

## CHƯƠNG 4: FILE AND REGISTRY OPERATIONS

1

*NGUYỄN THỊ LƯƠNG*

KHOA CÔNG NGHỆ THÔNG TIN  
TRƯỜNG ĐẠI HỌC ĐÀ LẠT

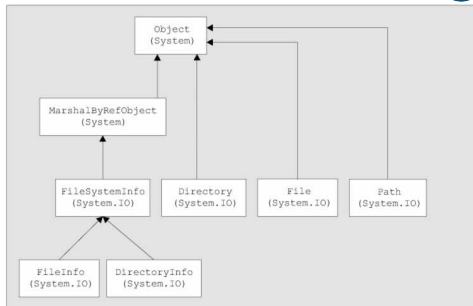
### Nội dung

2

- Quản lý tập tin (file)
- Di chuyển, sao chép, huỷ các file và folder
- Đọc và ghi văn bản trong các file
- Đọc và ghi các khoá trong Registry

## Quản lý tập tin (file)

3



- **System.MarshalByRefObject**  
– Lớp đối tượng cơ sở cho các lớp của .NET nó điều khiển từ xa; cho phép điều hành dữ liệu giữa các vùng ứng dụng.
- **FileSystemInfo**: Lớp đối tượng cơ sở biểu diễn thông tin của hệ thống file
- **FileInfo** and **File**: Các lớp này biểu diễn thông tin file
- **DirectoryInfo** and **Directory** – Các lớp này biểu diễn thông tin folder.
- **Path**: Lớp này chứa các thành phần tĩnh (Static) dùng quản lý đường dẫn.

## Quản lý tập tin (file) - FileSystemInfo

4

Thuộc tính	Mô tả
CreationTime	Thời gian tạo file, folder
DirectoryName (FileInfo), Parent (DirectoryInfo)	Tên đường dẫn thư mục chứa file, folder
Exists	Xác định file, folder đã tồn tại
Extension	Phần mở rộng của file nếu là folder trả về khoảng trắng
FullName	Tên đường dẫn file, folder
LastAccessTime	Thời gian file, folder truy xuất lần cuối
LastWriteTime	Thời gian file, folder được chỉnh sửa gần nhất
Name	Tên của file, folder
Root	Đường dẫn gốc
Length	Kích thước của file tính bằng bytes

## Quản lý tập tin (file) - FileSystemInfo

5

Phương thức	Mô tả
<b>Dùng cho cả FileInfo, DirectoryInfo</b>	
Create()	Tạo file, folder rỗng
Delete()	Hủy file, folder
MoveTo()	Di chuyển hoặc sửa tên file, folder
<b>Dùng cho FileInfo</b>	
CopyTo()	Sao chép file

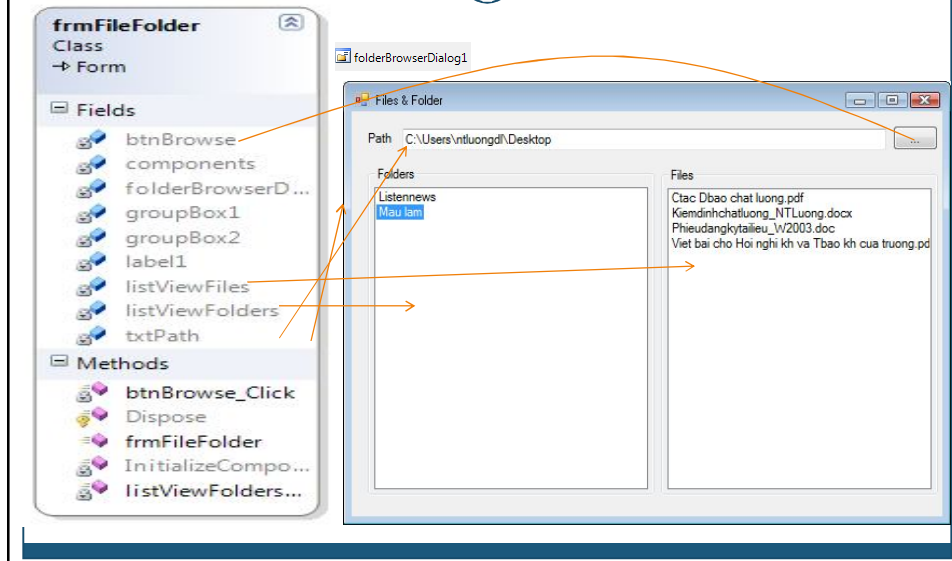
## Quản lý tập tin (file) - FileSystemInfo

6

Phương thức	Mô tả
<b>Dùng cho DirectoryInfo</b>	
GetDirectories()	Trả về một mảng các đối tượng của DirectoryInfo đại diện tất cả folders được chứa trong folder này.
GetFiles()	Trả về một mảng các đối tượng của FileInfo đại diện tất cả folders được chứa trong folder này
GetFileSystemObjects()	Trả về đối tượng FileInfo và DirectoryInfo như mảng của tham khảo FileSystemInfo .

