

原创

QStandardItemModel角色控制及QTreeView添加不同的右键菜单

2014-05-19 11:36:43 尘中远 阅读量 12252 更多

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。
本文链接：<https://blog.csdn.net/czyt1988/article/details/26018513>

1.概述

QTreeView最长用的一个功能就是作为导航栏，像vs里的项目结构树，word的文档结构图，资源管理器的文档结构，等等都是利用树形结构组织的，在型QStandardItemModel对树形控件节点的操作。但有时候，光有节点显示还是不够的，还需要和用户进行交互，如右键点击不同条目会出现不同菜单能。

在MFC里，树形控件CTreeCtrl是通过SetItemData函数来对节点设置一个指针的值，这个值可以是个指针或者DWORD值，可以设置一个自定义的标志TreeView比MFC的CTreeCtrl封装的更好，其功能更为强大，以至于它可以给每个节点设定非常非常多的值（只要你内存足够）。

QTreeView只负责显示渲染，数据都是Model来负责管理，Model和Item构成整个结构，具体可见第一篇。

在有了节点后我想对某些进行标记，需要用到setData函数。Qt中的mvc结构非常复杂，setData函数在model和item中都有，功能都一样，就QStandardItemModel来说，

义为：

```
virtual bool setData(const QModelIndex & index, const QVariant & value, int role = Qt::EditRole);
```

在QStandardItem中setData定义为

```
virtual void setData(const QVariant & value, int role = Qt::UserRole + 1);
```

通常使用的是QStandardItem的版本。

下面详细说说这个函数。

2.使用Role对QStandardItem设定值

setData函数就是给这个item设置一个QVariant的值，但是，这个函数有两个参数，第一个QVariant自然是需要设置的值，另一个是一个int型数据，Qt中把这个称为role角色，指定进入item的这个QVariant所扮演的角色，实际就是对设定值的标定，因为item可以设置许多值，这就需要用一个用以区分的标志，这个区分标志就叫角色。

Qt已经把经常用到的角色内容定义好了，我们可以自己定义角色标志，但不能和定义好的那些冲突，否则会不起作用。如要显示文字用Qt::DisplayRole，要告诉QTreeView颜色，就标定Qt::BackgroundRole，要改变字体就标定Qt::FontRole等等。从中可以看出，role就是一个标示，用来标定存放在item里面的值具体用于什么功能，系统默认的role就是Qt::ItemDataRole枚举。

Qt默认role其实就是一组宏（原理和MFC的消息类型一模一样），那怎么知道能不和Qt预先定义好的不起冲突呢，同MFC自定义消息一样，Qt要自定义角色，就从Qt::UserRole开始，我们做个类似于vs或Qt Creator的项目结构树，结构树里许多节点的功能不一样有节点代表文件夹，有的节点是cpp文件，有的节点是.h文件，如果要知道用户点击的当前节点内容，就需要给节点一些额外的标志。

下面用个例子来演示。

此例子用来复现Qt Creator的项目结构树

项目结构树节点主要有“根节点”，“文件夹节点”，“条目节点”这三种，于是可以定义一个role来对这三种情况做判别。

另外，“文件夹节点”又分为“cpp文件夹”，“h文件夹”等节点，这个也可以使用一个role作为区分。

“条目节点”有可能也会有h文件，cpp文件，或者其他文件之分，这里也可以使用一个role进行区分。

于是，这里就建立三个自定义role

```
#define ROLE_MARK Qt::UserRole + 1
#define ROLE_MARK_FOLDER Qt::UserRole + 2
#define ROLE_MARK_ITEM Qt::UserRole + 3
```

这三个role中，ROLE_MARK用于区分 “根节点”，“文件夹节点”，“条目节点” 这三种情况。

ROLE_MARK_FOLDER用于区分 “cpp文件夹”，“h文件夹” 等情况。

ROLE_MARK_ITEM用于区分 “条目节点” 有可能出现的种类。

当然，对于这种比较少的区分，用一个int型变量进行位位的或与操作来判断也是可以的，但这里主要为了演示role的使用方法。

下面在为三种role定义值，定义如下

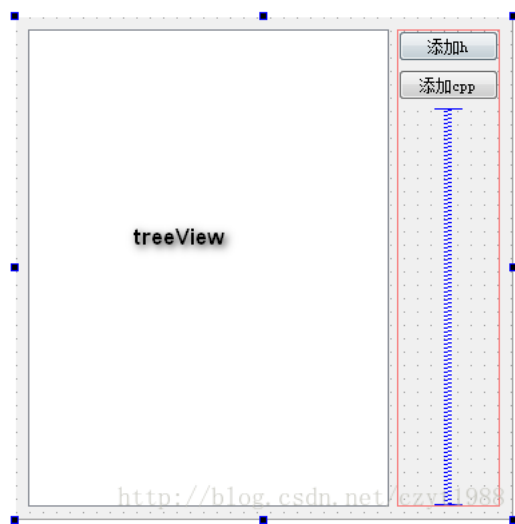
```
//对应ROLE_MARK
#define MARK_PROJECT 1 //这是总项目标记
#define MARK_FOLDER 2 //这是文件夹标记
#define MARK_ITEM 3 //条目标记

//对应ROLE_MARK_FOLDER, 标记folder种类
#define MARK_FOLDER_H 1 //头文件文件夹标记
#define MARK_FOLDER_CPP 2 //cpp文件文件夹标记

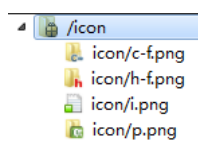
//对应ROLE_MARK_ITEM标记item种类
#define MARK_ITEM_H 1 //头文件条目
#define MARK_ITEM_CPP 2 //cpp文件条目
```

<https://blog.csdn.net/czyt1988/article/details/26018513>

1/16



加上图标文件



树形视图的初始化:

