Excel to SharePoint List

Author: William Bechard – [ury2ok2000@gmail.com](mailto:ury2ok2000@gmail.com)

Description:

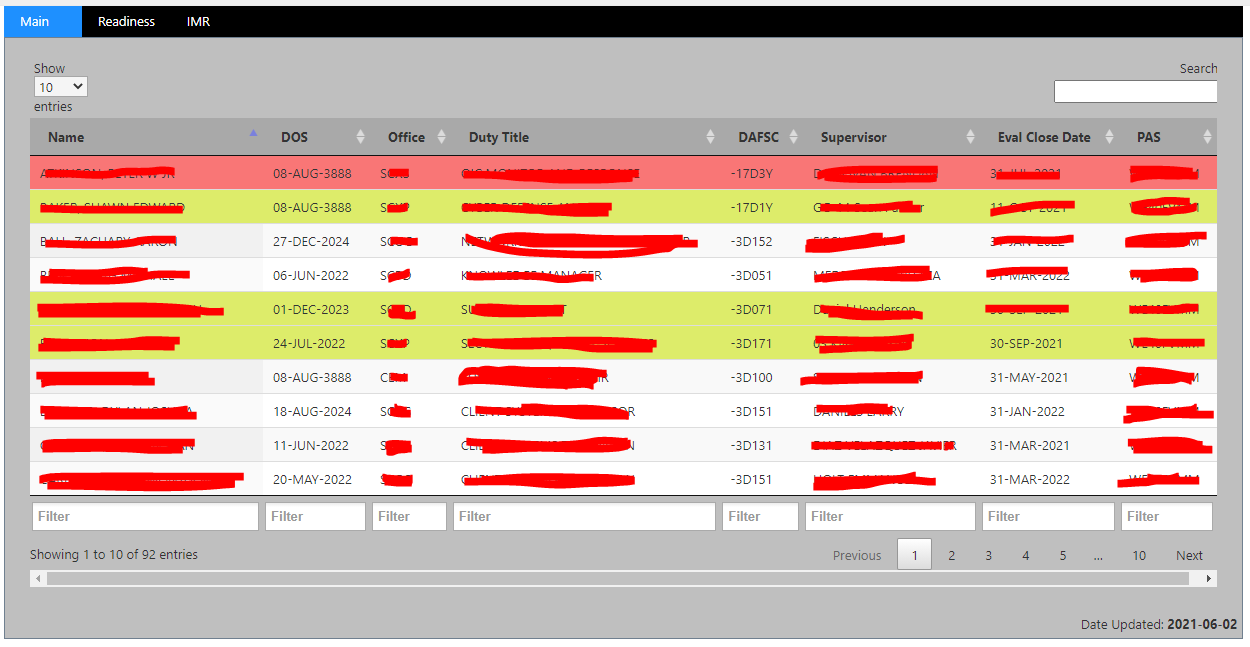


Image 1.0: result of the Main.html (Loaded from a Content Editor WebPart)

