

یکی از امکاناتی که در نرم افزارهای اتوماسیون مورد نیاز است، ذخیره اطلاعات، داخل فایل اکسل است و در صورتی که حجم این اطلاعات زیاد باشد زمان زیادی صرف این عمل خواهد شد. در زیر کلاسی را برای شما آماده نموده‌ام که 20 هزار رکورد را در 4 ثانیه، در فایل اکسل ذخیره می‌نماید. در این روش با استفاده از یک آرایه به نام rawdata این عمل انجام شده. توضیحات کدها نیز به صورت comment در کنار کدها آورده شده است.

```
//using System;
//using System.Data;
//using Microsoft.Office.Interop.Excel;

class FastExportingMethod
{
    //دیتابیل که حاوی اطلاعات می‌باشد dt= System.Data.DataTable
    //مسیر ذخیره شدن outputPath=
    public static string ExportToExcel(System.Data.DataTable dt, string outputPath)
    {
        try
        {
            //ساخت یک شی اکسل
            ApplicationClass excelApp = new ApplicationClass();

            //جدید Workbook ساخت یک
            Workbook excelWorkbook = excelApp.Workbooks.Add(Type.Missing);

            int sheetIndex = 0;

            //ساخت آرایه به طول تعداد سطرهای دیتابیل+1 و تعداد ستونهای دیتابیل
            object[,] rawData = new object[dt.Rows.Count + 1, dt.Columns.Count];

            //کپی نام ستونهای دیتابیل به عنوان هدر برای فایل اکسل در اولین سطر از آرایه
            for (int col = 0; col < dt.Columns.Count; col++)
            {
                rawData[0, col] = dt.Columns[col].ColumnName;
            }

            //کپی اطلاعات دیتابیل به داخل آرایه
            for (int col = 0; col < dt.Columns.Count; col++)
            {
                for (int row = 0; row < dt.Rows.Count; row++)
                {
                    rawData[row + 1, col] = dt.Rows[row].ItemArray[col].ToString();
                }
            }

            //محاسبه نام ستونهای اکسل

            string finalColLetter = string.Empty;
            string colCharset = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
            int colCharsetLen = colCharset.Length;

            if (dt.Columns.Count > colCharsetLen)
            {
                finalColLetter = colCharset.Substring((dt.Columns.Count - 1) / colCharsetLen - 1,
1);
            }

            finalColLetter += colCharset.Substring((dt.Columns.Count - 1) % colCharsetLen, 1);

            //ساخت یک Sheet
            Worksheet excelSheet = (Worksheet)excelWorkbook.Sheets.Add(
                excelWorkbook.Sheets.get_Item(++sheetIndex),
                Type.Missing, 1, XlSheetType.xlWorksheet);
            //تنظیم نام شیت به نام دلخواه
            excelSheet.Name = "List";
            //تنظیم خاصیت راست به چپ برای نمایش اطلاعات
            excelSheet.DisplayRightToLeft = true;

            //تعیین محدوده سطرها و ستونها
            string excelRange = string.Format("A1:{0}{1}", finalColLetter, dt.Rows.Count + 1);
            //انتقال اطلاعات از آرایه به شیت مورد نظر
```

```
excelSheet.get_Range(excelRange, Type.Missing).Value2 = rawData;

// ضخیم کردن اولین سطر برای عنوان ستونها
((Range)excelSheet.Rows[1, Type.Missing]).Font.Bold = true;

// تنظیم عرض ستونها به اندازه محتوای ستونها
for (int col = 0; col < dt.Columns.Count; col++)
{
    ((Range)excelSheet.Columns[col + 1]).EntireColumn.AutoFit();
}

// بستن Workbook
excelWorkbook.SaveAs(outputPath, XlFileFormat.xlWorkbookNormal, Type.Missing,
    Type.Missing, Type.Missing, Type.Missing, XlSaveAsAccessMode.xlExclusive,
    Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing);
excelWorkbook.Close(true, Type.Missing, Type.Missing);
excelWorkbook = null;

excelApp.Quit();
excelApp = null;

// Collect the unreferenced objects
GC.Collect();
GC.WaitForPendingFinalizers();

return "اطلاعات شما در مسیر انتخاب شده ذخیره گردید";
}
catch (Exception ex)
{
    // بدست آوردن کد خطا برای مدیریت خطاها
    int code = System.Runtime.InteropServices.Marshal.GetExceptionCode();

    return ex.Message + code;
}
}
```

نظرات خوانندگان

نویسنده: راضیه

تاریخ: ۱۹:۵۶ ۱۳۹۲/۱۱/۱۷

کتابخانه NPOI نیز دارای سرعت بسیار بالایی است. پیشنهاد می‌کنم حتما امتحانش کنید

[/http://npoi.codeplex.com](http://npoi.codeplex.com)

نمونه: [/http://www.zachhunter.com/2010/05/getting-started-with-npoi](http://www.zachhunter.com/2010/05/getting-started-with-npoi)

نویسنده: وحید نصیری

تاریخ: ۲۰:۲ ۱۳۹۲/۱۱/۱۷

برای کار با اکسل، اگر 2007 به بعد مدنظر است، بدون نیاز به نصب اکسل می‌شود با [EPP1us](#) هم کار کرد.