



제4장

데이터 형식과 내장함수-1

5. 내장 함수(수학 함수)

- ABS(숫자)

- 숫자의 절댓값을 계산
- 다음 쿼리문을 실행하면 절댓값인 '100'이 출력

```
SELECT ABS(-100);
```

- CEILING(숫자), FLOOR(숫자), ROUND(숫자)

- 올림, 내림, 반올림을 계산
- 다음 쿼리문을 실행하면 '5, 4, 5'가 출력

```
SELECT CEILING(4.7), FLOOR(4.7), ROUND(4.7);
```

- CONV(숫자, 원래 _ 진수, 변환할 _ 진수)

- 숫자를 원래 진수에서 변환할 진수로 변환
- 다음 쿼리문을 실행하면 16진수 AA를 2진수로 변환한 '10101010'과 10진수 100을 8진수로 변환한 '144'가 출력

```
SELECT CONV('AA', 16, 2), CONV(100, 10, 8);
```

5. 내장 함수(수학 함수)

- DEGREES(숫자), RADIANS(숫자), PI()
 - DEGREES() 함수는 라디안 값을 각도 값으로 변환
 - RADIANS() 함수는 각도 값을 라디안 값으로 변환
 - PI() 함수는 파이 값인 3.141592를 반환
 - 다음 쿼리문을 실행하면 파이의 각도 값인 '180'과 180의 라디안 값인 3.141592...가 출력

```
SELECT DEGREES(PI()), RADIANS(180);
```

- MOD(숫자1, 숫자2) 또는 숫자1 % 숫자2 또는 숫자1 MOD 숫자2
 - 숫자1을 숫자2로 나눈 나머지 값을 반환
 - 다음 쿼리문을 실행하면 모두 157을 10으로 나눈 나머지 값 '7'이 출력

```
SELECT MOD(157, 10), 157 % 10, 157 MOD 10;
```

- POW(숫자1, 숫자2), SQRT(숫자)
 - POW() 함수는 숫자1을 숫자2만큼 거듭제곱한 값을 반환
 - SQRT() 함수는 숫자의 제곱근을 반환
 - 다음 쿼리문을 실행하면 2의 3제곱인 8과 루트 9의 값인 3이 출력

```
SELECT POW(2,3), SQRT(9);
```

5. 내장 함수(수학 함수)

■ RAND()

- 0 이상 1 미만의 실수를 구함
- 다음 쿼리문을 실행하면 0~1 미만의 실수와 주사위 숫자가 출력

```
SELECT RAND(), FLOOR(1 + (RAND() * (6-1)));
```

■ SIGN(숫자)

- 숫자가 양수, 0, 음수인지 구하여 1, 0, -1 중 하나를 반환
- 다음 쿼리문을 실행하면 '1, 0, -1' 이 출력

```
SELECT SIGN(100), SIGN(0), SIGN(-100.123);
```

■ TRUNCATE(숫자, 정수)

- 숫자를 소수점을 기준으로 정수 위치까지 구하고 나머지는 버림
- 다음 쿼리문을 실행하면 '12345.12'와 '12300'이 출력

```
SELECT TRUNCATE(12345.12345, 2), TRUNCATE(12345.12345, -2);
```

3. 내장 함수(날짜/시간 함수)

- ADDDATE(날짜, 차이), SUBDATE(날짜, 차이)
 - 날짜를 기준으로 차이를 더하거나 뺀 날짜를 반환
 - 다음 쿼리문을 실행하면 31일 후 또는 한 달후인 '2020-02-01'과 31일 전 또는 한 달 전인 '2019-12-01'이 출력

```
SELECT ADDDATE('2020-01-01', INTERVAL 31 DAY), ADDDATE('2020-01-01', INTERVAL 1 MONTH);  
SELECT SUBDATE('2020-01-01', INTERVAL 31 DAY), SUBDATE('2020-01-01', INTERVAL 1 MONTH);
```

- ADDTIME(날짜/시간, 시간), SUBTIME(날짜/시간, 시간)
 - 날짜/시간을 기준으로 시간을 더하거나 뺀 결과를 반환
 - 다음 쿼리문의 첫 번째는 1시간 1분 1초 후인 '2020-01-02 01:01:00'과 2시간 10분 10초 후인 '17:10:10'이 출력되고, 두 번째는 1시간 1분 1초 전인 '2020-01-01 22:58:58'과 2시간 10분 10초 전인 '12:49:50'이 출력

```
SELECT ADDTIME('2020-01-01 23:59:59', '1:1:1'), ADDTIME('15:00:00', '2:10:10');  
SELECT SUBTIME('2020-01-01 23:59:59', '1:1:1'), SUBTIME('15:00:00', '2:10:10');
```

3. 내장 함수(날짜/시간 함수)

- YEAR(날짜), MONTH(날짜), DAY(날짜), HOUR(시간), MINUTE(시간), SECOND(시간), MICROSECOND(시간)
 - 날짜 또는 시간에서 연, 월, 일, 시, 분, 초, 밀리초를 구함
 - 다음 쿼리문을 실행하면 현재 연, 월, 일, 시, 분, 초, 밀리초가 출력

```
SELECT YEAR(CURDATE()), MONTH(CURRENT_DATE()), DAYOFMONTH(CURRENT_DATE);  
SELECT HOUR(CURTIME()), MINUTE(CURRENT_TIME()), SECOND(CURRENT_TIME),  
MICROSECOND(CURRENT_TIME);
```

