

实验四 文件操作

一、实验目的

1. 掌握窗口控件的使用方法；
2. 掌握文件系统的操作方法、File 类和 Directory 类的使用。

二、实验要求

根据要求，编写 C# 程序，并将程序代码和运行结果写入实验报告。

三、实验内容

1. 编写一个简易的文件管理器，通过本次实验，练习 TreeView、ListView 和 SplitContainer 控件的使用，同时熟悉 C# 文件系统的操作方法以及 File 类和 Directory 类的使用。

(1)新建 windows 应用程序。

(2)将窗体 Form 的 Text 属性设置为“简易的文件管理器”。

(3)窗体上添加 SplitContainer、TreeView、ListView 控件，设置属性。将 TreeView 控件的 Dock 属性设置为 Left，ListView 控件的 Dock 属性设置为 Fill,MultiSelect 属性设置为 False，View 属性设置为 List。

(4)在 Form1.cs 中，添加对命名空间 using System.IO 的引用。

(5)在 Form1 的构造函数中添加代码：

```
public Form1()
{
    InitializeComponent();
    TreeNode myComputerNode = new TreeNode("我的电脑");
    treeView1.Nodes.Add(myComputerNode);
    listViewShow(myComputerNode);
}
```

(6)编写 ListView 控件的方法 listViewShow(TreeNode dirNode)。

```
private void listViewShow(TreeNode dirNode)
{
    listView1.Clear();
    if (dirNode.Parent == null)
    {
        foreach (string DrvName in Directory.GetLogicalDrives())
        {
            ListViewItem aItem = new ListViewItem(DrvName);
```

```

        listView1.Items.Add(aItem);
    }
}
else
{
    foreach (string DirName in
Directory.GetDirectories((string)dirNode.Tag))
    {
        ListViewItem aItem = new ListViewItem(DirName);
        listView1.Items.Add(aItem);
    }
    foreach (string fileName in Directory.GetFiles((string)
dirNode.Tag))
    {
        ListViewItem aItem=new ListViewItem(fileName);
        listView1.Items.Add(aItem);
    }
}
}

```

(7)编写 listViewShow 的重载方法。

```

private void listViewShow(string dirName)
{
    listView1.Clear();
    try
    {
        foreach (string DirName in Directory.GetDirectories(dirName))
        {
            ListViewItem aItem = new ListViewItem(DirName);
            listView1.Items.Add(aItem);
        }
        foreach (string fileName in Directory.GetFiles(dirName))
        {
            ListViewItem aItem = new ListViewItem(fileName);
            listView1.Items.Add(aItem);
        }
    }
    catch { }
}

```

(8)编写 DirTreeShow(TreeNode dirNode)方法。

```

private void DirTreeShow(TreeNode dirNode)
{
    try

```

```

    {
        if (dirNode.Nodes.Count == 0)
        {
            if (dirNode.Parent == null)
            {
                foreach (string DrvName in Directory.GetLogicalDrives())
                {
                    TreeNode aNode= new TreeNode(DrvName);
                    aNode.Tag=DrvName;
                    dirNode.Nodes.Add(aNode);
                }
            }
            else
            {
                foreach(string DirName in
Directory.GetDirectories((string)dirNode.Tag))
                {
                    TreeNode aNode=new TreeNode(DirName);
                    aNode.Tag=DirName;
                    dirNode.Nodes.Add(aNode);
                }
            }
        }
    }
    catch {}
}

```

(9)编写 TreeView 控件的 AfterSelect 事件代码。

```

private void treeView1_AfterSelect_1(object sender, TreeViewEventArgs e)
{
    listViewShow(e.Node);
    DirTreeShow(e.Node);
}

```

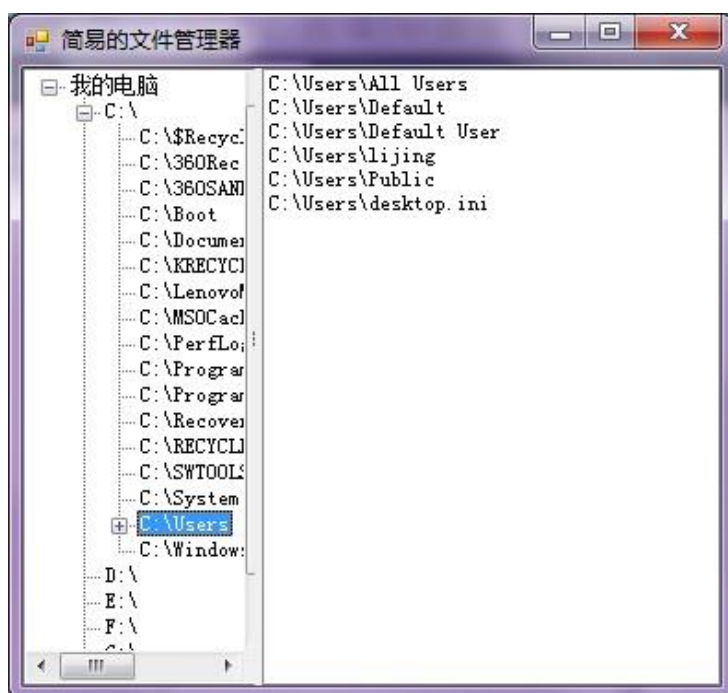
(10)编写 ListView 控件的 DoubleClick 事件代码。

```

private void listView1_DoubleClick(object sender, EventArgs e)
{
    foreach (int listIndex in listView1.SelectedIndices)
    {
        listViewShow(listView1.Items[listIndex].Text);
    }
}

```

(11)单击启动调试工具，运行结果参见下图。



注意：编写代码过程中，不可盲目照抄，仔细思考，加深内容的理解和使用。

四、实验总结

注：本部分写本次实验过程中出现的问题、如何解决、注意事项、以及自己的经验体会。