

# ExcelUtility 类库操作说明（动态生成多工作簿示例）

设计/开发：左文俊

新增方法一：由 GetFormatterContainer Func 委托导出基于 EXCEL 模板的多 Sheet 文件，方法定义如下：

```
/// <summary>
/// 由 GetFormatterContainer Func 委托导出基于 EXCEL 模板的多工作簿文件
/// </summary>
/// <typeparam name="T">数据源可枚举项类型</typeparam>
/// <param name="templatePath">模板路径</param>
/// <param name="sheetName">模板中使用的工作簿名称</param>
/// <param name="dataSource">数据源</param>
/// <param name="getFormatterContainer">生成模板数据格式化容器（SheetFormatterContainer）委托,在委托方法中实现模板的格式
化过程</param>
/// <param name="sheetSize">每个工作簿显示的数据记录数</param>
/// <param name="filePath">导出路径， 可选</param>
/// <returns></returns>
public static string ToExcelWithTemplate<T>(string templatePath, string sheetName, IEnumerable<T> dataSource, Func<IEnumerable<T>,
SheetFormatterContainer> getFormatterContainer, int sheetSize, string filePath = null)
```

简要说明上述方法实现原理步骤：

- 1.指定模板路径、初始工作簿名称、导出的数据源、每个工作簿显示的记录数、封装生成模板数据格式化容器（SheetFormatterContainer）委托,在委托方法中实现模板的格式化过程;
- 2.依据每个工作簿显示的记录数，循环拆分数据源，并计算出需要的工作簿总数以及生成模板数据格式化容器字典（Key:Sheet 名称，Value:模板数据格式化容器对象）；
- 3.生成 2 中计算的所需的工作簿的临时模板文件（存放在系统的本地临时目录：Temp）
- 4.调用 ToExcelWithTemplate 的其它重载方法（基于模板+多 sheet 生成 EXCEL 方法）来完成 EXCEL 的导出；
- 5.无论最终导出成功与否，将删除临时模板及临时模板配置文件；

测试示例代码如下：

```
/// <summary>
/// 测试方法：测试依据模板+DataTable 来生成多工作簿的 EXCEL
/// </summary>
[TestMethod]
public void TestExportToExcelWithTemplateByDataTable2()
{
    DataTable dt = GetDataTable();
    string templateFilePath = AppDomain.CurrentDomain.BaseDirectory + "/excel.xls"; //获得 EXCEL 模板路径
    string excelPath = ExcelUtility.Export.ToExcelWithTemplate<DataRow>(templateFilePath, "table", dt.Select(), (data) =>
    {
        SheetFormatterContainer formatterContainers = new SheetFormatterContainer(); //实例化一个模板数据格式化容器
        PartFormatterBuilder partFormatterBuilder = new PartFormatterBuilder(); //实例化一个局部元素格式化器
        partFormatterBuilder.AddFormatter("Title", "跨越 IT 学员"); //将模板表格中 Title 的值设置为跨越 IT 学员
        formatterContainers.AppendFormatterBuilder(partFormatterBuilder); //添加到工作簿格式容器中，注意只有添加进去了才会生效
        CellFormatterBuilder cellFormatterBuilder = new CellFormatterBuilder(); //实例化一个单元格格式化器
        cellFormatterBuilder.AddFormatter("rptdate", DateTime.Today.ToString("yyyy-MM-dd HH:mm")); //将模板表格中 rptdate 的值设置为
当前日期
        formatterContainers.AppendFormatterBuilder(cellFormatterBuilder); //添加到工作簿格式容器中，注意只有添加进去了才会生效
    })
    excelPath;
```

