

九天雁翎的博客

做专业的程序, 做更专业的产品.

目录视图 摘要视图 [RSS](#) 订阅

个人资料



九天雁翎

访问: 1958968次
积分: 23968
等级:
排名: 第101名

原创: 428篇 转载: 22篇
译文: 8篇 评论: 2793条

公告

公司一直在招聘Android应用开发, 有兴趣的见[招聘网页](#)

此博客将为九天雁翎的博客(原程序即人生)的编程相关内容的镜像站, 除了编程以外的内容这里就不发了, 毕竟是CSDN嘛.

我的链接

[九天雁翎的博客\(原程序即人生\)](#)
[我的新浪微博](#)
[我的豆瓣](#)
[我的GitHub](#)

链接

[午后绿茶](#)
[酷壳](#)
[可能吧 \(RSS\)](#)
[clayman's blog \(RSS\)](#)
[实时渲染资源](#)
[史上最强游戏开发网站](#)

文章搜索

文章分类

[社区之星王海庆: 速度激情, 学习极客](#) [Markdown轻松写博文](#) [微信公众平台开发](#) [天天爱答题 一大波C币袭来](#) [读文章送好书](#)

PyQt的学习（1）入门

分类: [【QT】](#) [【Python】](#) 2009-06-16 01:22 32666人阅读 评论(8) 收藏 举报

[qt](#) [python](#) [mfc](#) [signal](#) [正则表达式](#) [button](#)
[目录\(?\)](#) [\[+\]](#)

PyQt的学习（1）入门

write by 九天雁翎(JTianLing) -- blog.csdn.net/vagrxie

[讨论新闻组及文件](#)

动机:

刚学习完Python,借鉴以前学习C++时的经验, 当时用MFC巩固了C++的学习, 并且可以做出实际的程序, 给了我继续学习的动力, 学习Qt也是出于类似的目的。

为什么我要写, 为了学习. 不仅仅为了将来有个可以用的, 好用的跨平台GUI库, 也为了用Python快速开发时也能有个GUI库, 但是又不喜欢TK, 那么, 综合考虑, Qt也就是不错的考虑了。

综述

这些文章基本遵循《C++ GUI Qt 4编程》第2版（C++ GUI Programming with Qt4, Second Edition）的流程, 文字可能会很少, 除非自己想出的例子, 不然例子就是书中C++例子的Python版本, 基本上, 可以以此书为蓝本对照这些文章中的例子来看。

虽然用途不算太大, 但是考虑到PyQt的好教材是如此的稀少, 这样也不算完全没有价值吧, 毕竟还有只明白Python不懂C++的人。

这里有一本PyQt的书《[GUI Programming with Python: Qt Edition](#)》，但是老到基本上是属于Qt2时代的東西了。

至于PyQt 嘛, 你不会不知道到google上搜索一下, 取得其下载地址吧? 安装方式如此简单, 点击下一步以后, 就可以在Python中通过import PyQt4来使用这个库了。

HelloWorld

自从K&R后, 好像所有的程序语言讲解都是以HelloWorld开始, 我们也不免俗吧,我们从HelloQT开始. 并且, 这个例子也可以作为一个测试程序, 测试一下看看安装PyQt是否成功。

HelloQt.pyw

```
1 import sys
2 from PyQt4 import QtGui
3
4
```

- 【C++】 (126)
- 【Ruby】 (3)
- 【Python】 (19)
- 【JavaScript】 (1)
- 【C#】 (2)
- 【lua】 (14)
- 【Haxe】 (1)
- 【iPhone】 (9)
- 【Android】 (5)
- 【Fun】 (9)
- 【Linux】 (12)
- 【Windows】 (6)
- 【QT】 (11)
- 【随笔】 (69)
- 【产品设计】 (1)
- 【图形技术】 (36)
- 【我自己写的程序】 (27)
- 【汇编】 (13)
- 【游戏】 (33)
- 【书籍小评】 (5)
- 【算法】 (22)
- 【网络】 (23)
- 【翻译】 (13)
- 【转载】 (21)

文章存档

- 2013年08月 (1)
- 2013年07月 (1)
- 2013年06月 (1)
- 2013年05月 (1)
- 2013年04月 (2)

展开

阅读排行

- 小小游戏程序员工作两年 (60138)
- Python与C之间的相互调 (39342)
- 网络协议编写的三层境界 (39305)
- PyQt的学习（1）入门 (32647)
- Google为啥没有Java的s (30457)
- 异常处理与MiniDump详 (28325)
- 新浪微博与腾讯微博的开 (24076)
- 国外大牛对几大著名开源 (22974)
- Spine的使用(With Coco (21812)
- 小小游戏程序员工作一年 (21362)

```
5 app = QtGui.QApplication(sys.argv)
6 label = QtGui.QLabel("Hello Qt!")
7 label.show()
8
9 sys.exit(app.exec_())
```

这样，就是一个现实"Hello Qt"的窗口，利用了QLabel部件。--说明一下，Qt中将MFC中常常称作控件（control）的东西称作部件（Widget），其实一回事。

这里也可以看出Qt的足够简洁，我很欣赏，而其对象使用的风格和方式也是比较好的，不然，尝试用VS生成一个类似的MFC程序试试？^^

其实这里还不足以见证Qt的强大,看下面的例子，QLabel竟然至此HTML样式。。。。-_-!

