

列多窗口开发测试

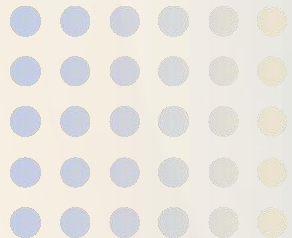
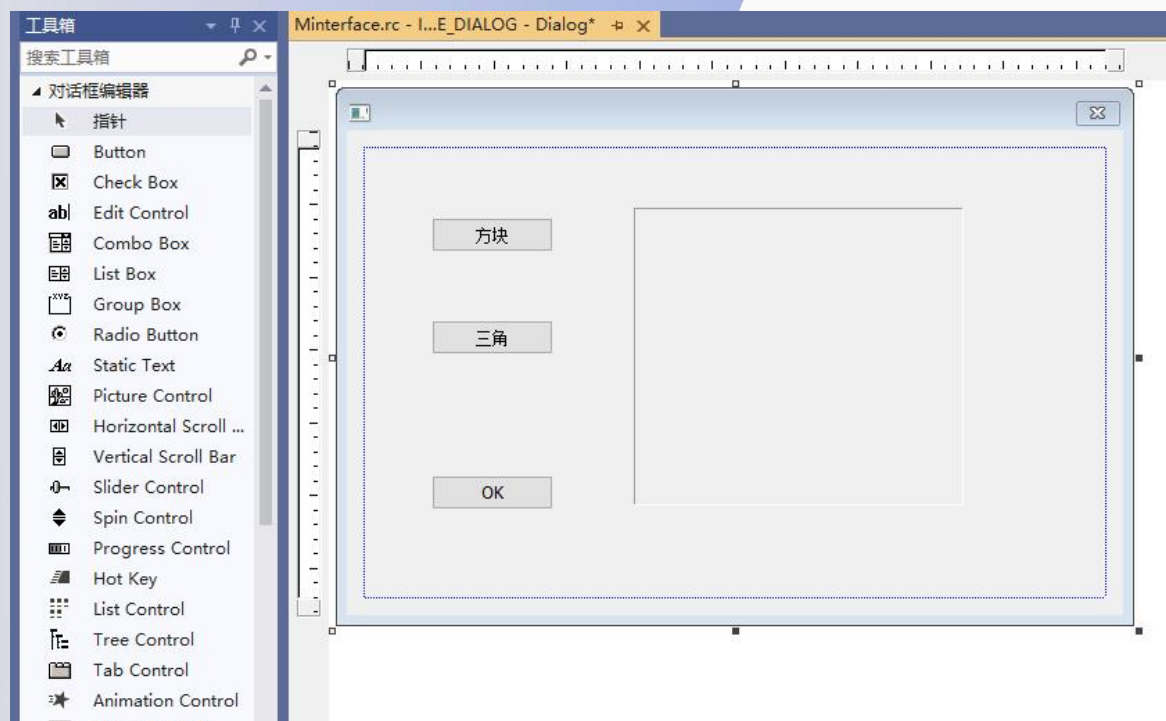


01

多窗口开发测试



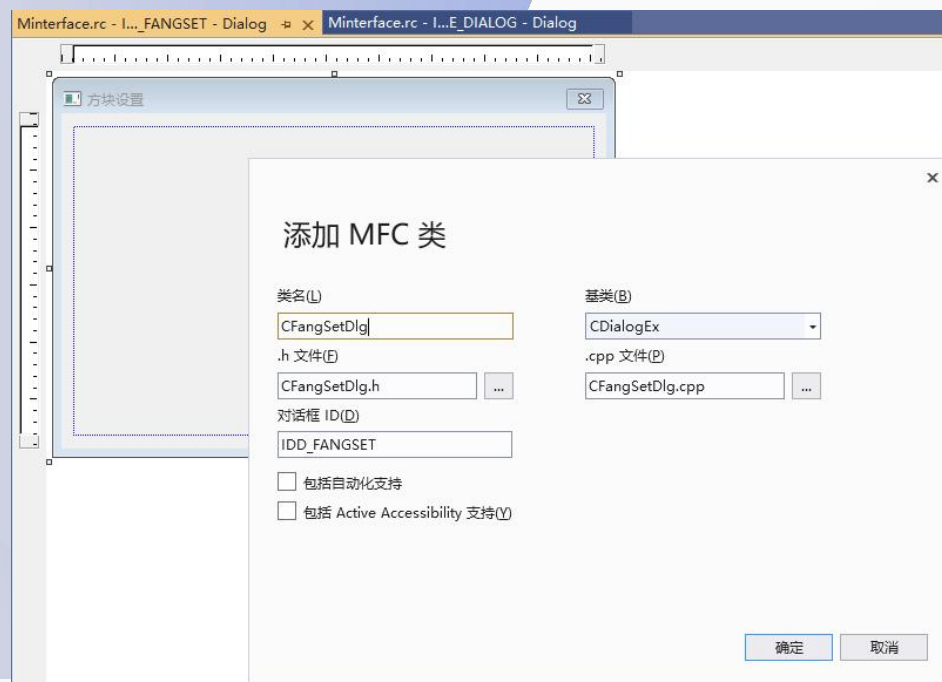
1. 打开vs2019，创建MFC项目“Minterface”为例。
2. 创建三个按钮和一个静态文本。三个按钮分别命名“方块”，ID更改为IDC_fang; “三角”，ID更改为IDC_san;、“OK”，ID更改为IDC_OK。静态文本清空，其属性里“静态边缘”设置为“true”，ID更改为IDC_DRAW。



