

iTextSharp پایه کار با فایل‌های PDF را ارائه می‌دهد اما ابزاری را جهت ساده‌تر سازی تولید فایل‌های PDF به همراه ندارد؛ هر چند مثلا امکان تبدیل HTML به PDF را دارا است اما باید گفت: «تا حدودی البته». اگر نیاز باشد جدولی را ایجاد کنیم باید کد نویسی کرد، اگر نیاز باشد تصویری اضافه شود به همین ترتیب و الی آخر. البته این را هم باید در نظر داشت که کد نویسی انعطاف قابل توجهی را در اختیار برنامه نویسی قرار می‌دهد؛ شاید به همین دلیل این روزها مباحث «Code first» بیشتر مورد توجه برنامه نویسی‌ها است، تا مباحث «Wizard first» یک دهه قبل!

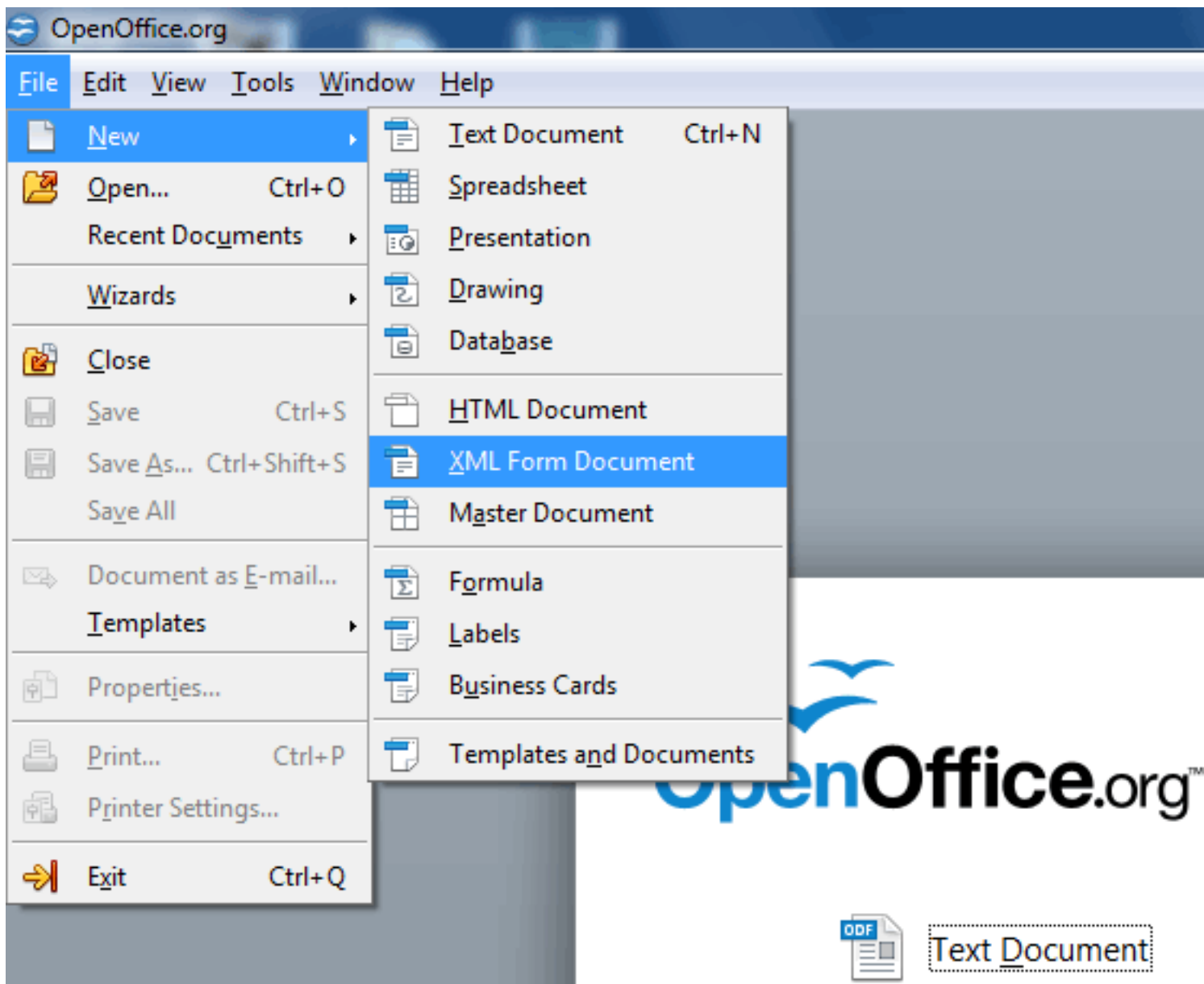
اما باز هم داشتن یک طراح بد نیست و می‌تواند در کاهش مدت زمان تولید نهایی یک فایل PDF مؤثر باشد. برای این منظور می‌توان از برنامه‌ی رایگان و معروف Open office استفاده کرد. توسط آن می‌توان یک فرم PDF را طراحی و سپس فیلدهای آن را (این قالب تهیه شده را) با iTextSharp پر کرد. این مورد می‌تواند برای تهیه گزارش‌هایی که تهیه آن‌ها با ابزارهای متداول گزارش سازی عموما میسر نیست، بسیار مناسب باشد.