## Ví dụ

7



## Ví dụ

8

```
private void btnBrowse_Click(object sender, EventArgs e)
{
    if (this.folderBrowserDialog1.ShowDialog() == DialogResult.OK)
    {
        this.txtPath.Text = this.folderBrowserDialog1.SelectedPath;
        DirectoryInfo currentdir = new DirectoryInfo(this.txtPath.Text);
        listViewFiles.Items.Clear();
        listViewFolders.Items.Clear();
        foreach (DirectoryInfo dir in currentdir.GetDirectories())
            listViewFolders.Items.Add(dir.Name);
    }
}

private void listViewFolders_SelectedIndexChanged(object sender, EventArgs e)
{
    if (this.listViewFolders.SelectedIndices.Count > 0)
    {
        ListViewItem lvitem = this.listViewFolders.SelectedItems[0];
        this.listViewFiles.Items.Clear();
        DirectoryInfo current = new
        DirectoryInfo(Path.Combine(this.txtPath.Text, lvitem.Text));
        foreach (FileInfo file in current.GetFiles())
            this.listViewFiles.Items.Add(file.Name);
    }
}
```

## Lớp Path

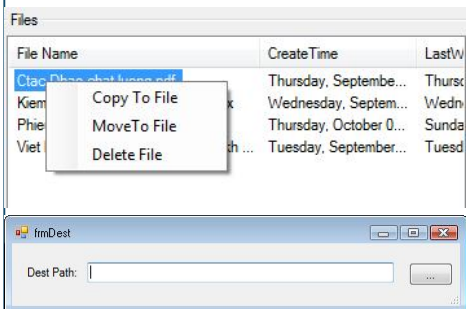
9

- Lớp Path trình bày các phương thức tĩnh để thực hiện các phép toán trên tên đường dẫn dễ dàng hơn
- Ví dụ:  
`Path.Combine(@"C:\My Documents", "ReadMe.txt");`

## Di chuyển, sao chép và hủy file hoặc folder

10

- Phương thức MoveTo, CopyTo, Delete



```
//filename: tên file
//destpath: đường dẫn đích

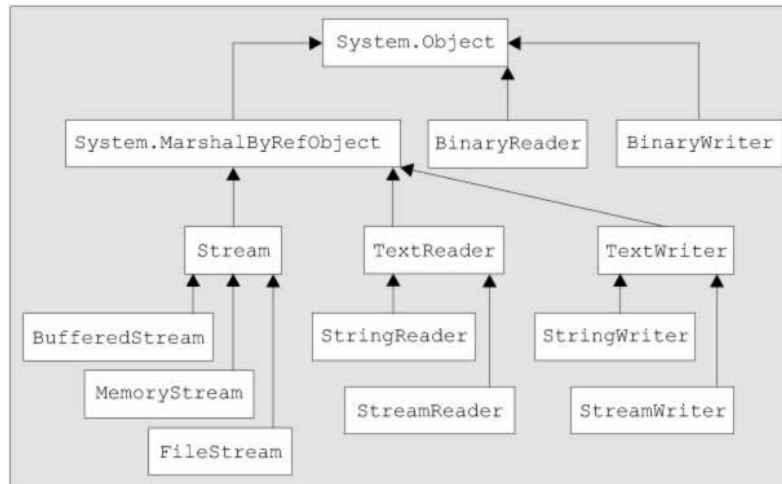
//Copy file
FileInfo fi = new FileInfo(filename);
fi.CopyTo(destpath);

//Move File
FileInfo fi = new FileInfo(filename);
fi.MoveTo(dest);

//Delete File
FileInfo fi = new FileInfo(filename);
fi.Delete();
```

## Namespace System.IO

11



## Binary Files

12

- `FileStream`
- Sử dụng `FileStream`

## FileStream

13

- Lớp FileStream cho phép mở những tập tin hiện hữu cũng như tạo mới file.
- Khi tạo tập tin lớp FileStream thường dùng những enum FileMode, FileAccess và FileShare
- FileStream được sử dụng đọc và viết dữ liệu vào hoặc từ một file.

