS (문자열 합치기)

- CONCAT_WS 는 구분자를 이용해 문자를 합칩니다.

- ① SELECT CONCAT('2023', '01', '01');
- ② SELECT CONCAT_WS('-', '2023', '01', '01');

3) INSTR (찾기)

- 찾는 문자열이 없으면 0 을 반환합니다.

- ① SELECT INSTR('abcd', 'b');

4) FORMAT

- 3자리마다 콤마를 추가해 줍니다. 소수점 2자리까지 표시합니다.
- 단 결과로 받는 데이터는 문자입니다.

- ① SELECT FORMAT(123456789.1234, 2);

문제3> 위의 값을 3자리 마다 콤마를 추가해 정수로 출력하세요.

5) LPAD, RPAD

- ① SELECT LPAD('1234', 6, '0'), RPAD('1234', 6, '0');

6) LTRIM, RTRIM, TRIM

- 좌/우 측 공백 제거하기
- 공백문자 이외 다른 문자도 TRIM 할 수 있습니다.

① SELECT LTRIM(' abc'), RTRIM('abc '), TRIM(' abc ');

② SELECT TRIM(BOTH 'a' FROM 'aababaa');

- 문자 'a'를 주어진 문자열의 양끝에서 제거합니다

7) LEFT, RIGHT, MID

- LEFT : 왼쪽에서 오른쪽으로 주어진 수 만큼 문자 추출하기
- RIGHT : 오른쪽에서 왼쪽으로 주어진 수 만큼 문자 추출하기
- MID : 문자열의 시작위치부터 주어진 수 만큼 문자 추출하기

① SELECT LEFT('가나다라마바', 3), RIGHT('가나다라마바', 3);

② SELECT MID('aBcDe', 2, 3);

8) LCASE, UCASE, LOWER, UPPER : 대소문자 변환하기

① SELECT LCASE('aBcDe'), UCASE('aBcDe'); -- 대문자, 소문자

② SELECT LOWER('aBcDe'), UPPER('aBcDe');

9) REPLACE : 주어진 문자열로 바꾸기

① SELECT REPLACE ('It is banana', 'banana', 'apple');

10) SUBSTRING, SUBSTRING_INDEX, LOCATE()

SUBSTRING_INDEX

- 찾는 문자열이 n 회 등장하면 그 이후 문자열을 버립니다.
- n 이 마이너스이면 오른쪽에서 시작합니다.

① SELECT SUBSTRING('abcdef', 3, 2); -- 문자열, 시작위치, 길이

② SELECT SUBSTRING_INDEX('aaa,bbb,ccc,ddd', ',', 2);

③ SELECT SUBSTRING_INDEX('aaa,bbb,ccc,ddd', ',', -2);

④ SELECT LOCATE('a', 'banana'); -- 'babana' 문자열에서 좌측에서부터의 'a'의 위치

⑤ SELECT LOCATE('a', 'banana', 3); -- 'babana' 문자열에서 좌측 3번째부터 그 후 'a'의 위치

문제4> 아래 email에서 id 만 추출해 주세요.

test999@gmail.com

문제5> 아래 주소 중에서 ① 도로명과 괄호안의 ② 동 명칭을 추출하세요.

서울특별시 용산구 서빙고로 137 (용산동6가)

▶ 사전작업

```
mysql> create table test (addr varchar(100));
```

```
Query OK, 0 rows affected (0.15 sec)
```

```
mysql> insert into test values ('서울특별시 용산구 서빙고로 137 (용산동6가)');
```

```
Query OK, 1 row affected (0.05 sec)
```

```
mysql> select * from test;
```

```
+-----+
| addr                                     |
+-----+
| 서울특별시 용산구 서빙고로 137 (용산동6가) |
+-----+
1 row in set (0.00 sec)
```

① 도로명 :

```
+-----+
| 도로명 |
+-----+
| 서빙고로 |
+-----+
1 row in set (0.00 sec)
```

② 동 주소 :

```
+-----+
| 동 주소 |
+-----+
| 용산동6가 |
+-----+
1 row in set (0.00 sec)
```

3. 날짜관련 함수

1) ADDDATE, SUBDATE, DATE_ADD, DATE_SUB : 날짜의 연산

- ① SELECT ADDDATE('2020-01-01', INTERVAL 31 DAY),
SUBDATE('2020-01-01', INTERVAL 7 DAY);
→ SELECT DATE('2020-01-01') + INTERVAL 31 DAY ; **같아요**
- ② SELECT DATE_ADD('2020-01-01', INTERVAL -31 DAY),
DATE_SUB('2020-01-01', INTERVAL 7 DAY);
- ③ SELECT ADDDATE('2020-01-01', INTERVAL -1 HOUR);
- ④ SELECT ADDDATE('2020-01-01', INTERVAL -1 MINUTE);
- ⑤ SELECT ADDDATE('2020-01-01', INTERVAL -1 SECOND);

2) CURDATE, CURTIME, NOW, SYSDATE : 현재 날짜와 시간의 반환

- CURDATE, CURTIME 는 날짜 또는 시간만 반환합니다.
- NOW, SYSDATE 는 날짜+시간을 반환합니다.
- 함수에게 주어지는 인자가 없습니다.

