

问题解决：

<http://www.bubuko.com/infodetail-505888.html>

repidxml

tinyxml

QQ TreeWdiget 抽屉效果

<http://blog.csdn.net/knowledgeapple/article/details/7990399>

Qlistwidget 在item中添加下划线

myListWidget->setStyleSheet( "QListWidget::item { border-bottom: 1px solid black; }" );

字体选中了颜色跟原来的一样(使用了代理)

<http://www.linuxidc.com/Linux/2011-04/34166.htm>

QSS

Qt 4.8: QAbstractItemModel Class Reference

Setstyle(“background-color : rbg(106,57,6,0); border:0px solid #e1e0e0;”)

设置背景颜色透明度为零 边框为0的控件

: fatal error LNK1104: 无法打开文件“debug\maskdemo.exe”

把生成的release文件删除重新构建即可

若碰到qt外遇到注释错误关闭

可能是由于文件编码格式不同造成的，linux上是utf-8

Windows上是gba

所以需要windows上按gba保存

QT无法调试程序输出 是因为开了两个 qt creator

这会导致测试连接信号与槽在后面使用qdebug()的时候输出不了测试数据！

【qt4】Qtreewidget item是中文点击之后没有出现itemclicked信号原因是因为解码的问题需要添加Q textCodec

Button设置样式表可以修改它的背景图片点击效果大小边框等

也可以通过Qplaette调色板+Qbrush画刷修改其背景图片

一张图片分成几部分显示，如按钮的点击松开hover显示等

<http://bbs.csdn.net/topics/390289292>

编码问题参考【[解决Qt中文乱码以及汉字编码的问题(UTF-8/GBK)](http://blog.csdn.net/libaineu2004/article/details/19245205)】

<http://blog.csdn.net/libaineu2004/article/details/19245205>

编码问题：gbk编码一个汉字表示2个字节，utf8则是一个汉字表示3个字节

注意utf8 Bom问题 BOM—Byte Order Mark 字节序标记

Windows是使用Bom来标记文本文件的编码方式的(网页不要使用BOM)

Qt 动画效果展示

如果继承qwidget的类在定义控件的时候控件没有显示，那么设置该控件的父类

或者是控件太小了

Qt stackedwidget虽然不能添加布局，但是可以将布局父类设为stackedwidget。

图案分类左边的控件使用listwidget。

QmainWindow Qdialog 跟Qwidget区别。

继承QmainWindow会自定生成一个widget，

而QWidget 跟QDialog 本身是没有自动生成widget的

而QWidget跟QDialog则需要设置布局，然后将控件添加进去显示。

代码控制鼠标样式状态

ui->InForButton->setStyleSheet("QPushButton{border-image: url(:/image/通知.png);}"

"QPushButton:hover{border-image: url(:/image/通知ss.png);}"

"QPushButton:pressed{border-image: url(:/image/通知s.png);}");

数据统计

第三方插件chartDirector 商用收费的…

Qt本身提供QWT

以及QcustomPlot

<http://echarts.baidu.com/demo.html#line-marker>等

折线图，饼状图效果

【

option = {

backgroundColor: '#BA9463',//背景色

color:['dark'],

title: {

text: '营业额',

subtext: ''

},

tooltip: {

trigger: 'axis'

},

legend: {

data:['最大值']

},

xAxis: {

splitLine:false,

type: 'category',

boundaryGap: false,

data: ['周一','周二','周三','周四','周五','周六','周日']

},

yAxis: {

splitLine:false,

type: 'value',

axisLabel: {

formatter: '{value} '

}

},

series: [

{

name:'最高额',

type:'line',

data:[9, 18, 30, 20, 35, 20, 40],

markPoint: {

color:['dark'],

data: [

{type: 'max', name: '最大值'},

{type: 'min', name: '最小值'}

]

},

markLine: {

color:['dark'],

data: [

{type: 'average', name: '平均值'}

]