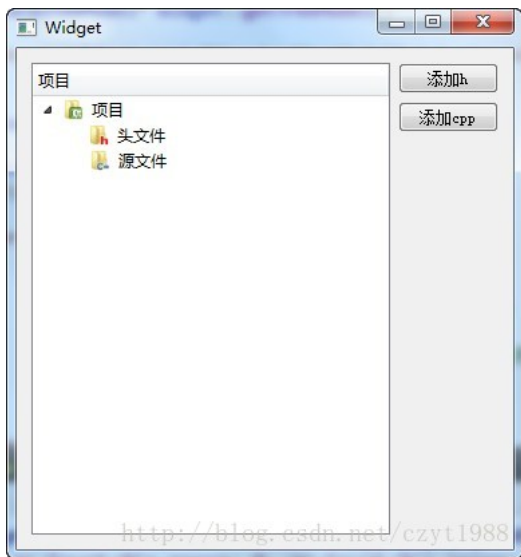
```

1 void Widget::init()
2 {
3     QStandardItemModel* model = new QStandardItemModel(ui->treeView);
4     model->setHorizontalHeaderLabels(QStringList()<<QStringLiteral("项目"));
5     //添加项目文件夹
6     QStandardItem* root = new QStandardItem(QIcon(":/icon/icon/p.png"),QStringLiteral("项目"));
7     root->setData(MARK_PROJECT,ROLE_MARK);//首先是项目中目录
8     root->setData(MARK_FOLDER,ROLE_MARK_FOLDER);//其次它属于文件夹
9     model->appendRow(root);
10    QStandardItem* folder = new QStandardItem(QIcon(":/icon/icon/h-f.png"),QStringLiteral("头文件"));
11    folder->setData(MARK_FOLDER,ROLE_MARK);//首先是文件夹
12    folder->setData(MARK_FOLDER_H,ROLE_MARK_FOLDER);//其次它属于头文件文件夹
13    root->appendRow(folder);
14    folder = new QStandardItem(QIcon(":/icon/icon/c-f.png"),QStringLiteral("源文件"));
15    folder->setData(MARK_FOLDER,ROLE_MARK);//首先是文件夹
16    folder->setData(MARK_FOLDER_CPP,ROLE_MARK_FOLDER);//其次它属于源文件文件夹
17    root->appendRow(folder);
18    ui->treeView->setModel(model);
19 }

```

在添加条目时，给一些特别的条目设定标志，如头文件文件夹，因为它是文件夹，因此首先给它设定角色为ROLE_MARK的值MARK_FOLDER，其次它在文件夹中属于头文件给他设定角色为ROLE_MARK_FOLDER的值MARK_FOLDER_H。这时，这个条目就有两个额外的值用于特殊的判断。

上面代码运行界面如下图所示：



当添加头文件时，需要在头文件文件夹加入条目，按钮“添加h”就是用来模拟添加头文件的过程。

添加头文件时，首先需要找到“头文件”这个文件夹对应的QStandardItem*，实现如函数getHeaderFolder，此函数实现需要先找到“项目”这个顶层文件夹，具体实现如下：

```

1  QStandardItemModel* Widget::getTreeModel()
2  {
3      return qobject_cast<QStandardItemModel*>(ui->treeView->model());
4  }
5
6  QList<QStandardItem*> Widget::getRoots()
7  {
8      QList<QStandardItem*> roots;
9      QStandardItemModel* model = getTreeModel();
10     for(int i=0; i < model->rowCount(); ++i)
11     {
12         roots.append(model->item(i));
13     }
14     return roots;
15 }
16
17 QStandardItem* Widget::getProjectFolder()
18 {
19     QList<QStandardItem*> roots = getRoots();
20     for(auto i=roots.begin(); i!=roots.end(); ++i){
21         if((*i)->data(ROLE_MARK) == MARK_PROJECT){
22             return (*i);
23         }
24     }
25     return nullptr;
26 }
27
28 QStandardItem* Widget::getHeaderFolder()
29 {
30     QStandardItem* project = getProjectFolder();
31     if(nullptr == project)
32         return nullptr;
33     for(int i=0; i < project->rowCount(); ++i)
34     {
35         QStandardItem* child = project->child(i);
36         QVariant var = child->data(ROLE_MARK_FOLDER);
37         if(!var.isValid())
38             continue; //说明不是ROLE_MARK_FOLDER，有可能是一些项目，对应项目结构树那个xxx.pro就是一个非文件夹条目
39         if(MARK_FOLDER_H == var.value<int>())
40             return child;
41     }
42     return nullptr;
43 }
44
45 QStandardItem* Widget::getSrcFolder()
46 {
47     QStandardItem* project = getProjectFolder();

```

```
48 | if(nullptr == project)49 | return nullptr;
50 | for(int i=0;i < project->rowCount();++i)
51 | {
52 |     QStandardItem* child = project->child(i);
53 |     QVariant var = child->data(ROLE_MARK_FOLDER);
54 |     if(!var.isValid())
55 |         continue;//说明不是ROLE_MARK_FOLDER, 有可能是一些项目, 对应项目结构树那个xxx.pro就是一个非文件夹条目
56 |     if(MARK_FOLDER_CPP == var.value<int>())
57 |         return child;
58 | }
59 | return nullptr;
60 | }
```

getTreeModel用于获取treeView的model;
getRoots用于获取所有根节点;
getProjectFolder用于获取“项目文件夹”;
getHeaderFolder用于获取“头文件文件夹”;
getSrcFolder用于获取“源文件文件夹”;
主要使用了data函数, 函数声明如下:

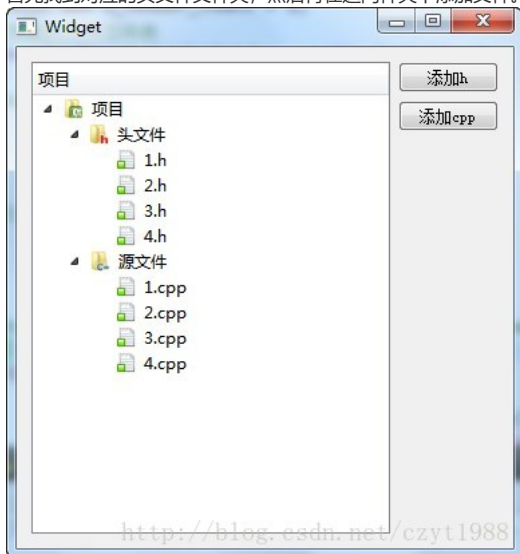
```
QVariant QStandardItem::data(int role = Qt::UserRole + 1) const
```

它会根据role, 返回对应的QVariant, 如果没有这个role, 返回的QVariant会是不可用, 可以通过QVariant的函数isValid进行判断。
QVariant函数内部的值需要先转换, 转换可以使用toInt函数或者使用一个通用的模板函数value。

下面看看按钮“添加头文件”的实现:

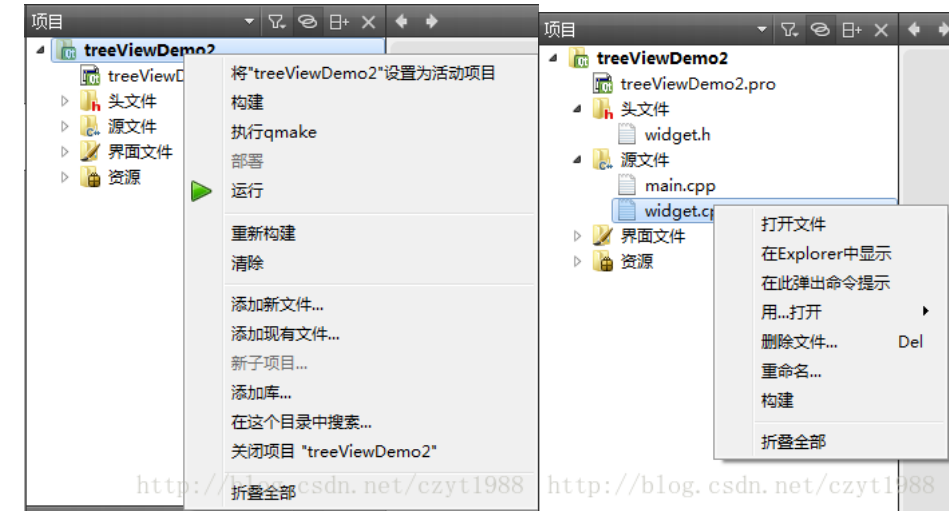
```
1 | void Widget::on_pushButton_clicked()
2 | {
3 |     static int s_header_count = 1;
4 |     //找到头文件文件夹
5 |     QStandardItem* headerFolder = getHeaderFolder();
6 |     if(headerFolder)
7 |     {
8 |         QStandardItem* item = new QStandardItem(QIcon(":/icon/icon/i.png")
9 |             ,QStringLiteral("%1.h").arg(s_header_count));
10 |         item->setData(MARK_ITEM,ROLE_MARK);//首先标定条目的类型 - 文件夹、项目、条目...
11 |         item->setData(MARK_ITEM_H,ROLE_MARK_ITEM);//再次标定项目的类型
12 |         headerFolder->appendRow(item);
13 |         ++s_header_count;
14 |     }
15 | }
```

