

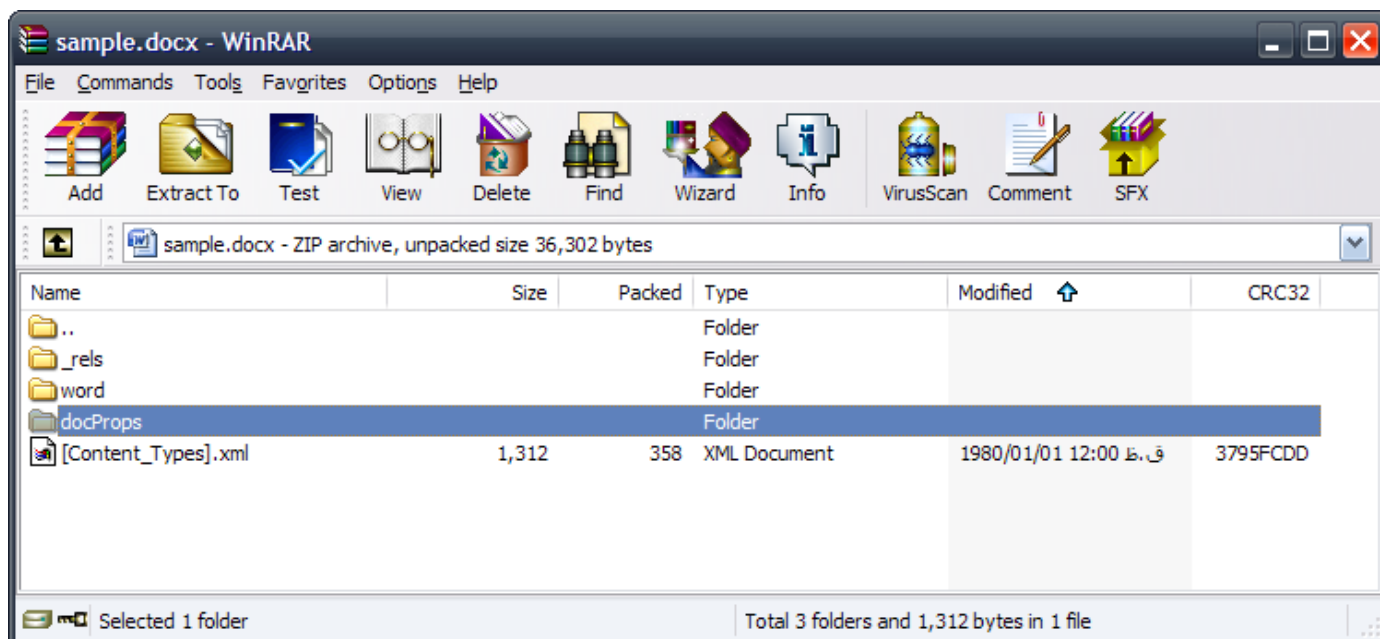
یکی از مواردی که ممکن است در محیط کاری با آن برخورد داشت، تقاضای تولید فایل word یک گزارش با فرمتی مشخص از یک برنامه ASP.Net است. برای مثال یک قالب درست کرده‌اند که header و footer و کلا یک فرمت رسمی دارد. الان برنامه شما باید این فایل word رسمی را با گزارشی که تولید می‌کند پر کند. حالا اینجا است که گرفتاری برنامه نویس شروع می‌شود! روی سرور باید word نصب باشد تا توسط اشیاء COM آن بتوان یک چنین کارهایی را آن‌هم با ASP.Net که به صورت پیش فرض کمترین سطح دسترسی را روی سیستم دارد انجام داد. یا اینکه باید به سراغ کامپوننت‌های تجاری رفت و حالا اینجا با این وضع تحریم و غیره چگونه بتوان آن‌ها را خریداری کرد یا شاید احتمالا در سایت‌های وارز بتوان نسخه تکه پاره شده آن‌ها را یافت. مشکلی هم که این نوع کامپوننت‌ها دارند این است که ممکن است سال دیگر اصلا ساپورت نشوند. محصولات مایکروسافت هم که مرتبا در حال به روز رسانی هستند. در این حالت برنامه متکی به این نوع کامپوننت‌های تجاری سورس بسته در همان نگارش قبلی خود مجبور است باقی بماند.

خوشبختانه با ارائه آفیس 2007 و فرمت OpenXML فایل‌های آن، این مشکل تقریبا مرتفع شده است. مایکروسافت نیز برای سهولت تولید این نوع اسناد، OpenXML SDK را ارائه داده است که از آدرس زیر قابل دریافت است:

[Open XML Format SDK 1.0](#)

البته پیش نمایش [نگارش دو](#) SDK آن نیز موجود است که در مطلب جاری به آن پرداخته نخواهد شد.