طراحی یک فرم PDF با استفاده از برنامه Open Office

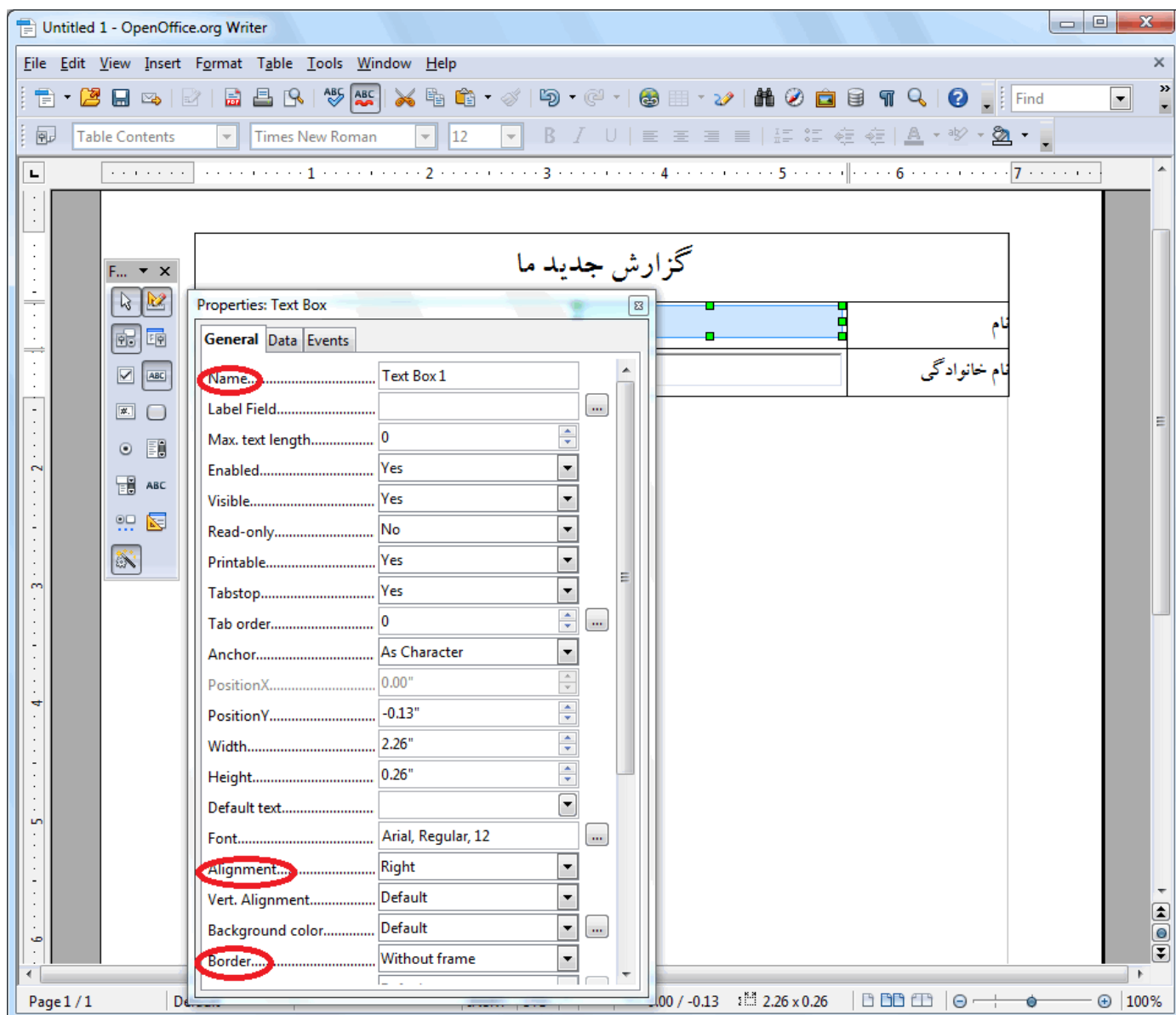
آخرین نگارش برنامه Open office را [از اینجا](#) می‌توانید دریافت کنید و آنچنان حجمی هم ندارد؛ حدودا 154 مگابایت است. پس از نصب و اجرای برنامه، حداقل به دو طریق می‌توان یک فرم جدید را شروع کرد:

الف) آغاز یک XML Form document جدید در Open office سبب خواهد شد که نوارهای ابزار طراحی فرم، مانند قرار دادن TextBox ، CheckBox و غیره به صورت خودکار ظاهر شوند.

ب) و یا آغاز یک سند معمولی و سپس مراجعه به منوی View->Toolbars->Form Controls هم همان حالت را به همراه خواهد داشت.

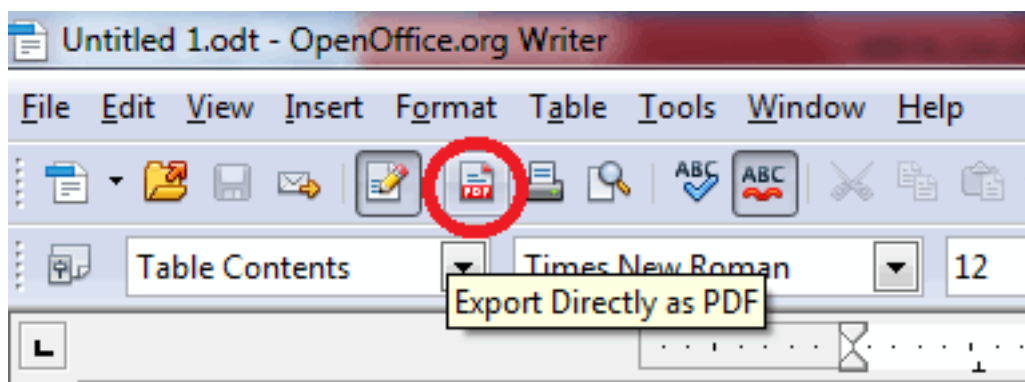


در اینجا برای طراحی یک گزارش یا فرم جدید تنها کافی است همانند روش‌های متداول تهیه یک سند معمولی رفتار کنیم و مواردی را که قرار است توسط iTextSharp مقدار دهی کنیم، با کنترل‌های نوار ابزار Form آن بر روی صفحه قرار دهیم که نمونه‌ی ساده آن‌را در شکل زیر ملاحظه می‌کنید:



برای گزارش‌های فارسی بهتر است Alignment یک کنترل به Right تنظیم شود و Border به حالت Without frame مقدار دهی گردد. نام این کنترل را هم بخاطر بسپارید و یا تغییر دهید. از این نام‌ها در iTextSharp استفاده خواهیم کرد. (صفحه خواص فوق با دوبار کلیک بر روی یک کنترل قرار گرفته بر روی فرم ظاهر می‌شود)

مرحله بعد، تبدیل این فرم به فایل PDF است. کلیک بر روی دکمه تهیه خروجی به صورت PDF در نوار ابزار اصلی آن برای اینکار کفایت می‌کند. این گزینه در منوی File نیز موجود است.



فرم‌های PDF تهیه شده در اینجا، فقط خواندنی هستند. مثلاً یک کاربر می‌تواند آن‌ها را پر کرده و چاپ کند. اما ما از آن‌ها در ادامه به عنوان قالب گزارشات استفاده خواهیم کرد. بنابراین جهت ویرایش فرم‌های تهیه شده بهتر است فایل‌های اصلی Open Office مرتبط را نیز درجایی نگهداری کرد و هر بار پس از ویرایش، نیاز است تا خروجی جدید PDF آن‌ها تهیه شود.