MoreHelloQt.pyw

```
1 import sys
2 from PyQt4 import QtGui
3
4
5 app = QtGui.QApplication(sys.argv)
6 label = QtGui.QLabel("Hello QT!")
7 label.show()
8
9 sys.exit(app.exec_())
```

Hello QT!

") #这里我没有办法让CSDN不将其解释为HTML。。所以，参考界面的源代码看看是什么吧

```
7 label.show()
8
9 sys.exit(app.exec_())
```

强大吧：)

Qt(C++) VS PyQt

这里，顺面比较一下PyQt与普通Qt（C++）生成程序的区别。一般而言，两者速度没有可比性，但是，速度在这里不是主要问题，原因在于PyQt的核心也就是Qt库，那是用C++写的，这样，一般而言不会占用太多时间的逻辑代码速度慢点，不会成为瓶颈，使得PyQt的程序可以跑得足够的快。但是，使用方式上，却没有失去Python的优雅语法，快速开发的能力，也结合了Qt的强大，呵呵，广告用语。。。。。。。。。。来点实在的。

左边的是用C++开发出来的Qt程序，右边是PyQt开发出来的程序，由于都是使用了同一个库，看不出两者的区别。基本上，多懂了一种语言，就多了一种选择。。。比如说这个时候的Python。

建立连接

quit.pyw

```
1 import sys
2 from PyQt4 import QtCore, QtGui
3
4 app = QtGui.QApplication(sys.argv)
5
6 quit = QtGui.QPushButton("Quit")
7
8 QtCore.QObject.connect(quit, QtCore.SIGNAL("clicked()"),
9                          app,
```

```
QtCore.SLOT("quit()"))
10
11 quit.show()
12 sys.exit(app.exec_())
```

这里展示了Qt的信槽模式，有点怪异的是，在Python中，信号，槽，都是用字符串来表示-_-!这点似乎有点奇怪。

我还不`懂Qt`的原理，也没有看过Qt的源代码，但是总是感觉这里奇怪，于是我翻看了一下Qt的SIGNAL，SLOT宏，于是一切也就没有那么奇怪了。

```
# define SLOT(a) qFlagLocation("1"#a QLOCATION)

# define SIGNAL(a) qFlagLocation("2"#a QLOCATION)
```

本来，他们就是以字符串来表示的。。。。。。我原本还以为在Qt中这些都是通过回调函数的形式出现的呢。。。。。。汗-_-!真那样，与一般的WxWidget原理有啥不同啊。。呵呵，看来，Qt的原理还得好好理解解。其实就现在的信息看，无非也就是Observer模式的一种扩展，再难也难不到哪儿去。

在PyQt的安装包中，有个tutorial，展示了更复杂一点的button使用方法，可以参考参考

morequit.pyw

```
1 import sys
2 from PyQt4 import QtCore, QtGui
3
4
5 app = QtGui.QApplication(sys.argv)
6
7 quit = QtGui.QPushButton("Quit")
8 quit.resize(75, 30)
9 quit.setFont(QtGui.QFont("Times", 18, QtGui.QFont.Bold))
10
11 QtCore.QObject.connect(quit, QtCore.SIGNAL("clicked()"),
12                         app, QtCore.SLOT("quit()"))
13
14 quit.show()
15 sys.exit(app.exec_())
```

分别是设置Button的大小和按钮文字的字体和样式，这个阶段就不深抠细节了，了解概念和Qt大概的风格就好。

窗口部件的布局

用过MFC的人都知道在MFC中创建一个随窗口动态改变控件大小的程序的困难。。。。。。需要每次

OnMove.OnSize的时候去重新计算控件的大小，为了保证布局合理，最好还得将所有控件的位置用百分比来计算，（要么就是更老的版本）。但是在Qt中好像就要简单的比

较多。但有小例：

layout.pyw

```
1 import sys
2 from PyQt4 import QtCore, QtGui
3
4 app = QtGui.QApplication(sys.argv)
5
```

