

Linux下程序开发：用QT创建新风格 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/144/2021_2022_Linux_E4_B8_8B_E7_A8_c103_144951.htm

1 . Qt的风格 a) Qt简介 Qt是一个跨平台的C 图形用户界面应用程序开发库，使用Qt可以开发出高质量的图形用户接口，它是完全面向对象的、易于扩展且允许真正的组件编程。Qt获得了很大的成功，特别是它的信号-槽机制是非常值得研究的通信机制，它也是Linux发行版标准组件KDE(K Desktop Enviroment)的基础。 b) 风格机制 Qt的风格机制实现了不同平台上的图形用户接口（GUI）的观感（look and feel），例如Windows平台上通常使用Windows或Windows-xp风格，而Unix平台上通常使用Motif、CDE风格。 下图显示了Qt中与风格相关的类的继承关系 QStyle是所有风格类的基类，它控制着所有的部件（widget即windows编程中的控件）的界面风格或观感，它定义了大量的枚举类型和十几个函数。枚举类型表示界面上的不同元素（如组合框中的按钮，按钮的边框等）；函数控制图形用户界面的绘制，但大多数函数基本上只是一些声明而没有函数实现，他们的实现在QCommonStyle、QWindowStyle、QMotifStyle及其子类中。QStyle只实现了3个函数drawItem(), itemRect(), visualRect()。 drawItem(): 负责绘制文本和像素图。 itemRect(): 返回文本或图像所占的区域。 visualRect(): 返回逻辑坐标，这个函数使Qt实现right-to-left风格（阿文、维文传统是文本从右向左显示，因此控件布局也是从右向左）。如下图所示：可以看到菜单、工具条是右对齐、单选框的按钮在右边 c) 创建新风格的步骤 在Qt中实现一种新风格的步骤很简单：只需选择一个

风格类（如QCommonStyle或QStyle）作为父类，然后实现感兴趣的函数即可。难点在于函数的实现。选择父类：可以选择QStyle, QCommonStyle, QWindowStyle, QMotifStyle以及他们的子类的任意一个作为父类。通常可以选择QWindowsStyle或QMotifStyle，也可以选择QCommonStyle甚至是QStyle，但是工作量会比较大，因为很多界面的细节需要自己实现。重新实现必要的函数：想修改界面风格的哪部分，就重新实现与绘制那部分相关的函数，下面解释一下我们要重载的QStyle中的几个函数，这几个函数控制着图形用户界面上不同元素的布局 and 观感。

```
1) void drawPrimitive( PrimitiveElement pe,
QPainter *p, const QRect &cg, SFlags flags = Style_Default,
const QStyleOption &opt = QStyleOption::Default ) . 100Test
```

下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com