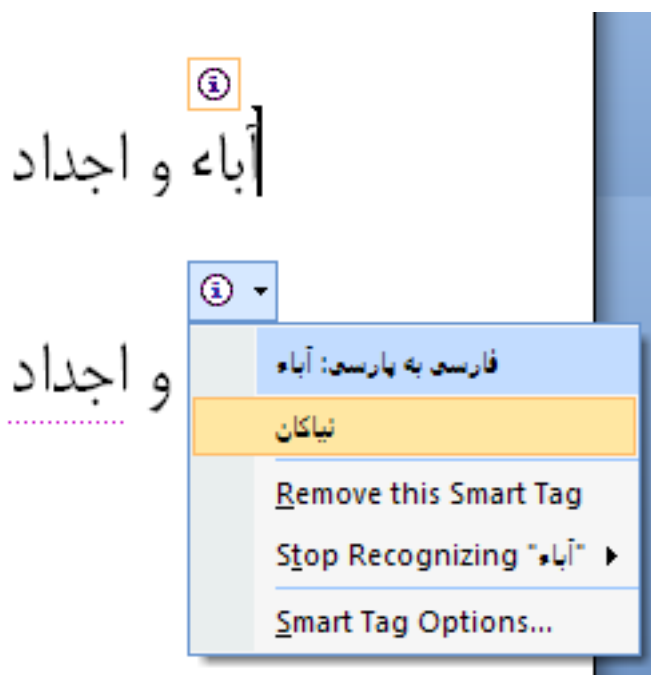


افزونه‌ی زیر کار مشخص کردن کلماتی که قابل تبدیل از فارسی به فارسی هستند را انجام می‌دهد (به صورت خودکار و در زمان تایپ) و همچنین امکان تبدیل خودکار آنها را نیز فراهم می‌کند.

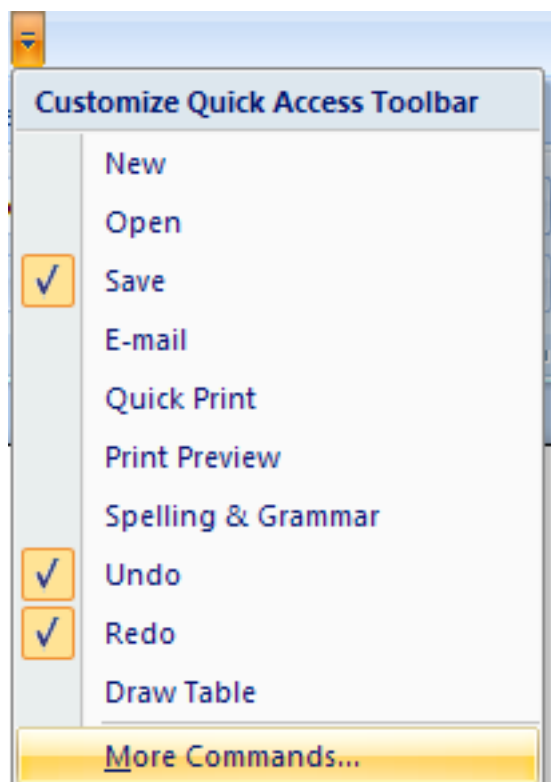


برای نصب آن باید به ترتیب زیر عمل کنید (مهم)  
[نصب دات نت فریم ورک 3 و نیم](#) (نزدیک به 231 مگابایت)  
اگر VS.Net 2008 SP1 را دارید نیازی به این بسته نخواهید داشت.

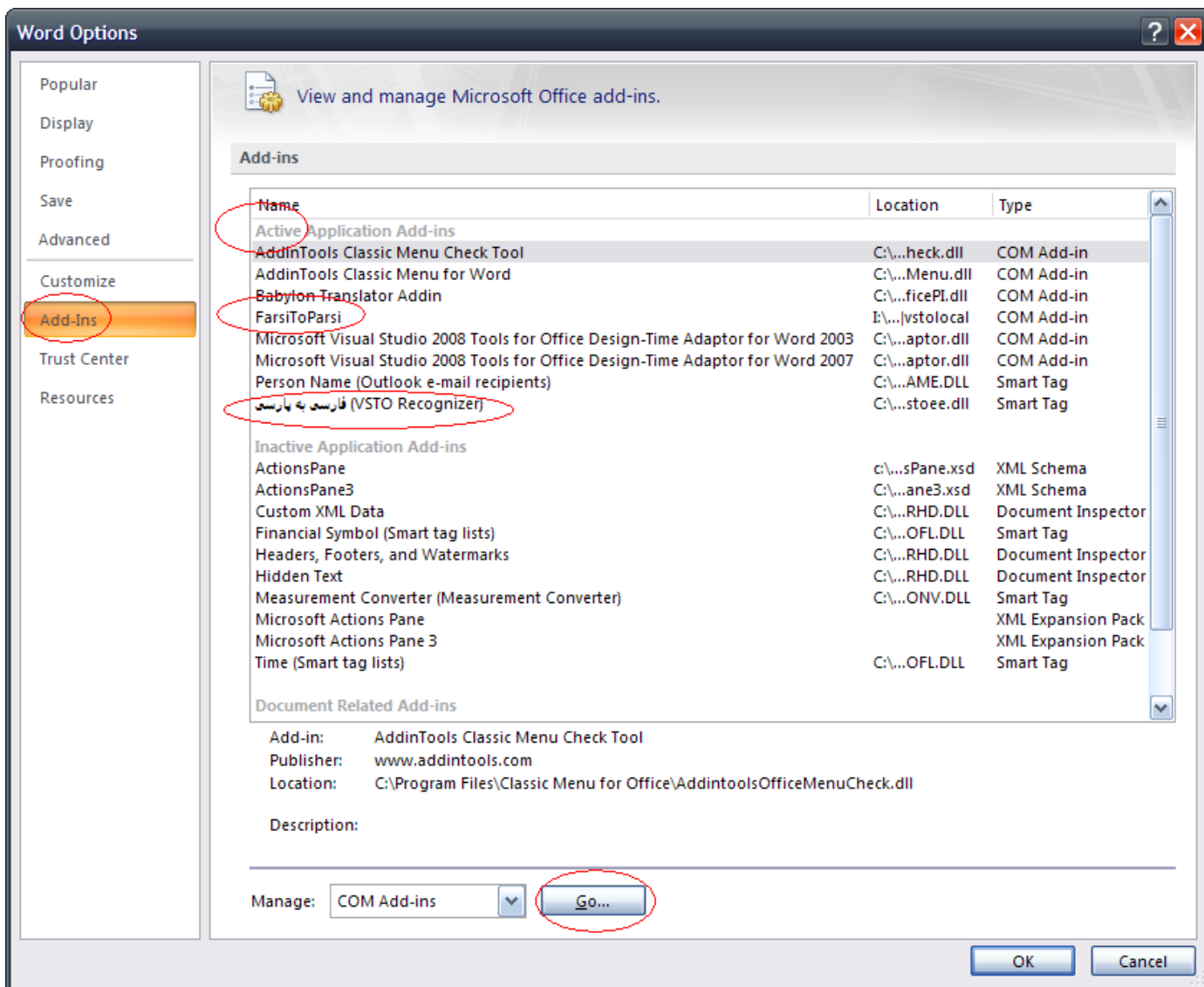
[نصب بسته به روز رسانی سیستم](#) (حدودا 14 مگ)  
[نصب افزونه](#) (حدودا 550 کیلوبایت)

لطفا این سه مرحله را به ترتیب انجام دهید در غیر اینصورت نتیجه نخواهید گرفت.  
همچنین بدیهی است که برنامه باید با دسترسی admin نصب شود.  
و همانطور که در عنوان این موضوع نیز ذکر گردید، این افزونه تنها برای MS-Word 2007 طراحی شده است.  
سورس کامل و جزئیات نحوه‌ی برنامه نویسی آن را در طی روزهای آینده در این بلاگ مشاهده خواهید کرد.

