# Luc Measurement Kinetic Organizer

To prepare tabular and tabular kinetic data for analysis in JMP



kenneth.berendzen@zmbp.uni-tuebingen.de

This is a GUI version of my scripts that convert plate reader output into a ready format needed by statistical programs like JMP®(SAS) or IBM®SPSS. The program is written in Java, but does not need an internet connection to run. It is just a GUI wrap for text manipulation. But, since it is a GUI, we can use drag drop functions to add our data. The data have to be prepared by the user as text files. This is very simple to do using the templates provided and following the instructions here. This program is the script mentioned in the publication: Wallmeroth and Jeschke et al., 2019; PMID: 30742656. This project is licensed under the GPLv3.

Program (jar), Source Code (zip), TEMPLATE files (xlsx), Tutorial (pdf), QuickGuide (pdf).

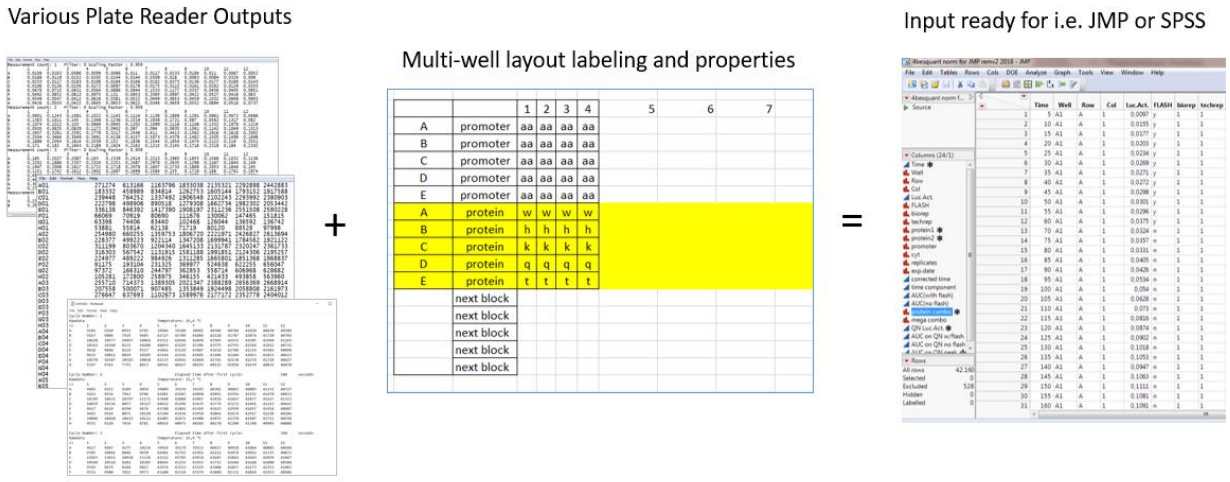



Figure 1. The Basic Idea

Your plate reader output needs to be graphed and analyzed. After drafting and the final output is a tab-delimited table that can be directly imported into JMP® or SPSS for analysis.

The idea is that graphing and primary statistical analysis in JMP® or SPSS is a much more intuitive and effective means of fondling your data to discover relationships as well as presenting it. But, input is made of columns of a specific datatype, followed by rows with the properties of individual items. When working with plate readers we can easily generate a lot of information, especially if one performs kinetic measurements, and this data is not usually in the format needed for JMP® or SPSS. One could move all the data by hand or use other programs/scripts to move the data around or even use the native analysis programs that many plate readers already come with, but the former is too time consuming and the

latter usually means that someone is blocking the machine. The Luc Measurement Kinetic Organizer solves this problem.

So, the data input and labels need to be in text files. A text file is a file that encodes only text, no colors, highlights, etc. This means that the basic idea is to open the output using a spreadsheet program of your choice and copy the data in the correct format into a text file using a simple text editor e.g. Notepad(Editor) or Notepad++. I recommend using UTF-8 format, but ASCII is also find in most cases. In addition, we need to have a way to label all of the wells in a multi-well plate. This is also done in a spread sheet program first, as it is much easier to label and mark things in color if it helps you keep track of what is what, and then copy the table into a text file as well. The text file will ignore all the color or lines of course. Once you have your data text file and your labels text file you are ready to use the program.

