



## 第9篇 (2D绘图) 渐变填充

代码地址：<https://github.com/Lornatang/QtStartQuicklyTutorial/tree/main/Painter01>

### 目录

目录

一、线性渐变

二、辐射渐变

三、锥形渐变

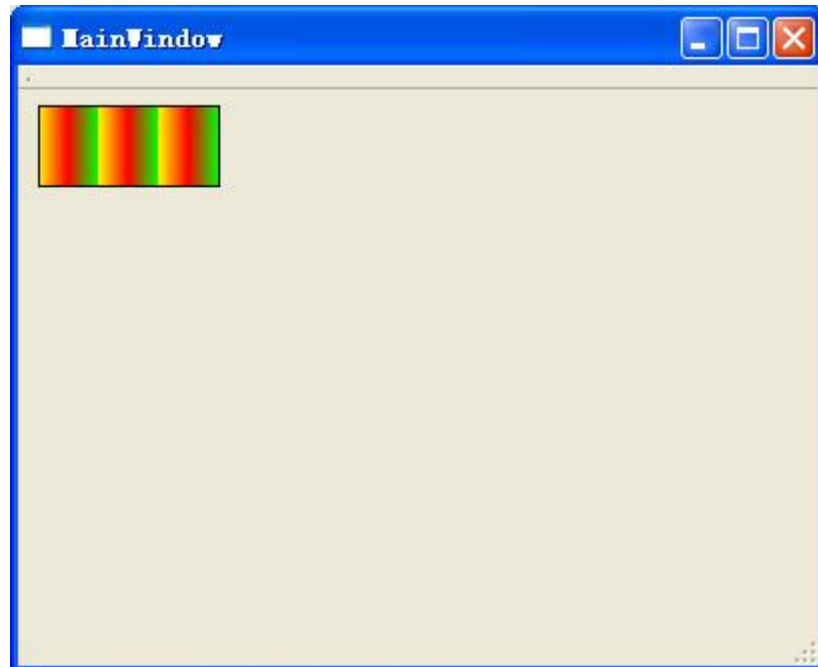
### 一、线性渐变

1. 我们仍然在上一节创建的项目中进行讲解。更改 `paintEvent()` 函数如下：

```
void MainWindow::paintEvent(QPaintEvent *)
{
    //线性渐变
    QLinearGradient linearGradient(QPointF(40, 190), QPointF(70, 190));
    //插入颜色
    linearGradient.setColorAt(0, Qt::yellow);
    linearGradient.setColorAt(0.5, Qt::red);
    linearGradient.setColorAt(1, Qt::green);
    //指定渐变区域以外的区域的扩散方式
    linearGradient.setSpread(QGradient::RepeatSpread);
    //使用渐变作为画刷
    QPainter painter(this);
    painter.setBrush(linearGradient);
}
```

```
painter.drawRect(10, 20, 90, 40);  
}
```