استفاده از iTextSharp جهت مقدار دهی فیلدهای یک فرم PDF

در ادامه می‌خواهیم این قالب گزارشی را که تهیه کردیم با کمک iTextSharp پر کرده و یک فایل PDF جدید تهیه کنیم. سورس کامل اینکار را در ذیل مشاهده می‌کنید:

```
using System;
using System.Diagnostics;
using System.IO;
using iTextSharp.text;
using iTextSharp.text.pdf;

namespace PdfForm
{
    class Program
    {
        //روش صحیح ثبت و معرفی فونت در این کتابخانه
        public static iTextSharp.text.Font GetTahoma()
        {
            var fontName = "Tahoma";
            if (!FontFactory.IsRegistered(fontName))
            {
                var fontPath = Environment.GetEnvironmentVariable("SystemRoot") +
                "\\fonts\\tahoma.ttf";
                FontFactory.Register(fontPath);
            }
            return FontFactory.GetFont(fontName, BaseFont.IDENTITY_H, BaseFont.EMBEDDED);
        }

        static void Main(string[] args)
        {
            string fileNameExisting = @"form.pdf";
            string fileNameNew = @"newform.pdf";

            using (var existingFileStream = new FileStream(fileNameExisting, FileMode.Open))
            using (var newFileStream = new FileStream(fileNameNew, FileMode.Create))
            {
                var pdfReader = new PdfReader(existingFileStream);
                using (var stamper = new PdfStamper(pdfReader, newFileStream))
                {
                    //نکته مهم جهت کار با اطلاعات فارسی
                    //در غیراینصورت شاهد ثبت اطلاعات نخواهید بود
                    stamper.AcroFields.AddSubstitutionFont(GetTahoma().BaseFont);

                    //تمام فیلدهای موجود در فرم
                    var form = stamper.AcroFields;

                    //مقدار دهی فیلدهای فرم
                    form.SetField("TextBox1", "مقدار1");
                }
            }
        }
    }
}
```


نظرات خوانندگان

نویسنده: باربد

تاریخ: ۱۳۹۱/۰۷/۱۰ ۱۰:۸

سلام

آیا میشه از طریق همین ادیتور ، پارامتر تصویری هم به فایل اضافه کرد؟

(مثلا عکس شخص ...)

- یک مشکل : وقتی Open office را دانلود میکنم ، موقع اجرا پیغام میده که مشکل داره و کامل دانلود نشده ، در صورتیکه اینجوری نیست . (حجم فایلش هم 130 MB هست)
سپاسگزارم

نویسنده: وحید نصیری

تاریخ: ۱۳۹۱/۰۷/۱۰ ۱۰:۵۶

- در مورد اجرا نشدن برنامه نصاب نظری ندارم. عموما این فایلها دارای یک امضای دیجیتالی md5 یا sha1 منتشر شده در سایت اصلی هم هستند. مقایسه کنید آیا کامل دریافت شده یا نه.

- در مورد تصویر، میتونید از روشهای متداول iTextSharp استفاده کنید. PDF در اصل یک قالب برداری است. شما یک Canvas دارید که میتونید در هر جایی از آن هر شیءایی را قرار دهید. برای نمونه در مثال فوق:

```
PdfContentByte content = stamper.GetOverContent(pdfReader.NumberOfPages);  
Image image = Image.GetInstance(imagePath);  
image.SetAbsolutePosition(450,650);  
image.ScaleAbsolute(200,200);  
content.AddImage(image);
```

شما به کمک stamper دسترسی به این Canvas پیدا می کنید. سپس در هر مختصات دلخواهی مطابق کدهای فوق، تصویر مورد نظر را قرار دهید.