- ① SELECT CURDATE(), CURTIME(), NOW(), SYSDATE();

문제6> 위에서 생성한 test 테이블에 insert_date 필드를 추가(Alter 구문)하고 default값을 입력 시 날짜로 처리되도록 하세요. - (인터넷 검색해서 합니다.)

- ▶ 수정 후 아래의 주소를 입력하고 확인합니다.
- insert into test(addr) values('서울특별시 종로구 청와대로 1');

3) YEAR, MONTH, DAY, HOUR, MINUTE, SECOND, MICROSECOND

- 주어진 날짜 데이터(시간포함)에서 원하는 날짜 및 시간 정보를 얻어옵니다.

- ① SELECT YEAR(NOW()), MONTH(NOW()), DAY(NOW()),
HOUR(NOW()), MINUTE(NOW()), SECOND(NOW()), MICROSECOND(NOW());

4) MONTHNAME, DAYNAME, DAYOFWEEK, WEEKDAY, LAST_DAY

- ① 영어 월 이름 : SELECT MONTHNAME('2000-07-21');
- ② 영어 요일이름 : SELECT DAYNAME('2001-06-22');

③ SELECT WEEKDAY('2024-03-11');

- 한주의 몇번째 요일인지 해당하는 숫자를 반환합니다.
- 월요일(0) ,화요일(1), ..., 일요일(6)
- DAYOFWEEK와 WEEKDAY 함수의 결과는 같습니다.

④ SELECT DAYOFWEEK(NOW()), MONTHNAME(NOW()), DAYOFYEAR(NOW());

⑤ 해당월의 마지막 날짜 : SELECT LAST_DAY('2020-02-04');

5) DATE, TIME, DATEDIFF, TIMEDIFF

① SELECT DATE(NOW()), TIME(NOW());

② SELECT DATEDIFF('2020-1-5', '2020-1-1'), TIMEDIFF('14:30:00', '06:30:00');

6) DATE_FORMAT

① SELECT DATE_FORMAT(NOW(), '%Y-%m-%d') FROM DUAL;

② SELECT DATE_FORMAT(NOW(), '%H:%i:%S') FROM DUAL;

문제7> 다음을 해결해 주세요.

① 아이돌 장원영(2004-8-31), 안유진(2003-9-1)의 만 나이를 구해 주세요.

```
+-----+
| 장원영 만 나이 | 안유진 만 나이 |
+-----+
| 만 19세       | 만 20세       |
+-----+
1 row in set (0.00 sec)
```

② 아이브 이서가 책을 2024-02-28에 빌려 갔습니다. 오늘까지 며칠째 책을 갖고 있나요.

```
+-----+
| 이서가 빌려간 날 수 |
+-----+
| 11 |
+-----+
1 row in set (0.00 sec)
```

문제8> 날짜 검색 기간을 만들어 주세요

① 지난달의 첫 날짜와 마지막 날짜구하기

```
+-----+
| 지난달 첫 날 | 지난달 마지막 날 |
+-----+
| 2024-02-01  | 2024-02-29      |
+-----+
1 row in set (0.00 sec)
```

② 최근 1개월(예 : 오늘 2023-2-14 → 2023-1-15 ~ 2023-2-14)

```
+-----+
| 최근 1개월 전 | 오늘 |
+-----+
| 2024-02-11    | 2024-03-10 |
+-----+
1 row in set (0.00 sec)
```

③ 최근 3개월(예 : 오늘 2023-7-14 → 2023-4-15 ~ 2023-7-14)

```
+-----+
| 최근 3개월 전 | 오늘 |
+-----+
| 2023-12-11    | 2024-03-10 |
+-----+
1 row in set (0.00 sec)
```

문제9> 3월에 태어난 여성 회원 목록 출력하기

MEMBER_PROFILE 테이블에서 생일이 3월인 여성 회원의 ID, 이름, 성별, 생년월일을 조회하는 SQL문을 작성해주세요. 이때 전화번호가 NULL인 경우는 출력대상에서 제외시켜 주시고, 결과는 회원ID를 기준으로 오름차순 정렬해주세요.