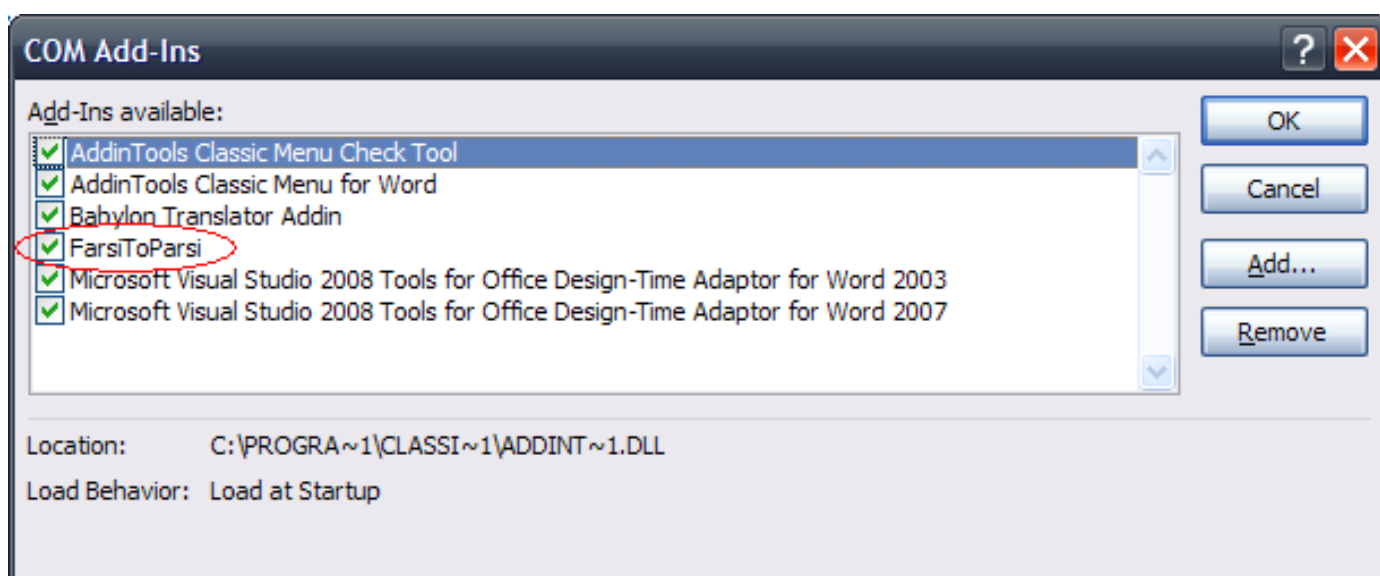
پس از نصب به گزینه‌ی word options مراجعه کنید: (جهت اطمینان از نصب آن)



این افزونه را باید در لیست مربوطه مشاهده نمائید:



هر زمانیکه مایل به عدم استفاده از آن باشید با کلیک بر روی دکمه Go ، در صفحه بعد می‌توان افزونه را حذف کرد:



مثالی دیگر از نحوه‌ی بکارگیری این افزونه:



## نظرات خوانندگان

نویسنده: Aziz

تاریخ: ۱۳۸۷/۰۸/۱۶ ۱۵:۵۴:۰۰

عالی... محشر... دستت درد نکنه.

نویسنده: حسن

تاریخ: ۱۳۸۷/۰۸/۱۶ ۱۶:۳۴:۰۰

عالی! این افزونه کوچولو می‌ارزه تا برم ورد 2007 رو نصب کنم! باورتون نمی‌شه، مدت‌ها بود می‌خواستم یه همچین چیزی بنویسم، البته نه توی ورد، فکرشو کرده بودم که یه محیط تایپ کوچولو باشه. ولی این بهتره! دستت درد نکنه!

نویسنده: Anonymous

تاریخ: ۱۳۸۷/۰۸/۱۷ ۱۰:۲۷:۰۰

وحیدجان کارتون حرف نداره ..

نویسنده: بهرام

تاریخ: ۱۳۸۷/۰۸/۱۷ ۲۳:۵۹:۰۰

با سپاس فراوان  
برای از بین رفتن ی عربی نیز پیشنهاد می‌کنم از این صفحه کلید فارسی استفاده کنید:

[Persian Keyboard](#)

نویسنده: Amirreza

تاریخ: ۱۳۸۷/۰۸/۲۰ ۰۸:۱۱:۰۰

با درود و سپاس فراوان  
وحید جان من 1.1, 2, 3, 3.5 net را روی سیستم نصب دارم و بسته به روز رسانی شما را با حجم 14 مگ را نصب و سپس بسته نصب افزونه را نیز نصب کردم ولی هنوز نتیجه نگرفتم.  
البته باید بگویم net من حجمی تقریباً برابر با 50 مگ دارد.

نویسنده: وحید نصیری

تاریخ: ۱۳۸۷/۰۸/۲۰ ۱۰:۲۹:۰۰

چند مورد رو می‌تونید بررسی کنید  
- البته نیازی نیست آن 200 و خرده‌ای مگ را دریافت کنید. علت بالا بودن حجم آن این است که 32 بیتی و 64 بیتی و غیره، همه را با هم دارد. به آدرس زیر مراجعه کنید

<http://msdn.microsoft.com/en-us/netframework/aa569263.aspx>

روی دکمه نصب کلیک کنید تا یک برنامه 2 مگی در اختیار شما قرار دهد. این برنامه قسمت‌های مورد نیاز سیستم شما رو از سایت مایکروسافت دریافت و نصب خواهد کرد. بنابراین حجم کمتری دارد.

- آیا این افزونه به لیست add-in ها در word اضافه شده؟ (شکل سوم در این صفحه)

- event viewer ویندوز را باز کنید. در قسمت run ویندوز تایپ کنید eventvwr.msc و سپس enter (و یا به قسمت administrative tools ویندوز مراجعه کنید این برنامه را مشاهده خواهید کرد). در صفحه باز شده به قسمت applications مراجعه کنید. آیا موردی با source مساوی FarsiToParsi از نوع خطا با آیکون قرمز مشاهده می‌شود؟ اگر بله، لطفاً دوبار روی آن کلیک کنید و توضیحات صفحه باز شده را اینجا ارسال کنید. از روی این خطا می‌شود مشکل را بهتر بررسی کرد.

با تشکر

نویسنده: Amirreza  
تاریخ: ۱۳۸۷/۰۸/۲۷ ۰۹:۲۱:۰۰

با درود و سپاس  
وحید جان همه چیز روبراه است اما در زمان نصب افزونه در پایان نصب یکدفعه note pad باز شده و متنی را نمایش می‌دهد که  
بدلیل داشتن عبارات اج تی تی پی و غیره نمی‌توان آنها را در اینجا درج نمود.  
البته من کلمات را جدا جدا نوشتم تا بتوان این عبارت را در اینجا درج کنم و در ضمن این افزونه در لیست add-in وجود ندارد و  
در Event viewer نیز خطایی نیست. وهنگامی که سعی کردم از طریق add آنرا نصب کنم باز هم همان صفحه Note Pad در پایان کار  
نمایش داده شد.

با سپاس

با سپاس.

نویسنده: وحید نصیری  
تاریخ: ۱۳۸۷/۰۸/۲۷ ۱۳:۵۳:۰۰

اگر در پایان کار این خطا را می‌گیرید احتمالا یا word مورد استفاده 2007 نیست یا قسمت به روز رسانی سیستم درست نصب  
نشده.  
فایل‌های به روز رسانی سیستم در مسیر زیر موجود هستند  
c:\Program Files\pDateHelperSysUpdate  
یکبار جهت اطمینان بیشتر اینها را دستی هم اجرا و نصب کنید.  
بعلاوه در مسیر زیر  
c:\Program Files\FarsiToParsi  
فایل اصلی نصب همان FarsiToParsi.vsto است که پس از نصب بسته به روز رسانی سیستم و وجود word 2007 با دوبار کلیک  
کردن بر روی آن، کار نصب افزونه را انجام می‌دهد.  
اگر با این موارد مشکل برطرف نشد، لطفا یک print screen از تصویر خطا را مثلا در tinypic.com آپلود کند و اینجا لینک بدهید.  
با سپاس

نویسنده: Anonymous  
تاریخ: ۱۳۸۸/۰۳/۰۹ ۱۰:۰۴:۴۵

سلام واقعا دستت درد نکه

اگه امکانش باشه یک جایی این فرهنگ لغت ( معادل کلمات عربی) رو بشه بروز و تکمیل کرد ؛ خیلی عالی میشه  
منظورم اینه که بشه فرهنگنامه این پلاگین شما کاملتر بشه ؛ توسط همه بالاخص خود کاربر و تایپیست

