

ASP.NET 编程中对 Excel 文档操作的探讨及应用^①

张文博 余文芳 (武汉通信指挥学院 6 系 22 队 湖北 武汉 430010)

摘要: 通过对 Asp.net 编程中常用的操作 Excel 文档方法的探讨, 得出了两种实际可行的方法, 并在具体项目中得到了应用, 实现了 Excel 文档导入导出数据库的功能, 解决了数据库系统开发中的一个实际问题。

关键词: Asp.net; Excel; 数据库

Discussion and Application of Excel Documents in ASP.NET Programming

ZHANG Wen-Bo, YU Wen-Fang

(Sixth Department, Team 22, Commanding Communications Academy, Wuhan 430010, China)

Abstract: From the discussion of how to operate Excel in asp.net programming, two feasible methods are found and applied in a specific project. The implementation of exporting Excel documents from the database and importing Excel documents to the database is realized, which solves a practical problem in asp.net programming.

Keywords: Asp.net; Excel; database

1 引言

在 Asp.net 开发中, 我们经常会使用到各种数据库系统, 但用户对这些数据库的应用不太熟悉, 一些用户可能希望将处理好的 Excel 文档数据导入到已有的数据库系统中, 以便对数据进行维护、更新; 还有用户可能希望将数据库中的数据导出到熟悉的 Excel 文档中, 利用 Excel 强大的功能对数据进行统计、分析等操作。所以在开发过程中有必要对此功能进行实现。

2 Asp.net 下操作 Excel 文档的两种常用方法

通过在实际中的归纳总结, 笔者发现在 Asp.net 下对 Excel 文档的操作有以下两种常用的方法:

2.1 将 Excel 文档作为数据源进行操作

在 .net 环境下, 可以直接以数据源的形式操作 Excel, 和操作 Access 数据库类似, 只是在一些细节方面稍有改变。我们这样连接 Excel 数据源: Pro-

vider=Microsoft.Jet.OLEDB.4.0;Data Source=xls-Path;Extended Properties=Excel8.0, 其中 xlsPath 为 Excel 文档路径。

上面的连接字符串可能我们已经很熟悉了, 需要注意的是这里要设置“Extended Properties”为 Excel 的版本, 现在已经有的版本为 5.0、7.0、8.0 等。Extended Properties 还有另外一个参数, 用来区别数据标题和数据值。我们知道, Excel 对每一列取名和 Access 不一样, Excel 中列的名字和数据的值没有区别, 都是放在数据区中, 这里就可以设置一个 HDR 参数, 该参数可以设置为“YES”或者“NO”, 如果设置为“Yes”, 数据就是按照 Excel 的实际数据格式现实, 如果设置为“NO”, 会在每列数据加一个“F1”、“F2”等的数据头信息。

设置好连接字符串后就可以像操作一般的数据库一样对 Excel 文档进行操作了。

^① 收稿时间:2009-06-16

2.2 调用 com 组件对 Excel 文档进行操作

在 .net 环境下可调用 Excel 的 com 组件对 Excel 文档进行操作, 在 com 组件中定义了一个命名空间 Excel, 在此空间中封装了 4 个对象: [1]

(1) Application 对象

Application 对象处于 Excel 对象层次结构的顶层, 表示 Excel 自身的运行环境。

(2) WorkBook 对象

WorkBook 对象直接处于 Application 对象的下一层, 表示一个 Excel 工作簿文件。

(3) WorkSheet 对象

WorkSheet 对象包含于 WorkBook 对象, 表示一个 Excel 工作表。

(4) Range 对象

Range 对象包含于 WorkSheet 对象, 表示 Excel 工作表的一个或多个单元格。

下面的代码可以实现对 Excel 文档的简单操作:[2]

```
Excel.Application excel=new Excel.Application(); //实例化一个 Excel 对象
excel.Workbooks.Add(true); // 创建一个 Excel 工作簿
excel.Visible = true; //使 Excel 可视
```