首先找到对应的头文件文件夹, 然后再在这问文件夹下添加文件。运行效果如下图:

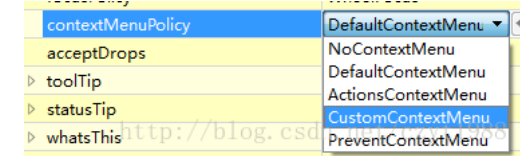


3.给树形视图设置右键菜单

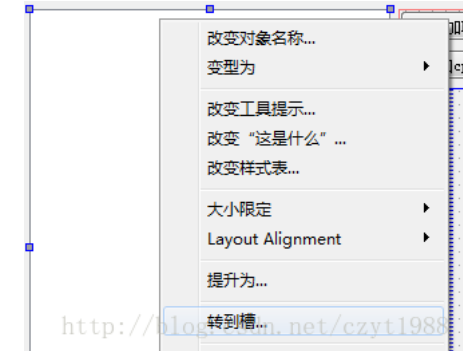
树形视图最大的优点是清晰的逻辑关系, 可以明显的看出每个条目的父子关系, 对于大型工程来说显得尤为重要。由于条目之间功能不同, 对条目的操作也会有不同的响应
Creator的项目结构树, 对顶层跟项目点右键和对其它节点右键是弹出不同的菜单的, 如下图所示



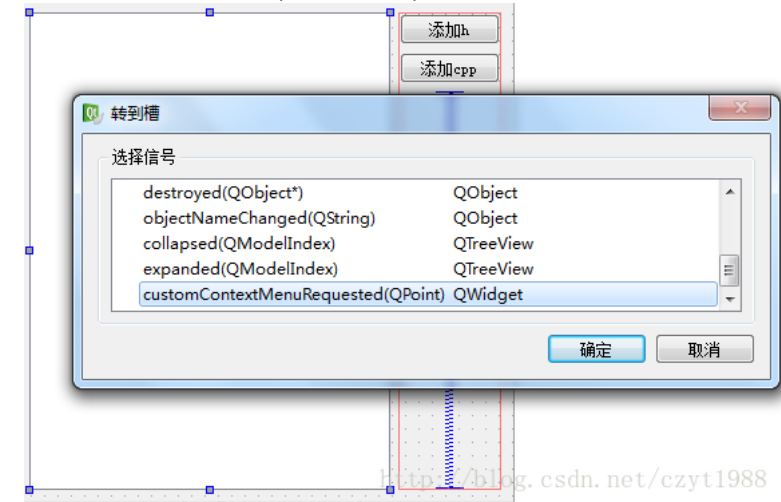
这里涉及到两个方面，一个是给QTreeView添加菜单，另一个是对右击的节点进行判断。
给QtreeView添加右键菜单，首先需要把contextMenuPolicy属性设置为：CustomContextMenu。



在ui 编辑器中右击QTreeView，选择转到槽



选择customContextMenuRequested(QPointpos)信号



这时，会自动添加对应的槽函数：

```
1 void Widget::on_treeView_customContextMenuRequested(const QPoint &pos)
2 {
3
4 }
```

当然也可以使用代码添加！
此槽函数接收一个点坐标，用于标定点击的方位，可以使用indexAt函数来获取具体点击的条目。

```
virtual QModelIndex indexAt(const QPoint & point) const
在添加Menu前要先创建menu。声明两个menu的成员变量，记得添加对应的头文件
#include <QMenu>
#include <QAction>
```

```
QMenu* m_projectMenu;
QMenu* m_itemMenu;
```