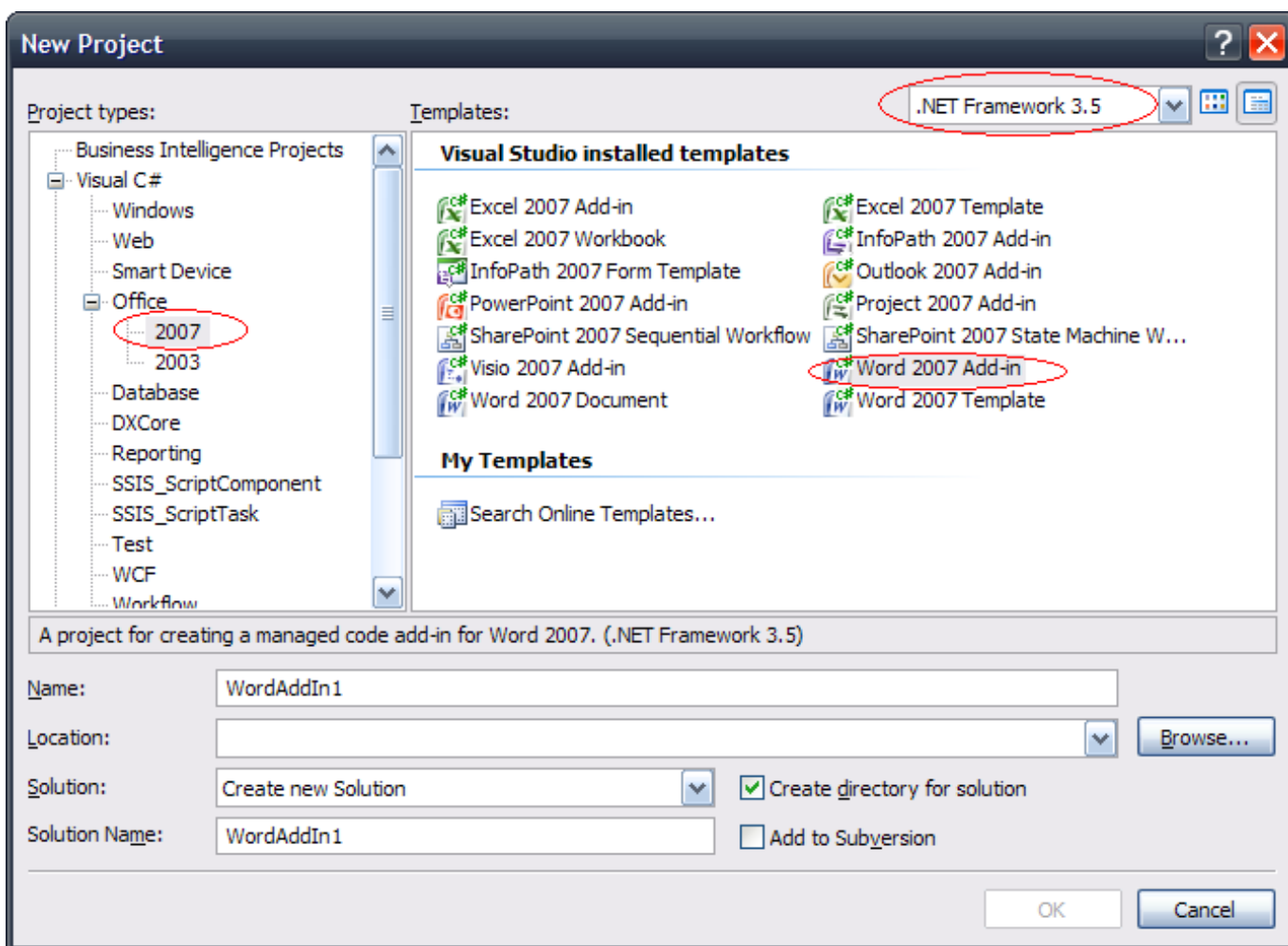
نویسنده: وحید نصیری  
تاریخ: ۱۳۸۸/۰۳/۰۹ ۱۲:۳۰:۲۷

سلام،  
فعلا می‌تونید با استفاده از افزونه زیر:  
<https://addons.mozilla.org/en-US/firefox/addon/5817>  
و یا برنامه Lita

<http://www.dehats.com/drupal/?q=node/58>

فایل ErrorsBank.sqlite را باز کرده (در محل نصب قابل دسترسی است) و رکوردهای آنرا ویرایش یا کم و زیاد کنید.

این افزونه با استفاده از ابزار Visual Studio Tools for Office که به VSTO مشهور شده است، تهیه شد. در بسته به روز رسانی سیستم که در ذیل (معرفی افزونه) نیز معرفی شد نگارش sp1 vsto3.0 آن به صورت خودکار نصب خواهد شد. برای ایجاد این پروژه در VS.Net 2008، تنها کافی است یک پروژه جدید Word add-in را آغاز نمائیم. (شکل زیر)

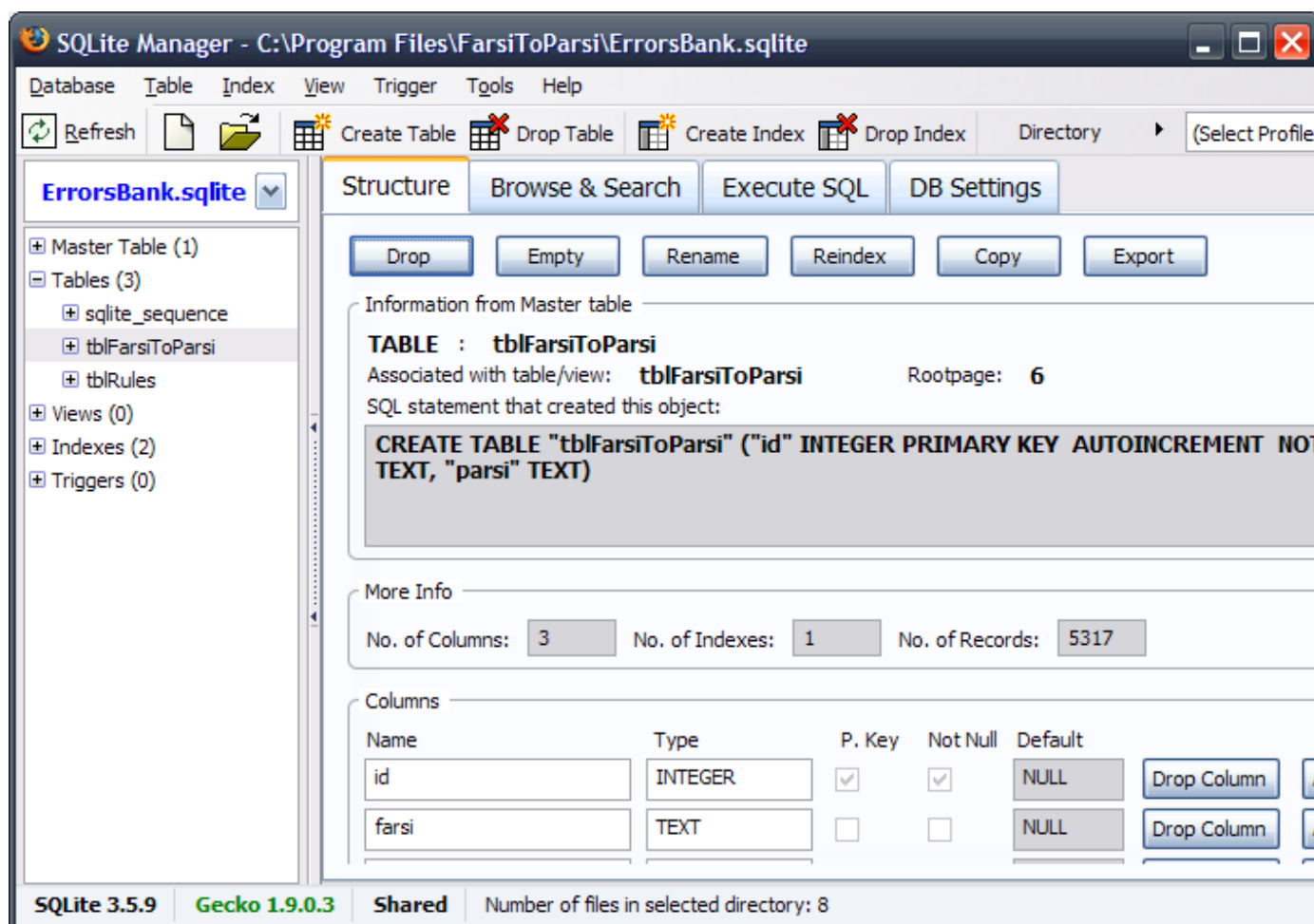


قبل از ادامه بحث، بهتر است در مورد بانک اطلاعاتی مورد استفاده نیز توضیح داده شود. در اینجا از SQLite استفاده شد. (بسیار سبک، کم حجم و سریع است و اساساً یک کاربر نهایی برای تنظیمات آن نیازی نیست اطلاعاتی داشته باشد). بسته به روز رسانی سیستم (در مطلب قبلی)، این مورد را نیز به صورت خودکار نصب خواهد کرد (در GAC باید نصب شود وگرنه افزونه قادر به یافتن آن نخواهد شد).