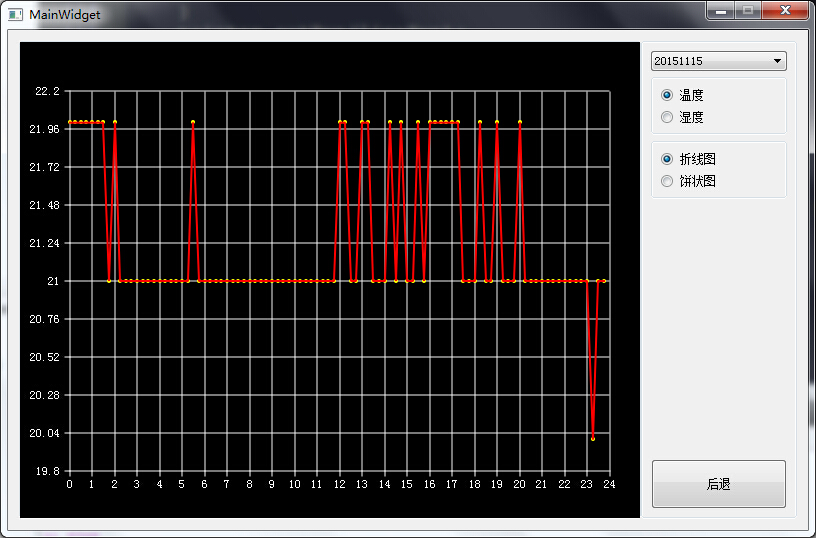
}

},

]

};

】



Qwt

可以实现

源码在qqfile

Qt msvc2013版本debug不能显示图片，换成release即可

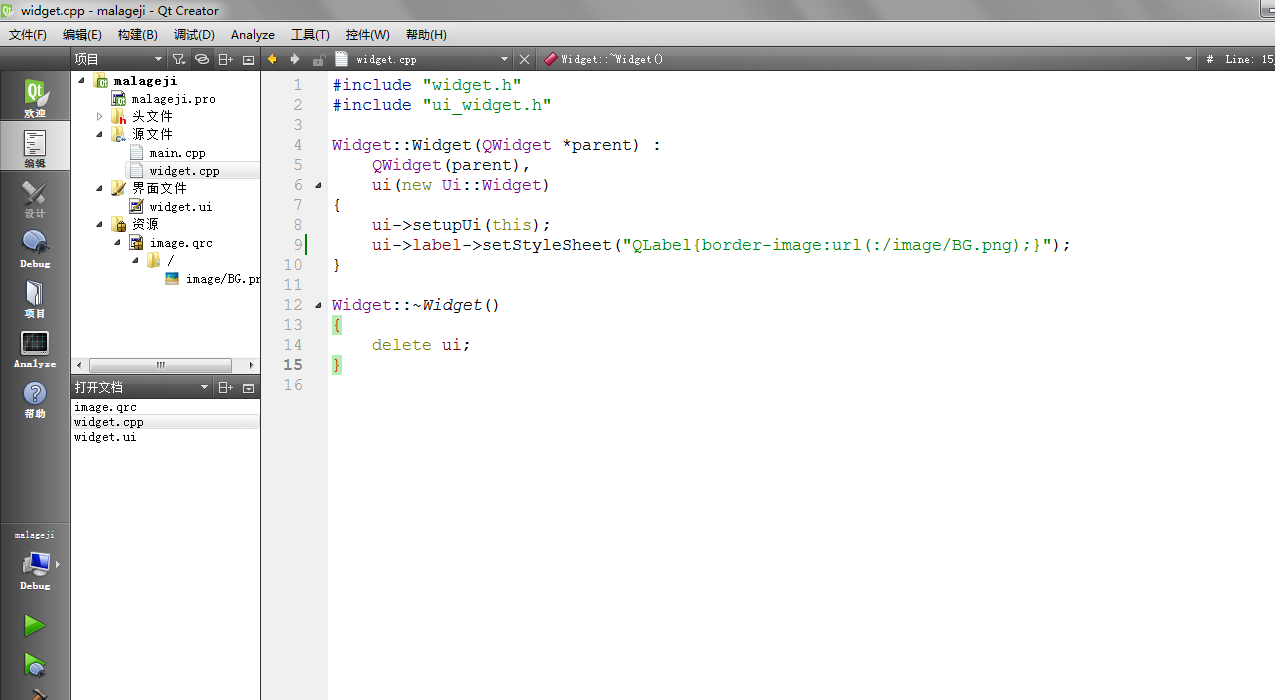
如果是当前已经构建了debug的版本显示不了，那么需要清理当前工程，重新构建release版本即可。但是如果又用debug版本的话还是显示不了，网上有说参考把C:\Qt\Qt5.5.1\5.5\msvc2013\plugins\imageformats

