**c#(winform)中ComboBox和ListBox添加项 键值对 形式**

原创 2012年12月19日 10:33:53

* 7718

转载于：[http://blog.csdn.net/fcsh820/article/details/3867053](http://blog.csdn.net/fcsh820/article/details/3867053" \t "_blank)

感谢原作者分享这么好的文章。

WinForm下的ComboBox默认是以多行文本来设定显示列表的, 这通常不符合大家日常的应用,

因为大家日常应用通常是键/值对的形式去绑定它的.

那么用键值对的形式如何做?

因为Combox的每一个项的值是一个object, 实际上就是一个键/值对.  
我用的是下面这个类的实例作为它的一个项:

    /// <summary>  
    /// ComboBox的项  
    /// </summary>  
    class ListItem : System.Object  
    {  
        private string \_Value = string.Empty;  
        private string \_Text = string.Empty;  
  
        /// <summary>  
        /// 值  
        /// </summary>  
        public string Value  
        {  
            get { return this.\_Value; }

           set { this.\_Value=value; }  
        }  
        /// <summary>  
        /// 显示的文本  
        /// </summary>  
        public string Text  
        {  
            get { return this.\_Text; }

           set { this.\_Text=value; }  
        }  
  
        public ListItem(string value, string text)  
        {  
            this.\_Value = value;  
            this.\_Text = text;  
        }  
        public override string ToString()  
        {  
            return this.\_Text;  
        }  
  
    }

 通过这个类就可以定义ComboBox的值了, 首先我们定义一个ListItem的清单作为ComboBox的数据源:

            List<ListItem> items = new List<ListItem>();  
            items.Add(new ListItem("0", "Item\_0\_Text"));  
            items.Add(new ListItem("1", "Item\_1\_Text"));  
            items.Add(new ListItem("2", "Item\_2\_Text"));  
            items.Add(new ListItem("3", "Item\_3\_Text"));  
            items.Add(new ListItem("4", "Item\_4\_Text"));  
            items.Add(new ListItem("5", "Item\_5\_Text"));

 然后进行相应的设置:

            //将数据源的属性与ComboBox的属性对应  
            drpTest.DisplayMember = "Text";        //显示  
            drpTest.ValueMember = "Value";        //值

然后进就可以进行绑定了:

            drpTest.DataSource = items;        //绑定数据

绑定数据之后, 就可以对其进行默认选择项的设置, 取值等操作:

            drpTest.SelectedValue = "4";        //设定选择项  
  
            //取得当前选择的项  
            ListItem selectedItem = (ListItem)drpTest.SelectedItem;  
            string value = selectedItem.Value;    //值  
            string text = selectedItem.Text;    //显示的文字

////////////////////////////////////////////////////////////////////////////////////////////

根据原作者的文章。我想到了这个方法：

        private void LoadComboboxYear()

        {

            List<KeyValuePair<string, int>> listItem = new List<KeyValuePair<string, int>>();

            listItem.Add(new KeyValuePair<string, int>("2011年", 2011));

            listItem.Add(new KeyValuePair<string, int>("2012年", 2012));

            comboBox3.DataSource = listItem;

            comboBox3.DisplayMember = "Key";

            comboBox3.ValueMember = "Value";

            comboBox3.SelectedIndex = 0;

        }

取数据：

        private void button1\_Click(object sender, EventArgs e)

        {

            KeyValuePair<string, int> keyValue = (KeyValuePair<string,int>)comboBox3.SelectedItem;

            MessageBox.Show("key:" + keyValue.Key + ",value:" + keyValue.Value);

        }