برای ایجاد این بانک اطلاعاتی، از افزونه [SQLite manager](#) برای فایرفاکس استفاده شد. (این افزونه رایگان شما را از هر ابزار جانبی برای مدیریت یک بانک اطلاعاتی SQLite بی‌نیاز می‌کند) برای مثال فایل ErrorsBank.sqlite برنامه افزونه فارسی به پارسی را توسط افزونه SQLite manager فایرفاکس باز کنید (این



فایل را در محل نصب افزونه می‌توانید پیدا کنید). در اینجا می‌توان جداول جدید را ایجاد کرد، کوئری‌های دلخواه را اجرا نمود و یا اطلاعات را مرور کرده، حذف یا ویرایش کرد (شکل زیر).



و خوشبختانه این بانک اطلاعاتی و محصول کننده‌های آن با اطلاعات یونیکد فارسی هیچ مشکلی ندارند و برای کارهایی با وسعت کم و تعداد رکورد پائین یکی از بهترین انتخاب‌ها به‌شمار می‌روند. نحوه استفاده از SQLite نیز در دات نت بسیار ساده است. اگر با ADO.Net کار کرده باشید، پس از افزودن ارجاعی از اسمبلی System.Data.SQLite.DLL به پروژه و معرفی فضای نام آن به پروژه، تنها کافی است در کدهای قبلی خود برای مثال SqlConnection را به SQLiteConnection تغییر دهید و امثال آن. یعنی دانش ADO.Net شما در اینجا نیز کاملاً قابل استفاده خواهد بود و نیازی نیست مدتی را صرف آشنا شدن با کلاس‌ها و مفاهیم جدید نمائید (البته این تنها زمانی معنا خواهد داشت که به ویزاردها عادت نکرده باشید و کارهای خود را با کد نویسی انجام داده باشید). تنها یک نکته را باید به‌خاطر داشت و آن هم مربوط است به ساز و کار درونی SQLite. هنگام انجام عملیات insert یا update حتماً از transaction استفاده کنید تا سرعت کوئری‌های شما در SQLite به نحو شگفت‌انگیزی افزایش یابد. مثالی در این مورد را در فایل chm راهنمای SQLite.NET می‌توانید پیدا کنید.

مطلب دیگری که پیش از پرداختن به کد نویسی افزونه باید با آن آشنا شویم، مفهوم smart tags در مجموعه آفیس است که در این پروژه از آن استفاده گردید.

smart tags در مجموعه آفیس برچسب‌هایی هستند که به صورت خودکار توسط یکی از محصولات آفیس مثلاً ورد یا اکسل و امثال آن، پس از تشخیص یک کلمه خاص ایجاد می‌شوند و می‌توان اعمالی را به این برچسب ایجاد شده انتساب داد. برای مثال در

اینجا امکان جایگزین کردن کلمه فارسی با معادل پارسی در نظر گرفته شد.

[ویدیویی](#) در مورد نحوه ایجاد اسمارت تگ‌ها در VS.Net و یا [مثالی پیشرفته‌تر](#) در مورد تشخیص دمای فارنهایت در یک متن و ایجاد smart tag مخصوص به آن برای تبدیل به سلسیوس. (از regular expressions جهت یافتن یک الگو در متن استفاده شده است)

در این پروژه، حدود 3800 واژه فارسی به یک smart tag انتساب داده می‌شود (در روال استاندارد ThisAddIn\_Startup). سپس در هنگام نمایش آن، معادل پارسی کلمه نیز به منوی باز شده افزوده گشته و در روال رخداد کلیک آن، تعویض کلمه تشخیص داده شده با واژه پیدا شده صورت خواهد گرفت.

در ادامه فرض بر این است که یک پروژه جدید word add-in را در VS.Net ایجاد کرده‌اید و همچنین ارجاعی را به فایل System.Data.SQLite.DLL افزوده‌اید.

```
using System;
using System.Diagnostics;
using Microsoft.Office.Tools.Word;
using Action = Microsoft.Office.Tools.Word.Action;

private SmartTag _st;
private void init()
{
    try
    {
        //Enable Smart Tags in Word
        if (!Application.Options.LabelSmartTags)
        {
            //ممکن است اسمارت تگ‌ها در ورد غیرفعال باشند. به این صورت می‌شود آنها را فعال کرد
            Application.Options.LabelSmartTags = true;
        }

        _st = new SmartTag(@"www.microsoft.com/Demo#FarsiSmartTag", @"فارسی به پارسی");

        //دریافت واژه‌های فارسی از دیتابیس و افزودن خودکار آنها به اسمارت تگ‌ها
        if (!DBhelper.AddSmartTagItems(_st, "select distinct farsi from tblFarsiToParsi"))
            return;

        Action stActions = new Action("تبدیل"; //تعریف یک اکشن جدید
        stActions.Click += stActions_Click; //روال‌های رخداد گردان
        stActions.BeforeCaptionShow += stActions_BeforeCaptionShow;
        _st.Actions = new[] { stActions };
        VstoSmartTags.Add(_st); //افزودن اسمارت تگ به مجموعه
    }
    catch (Exception ex)
    {
        EventLog.WriteEntry("FarsiToParsi", ex.ToString(), EventLogEntryType.Error, 7);
    }
}

private void ThisAddIn_Startup(object sender, EventArgs e)
{
    init();
}
```

دو روال رخداد گردان زیر نیز جهت تغییر عنوان پیش فرض به واژه یافته شده در لحظه نمایش منو و روال کلیک نیز ایجاد خواهد شد:

```
static void stActions_BeforeCaptionShow(object sender, ActionEventArgs e)
{
    try
    {
        Action clickedAction = sender as Action;
        if (clickedAction != null)
        {
            string parsi = DBhelper.FindParsi(e.Text); //معادل پارسی از دیتابیس دریافت می‌شود
            clickedAction.Caption = (parsi == string.Empty ? e.Text : parsi);
        }
    }
    catch (Exception ex)
```

```

        {
            EventLog.WriteEntry("FarsiToParsi", ex.ToString(), EventLogEntryType.Error, 7);
        }
    }

    static void stActions_Click(object sender, EventArgs e)
    {
        try
        {
            Action clickedAction = sender as Action;
            if (clickedAction != null)
            {
                e.Range.Text = clickedAction.Caption; // جایگزینی متن موجود با عنوانی که پیشتر پارسی
                شده است
            }
        }
        catch (Exception ex)
        {
            EventLog.WriteEntry("FarsiToParsi", ex.ToString(), EventLogEntryType.Error, 7);
        }
    }
}

```

نکته‌ای را که در اینجا باید حتما رعایت کرد بحث exception handling است. خصوصا در روال استاندارد ThisAddIn\_Startup. اگر در این روال خطایی مدیریت نشده رخ دهد، word افزودنی شما را به صورت غیرفعال به مجموعه اضافه خواهد کرد و فعال سازی بعدی آن پس از اصلاح کد واقعا مشکل خواهد بود. همانطور که ملاحظه می‌کنید تمامی خطاها در event log ویندوز نوشته می‌شوند.

همچنین باید دقت داشت که اگر متغیری در سطح کلاس تعریف نشود به احتمال زیاد تا دقایقی بعد توسط garbage collector دیار باقی خواهد شتافت (تعریف st\_ در اینجا). اینجاست که شاید ساعت‌ها وقت صرف کنید که چرا روال‌های رخداد گردان دیگر اجرا نمی‌شوند. چرا افزونه دیگر کار نمی‌کند.

همین! کل سورس این add-in منهای بحث دریافت اطلاعات از دیتابیس همین بود! وظیفه‌ی تشخیص کلمات معرفی شده به ms-word به‌عهده‌ی خود آن است و این کار را نیز به‌خوبی انجام می‌دهد. در گذشته‌های نچندان دور ایجاد یک افزونه برای word واقعا مشکل بود که با این روش بسیاری از موانع برطرف شده است.

