

柳北风儿~~~~~欲宇仙灵

- [博客](#)
- [微博](#)
- [相册](#)
- [收藏](#)
- [留言](#)
- [关于我](#)

Qt-在表格(QTableView)中插入图片

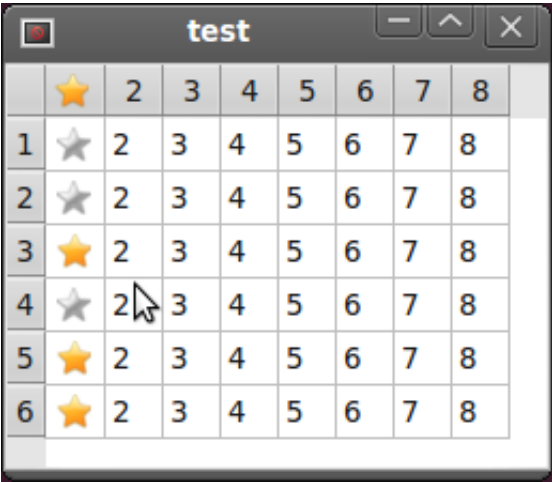
- 博客分类：

 - [Qt](#)

QTableView插入图片DelegateQt

通过Qt的委托，可以实现自定义QTableView中每一项的特殊实现方式。



本例，利用Delegate 实现在QTableView中插入Pixmap图



Cpp代码

- [文件] main.cpp ~ 273B 下载(10)
- /**
- * @file main.cpp

```
4.  * @brief
5.  * @author xiangxw
6.  * @version 0.0
7.  * @date 2011年月日
8.  */
9.
10. #include<QApplication>
11. #include"MyTableView.h"
12.
13.
14. int main(int argc, char *argv[])
15. {
16.     QApplication app(argc, argv);
17.
18.     MyTableView table;
19.     table.show();
20.
21.     return app.exec();
22. }
```



Cpp代码  

```
1. [文件] MyTableView.h ~ 1KB    下载(10)
2. /**
3.  * @file MyTableView.h
4.  * @brief
5.  * @author xiangxw
6.  * @version 0.0
7.  * @date 2011年月日
8.  */
9.
10. #ifndef MYTABLEVIEW_H
11. #define MYTABLEVIEW_H
12.
13. #include<QTableView>
14. #include<QItemDelegate>
15. #include<QStandardItemModel>
16. #include<QPixmap>
17.
18. class MyTableView;
19. class MyItemDelegate;
20. class MyStandardItemModel;
21.
22. class MyTableView:public QTableView
23. {
24. public:
25.     MyTableView(QWidget * parent=0);
26.     virtual ~ MyTableView() {}
27.
```

```

28. protected:
29.     void mouseMoveEvent(QMouseEvent * event);
30.
31. private:
32.     MyItemDelegate * delegate;
33.     MyStandardItemModel * model;
34. };
35.
36. class MyItemDelegate:public QItemDelegate
37. {
38. public:
39.     MyItemDelegate(QObject * parent=0);
40.     virtual ~ MyItemDelegate() {}
41.
42.     void paint(QPainter * painter,
43.         const QStyleOptionViewItem & option,
44.         const QModelIndex & index) const;
45.     bool editorEvent(QEvent * event,
46.         QAbstractItemModel * model,
47.         const QStyleOptionViewItem & option,
48.         const QModelIndex & index);
49. private:
50.     QPixmap favouritePixmap;
51.     QPixmap notFavouritePixmap;
52. };
53. class MyStandardItemModel:public QStandardItemModel
54. {
55. public:
56.     MyStandardItemModel(QObject * parent=0)
57.         :QStandardItemModel(parent) {}
58.     virtual ~ MyStandardItemModel() {}
59.     QVariant data(const QModelIndex & index,
60.         int role=Qt::DisplayRole) const;
61.     QVariant headerData(int section,
62.         Qt::Orientation orientation,
63.         int role=Qt::DisplayRole) const;
64. };
65. #endif

```

Cpp代码  

```

1. [文件] MyTableView.cpp ~ 3KB    下载(16)
2. /**
3.  * @file MyTableView.cpp
4.  * @brief
5.  * @author xiangxw
6.  * @version 0.0
7.  * @date 2011年月日
8.  */
9.

```

```

10. #include<QPainter>
11. #include<QMouseEvent>
12. #include"MyTableView.h"
13.
14. #ifndef NDEBUB
15. #include<QDebug>
16. #endif
17.
18. MyTableView::MyTableView(QWidget * parent)
19. :QTableView(parent)
20. {
21.     delegate=new MyItemDelegate;
22.     model=new MyStandardItemModel;
23.     model->setRowCount(6);
24.     model->setColumnCount(8);
25.
26.     this->setModel(model);
27.     this->setItemDelegate(delegate);
28.
29.     this->resizeColumnsToContents();
30.     this->resizeRowsToContents();
31.     this->setEditTriggers(QAbstractItemView::NoEditTriggers);
32.     this->setSelectionBehavior(QAbstractItemView::SelectRows);
33.     this->setMouseTracking(true);//important
34. }
35.
36. void MyTableView::mouseMoveEvent(QMouseEvent * event)
37. {
38.     int column=this->columnAt(event->x());
39.     int row=this->rowAt(event->y());
40.     if(column==0 && row!=-1){
41.         this->setCursor(Qt::PointingHandCursor);
42.     }
43.     else{
44.         this->setCursor(Qt::ArrowCursor);
45.     }
46. }
47.
48. MyItemDelegate::MyItemDelegate(QObject * parent)
49. :QItemDelegate(parent)
50. {
51.     favouritePixmap=QPixmap(":/favourite.png");
52.     notFavouritePixmap=QPixmap(":/no-favourite.png");
53. }
54.
55. void MyItemDelegate::paint(QPainter * painter,
56.                             const QStyleOptionViewItem & option,
57.                             const QModelIndex & index) const
58. {
59.     if(index.column()!=0){
60.         QItemDelegate::paint(painter, option, index);
61.         return;
62.     }
63.     const QAbstractItemModel * model=index.model();

