在使用xutils时post请求传递中文到服务端Mysql数据库出现中文乱码。

这个问题困惑了我很久，也走了很多弯路。经过不断的探究终于找到了问题的解决方案。

第一：查看自己的mysql的编码。cmd—>mysql –uroot –p -->输入自己的mysql的密码—>输入 SHOW VARIABLES LIKE 'character%';（如果装mysql时默认安装的话，会出现很多latin1，而不是我下面的全部utf-8,这是因为mysql默认的编码是latin1）,如果全是Latin1的话，想要不出现中文乱码的话，就要改为utf-8了。

第二：更改Latin1为utf-8。找到安装mysql的路径下的my.ini文件。打开这个文件。

分别在：[client] 下一行加入default-character-set=utf8

[mysql] 下一行加入default-character-set=utf8

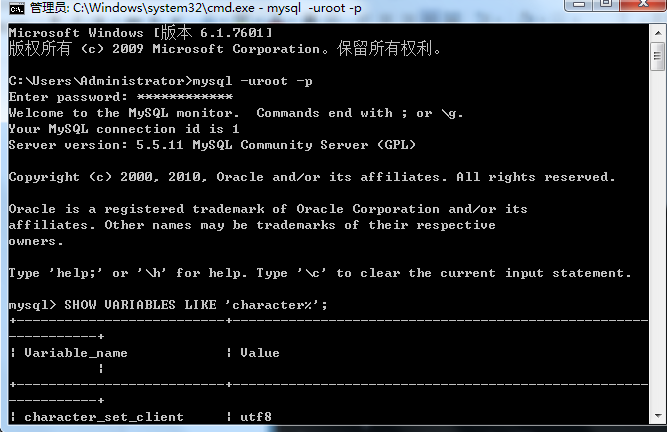
[mysqld] 下一行加入character-set-server=utf8（这里注意较老的版本mysql加入default-character-set=utf8是可以的，但5.5就不可以了，重启mysql就会报错）

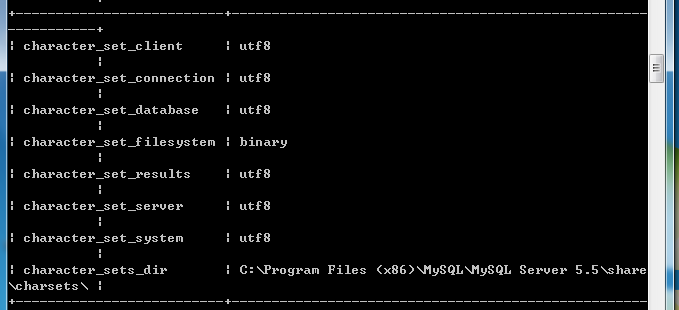
找到# The default character set that will be used when a new schema or table is

# created and no character set is defined

character-set-server= latin1改为character-set-server=utf8。

这样就可以将Latin1为utf-8。重启mysql（右击我的电脑->管理->服务与应用程序->服务，找到mysql重启），当你重复第一步骤时会发现。大多数Latin1已经变为utf-8。可惜，可惜，有一项（character-set-server）还是为Latin1，这是由于当时装mysql时，默认安装时server编码默认为Latin1，这时，不要慌，mysql还是挺好的，可以重新设置编码。找到mysql的bin目录下的MySQLInstanceConfig.exe，启动这个就可以程序，将server的编码改为utf-8就行。重启mysql，执行第一步骤。会发现所有的Latin1已经改为utf-8(当然有一项filesystem就是binary，无需改的)。





第三：如果你使用的tomcat进行通信的话，由于tomcat的默认编码是iso8859-1。这是就需要修改tomcat的编码：修改tomcat下的conf/server.xml文件

找到如下代码：

<Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000" redirectPort="8443" />

添加URIEncoding="UTF-8" 就行了。

第四：在传输中文时用

String deviceIntoTime=deviceInfEntity.getDeviceIntoTime();

//如果有中文，要进行URLEncoder.encode 方法

params.addBodyParameter("deviceIntoTime", URLEncoder.encode(deviceIntoTime, "utf-8"));

接收时用：

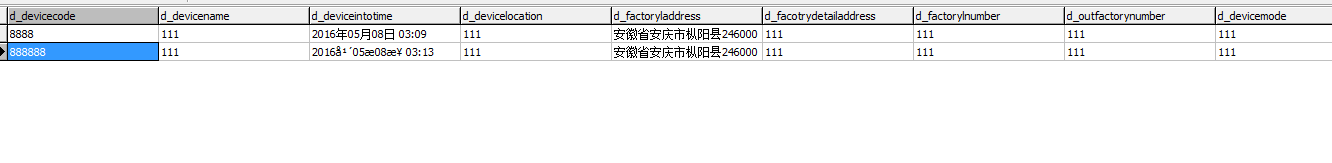
String deviceIntoTime = URLDecoder.decode(request.getParameter("deviceIntoTime"), "utf-8");

第五：经过上面的步骤，存储到mysql的中文数据还是乱码，这是你的数据库建立有问题了。

比如我就出现：

Incorrect string value: '\\xE6\\xB7\\xB1\\xE5\\x85\\xA5...' for column 'title' at row 1"

这时修改数据库的属性，选择charset为utf-8,collation为utf8-unicode-ci。记得还有表中涉及中文的列也要选择charset为utf-8,collation为utf8-unicode-ci。这样就行了。



总结：经过这个问题，我明白了mysql和tomcat的编码形式，如何传递中文不乱码。这个问题困惑了我很久，终于解决了。