# Qt中文显示乱码解决方法

2019-05-02 11:00:23 [byxdaz](https://me.csdn.net/byxdaz" \t "https://blog.csdn.net/byxdaz/article/details/_blank) 阅读数 873更多

分类专栏： [QT](https://blog.csdn.net/byxdaz/category_8858602.html" \t "https://blog.csdn.net/byxdaz/article/details/_blank)

版权声明：本文为博主原创文章，遵循[CC 4.0 BY-SA](http://creativecommons.org/licenses/by-sa/4.0/" \t "https://blog.csdn.net/byxdaz/article/details/_blank)版权协议，转载请附上原文出处链接和本声明。

本文链接：<https://blog.csdn.net/byxdaz/article/details/89743257>

Qt中文显示乱码解决方法  
一、Qt4中的解决方法  
方法1：在程序开始时增加初始编码类型  
#include < QTextCodec >  
int main(int argc, char \*\*argv)  
{  
....................  
QTextCodec::setCodecForTr(QTextCodec::codecForName("UTF8"));  
QTextCodec::setCodecForLocale(QTextCodec::codecForName("UTF8"));  
QTextCodec::setCodecForCStrings(QTextCodec::codecForName("UTF8"));  
..........................  
}  
Qt5中， 取消了QTextCodec::setCodecForTr()和QTextCodec::setCodecForCString()这两个函数，而且网上很多都是不推荐这种写法。  
方法2：直接转换  
加上#include <qtextcodec.h>   
QTextCodec \*codec = QTextCodec::codecForName("GBK");//修改这两行   
w.setWindowTitle(codec->toUnicode("中文按钮"));   
完整代码：

#include "helloqt.h"

#include <QtWidgets/QApplication>

#include <qlabel.h>

#include <qtextcodec.h>

int main(int argc, char \*argv[])

{

    QApplication a(argc, argv);

    HelloQt w;

    QTextCodec \*codec = QTextCodec::codecForName("GBK");

    w.setWindowTitle(codec->toUnicode("中文按钮"));

    w.resize(300, 140);

    QLabel label("test",&w);

    label.setGeometry(100, 50, 160, 30);

    w.show();

    return a.exec();

}

二、Qt5中的解决方法  
方法1：先将对应的cpp文件用windows自带的记事本打开，另存为UTF-8格式，然后在代码中，遇到中文字符，使用QStringLiteral("中文")进行修饰。这种方式每次新建一个class就要切出去用记事本编辑一下实在麻烦，而且每个中文字符串都要用QStringLiteral来修饰，实在麻烦。

#include "helloqt.h"

#include <QtWidgets/QApplication>

#include <qlabel.h>

int main(int argc, char \*argv[])

{

    QApplication a(argc, argv);

    HelloQt w;

    w.setWindowTitle(QStringLiteral("中文按钮"));

    w.resize(300, 140);

    QLabel label("test",&w);

    label.setGeometry(100, 50, 160, 30);

    w.show();

    return a.exec();

}

方法2：在头文件申明中加上  
#pragma execution\_character\_set("utf-8")  
方法3：直接转换  
加上#include <qtextcodec.h>   
QTextCodec \*codec = QTextCodec::codecForName("GBK");//修改这两行   
w.setWindowTitle(codec->toUnicode("中文按钮"));