多窗口开发测试



3. 在“资源视图”中单击鼠标右键选择“添加资源”。
4. 新建两个“Dialog”，把两个对话框的“确定”、“取消”都删掉，然后单击鼠标右键选择“属性”，ID分别更改为：IDD_FANGSET、IDD_SANSET；描述文字更改为：“方块设置”、“三角设置”。
5. 在方块设置界面单击鼠标右键选择“添加类”，添加MFC类，类名设置为“CFangSetDlg”。在三角设置界面进行同样的操作，类名设置为“CSanSetDlg”。



多窗口开发测试

6. 鼠标左键双击“资源视图”中“DD_MINTERFACE_DIALOG”回到主界面，接着左键双击“方块”按钮，添加如下代码。

```
void CMinterfaceDlg::OnBnClickedfang()  
{  
    // TODO: 在此添加控件通知处理程序代码  
    flag = 1;  
    CFangSetDlg dlg;  
    dlg.DoModal();//调出类所对应的对话框  
}
```

同样添加“三角”按钮代码。

```
void CMinterfaceDlg::OnBnClickedfang()  
{  
    // TODO: 在此添加控件通知处理程序代码  
    flag = 2;  
    CSanSetDlg dlg;  
    dlg.DoModal();//调出类所对应的对话框  
}
```

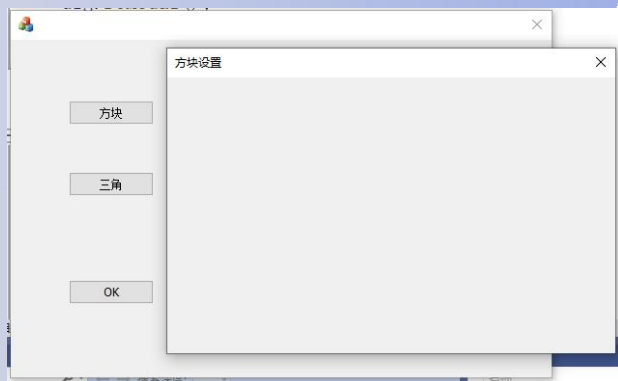
此时需要在“MinterfaceDlg.cpp”程序的上方对新建的MFC类进行声明，同时将flag定义为int型变量。



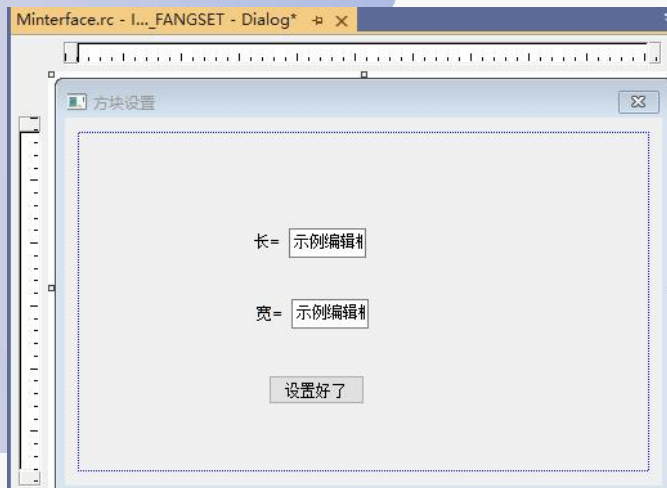
01

多窗口开发测试

编译生成无误后，开始调试，鼠标左键单击“方块”按钮时，对应的窗口会自动弹出。



7. 打开“方块设置”窗口，添加两个静态文本、两个示例编辑框和一个按钮。示例编辑框ID分别为IDC_EDIT_LENGTH、IDC_EDIT_WIDTH，按钮ID为IDC_FANGOK。



多窗口开发测试

鼠标左键双击“设置好了”，添加如下代码。

```
void CFangSetDlg::OnBnClickedFangok()
{
    // TODO: 在此添加控件通知处理程序代码
    extern double m_length, m_width;
    CString ch_length, ch_width;

    GetDlgItemText(IDC_EDIT_LENGTH, ch_length); //从窗口获取输入的字符串
    GetDlgItemText(IDC_EDIT_WIDTH, ch_width);

    m_length = _ttof(ch_length); //将字符串转换为方块长的数值
    m_width = _ttof(ch_width);
    CFangSetDlg::EndDialog(0);
}
```

