

MySQL中char、varchar和nvarchar的区别

一、char和varchar的区别

char是固定长度的，而varchar会根据具体的长度来使用存储空间，另外varchar需要用额外的1-2个字节存储字符串长度。

- 1). 当字符串长度小于255时，用额外的1个字节来记录长度
- 2). 当字符串长度大于255时，用额外的2个字节来记录长度

比如char(255)和varchar(255)，在存储字符串"hello world"时，char会用一块255个字节的空放那个11个字符；而varchar就不会用255个，它先计算字符串长度为11，然后再加上一个记录字符串长度的字节，一共用12个字节存储，这样varchar在存储不确定长度的字符串时会大大减少存储空间。

二、varchar和nvarchar的区别

1. varchar(n)：长度为n个字节的可变长度且非Unicode的字符数据。n必须是一个介于1和8,000之间的数值。存储大小为输入数据的字节的实际长度

2. nvarchar(n)：包含n个字符的可变长度Unicode字符数据。n的值必须介于1与4,000之间。字节的存储大小是所输入字符个数的两倍。所输入的数据字符长度可以为零。

两字段分别有字段值：我和coffee那么varchar字段占2×2+6=10个字节的存储空间，而nvarchar字段占8×2=16个字节的存储空间。

如字段值只是英文可选择varchar，而字段值存在较多的双字节（中文、韩文等）字符时用nvarchar

类型	长度	使用说明	长度说明
char (n)	定长	索引效率高 程序里面使用trim去除多余的空白	n 必须是一个介于 1 和 8,000 之间的数值,存储大小为 n 个字节
varchar (n)	变长	效率没char高 灵活	n 必须是一个介于 1 和 8,000 之间的数值。存储大小为输入数据的字节的实际长度，而不是 n 个字节
text (n)	变长	非Unicode数据	不用指定长度
nchar (n)	定长	处理unicode数据类型(所有的字符使用两个字节表示)	n 的值必须介于 1 与 4,000 之间。存储大小为 n 字节的两倍
nvarchar (n)	变长	处理unicode数据类型(所有的字符使用两个字节表示)	n 的值必须介于 1 与 4,000 之间。字节的存储大小是所输入字符个数的两倍。所输入的数据字符长度可以为零
ntext (n)	变长	处理unicode数据类型(所有的字符使用两个字节表示)	不用指定长度

三、char和varchar的使用场景

- 1. 从字符长度的角度考虑：

- (1). 长度较短的字段，使用char。如门牌号：101, 201, ...
- (2). 固定长度的字段，使用char。如性别、身份证号、手机号等。（因为数据都是固定长度，varchar根据长度动态存储的特性就没作用了，而且还要占一个字节来存储长度）
- (3). 字段的长度是否相近，如果某个字段其长度虽然比较长，但是其长度总是近似的，如一般在90个到100个字符之间，甚至是相同的长度。此时比较适合采用CHAR字符类型。

## 2. 从碎片角度考虑：

使用CHAR时，由于存储空间都是一次性分配的。从这个角度来讲，不存在碎片的困扰。而使用varchar时，因为存储的长度是可变的，当数据长度在更改前后不一致时，就不可避免地会出现碎片的问题。故使用varchar时，数据库管理员要时不时的对碎片进行整理。如执行数据库导出导入作业来消除碎片。

## 3. 即使使用Varchar数据类型，也不能够太过于慷慨。

虽然varchar可以自动根据长度调整存储空间，但是varchar(100)和varchar(255)还是有区别的：

假设它们都存储了90个字符的数据，那么它们在磁盘上的存储空间是相同的（硬盘上的存储空间是根据实际字符长度来分配存储空间的）。但对于内存来说，则不是这样的，内存是使用varchar中定义的长度(这里为100或255)的内存块来保存值。

所以如果某些字段会涉及到文件排序或者基于磁盘的临时表时，分配VARCHAR的长度时仍然不能过于慷慨，需要评估实际需要的长度，然后设置一个合适的长度，不能随意设置长度。

## 四、varchar和nvarchar的使用场景

1. nvarchar适用中文和其他字符，其中N表示Unicode编码，可以解决多语言之间的转换问题