

C# 系统应用之获取IE浏览记录和IE地址栏输入网址

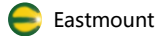
原创 Eastmount 最后发布于2014-04-03 02:16:40 阅读数 6173 ☆ 收藏

展开



Python+TensorFlow人工智能

该专栏为人工智能入门专栏，采用Python3和TensorFlow实现人工智能相关算法。前期介绍安装流程、基础语法...



¥9.90

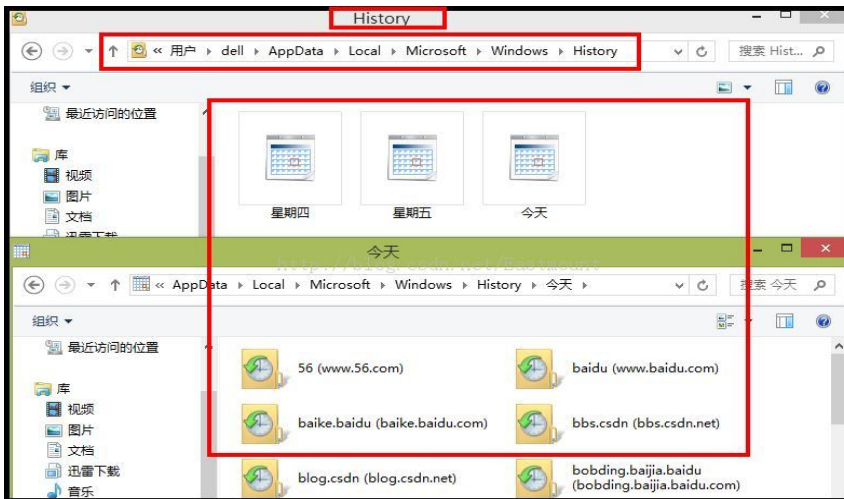
去订阅

该文章是“个人电脑历史记录清除软件”项目的系统应用系列文章。

前面“C# 系统应用之清除Cookies、IE临时文件、历史记录”中已经讲述了借助RunDll32.exe运行dll文件实现清除IE缓存操作,同时网上有很多方法讲述删除操作的,但怎样获取IE浏览器中最近浏览的网站历史记录和IE浏览器的地址栏输入的网址呢?这是我这篇文章需要讲解的知识。

一.Environment.GetFolderPath方法获取IE历史记录

前文说过在Windows中IE历史记录的位置为:"C:\Users\ dell\AppData\Local\Microsoft\Windows\History",历史记录是存放最近时间访问过的网站地址,它以时间和站点存储.如下图所示:



那么,我们可以通过Environment.GetFolderPath(Environment.SpecialFolder.History)获取IE历史记录。

```
private void button3_Click(object sender, EventArgs e)
{
    //清空listBox
    listBox1.Items.Clear();
    //获取Internet历史记录文件路径
    string dirPath = Environment.GetFolderPath(Environment.SpecialFolder.History);
    listBox1.Items.Add("Internet历史记录路径:");
    listBox1.Items.Add(dirPath);

    //遍历所有的文件夹 显示所有文件
    DirectoryInfo dir = new DirectoryInfo(dirPath);
    int num = 1;
    foreach (FileInfo file in dir.GetFiles("*.htm", SearchOption.AllDirectories))
    {
        try
        {
            listBox1.Items.Add("(" + num + ") " + file);
            num++;
        }
        catch (Exception msg) //异常处理
        {
        }
    }
}
```

```

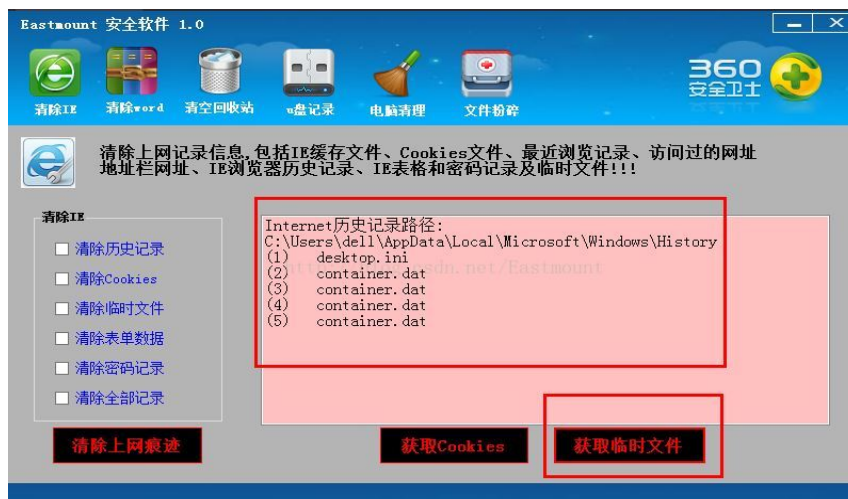
        MessageBox.Show(msg.Message);
    }
}

