# **[MySQL字符集及校对规则的理解](http://www.cnblogs.com/geaozhang/p/6724393.html)**

阅读目录：MySQL的字符集和校对规则

* [MySQL的字符集](https://www.cnblogs.com/geaozhang/p/6724393.html?utm_source=itdadao&utm_medium=referral" \l "zifuji)
  + [MySQL与字符集](https://www.cnblogs.com/geaozhang/p/6724393.html?utm_source=itdadao&utm_medium=referral" \l "mysqlyuzifuji)
  + [正确使用字符集](https://www.cnblogs.com/geaozhang/p/6724393.html?utm_source=itdadao&utm_medium=referral" \l "zhengqueshiyongzifuji)
  + [MySQL客户端与字符集](https://www.cnblogs.com/geaozhang/p/6724393.html?utm_source=itdadao&utm_medium=referral" \l "kehuduanyuzifuji)
  + [字符集编码转换原理](https://www.cnblogs.com/geaozhang/p/6724393.html?utm_source=itdadao&utm_medium=referral" \l "zifubianmazhuanhuanyuanli)
  + [字符集常见处理操作](https://www.cnblogs.com/geaozhang/p/6724393.html?utm_source=itdadao&utm_medium=referral" \l "zifujichanjianchuli)
  + [字符集的正确实践](https://www.cnblogs.com/geaozhang/p/6724393.html?utm_source=itdadao&utm_medium=referral" \l "zhengqueshijian)
* [MySQL的校对规则](https://www.cnblogs.com/geaozhang/p/6724393.html?utm_source=itdadao&utm_medium=referral" \l "jiaoduiguiz)

一、字符集(Character set)

　　是多个字符(英文字符，汉字字符，或者其他国家语言字符)的集合，字符集种类较多，每个字符集包含的字符个数不同。

特点：

①字符编码方式是用一个或多个字节表示字符集中的一个字符

②每种字符集都有自己特有的编码方式，因此同一个字符，在不同字符集的编码方式下，会产生不同的二进制

常见字符集：

　　ASCII字符集：基于罗马字母表的一套字符集，它采用1个字节的低7位表示字符，高位始终为0。

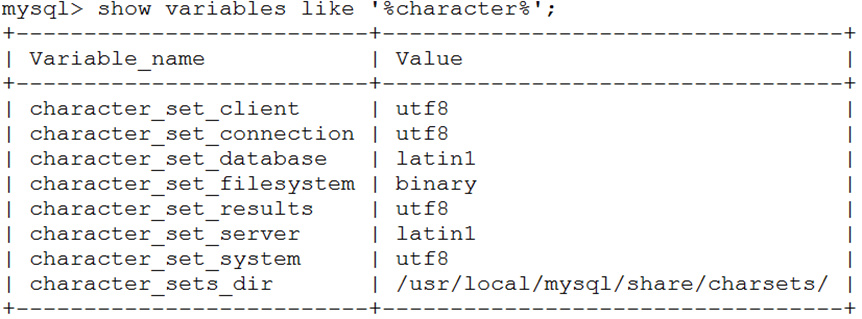
　　LATIN1字符集：相对于ASCII字符集做了扩展，仍然使用一个字节表示字符，但启用了高位，扩展了字符集的表示范围。

　　GBK字符集：支持中文，字符有一字节编码和两字节编码方式。

　　UTF8字符集：Unicode字符集的一种，是计算机科学领域里的一项业界标准，支持了所有国家的文字字符，utf8采用1-4个字节表示字符。

1、MySQL与字符集

　　只要涉及到文字的地方，就会存在字符集和编码方式。MySQL系统变量值：



2、正确使用字符集

　　数据库服务端的字符集具体要看存储什么字符



以上这些参数如何起作用：

1.库、表、列字符集的由来

　　①建库时，若未明确指定字符集，则采用character\_set\_server指定的字符集。

　　②建表时，若未明确指定字符集，则采用当前库所采用的字符集。

　　③新增时，修改表字段时，若未明确指定字符集，则采用当前表所采用的字符集。

2.更新、查询涉及到得字符集变量

　　更新流程字符集转换过程：character\_set\_client-->character\_set\_connection-->表字符集。

　　查询流程字符集转换过程：表字符集-->character\_set\_result

3.character\_set\_database

　　当前默认数据库的字符集，比如执行use xxx后，当前数据库变为xxx，若xxx的字符集为utf8，那么此变量值就变为utf8(供系统设置，无需人工设置)。

3、MySQL客户端与字符集

1.对于输入来说：

　　客户端使用的字符集必须通过character\_set\_client、character\_set\_connection体现出来：

　　　　①在客户端对数据进行编码（Linux：utf8、windows：gbk）

　　　　②MySQL接到SQL语句后(比如insert)，发现有字符，询问客户端通过什么方式对字符编码：客户端通过character\_set\_client参数告知MySQL客户端的编码方式(所以此参数需要正确反映客户端对应的编码)