在构造函数中创建menu

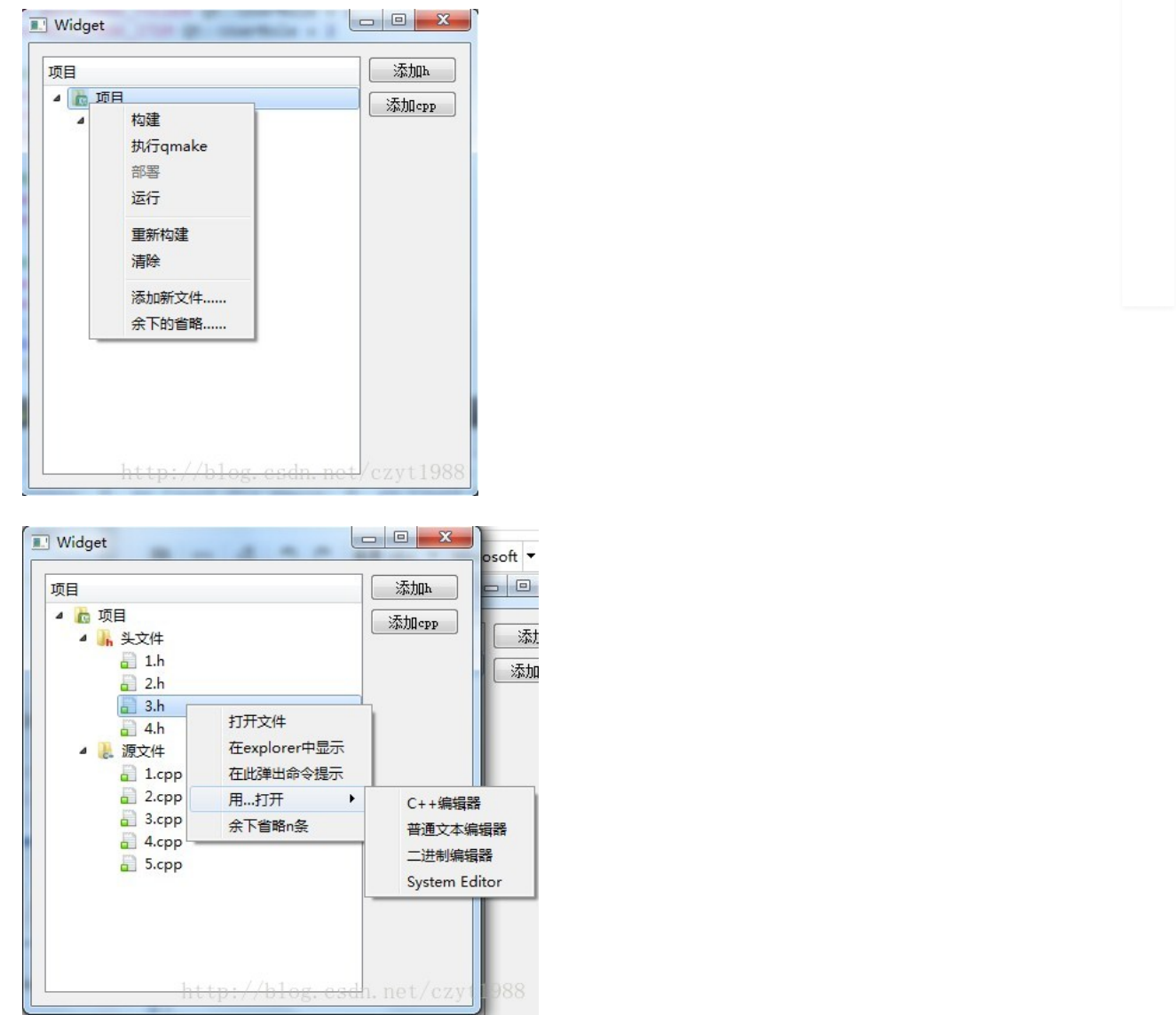
```
1      m_projectMenu = new QMenu(this);
2      m_itemMenu = new QMenu(this);
3
4      QAction* ac = nullptr;
5      ac = new QAction(QStringLiteral("构建"),this);
6      m_projectMenu->addAction(ac);
7
8      ac = new QAction(QStringLiteral("执行qmake"),this);
9      m_projectMenu->addAction(ac);
10
11     ac = new QAction(QStringLiteral("部署"),this);
12     ac->setEnabled(false);
13     m_projectMenu->addAction(ac);
14
15     ac = new QAction(QStringLiteral("运行"),this);
16     m_projectMenu->addAction(ac);
17
18     m_projectMenu->addSeparator();
19
20     ac = new QAction(QStringLiteral("重新构建"),this);
21     m_projectMenu->addAction(ac);
22
23     ac = new QAction(QStringLiteral("清除"),this);
24     m_projectMenu->addAction(ac);
25
26     m_projectMenu->addSeparator();
27
28     ac = new QAction(QStringLiteral("添加新文件....."),this);
29     m_projectMenu->addAction(ac);
30
31     ac = new QAction(QStringLiteral("余下的省略....."),this);
32     m_projectMenu->addAction(ac);
33
34     //
35
36     ac = new QAction(QStringLiteral("打开文件"),this);
37     m_itemMenu->addAction(ac);
38     ac = new QAction(QStringLiteral("在explorer中显示"),this);
39     m_itemMenu->addAction(ac);
40     ac = new QAction(QStringLiteral("在此弹出命令提示"),this);
41     m_itemMenu->addAction(ac);
42
43     QMenu* itemChildMenu = new QMenu(m_itemMenu);
44     itemChildMenu->setTitle(QStringLiteral("用...打开"));
45     ac = new QAction(QStringLiteral("C++编辑器"),this);
46     itemChildMenu->addAction(ac);
47     ac = new QAction(QStringLiteral("普通文本编辑器"),this);
48     itemChildMenu->addAction(ac);
49     ac = new QAction(QStringLiteral("二进制编辑器"),this);
50     itemChildMenu->addAction(ac);
51     ac = new QAction(QStringLiteral("System Editor"),this);
52     itemChildMenu->addAction(ac);
53
54     m_itemMenu->addAction(itemChildMenu->menuAction());
55
56     ac = new QAction(QStringLiteral("余下省略n条"),this);
57     m_itemMenu->addAction(ac);
```

on_treeView_customContextMenuRequested槽函数具体实现代码如下:

```
1 void Widget::on_treeView_customContextMenuRequested(const QPoint &pos)
2 {
3     QModelIndex index = ui->treeView->indexAt(pos);
4     QVariant var = index.data(ROLE_MARK);
5     if(var.isValid())
```

```
6 | {
7 |     if(MARK_PROJECT == var.toInt())
8 |         m_projectMenu->exec(QCursor::pos()); //弹出右键菜单，菜单位置为光标位置
9 |     else if(MARK_ITEM == var.toInt())
10 |         m_itemMenu->exec(QCursor::pos());
11 | }
12 | }
```

首先用indexAt获取当前点击条目的QModelIndex。通过QModelIndex获取条目的data，QTreeView的数据可以通过model，item，index三者任意一个获取，非常方便。之前对项目 and 条目通过ROLE_MARK角色做过标记，只要判断ROLE_MARK角色，就可以区分点击的是根节点项目还是任意一个条目。具体效果见下图



4.系统role的使用

在设定data时，需要制定角色，而自定义角色都是从Qt::UserRole开始往上延伸的，那么Qt::UserRole之前的那些内容是什么呢。Qt为我们定义了一些常用的角色，在Qt::ItemData中。在Qt说明文档中有详细说明(比较懒不想copy，截了一个图)

The general purpose roles (and the associated types) are:

Constant	Value	Description
Qt::DisplayRole	0	The key data to be rendered in the form of text. (QString)
Qt::DecorationRole	1	The data to be rendered as a decoration in the form of an icon. (QColor, QIcon or QPixmap)
Qt::EditRole	2	The data in a form suitable for editing in an editor. (QString)
Qt::ToolTipRole	3	The data displayed in the item's tooltip. (QString)
Qt::StatusTipRole	4	The data displayed in the status bar. (QString)
Qt::WhatsThisRole	5	The data displayed for the item in "What's This?" mode. (QString)
Qt::SizeHintRole	13	The size hint for the item that will be supplied to views. (QSize)

Roles describing appearance and meta data (with associated types):

Constant	Value	Description
Qt::FontRole	6	The font used for items rendered with the default delegate. (QFont)
Qt::TextAlignmentRole	7	The alignment of the text for items rendered with the default delegate. (Qt::AlignmentFlag)
Qt::BackgroundRole	8	The background brush used for items rendered with the default delegate. (QBrush)
Qt::BackgroundColorRole	8	This role is obsolete. Use BackgroundRole instead.
Qt::ForegroundColorRole	9	The foreground brush (text color, typically) used for items rendered with the default delegate. (QBrush)
Qt::TextColorRole	9	This role is obsolete. Use ForegroundColorRole instead.
Qt::CheckStateRole	10	This role is used to obtain the checked state of an item. (Qt::CheckState)
Qt::InitialSortOrderRole	14	This role is used to obtain the initial sort order of a header view section. (Qt::SortOrder). This role was introduced in Qt 4.8.

