# Qt 之自定义界面（实现无边框、可移动）

# 简述

UI设计是指对软件的人机交互、操作逻辑、界面美观的整体设计。好的UI设计不仅是让软件变得有个性、有品位，还要让软件的操作变得舒适简单、自由，充分体现软件的定位和特点。

爱美之心人皆有之。其实软件界面就像工业造型一样，是产品的重要卖点。一个产品拥有美观的界面会给人带来舒适的视觉享受，拉近人与产品的距离，是建立在科学性之上的艺术设计。

检验一个界面的标准-既不是某个项目开发组领导的意见，也不是项目成员投票的结果，而是终端用户的感受。

所以，对于大多数商业软件或者专业产品来说，Qt的本地化样式显然已经不能满足期望，我们需要进行大量的样式美化，才能够让它更有个性、有品位。

下面我们来实现这样一个界面。

这里写图片描述

# 无边框

设置窗口标志为无边框

setWindowFlags(Qt::FramelessWindowHint | windowFlags());

关于WindowFlags枚举值很多，其中有一个WindowStaysOnTopHint，可以让窗体位于所有界面之上，类似于QQ主面板，一打开就在所有软件之上-比较霸道。

设置无边框之后，窗体是无法移动的，下面我们介绍两种方式来进行移动。

# 可移动

## Qt事件

*// 鼠标相对于窗体的位置 event->globalPos() - this->pos()*

void Widget::mousePressEvent(QMouseEvent \*event)

{

if (event->button() == Qt::LeftButton)

{

m\_bPressed = true;

m\_point = event->pos();

}

}

*// 若鼠标左键被按下，移动窗体位置*

void Widget::mouseMoveEvent(QMouseEvent \*event)

{

if (m\_bPressed)

move(event->pos() - m\_point + pos());

}

*// 设置鼠标未被按下*

void Widget::mouseReleaseEvent(QMouseEvent \*event)

{

Q\_UNUSED(event);

m\_bPressed = false;

}

## Win API

包含头文件与需要用到的库

#ifdef Q\_OS\_WIN

#pragma comment(lib, "user32.lib")

#include <qt\_windows.h>

#endif

发送系统消息，进行移动

void Widget::mousePressEvent(QMouseEvent \*event)

{

#ifdef Q\_OS\_WIN

if (ReleaseCapture())

SendMessage(HWND(winId()), WM\_SYSCOMMAND, SC\_MOVE + HTCAPTION, 0);

event->ignore();

#else

#endif

}