```

```

64.     QVariant var=model->data(index,Qt::CheckStateRole);
65.     if(var.isNull()) var=false;
66.     const QPixmap & star=var.toBool()?
67. favouritePixmap:notFavouritePixmap;
68.
69.     int width=star.width();
70.     int height=star.height();
71.     QRect rect=option.rect;
72.     int x=rect.x()+rect.width()/2-width/2;
73.     int y=rect.y()+rect.height()/2-height/2;
74.
75.     painter->drawPixmap(x,y,star);
76. }
77.
78. bool MyItemDelegate::editorEvent(QEvent * event,
79.                                   QAbstractItemModel * model,
80.                                   const QStyleOptionViewItem & /*option*/,
81.                                   const QModelIndex & index)
82. {
83.     if(event->type()==QEvent::MouseButtonPress &&
84.         index.column()==0) {
85.         QVariant var=model->data(index,Qt::CheckStateRole);
86.         bool isFavourite=var.toBool();
87.         if(var.isValid())
88.             isFavourite=isFavourite?false:true;
89.         else
90.             isFavourite=true;
91.         model->setData(index,isFavourite,Qt::CheckStateRole);
92.         return true;//I have handled the event
93.     }
94.
95.     return false;
96. }
97.
98. QVariant MyStandardItemModel::data(
99.                                     const QModelIndex & index,
100.                                    int role) const
101. {
102.     int column=index.column();
103.
104.     if(role==Qt::DisplayRole && column!=0)
105.         return column+1;
106.     if(role==Qt::ToolTipRole && column==0)
107.         return tr("love");
108.     return QStandardItemModel::data(index,role);
109. }
110.
111. QVariant MyStandardItemModel::headerData(int section,
112.                                           Qt::Orientation orientation,
113.                                           int role) const
114. {
115.     if(section==0 && orientation==Qt::Horizontal) {
116.         if(role==Qt::DecorationRole)
117.             return QIcon(":/favourite.png");

```

```



118.         if(role==Qt::DisplayRole)
119.             return "";
120.         if(role==Qt::ToolTipRole)
121.             return tr("love");
122.     }
123.     return QStandardItemModel::headerData(section, orientation, role);
124. }

```

转: http://www.oschina.net/code/snippet_96486_4133

- [查看图片附件](#)



分享到:  

[\(四\)Paint函数实现QTableView中增加QCheck ...](#) | [两款免费DCIOM 图像浏览软件介绍和DICOM图 ...](#)

- 2012-05-23 09:37
- 浏览 5670
- [评论\(0\)](#)
- 分类: [编程语言](#)
- [相关推荐](#)

评论



发表评论



[您还没有登录, 请您登录后再发表评论](#)



qimo601

- 浏览: 982865 次
- 性别: 
- 来自: 苏州
-  我现在离线

最近访客

[更多访客>>](#)



[dylinshi126](#)



[bububu2012](#)



[lsuming](#)



[blindness_god](#)

文章分类

- [全部博客 \(335\)](#)
- [技术总结 \(53\)](#)
- [书籍分享 \(9\)](#)
- [网络 \(7\)](#)
- [操作系统 \(3\)](#)
- [C/C++ \(63\)](#)
- [C# \(1\)](#)
- [java \(2\)](#)
- [web前台技术 \(10\)](#)
- [数据库 \(17\)](#)
- [项目管理 \(2\)](#)
- [Qt \(119\)](#)
- [图像处理 \(13\)](#)
- [DCMTK \(33\)](#)

社区版块

- [我的资讯 \(0\)](#)
- [我的论坛 \(7\)](#)
- [我的问答 \(4\)](#)

存档分类

- [2012-12 \(1\)](#)
- [2012-11 \(5\)](#)
- [2012-10 \(6\)](#)
- [更多存档...](#)

最新评论

- [NO.6: 我怎么看不到设置的谷歌广告在哪里显示的啊?
自由在博客里插入广告, 有钱你不赚吗???](#)
- [thisisw: qxorm下载后 解压失败:\)
C++ 对象关系映射 \(ORM\) 介绍](#)
- [qimo601: wenluderen 写道感谢 博主解决我一个长达3个月的困扰 ...
VS2008中VC 项目 文件目录的管理 \(最终无错误版\)](#)
- [wenluderen: 感谢 博主解决我一个长达3个月的困扰 谢谢
VS2008中VC 项目 文件目录的管理 \(最终无错误版\)](#)

- [美丽的小岛: ...](#)
[Qt自定义委托在QTableView中绘制控件、图片、文字](#)

声明：ITeye文章版权属于作者，受法律保护。没有作者书面许可不得转载。若作者同意转载，必须以超链接形式标明文章原始出处和作者。

© 2003-2015 ITeye.com. All rights reserved. [京ICP证110151号 京公网安备110105010620]