这个路径下的imageformats考到exe同级目录下即可。

最终malageji.pro上测试成功【不知道是什么原因了】

上面说的，执行应该是对应release的qwt.dll调用这个，若是debug则是调用qwtd.dll

执行以下qmake



at(1)相当引用数组下表为1的变量

**修改基类：**

因为Qt中继承的基本类型就MainWindow、QDialog、QWidget，这是创建工程文件的时候只能选这三个，然而当需要重新实现某个功能函数的时候就需要继承这个类，如实现delegate在qlistview，那么就需要去继承这个QitemDelegate然后自定义这个委托，并且设置这个委托的位置，重载paint(\*\*\*\*\*);去实现，重新构建控件的位置大小以及图片样式等。

样式表设计格式例子：

例如：QpushButton

.设置所有的相同部件都使用相同的样式。我们在mainwindow.cpp的构造函数中添加如下代码，这种方式是给所有QPushButton类对象设置的样式：setStyleSheet("QPushButton { color: white }");

这种是为某个Btn设置样式表，文本颜色为蓝色

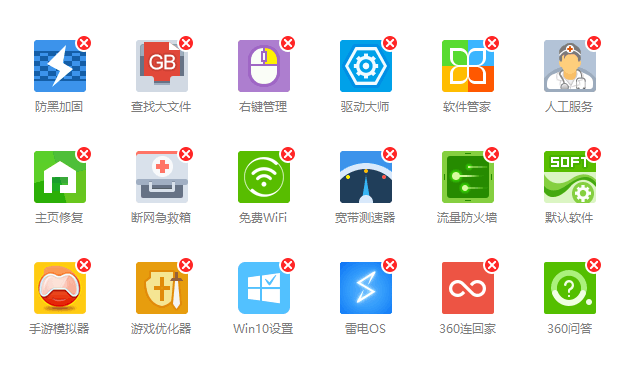
ui->pushButton->setStyleSheet("color: blue");

这种是为某个Btn设置样式表，结果为红底蓝字ui->pushButton->setStyleSheet("background-color:red; color: blue");

Qt 本身有提供操去实现 窗体的最大化、最小化、关闭等

QWidget：：showMaximized(); 最大  
QWidget：：showMinimized();最小  
QWidget：：showNormal();还原

实现这个需要tableview+delegate 已经实现



静态页面new一次就使用，避免内存泄露，看下叶海的demo

一般情况下new出来的控件后面没有父类的话就是0x0【十六进制等于0】

如果加上父类设置this的话，在qDebug()输出的是当前类名以及该类的地址.

Setfont 设置字体

setFont(QFont("微软雅黑",14,1))

QTableView {

border: 1px solid rgb(45, 45, 45);

background: rgb(57, 58, 60);

gridline-color: rgb(60, 60, 60);

}

QTableView {

background: rgb(0,0,0);

}

QTableView::item {

border: 1px solid rgb(255,255,255);0,0,0黑色，255,255,255白色

background: rgb(0, 0, 0);

border-right: 1px solid rgb(45, 45, 45); //item右边留的空白长度以及颜色

border-bottom: 1px solid rgb(45, 45, 45); //item下面留的空白以及颜色

}

QTableView::item:selected {background: rgba(255, 255, 255, 40);}

QTableView::item:selected:!active {color: white;}

QTableView::item:!selected {background: rgba(255, 255, 255);} item不被选中的时候颜色默认为白色，但是我单纯想让字体为白色

//模型文本居中对齐

case Qt::TextAlignmentRole:

{

if(role == Qt::TextAlignmentRole)

return Qt::AlignCenter;

break;

}

设置编码

QTextCodec::setCodecForTr(QTextCodec::codecForName("system"));

Qsqldatabase:

QSqlQuery sql\_query;

sql\_query.prepare(str);

qt 自带sqlite用法，不需要配置任何文件，不想odbc那么复杂，轻量级数据传输，简单方便,数据库乱码的话要在数据库设置编码。

创建数据库类型，QSQLITE，sqlserver是ODBC

QSqlDatabase database = QSqlDatabase::addDatabase("QSQLITE");

创建数据库名字

database.setDatabaseName("mydatabase.db");

创建用户名

database.setUserName("root");

创建密码

database.setPassword("123456");

动态建表使用Qstring

QString create\_sql = "create table %1 (time varchar(10) primary key, H2S int, CO int, SO2 int, CF4 int)";

