How to create a Text Editor in Symbian

This article shows how to add a rich text editor to your Symbian C++ application.

The example defines `CEditorContainer`, a simple container which you can use to add the editor into your project. To modify the font used with the editor you can select another font in the `CreateEditorL()` method instead of `LatinBold16()`. The font color can be set by changing the `KRgbBlack` constant used in the `TCharFormat`.

Note that using a rich text editor can cause wasted resources if you don't need different paragraph or character formatting. In these cases you should use CEikEdwin which will allow you to set the font and color for the whole text. Also you could use `CEikRichTextEditor` directly or derive from it rather than using a container within a container.

Link against:

```
avkon.lib eikcoctl.lib eikctl.lib form.lib uiklaf.lib
```

Header Required:

```
#include <eikrted.h>
```

Required capabilities:

```
None
```

Editor_Container.cpp

```
#include<Editor_container.h>
void CEditorContainer::ConstructL(void)
{
 CreateWindowL();

 SetRect(CEikonEnv::Static()->EikAppUi()->ClientRect());

 CreateEditorL();

 ActivateL();
}

void CEditorContainer::CreateEditorL(void)
{
 iEditor = new (ELeave) CEikRichTextEditor;
 iEditor->SetAknEditorFlags(EAknEditorFlagEnableScrollBars);
 iEditor->SetContainerWindowL(*this);
 iEditor->SetAvkonWrap(ETrue);
 iEditor->ConstructL(this,0,0,0);

 iEditor->SetFocus(ETrue);
 iEditor->EnableCcpuSupportL(ETrue);
 iEditor->SetCursorPosL(0, EFalse);
 iEditor->SetAvkonWrap(ETrue);
```


(C) Copyright Nokia 2014. All rights reserved.

```
 TFontSpec fontspec = LatinBold16()->FontSpecInTwips();
 TCharFormat charFormat( fontspec.iTypeface.iName, fontspec.iHeight );
 TCharFormatMask charFormatMask;

 charFormat.iFontPresentation.iTextColor = KRgbBlack;
 charFormatMask.SetAttrib(EAttColor);

 charFormatMask.SetAttrib(EAttFontTypeface);
 charFormatMask.SetAttrib(EAttFontHeight);
 iEditor->ApplyCharFormatL(charFormat, charFormatMask);

 iEditor->UpdateScrollBarsL();
 SizeChanged();
}


CEditorContainer::~CEditorContainer()
{
 delete iEditor;
}

void CEditorContainer::HandleResourceChange(TInt aType)
{
 TRect rect;

 if ( aType==KEikDynamicLayoutVariantSwitch )
 {
  AknLayoutUtils::LayoutMetricsRect( AknLayoutUtils::EMainPane, rect );
  SetRect(rect);
  }

 CCoeControl::HandleResourceChange(aType);
}

void CEditorContainer::SizeChanged()
{
 TRect rect = Rect();

 if(iEditor)
 {
 #ifdef __SERIES60_3X__
  TRect ScrollBarRect = iEditor->ScrollBarFrame()->VerticalScrollBar()->Rect();
  iEditor->SetExtent(TPoint(0,StatusLineHeight),
  TSize(rect.Width()-ScrollBarRect.Width(), rect.Height()-StatusLineHeight));
 #else
  iEditor->SetExtent(TPoint(0,StatusLineHeight),
  TSize(rect.Width(), rect.Height()-StatusLineHeight));
 #endif
 }
}

TInt CEditorContainer::CountComponentControls() const
{
 if(iEditor)
 {
  return 1;
 }
 else
 {
  return 0;
```


(C) Copyright Nokia 2014. All rights reserved.

```
 }
}

CCoeControl* CEditorContainer::ComponentControl(TInt /*aIndex*/) const
{
 return iEditor;
}

void CEditorContainer::Draw(const TRect& aRect) const
{
 CWindowGc& gc = SystemGc();
 gc.Clear();
}

TKeyResponse CEditorContainer::OfferKeyEventL(const TKeyEvent& aKeyEvent,
TEventCode aType)
{
 TKeyResponse Ret(EKeyWasNotConsumed);

 if (iEditor)
 {
  Ret = iEditor->OfferKeyEventL(aKeyEvent, aType);
 }

 return Ret;
}

void CEditorContainer::SetEditorTextL(const TDesC& aText)
{
 if (iEditor)
 {
  iEditor->ClearSelectionL();
  iEditor->RichText()->Reset();

  iEditor->RichText()->InsertL(0,aText);

iEditor->RichText()->InsertL(iEditor->RichText()->LdDocumentLength(),CEditableText::ELineBreak);

  iEditor->UpdateScrollBarsL();
  iEditor->DrawNow();
 }
}
```

Editor_Container.h

```
class CEditorContainer : public CCoeControl
    {
    public:
        void ConstructL(void);
        ~CEditorContainer();
        void SetEditorTextL(const TDesC& aText);
    public:
 TInt CountComponentControls() const;
        CCoeControl* ComponentControl(TInt aIndex) const;
     TKeyResponse OfferKeyEventL(const TKeyEvent& aKeyEvent,TEventCode aType);
    private:
     void CreateEditorL(void);
        virtual void SizeChanged();
```


(C) Copyright Nokia 2014. All rights reserved.

```
  virtual void HandleResourceChange(TInt aType);
         void Draw(const TRect& aRect) const;
     private:
   CEikRichTextEditor* iEditor;
     };
```


(C) Copyright Nokia 2014. All rights reserved.