کلاس DBHelper که کار دریافت اطلاعات واژه‌ها را از دیتابیس SQLite انجام می‌دهد به شرح زیر است:

```

using System;
using System.Data.SQLite;
using System.Diagnostics;
using System.Reflection;
using Microsoft.Office.Tools.Word;

namespace Farsi2Parsi
{
    class DBHelper
    {
        #region Methods (2)

        // Public Methods (2)

        public static bool AddSmartTagItems(SmartTag st, string strSQL)
        {
            SQLiteDataReader myReader = null;
            SQLiteCommand sqlCmd = null;
            bool ret = false;
            try
            {
                SQLiteConnection sqlCon = new SQLiteConnection
                {
                    ConnectionString = "Data Source=" + ConStr.ConnectionString
                };
                sqlCon.Open();
                sqlCmd = new SQLiteCommand(strSQL, sqlCon);
                myReader = sqlCmd.ExecuteReader();

                if (myReader != null)
                    while (myReader.Read())

```

```

        {
            if (myReader.GetValue(0) != DBNull.Value)
                st.Terms.Add(myReader.GetValue(0).ToString());
        }

        ret = true;
    }
    catch (Exception ex)
    {
        EventLog.WriteEntry("FarsiToParsi", ex + "\n" + Environment.CurrentDirectory + "\n" +
            Assembly.GetExecutingAssembly().Location, EventLogEntryType.Error, 7);
    }
    finally
    {
        if (myReader != null)
            myReader.Close();

        if (sqlCmd != null)
            sqlCmd.Connection.Close();
    }
    return ret;
}

public static string FindParsi(string farsi)
{
    SQLiteDataReader myReader = null;
    SQLiteCommand sqlCmd = null;
    string ret = string.Empty;
    string strSQL = "select parsi from tblFarsiToParsi where farsi='" + farsi.Replace("'", "''")
+ "'";
    try
    {
        SQLiteConnection sqlCon = new SQLiteConnection
        {
            ConnectionString = "Data Source=" + ConStr.ConnectionString
        };
        sqlCon.Open();
        sqlCmd = new SQLiteCommand(strSQL, sqlCon);
        myReader = sqlCmd.ExecuteReader();

        if (myReader != null)
        {
            myReader.Read(); //اولین مورد کافی است
            if (myReader.GetValue(0) != DBNull.Value)
                ret = myReader.GetValue(0).ToString();
        }
    }
    catch (Exception ex)
    {
        EventLog.WriteEntry("FarsiToParsi", ex + "\n" + Environment.CurrentDirectory + "\n" +
            Assembly.GetExecutingAssembly().Location, EventLogEntryType.Error, 8);
    }
    finally
    {
        if (myReader != null)
            myReader.Close();

        if (sqlCmd != null)
            sqlCmd.Connection.Close();
    }
    return ret;
}
}
#endregion Methods
}

```

همانطور که پیشتر نیز عنوان شد اگر با ADO.net آشنایی داشته باشید، هیچ نکته‌ی خاص جدیدی را در اینجا مشاهده نخواهید کرد و تنها یک سری امور روزمره کاری با ADO.net مطرح شده است، باز کردن کانکشن، اجرای کوئری، دریافت اطلاعات و پاکسازی نهایی. (قسمت finally را با استفاده از عبارت using می‌شود حذف کرد)

هنگام نصب برنامه، مسیر پوشه نصب در رجیستری ویندوز توسط نصاب نوشته خواهد شد. از همین مورد برای ایجاد رشته اتصالی به دیتابیس استفاده گردید.

```
class ConStr
{
    public static string ConnectionString
    {
        get
        {
            return
Microsoft.Win32.Registry.LocalMachine.OpenSubKey("SOFTWARE\\FarsiToParsi").GetValue("folder") +
"\\ErrorsBank.sqlite";
        }
    }
}
```

سورس کامل این افزونه را به صورت یک پروژه VS.Net 2008 SP1 از [اینجا](#) می‌توانید دریافت کنید.  
نصاب برنامه با استفاده از [NSIS](#) ایجاد شده که در روزی دیگر درباره‌ی آن توضیح خواهیم داد.  
اگر قصد داشته باشید از روش‌های متداول استفاده کنید، مشاهده ویدیوی زیر توصیه می‌شود:

<http://msdn.microsoft.com/en-us/office/bb851702.aspx>

برای توزیع این نوع افزونه‌ها علاوه بر دات نت فریم ورک، به چهار به روز رسانی دیگر نیز نیاز خواهد بود:  
به روز رسانی نصاب ویندوز (که احتمالاً نصب هست)

WindowsInstaller-KB893803-v2-x86.exe

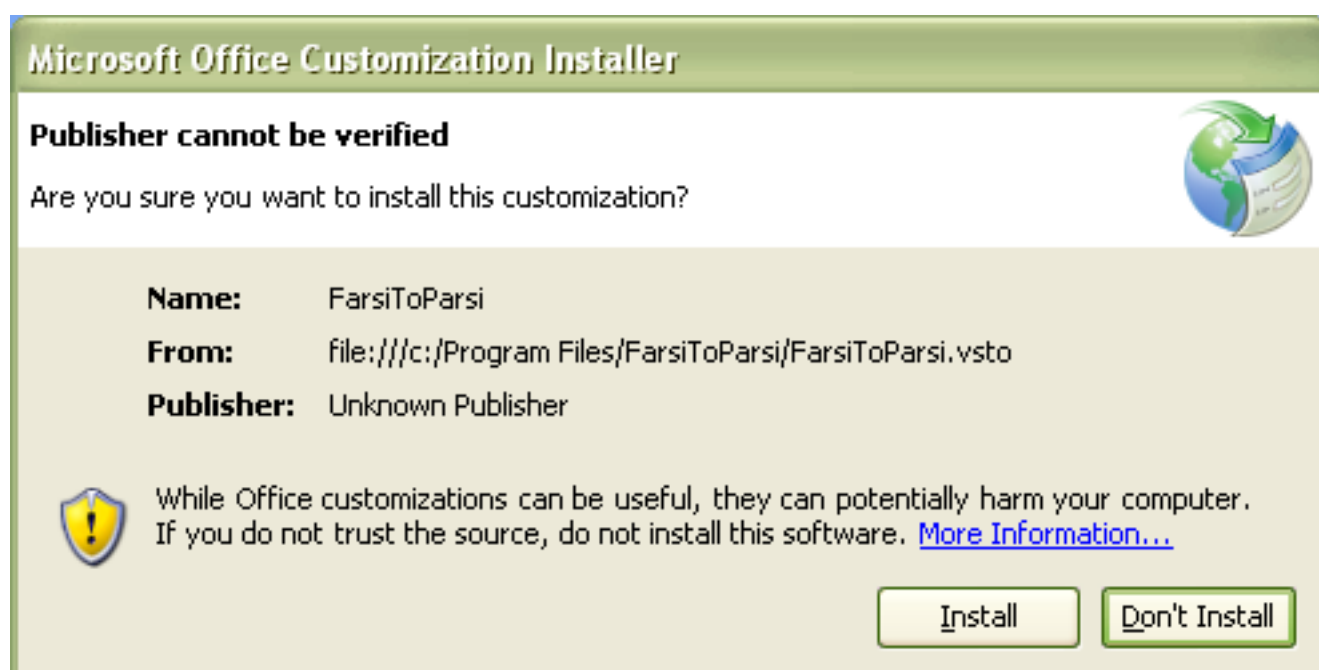
: Microsoft Office System Update: Redistributable Primary Interop Assemblies  
o2007pia.msi

نصب و همچنین sp1 آن

vstor30.exe

vstor30sp1-KB949258-x86.exe

این موارد را من در بسته [به روز رسانی سیستم](#) قرار داده‌ام که به صورت خودکار و یکی پس از دیگری اجرا و نصب خواهند شد.  
پس از آن با کلیک بر روی فایل با پسوند vsto که در پوشه build برنامه موجود است، می‌توان افزونه را نصب کرد (click once installation).



سایر اطلاعات در مورد پروژه‌های VSTO را می‌توان از طریق وبلاگ رسمی آنها دنبال کرد:

[/http://blogs.msdn.com/vsto](http://blogs.msdn.com/vsto)

ایده‌های دیگری را هم در همین رابطه می‌توان پیاده سازی کرد. برای مثال درست کردن یک افزونه برای بررسی آئین نگارش فارسی در متون word. دقیقاً با همین روش قابل پیاده سازی است و یا ایجاد غلط یاب بهتری نسبت به آنچه که هم اکنون برای آفیس 2003 توسط مایکروسافت ارائه شده است (این غلط یاب با صفحه کلید استاندارد تایپ ایران همخوانی ندارد، به همین جهت با استقبال نیز مواجه نشد).

