7장 2교시 데이터형 변환, 내장 함수

**명시적인 형 변환 Explict conversion**

CAST(A as B)

* CAST(AVG(amount) as SIGNED INTEGER)

CONVERT(A, B)

* CONVERT(AVG(amount), SIGNED INTEGER)

둘 다 다른 데이터 형식으로 변환하는 데 쓰인다.

다만 CONVERT 함수에는 CAST 함수에 없는 스타일을 정의할 수 있는 기능이 있다.

SELECT CAST(‘2020/12/12’ as DATE);

* 2020$12$12, 2020/12/12, 2020%12%12, 2020@12@12
* 연, 월, 일 사이 기호는 위 4가지 모두 가능

CONCAT: 문자열을 잇는 함수

* SELECT CONCAT('ABC', ‘DEF')
  + ABCDEF

**암시적인 형 변환 Implicit conversion**

* SELECT '100' + '200' ;
  + 문자와 문자를 더함 (정수로 변환되서 연산됨)
  + 300 (100 + 200)
* SELECT CONCAT('100', '200');
  + 문자와 문자를 연결 (문자로 처리)
  + 100200
* SELECT CONCAT(100, '200');
  + 정수와 문자를 연결 (정수가 문자로 변환되서 처리)
  + 100200
* SELECT 1 > '2mega';
  + 정수인 2로 변환되어서 비교
  + 0 (False)
* SELECT 3 > '2MEGA';
  + 정수인 2로 변환되어서 비교
  + 1 (True)
* SELECT 0 = 'mega';
  + 문자는 0으로 변환됨
  + 1 (True)

**MySQL 내장 함수**

* SELECT IF (100>200, 'ㅁ', 'aadsf');
  + 2번째 인자가 참일 때, 3번째 인자가 거짓일 때
  + aadsf
* SELECT IFNULL(NULL, '널이군요'), IFNULL(100, '널이군요');
  + NULL이면 뒤에가 출력, NULL이 아니면 앞이 출력
  + 널이군요, 100
* SELECT NULLIF(100,100), IFNULL(200,100);
  + 값이 같으면 NULL, 다르면 앞의 값
  + NULL, 200
* SELECT CASE 10

WHEN 1 THEN '일'

WHEN 5 THEN '오'

WHEN 10 THEN '십'

ELSE '모름'

END AS 'CASE연습';

* + CASE
  + 십
* SELECT ASCII('A'), CHAR(65);
  + 문자의 아스키값이나 아스키값의 해당하는 문자 리턴
  + 65, A
* SELECT BIT\_LENGTH('abc'), CHAR\_LENGTH('abc'), LENGTH('abc');
* SELECT BIT\_LENGTH('가나다'), CHAR\_LENGTH('가나다'), LENGTH('가나다');
  + BIT\_LENGTH: Bit 크기 반환
  + CHAR\_LENGTH: 문자의 개수 반환
  + LENGTH: 할당된 Byte 수 반환 (글자당 영문 1바이트 한글 3바이트)
  + 24 Bit (9\*8), 3개, 3 Byte
  + 72 Bit (9\*8), 3개, 9 Byte
* SELECT CONCAT\_WS('/', '2025', '01', '01');
  + 구분자를 넣어서 문자열을 이음
  + 2025/01/01

SELECT ELT(2, '하나', '둘', '셋'), FIELD('둘', '하나', '둘', '셋'), FIND\_IN\_SET('둘', '하나,둘,셋'), INSTR('하나둘셋', '둘'), LOCATE('둘', '하나둘셋');

* + ELT: n번째 위치 반환
    - 둘
  + FIELD: 찾을 문자열 위치 반환하거나 없으면 0
    - 2
  + FIND\_IN\_SET: 찾을 문자열을 문자열 리스트에서 찾아서 위치 반환
    - 2
  + INSTR: 기준 문자열에서 부분 문자열 위치 반환
    - 셋
  + LOCATE: INSTR와 같지만 파라미터 순서가 반대
    - 3
* SELECT FORMAT(123456.123456, 4);
  + Formatting: 소수점 아래 자릿수 반올림하여 표기 및 1000 단위마다 콤마
  + 123,456.1235
* SELECT BIN(31), HEX(31), OCT(31);
  + 이진수, 16진수 8진수
  + 11111, 1F, 37
* SELECT INSERT('abcdefghi', 3, 4, '@@@@'), INSERT('abcdefghi', 3, 2, '@@@@');
  + 기준 문자열의 위치 A부터 B 길이만큼 지우고 C 문자열로 대체
  + ab@@@@ghi, ab@@@@efghi
* SELECT LEFT('abcdefghi', 3), RIGHT('abcdefghi', 3);
  + 왼쪽 또는 오른쪽 문자열의 길이만큼 반환
  + abc, ghi
* SELECT LOWER('abcdEFGH'), UPPER('abcdEFGH');
  + 대문자, 소문자 변환
  + abcdefgh, ABCDEFGH
* SELE CT LPAD('이것이', 8, '##'), RPAD('이것이', 5, '##');
  + Padding: 왼쪽 패딩, 오른쪽 패딩
  + #####이것이, 이것이##
* SELECT LTRIM(' 이것이'), RTRIM('이것이 ');
  + 왼쪽, 오른쪽 공백 제거
  + 이것이
* SELECT TRIM(' 이것이 '), TRIM(BOTH 'ㅋ' FROM 'ㅋㅋㅋ재밌어요.ㅋㅋㅋ');
  + 앞뒤 공백 전부 제거하거나 혹은 정한 문자 제거
  + 이것이, 재밌어요
* SELECT REPEAT('이것이', 3);
  + 문자열 반복
  + 이것이이것이이것이
* SELECT REPLACE ('이것이 MySQL이다', '이것이' , 'This is');
  + 문자열 대체
  + This is MySQL이다
* SELECT REVERSE ('MySQL');
  + 문자열 거꾸로
  + LQSyM
* SELECT CONCAT('이것이', SPACE(10), 'MySQL이다');
  + 길이만큼 공백 반환
  + 이것이 MySQL이다
* SELECT SUBSTRING('대한민국만세', 4, 3);
  + 시작 위치부터 길이만큼 문자 반환
  + 국만세
* SELECT SUBSTRING\_INDEX('cafe.naver.com', '.', 2), SUBSTRING\_INDEX('cafe.naver.com', '.', -2);
  + 구분자가 왼쪽부터 찾아서 n번만큼 나오면 그 이후는 제거 (음수일 시 오른쪽부터 탐색)
  + cafe.naver, naver.com

