

MySql

5 . 1 MySQL의 데이터 형식

❖ MySQL에서 지원하는 데이터 형식의 종류

■ Data Type으로 표현

- 데이터 형식, 데이터형, 자료형, 데이터 타입

■ 데이터 형식에 대한 이해

- SELECT문 더욱 잘 활용
- 테이블의 생성 효율적으로 하기 위해 필요

■ MySQL에서 데이터 형식의 종류는 30개 정도

- 중요하고 자주 쓰는 형식에 대해 중점 학습



5 . 1 MySQL의 데이터 형식

❖ MySQL에서 지원하는 데이터 형식의 종류

■ 숫자 데이터 형식

데이터 형식	바이트 수	숫자 범위	설명
BIT(N)	N/8		1~64bit를 표현. b'0000' 형식으로 표현
TINYINT	1	-128~127	정수
★SMALLINT	2	-32,768~32,767	정수
MEDIUMINT	3	-8,388,608~8,388,607	정수
★INT INTEGER	4	약 -21억~+21억	정수
★BIGINT	8	약 -900경~+900경	정수
★FLOAT	4	-3.40E+38~-1.17E-38	소수점 아래 7자리까지 표현
★DOUBLE REAL	8	-1.22E-308~1.79E+308	소수점 아래 15자리까지 표현
★DECIMAL(m, [d]) NUMERIC(m, [d])	5~17	$-10^{38}+1 \sim +10^{38}-1$	전체 자릿수(m)와 소수점 이하 자릿수(d)를 가진 숫자형 예) decimal(5, 2)은 전체 자릿수를 5자리로 하되, 그 중 소수점 이하를 2자리로 하겠다는 의미



5 . 1 MySQL의 데이터 형식

❖ MySQL에서 지원하는 데이터 형식의 종류

■ 문자 데이터 형식

데이터 형식		바이트 수	설명
★CHAR(n)		1~255	고정길이 문자형. n을 1부터 255까지 지정. character의 약자 그냥 CHAR만 쓰면 CHAR(1)과 동일
★VARCHAR(n)		1~65535	가변길이 문자형. n을 사용하면 1부터 65535 까지 지정. Variable character의 약자
BINARY(n)		1~255	고정길이의 이진 데이터 값
VARBINARY(n)		1~255	가변길이의 이진 데이터 값
TEXT 형식	TINYTEXT	1~255	255 크기의 TEXT 데이터 값
	TEXT	1~65535	N 크기의 TEXT 데이터 값
	MEDIUMTEXT	1~16777215	16777215 크기의 TEXT 데이터 값
	★LONGTEXT	1~4294967295	최대 4GB 크기의 TEXT 데이터 값
BLOB 형식	TINYBLOB	1~255	255 크기의 BLOB 데이터 값
	BLOB	1~65535	N 크기의 BLOB 데이터 값
	MEDIUMBLOB	1~16777215	16777215 크기의 BLOB 데이터 값
	★LONGBLOB	1~4294967295	최대 4GB 크기의 BLOB 데이터 값
ENUM(값들...)		1 또는 2	최대 65535개의 열거형 데이터 값
SET(값들...)		1, 2, 3, 4, 8	최대 64개의 서로 다른 데이터 값



5 . 1 MySQL의 데이터 형식

❖ MySQL에서 지원하는 데이터 형식의 종류

■ 날짜와 시간 데이터 형식

데이터 형식	바이트 수	설명
★DATE	3	날짜는 1001-01-01~9999-12-31까지 저장되며 날짜 형식만 사용 'YYYY-MM-DD' 형식으로 사용됨
TIME	3	-838:59:59.000000~838:59:59.000000까지 저장되며 'HH:MM:SS' 형식으 로 사용
★DATETIME	8	날짜는 1001-01-01 00:00:00~9999-12-31 23:59:59까지 저장되며 형식은 'YYYY-MM-DD HH:MM:SS' 형식으로 사용
TIMESTAMP	4	날짜는 1001-01-01 00:00:00~9999-12-31 23:59:59까지 저장되며 형식은 'YYYY-MM-DD HH:MM:SS' 형식으로 사용. time_zone 시스템 변수와 관련이 있으며 UTC 시간대로 변환하여 저장
YEAR	1	1901~2155까지 저장. 'YYYY' 형식으로 사용

	DATE			TIME			DATETIME
▶	2020-10-19		▶	12:35:29		▶	2020-10-19 12:35:29



5 . 1 MySQL의 데이터 형식

❖ 변수의 사용

- Workbench를 재시작할 때까지는 계속 유지
- Workbench를 닫았다가 재시작하면 소멸
- 변수의 선언과 값의 대입 형식

```
SET @변수이름 = 변수의 값 ;      -- 변수의 선언 및 값 대입  
SELECT @변수이름 ;              -- 변수의 값 출력
```