فایل‌های office 2007 از یک فایل zip تشکیل شده از چند فایل xml داخل آن، ایجاد شده‌اند. برای مثال یک فایل docx را با winrar یا امثال آن باز کنید (تصویر زیر):



برای کار با اینگونه اسناد باید با اصطلاحات زیر آشنا شد:

Package : فایل zip شما (همان فایل برای مثال docx) اینجا یک بسته نام دارد.

Parts : اجزای این بسته که همان فایل‌های آن هستند، parts نامیده شده اند.

Relations : اگر به فایل‌های موجود در یک بسته دقت کنید، فایلهایی با پسوند rels را خواهید دید که بیانگر نحوه ارتباط Parts با یکدیگر هستند.

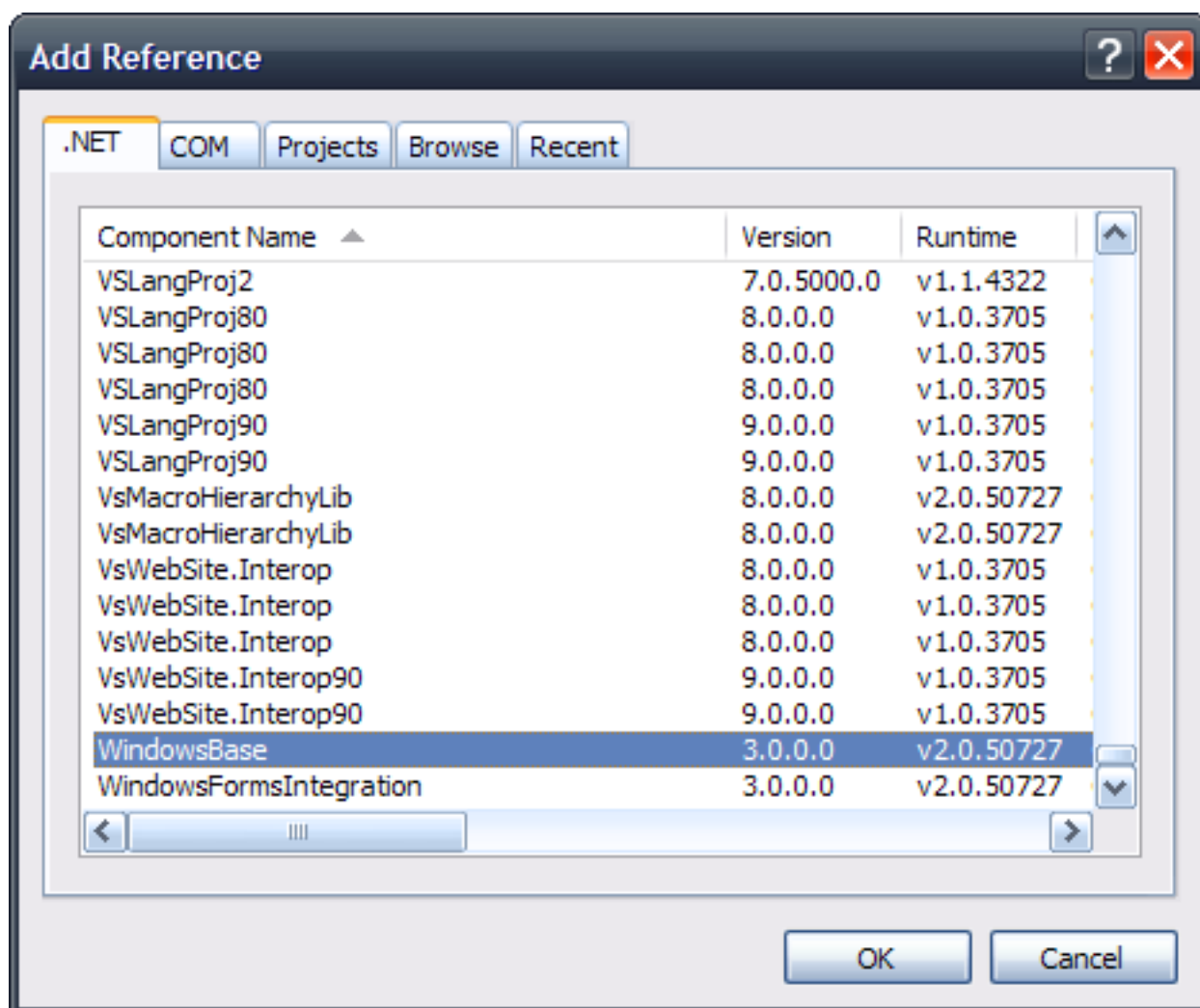
Relations Ids: هر ارتباط با یک ID منحصر بفرد تعریف می‌گردد.