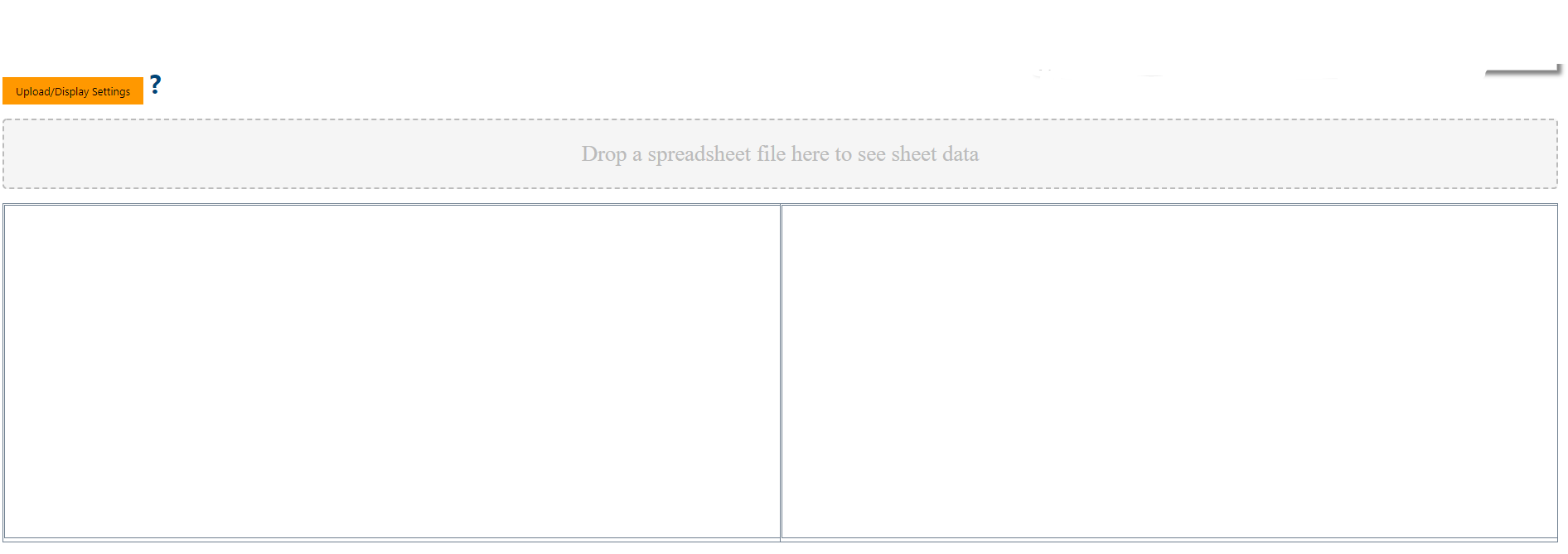


Image 1.1: result of the ExcelAdmin.html (Loaded from a Content Editor WebPart)

Drag and drop solution. After initial setup, an Excel (XLS, XLST) is dropped into a zone on the page (ExcelAdmin.html). From there the code puts it into a SharePoint list and displays it as a DataTable on the (Main.html) webPart. The program is broken into 2 html parts (and thus 2 webparts), so that the ExcelAdmin can be used on a webPart that uses a Target Audience (so that only those who would do the drag and drop would be able to change the setup configurations).

**Requirements**: (CDN)

* Jquery 3.6
* XLS / Shim.min.js
* XLSX.full.min.js
* Jquery-ui
* blockUI
* dataTables
* jquery.modal
* moment
* datetime-moment.js
* chosen.js
* w3.css

**Compatibility**: (Could work with other SP versions too, but only tested on the below)

* SP 2013
* SPO

**Setup**

****

1. Upload both HTML files



1. Add Content Editor Web Parts to your SharePoint Page, each referencing a different HTML file.

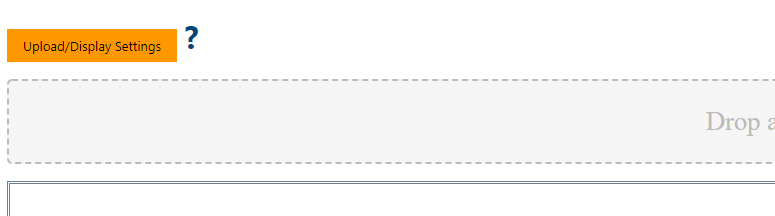
Recommend setting the bottom Web Part as the ExcelAdmin.html, and set its Target Audience setting to only those who should be dropping/configuring Excel files. In this way it is not visible to most users (so they cannot mess up any configurations)

Note: After the page is loaded for the first time 2 lists will be created in the Site Contents, ExcelData and ExcelSettings. The code creates these for you, you do not have to create them yourself.

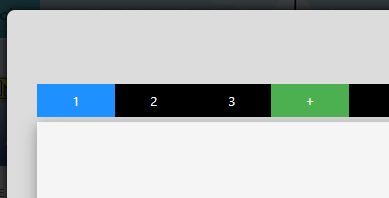
**How To Use**

A Configuration file needs to exist, before the drag and drop will know what to do with the Excel file that is dropped into the drop zone. Thus the first step is to create a configuration for a particular Excel file by clicking on the Upload/Display Settings orange button (which is part of the ExcelAdmin.html webPart).

