

- 2550mAh电
- 日本雾霾之战：民众与政府的博弈
- 南非猎豹冒死捕杀豪猪被扎满嘴刺
- 别光顾着把枪口对准柴静
- 马来西亚男子与眼镜王蛇接吻(图)
- 【早评】降息利好将会在后市中逐
- 哪些人能从柴静的“穹顶之下”赚
- “深圳机场撞人事件”不是一个人
- 政采剔除国外品牌：早该说了！
- 【DIY】做一枚宫灯迎元宵



[查看更多>>](#)

谁看过这篇博文

	3月2日
	2月28日
	2月26日
	2月23日
	2月22日
	2月13日
	2月12日
	2月6日
	2月6日
	2月4日
	2月3日
	1月29日

```
font.family: "Helvetica"
font.pointSize: 20
color: "white"
}

Text {
    color: Qt.rgb(0, 0, 255, 100)
    font.pointSize: 16
    text: Qt.md5("hello, world")
}
}
```

Window: 窗体
Text: 既可以显示普通文本和富的文本项。例如，可以定义特定字体、大小、颜色的文本。
Column : 子项按列显示
width、height、color、text等就不过多说明了，分别代表：宽、高、颜色、文本。
anchors.centerIn: parent表示位于窗体中央
spacing: 10表示各个子项的间距为10
Qt.md5("hello, world")使用md5对"hello, world"加密

运行：

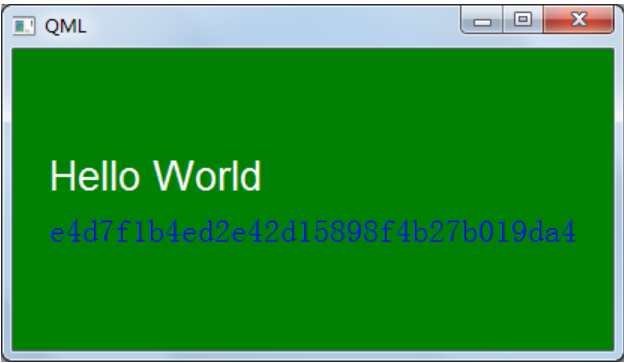
- 方式一：右键.qml文件->打开方式（选择qml.exe或qmlviewer.exe）即可。
- 方式二：使用QQuickView模块。

在main方法中添加以下代码：

```
QQmlEngine *engine = new QQmlEngine();
QQmlComponent *component = new QQmlComponent(engine);
component->loadUrl(QUrl("qrc:/qml/HelloWorld.qml")); //QUrl::fromLocalFile("HelloWorld.qml")
if(!component->isReady())
{
    qDebug() < "Warning: " < QString(component->errorString());

    return -1;
}
QObject *object = component->create();
QQuickWindow *window = (qobject_cast<QQuickWindow*>)(object);
window->show();
```

效果如下：



使用VS，有可能遇到以下错误：

fatal error C1083: 无法打开包括文件：“GLSL/glsl.h”：No such file or directory

解决办法如下：

右键工程->属性->C/C++->常规->附加包含目录（添加：\$(QTDIR)\include\QtANGLE）即可！

在Qt5，所有QML应用呈现一个OpenGL场景图的架构，而不是在Qt4使用的图形视图框架。由于这种建筑的变化规模，在C++ API已经被广泛的重组和的QtDeclarative模块已被弃用换为成两个新模块：Qt QML，它实现的QML引擎和语言的基础设施，和Qt Quick，它实现了视觉帆布和场景图的后端。

以前在QtDeclarative模块的所有类都已经放入了Qt QML和Qt Quick的模块，他们的类的名称已被更改，以反映其新的模块位置。类名的变化如下：

Qt QML	Qt Quick
<ul style="list-style-type: none">QDeclarativeComponent -> QQmlComponentQDeclarativeContext -> QQmlContextQDeclarativeEngine -> QQmlEngineQDeclarativeError -> QQmlErrorQDeclarativeExpression -> QQmlExpressionQDeclarativeExtensionPlugin -> QQmlExtensionPluginQDeclarativeInfo -> QQmlInfoQDeclarativeListReference -> QQmlListReferenceQDeclarativeNetworkAccessManagerFactory -> QQmlNetworkAccessManagerFactoryQDeclarativeParserStatus -> QQmlParserStatusQDeclarativeProperty -> QQmlPropertyQDeclarativePropertyMap -> QQmlPropertyMapQDeclarativePropertyValueSource -> QQmlPropertyValueSourceQDeclarativeScriptString -> QQmlScriptString	<ul style="list-style-type: none">QDeclarativeItem -> QQuickItemQDeclarativeView -> QQuickViewQDeclarativeImageProvider -> QQuickImageProvider

在Qt5中要使用的新QQml* QQuick*类，需要在.pro文件中引入适宜的模块链接。例如，下面的Qt QML和Qt Quick的模块链接：

- QT += qml quick
- 然后需要包括需要的头文件：
- #include
 - #include
- (QtDeclarative模块仍然适用于Qt Quick 1，但是，它不应该被用于新的应用程序)

注：
技术在于交流、沟通，转载请注明出处并保持作品的完整性。
作者：`☆奋斗ing♥孩子` 原文：http://blog.sina.com.cn/s/blog_a6fb6cc90101g8t8.html。

3

喜欢

分享：       

阅读(1475) | 评论 (1) | 收藏(0) | 已有1人转载▼ | 喜欢▼ | 打印

已投稿到： 排行榜

前一篇：Qt之360安全卫士（勋章墙）
后一篇：QML之Text

评论

重要提示：警惕虚假中奖信息

[发评论]

用户6chyzhnsho

关于qml学习的









www.seanyxie.com



2014-10-4 20:21

回复 (0)

发评论

一去、二三里：

☐  分享到微博 

☐ 匿名评论

验证码: 请点击后输入验证码 收听验证码

发评论

以上网友发言只代表其个人观点，不代表新浪网的观点或立场。

＜ 前一篇

Qt之360安全卫士（勋章墙）

后一篇 ＞

QML之Text

新浪BLOG意见反馈留言板 不良信息反馈 电话：4006900000 提示音后按1键（按当地市话标准计费） 欢迎批评指正
新浪简介 | About Sina | 广告服务 | 联系我们 | 招聘信息 | 网站律师 | SINA English | 会员注册 | 产品答疑

Copyright © 1996 - 2014 SINA Corporation, All Rights Reserved
新浪公司 版权所有