اگر علاقمند باشید که پوستری را در این رابطه مشاهده نمایید می‌توان به آدرس زیر مراجعه نمود.

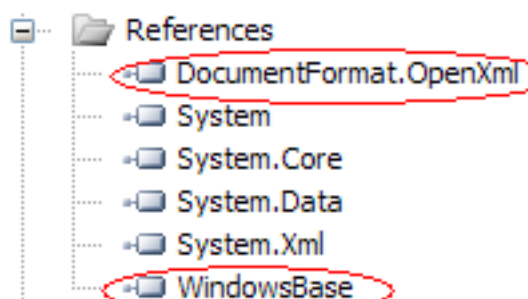
[Open XML Developer Map](#)

نحوه استفاده از OpenXML SDK در دات نت:

ابتدا باید ارجاعی را به فایل DocumentFormat.OpenXml.dll که پس از نصب در مسیر OpenXMLSDK\1.0.1825\lib قرار گرفته است به پروژه افزود. سپس نیاز است تا ارجاعی به کتابخانه WindowsBase نیز به برنامه افزوده شود (تصویر زیر). افزودن ارجاعی به این کتابخانه جهت کامپایل برنامه ضروری است (شکل زیر).



تا اینجا ارجاعات برنامه به صورت زیر خواهند بود:



یک مثال ساده:

قصد داریم یک فایل docx ساده را با استفاده از OpenXML SDK ایجاد کنیم. در مثال زیر فرمت متغیر docx را می‌توان با ایجاد یک فایل docx ساده در word و سپس باز کردن بسته فشرده شده آن و مشاهده محتوای فایل word\document.xml بدست آورد.

```
using System.IO;
using System.Text;
using DocumentFormat.OpenXml;
using DocumentFormat.OpenXml.Packaging;

namespace OpenXMLTestApp
{
    class CWord
    {
        public static void CreateDocument(string documentFileName, string text)
        {
            using (WordprocessingDocument wordDoc =
                WordprocessingDocument.Create(documentFileName, WordprocessingDocumentType.Document))
            {
                MainDocumentPart mainPart = wordDoc.AddMainDocumentPart();

                string docXml =
                    @"<?xml version=""1.0"" encoding=""UTF-8"" standalone=""yes""?>
                    <w:document
                    xmlns:w=""http://schemas.openxmlformats.org/wordprocessingml/2006/main"">
                    <w:body><w:p><w:r><w:t>#REPLACE#</w:t></w:r></w:p></w:body>
                    </w:document>";

                docXml = docXml.Replace("#REPLACE#", text);

                using (Stream stream = mainPart.GetStream())
                {
                    byte[] buf = (new UTF8Encoding()).GetBytes(docXml);
                    stream.Write(buf, 0, buf.Length);
                }
            }
        }
    }
}
```

و نحوه استفاده از آن می‌تواند به صورت زیر باشد:

```
CWord.CreateDocument("test.docx", "سلام دنیا");
```

این کتابخانه کار ایجاد فایل‌های xml، تولید روابط بین آنها و همچنین بسته بندی و zip کردن نهایی را به صورت خودکار انجام می‌دهد.

برای مطالعه بیشتر می‌توان به منابع زیر مراجعه نمود:

یک ویدیوی آموزشی رایگان از مایکروسافت

[دریافت](#)

سؤالات متداول در MSDN

<http://msdn.microsoft.com/en-us/library/bb491088.aspx>

البته اگر پس از نصب SDK به پوشه doc آن مراجعه نمائید، این سؤال و جواب‌ها را در فایل راهنمای chm آن نیز می‌توان پیدا کرد.

مثال دیگری در مورد ایجاد یک گزارش از بانک اطلاعاتی و گرفتن خروجی docx از آن

<http://openxmldeveloper.org/articles/GenerateWordTable.aspx>

البته این مثال خیلی قدیمی است و قسمت‌های کار با پکیج را با SDK ارائه شده می‌توان به صورت خودکار انجام داد. اما حداقل نحوه تولید جداول استاندارد OpenXML را می‌توان از آن ایده گرفت.

[مثالی](#) در مورد نحوه قرار دادن عکس در فایل docx تولیدی

همچنین مثال‌های بیشتری را در وبلاگ‌های مربوطه می‌توان یافت:

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

<http://blogs.msdn.com/ericwhite/default.aspx>

نظرات خوانندگان

نویسنده: افشار محبی
تاریخ: ۱۳۸۷/۰۹/۲۰ ۰۹:۲۸:۰۰

این نوشته برای من خیلی مفید بود. سپاس.

نویسنده: مهدی پایروند
تاریخ: ۱۳۸۷/۰۹/۲۳ ۱۳:۱۵:۰۰

واقعا عالی بود ممنون.
من از طریق سایت مهدی یوسقی با شما آشنا شدم