



Following code snippets illustrate Listbox creation using resource. Here SingleStyle ListBox is used just for illustration.

Contents

- [1 Step 1: Supporting all necessary languages](#)
- [2 Step 2: Defining listbox in project resources](#)
- [3 Step 3: CYrContainer.h](#)
- [4 Step 4: CYrContainer.cpp](#)
- [5 Related Links](#)

Step 1: Supporting all necessary languages

File CYrRss.I01:

```
// English language support

#define lb_item1 "\tItem1\t"
#define lb_item2 "\tItem2\t"
#define lb_item3 "\tItem3\t"
#define lb_item4 "\tItem4\t"
#define lb_item5 "\tItem5\t"
#define lb_item6 "\tItem6\t"
#define lb_item7 "\tItem7\t"
#define lb_item8 "\tItem8\t"
#define lb_item9 "\tItem9\t"
```

File CYrRss.I16:

```
// Russian language support - russian text should be saved in UTF 8 !
CHARACTER_SET UTF8

#define lb_item1 "\t???????1\t"
#define lb_item2 "\t???????2\t"
#define lb_item3 "\t???????3\t"
#define lb_item4 "\t???????4\t"
#define lb_item5 "\t???????5\t"
#define lb_item6 "\t???????6\t"
#define lb_item7 "\t???????7\t"
#define lb_item8 "\t???????8\t"
#define lb_item9 "\t???????9\t"
```

Step 2: Defining listbox in project resources

File CYrRss.rss:

```
CHARACTER_SET UTF8

#ifdef LANGUAGE_01
#include "CYrRss.l01"
#endif

#ifdef LANGUAGE_16
#include "CYrRss.l16"
#endif

...

RESOURCE LISTBOX r_list_box
{
    flags = EAknListBoxSelectionList;
    array_id= r_list_box_item_array;
}

RESOURCE ARRAY r_list_box_item_array
{
    items =
    {
        LBUF { txt = lb_item1;},
        LBUF { txt = lb_item2;},
        LBUF { txt = lb_item3;},
        LBUF { txt = lb_item4;},
        LBUF { txt = lb_item5;},
        LBUF { txt = lb_item6;},
        LBUF { txt = lb_item7;},
        LBUF { txt = lb_item8;},
        LBUF { txt = lb_item9;},
    };
}
```

Step 3: CYrContainer.h

- Open your **CYrContainer.h** file.
- Include entries for required header files.

```
#include <aknlists.h> // For List Box
#include <barsread.h> //For Resource Reader
```

- Add function which will be needed for getting Key response.

```
TKeyResponse OfferKeyEventL(const TKeyEvent &aKeyEvent, TEventCode aType);
```

- Declare object of Listbox:

```
private: // data
// ..
CAknSingleStyleListBox* iListBox;
```

- Define in mmp-file library:

```
LIBRARY baf1.lib
```

Step 4: CYrContainer.cpp

- Open your "CyrContainer.cpp" file.
- Include an additional entry for .rsg file. Like:

```
#include <YrListBox.rsg>
```

- Now in the **ConstructL()** function of your **CYrContainer.cpp** file:

```
void CYrContainer::ConstructL(const TRect& aRect)
{
    CreateWindowL();

    iListBox= new (ELeave) CAknSingleStyleListBox();
    iListBox->SetContainerWindowL(*this);

    TResourceReader rr;
    iCoeEnv->CreateResourceReaderLC(rr, R_LIST_BOX);
    iListBox->ConstructFromResourceL(rr);
    iListBox->CreateScrollBarFrameL(ETrue);
    iListBox->ScrollBarFrame()->SetScrollBarVisibilityL(CEikScrollBarFrame::EOn, CEikScrollBarFra

    CleanupStack::PopAndDestroy();//rr

    SetRect(aRect);
    ActivateL();
}

void CYrContainer::SizeChanged()
{
    iListBox->SetExtent(TPoint(0,0), iListBox->MinimumSize());
}

TInt CYrContainer::CountComponentControls() const
{
    return 1; // return number of controls inside this container
}

CCoeControl* CYrContainer::ComponentControl(TInt aIndex) const
{
    switch ( aIndex )
    {
        {
            case 0:
                return iListBox;
            default:
                return NULL;
        }
    }
}

TKeyResponse CYrContainer::OfferKeyEventL(const TKeyEvent &aKeyEvent, TEventCode aType)
{
    return iListBox->OfferKeyEventL(aKeyEvent, aType);
}
```

Related Links

- [How to create a simple listbox](#)
 - [Querying selection with list](#)
 - [TSS000647 - Applying actions on list box items](#)
 - [How to add Marquee effect in ListBox](#)
 - [How to add ScrollBar in ListBox](#)
 - [How to add additional data to the list box?](#)
 - [List box example](#)
-