## نظرات خوانندگان

نویسنده: Afshar Mohebbi

تاریخ: ۱۳۸۷/۰۸/۱۷ ۱۲:۱۵:۰۰

کار خیلی جالبی انجام داده‌اید. موفق باشید.

نویسنده: Anonymous

تاریخ: ۱۳۸۷/۰۸/۱۷ ۲۲:۲۵:۰۰

واقعا دستت درد نکنه

چند تا سوال داشتم

میشه دیتا بیس رو xml گذاشت؟ (با یونیکد مشکل پیدا نمیکنه!)

اگه امکان داره چند کتاب الکترونیکی خوب رو که در این زمینه ها مطالعه کردی معرفی کنید

مرسی

نویسنده: وحید نصیری

تاریخ: ۱۳۸۷/۰۸/۱۸ ۰۰:۴۰:۰۰

- فایل xml با یونیکد مشکلی ندارد و اینجا هم میشد از xml استفاده کرد.

برای نمونه یکی از برنامه نویسان گرامی در سایت

<http://psoft.ir/?p=23>

از xml بعنوان دیتابیس نمایش اشعار، پند و اندرز و امثال آن استفاده کرده.

- حقیقت امر این است که در این سایت من از بحث وارز و امثال آن پرهیز خواهم داشت. ولی محض نمونه این عبارات را به همراه pdf در انتهای آن در گوگل جستجو کنید کتابهای مربوطه رو می‌تونید به سادگی پیدا کنید:

Wrox.Professional.VSTO.2005.Visual.Studio.2005.Tools.for.Office

Visual.Studio.Tools.for.Office.Using.C.Sharp.with.Excel.Word.Outlook.and.InfoPath

موفق باشید

نویسنده: امین

تاریخ: ۱۳۸۷/۰۸/۱۸ ۰۷:۰۴:۰۰

سلام آقای نصیری

خسته نباشید. امیدوارم که همیشه موفق باشید و سلامت تا ما هم همیشه از وبلاگتون استفاده کنیم.

نویسنده: مهدی پایروند

تاریخ: ۱۳۸۸/۰۳/۰۵ ۱۸:۰۵:۱۶

سلام، لطفا در مورد نحوه کار با تاریخ در دیتا بیس sqlite کمی توضیح و راهنمایی بفرمایید.

با تشکر

نویسنده: وحید نصیری

تاریخ: ۱۳۸۸/۰۳/۰۵ ۱۸:۱۷:۱۱

سلام،

تاریخ که احتمالا تاریخ فارسی مد نظر شما است همیشه بین علما در این مورد اختلاف بوده و هست ! :

یکی تاریخ رو به صورت عددی با فرمت یونیکس ذخیره میکنه (unixepoch). بعد تمام مقایسه‌های لازم هم درست صورت خواهد

گرفت (مثلا از تاریخ تا تاریخ) چون با عدد کار می‌کنید.

یکی مثل قبض برق و آب و تلفن از رشته استفاده می‌کنه ولی با این شرط که حتما روز و تاریخ دو رقمی ثبت شوند و اگر یک رقمی بود قبل از آن یک صفر قرار گیرد تا مقایسه از تاریخ تا تاریخ همیشه کار کند (از زمان فاکس پروی مرحوم تا به امروز)

یکی هم زمان رو از نوع datetime استاندارد در نظر می‌گیرد که در sqlite تقریبا اثری ندارد. اما sqlite یک تابع به نام date معرفی کرده که به خوبی بر روی رشته‌های تاریخی کار می‌کند و بسیار توانمند است:

<http://www.sqlite.org/cvstrac/wiki?p=DateAndTimeFunctions>

نویسنده: مهدی پایروند

تاریخ: ۱۳۸۸/۰۳/۰۵ ۱۸:۵۷:۳۱

ltdn f,n , llkمفید بود و ممنون از راهنماییتون استاد (:



عنوان: نصب یک اسمبلی دات نت در GAC

نویسنده: وحید نصیری

تاریخ: ۱۳۸۷/۰۸/۱۹ ۱۱:۴۰:۴۱

آدرس: [www.dotnettips.info](http://www.dotnettips.info)

برچسب‌ها: VSTO

افزونه فارسی به پارسی را قبل از ارائه در سایت، بر روی یک ماشین مجازی هم تست کردم. برای این منظور از [Microsoft virtual pc](#) استفاده شد. البته در مقابل امکانات [VMware](#) شاید حرفی برای گفتن نداشته باشد ولی خوب جهت مقاصد تست نرم افزار بر روی یک سیستم عاری از وسایل برنامه نویسی مناسب است. (برای نصب یک سیستم عامل توسط آن برای مثال می‌شود از سی دی آن OS یک فایل ISO تهیه کرد و مسیر این فایل ISO را به ماشین مجازی معرفی کرد. سپس سیستم بوت شده و روال نصب مطابق معمول خواهد بود)

اولین مشکلی که پس از تست بر روی سیستم مجازی رخ داد، پیغام یافت نشدن اسمبلی مربوط به SQLite بود. نرم افزار word هنگام اجرای افزونه‌های دات نت، آنها را در مسیری با یک نامگذاری منحصر بفرد کپی می‌کند و تنها هم همان اسمبلی افزونه را کپی می‌کند و نه سایر موارد همراه را. برای پیدا کردن این مسیر می‌شود از روش زیر استفاده کرد:

```
using System.Reflection;
Assembly.GetExecutingAssembly().Location
```

در این مسیر اسمبلی SQLite وجود ندارد و به همین دلیل هم بارگذاری نخواهد شد. بهترین راه حل برای رفع این مشکل، نصب اسمبلی مربوطه در [GAC](#) یا global assembly cache است.

برای نصب اسمبلی در GAC استفاده از برنامه [gacutil توصیه شده](#) است. این برنامه به همراه SDK دات نت فریم ورک ارائه می‌شود و الزامی ندارد که کاربر نهایی این برنامه را داشته باشد. خوشبختانه با استفاده از برنامه نویسی هم می‌شود یک نمونه از برنامه Gacutil را خودمان ایجاد کنیم (برای مثال ایجاد یک برنامه کنسول و دریافت مسیر از طریق آرگومان‌های ارسالی به آن):

```
new System.EnterpriseServices.Internal.Publish().GacInstall(path);
```

در اینجا باید ارجاعی از System.EnterpriseServices نیز به برنامه اضافه شود. این روش در مورد اسمبلی SQLite که دارای امضای دیجیتال است کار خواهد کرد. اما اگر قصد داشته باشید به صورت عمومی از آن استفاده کنید، باید ابتدا بررسی کرد که آیا فایل اسمبلی دارای امضای دیجیتال است یا خیر. برای این منظور می‌توان مقدار [عبارت زیر](#) را ارزیابی کرد:

```
Assembly.LoadFile(path).GetName().GetPublicKey().Length
```

اگر این طول بزرگتر از صفر بود به این معنا است که فایل اسمبلی دارای امضای دیجیتال است و می‌توان آنرا در GAC نصب کرد. لازم به ذکر است که متد معرفی شده برای نصب در GAC در صورت عدم موفقیت هیچ پیغام خطا یا exception ایی را در برنامه تولید نخواهد کرد. اما پیغام خطای حاصل را در event log ویندوز می‌توان مشاهده کرد.

عنوان: افزونه تاریخ فارسی برای outlook 2007

نویسنده: وحید نصیری

تاریخ: ۱۳۸۷/۰۸/۲۵ ۰۳:۰۰

آدرس: [www.dotnettips.info](http://www.dotnettips.info)

برچسب‌ها: VSTO

عموما در محیط کاری اگر شبکه ویندوزی و mail server مورد استفاده هم ms exchange باشد، به طور قطع از outlook برای انجام امور روزمره ارسال و دریافت ایمیل استفاده می‌شود. طبق معمول هم مشکل ما تاریخ فارسی است! یکی از شرکت‌های ایرانی که در این باره محصولی را ارائه داده با hook کردن تاریخ ویندوز، هر جایی که تاریخی قرار است نمایش داده شود، آنرا فارسی می‌کند. این محصول دو ایراد دارد: الف) رایگان نیست! ب) این hook بر روی عملکرد سایر برنامه‌ها تاثیرگذار است. برای مثال برنامه‌های دات نت تاریخ قمری را نمایش خواهند داد، بر روی عملکرد و کارایی کلی سیستم تاثیر منفی دارد و مشکلاتی از این دست.

