# winform

## 窗体

### 创建

右键工程，”添加”->”windows窗体”。（假设Name为form2）

Form2 f=new Form2();

f.Show(); //第一次调用该函数时，窗体才会创建

f.Hide();

窗口必须在ui线程创建

ff.Show(); 非模式对话框，需要在ui线程创建

ff.ShowDialog(); 模式对话框，自带ui线程

### 屏蔽关闭按钮

override窗口的OnFormClosing事件

e.Cancel = true;

this.Hide();

可以使alt+F4与关闭按钮都失效，并隐藏窗体

### 屏蔽最大化按钮

this.MaximizeBox = false;

### 屏蔽所有系统按钮

this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.SizableToolWindow;

最大化、最小化、关闭按钮都消失，alt+f4失效，上面出现一个红色×用来隐藏窗口

## 定时器

资源中，所有windows窗体内，可以找到定时器控件，将控件加入到窗体。在定时器控件的属性设置中，Name代表定时器的名字（如timer1），Interval代表定时器的间隔时间（1000代表1s），在属性的事件里，可以设置定时器响应函数的名字，并且回车生成函数，也可以直接点击定时器，生成并进入函数。使用时，timer1.Start();开启定时器，timer1.Stop();取消定时器。

## 屏幕大小

//获取工作区大小

Rectangle rect = new Rectangle();

rect = Screen.GetWorkingArea(this);

///获取整个屏幕大小，如果需要全屏，设置ResizeMode为NoResize;

Rectangle rect = System.Windows.Forms.SystemInformation.VirtualScreen;

## 控件大小/位置

this.\*\*.Left = 0;

this.\*\*.Top = 0;

this.\*\*.Width = 1000;

this.\*\*.Height = 500;

## 消息

### 截获消息

protected override void DefWndProc(ref Message m)

{

switch (m.Msg)

{

case 0x0400 + 1:

break;

default:

base.DefWndProc(ref m);

break;

}

}

### 鼠标消息

string s = string.Format("{0},{1}", e.X, e.Y);

MessageBox.Show(s, "鼠标位置"); //弹出消息盒子

### 键盘消息

if (e.KeyCode == Keys.A && e.Control){ 同时按下control和a }

Keys类定义了所有的按键，对于control，alt这类的键，只能用e.Control这种布尔值进行判断，不能用KeyCode和Keys进行比较。

## 窗口设置

设置属性即可

### 顶层窗口

TopMost

### 去除标题和边框

WindowStyle none

ResizeMode NoResize

### 全屏

WindowState Maximized

### 设置窗口位置

this.Left = 0;

this.Top = 0;

this.Width = 1000;

this.Height = 500;

## 控件

### 线程中访问控件

delegate void SetTextBoxCallBack(string txt);

void SetTextBox(string txt){this.textBox1.Text = txt;}

线程中使用：

this.Invoke(new SetTextBoxCallBack(SetTextBox), new object[] { i.ToString() });

### Anchor

该属性可以让控件在保持与边界的位置，即设置了四个方向，则窗口拉伸时，控件会按照比例进行拉伸。

### 可见性

\*\*.Visibility = System.Windows.Visibility.Hidden;

### Button

可以在属性中设置图案

在消息响应函数中可以使用参数获取对象

((Button)sender).Text = "xx";

动态创建

Button btn = new Button();

btn.Text = "xx";

btn.Click += new EventHandler(btn\_Click);

Controls.Add(btn);

### Label

this.label1.Content= "xx";

label1是在窗体中的Label控件的名称（可以通过属性Name进行修改）

### TextBox

string s=this.textBox1.Text;

string s = "xx";

this.textBox1.AppendText(s);

### ListBox

listBox1.Items.Add(s);

listBox1.SelectedIndex = 1;

### listView

#### 创建

listView1.GridLines = true;

listView1.FullRowSelect = true;

listView1.View = View.Details;

listView1.Scrollable = true;

listView1.MultiSelect = true;

listView1.BackColor = Color.HotPink;//背景颜色

listView1.ForeColor = Color.Yellow;//文字颜色

//创建表头

listView1.Columns.Add("ID", 100, HorizontalAlignment.Center);

listView1.Columns.Add("Name", 100, HorizontalAlignment.Center);

//加入一行数据

ListViewItem item = new ListViewItem();

item.SubItems.Clear();

//ID列

item.UseItemStyleForSubItems = false;

item.SubItems[0].Text = "0";

item.SubItems[0].BackColor = Color.Red;

//Name列

item.SubItems.Add(new ListViewItem.ListViewSubItem());

item.SubItems[1].Text = "ABC";

item.SubItems[1].BackColor = Color.Green;

//数据加入控件

listView1.Items.Add(item);

//加入一行数据

listView1.Items.Add(new ListViewItem(new string[] { "1","DEF"}));

#### 删除

//删除第一行数据

listView1.Items[0].Remove();

//删除所有数据

while (listView1.Items.Count != 0)

{

listView1.Items[0].Remove();

}

### pictureBox

pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;

//pictureBox1.Load("D:\\1.jpg");

pictureBox1.Image = Bitmap ;

### progress

progressBar1.Value = 100;//最大位置为100

### menuStrip

点击栏目，可以键入文字，下拉可以选择键入类型（如分割线等）。

在键入时，&可以增加alt的热键。

每个栏目都有属性，可以在属性中ShortCutKeys中增加快捷键，在事件中增加click属性可以添加回调函数。

### 工具栏

#### ToolStrip

选择该控件，会在控件右上角出现一个小箭头，可以点击出一个菜单：选择插入标准项，可以插入打开，打印等标准的图标。选择编辑项，可以打开一个对话框，可以在其中为工具栏增加项目，包括按钮，文字等类型。

每个栏目都有属性，可以在属性中增加click事件。

#### toolStripContainer

dock属性设置为fill

this.toolStripContainer1.TopToolStripPanel.Controls.Add(this.toolStrip1);

Controls.Add(this.toolStripContainer1);

### statusStrip

在加入项目的对话框中，增加项目后，为Name属性填写合适的名称如abc。在程序中，则可以abc.text=”xx”;

## FLASH

工具栏，空白处右键，选择项，COM组件，Shockwave flash Object

播放

this.axShockwaveFlash1.LoadMovie(0, "d:\\1.swf");//必须使用全路径

接收命令

控件属性，FSCOMMAND

string s = e.command;

## MDI

假设主窗口Name属性为默认的Form1

1）设置主窗口IsMdiContain属性为true

2）创建新的winform窗口（form2），在代码中修改构造函数

public Form2(Form1 parent)

{

InitializeComponent();

this.MdiParent = parent;

}

3）使用

Form2 child = new Form2(this);

child.Show();