5 . 1 MySQL의 데이터 형식

❖ MySQL 내장 함수

■ 제어 흐름 함수

- 제어 흐름 함수는 프로그램의 흐름 제어하는 역할
- IF (수식, 참, 거짓)
 - 수식이 참 또는 거짓인지 결과에 따라서 2중 분기
- IFNULL(수식1, 수식2)
 - 수식1이 NULL이 아니면 수식1이 반환
 - 수식1이 NULL이면 수식2가 반환



5 . 1 MySQL의 데이터 형식

❖ MySQL 내장 함수

■ 문자열 함수

- UCASE(문자열), LCASE(문자열)
 - 소문자를 대문자로, 대문자를 소문자로 변경
- UPPER(문자열), LOWER(문자열)
 - 소문자를 대문자로, 대문자를 소문자로 변경
- LPAD(문자열, 길이, 채울 문자열), RPAD(문자열, 길이, 채울 문자열)
 - 문자열을 길이만큼 늘린 후에 빈 곳을 채울 문자열로 채움
- LTRIM(문자열), RTRIM(문자열)
 - 문자열의 왼쪽/오른쪽 공백을 제거
 - 중간 공백은 제거되지 않음



5 . 1 MySQL의 데이터 형식

❖ MySQL 내장 함수

■ 문자열 함수

- TRIM(문자열), TRIM(방향 자를_문자열 FROM 문자열)
 - TRIM(문자열)은 문자열의 앞뒤 공백을 모두 없앴
 - TRIM(방향 자를_문자열 FROM 문자열) 에서 방향은 LEADING(앞), BOTH(양쪽), TRAILING(뒤) 으로 표시
- REPEAT(문자열, 횟수)
 - 문자열을 횟수만큼 반복
- REPLACE(문자열, 원래 문자열, 바꿀 문자열)
 - 문자열에서 원래 문자열을 찾아서 바꿀 문자열로 바꿈
- REVERSE(문자열)
 - 문자열의 순서를 거꾸로 바꿈



5 . 1 MySQL의 데이터 형식

❖ MySQL 내장 함수

■ 문자열 함수

- SPACE(길이)
 - 길이만큼의 공백을 반환
- SUBSTRING(문자열, 시작위치, 길이) 또는 SUBSTRING(문자열 FROM 시작위치 FOR 길이)
 - 시작위치부터 길이만큼 문자를 반환
 - 길이가 생략되면 문자열의 끝까지 반환
- SUBSTRING_INDEX(문자열, 구분자, 횟수)
 - 문자열에서 구분자가 왼쪽부터 횟수 번째까지 나오면 그 이후의 오른쪽은 버림
 - 횟수가 음수면 오른쪽부터 세고 왼쪽을 버림



5 . 1 MySQL의 데이터 형식

❖ MySQL 내장 함수

■ 수학 함수

- ABS(숫자)
 - 숫자의 절댓값 계산
- ACOS(숫자), ASIN(숫자), ATAN(숫자), ATAN2(숫자1, 숫자2), SIN(숫자), COS(숫자), TAN(숫자)
 - 삼각 함수와 관련된 함수 제공
- CEILING(숫자), FLOOR(숫자), ROUND(숫자)
 - 올림, 내림, 반올림 계산
- CONV(숫자, 원래 진수, 변환할 진수)
 - 숫자를 원래 진수에서 변환할 진수로 계산



5 . 1 MySQL의 데이터 형식

❖ MySQL 내장 함수

■ 수학 함수

- RAND()
 - RAND()는 0 이상 1 미만의 실수 구함
 - ‘ $m \leq$ 임의의 정수 $< n$ ’ 를 구하고 싶다면 $\text{FLOOR}(m + (\text{RAND}() * (n - m)))$ 사용
- SIGN(숫자)
 - 숫자가 양수, 0, 음수인지 판별
 - 결과는 1, 0, -1 셋 중에 하나 반환
- TRUNCATE(숫자, 정수)
 - 숫자를 소수점을 기준으로 정수 위치까지 구하고 나머지는 버림



5 . 1 MySQL의 데이터 형식

❖ MySQL 내장 함수

■ 날짜 및 시간 함수

- ADDDATE(날짜, 차이), SUBDATE(날짜, 차이)
 - 날짜를 기준으로 차이를 더하거나 뺀 날짜 구함
- ADDTIME(날짜/시간, 시간), SUBTIME(날짜/시간, 시간)
 - 날짜/시간을 기준으로 시간을 더하거나 뺀 결과를 구함
- CURDATE(), CURTIME(), NOW(), SYSDATE()
 - **CURDATE()**는 현재 연-월-일
 - **CURTIME()**은 현재 시:분:초
 - **NOW()**와 **SYSDATE()**는 현재 ‘연-월-일 시:분:초



5 . 2 조인

❖ 조인 (Join)

■ 조인의 개념?