```

其中Environment.GetFolderPath(Environment.SpecialFolder.XXX)用于检索系统特殊文件夹的路径,常见有:

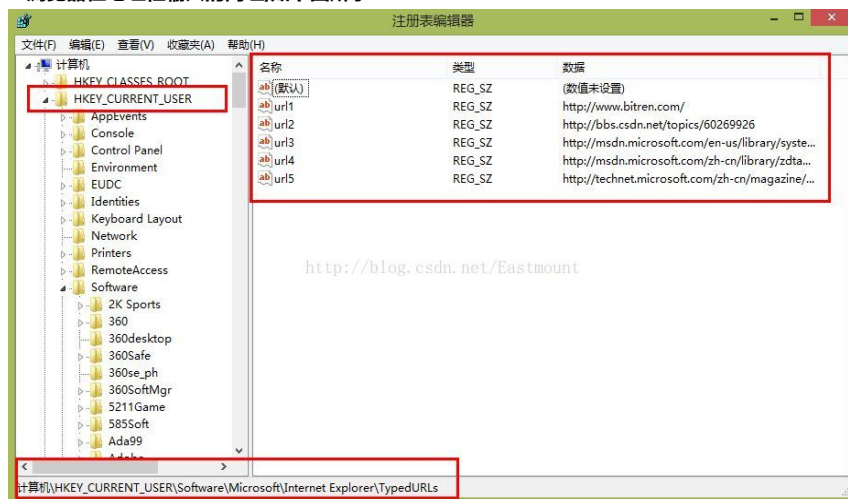
- (1).History 用作 Internet 历史记录项的公共储存库的目录
- (2).Cookies 用作 Internet Cookie 的公共储存库的目录
- (3).InternetCache 用作 Internet 临时文件的公共储存库的目录
- (4).Recent 包含用户最近使用过的文档的目录
- (5).MyPictures “My Pictures” 文件夹
- (6).MyDocuments “我的电脑” 文件夹
- (7).ProgramFiles “Program files” 目录

但是运行它进行删除delete操作或获取操作时,它常常会有系统文件,很多文件不能访问到.在删除中就会遇到"该文件正在另一进程使用,因此该进程无法访问此文件"或"文件访问被拒绝".而采用RunDll32.exe调用Win32 API函数ShellExecute()执行实现的.同样,使用它获取得到的历史记录如下图所示.显然不是我想要的结果,同时在清除Cookies后还是能显示很多Cookies信息.猜测是Google、360等浏览器的?

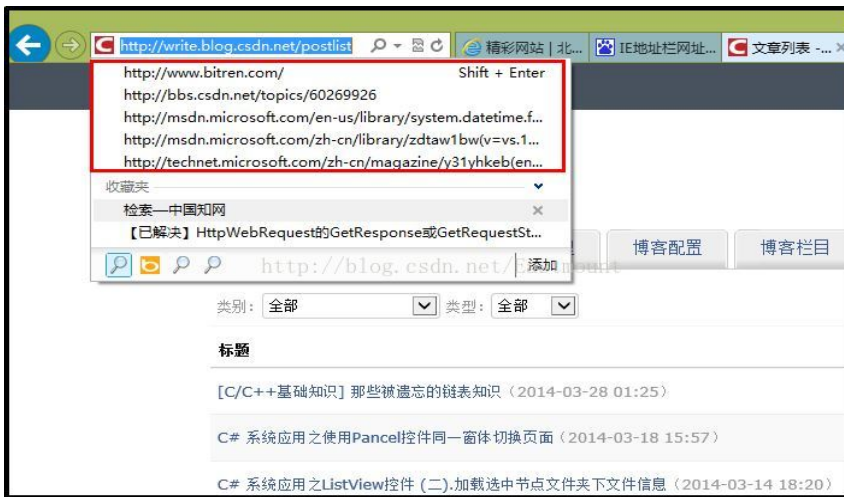


二.RegistryKey注册表获取IE地址栏网址

打开运行(Ctrl+R)并输入"regedit"打开注册表,在"HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\TypedURLs"下可以看见最近使用IE浏览器在地址栏输入的网址,如下图所示:



其中对应的IE浏览器如下图所示:



通过下面的代码,借助RegistryKey注册表顶级节点和获取该路径下的文件,即可显示相应的显示IE浏览器的地址栏网址.注意RegistryKey需要引用命名空间 using Microsoft.Win32.

```

/// <summary>
/// 获取IE地址栏输入网址
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private void button2_Click(object sender, EventArgs e)
{
    // 定义注册表顶级节点 其命名空间是using Microsoft.Win32;
    RegistryKey historykey;
    // 检索当前用户CurrentUser子项Software\Microsoft\Internet Explorer\typedURLs
    historykey = Registry.CurrentUser.OpenSubKey("Software\\Microsoft\\Internet Explorer\\typedURLs", true);
    if (historykey != null)
    {
        // 获取检索的所有值
        String[] names = historykey.GetValueNames();
        foreach (String str in names)
        {
            listBox1.Items.Add(historykey.GetValue(str).ToString());
        }
    }
    else
    {
        MessageBox.Show(this, "IE地址栏没有要删除的网址", "提示对话框", MessageBoxButtons.OK,
        MessageBoxIcon.Warning);
    }
}

```

运行结果如下图所示:



三.COM接口IUrlHistoryStg2获取IE浏览记录

这里主要通过王集鹤老师讲述的方法,通过IE提供的COM接口实现的.在此感谢他的文章.

参考文章:<http://bbs.csdn.net/topics/290070046>代码如下:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Runtime.InteropServices; // 命名空间
using System.Reflection;           // 提供加载类型 Pointer 指针
using Microsoft.Win32;             // RegistryKey

namespace GetIE
{
    #region COM接口实现获取IE历史记录
    // 自定义结构 IUrlHistory
    public struct STATURL
    {
        public static uint SIZEOF_STATURL =
            (uint)Marshal.SizeOf(typeof(STATURL));
        public uint cbSize;           // 网页大小
        [MarshalAs(UnmanagedType.LPWStr)] // 网页Url
        public string pwcsUrl;
        [MarshalAs(UnmanagedType.LPWStr)] // 网页标题
        public string pwcsTitle;
        public System.Runtime.InteropServices.ComTypes.FILETIME
            ftLastVisited,           // 网页最近访问时间
            ftLastUpdated,          // 网页最近更新时间
            ftExpires;
        public uint dwFlags;
    }

    // ComImport 属性通过guid调用com组件
    [ComImport, Guid("3C374A42-BAE4-11CF-BF7D-00AA006946EE"),
        InterfaceType(ComInterfaceType.InterfaceIsIUnknown)]
    interface IEnumSTATURL
    {
    }
```

```

[PreserveSig]
    // 搜索IE历史记录匹配的搜索模式并复制到指定缓冲区
    uint Next(uint celt, out STATURL rgelt, out uint pceltFetched);
    void Skip(uint celt);
    void Reset();
    void Clone(out IEnumSTATURL ppenum);
    void SetFilter(
        [MarshalAs(UnmanagedType.LPWStr)] string pszFilter,
        uint dwFlags);
}

[ComImport, Guid("AFA0DC11-C313-11d0-831A-00C04FD5AE38"),
    InterfaceType(ComInterfaceType.InterfaceIsIUnknown)]
interface IUrlHistoryStg2
{
    #region IUrlHistoryStg methods
    void AddUrl(
        [MarshalAs(UnmanagedType.LPWStr)] string pocsUrl,
        [MarshalAs(UnmanagedType.LPWStr)] string pocsTitle,
        uint dwFlags);

    void DeleteUrl(
        [MarshalAs(UnmanagedType.LPWStr)] string pocsUrl,
        uint dwFlags);

    void QueryUrl(
        [MarshalAs(UnmanagedType.LPWStr)] string pocsUrl,
        uint dwFlags,
        ref STATURL lpSTATURL);

    void BindToObject(
        [MarshalAs(UnmanagedType.LPWStr)] string pocsUrl,
        ref Guid riid,
        [MarshalAs(UnmanagedType.IUnknown)] out object ppvOut);