Accessibility roles (with associated types):

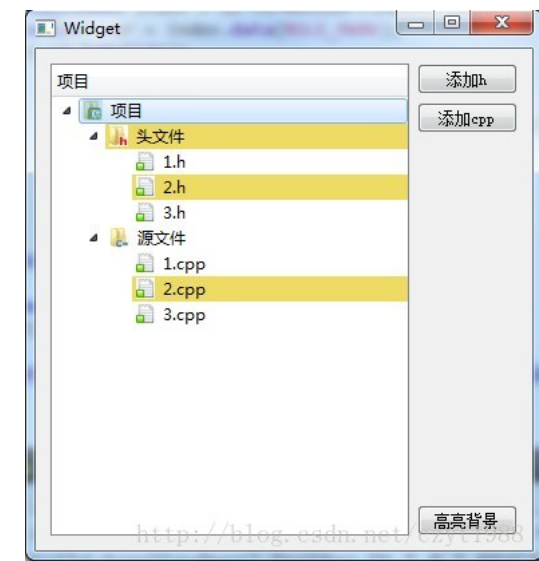
Constant	Value	Description
Qt::AccessibleTextRole	11	The text to be used by accessibility extensions and plugins, such as screen readers. (QString)
Qt::AccessibleDescriptionRole	12	A description of the item for accessibility purposes. (QString)

我们最常用的就是Qt::DisplayRole，也许你用QStandardItemModel从来都不会在代码中用到它，但是，在显示文字过程中，都会调用此role的值。只要给这些系统role复制，在视图上就会有对应的效果。

如Qt::BackgroundRole用于设置背景色，只要调用setData时把role设置为Qt::BackgroundRole，同时传入的值为一个颜色值，那么它就会在设置背景颜色。

```
1 void Widget::on_pushButton_3_clicked()
2 {
3     QModelIndex index = ui->treeView->currentIndex();
4     if(!index.isValid())
5         return;
6     getTreeModel()->itemFromIndex(index)->setData(QColor(232,209,57,200),Qt::BackgrounRole);
7 }
```

效果如下：



Qt::ToolTipRole用于给条目添加额外的说明

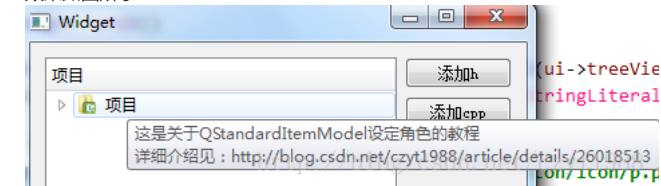
在根节点加入一个额外说明如下：

```
1 void Widget::init()
2 {
3     QStandardItemModel* model = new QStandardItemModel(ui->treeView);
4     model->setHorizontalHeaderLabels(QStringList()<<QStringLiteral("项目"));
```



```
5 |
6 | //添加项目文件夹
7 | QStandardItem* root = new QStandardItem(QIcon(":/icon/icon/p.png"),QStringLiteral("项目"));
8 | root->setData(MARK_PROJECT,ROLE_MARK); //首先它是项目中目录
9 | root->setData(MARK_FOLDER,ROLE_MARK_FOLDER); //其次它属于文件夹
10 | root->setData(
11 |     QStringLiteral("这是关于QStandardItemModel设定角色的教程\n详细介绍见: http://blog.csdn.net/czyt1988/article/details/26018513"),Qt::ToolTipRole
12 | );
13 |
14 | .....
15 | }
```

效果如图所示:



Qt::TextColorRole用于改变文字颜色， Qt::TextAlignmentRole改变对齐方式， Qt::FontRole控制字体等等， 这里不——介绍。

ps:
高亮背景后需要把高亮取消，就需要遍历所有子节点，并把设置有Qt::BackgroundRole角色的data设置为QVariant();具体遍历见[http://blog.csdn.net/czyt1988/article/de](http://blog.csdn.net/czyt1988/article/details/26018513)使用了C++11的一些新特性。

```
1 | void Widget::on_pushButton_4_clicked()
2 | {
3 |     //涉及到遍历，因此使用回调函数，把遍历需要执行的函数传给封装好的遍历
4 |     StandardItemModelEx::ergodicAllItem(getTreeModel()
5 |         ,std::bind(&Widget::callback_clearColor,this,std::placeholders::_1));
6 | }
7 |
8 | void Widget::callback_clearColor(QStandardItem* item)
9 | {
10 |     item->setData(QVariant(),Qt::BackgroundRole);
11 | }
```


demo代码: <http://download.csdn.net/detail/czyt1988/7368399>

书籍推广

C++ GUI Qt 4编程（第二版）	Qt中的C++技术	Qt5开发及实例	ARM9嵌入式系统设计直通车

文章最后发布于: 2

有 0 个人打赏



一体化工控机嵌入式工控机

嵌入式工控机

2.1万阅读



想对作者说点什么



虫探寒封 2年前 #2楼

很棒很有用



金十 5年前 #1楼

等了许久的QTreeView终于新鲜出炉！写的很详细！要西！

QTreeView使用总结7，右键菜单

阅读数 2525

1, 简介最开始我要做右键菜单时, 理所当然的想到的是右键单击的信号, 这样是可以的。后来发现原来Qt给QWidg... 博文 来自: 逆风微光

Qt使用的QTreeView对象实现树状链表简单显示。

阅读数 6476

简言: 本人并不专研Qt, 只是写小项目时遇到问题和为了解决这个某个问题, 特意学习了QTreeView的知识点。本... 博文 来自: fangjiaze444的博客

Qt编程27: QStandardItemModel与QTreeView的使用 (显示目录下文件)

阅读数 6034

使用QTreeView只是做为显示数据使用#ifndef MIRALLTREEVIEW_H#define MIRALLTREEVIEW_H#include<includ... 博文 来自: jhe_zhang-blog

QTreeView使用总结13，自定义model示例，大大优化性能和内存

阅读数 3563

1, 简介前面简单介绍过Qt的模型/视图框架, 提到了Qt预定义的几个model类型: QStringListModel: 存储简单的... 博文 来自: 逆风微光



分体式电磁流量计 一体式电磁流量计

电磁流量计
2.3万阅读

Qt树形控件QTreeView使用

阅读数 1万+

原文地址QTreeView和QStandardItemModel的使用QtreeView是ui中最常用的控件, Qt中QTreeWidget比QTree... 博文 来自: wukai_std的博客

PyQt4 QTreeView的简单树

阅读数 1386

简单的树结构, 目前的残留问题是无法自动调整QTreeView的宽度1、新建文件夹, 任意英文名字, 比如Main创建py... 博文 来自: yangyang022的博客

如何利用 QStandardItem 存放任意数据类型?

一般我们使用 QStandardItem 和 QStandardItemModel 时, 会这样做 QStandardItemModel model = new QStandar...