افزونه زیر بدون دستکاری تاریخ ویندوز، دو کار را در MS outlook 2007 انجام خواهد داد:

الف) اضافه کردن ستون "تاریخ دریافت" شمسی

From	Subject	Received	Size	C... تاریخ دریافت
app_err_log@r...	پیغام خطای برنامه	Fri 2008/11/14 11:02 ب.ظ	4 KB	جمعه ۲۴ آبان ۱۳۸۷ - ۱۱:۰۲ ب.ظ

ب) در متن دریافتی، تمام تاریخ‌های sent موجود را یافته و شمسی می‌کند

From: Esmaili, Behnam

Sent: سه شنبه 21 آبان 1387 - 03:08 ب.ظ

To: Nasiri, Vahid

Subject: RE:

دریافت افزونه:

لطفا اینجا کلیک کنید.

تذکر مهم:

نصب دو بسته به روز رسانی سیستم و دات نت فریم ورک را پیش از نصب این افزونه فراموش نکنید.

[word-2007/افزونه-فارسی-به-پارسی-برای-2007-word](http://www.dotnettips.info/Post/14/افزونه-فارسی-به-پارسی-برای-2007-word)

در کل برای من کار راه انداز بوده (:

در طی روزهای آتی، سورس کامل و نحوه برنامه نویسی آنرا بررسی خواهیم کرد.



## نظرات خوانندگان

نویسنده: sajjad

تاریخ: ۱۳۸۹/۰۳/۰۴ ۱۲:۲۱:۳۲

سلام. خیلی جالب بود.  
البته برای من در قسمت sent همون تاریخ میلادی رو نمایش میده.  
آیا این بسته رو کامل کردید؟  
مثلا طوری باشه که در تقویم هم بتونه تاریخ رو شمسی نمایش بده؟

نویسنده: وحید نصیری

تاریخ: ۱۳۸۹/۰۳/۰۴ ۱۴:۱۳:۰۱

خیر. هستند یک سری شرکت که با hook تاریخ ویندوز اینکار رو کردند ولی این روش در سطح سیستم اثر گذار است و مشکل ساز و به شخصه استفاده نمی‌کنم.  
همین روش مختصر و مفید بالا بدون دستکاری ویندوز به نظر من ارجحیت بیشتری دارد.

نویسنده: وحید نصیری

تاریخ: ۱۳۸۹/۰۳/۰۴ ۱۴:۱۵:۰۲

در مورد قسمت sent , یک سری الگوی تاریخ رو این برنامه تشخیص می‌ده. ممکنه تاریخ انگلیسی یا هر تاریخ ذکر شده دیگری در آن با الگوهای پیش فرض تطابق نداشته باشد. می‌تونید یک عکس یا حتی یک خروجی Html از این صفحه برای من ارسال کنید تا بررسی کنم (کلیک راست و مشاهده سورس ایمیل).  
با تشکر

قبل از شروع بحث، سورس کامل پروژه را از [اینجا](#) دریافت کنید (یک پروژه VSTO از نوع outlook add-in در VS.Net 2008 SP1).

توضیحات مربوطه را به دو قسمت تقسیم کرده‌ام. قسمت اول یافتن تاریخ‌های sent و فارسی کردن آنها و قسمت بعدی نحوه اضافه کردن یک ستون و مقدار دهی آن (در روزی دیگر).

متن ایمیل‌های دریافتی در آوت‌لوک‌های جدید عموماً به دو فرمت HTML و یا RichText دریافت می‌شوند. حالت‌های دیگری هم مانند plain و unspecified هم موجود هستند که حتی اگر ایمیلی را به صورت plain ارسال نمائید، با فرمت RichText نمایش داده خواهد شد (بنابراین بر اساس آزمایشات من بررسی این دو فرمت کفایت می‌کند).

برای اینکه قسمت‌های sent را پیدا کنیم در ابتدا باید سورس صفحه را بررسی نمائیم (کلیک راست و view source). در حالت فرمت HTML داریم:

```
<p class=MsoNormal><b><span style='font-size:10.0pt;font-family:"Tahoma", "sans-serif"'>From:</span></b><span style='font-size:10.0pt;font-family:"Tahoma", "sans-serif"'> Nasiri, Vahid <br><b>Sent:</b> <span lang=AR-SA dir=RTL>our date goes here</span><br><b>To:</b> xyz<br><b>Subject:</b> our subject</p>
```

و در حالت ایمیل‌های RichText خواهیم داشت:

```
From: tst@tst.net<br>Sent:<span lang=AR-SA dir=RTL>our date goes here</span><br>To: Nasiri, Vahid<br>Subject: <span lang=AR-SA dir=RTL>xyz</span><br>
```

خوب، برای پیدا کردن عبارت تاریخ قسمت sent چه باید کرد؟ (our date goes here در اینجا) استفاده از روش‌های متداول کار با رشته‌ها در اینجا به علت انبوهی از تگ‌های HTML اصلاً مقرون به صرفه نیست و کند خواهد بود. خوشبختانه با وجود کتابخانه regular expressions در دات نت، پیدا کردن عبارتی که از یک الگوی خاص پیروی می‌کنند به سادگی و با سرعت بسیار بالایی قابل انجام است. پیشنهاد من برای دو فرمت بالا به صورت زیر بوده: (شاید شما الگوی دیگری را یافتید، زیبایی اوپن سورس :)

```
private const string REGEXHTMLPATTERN = @"(?s)>\s(.+?)<br>";
private const string REGEXPLAINTEXTPATTERN = "(?s)Sent:(.+?)<br>";
```

برای مثال در حالت دوم هر چیزی که بین sent و br قرار می‌گیرد در کل متن بررسی خواهد شد (با استفاده از MatchCollection فضای نام System.Text.RegularExpressions). در اینجا اگر Convert.ToDateTime موفق بود یعنی تاریخ قابل تبدیل است (البته قبل از تبدیل تمام تگ‌های HTML احتمالی هم تمیز خواهند شد) و ما آنرا با استفاده از تابع DateTimeToFarsiStr در کلاس cDate به نمونه شمسی تبدیل کرده (لطفاً به سورس برنامه مراجعه کنید) و نهایتاً آنرا در متن جایگزین می‌کنیم. سرعت استفاده از RegularExpressions فوق العاده بالا است و برای نمونه در ایمیلی با بیش از 20 رپیلای در کسری از ثانیه کل

این عملیات انجام خواهد شد.

تا اینجا بررسی کلی الگوریتم مورد استفاده قسمت اول به پایان می‌رسد.

بیشترین وقتی که در این پروژه صرف شد نحوه پیدا کردن شیء MailItem جاری باز شده با استفاده از رخدادهای آوت‌لوک بود (مدت مدیدی را برای این مورد وقت گذاشتم! چون عملاً در هیچ کتابی به این مباحث پرداخته نمی‌شود و باید کل نت را زیر و رو کرد). دو مورد را باید بررسی کرد. الف) inspector ها (صفحه‌ای که جهت ایجاد یک ایمیل جدید یا رپیلای به ایمیل جاری باز می‌شود، inspector نام دارد) ب) ActiveExplorer ها (صفحه‌ای که لیست ایمیل‌ها را نمایش می‌دهد و این صفحه می‌تواند در فولدرهای مختلفی که شما ایجاد کرده‌اید نیز نمایش داده شود بنابراین بررسی inbox به تنهایی کافی نیست) نحوه ایجاد اشیاء مربوطه و تحت نظر قرار دادن آنها را در روال ThisAddIn\_Startup فایل ThisAddIn.cs می‌توانید مشاهده نمائید. نکته مهمی که اینجا وجود دارد، تعریف این اشیاء در سطح کلاس است. در غیراینصورت با اولین خانه تکانی garbage collector، اشیاء شما (بدلیل نبود ارجاعی فعال به آنها) معدوم خواهند شد(!) و دیگر روال‌های رخداد گردان تعریف شده کار نخواهند.

اضافه کردن یک ستون در آوت لوک کار ساده‌ای است. برای مثال زمانی که inbox باز است، بر روی قسمت نمایش ایمیل‌ها کلیک راست کرده و مسیر زیر را طی کنید:

در اینجا فرض بر این است که از منوی اصلی view->reading pane->bottom انتخاب شده است.

Customize current view -> fields -> new field

به این صورت می‌شود یک UserDefinedProperties را تعریف و سپس آن را به ViewFields موجود اضافه کرد. نحوه انجام اینکار را در تابع [addNewCol](#) می‌توانید مشاهده نمایید.