- 두 개 이상의 테이블을 서로 묶어서 하나의 결과 집합으로 만들어 내는 작업

■ 데이터베이스의 테이블

- 여러 개의 테이블로 분리하여 저장
 - 중복과 공간 낭비를 피하고 데이터의 무결성 위함
- 분리된 테이블들은 서로 관계(Relation) 를 가짐
 - 1대 다 관계에서 일어나는 데이터 처리 필요성



5 . 2 조인

❖ INNER JOIN(내부 조인)

■ 조인 중에서 가장 많이 사용되는 조인

- 대개의 업무에서 조인은 INNER JOIN 사용
- 일반적으로 JOIN이라고 얘기하는 것이 이 INNER JOIN 지칭
- 사용 형식

```
SELECT <열 목록>  
FROM <첫 번째 테이블>  
      INNER JOIN <두 번째 테이블>  
      ON <조인될 조건>  
[WHERE 검색조건]
```

- 실습 예제 P. 264~272



5 . 2 조인

❖ OUTER JOIN(외부 조인)

■ 조인의 조건에 만족되지 않는 행까지도 포함시키는 것

- ‘왼쪽 테이블의 것은 모두 출력되어야 한다’ 고 해석하면 이해 쉬움

```
SELECT <열 목록>  
FROM <첫 번째 테이블(LEFT 테이블)>  
    <LEFT | RIGHT | FULL> OUTER JOIN <두 번째 테이블(RIGHT 테이블)>  
    ON <조인될 조건>  
[WHERE 검색조건] ;
```

- 실습 예제 P. 273~276

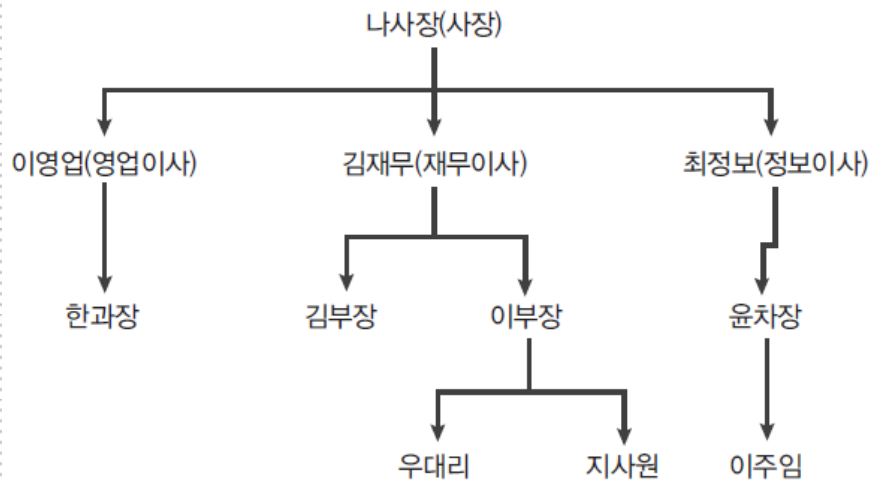


5.2 조인

❖ SELF JOIN(자체 조인)

■ 자기 자신과 자기 자신이 조인한다는 의미

- 조직도와 관련된 테이블



직원 이름(EMP) - 기본 키	상관 이름(MANAGER)	구내 번호
나사장	없음(NULL)	0000
김재무	나사장	2222
김부장	김재무	2222-1
이부장	김재무	2222-2
우대리	이부장	2222-2-1
지사원	이부장	2222-2-2
이영업	나사장	1111
한과장	이영업	1111-1
최정보	나사장	3333
윤차장	최정보	3333-1
이주임	윤차장	3333-1-1

5.2 조인

❖ UNION / UNION ALL / NOT IN / IN

■ 두 쿼리의 결과를 행으로 합치는 것

SELECT stdName, addr FROM stdTbl

stdName	addr
김범수	경남
성시경	서울
조용필	경기
은지원	경북
바비킴	서울

UNION ALL

SELECT clubName, roomNo FROM clubTbl

clubName	roomNo
수영	101호
바둑	102호
축구	103호
봉사	104호

stdName	addr
김범수	경남
성시경	서울
조용필	경기
은지원	경북
바비킴	서울
수영	101호
바둑	102호
축구	103호
봉사	104호

```
SELECT 문장1  
    UNION [ALL]  
SELECT 문장2
```

