通过QFont可以部分Qt控件的文字的大小、字体、字间距、下划线等属性。

 通过QFontInfo以及QFontMetrics，可以获取文字的一些属性。

使用qApp->setFont(QFont())可以设置默认文字属性，但是对于带有文字属性编辑功能的Qt控件（QTextEdit、QTextBrowser），使用setFont()不会“完全”生效。

还有一个字体设置对话框类：QFontDialog。使用getFont手动设置文字属性：

//定义bool型输出变量

bool enable = false;

//调用文字属性设置变量,传输输入变量enable用于保存是否点击了ok按钮，第二个参数为文字属性对话框默认值

QFont font = QFontDialog::getFont(&enable,QFont("宋体",20,QFont::Bold),this,"自定义文字属性");

//当点击属性对话框的ok按钮后，变量enable的值为true

if(enable)

{

ui->label->setFont(font);

}

 //调用各种方法设置

// qApp->setFont(QFont("宋体",20,QFont::Bold));

QFont font;

//设置文字字体

font.setFamily("宋体");

//设置文字大小为50像素

font.setPixelSize(50);

//设置文字为粗体

font.setBold(true); //封装的setWeight函数

//设置文字为斜体

font.setItalic(true); //封装的setStyle函数

//设置文字大小

font.setPointSize(20);

//设置文字倾斜

font.setStyle(QFont::StyleItalic);

//设置文字粗细//enum Weight 存在5个值

font.setWeight(QFont::Light);

//设置文字上划线

font.setOverline(true);

//设置文字下划线

font.setUnderline(true);

//设置文字中划线

font.setStrikeOut(true);

//设置字间距%

font.setLetterSpacing(QFont::PercentageSpacing,300); //300%,100为默认

//设置字间距像素值

font.setLetterSpacing(QFont::AbsoluteSpacing,20); //设置字间距为100像素

//设置首个字母大写（跟参数有关，也可以设置全部大写AllUppercase）

font.setCapitalization(QFont::Capitalize);

//通过QFontMetrics获取字体的值

QFontMetrics fm(font);

qDebug() << fm.height(); //获取文字高度

qDebug() << fm.maxWidth(); //获取文字宽度

//通过QFontInfo获取也能获取字体信息

QFontInfo fInfo(font);

qDebug() << fInfo.family() <<" "<<fInfo.style() << fInfo.pixelSize() << fInfo.overline();

//设可以单独置QPlainTextEdit字体

//ui->plainTextEdit->setFont(font);

//将当前设置的字体设置为默认字体

qApp->setFont(font);

————————————————

版权声明：本文为CSDN博主「jhe\_zhang」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/jiezhj/article/details/37597929