数值型：

整数：**tinyint**(1byte --> -128 —— 127 )

**tinyint unsigned**( 0 -- 255 )

**int**(4byte --> -2^31 -- 2^31-1 ),

**bigint**(8byte),

浮点数：

decimal（ 64位 ）比float（ 5 -- 7位） 的精度更大，float存在精度缺失。

**decimal( m,n )**

**float( m,n )**:m表示总长度，n表示小数位数

eg:float(4,2) -99.99 --- 99.99

float(4,2) unsigned 0 -- 99.99

文本：

一个中文字和一个英文字母都表示一个字符。

**char(M)**:定长字符串，M 最大值为 255，单位为字符

**varchar(M)**：非定长字符串 M最大值为65532，mysql规定一行长度不超过65535个字节，其中 1 -- 3 个字节用来描述字符串长度

定长字符串检索效率高，空间利用率低；（身份证号，手机号，银行卡号等字段）

变长字符串检索效率低，空间利用率高（姓名一般定义为20，地址等字段）

**text:**大文本

时间类型：

**DATE:**日期类型 yyyy-mm-dd : 年 月 日 3字节

**DATETIME :** 日期类型 YYYY-MM-DD HH:mm:ss 年月日时分秒 8字节

**TIMESTAMP:**时间戳 日期类型 YYYY-MM-DD HH:mm:ss 4字节

每当有数据的插入或更新时，自动填充当前时间，无需用户插入

从1970-1-1 00：00：01开始到现在经过的秒数

集合:

**枚举:ENUM 表示单选**

enum(’选项1’，’选项2’，’选项3’....),最终只存储其中具体某一项

**集合：SET**

set(’选项1’，’选项2’，’选项3’....) 最终可存储多个选项

**检索set集合指定的选项** find\_in\_set(’要查询的选项’，set集合)

eg: select \* from [table\_name] where find\_in\_set(’睡觉’，[set\_name]);

5.关于varchar(len),len到底是多大，这个len值，和表的编码密切相关：

varchar长度可以指定为0到65535之间的值，但是有1 - 3 个字节用于记录数据大小，所以说有效字节数是 65532。 当我们的表的编码是utf8时，varchar(n)的参数n最大值是65532/3=21844[因为utf中，一个字符占用3个字 节]，如果编码是gbk，varchar(n)的参数n最大是65532/2=32766（因为gbk中，一个字符占用2字节）。

char 适合长度固定的数据

如果数据长度有变化,就使用变长(varchar),

定长的磁盘空间比较浪费，但是效率高。

变长的磁盘空间比较节省，但是效率低。