

新浪博客

[\[公告\]](#)

[一去、二三里](#)

[个人中心](#)

[发博文](#)

[消息](#)



心在山间

一去、二三里的博客

<http://blog.sina.com.cn/liang19890820>

[首页](#) | [博文目录](#) | [图片](#) | [关于我](#)

发博文

页面设置

个人中心

个人资料

[管理]

一去、二三里

Qing

微博

博客等级: **18**

博客积分: **78** 新

博客访问: **407,806**

关注人气: **378**

获赠金笔: **12**

赠出金笔: **0**

荣誉徽章:

相关博文

- NodeSecurity

一去、二三里
- Qt之QTableView

一去、二三里
- Qt之模型/视图（委托）

一去、二三里
- Qt之透明提示框（模拟QQ）

一去、二三里
- 一款VS开发利器-VisualAssistX

一去、二三里
- Qt之自定义提示框二（右下角冒泡

一去、二三里
- cin详解（cin.get()、cin.getlin

一去、二三里
- Qt之自定义界面（二）添加最小化

一去、二三里
- QTableWidget详解（样式、右键菜

一去、二三里
- QMenu的个性化定制

一去、二三里

更多>>

正文

字体大小: 大 中 小

Qt之模型/视图 (实时更新数据) (2014-01-09 15:59:48) [编辑][删除]

 转载 ▼

标签: qt mvc 模型/视图 qt进度条 it 分类: Qt

上两节简单介绍了Qt中对于模型/视图的编程,大部分助手里说的很清楚了,现在就开始实战部分吧!
在实际应用中,视图展示的数据往往并非一成不变的,那么如何实时更新成了一个很重要的问题!

功能:

- (1) 添加委托 (进度条)
- (2) 显示文件名称、大小、进度、速度、剩余时间、状态等。
- (3) 可进行添加、更新、删除、清空等操作。
- (4) 实时更新数据

先看一个效果图:



name	size	progress	speed	left time	state
file : 0	0KB	91%	0KB/s	08:08:08	uploading
file : 1	1KB	36%	1KB/s	08:08:08	uploading
file : 2	2KB	74%	2KB/s	08:08:08	uploading
file : 3	3KB	20%	3KB/s	08:08:08	uploading
file : 4	4KB	96%	4KB/s	08:08:08	uploading
file : 5	5KB	21%	5KB/s	08:08:08	uploading
file : 6	6KB	48%	6KB/s	08:08:08	uploading
file : 7	7KB	99%	7KB/s	08:08:08	uploading
file : 8	8KB	68%	8KB/s	08:08:08	uploading
file : 9	9KB	84%	9KB/s	08:08:08	uploading
file : 10	10KB	0%	10KB/s	08:08:08	uploading

委托 (进度条):

```
ProgressBarDelegate::ProgressBarDelegate(QObject *parent)
: QTableWidgetItem(parent)
{
}
```

```
void ProgressBarDelegate::paint(QPainter* painter, const QStyleOptionViewItem& option, const
QModelIndex& index) const
{
    if(index.column() == 2)
    {
        int progress = index.model ()->data(index, Qt::DisplayRole).toInt ();
        QStyleOptionProgressBarV2 progressBarOption;
        progressBarOption.state = QStyle:: State_Enabled;
        progressBarOption.direction = QApplication:: layoutDirection ();
        progressBarOption.rect = option.rect;
```

- 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渐
- 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



苗族情人节捕鱼比赛
马语者表演驯马神技
母狮惨遭水牛顶伤
抓拍鲨鱼呲牙怪笑照
曼妙旅程：潜拍海底世界
嬉戏野兔击掌求偶

[查看更多>>](#)

谁看过这篇博文

	新湖期...	3月2日
	云知道	3月1日
	之唯伊人	2月28日
	柴小希要...	2月28日
	MrGoodman	2月27日
	Guo_X	2月27日
	moonlqer	2月26日
	储小团	2月25日
	右右youyou	2月24日
	刘燕_85892	2月23日
	todaysky	2月22日
	somelany0	2月17日

```
progressBarOption.fontMetrics = QApplication:: fontMetrics ();
progressBarOption.minimum = 0;
progressBarOption.maximum = 100;
progressBarOption.textAlignment = Qt:: AlignCenter;
progressBarOption.textVisible = true;
progressBarOption.progress = progress;
progressBarOption.text = QString("%1%").arg(progressBarOption.progress);
QApplication:: style ()->drawControl(QStyle::CE_ProgressBar, &progressBarOption, painter);
} else {
return QItemDelegate::paint (painter, option, index);
}
}
```