## Sử dụng FileStream (tt)

14

- Khởi tạo 1 đối tượng FileStream cần 4 phần sau:
  - **file:** Tên file bạn muốn truy xuất .
  - **mode:** cho biết bạn muốn mở file như thế nào.
  - **access:** cho biết bạn muốn truy xuất file như thế nào – bạn định đọc hoặc viết file hoặc cả hai.
  - **share** access – khả năng truy xuất file.

Enumeration	Values
FileMode	Append, Create, CreateNew, Open, OpenOrCreate, or Truncate
FileAccess	Read, ReadWrite, or Write
FileShare	Inheritable, None, Read, ReadWrite, or Write

## Sử dụng FileStream (tt)

15

- Sử dụng BufferedStream :

- Buffered stream cho phép hệ điều hành tạo buffer riêng, rồi đọc dữ liệu vào hoặc viết dữ liệu lên ổ đĩa.

```
FileStream inputstream = File.OpenRead(@"C:\YServer.txt");
FileStream outputstream = File.OpenWrite(@"C:\thu.txt");
BufferedStream bufInput = new BufferedStream(inputstream);
BufferedStream bufOutput = new
BufferedStream(outputstream);
byte[] buffer= new byte[100];
int bytesRead;
while ((bytesRead = bufInput.Read(buffer, 0, 50)) > 0)
    bufOutput.Write(buffer, 0, bytesRead);
bufOutput.Close();
bufInput.Close();
inputstream.Close();
outputstream.Close();
```

## Sử dụng FileStream (tt)

16

- Sử dụng BinaryWriter:

```
private void WriteReadFile(string filename)
{
    if (File.Exists(filename))
        return;
    //Ghi dữ liệu sử dụng BinaryWriter
    FileStream fs = new FileStream(filename, FileMode.CreateNew);
    BinaryWriter w = new BinaryWriter(fs);
    for (int i = 0; i < 11; i++)
        w.Write((int)i);
    w.Close();
    fs.Close();
    //Đọc dữ liệu sử dụng BinaryReader
    fs = new FileStream(filename, FileMode.Open, FileAccess.Read);
    BinaryReader r = new BinaryReader(fs);
    string s = null;
    for (int i = 0; i < 11; i++)
        s+=r.ReadInt32().ToString()+"\t";
    MessageBox.Show("Du lieu doc:" + s);
    r.Close();
    fs.Close();
}
```



## Text File

17

- **StreamReader:**

- Lớp StreamReader được dẫn xuất từ một lớp abstract mang tên TextReader cũng giống như String Reader.
- Dùng để đọc và peek lên một dòng ký tự (character stream) từ 1 văn bản

- **StreamWriter:**

- Lớp StreamWriter và StringWriter cũng được dẫn xuất từ một lớp abstract mang tên TextWriter.
- Dùng để viết những dữ liệu lên văn bản

## StreamWriter

18

- **Phương thức:**

- Write(object): Ghi dữ liệu xuống file
- WriteLine(object);
- Close(): Đóng file
- Khai báo và khởi tạo đối tượng:  
`StreamWriter sr = new StreamWriter(TenFile);`

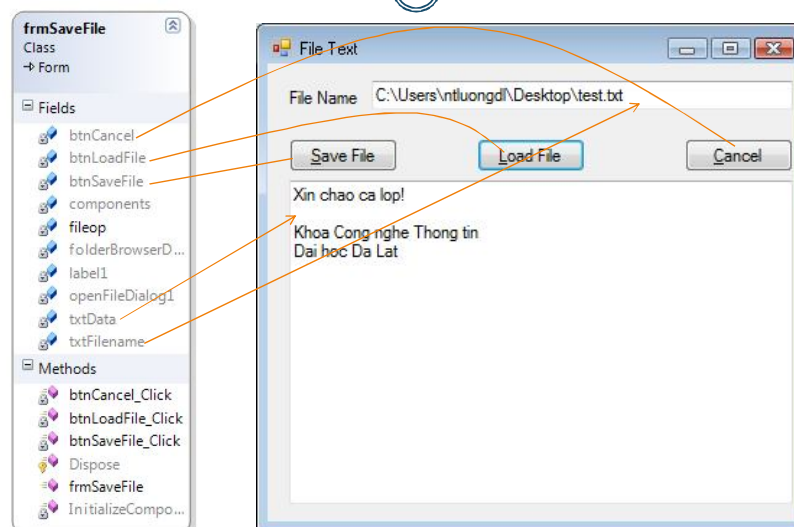
## StreamReader

19

- Đọc dữ liệu từ File
  - Khai báo và khởi tạo:  
`StreamReader sd = new StreamReader("D:\\data2.txt");`
  - `File.Exists("D:\\data3.txt");`
  - `String ReadLine():` Đọc 1 dòng
  - `String ReadToEnd():` Đọc toàn bộ nội File
  - `Close():` đóng file.