    IEnumSTATURL EnumUrls();
    #endregion

    void AddUrlAndNotify(
        [MarshalAs(UnmanagedType.LPWStr)] string pocsUrl,
        [MarshalAs(UnmanagedType.LPWStr)] string pocsTitle,
        uint dwFlags,
        [MarshalAs(UnmanagedType.Bool)] bool fWriteHistory,
        [MarshalAs(UnmanagedType.IUnknown)] object /*IOleCommandTarget*/
        poctNotify,
        [MarshalAs(UnmanagedType.IUnknown)] object punkISFolder);

    void ClearHistory(); // 清除历史记录
}

[ComImport, Guid("3C374A40-BAE4-11CF-BF7D-00AA006946EE")]
class UrlHistory /* : IUrlHistoryStg[2] */ { }
#endregion

// 调用COM接口IUrlHistory方法实现
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }
}

```

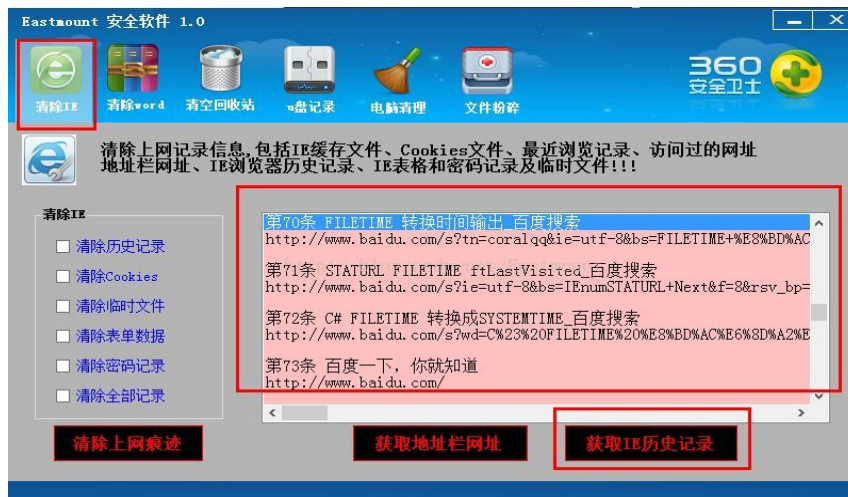
```

private void button1_Click(object sender, EventArgs e)
{
    IUrlHistoryStg2 vUrlHistoryStg2 = (IUrlHistoryStg2)new UrlHistory();
    IEnumSTATURL vEnumSTATURL = vUrlHistoryStg2.EnumUrls();
    STATURL vSTATURL;
    uint vFetched;

    while (vEnumSTATURL.Next(1, out vSTATURL, out vFetched) == 0)
    {
        richTextBox1.AppendText(string.Format("{0}\r\n{1}\r\n", vSTATURL.pwcsTitle,
vSTATURL.pwcsUrl));
    }
}
}

```

经过我的处理后运行结果如下图所示:



这里也提供他的一篇文章,采用另外一种方法调用IE的API函数实现 http://blog.sina.com.cn/s/blog_589d32f5010007xf.html

但是我也遇到了一个问题,就是它的ftLastVisited(The last time the user visited this page)存储该网页最后访问时间,想通过该时间进行获取今天访问的或排序输出前100.但是获取该时间时总输出错误,同时将FILETIME转换成SYSTEMTIME或time_t都没成功.希望以后能解决.

最后该文章主要是结合自己的实际东西讲解,如果你刚好遇到类似的问题就可能对你有所帮助.同时如果在文章中遇到错误或不足的地方,请海涵!最重要的是感谢上面提到的博主.希望能把获取时间等问题也解决.请尊重作者的劳动果实,勿喷!!!

(By:Eastmount 2014-4-3 夜2点半 原创CSDN<http://blog.csdn.net/eastmount/>)

👍 点赞 2 ☆ 收藏 📄 分享 ...



Eastmount 博客专家

发布了450 篇原创文章 · 获赞 6227 · 访问量 496万+

他的留言板

关注