예시

MEMBER_PROFILE 테이블이 다음과 같을 때

MEMBER_ID	MEMBER_NAME	TLNO	GENDER	DATE_OF_BIRTH
jiho92@naver.com	이지호	01076432111	W	1992-02-12
jiyeon22@hotmail.com	김지윤	01032324117	W	1992-02-22
jihoon93@hanmail.net	김지훈	01023258688	M	1993-02-23
seoyeons@naver.com	박서연	01076482209	W	1993-03-16
yoonsy94@gmail.com	윤서연	NULL	W	1994-03-19

SQL을 실행하면 다음과 같이 출력되어야 합니다.

MEMBER_ID	MEMBER_NAME	GENDER	DATE_OF_BIRTH
seoyeons@naver.com	박서연	W	1993-03-16

주의사항!

DATE_OF_BIRTH의 데이트 포맷이 예시와 동일해야 정답처리 됩니다. 현재 DATE_OF_BIRTH 필드에 입력되어 있는 데이터의 일반 형태는

1994-03-19 00:00:00 형태입니다.

4. 논리 함수

1) IF, IFNULL

- 조건식, 참일때 반환값, 거짓일때 반환값
- 첫번째가 NULL 이면 두번째 값 반환

- ① SELECT IF(a>1, 'A', 'B');
- ② SELECT IFNULL(a, 'ERR');

5. 통계 함수 및 CASE 구문

1) COUNT, AVG, SUM, MIN, MAX

- ① SELECT COUNT(column), AVG(column), SUM(column), MIN(column), MAX(column);

2) CASE 구문

```
CASE 값
WHEN 값1 THEN 내용
WHEN 값2 THEN 내용
WHEN 값3 THEN 내용
WHEN 값4 THEN 내용
END
```

문제10> 오늘은 무슨 요일인가요.. 세가지 방법으로 해결해 주세요

```
+-----+
| 오늘은 | 무슨요일? |
+-----+-----+
| 2024-03-10 | 일요일 |
+-----+-----+
1 row in set (0.00 sec)
```

- ① CASE ~ WHEN ~ THEN ~ END
- ② MID
- ③ SUBSTRING

6. 중복 제거 Distinct

① 언제 쓸까요?

예를 들어, 회사 직원 정보를 담은 employees 테이블에 아래와 같은 데이터가 있습니다.

	id	name	position	department
1	1	Alice	Developer	IT
2	2	Bob	Developer	IT
3	3	Carol	Analyst	IT
4	4	David	Developer	IT
5	5	Eve	Analyst	Finance

① 중복 제거

이때 회사의 직책(position)별로 인원을 파악하고 싶으면 distinct를 사용해 중복되지 않은 직책만 조회할 수 있습니다.

SELECT DISTINCT position FROM employees;

	position
1	Analyst
2	Developer

② 여러 컬럼에 distinct 사용

여러 컬럼에 대해 중복을 제거해서 조회하고 싶을 때도 distinct를 사용할 수 있습니다. 예를 들어, 아래와 같이 고유한 position과 department에 대해 조회할 수 있습니다.

SELECT DISTINCT position, department FROM employees;

	position	department
1	Analyst	Finance
2	Analyst	IT
3	Developer	IT

결과를 보면 알 수 있듯 distinct를 여러 컬럼에 사용하면 고유한 '조합'을 반환합니다. position 컬럼에 Analyst 값과 department 컬럼에 IT 값이 각 2번 나오는 것을 보고 언뜻 중복이 제거된 게 맞나 싶을 수 있지만, 자세히 보면 position과 department로 만들 수 있는 고유한 조합을 출력하는 것을 확인할 수 있습니다.

③ count와 함께 사용한 distinct

count와 distinct를 함께 사용하면 중복이 제거된 값의 개수를 조회할 수 있습니다.

employees 테이블에서 순수하게 직책(position)의 개수를 조회하려면 아래와 같이 쿼리를 실행하면 됩니다.

SELECT COUNT(DISTINCT position) FROM employees;

7. Limit와 offset

아래와 같은 테이블이 있습니다.

SELECT * FROM movie;

id	review	title	director	released_year
1	재밌다	설국열차	봉진호	2015
2	슬프다	1번방의 기적	김민수	2100
3	무섭다	괴물	봉준호	2002
4	감동적이다	인생은 아름다워	김민식	2009
5	또 보고싶다	어벤져스	토마스	2991

▶ LIMIT와 offset 사용하기

① 1개의 행만 출력

SELECT * FROM movie limit 1;

	id	review	title	director	released_year
▶	1	재밌다	설국열차	봉진호	2015

② offset : 몇 번째 행부터 출력할지 결정

select * from movie limit a,b;

- a: offset으로 a행부터 출력됩니다. offset의 기본숫자는 0부터 시작합니다.
- b: limit 출력할 행의 숫자를 결정합니다.

SELECT * FROM movie limit 3, 2;

	id	review	title	director	released_year
▶	4	감동적이다	인생은 아름다워	김민식	2009
	5	또 보고싶다	어벤져스	토마스	2991