在程序结束的时候数据库需要关闭。

动态插入数据，多个表，也是使用Qstring

QString insert\_sql = "INSERT INTO %1 (time, H2S, CO, SO2, CF4)""VALUES (:time, :H2S, :CO, :SO2, :CF4)";

同理查询全部数据或者部分数据，删除都是一样的，使用Qstring进行动态操作

QString select\_all\_sql = "select \* from %1";

QSqlQuery sql\_query;

sql\_query.prepare(“数据库语句”);

sql\_query.exec();执行刚才操作的数据库语句，返回bool值，出错根据qsqlerror判断。

基本用法增删改查，速度的话需要使用连接查询等优化。未设置主键等。

qDebug()<<\_\_FILE\_\_<<\_\_LINE\_\_<<\_\_FUNCTION\_\_;

\_\_FILE\_\_ 输出信息在哪个文件下被调用

\_\_LINE\_\_ 输出信息在哪一行被调用

\_\_FUNCTION\_\_输出信息在哪个功能模块下被调用

//magnifier->setMouseButton(Qt::LeftButton,Qt::NoModifier);//左点击放大缩小

Linux下Qt碰到Error While building/deploying project sc(kit：桌面/4.8.6)

When executing step “Make”

这个是执行人家编译过的程序然后找不到对应的debug，这种情况下就需要删除\*.user文件，然后使用qt creator重新打开.pro工程文件，然后重新编译即可。

CrossEffectDevice\* ef = d->model->data(d->model->index(index.row(), 0), Qt::UserRole).value<CrossEffectDevice\*>();

上面这句等价下面这句也就是类型转换

//==CrossEffectDevice\*ef=d->model->data(d->model->index(index.row(), 0), Qt::UserRole).toCrossEffectDevice\*;

Qwt中折线的隐藏与显示

折线是通过curve->attached(qwt)添加进来的

同理可以使用curve->detached();来清除

但是如果数据大了建议使用下面方法

Curve->hide();

Curve->show();

无窗口边框最小化之后窗口控件不刷新也不改变解决办法

<http://bbs.csdn.net/topics/391955966>

以及qwt不能qdebug跟release混编，

解决办法：

先把qwtd.dll  qwt.dll考到新建文件夹lib下，然后对应使用的编译方式就调用对应的dll

在pro工程文件中调用：

debug：      LIBS += -L$$PWD/lib/  -l pwtd               //debug编译的时候就把下面release注释

release ： 　LIBS += -L$$PWD/lib/  -l pwt                 //同理要发布的时候则把上面debug给注释掉否则会崩溃

注意：不能同时调用，即不能混合编译，默认调用第一条，若是debug在release，然后运行release的话一样会出现runtime error，

1. #registrationDialog QLineEdit { **color**: brown }
2. 选择器修改QLineEdit的样式
3. QLineEdit { **color**: red }
4. QLineEdit[readOnly="true"] { **color**: gray }
5. #registrationDialog QLineEdit { **color**: brown }
7. // 动态属性，setProperty
9. \*[mandatoryField="true"]
10. {
11. **background-color**: yellow
12. }

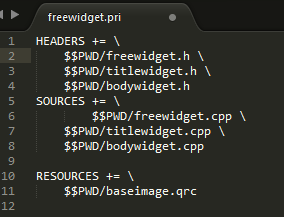
如何使用.Pri文件

在pro文件中添加 INCLUDEPATH+=$$PWDMyPro/文件夹名字

include(文件夹名字/文件.pri)



而pri中的资源文件是后面加入工程自己在添加的如下



Qt助手上也有比较详细的说明，选择器以及伪选择器，现在只是做个简单的搬运&翻译工。

当然也可以重载QPushButton 然后跟踪鼠标党鼠标滑过Button的时候，使用画板去更改ButtonText的颜色。

MyButton= new QPushButton("LiuLikeQian",this);

MyButton->setStyleSheet("QPushButton{background:transparent;font:12pt 微软雅黑 blod;}"

　　　　　　　　　　　　　　"QPushButton:hover{color:red;}"

　　　　　　　　　　　　　　"QPushButton:pressed{color:blue}");

QT cl: 命令行 error D8004 :“/I”需要参数

参数定义缺少new

参数声明对应应该是指针却少了\*

指针引用成员函数应该用箭头，却用成了.