## Ví dụ

20



## Ví dụ

21

```

public int WriteFileText(string name, string text)
{
    if (File.Exists(name))
        return 0;
    FileStream fs = new FileStream(name, FileMode.CreateNew);
    StreamWriter sw = new StreamWriter(fs);
    sw.Write(text);
    sw.Close();
    fs.Close();
    return 1;
}

public string ReadFileText(string name)
{
    string str = null;
    if (File.Exists(name))
    {
        FileStream fs = new FileStream(name, FileMode.Open);
        StreamReader sw = new StreamReader(fs);
        str=sw.ReadToEnd();
        sw.Close();
        fs.Close();
    }
    return str;
}

```

## Ví dụ

22

```

private void btnSaveFile_Click(object sender, EventArgs e)
{
    string str = null;
    if (this.txtFilename.Text.Length > 0)
    {
        if (folderBrowserDialog1.ShowDialog() == DialogResult.OK)
        {
            str = Path.Combine(folderBrowserDialog1.SelectedPath,
                this.txtFilename.Text);
            fileop.WriteFileText(str, this.txtData.Text);
        }
    }
    else MessageBox.Show("Nhập tên file");
}

private void btnLoadFile_Click(object sender, EventArgs e)
{
    string str = null;
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        str = openFileDialog1.FileName;
        this.txtFilename.Text = str;
        this.txtData.Text = fileop.ReadFileText(str);
    }
}

```

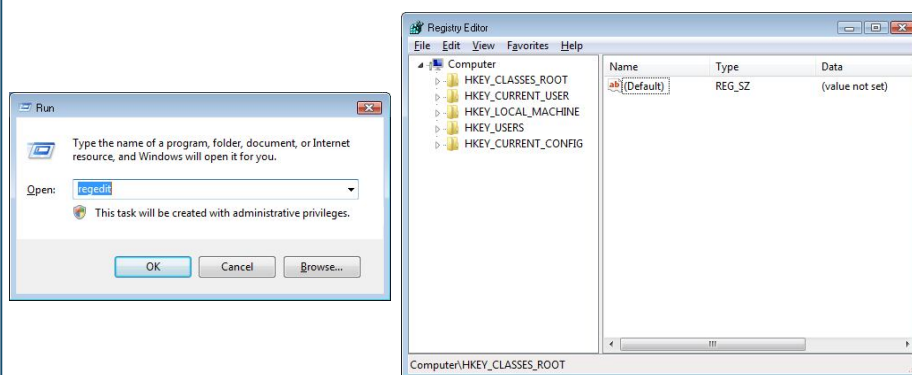
## Thao tác với Registry

23

- Giới thiệu
- The .NET Registry Classes

## Giới thiệu Registry

24



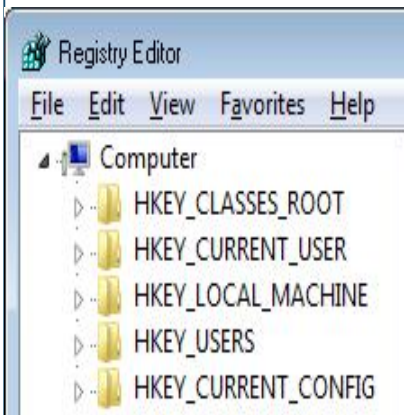
## The .NET Registry Classes

25

- Sử dụng:
  - Registry
  - RegistryKey

## Registry

26



- Trả về đối tượng `RegistryKey`

```
Registry.ClassesRoot;  
Registry.CurrentUser;  
Registry.LocalMachine;  
Registry.Users;  
Registry.CurrentConfig;
```