论坛

Qt之界面数据存储与获取

阅读数 184

简述在GUI开发中, 往往需要在界面中存储一些有用的数据, 这些数据可以来自配置文件、注册表、数据库、或者是S... 博文 来自: weixin_33708432...

Qt中使用model->setData () 的方法如何设置指定单元格的背景颜色?

-

问答

QStandardItemModel角色控制及QTreeView添加不同的右键..._CSDN博客

QStandardItemModel角色控制及QTreeView加入不同的右键..._CSDN博客

AI实训路线推荐：考虑转型/跳槽程序员免费领取，自测专业程度！

如何能够短时间内抓住技术重点，打造属于自己的“offer收割机”？

setData不起作用

主页面引用组件(index.wxml) <uploadImages bind:myevent="imagesEvent" wx-if="{{pageType == 'add'}}"></uplo...

论坛

QStandardItemModel角色控制及QTreeView添加不同的右键..._CSDN博客

Qt树形控件QTreeView使用1——节点的添加删除操作 - 尘..._CSDN博客

QStandardItem和QListWidgetItem的setData方法

阅读数 3568

在做界面时经常用到树形显示和列表显示。1、树形结构 使用控件QTreeView, 关联一个数据Model, 给Model添... 博文 来自: Lance



逆风微光
155篇文章
排名:6000+
[关注](#)



fangjiaze444
16篇文章
排名:千里之外
[关注](#)



jhe_zhang
100篇文章
排名:千里之外
[关注](#)



AlwaysOnTheWay
122篇文章
排名:千里之外
[关注](#)

QTreeView和Qwidget右键菜单的弹出 - 王文松的技术工坊 - CSDN博客	
QTreeView使用总结7,右键菜单 - 逆风微光 - CSDN博客	
Qt树形控件QTreeView使用1——节点的添加删除操作 复选框的设置	阅读量 3万+
QtreeView是ui中最常用的控件，Qt中QTreeWidget比QTreeView更简单，但没有QTreeView那么灵活（QTreeWi... 博文 来自: huyisu的专栏	
QT 之 QStandardItemModel	阅读量 3778
该QStandardItemModel类提供用于存储自定义数据的通用模型 博文 来自: 默默的努力	
Qt 之 QTreeView添加右键菜单 - kevin_org - CSDN博客	
QTreeView使用总结13,自定义model示例,大大优化性能和..._CSDN博客	
QStandItem::setData()函数用法（解决设置了QIcon后，setData函数无效问题）(Qt)	阅读量 117
博主用QStandItem::setData()函数后，发现设置了QStandItem::setIcon()函数后，用setData()函数无效。实际上，... 博文 来自: 净无邪博客	
羡慕AI高薪岗！为什么这类程序员不建议大家转型？	
被众多开发工程师羡慕的AI程序员为啥这么高薪！30w只是白菜价有啥要求？	
QTreeView之setData绑定一个对话框 - toby54king的博客 - CSDN博客	
QTreeView（QStandardItemModel）中如何在每列添加按钮和复选框等	
QStandardItemModel model = new QStandardItemModel(4,1); QStandardItem *item1 = new QStandardItem("f... 论坛	
怎么在QStandardItemModel里面添加对应表头 的xml文档数据	
如题 Qt..... 论坛	
QStandardItemModel插入数据的问题	
我使用了QStandardItemModel存储数据，格式如下 c1 c2 c3 2 3 6 然后监听了dataChanged信号，当c2列的值发生变化... 论坛	
PyQt4如何更新treeView里model的数据内容	
1、我自定义了一个QAbstractItemModel 2、使用QTreeView去加载这个model 3、视图内已经加载好model内的数据，... 论坛	
QTreeView使用总结11，数据过滤，使用代理model，简单过滤	阅读量 1878
1，简介有时候需要对tree内的数据按一些条件进行过滤显示，比如按搜索条件过滤，这时候就用到了代理model。... 博文 来自: 逆风微光	
那些拿到 60K Offer 的 AI 程序员，后来都怎么样了？	
刚刚拿到阿里offer，工作地点杭州。值得去吗？	
QTreeView和Qwidget右键菜单的弹出	阅读量 714
需要在QTreeView和Qwidget弹出右键菜单，并在指定位置响应它。Qwidget连接的是Matplotlib，因此需要注意的... 博文 来自: 王文松的技术工坊	
model中 setData () 函数 flag () 函数作用	阅读量 5663
如果允许用户编辑view中的item，那么必须重写model中的setData()函数和flag () 函数， //允许用户编辑item，... 博文 来自: 证券指标的专栏	
Qt 之 QTreeView添加右键菜单	阅读量 548
转自: https://blog.csdn.net/huapenguag/article/details/506697011 在目标View上设置右键策略//!支持右键菜... 博文 来自: kevin_org	
QStandardItemModel简介	阅读量 903
类QabstractItemModel，QabstractListModel，QabstractTableModel不保存数据，用户需要从这些类派生出子... 博文 来自: 生有涯，知无涯	

QStandardItemModel角色控制及QTreeView加入不同的右键菜单	阅读数 5
1.概述QTreeView最长用的一个功能就是作为导航栏，像vs里的项目结构树，word的文档结构图，资源管理器的文... 博文 来自： weixin_30519074	
从零开始深度学习102讲，送配套图书！ 从原理到实战，美女科学家带你入门深度学习	
QStandardItem如何修改列抬头的颜色	
如上图的语句没有效果	论坛
QStandardItem 和 Qtableview 动态更新的问题	
现在遇到这样一个问题，调用了QStandardItem 中的 setData后emit了dataChanged，tableview不能时时更新 1....	论坛
QTreeView使用总结4， Model/View框架介绍	阅读数 1746
1，简介Model/View架构分为三部分：模型、视图和委托。主要目的是将数据的存储与显示分离。Model模型：对... 博文 来自： 逆风微光	
关于QTableView添加QStandardItemModel，查找的问题	
QStandardItem *model; int row = model->rowCount(); model->insertRow(row,QModelIndex()); model->	论坛
QML中使用QAbstractListModel，setdata怎么传递参数QModelIndex	
初学QML，和C++混编遇到一些问题 一个Listview显示数据, 自定义一个QAbstractListModel，需要数据修改所以实现 1. ...	论坛
推动全社会公益氛围形成，使公益与空气和阳光一样触手可及。 公益缺你不可，众多公益项目等你PICK——百度公益 让公益像「空气和阳光」一样触手可及！ gongyi.baidu.com	
QStandardItem中setData()函数无效情况的说明	阅读数 3555
今天在使用QStandardItem时遇到了setData()函数执行无效的情况代码如下： QStandardItem*Item=newQStand... 博文 来自： seven407的专栏	
Qt QTreeView使用-QStandardItemModel的使用	05-04
代码里面有tree view的节点操作，包括添加，当前点击检查，遍历等。具体介绍见： Qt树形控件QTreeView使用1——节...	下载
如何清除QTreeView里面的model	
请问下怎么清除QTreeView里面的model，就像QComboBox的clear () 函数那样，一下把里面的东西全清空了	论坛
QTreeView树形视图使用自定义模型model	03-24
QTreeView树形视图使用自定义模型model，不是改造QT例子。	下载
QTreeView使用总结14，自定义model，控制对齐和颜色	阅读数 1880
1，简介前文介绍了使用自定义model的基本方法，重点强调了该方法带来的性能上的优点。其实自定义model还能... 博文 来自： 逆风微光	
一本可陪伴一辈子的笔记本 可擦可写，可循环利用，支持OCR识别，让你的笔记本变得智能	
实现右键菜单编辑功能treeview控件	03-13
试过用复合控件包含treeview控件，但是失败，这次在右键菜单调用tree的回发js成功可以在页面后台绑定被编辑的节点， ...	下载
QTreeView model/View	阅读数 581
QTreeView顾名思义是树形结构，分级目录。直接上代码吧。首先是Item的定义，item的定义有点像链表，记录父... 博文 来自： u011962883的专栏	
为QTreeView添加右键菜单	阅读数 2550
1在目标View上设置右键策略//!支持右键菜单 m_treeView->setContextMenuPolicy(Qt::CustomContextMenu);2... 博文 来自： uag 的博客	
QTreeView和QStandardItemModel的简单使用	阅读数 260
Dialog.h#ifndef DIALOG_H#define DIALOG_H#include<QtGui/QDialog>;namespace Ui{class Dialog;}cl... 博文 来自： 飞空静渡	
学会了这些技术，你离BAT大厂不远了	阅读数 13万+
每一个程序员都有一个梦想，梦想着能够进入阿里、腾讯、字节跳动、百度等一线互联网公司，由于身边的环境等原... 博文	