来源： pyqt

```
6 window = QtGui.QWidget()
7
8 spinBox = QtGui.QSpinBox()
9 slider = QtGui.QSlider(QtCore.Qt.Horizontal)
10 spinBox.setRange(0, 130)
11 slider.setRange(0, 130)
12
13 QtCore.QObject.connect(spinBox, QtCore.SIGNAL("valueChanged(int)"),
14                         slider,
15                         QtCore.SLOT("setValue(int)"))
16 QtCore.QObject.connect(slider, QtCore.SIGNAL("valueChanged(int)"),
17                         spinBox,
18                         QtCore.SLOT("setValue(int)"))
19 spinBox.setValue(35)
20
21 layout = QtGui.QHBoxLayout()
22 layout.addWidget(spinBox)
23 layout.addWidget(slider)
24 window.setLayout(layout)
25
26 window.show()
27 sys.exit(app.exec_())
```

行了，基本上PyQt的程序是怎么样子的，大概有个了解了：），下面就开始慢慢来了。我也需要睡觉去了。

write by 九天雁翎(JTianLing) -- blog.csdn.net/vagrxie

上一篇 [多鼠标的资料少到英国人用Google翻译我的文章看-_-!](#)
下一篇 [PyQt\(2\) 对话框](#)

主题推荐 [pyqt](#) [正则表达式](#) [helloworld](#) [源代码](#) [跨平台](#)

猜你在找

- | | |
|---------------------------------|-----------------------------|
| Qt Designer 布局 1 PyQt学习基础 | 【精品课程】备战2015信息系统项目管理师-案例梳理与 |
| PyQt学习1信号与槽道 | 【精品课程】华为IE讲师：直通华为HCNA课程实战第一 |
| PyQt挖地雷游戏学习笔记1 | 【精品课程】通过经典案例教你学Javascript |
| Cocos2d-x中应用C++11正则表达式检验邮箱地址跨平台 | 【精品课程】2015软考网络工程师—基础知识视频教程 |
| 来自于PyQt的HelloWorld | 【精品课程】Android入门实战教程 |

准备好了么？跳 吧 ！ 更多职位尽在 **CSDN JOB**

西安 元智系统招聘 Rails 工程师 1 名	我要跳槽	咪咕音乐互联网营销策划1(音乐企宣方向)	我要跳槽
西安元智系统技术有限责任公司	4-6K/月	陕西英豪人力资源有限公司	8-16K/月
武汉1w+招聘资深 .NET 软件工程师	我要跳槽	1. .Net性能优化专家	我要跳槽
中国惠普	10-15K/月	携程旅行网	18-30K/月

查看评论

8楼 [zqiang3](#) 2014-02-04 12:24发表



边练边学，很实用

7楼 [gcyxf](#) 2014-01-02 00:53发表



学习了~！

6楼 [阿凯99](#) 2012-09-20 08:43发表



好文章，对初学的我帮助很大。

5楼 [tranq](#) 2011-04-27 23:47发表



谢谢，正准备在手机上面写点程序玩呢

4楼 [匿名用户](#) 2010-01-24 14:24发表



[e01]非常好，虽然是入门，但是很详细

3楼 [匿名用户](#) 2009-12-21 19:17发表



谢谢，学习pyQT中

2楼 [idotop](#) 2009-10-15 22:15发表



正在学习中.....

1楼 [blackhailang](#) 2009-10-08 16:58发表



谢谢，学习了，现在正在研究python

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

- 全部主题
- Hadoop
- AWS
- 移动游戏
- Java
- Android
- iOS
- Swift
- 智能硬件
- Docker
- OpenStack
- VPN
- Spark
- ERP
- IE10
- Eclipse
- CRM
- JavaScript
- 数据库
- Ubuntu
- NFC
- WAP
- jQuery
- BI
- HTML5
- Spring
- Apache
- .NET
- API
- HTML
- SDK
- IIS
- Fedora
- XML
- LBS
- Unity
- Splashtop
- UML
- components
- Windows Mobile
- Rails
- QEMU
- KDE
- Cassandra
- CloudStack
- FTC
- coremail
- OPhone
- CouchBase
- 云计算
- iOS6
- Rackspace
- Web App
- SpringSide
- Maemo
- Compuware
- 大数据
- aptech
- Perl
- Tornado
- Ruby
- Hibernate
- ThinkPHP
- HBase
- Pure
- Solr
- Angular
- Cloud Foundry
- Redis
- Scala
- Django
- Bootstrap

公司简介 | 招贤纳士 | 广告服务 | 银行汇款帐号 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-600-2320 | 北京创新乐知信息技术有限公司 版权所有 | 江苏乐知网络技术有限公司 提供商务支持
京 ICP 证 070598 号 | Copyright © 1999-2014, CSDN.NET, All Rights Reserved