ExcelTemplate Output Options

ExcelTemplate has several different options for saving an ExcelTemplate generated file.

After creating a workbook with ExcelTemplate, you can:

- Save the output to disk
- Write the workbook to a stream
- Stream the workbook to the client and open it in Excel
- Stream the workbook to the client and open it in the browser
- Pass the Template to ExcelApplication

Save the output to disk

Save(outputFileName)

This method saves the generated Excel file on the server.

outputFileName specifies a complete path and file name for the generated file. ExcelWriter will save the file to this location. If a file with the same name exists, it will be overwritten by the new Excel file.

Example

```
excelTmpl.Save(@"C:\Reports\Orders.xls");
```

```
excelTmpl.Save(@"C:\Reports\Orders.xls")
```

Write the workbook to a stream

Save(outputStream)

This method writes the generated Excel file to the specified System.IO.Stream, or a class derived from System.IO.Stream (for example, System.IO.FileStream).

If you pass Save a System.IO.FileStream, ExcelWriter will save the generated file on the server. If you pass Save Response.OutputStream, ExcelWriter will stream the the generated file to the client.

Example

```
//-- FileStream and FileMode are in the System.IO namespace
FileStream fstream = new FileStream(@"C:\temp\outfile.xls", FileMode.Create);

//-- Pass the FileStream to ExcelTemplate
excelTmpl.Save(fstream);

//-- Close the FileStream (could be in a finally block)
fstream.Close();
```

```
'-- FileStream and FileMode are in the System.IO namespace
Dim fstream As New FileStream("C:\temp\outfile.xls", FileMode.Create)

'-- Pass the FileStream to ExcelTemplate
excelTmpl.Save(fstream)

'-- Close the FileStream (could be in a finally block)
fstream.Close()
```

Stream the workbook to the client and open it in Excel

Save(response)

This method streams the generated Excel file to the client.

If you pass Save an HttpResponse object, ExcelWriter will stream the generated file to the client. If the user chooses to open (rather than save) the file, it will open in the browser window. The browser will use the ExcelWriter script name as the default file name displayed in the File Download dialog. To set a different file name and/or to open the file in Microsoft Excel, use the signature Save(response, attachmentName, OpenInBrowser).

Example

```
excelTmpl.Save(Page.Response);
```

```
excelTmpl.Save(Page.Response)
```

Stream the workbook to the client and open it in the browser

Save(response, attachmentName, OpenInBrowser)

This method streams the generated Excel file to the client.

If you pass Save an HttpResponse object, ExcelWriter will stream the generated file to the client. This method allows you to specify a default client-side file name, and whether the file should be opened in the browser window or in Microsoft Excel.

The parameter attachmentName specifies a name for the generated Excel file; this name will be displayed in the download dialog when the file is streamed to the browser. If the parameter openInBrowser is set to true, and the user chooses to open the file, the file will open in the browser window. If openInBrowser is set to false, and the user chooses to open the file, the file will open in Microsoft Excel. By default, the file will open in the browser window. Note: not all browsers can embed an Excel file in the browser window.

Example

```
excelTmpl.Save(Page.Response, "Output.xls", false);
```

```
excelTmpl.Save(Page.Response, "Output.xls", False)
```

Pass the Template to ExcelApplication

You can use ExcelTemplate to open and populate an ExcelWriter template, then pass the populated workbook to ExcelApplication for additional processing. In this case, do not call Save. Instead, pass the ExcelTemplate object to ExcelApplication's open method:

Example

```
 ExampleExcelTemplate xlt = new ExcelTemplate();

xlt.Process();

ExcelApplication xla = new ExcelApplication();
Workbook wb = xla.Open(xlt);
```