<div>程序员真是太太太太有趣了！！</div> <div>网络上虽然已经有了很多关于程序员的话题，但大部分人对这个群体还是很陌生。我们在谈论程序员的时候，究竟该...</div>	阅读量 4万+
<div>史上最详细的IDEA优雅整合Maven+SSM框架（详细思路+附带源码）</div> <div>网上很多整合SSM博客文章并不能让初探ssm的同学思路完全的清晰，可以试着关掉整合教程，摇两下头骨，哈一大...</div>	阅读量 2万+ 博文
<div>浏览 GitHub 太卡了？教你两招！</div> <div>老实说，GitHub 在国内的使用体验并不算太好，这其中最大的原因就是网络了。GitHub 访问起来比较卡，这个看...</div>	阅读量 2万+ 博文
<div>吃人的那些 Java 名词：对象、引用、堆、栈</div> <div>作为一个有着 8 年 Java 编程经验的 IT 老兵，说起来很惭愧，我被 Java 当中的四五个名词一直困扰着：**对象、引...</div>	阅读量 1万+ 博文
<div>Google 为中国开发者都带来了什么？</div> <div>Android 10 原生支持 5G，Flutter 1.9、Dart 2.5 正式发布！作者 唐小引写于上海世博中心 封面 CSDN 付费下...</div>	阅读量 2551 博文
<div>便携式 流量计 进口流量计 3211阅读</div>	
<div>我花了一夜用数据结构给女朋友写个H5走迷宫游戏</div> <div>起因 又到深夜了，我按照以往在csdn和公众号写着数据结构！这占用了我大量的时间！我的超越妹妹严重缺乏陪伴...</div>	阅读数 14万+ 博文
<div>接班马云的为何是张勇？</div> <div>上海人、职业经理人、CFO 背景，集齐马云三大不喜欢的张勇怎么就成了阿里接班人？作者 王琳 本文经授权转载自...</div>	阅读量 3万+ 博文
<div>让程序员崩溃的瞬间（非程序员勿入）</div> <div>今天给大家带来点快乐，程序员才能看懂。来源：https://zhuanlan.zhihu.com/p/47066521 1. 公司实习生找 Bug...</div>	阅读数 19万+ 博文
<div>Spring高级技术梳理</div> <div>Spring高级技术梳理 序言正文SpringDate部分Spring全家桶之SpringData——预科阶段Spring全家桶之SpringDat...</div>	阅读量 712 博文
<div>如何在Windows中开启"上帝模式"</div> <div>原文链接：https://mp.weixin.qq.com/s?__biz=MzlwMjE1MjMyMw==&mid=2650202982&idx=1&...</div>	阅读量 5119 博文
<div>便携式 流量计 进口流量计 3211阅读</div>	
<div>Docker 零基础从入门到使用</div> <div>诺！这只可爱的小鲸鱼就是docker了！ Docker 是什么？ Docker 是一个开源的应用容器引擎，让开发者可以打包他...</div>	阅读量 1万+ 博文
<div>分享靠写代码赚钱的一些门路</div> <div>作者 mezod，译者 josephchang10如今，通过自己的代码去赚钱变得越来越简单，不过对很多人来说依然还是很难...</div>	阅读量 4万+ 博文
<div>对计算机专业来说学历真的重要吗？</div> <div>我本科学校是渣渣二本，研究生学校是985，现在毕业五年，校招笔试、面试，社招面试参加了两年了，就我个人的...</div>	阅读量 6万+ 博文
<div>技术人员要拿百万年薪，必须要经历这9个段位</div> <div>很多人都问，技术人员如何成长，每个阶段又是怎样的，如何才能走出当前的迷茫，实现自我的突破。所以我结合我...</div>	阅读量 1万+ 博文
<div>多线程编程是后台开发人员的基本功</div> <div>这里先给大家分享一下小故事：在我刚开始参加工作的那年，公司安排我开发一款即时通讯软件（IM，类似于 QQ ...</div>	阅读量 6879 博文
<div>win10电脑工具整理 - 常用工具！</div> <div>如题，本文主要为博主对电脑上安装的一些软件，所做的整理，当做备份用吧。一、分类 系统工具 办公软件 编程开...</div>	阅读量 4081 博文
<div>动画：用动画给面试官解释 TCP 三次握手过程</div> <div>作者 小鹿 来源 公众号：小鹿动画学编程 写在前边 TCP 三次握手过程对于面试是必考的一个，所以不但要掌握 TC...</div>	阅读量 2万+ 博文

