



[管理]



荣誉徽章:

更多>>

正文

字体大小: 大 中 小

(2013-07-24 19:28:20) [编辑][删除]

+ 转载 ▼

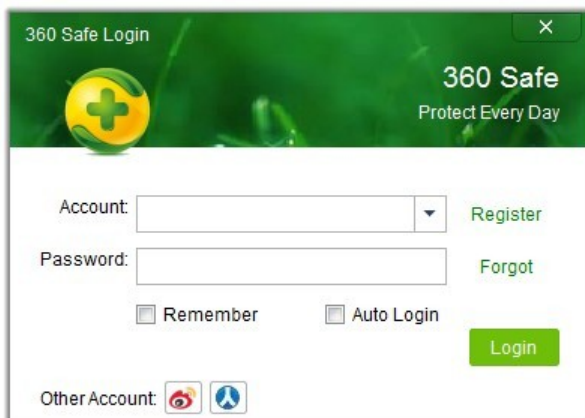
标签: qt qt窗体阴影 qt阴影 阴影效果 it 分类: Qt

阴影边框很常见，诸如360以及其他很多软件都有类似效果，了解CSS3的同学们应该都知道box-shadow，它就是来设定阴影效果的，那么Qt呢？看过一些资料，说是QSS是基于CSS2的，既然如此，box-shadow是基于CSS3的！那么Qt定然就用不了！

搜了一些资料，每张图片都做成阴影效果的固然不可能，直接舍弃（即使可以，也不采纳）。如果实时的去画图，效率太低，最后选择了拼图的方式！



效果如下:



左上角、左下角、右上角、右下角、上、下、左、右，这几个方向都绘制对应的图即可！

- 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：早该说不了！
- 【DIY】做一枚宫灯迎元宵



查看更多>>

谁看过这篇博文

	Arrow	今天09:08
	Bass	2月26日
	卫城	2月26日
	自然麻雀	2月26日
	右右youyou	2月24日
	用户16372...	2月20日
	大道至简	2月13日
	Evoma	2月11日
	龙龙8810	2月10日
	roy7124	2月10日
	你是猪么2...	2月9日
	enaldick	2月6日

```
#include "shadow_widget.h"

ShadowWidget::ShadowWidget(QWidget *parent)
    : QDialog(parent)
{
    setWindowFlags(Qt::FramelessWindowHint | Qt::Dialog);
    setAttribute(Qt::WA_TranslucentBackground);
}

ShadowWidget::~ShadowWidget()
{
}

void ShadowWidget::paintEvent(QPaintEvent *event)
{
    QPainter painter(this);
    this->drawShadow(painter);
    painter.setPen(Qt::NoPen);
    painter.setBrush(Qt::white);
    painter.drawRect(QRect(SHADOW_WIDTH, SHADOW_WIDTH, this->width()-2*SHADOW_WIDTH, this->height()-2*SHADOW_WIDTH));
}

void ShadowWidget::drawShadow(QPainter &painter)
{
    //绘制左上角、左下角、右上角、右下角、上、下、左、右边框
    QList pixmaps;
    pixmaps.append(QPixmap(":/shadow/shadow_left"));
    pixmaps.append(QPixmap(":/shadow/shadow_right"));
    pixmaps.append(QPixmap(":/shadow/shadow_top"));
    pixmaps.append(QPixmap(":/shadow/shadow_bottom"));
    pixmaps.append(QPixmap(":/shadow/shadow_left_top"));
    pixmaps.append(QPixmap(":/shadow/shadow_right_top"));
    pixmaps.append(QPixmap(":/shadow/shadow_left_bottom"));
    pixmaps.append(QPixmap(":/shadow/shadow_right_bottom"));

    painter.drawPixmap(0, 0, SHADOW_WIDTH, SHADOW_WIDTH, pixmaps[4]);
    painter.drawPixmap(this->width()-SHADOW_WIDTH, 0, SHADOW_WIDTH, SHADOW_WIDTH, pixmaps[5]);
    painter.drawPixmap(0, this->height()-SHADOW_WIDTH, SHADOW_WIDTH, SHADOW_WIDTH, pixmaps[6]);
    painter.drawPixmap(this->width()-SHADOW_WIDTH, this->height()-SHADOW_WIDTH, SHADOW_WIDTH, SHADOW_WIDTH, pixmaps[7]);
    painter.drawPixmap(0, SHADOW_WIDTH, SHADOW_WIDTH, this->height()-2*SHADOW_WIDTH, pixmaps[0].scaled(SHADOW_WIDTH, this->height()-2*SHADOW_WIDTH));
    painter.drawPixmap(this->width()-SHADOW_WIDTH, SHADOW_WIDTH, SHADOW_WIDTH, this->height()-2*SHADOW_WIDTH, pixmaps[1].scaled(SHADOW_WIDTH, this->height()-2*SHADOW_WIDTH));
    painter.drawPixmap(SHADOW_WIDTH, 0, this->width()-2*SHADOW_WIDTH, SHADOW_WIDTH, pixmaps[2].scaled(this->width()-2*SHADOW_WIDTH, SHADOW_WIDTH));
    painter.drawPixmap(SHADOW_WIDTH, this->height()-SHADOW_WIDTH, this->width()-2*SHADOW_WIDTH, SHADOW_WIDTH, pixmaps[3].scaled(this->width()-2*SHADOW_WIDTH, SHADOW_WIDTH));
}
```

写一个公共的类，如果窗口要实现阴影效果直接继承就行了！SHADOW_WIDTH为阴影边框的像素，可以根据自己的需求自由调节！
也可使用QGraphicsDropShadowEffect来实现。。。

更多参考：
• Qt之再谈阴影边框.

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_a6fb6cc90101e0c7.html。



分享：

阅读 | 评论 | 收藏 | 已有6人转载▼ | 喜欢▼ | 打印 已投稿到： 排行榜

前一篇：Qt4过渡至Qt5
后一篇：Qt之再谈阴影边框

评论

重要提示：警惕虚假中奖信息

[发评论]

codearts
楼主又出好文章了，学习。

qDrawBorderPixmap 用这个函数绘制边框更简单
2013-7-26 16:55

回复 (1)

发评论

☐ 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 >

Qt4过渡至Qt5

Qt之再谈阴影边框