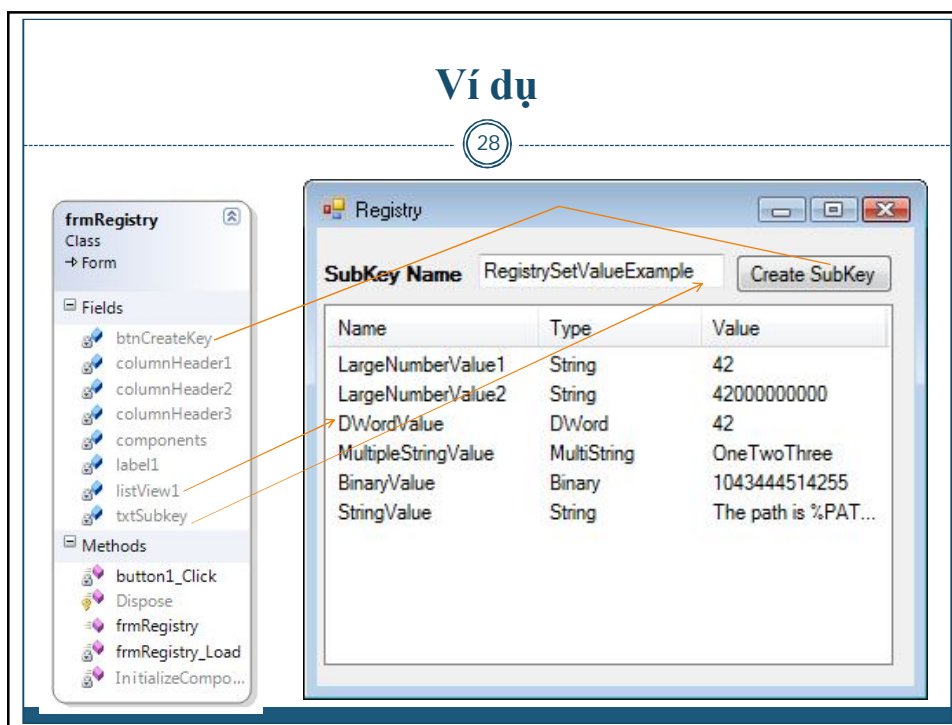
## RegistryKey

27

Phương thức	Mô tả
CreateSubKey	Tạo SubKey
DeleteSubKey	Xóa 1 SubKey
DeleteSubKeyTree	Xóa 1 SubKeyTree
DeleteValue	Xóa giá trị của Key
OpenSubKey	Trả về 1 SubKey

## Ví dụ

28



```

if (this.txtSubkey.Text.Length > 0)
{
    // Delete and recreate the test key.
    Registry.CurrentUser.DeleteSubKey(this.txtSubkey.Text, false);
    RegistryKey rk =
        Registry.CurrentUser.CreateSubKey(this.txtSubkey.Text);
    // Create name/value pairs.
    rk.SetValue("LargeNumberValue1", (long)42);
    rk.SetValue("LargeNumberValue2", 420000000000);
    rk.SetValue("DWordValue", 42);
    rk.SetValue("MultipleStringValue", new string[] { "One", "Two",
        "Three" });
    rk.SetValue("BinaryValue", new byte[] { 10, 43, 44, 45, 14, 255 });
    rk.SetValue("StringValue", "The path is %PATH%");
    string[] valueNames = rk.GetValueNames();
    ListViewItem lv;
    string str=null;
    foreach (string s in valueNames)
    {
        RegistryValueKind rvk = rk.GetValueKind(s);
        switch (rvk)
        {
            case RegistryValueKind.MultiString:
                string[] values = (string[])rk.GetValue(s);
                lv = new ListViewItem(s);
                lv.SubItems.Add(rvk.ToString());
                str=values[0]+"\\t";
                for (int i = 1; i < values.Length; i++)
                    str+=values[i]+"\\t";
                lv.SubItems.Add(str);
                this.listView1.Items.Add(lv);
                break;
            case RegistryValueKind.Binary:
                byte[] bytes = (byte[])rk.GetValue(s);
                lv = new ListViewItem(s);
                lv.SubItems.Add(rvk.ToString());
                str = rvk + "\\t";
                str= bytes[0]+"\\t";
                for (int i = 1; i < bytes.Length; i++)
                    str+= bytes[i]+"\\t";
                lv.SubItems.Add(str);
                this.listView1.Items.Add(lv);
                break;
            default:
                lv = new ListViewItem(s);
                lv.SubItems.Add(rvk.ToString());
                lv.SubItems.Add(rk.GetValue(s).ToString());
                this.listView1.Items.Add(lv);
                break;
        }
    }
}

```

## KẾT THÚC CHƯƠNG

30

# Q&A