

정수 타입

데이터타입	범위		바이트
	signed	unsigned	
TINYINT	-128 ~ 127	0 ~ 255	1 바이트
SMALLINT	-32768 ~ 32767	0 ~ 65535	2 바이트
MEDIUMINT	-8388608 ~ 8388607	0 ~ 16777215	3 바이트
INT	-2147483648 ~ 2147483647	0 ~ 4294967295	4 바이트
BIGINT	-9223372036854775808 ~ 9223372036854775807	0 ~ 18446744073709551615	8 바이트
DECIMAL	<p><u>전체자리수(precision)와 소수점자리수(scale)를 가짐</u></p> <ul style="list-style-type: none"> DECIMAL(5)의 경우: -99999 ~ 99999 DECIMAL(5, 1)의 경우: -9999.9 ~ 9999.9 DECIMAL(5, 2)의 경우: -999.99 ~ 999.99 		가변
FLOAT	$-3.402823466 \times 10^{38} \sim 3.402823466 \times 10^{38}$		4 바이트
DOUBLE	$-1.7976931348623157 \times 10^{308} \sim 1.7976931348623157 \times 10^{308}$		8 바이트

mysql 에서 bigint(20), int(11) 의 의미

Database - 2013. 2. 27. 10:48

mysql 에서 bigint 는 8바이트(64비트) 정수값이고, int 는 4바이트(32비트) 정수값이다.

그런데 뒤에 붙는 숫자값은 무슨 의미일까?



메뉴얼에서 보면 이 값은 zerofill 자리수를 의미하지, 저장될 숫자의 자리수를 의미하지 않는다.



일반적으로 DBMS 에서 이들 숫자값은 저장될 숫자의 크기를 의미하는데

mysql 에서는 단순히 결과값을 출력할 때, 숫자를 출력할 때 앞에 0을 채울 때 몇개를 채울지 hint 로 사용된다.

```
CREATE TABLE foo ( bar INT(20) ZEROFILL );
INSERT INTO foo (bar) VALUES (1234);
SELECT bar from foo;
```

bar
00000000000000001234

20자리

실제로 bigint(5), int(2) 타입에 큰 값을 저장해도 잘 저장된다.

```
CREATE TABLE foo(bar1 BIGINT(5), bar2 INT(2));
INSERT INTO foo(bar1, bar2) VALUES (55555555, 22222222);
SELECT * FROM foo;
```

bar1	bar2
55555555	22222222