模型:

```
TableModel::TableModel(QObject *parent)
: QAbstractTableModel (parent), arr_row_list (NULL)
{

}

TableModel::~~TableModel (void)
{
    arr_row_list = NULL;
}

void TableModel::setHorizontalHeaderList(QStringList horizontalHeaderList)
{
    horizontal_header_list = horizontalHeaderList;
}

void TableModel::setVerticalHeaderList (QStringList verticalHeaderList)
{
    vertical_header_list = verticalHeaderList;
}

int TableModel::rowCount(const QModelIndex &parent) const
{
    if(vertical_header_list.size() > 0)
        return vertical_header_list.size();

    if(NULL == arr_row_list)
        return 0;
    else
        return arr_row_list->size();
}

int TableModel::columnCount(const QModelIndex &parent) const
{
    if(horizontal_header_list.size() > 0)
        return horizontal_header_list.size();

    if(NULL == arr_row_list)
        return 0;
    else if(arr_row_list->size() < 1)
        return 0;
    else
        return arr_row_list->at(0).size();
}

QVariant TableModel::data(const QModelIndex &index, int role) const
{
    if (!index.isValid())
        return QVariant();

    if(NULL == arr_row_list)
        return QVariant();
```

```
if(arr_row_list->size() < 1)
    return QVariant();

if (role == Qt::TextAlignmentRole)
{
    return int(Qt::AlignLeft | Qt::AlignVCenter);
}
else if (role == Qt::DisplayRole)
{
    if(index.row() >= arr_row_list->size())
        return QVariant();
    if(index.column() >= arr_row_list->at(0).size())
        return QVariant();
    return arr_row_list->at(index.row()).at(index.column());
}
return QVariant();
}

QVariant TableModel::headerData(int section, Qt::Orientation orientation, int role) const
{
    if(role==Qt::DisplayRole)
    {
        if(orientation==Qt::Horizontal) // 水平表头
        {
            if(horizontal_header_list.size() > section)
                return horizontal_header_list[section];
            else
                return QVariant();
        }
        else
        {
            if(vertical_header_list.size() > section)
                return vertical_header_list[section]; // 垂直表头
            else
                return QVariant();
        }
    }
    return QVariant();
}

Qt::ItemFlags TableModel::flags(const QModelIndex &index) const
{
    if (!index.isValid())
        return Qt::NoItemFlags;

    Qt::ItemFlags flag = QAbstractItemModel::flags(index);

    // flag|=Qt::ItemIsEditable // 设置单元格可编辑, 此处注释, 单元格无法被编辑
    return flag;
}

void TableModel::setModalDatas(QList< QStringList > *rowlist)
{
    arr_row_list = rowlist;
}

void TableModel::refrushModel()
{
    beginResetModel();
    endResetModel();

    emit updateCount(this->rowCount(QModelIndex()));
}
```

视图:

```
TableView::TableView(QWidget *parent)
: QTableView(parent)
{
    this->setAlternatingRowColors(true);
    this->setStyleSheet( "QTableView{background-color: rgb(250, 250, 115);"
    "alternate-background-color: rgb(141, 163, 215);} " );
    this->setSelectionBehavior(QAbstractItemView::SelectRows);
    this->horizontalHeader()->setStretchLastSection(true);
    this->horizontalHeader()->setHighlightSections(false);
    this->verticalHeader()->setVisible(false);
    this->setShowGrid(false);
    this->setEditTriggers(QAbstractItemView::NoEditTriggers);
    this->setSelectionMode(QAbstractItemView::ExtendedSelection);

    model = new TableModel();
    this->setModel(model);
    this->initHeader();
    model->setModalDats(&grid_data_list);
    progressbar_delegate = new ProgressBarDelegate();
    this->setItemDelegate(progressbar_delegate);

    connect(model, &TableModel::updateCount, this, &TableView::updateCount);

    this->initHeader();
}

TableView::~TableView(void)
{
    if(progressbar_delegate) {
        delete progressbar_delegate;
        progressbar_delegate = NULL;
    }

    if(model) {
        delete model;
        model = NULL;
    }
    grid_data_list.clear();
}

void TableView::addRow(QStringList rowList)
{
    grid_data_list.append(rowList);
    model->refrushModel();
}

void TableView::remove()
{
    QModelIndexList model_index_list = this->selectedIndexes();
    int model_count = model_index_list.count();
    if(model_count <= 0)
        return;

    QList list_row;
    for(int i=model_count-1; i>=0; i--)
    {
        QModelIndex model_index = model_index_list.at(i);
        int row = model_index.row();
        if(!list_row.contains(row))
            list_row.append(row);
    }

    if(list_row.isEmpty())
        return;
```

```
qSort(list_row);

for(int i=list_row.count()-1; i>=0; i--)
{
    grid_data_list.removeAt(list_row.at(i));
}

model->refrushModel();
}

void TableView::clear()
{
    grid_data_list.clear();
    model->refrushModel();
}

int TableView::rowCount()
{
    return model->rowCount(QModelIndex());
}

void TableView::initHeader()
{
    QStringList header;
    header << tr("name") << tr("size") << tr("progress") << tr("speed") << tr("left time") <<
tr("state");
    model->setHorizontalHeaderList(header);
}

void TableView::changeValue()
{
    //这里以10行尾列进行更新
    int row_count = this->rowCount();
    if(row_count < 10)
        return;

    for(int i=0; i<10; i++)
    {
        QStringList file_list = grid_data_list.at(i);
        int progress = rand()0;
        QStringList row_list;
        row_list << file_list.at(0) << file_list.at(1) << QString::number(progress) << file_list.at(3) <<
file_list.at(4) << file_list.at(5);
        grid_data_list.replace(i, row_list);
    }

    model->refrushModel();
}
```

完整的工程（源码）下载地址：<http://download.csdn.net/detail/u011012932/6829783>。

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_a6fb6cc90101hhse.html。

1

喜欢

分享：        

阅读(5665) | 评论(4) | 收藏(2) | 已有6人转载▼ | 喜欢▼ | 打印

已投稿到： 排行榜

前一篇：Qt之模型/视图（委托）
后一篇：Qt之模型/视图（自定义风格）

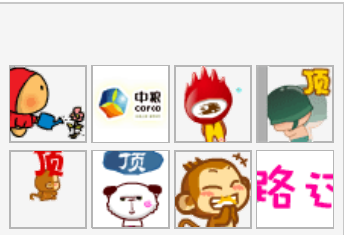
评论 重要提示：警惕虚假中奖信息 [发评论]

Michael
如何更改进度条的样式呢？ 比如背景色，风格等。
2014-5-8 23:20 回复(1)

kgcf2011
当点击update后无法选中列表
2014-8-14 14:02 回复(1)

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 后一篇 ＞
Qt之模型/视图（委托） Qt之模型/视图（自定义风格）