اما مقدار دهی ردیف‌های این ستون ایجاد شده کار ساده‌ای نیست و نیاز است تا با نکته زیر آشنا بود:

```
<view type="table">
<viewname>Messages</viewname>
<column>
<heading>تاریخ دریافت</heading>
<prop>http://schemas.microsoft.com/mapi/string/{00020329-0000-0000-C000-000000000046}/تاریخ%20دریافت</prop>
<type>string</type>
<width>150</width>
<style>padding-left:3px;;text-align:left</style>
<editable>1</editable>
</column>
</view>
```

Outlook ساختار ستون‌های موجود را با فرمت xml نگهداری می‌کند و اگر نیاز داشتید تا ردیف ستونی را مقدار دهی کنید باید ابتدا مقدار تگ prop مربوط به آن ستون را دریافت کرده و سپس بر اساس آن، کار مقدار دهی را انجام دهید. نحوه انجام اینکار را در تابع getColumnProp می‌توانید ملاحظه نمایید و نهایتاً نحوه استفاده از این خاصیت دریافت شده جهت مقدار دهی یک ردیف در تابع expSelectionChange ارائه شده است.

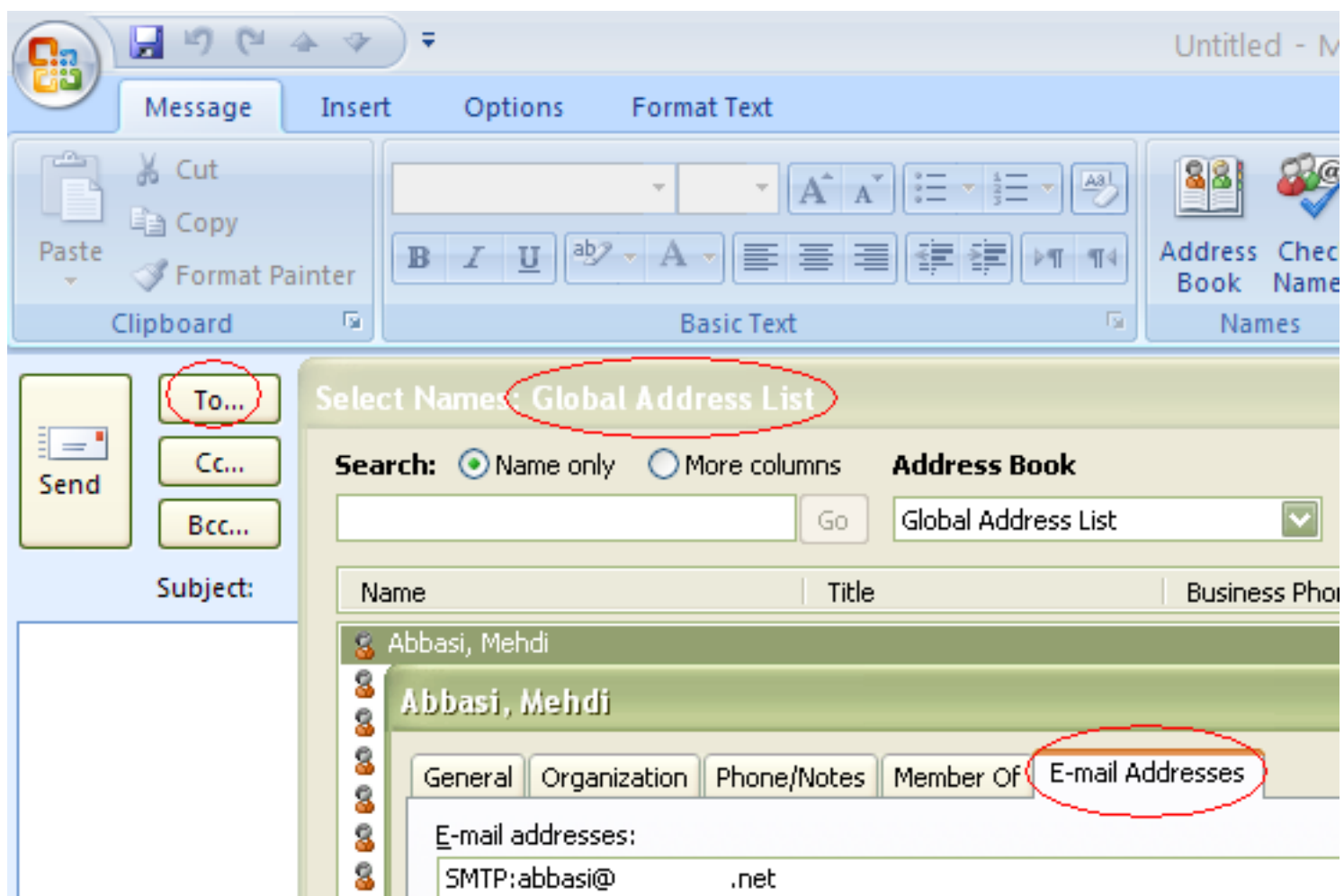
اگر علاقمند بودید که مقدار کامل این ساختار را مشاهده نمایید در تابع addNewCol، پس از تعریف curView سطر زیر را اضافه کنید:

```
MessageBox.Show(curView.XML);
```

## لیست کردن ایمیل‌های موجود در Global address list

عنوان: وحید نصیری  
نویسنده: ۲۰:۲۵:۰۰ ۱۳۸۸/۰۲/۳۰  
تاریخ: [www.dotnettips.info](http://www.dotnettips.info)  
آدرس: VSTO  
برچسب‌ها:

Global Address List یا به اختصار GAL و یا همان Microsoft Exchange Global Address Book ، حاوی اطلاعات تمامی کاربران تعریف شده در Exchange server مایکروسافت است و زمانیکه outlook در شبکه به exchange server متصل می‌شود، کاربران می‌توانند با کمک آن لیست اعضا را مشاهده کرده ، یک یا چند نفر را انتخاب نموده و به آن‌ها ایمیل ارسال کنند (شکل زیر):



نیاز بود تا این لیست تعریف شده در مایکروسافت اکسچنج، با اطلاعات یک دیتابیس مقایسه شوند که آیا این اطلاعات مطابق رکوردهای موجود تعریف شده یا خیر. بنابراین اولین قدم، استخراج email های موجود در GAL بود (دسترسی به همین برگه‌ی email address که در شکل فوق ملاحظه می‌کنید از طریق برنامه نویسی) که خلاصه آن تابع زیر است:  
جهت استفاده از آن ابتدا باید یک ارجاع به کتابخانه COM ایی به نام Microsoft Outlook Object Library اضافه شود.

```
using System.Collections.Generic;  
using System.Reflection;  
using Microsoft.Office.Interop.Outlook;  
  
namespace GAL
```



```

{
    //add a reference to Microsoft Outlook 12.0 Object Library
    class COutLook
    {
        public struct User
        {
            public string Name;
            public string Email;
        }

        public static List<User> ExchangeServerEmailAddresses(string userName)
        {
            List<User> res = new List<User>();
            //Create Outlook application
            Application outlookApp = new Application();
            //Get Mapi NameSpace and Logon
            NameSpace ns = outlookApp.GetNamespace("MAPI");
            ns.Logon(userName, Missing.Value, false, true);

            //Get Global Address List
            AddressLists addressLists = ns.AddressLists;
            AddressList globalAddressList = addressLists["Global Address List"];
            AddressEntries entries = globalAddressList.AddressEntries;
            foreach (AddressEntry entry in entries)
            {
                ExchangeUser user = entry.GetExchangeUser();
                if (user != null && user.PrimarySmtpAddress != null && entry.Name != null)
                {
                    res.Add(new User
                    {
                        Name = entry.Name,
                        Email = user.PrimarySmtpAddress
                    });
                }
            }

            ns.Logoff();

            // Clean up.
            outlookApp = null;
            ns = null;
            addressLists = null;
            globalAddressList = null;
            entries = null;

            return res;
        }
    }
}

```

و نحوه استفاده از آن هم به صورت زیر می‌تواند باشد:

```

List<COutLook.User> data = COutLook.ExchangeServerEmailAddresses("nasiri");
foreach (var list in data)
{
    //....
}

```

در اینجا Nasiri نام کاربری شخص در دومین است (کاربر جاری لاگین کرده در سیستم). تنها نکته‌ی مهم این کد، مهیا نبودن فیلد ایمیل در شیء [AdderssEntry](#) است که باید از طریق متد [GetExchangeUser](#) آن اقدام شود.