为什么程序员在学习编程的时候什么都记不住？	阅读数 2万+
在程序员的职业生涯中，记住所有你接触过的代码是一件不可能的事情！那么我们该如何解决这一问题？作者 Dylan...	博文
JAVA实现商品信息管理系统	阅读数 5897
任务与实现 超市商品管理系统 题目要求 超市中商品分为四类，分别是食品、化妆品、日用品和饮料。每种商品都包...	博文
500行代码，教你用python写个微信飞机大战	阅读数 5万+
这几天在重庆微信小游戏的飞机大战，玩着玩着就在思考人生了，这飞机大战怎么就可以做的那么好，操作简单，简...	博文
唐僧团队要裁员，你会裁谁？	阅读数 4万+
提问： 西游记取经团为了节约成本，唐太宗需要在这个团队里裁掉一名队员，该裁掉哪一位呢，为什么? 为了完成西...	博文
2019诺贝尔经济学奖得主：贫穷的本质是什么？	阅读数 7865
2019年诺贝尔经济学奖，颁给了来自麻省理工学院的 阿巴希·巴纳吉（Abhijit Vinayak Banerjee）、艾丝特·杜芙若...	博文
linux：最常见的linux命令（centOS 7.6）	阅读数 9897
最常见，最频繁使用的20个基础命令如下： 皮一下，这都是干货偶，大佬轻喷 一、linux关机命令： 1.shutdown命...	博文
只因写了一段爬虫，公司200多人被抓！	阅读数 10万+
“一个程序员写了个爬虫程序，整个公司200多人被端了。” “不可能吧！” 刚从朋友听到这个消息的时候，我有点...	博文
别在学习框架了，那些让你起飞的计算机基础知识。	阅读数 4万+
我之前里的文章，写的大部分都是与计算机基础知识相关的，这些基础知识，就像我们的内功，如果在未来想要走的...	博文
MySQL数据库—SQL汇总	阅读数 9766
一、准备 下文整理常见SQL语句的用法，使用MySQL5.7测试，参考了尚硅谷MySQL教程及用例。用例sql： 链接: h...	博文
动画：用动画给女朋友讲解 TCP 四次分手过程	阅读数 2万+
作者 小鹿 来源 公众号：小鹿动画学编程 写在前边 大家好，我们又见面了，做为一个业余的动画师，上次的用动...	博文
程序员必须掌握的核心算法有哪些？	阅读数 6万+
由于我之前一直强调数据结构以及算法学习的重要性，所以就有一些读者经常问我，数据结构与算法应该要学习到哪...	博文
SQL基本语法入门 看这里就够了	阅读数 2117
SQL执行顺序 第一步：执行FROM 第二步：WHERE条件过滤 第三步：GROUP BY 分组 第四步：执行SELECT 投影...	博文
如何优化MySQL千万级大表，我写了6000字的解读	阅读数 3万+
这是学习笔记的第2138篇文章 千万级大表如何优化，这是一个很有技术含量的问题，通常我们的直觉思维都会跳转...	博文
面试最后一问：你有什么问题想问我吗？	阅读数 3万+
尽管，我们之前分享了这么多关于面试的主题： 高薪必备的一些Spring Boot高级面试题 面试必问：设计模式遵循的...	博文
python 程序员进阶之路：从新手到高手的100个模块	阅读数 4万+
在知乎和CSDN的圈子里，经常看到、听到一些 python 初学者说，学完基础语法后，不知道该学什么，学了也不知...	博文
Vue + Spring Boot 项目实战（十三）：使用 Shiro 实现用户信息加密与登录认证	阅读数 3682
如题，希望这篇文章能解答你对 shiro 的一些疑问。	博文
Python——画一棵漂亮的樱花树（不同种樱花+玫瑰+圣诞树喔）	阅读数 2万+
最近翻到一篇知乎，上面有不少用Python（大多是turtle库）绘制的树图，感觉很漂亮，我整理了一下，挑了一些我...	博文
给 python 初学者的四条忠告	阅读数 1万+
学习一种编程语言，首先要找一款合用的集成开发工具，似乎是自然而然的想法。为什么不呢？IDE可以自动补齐， ...	博文
Linux/C/C++ 不可错过的好书	阅读数 5809
来源：公众号【编程珠玑】 作者：守望先生 ID：shouwangxiansheng 前言 经常有读者让我推荐书籍，这次我就把...	博文
史上最强Tomcat8性能优化	阅读数 2万+
文章目录授人以鱼不如授人以渔目的服务器资源Tomcat配置优化Linux环境安装运行Tomcat8AJP连接执行器（线程...	博文

漫话：什么是 https ?这应该是全网把 https 讲的最好的一篇文章了

今天这篇文章，讲通过对话的形式，让你由浅入深着知道，为什么 Https 是安全的。 一、对称加密 一禅：在每次发...

阅读量 1万+

史上最全的mysql基础教程

启动与停止 启动mysql服务 sudo /usr/local/mysql/support-files/mysql.server start 停止mysql服务 sudo /usr/l...

阅读量 ...

c# mysql插入 c# sha256 加密 c#窗体上的叉添加事件 c#打印图片文件 c#後台調前台js c#控制台美化 c# 标记 浮点型 c#获取当前的农历日期 c# 构造函数重载 c# 页面功能注释



尘中远

TA的个人主页 >

私信

关注

原创42 粉丝896 获赞490 评论655 访问79万+

等级: 博客 6 周排名: 1万+

积分: 4764 总排名: 8885

勋章: 



超声波流量计

归档

2019年5月

1篇

2018年12月

1篇

2018年11月

3篇

2018年4月

1篇

2018年3月

1篇

2018年2月

1篇

2018年1月

1篇

2017年11月

1篇

展开

最新文章

数据驱动智能制造论坛经验及知识分享

频谱分析-FFT之后的那些事情

记录一次失败的控件编写经历——initStyleOption居然不是虚函数

Qt编写控件时遇到underMouse判断错误的情况(误判State_MouseOver)

记录一次从MinGw转到MSVC编译器的错误经历

热门文章

https://blog.csdn.net/czyt1988/article/details/26018513

15/16

- Qt使用多线程的一些心得——1.继承QThread的多线程使用方法

阅读数 70699
- TeeChart替代品，MFC下好用的高速绘图控件-(Hight-Speed Charting)

阅读数 59684
- 绘图控件第五讲——绘制动态曲线

阅读数 54661
- VS2010 使用TeeChart绘图控件 - 之一 - 控件和类的导入

阅读数 53225
- Qt树形控件QTreeView使用1——节点的添加删除操作

阅读数 53086

分类专栏

 C++

13篇

 qt

25篇

 matlab

3篇

 python

 mfc

5篇

展开

- 最新评论
- 频谱分析-FFT之后的那些事情

Skilla: 注意N是采样点而不是进行FFT的点数，这句话是什么意思？FFT点数和采样点数不一致吗 ...
- 绘图控件第五讲——绘制动态曲线

zhuzemin45: 很仔细地看了你的文章，写得很非常精彩，但遗憾的是想参考代码时才知道，代码 ...
- VS2010 使用TeeChart...

zhao_7777: 大佬TEEchar 5 控件可以发一下吗，找了半天都找不到
- TeeChart替代品，MFC下好...

misterwang1: [reply]alf929531643[/reply] 能用吗，我也想啊知道
- QDockWidget嵌套布局详解...

czyt1988: 上面下载链接本来是免积分的，都被csdn私自篡改改为50积分，文章内容和源码已经同 ...



11.11域名特惠_热门域名1元起

购买建站套餐更划算,建站主机3.5折,加1元享爆款优质域名,全网超值套餐



CSDN学院



CSDN企业招聘

 QQ客服

 kefu@csdn.net