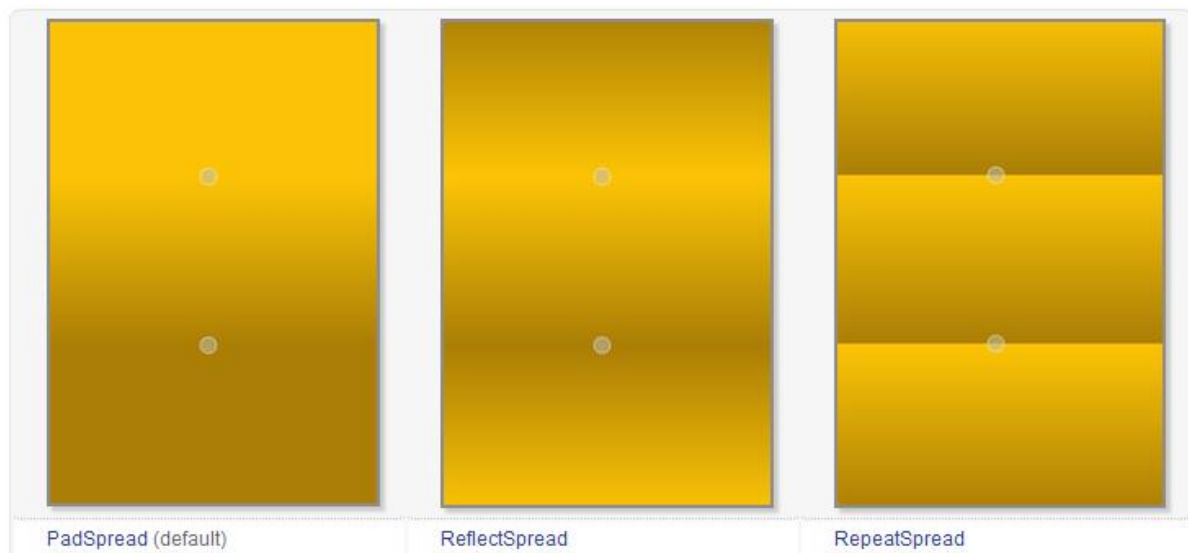
运行程序，效果如下图所示。



## 2. 介绍

对于线性渐变 `QLinearGradient::QLinearGradient ( const QPointF & start, const QPointF & finalStop )` 需要指定开始点 `start` 和结束点 `finalStop`，然后将开始点和结束点之间的区域进行等分，开始点的位置为 `0.0`，结束点的位置为 `1.0`，而它们之间的位置按照距离比例进行设定，然后使用 `QGradient::setColorAt( qreal position, const QColor & color )` 函数在指定的位置 `position` 插入指定的颜色 `color`，当然，这里的 `position` 的值要在0到1之间。

这里还可以使用 `setSpread()` 函数来设置填充的扩散方式，即指明在指定区域以外的区域怎样进行填充。扩散方式由 `QGradient::Spread` 枚举变量定义，它一共有三个值，分别是 `QGradient::PadSpread`，使用最接近的颜色进行填充，这是默认值，如果我们不使用 `setSpread()` 指定扩散方式，那么就会默认使用这种方式；`QGradient::RepeatSpread`，在渐变区域以外的区域重复渐变；`QGradient::ReflectSpread`，在渐变区域以外将反射渐变。在线性渐变中这三种扩散方式的效果下图所示。要使用渐变填充，可以直接在 `setBrush()` 中使用，这时画刷风格会自动设置为相对应的渐变填充。