　　　　③当MySQL发现客户端的client所传输的字符集与自己的connection不一样时，会将client的字符集转换为connection的字符集

　　　　④MySQL将转换后的编码存储到MySQL表的列上，在存储的时候再判断编码是否与内部存储字符集（按照优先级判断字符集类型）上的编码一致，如果不一致需要再次转换

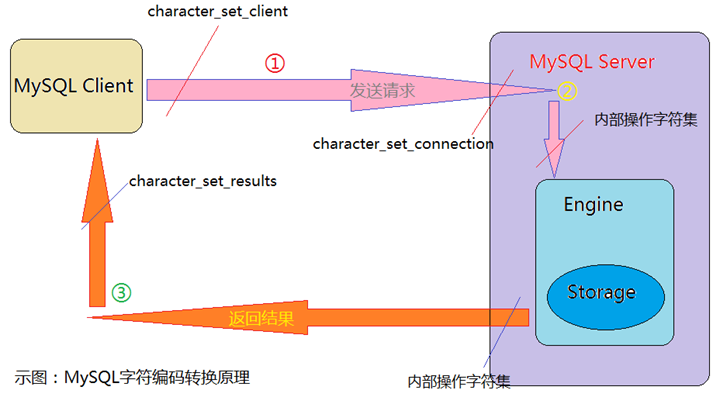
2.对于查询来说：

　　客户端使用的字符集必须通过character\_set\_results来体现，服务器询问客户端字符集，通过character\_set\_results将结果转换为与客户端相同的字符集传递给客户端。(character\_set\_results默认等于character\_set\_client)

4、MySQL字符编码转换原理：

　　问：若character\_set\_client为UTF8，而character\_set\_database为GBK，则会出现需要进行编码转换的情况，字符集转换的原理是什么？

　　答：假设gbk字符集的字符串“你好”，需要转为utf8字符集存储，实际就是对于“你好”字符串中的每个汉字去utf8编码表里面查询对应的二进制，然后存储。



图解字符集转换过程：

　　①MySQL Server收到请求时将请求数据从character\_set\_client转换为character\_set\_connection;

　　②进行内部操作前将请求数据从character\_set\_connection转换为内部操作字符集

确定步骤：

--使用每个数据字段的CHARACTER SET设定值；

--若上述值不存在，则使用对应数据表的DEFAULT CHARACTER SET设定值；

--若上述值不存在，则使用对应数据库的DEFAULT CHARACTER SET设定值；

--若上述值不存在，则使用character\_set\_server设定值；

　　③将操作结果从内部操作字符集转换为character\_set\_results。

5、字符集常见处理操作

1.查看字符集编码设置

　　mysql> show variables like '%character%';

2.设置字符集编码

　　mysql> set names 'utf8';

相当于同时：

　　set character\_set\_client = utf8;

　　set character\_set\_results = utf8;

　　set character\_set\_connection = utf8;

3.修改数据库字符集

　　mysql> alter database database\_name character set xxx;

只修改库的字符集，影响后续创建的表的默认定义；对于已创建的表的字符集不受影响。（一般在数据库实现字符集即可，表和列都默认采用数据库的字符集）

4.修改表的字符集

　　mysql> alter table table\_name character set xxx；

只修改表的字符集，影响后续该表新增列的默认定义，已有列的字符集不受影响。

　　mysql> alter table table\_name convert to character set xxx;

同时修改表字符集和已有列字符集，并将已有数据进行字符集编码转换。

5.修改列字符集

格式：

ALTER TABLE table\_name MODIFY

column\_name {CHAR | VARCHAR | TEXT} (column\_length)

    [CHARACTER SET charset\_name]

    [COLLATE collation\_name]

mysql> alter table table\_name modify col\_name varchar(col\_length) character set xxx;