**STEP 1:**

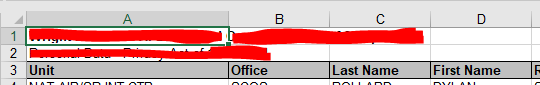


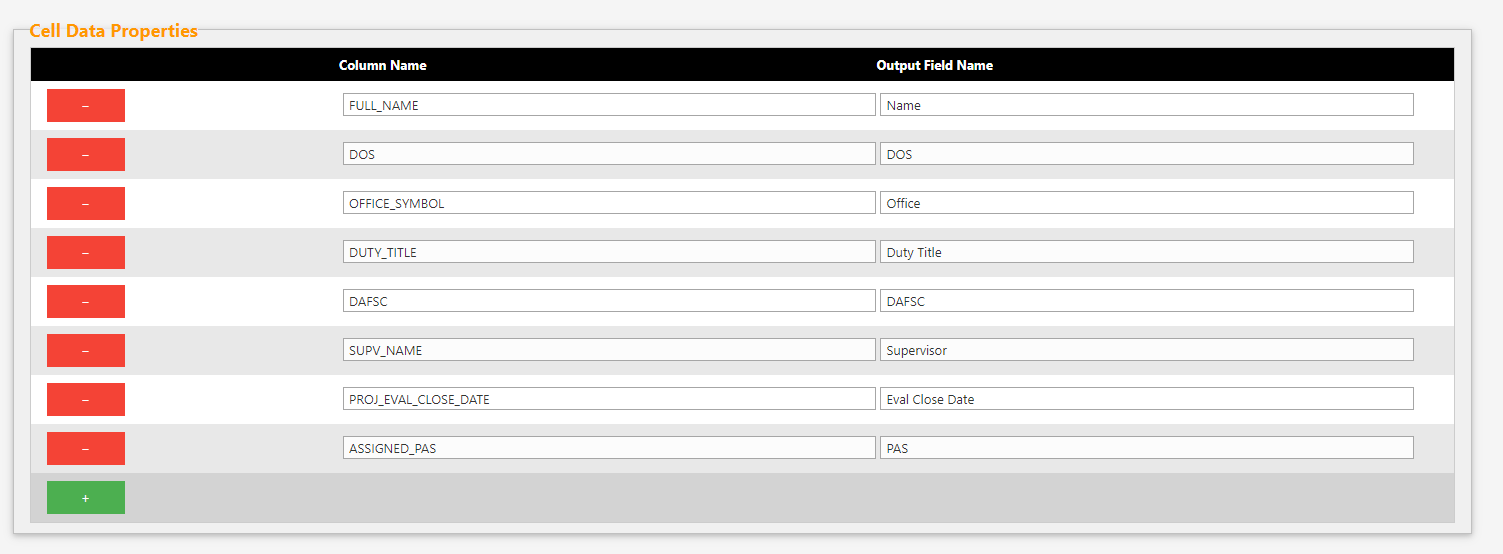
This will open a Window in which the settings can be configured. This window is broken up into **4** parts.

**Part 1** is the top navigation. Each TAB represents a different configuration file. You can click the + button to add a new file/tab which will represent a new Excel Configuration File (Otherwise you can click on the TAB of the Excel Configuration File that you want to EDIT).