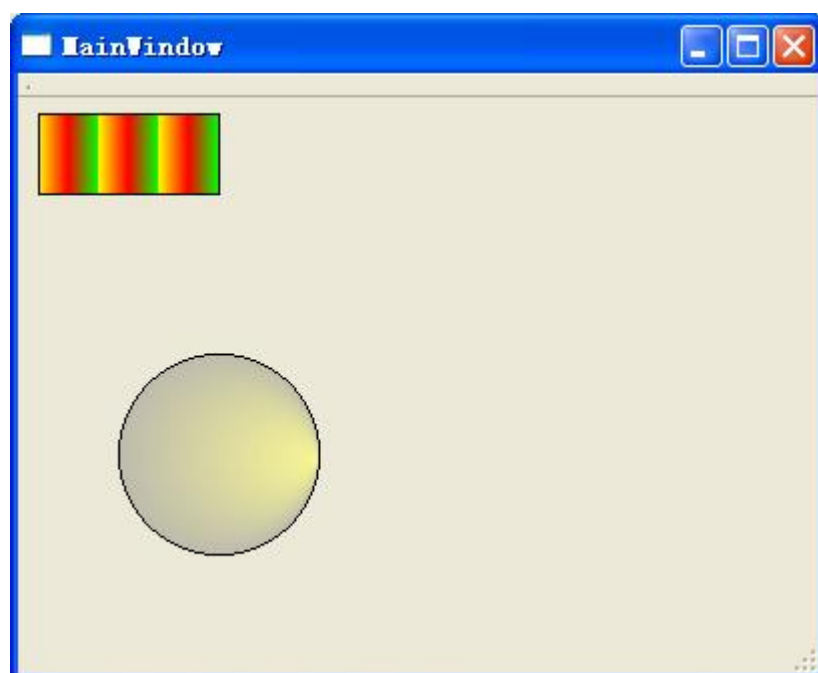


## 二、辐射渐变

1. 继续在 `paintEvent()` 函数中添加如下代码：

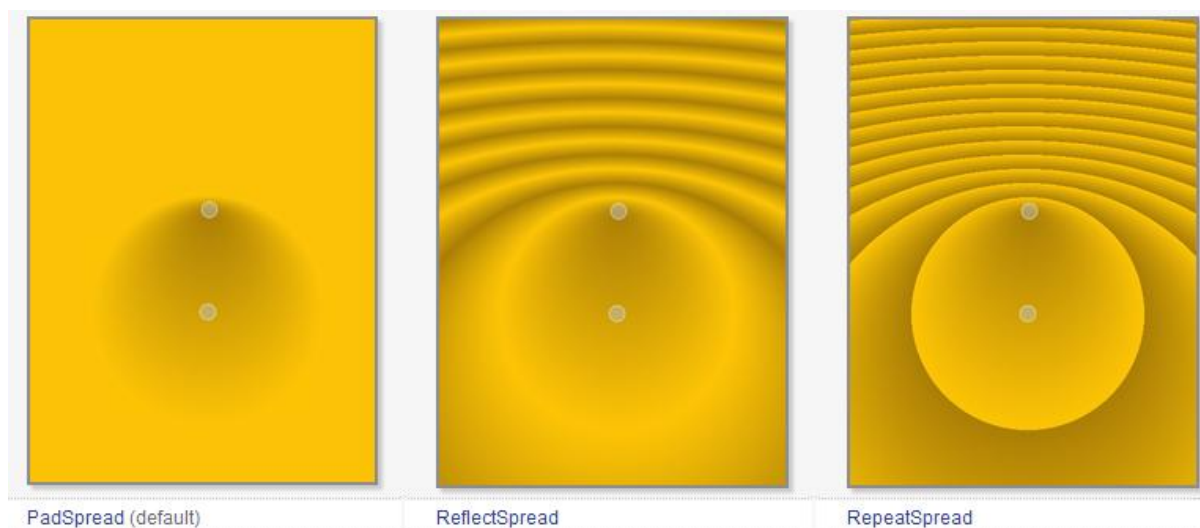
```
//辐射渐变
QRadialGradient radialGradient(QPointF(100, 190),50,QPointF(275,200));
radialGradient.setColorAt(0, QColor(255, 255, 100, 150));
radialGradient.setColorAt(1, QColor(0, 0, 0, 50));
painter.setBrush(radialGradient);
painter.drawEllipse(QPointF(100, 190), 50, 50);
```

运行程序，效果如下图所示。



## 2. 介绍

对于辐射渐变 `QRadialGradient::QRadialGradient( const QPointF & center, qreal radius, const QPointF & focalPoint )` 需要指定圆心 `center` 和半径 `radius`，这样就确定了一个圆，然后再指定一个焦点 `focalPoint`。焦点的位置为0，圆环的位置为1，然后在焦点和圆环间插入颜色。辐射渐变也可以使用 `setSpread()` 函数设置渐变区域以外的区域的扩散方式，三种扩散方式的效果如下图所示。

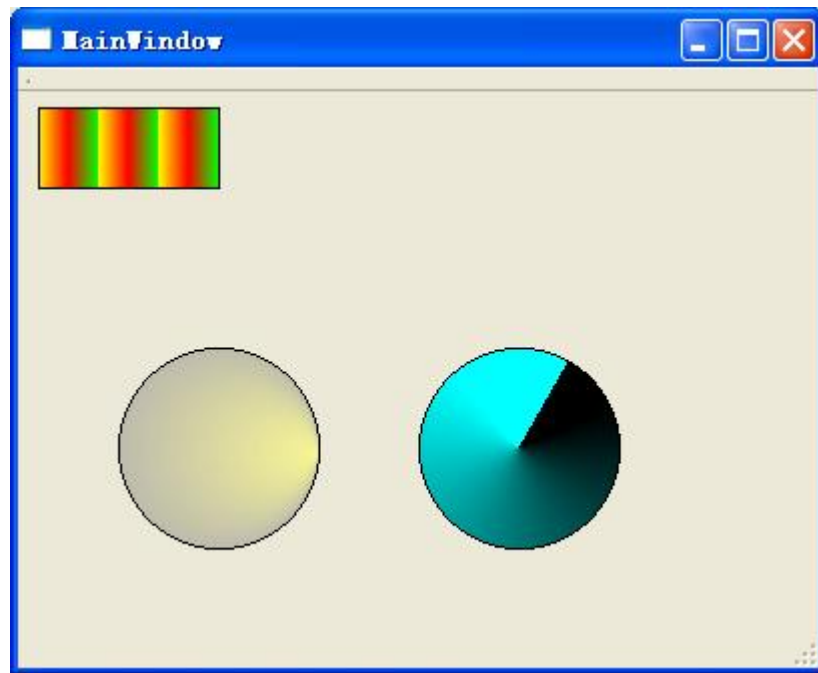


## 三、锥形渐变

1. 接着在 `paintEvent()` 函数里面添加如下代码：

```
//锥形渐变
QConicalGradient conicalGradient(QPointF(250, 190), 60);
conicalGradient.setColorAt(0.2, Qt::cyan);
conicalGradient.setColorAt(0.9, Qt::black);
painter.setBrush(conicalGradient);
painter.drawEllipse(QPointF(250, 190), 50, 50);
```

运行程序，效果如下图所示。



## 2. 介绍

对于锥形渐变 `QConicalGradient::QConicalGradient ( const QPointF & center, qreal angle )` 需要指定中心点 `center` 和一个角度 `angle`（其值在0到360之间），然后沿逆时针从给定的角度开始环绕中心点插入颜色。这里给定的角度沿逆时针方向开始的位置为0，旋转一圈后为1。`setSpread()` 函数对于锥形渐变没有效果。