新建StdAfx.h，将两个double变量m_length、m_width定义为全局变量。



多窗口开发测试

8. 打开“三角设置”窗口，添加三个静态文本、六个示例编辑框和一个按钮。示例编辑框ID分别为IDC_EDIT_AX、IDC_EDIT_AY、IDC_EDIT_BX、IDC_EDIT_BY、IDC_EDIT_CX、IDC_EDIT_CY，按钮ID为IDC_SANOK。鼠标左键双击“设置好了”，添加如下代码。

```
void CSanSetDlg::OnBnClickedSanok()
{
    // TODO: 在此添加控件通知处理程序代码
    extern double m_ax, m_ay, m_bx, m_by, m_cx, m_cy;
    CString ch_ax, ch_ay, ch_bx, ch_by, ch_cx, ch_cy;
    GetDlgItemText(IDC_EDIT_AX, ch_ax);
    GetDlgItemText(IDC_EDIT_AY, ch_ay);
    GetDlgItemText(IDC_EDIT_BX, ch_bx);
    GetDlgItemText(IDC_EDIT_BY, ch_by);
    GetDlgItemText(IDC_EDIT_CX, ch_cx);
    GetDlgItemText(IDC_EDIT_CY, ch_cy);
    m_ax = _ttof(ch_ax);
    m_ay = _ttof(ch_ay);
    m_bx = _ttof(ch_bx);
    m_by = _ttof(ch_by);
    m_cx = _ttof(ch_cx);
    m_cy = _ttof(ch_cy);
    CSanSetDlg::EndDialog(0);
}
```



9. 鼠标左键双击“资源视图”中“DD_MINTERFACE_DIALOG”回到主界面，接着左键双击“OK”按钮，添加如下代码。

```
void CInterfaceDlg::OnBnClickedOk()
{
    // TODO: 在此添加控件通知处理程序代码
    CWnd *pWnd = GetDlgItem(IDC_DRAW);
    CDC *pDC = pWnd->GetDC();
    CBrush WhiteBrush;
    WhiteBrush.CreateSolidBrush(RGB(255, 255, 255));
    CRect rect;
    pWnd->Invalidate();
    pWnd->UpdateWindow();
    pWnd->GetClientRect(&rect);
    switch (flag)
    {
```



多窗口开发测试

case 1: // fangkuai

```
pDC->MoveTo(20, 20);  
pDC->LineTo(20, 20 + m_width);  
pDC->MoveTo(20, 20 + m_width);  
pDC->LineTo(20 + m_length, 20 + m_width);  
pDC->MoveTo(20 + m_length, 20 + m_width);  
pDC->LineTo(20 + m_length, 20);  
pDC->MoveTo(20 + m_length, 20);  
pDC->LineTo(20, 20);  
break;
```

case 2: //sanjiao

```
pDC->MoveTo(m_ax, m_ay);  
pDC->LineTo(m_bx, m_by);  
pDC->MoveTo(m_bx, m_by);  
pDC->LineTo(m_cx, m_cy);  
pDC->MoveTo(m_cx, m_cy);  
pDC->LineTo(m_ax, m_ay);  
break;
```

default:

```
break;
```

```
}
```

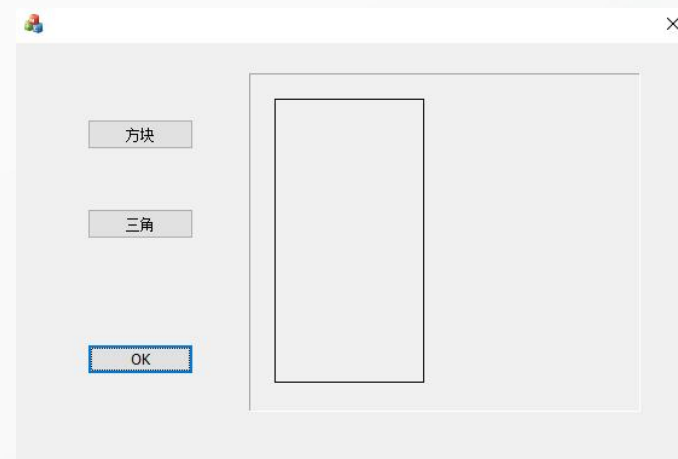
```
}
```



01

多窗口开发测试

编译生成无误后，开始调试，鼠标左键单击“方块”按钮时，对应的窗口会自动弹出，如左图所示。自定义长、宽的数值，单击设置好了，点击“OK”按钮，运行结果如右图所示。



谢谢观看