6、字符集的正确实践：

MySQL软件工具本身是没有字符集的，主要是因为工具所在的OS的字符集（Windows：gbk、Linux：utf8），所以字符集的正确实践非常重要：

　　1.对于insert来说，character\_set\_client、character\_set\_connection相同，而且正确反映客户端使用的字符集

　　2.对于select来说，character\_set\_results正确反映客户端字符集

　　3.数据库字符集取决于我们要存储的字符类型

　　4.字符集转换最多发生一次，这就要求character\_set\_client、character\_set\_connection相同

　　5.所有的字符集转换都发生在数据库端

综述：

　　1、建立数据库的时候注意字符集（gbk、utf8）；

　　2、连接数据库以后，无论是执行dml还是select，只要涉及到varchar、char列，就需要设置正确的字符集参数。

二、校对规则collation校对

　　字符集是一套符号和对应的编号

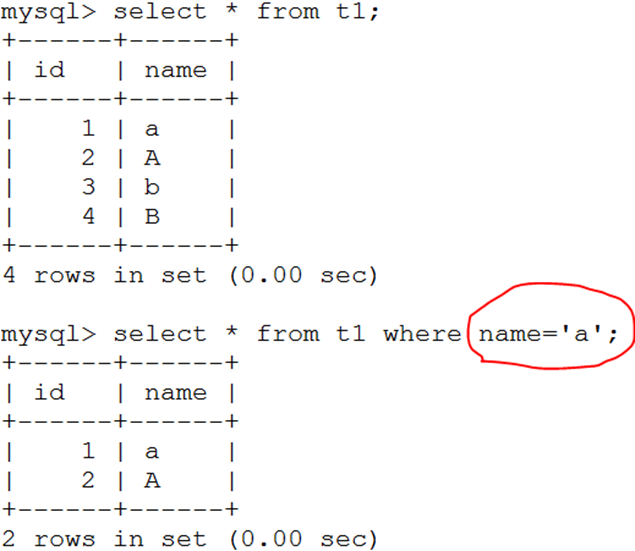
查看数据库支持的所有字符集(charset)：

mysql> show character set;

校对规则(collation)：

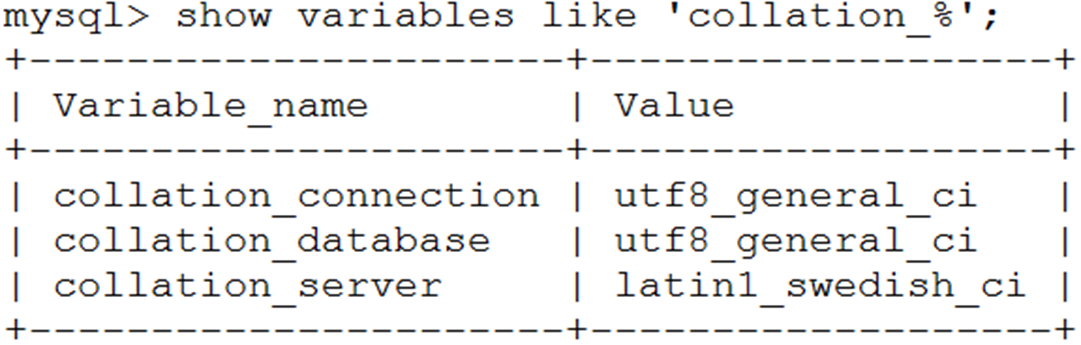
　　是在字符集内用于字符比较和排序的一套规则，比如有的规则区分大小写，有的则无视。

mysql> create table t1(id int,name varchar(20));    #t1建表没有指定校对规则



 mysql> show collation;  #查看数据库支持的所有校对规则

mysql> show variables like 'collation\_%';  #查看当前字符集和校对规则设置



校对规则特征：

　　①两个不同的字符集不能有相同的校对规则；

　　②每个字符集有一个默认校对规则；

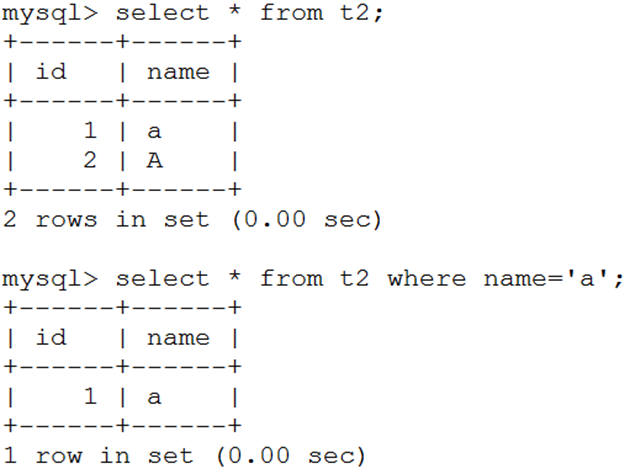
　　③存在校对规则命名约定：以其相关的字符集名开始，中间包括一个语言名，并且以\_ci（大小写不敏感）、\_cs（大小写敏感）或\_bin（二元）结束。

注意：

　　系统使用utf8字符集，若使用utf8\_bin校对规则执行SQL查询时区分大小写，使用utf8\_general\_ci不区分大小写(默认的utf8字符集对应的校对规则是utf8\_general\_ci)。

示例：

mysql> create table t2(id int,name varchar(20)) character set=gbk collate=gbk\_bin;    #t2建表指定校对规则（区分大小写）



@author：http://www.cnblogs.com/geaozhang/