//实例化一个表格格式化器，data 是已拆分后的数据源(这里是 10 条记录)，name 表示的模板表格中第一行第一个单元格要填充的数据参数名

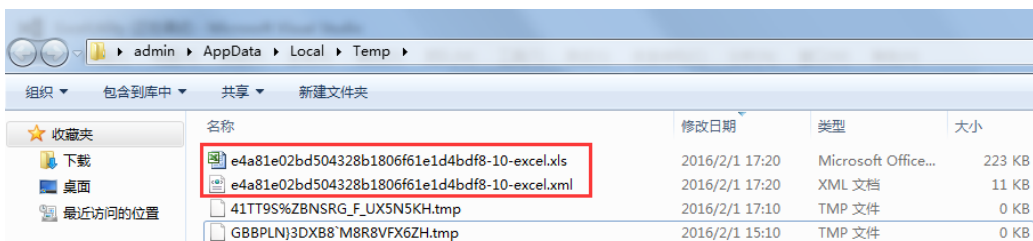
TableFormatterBuilder<DataRow> tableFormatterBuilder = new TableFormatterBuilder<DataRow>(data, "name");//这里的数据源设置：data 是重点

```
tableFormatterBuilder.AddFormatters(new Dictionary<string, Func<DataRow, object>>{
    {"name", r=>r["Col1"]},//将模板表格中 name 对应 DataTable 中的列 Col1
    {"sex", r=>r["Col2"]},//将模板表格中 sex 对应 DataTable 中的列 Col2
    {"km", r=>r["Col3"]},//将模板表格中 km 对应 DataTable 中的列 Col3
    {"score", r=>r["Col4"]},//将模板表格中 score 对应 DataTable 中的列 Col4
    {"result", r=>r["Col5"]},//将模板表格中 result 对应 DataTable 中的列 Col5
});

formatterContainers.AppendFormatterBuilder(tableFormatterBuilder);//添加到工作簿格式容器中，注意只有添加进去了才会生效
return formatterContainers;//返回一个模板数据格式化容器
}, 10);//注意这里的 10 表示动态生成新的工作簿，且每个工作簿显示 10 条记录
Assert.IsTrue(File.Exists(excelPath));
}
```

测试结果如下：

1. 生成的临时模板及模板配置文件：



2. 导出的 EXCE 结果如下：



3. 临时模板及模板配置文件在导出结束后自动被清除。

新增方法二：增加由 DataTable 导出多 Sheet Excel 方法（准确的说是修改 ToExcel 方法，增加一个 sheetSize 参数），方法定义如下：

```
/// <summary>
/// 由 DataTable 导出 Excel
/// </summary>
/// <param name="sourceTable">要导出数据的 DataTable</param>
/// <param name="sheetName">工作簿名称，可选</param>
```

修改代码部份说明：增加依据指定的每个工作簿显示的记录数(sheetSize)来循环拆分数据源及创建多个工作簿；

```

/// <summary>
/// 测试方法：测试将 DataTable 导出到多工作簿 EXCEL
/// </summary>
[TestMethod]
public void TestExportToExcelByDataTable8()
{
    DataTable dt = GetDataTable();

    string excelPath = ExcelUtility.Export.ToExcel(dt, "sheet", sheetSize: 10); //指定每个工作簿显示的记录数

    Assert.IsTrue(File.Exists(excelPath));
}

```

	A1	B1	C1	D1	E1	F1	G1	H1	I1
1	Col1	Col2	Col3	Col4	Col5	Col6	Col7	Col8	
2	Name1	男	科目1	1.00	待定测测测测测测测测	fa577ac171e74f74b0093abdc3f11e84			
3	Name2	女	科目2	2.00	待定测测测测测测测测	56519b0e12d34d6eab3f2bd692c64efb			
4	Name3	男	科目3	3.00	待定测测测测测测测测	df0273ab31744d389c134b0ea2c4e577			
5	Name4	女	科目4	4.00	待定测测测测测测测测	b71bd4942bf94b549b11fb593475cc23			
6	Name5	男	科目5	5.00	待定测测测测测测测测	4586badc7b7046efaf956b695bcd2026			
7	Name6	女	科目6	6.00	待定测测测测测测测测	e68c416547f84e4a9b19b0805cebd35e			
8	Name7	男	科目7	7.00	待定测测测测测测测测	789e574142ae4e68b8c69bc9146c820e			
9	Name8	女	科目8	8.00	待定测测测测测测测测	73d17c9f11814f2f95f1579054d6c81d			
10	Name9	男	科目9	9.00	待定测测测测测测测测	7ce95017f11449a6ab6cf32cdf22e17			
11	Name10	女	科目10	10.00	待定测测测测测测测测	f991d96aa8f5471da47573ea42072c75			
12									
13									
14									

sheet1 / sheet2 / sheet3 / sheet4 / sheet5 / sheet6 / sheet7 / sheet8 / sheet9 / sheet10

就绪