**수학함수**

* 자주 안 쓰이므로 참조만! (Math 프로그램이 함수와 거의 동일)
* ABS - 절대값
* ACOS, ASIN, ATAN - 삼각함수
* CEILING, FLOOR, ROUND - 올림, 내림, 반올림
* CONV - 진수 변환
* DEGREES, RADIAN, PI
* EXP, LN, LOG
* MOD(m, n), m % n, m MOD n
* POW, SQRT
* RAND
* SIGN
* TRUNCATE

**날짜 및 시간 함수**

* SELECT ADDDATE('2025-01-01', INTERVAL 31 DAY), ADDDATE('2025-01-01', INTERVAL 1 MONTH);
* SELECT SUBDATE('2025-01-01', INTERVAL 31 DAY), SUBDATE('2025-01-01', INTERVAL 1 MONTH);
  + 날짜를 기준으로 차이만큼 더하거나 뺀다
* SELECT ADDTIME('2025-01-01 23:59:59', '1:1:1'), ADDTIME('15:00:00', '2:10:10');
* SELECT SUBTIME('2025-01-01 23:59:59', '1:1:1'), SUBTIME('15:00:00', '2:10:10');
  + 날짜/시간을 기준으로 시간만큼 더하거나 뺀다
* **SELECT YEAR(CURDATE()), MONTH(CURDATE()), DAYOFMONTH(CURDATE);**
  + 현재 날짜 및 시간 반환 (자주 쓰임)
* **SELECT HOUR(CURTIME()), MINUTE(CURRENT\_TIME()), SECOND(CURRENT\_TIME), MICROSECOND(CURRENT\_TIME);**
  + 시간에서 연, 월, 일, 시, 분, 초 등만 추출 (자주 쓰임)
* SELECT DATE(NOW()), TIME(NOW());
  + 날짜만 혹은 시간만 추출
* SELECT DATEDIFF('2025-01-01', NOW()), TIMEDIFF('23:23:59', '12:11:10');
  + 날짜의 일 수 차이 혹은 시간 차이
* SELECT DAYOFWEEK(CURDATE()), MONTHNAME(CURDATE()), DAYOFYEAR(CURDATE());
  + 현재 요일(1 일요일, 7 토요일), 현재 월, 일년 중 몇 일이 지났는지 반환
* SELECT LAST\_DAY('2025-02-01');
  + 주어진 날짜의 마지막 날짜 확인 (윤달 혹은 윤년)
  + 2025-02-29
* SELECT MAKEDATE(2025, 32);
  + 연도에서 주어진 숫자만큼 일수가 지난 날짜 반환
* SELECT MAKETIME(12, 11, 10);
  + 주어진 숫자로 시:분:초 형식 시간 반환
* SELECT PERIOD\_ADD(202501, 11), PERIOD\_DIFF(202501, 202312);
  + 연월에서 개월이 지난 연월 반환, 연월에서 개월 차이 반환
* SELECT QUARTER('2025-07-07');
  + 4분기 중 몇 분기인지
* SELECT TIME\_TO\_SEC('12:11:10');
  + 시간을 초 단위로 구한다

**시스템 정보 함수**

* USER(), DATABASE()
  + 현재 사용자 및 현재 데이터베이스
* **FOUND\_ROWS()**
  + SELECT문에서 조회된 행의 개수를 구한다
* **ROW\_COUNT()**
  + 앞의 INSERT, UPDATE, DELETE에서 입력, 수정, 삭제된 행의 개수를 구한다
  + CREATE, DROP은 0, SELECT는 -1
* VERSION()
  + MySQL 버전
* SLEEP(초)
  + 몇 초동안 쿼리의 실행을 멈춘다

7장 3교시 영화사이트대용량 데이터 구축 실습