3 Asp.net下实现Excel文档导入导出数据库

在 Asp.net 编程中, 可以通过使用 SQL 语句对数据库中的数据进行添加、修改、删除等操作, 在操作尚未执行之前, 数据是以 DataSet 或者 DataTable 的形式存在于内存中的。也就是说, DataSet(或者 DataTable)在 Excel 导入导出到数据库的过程中起到了桥梁作用。

不管使用何种数据库系统, 如果能以 DataSet 形式从 Excel 文档中读取数据, 就能实现数据的导入; 反之, 如果能够将 DataSet 形式的数据写入到 Excel, 则能实现数据的导出。

基于上述原理, 我们很容易实现某院校“综合教务管理信息系统”中“成绩导入导出”模块的功能。如图 1 所示:



图 1 综合教务管理信息系统成绩导入导出页面

3.1 将 Excel 文档中学生成绩导入到数据库

图 2 所示为一份学生成绩单的 Excel 文档, 其中“学号”、“成绩”分别对应数据库中的两个字段。

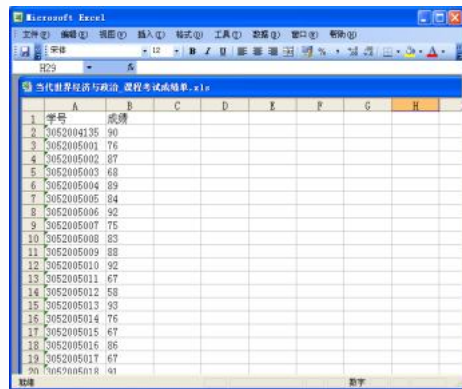


图 2 学生成绩单的 Excel 文档

教师可以将 Excel 文档形式的成绩单先上传到数据库服务器, 服务器端将该 Excel 文档作为数据源, 以 DataTable 形式读取其中学生成绩数据, 然后将 DataTable 中的数据写入到数据库中。服务端程序读取 Excel 文档的代码如下:

```
private DataTable GetXlsData(string xlsPath)
{
    String conStrXls_F = "Provider=Microsoft.
Jet.OLEDB.4.0;Extended Properties='Excel8.0;HDR
=Yes;IMEX=1\';Data Source='{0}\'";
    string conStr=string.Format(conStrXls_F,xls-
Path);
    string sqlStr = "select * from [sheet1$]";
    OleDbDataAdapter da = new OleDbData
```

```

Adapter();
    DataTable dt = new DataTable();
    OleDbConnection con = null;
    Try
    {
        con = new OleDbConnection(conStr);
        da.SelectCommand = new OleDb
Command(sqlStr, con);
        da.Fill(dt);
        return dt;
    }
    catch (Exception ex)
    {
        if (con.State == ConnectionState.Open)
        {
            con.Close();
        }
        return null;
    }
}

```

```

//DataTable 行数
int colCount = dt.Columns.Count;
//DataTable 列数
//利用二维数组批量写入
string[,] arr = new string[rowCount, colCount];
for (int j = 0; j < rowCount; j++)
{
    for (int k = 0; k < colCount; k++)
    {
        arr[j, k] = dt.Rows[j][k].ToString();
    }
}
range = (Excel.Range)workSheet. Cells[top,
left];
range = range.get_Resize(rowCount,
colCount);
range.Value = arr;
}

```

3.2 将数据库中成绩导出到 Excel 文档

当用户从服务器导出成绩时，服务器端先从数据库查找用户请求的数据，存放在 **DataTable** 中，然后调用 **com** 组件对 **Excel** 文档进行操作，将数据写入到 **Excel** 文档。相关代码如下：

```

public void DataTableToExcel(DataTable dt, int
top, int left)
{
    int rowCount = dt.Rows.Count;

```

4 结语

本文针对 **Asp.net** 编程中对 **Excel** 文档操作两种常用方法，实现了 **Excel** 文档导入导出数据库的功能，由于现代化办公环境下 **Excel** 文档的广泛使用，因此具有一定的实用性。

参考文献

- 1 陈卫军. 基于 C#.NET 的 Excel 编程. 科教文汇, 2008, (3):193-194.
- 2 柳盛, 王东明. 基于 C# 的 Excel 报表自动生成. 科技信息, 2008, (7):4.