نویسنده: باربد

تاریخ: ۱۳۹۱/۰۷/۱۰ ۱۱:۳۷

سپاس آقای نصیری

نویسنده: M.B

تاریخ: ۱۳۹۱/۰۷/۲۸ ۱۰:۵۱

با سلام، من زمانی که کدهای مربوطه را می نویسم و فرم رو اجرا می کنم بلافاصله یک فایل PDF برای من باز میشه که در اون کلمه آزمایش نوشته شده است کدهای من به صورت زیر هستند

```
public static Font GetTahoma()  
{  
    var fontName = "Tahoma";  
    if (!FontFactory.IsRegistered(fontName))  
    {  
        var fontPath = Environment.GetEnvironmentVariable("SystemRoot") + "\\fonts\\tahoma.ttf";  
        FontFactory.Register(fontPath);  
    }  
    return FontFactory.GetFont(fontName, BaseFont.IDENTITY_H, BaseFont.EMBEDDED);  
}
```

```
string fileNameExisting = @"Test.pdf";  
string fileNameNew = @"newform.pdf";  
using (var existingFileStream = new FileStream(fileNameExisting, FileMode.Open))  
    using (var newFileStream = new FileStream(fileNameNew, FileMode.Create))
```

```
{
    var pdfReader = new PdfReader(existingFileStream);
    using (var stamper = new PdfStamper(pdfReader, newFileStream))
    {
        // نکته مهم جهت کار با اطلاعات فارسی
        // در غیراینصورت شاهد ثبت اطلاعات نخواهید بود
        stamper.AcroFields.AddSubstitutionFont(GetTahoma().BaseFont);

        // تمام فیلدهای موجود در فرم
        var form = stamper.AcroFields;

        // مقدار دهی فیلدهای فرم
        form.SetField("Text1", "1مقدار");
        form.SetField("Text2", "2مقدار");
        form.SetField("Text3", "3مقدار");
        form.SetField("Text4", "4مقدار");
        form.SetField("Text5", "5مقدار");
        form.SetField("Text6", "6مقدار");

        // به این ترتیب فرم دیگر توسط کاربر قابل ویرایش نخواهد بود
        // stamper.PartialFormFlattening --> جهت غیرقابل ویرایش نمودن فیلدی مشخص
        stamper.FormFlattening = true;

        stamper.Close();
        pdfReader.Close();
    }
}

Process.Start("newform.pdf");
```

و محتوای فایل Test.PDF

	شماره شناسنامه		شماره پرسنلی
	کد ملی		نام و نام خانوادگی
	مقطع		نام پدر

و محتوای فایل جدید که برای من ایجاد می‌کند

آزمایش

ممنون .

نویسنده: وحید نصیری
تاریخ: ۱۱:۲۱ ۱۳۹۱/۰۷/۲۸

در سیستم شما تداخل وجود دارد. کلمه «آزمایش» متعلق به فایل قبلی دیگری است که دارید. فایل‌های آزمایشی خروجی PDF موجود را کلاً حذف کنید و بعد کدهای فوق را اجرا کنید. Process.Start را هم حذف کرده و خروجی را دستی و خارج از VS.NET بررسی کنید.

نویسنده: پویا امینی
تاریخ: ۱۱:۴۰ ۱۳۹۱/۰۷/۲۸

بخشید آقای نصیری من وقتی Process.Start رو حذف کنم کجا می‌تونم محتوای فایل NewForm.PDF رو ببینم؟ چون درون Sloution چنین فایلی برای من ایجاد نمی‌شود.

نویسنده: وحید نصیری
تاریخ: ۱۱:۵۳ ۱۳۹۱/۰۷/۲۸

معمولاً درون پوشه bin\debug یا bin\release تشکیل می‌شود. اما جهت اطمینان می‌تونید مسیر دهی کامل کنید:

```
string fileNameExisting = @"c:\Test.pdf";  
string fileNameNew = @"c:\newform.pdf";
```

نویسنده: پویا امینی
تاریخ: ۰:۳۲ ۱۳۹۱/۰۷/۲۹

خیلی کارت درسته جناب نصیری، من اومدم مسیر دهی خودم رو به صورت زیر انجام دادم

```
string fileNameExisting = HttpRuntime.AppDomainAppPath + @"TestOpenOffice.pdf";  
string fileNameNew = HttpRuntime.AppDomainAppPath + @"newform.pdf";
```

و Process.Start

```
Process.Start(HttpRuntime.AppDomainAppPath+"newform.pdf");
```

یه دنیا ممنون جناب نصیری

نویسنده: وحید نصیری
تاریخ: ۰:۴۵ ۱۳۹۱/۰۷/۲۹

اگر برنامه ویندوزی است بهتره از Application.StartupPath استفاده کنید (تعریف شده در اسمبلی System.Windows.Forms). اگر برنامه وب است، از Server.MapPath استفاده کنید.