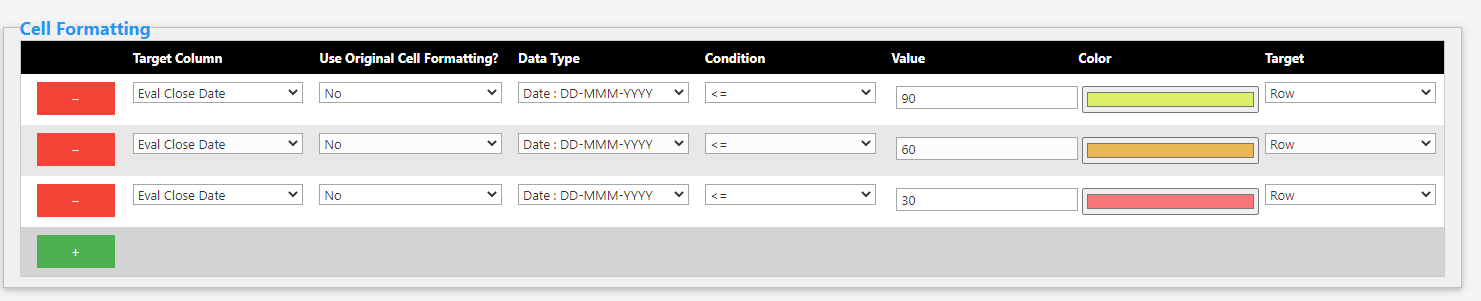


**Part 2** are the Excel Settings that relate to the FILE itself. Each setting I will go over:

* **File (Unique Keyword)** – A unique word that exists in the Excel’s File Name. This is how the code knows which configuration file matches which Excel file that is dropped.
* **Header Row** – The Row that contains the Header within the Excel file. In the below example it is Row 3, so the value here would be 3. 
* **Tab Position / Tab Title** – The position and title of the TAB on the Main Page (not this Setup page)
* **Compare (Output Name)-** Could be used in the future, not integrated now, but it is the Output Field Name of the Column that has unique values. The intent is it could be used in the future to determine if a record already exists (and whether to update it or replace it based on a future decision).
* **End Condition –** How will the code know what is the last ROW to grab data from. This will likely be either left blank (indicating that the first time the code encounters a ROW whose first column has no Data it should stop processing the Excel file, or some specified unique text (For a lot of the files I use this is a string of text, which must be copied EXACTLY that says: The information herein is For Official Use Only (FOUO)…..)
* **Target HTML Body / target HTML Nav-** Can leave as default values, unless you have custom HTML on the page where you would like to put the DataTables body and the Nav portion of the page (if that is the case you would put their unique HTML IDs in their respective spots).

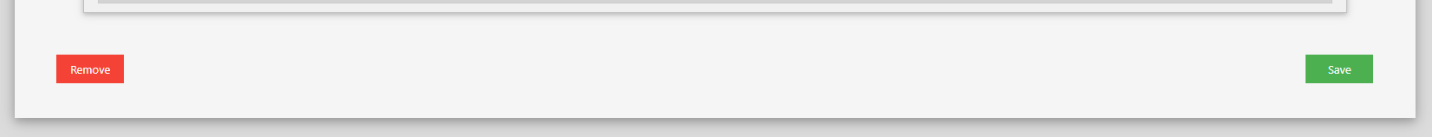
**Part 3**

This is the properties for the Excel Content (the data within the Excel pages). It exists to specify which Columns the program should use to Create the SharePoint List from. It does this based on the specific Column Name from the file (remember we specified the ROW this is on in the previous part, Header Row). This needs to match the Cell text exactly. The second parameter is the Output Field Name. This is the name we want for the DataTable on the main page. For example from the above picture you can tell that the First Column we want to read from has the specific name FULL\_NAME in the Header Row, however we would like that to be named just Name on the Data Table (on the main Page). To add another Column to capture, just click the Green + button (likewise clicking the Red – removes that Column).

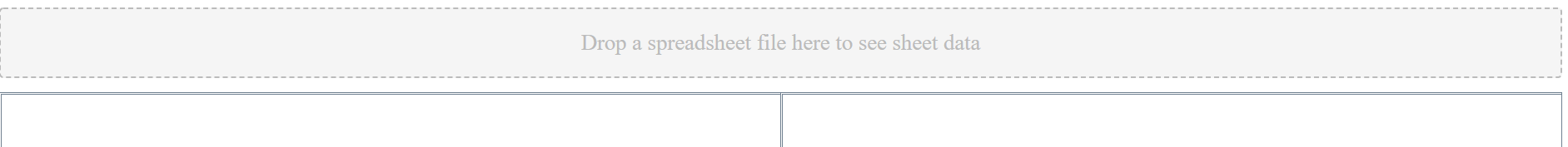
**Part 4**

Used for Formatting the data in the DataTable. Each row is a different Rule: The settings are:

* **Target Column**- pre-filled out drop-down of each Column from the Cell Data Properties (setup in Part 3). Simple select the Column that you are Targeting.
* **Use Original Cell Formatting** – If you want to use the colors/formatting from the original file for this Column then choose **Yes**, otherwise **No**.
* **Data Type** – specify the type of data we are comparing.
* **Condition** – specify the condition we are testing for
* **Value** – value we are testing for
* **Color** – if **TRUE** then this will be the color used
* **Target** – **Row** (Color the entire row), or **Cell** (Color just the cell)

When Done, simply click the Green Save button.

**STEP 2:**



Just drag and drop the Excel file into the Drop Zone. Multiple selected files can be dropped at one time, it will process each separately. The result(s) will be displayed in the 2 boxes below the Drop Zone.