

# Modifying an Existing Spreadsheet

ExcelApplication can open and modify an existing spreadsheet. The new spreadsheet can either be saved to disk (overwriting the original file or saved with a new name) or streamed to the browser. This allows you to use a preset format for similar spreadsheets, rather than recreate the format for each.

Alternatively, use an [ExcelWriter template](#).

All features in the original workbook - including macros, charts, PivotTables, and VBA - will be preserved in the new workbook.

## How to Modify an Existing Spreadsheet

To modify an existing spreadsheet:

1. Create an instance of ExcelApplication:

```
ExcelApplication xla = new ExcelApplication();
```

2. Open an Excel workbook from a file path or a System.IO.Stream:

```
//--- Open a workbook from a file path:  
Workbook wb = xla.Open(@"C:\Reports\Report.xlsx");
```

Or:

```
//--- Open a workbook from a Stream:  
FileStream fs = new FileStream(@"C:\Sales\2003\June.xlsx", FileMode.Open);  
Workbook wb = xla.Open(fs);
```

3. Return a Worksheet object:

```
Worksheet ws = wb.Worksheet[0];
```

4. Modify the worksheet. For example, add a cell value:

```
ws.Cells[0, 0].Value = "Welcome to SoftArtisans OfficeWriter";
```

5. Save the modified file with a new name, or stream it to the client:

```
//--- Save the file on the server:  
xla.Save(wb, @"C:\temp\out.xlsx");
```

Or:

```
//--- Stream the file to the client via HTTP:  
//--- false means Open In Excel, true means Open In Browser (IE only)  
xla.Save(wb, Page.Response, "out.xlsx", false);
```

For more information on save options, see [Output Options](#).