- DATE(), TIME()
 - DATETIME 형식에서 연-월-일과 시:분:초만 추출
 - 다음 쿼리문을 실행하면 현재 연-월-일과 시:분:초가 출력

```
SELECT DATE(NOW()), TIME(NOW());
```

- DATEDIFF(날짜1, 날짜2), TIMEDIFF(날짜1또는 시간1, 날짜1또는 시간2)
 - DATEDIFF() 함수는 날짜1-날짜2의 결과를 반환
 - 다음 쿼리문을 실행하면 2023년 1월 1일에서 오늘 날짜를 뺀 일자와 '11:12:49'가 출력

```
SELECT DATEDIFF('2023-01-01', NOW()), TIMEDIFF('23:23:59', '12:11:10');
```

3. 내장 함수(날짜/시간 함수)

- DAYOFWEEK(날짜), MONTHNAME(), DAYOFYEAR(날짜)

- DAYOFWEEK() 함수는 요일(1: 일~7: 토) 반환
- MONTHNAME() 함수는 월의 영문(January ~December) 반환
- DAYOFYEAR() 함수는 1년 중 몇 번째 날(1~366)인지를 반환
- 다음 쿼리문을 실행하면 현재 요일이 숫자로, 현재 월이 영문으로, 1년 중 며칠이 지났는지 숫자로 출력

```
SELECT DAYOFWEEK(CURDATE()), MONTHNAME(CURDATE()), DAYOFYEAR(CURDATE());
```

- LAST_DAY(날짜)

- 입력한 월의 마지막 날짜를 반환
- 다음 쿼리 문을 실행하면 '2020-02-29'가 출력

```
SELECT LAST_DAY('2020-02-01');
```

- MAKEDATE(연도, 정수)

- 연도의 첫날부터 정수만큼 지난 날짜를 반환
- 음 쿼리문을 실행하면 2020년 1월 1일부터 32 일이 지난 날짜인 '2020-02-01'이 출력

```
SELECT MAKEDATE(2020, 32);
```

3. 내장 함수(날짜/시간 함수)

- MAKETIME(시, 분, 초)

- 시, 분, 초를 이용하여 '시:분:초'의 TIME 형식을 만들
- 다음 쿼리문을 실행하면 '12:11:10'의 TIME 형식을 출력

```
SELECT MAKETIME(12, 11, 10);
```

- PERIOD_ADD(연월, 개월수), PERIOD_DIFF(연월1, 연월2)

- PERIOD_ADD() 함수는 연월부터 개월 수만큼 지난 연월을 반환
- PERIOD_DIFF()는 연월1-연월2의 개월 수를 반환
- 다음 쿼리 문을 실행하면 '2020년 12월'과 '13개월'이 출력

```
SELECT PERIOD_ADD(202001, 11), PERIOD_DIFF(202001, 201812);
```

- QUARTER(날짜)

- 날짜가 4분기 중에서 몇 분기인지를 반환
- 다음 쿼리문을 실행하면 7월 7일에 해당하는 '3분기'가 출력

```
SELECT QUARTER('2020-07-07');
```


3. 내장 함수(날짜/시간 함수)

- TIME_TO_SEC(시간)
 - 시간을 초 단위로 반환
 - 다음 쿼리문을 실행하면 '43870초'가 출력

```
SELECT TIME_TO_SEC('12:11:10');
```

- CURDATE(), CURTIME(), NOW(), SYSDATE()
 - CURDATE()는 현재 '연-월-일' 반환
 - CURTIME()은 현재 '시:분:초' 반환
 - NOW()와 SYSDATE()는 현재 '연-월-일 시:분:초' 반환

3. 내장 함수(시스템/정보 함수)

- USER(), DATABASE()

- 현재 사용자와 현재 선택된 데이터베이스를 반환
- 다음 쿼리문을 실행하면 현재 사용자와 현재 선택된 데이터베이스가 출력

```
SELECT CURRENT_USER(), DATABASE();
```

- FOUND_ROWS()

- 바로 앞의 SELECT 문에서 조회된 행의 개수를 반환
- 다음 쿼리문에서는 고객 테이블의 10개 행을 조회했으므로 '10'이 출력

```
SELECT * FROM userTBL;  
SELECT FOUND_ROWS();
```

- ROW_COUNT()

- 바로 앞의 INSERT, UPDATE, DELETE 문에서 삽입, 수정, 삭제된 행의 개수를 반환
- 다음 쿼리문은 UPDATE 문에서 구매 테이블의 12개 행을 변경했으므로 '12'가 출력

```
UPDATE buyTBL SET price=price * 2;  
SELECT ROW_COUNT();
```

3. 내장 함수(시스템/정보 함수)

- SLEEP(초)

- 쿼리의 실행을 잠깐 멈춤
- 다음 쿼리문을 실행하면 5초 동안 멈춘 후 결과가 출력

```
SELECT SLEEP(5);  
SELECT '5초 후에 이게 보여요';
```

- VERSION()

- 현재 MySQL의 버전을 출력

감사합니다.