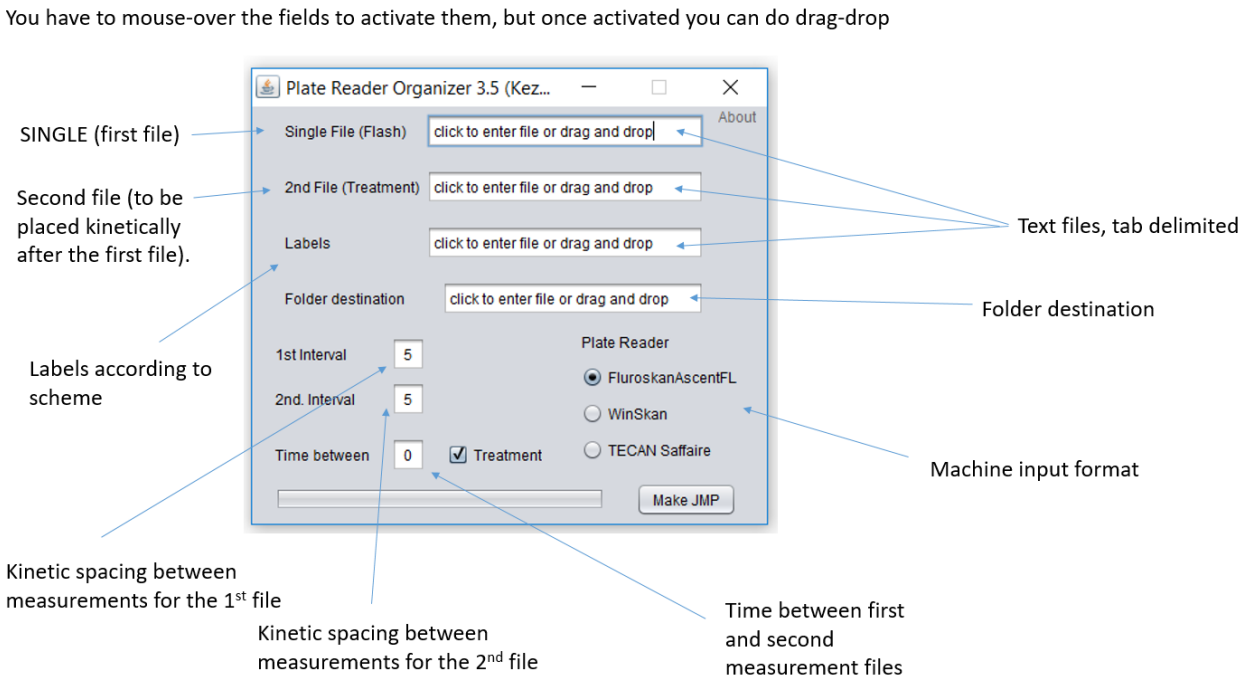



Figure 2. Explanation of the GUI interface.

## Preparing data input

The program accepts three input formats: FluoroSkanFL (Thermo Fisher Scientific), WinSkan (Berthold) format, and TECAN Safire. Keep in mind that my program is not affiliated with any of the companies mentioned here. If you don't have exactly the same machine, don't panic. It is likely that the same format is still used on your machine. You can find examples of all three formats in the TEMPLATE file. Just make sure that one of the formats matches. If you really like the program and don't want to program it yourself (the source files are available), then write me and send me an example of the output you have and I will consider adding it to the GUI.

The program was designed with kinetic experiments in mind. In fact, it was designed for the transfection assays in publication Wallmeroth and Jeschke et al., 2019 whereby there is an initial kinetic measurement was followed by a treatment kinetic measurement. Therefore, the program accepts two data files and concatenates them together with the time running after the first kinetic file and whatever time delay existed before running the second kinetic measurement. Your data does not need to be a kinetic measurement nor does it have to have a second treatment file. In this case, you will have a single output. The program will still add a time to it, but you can just delete it, as it would be meaningless.

## FluoroSkanFL format

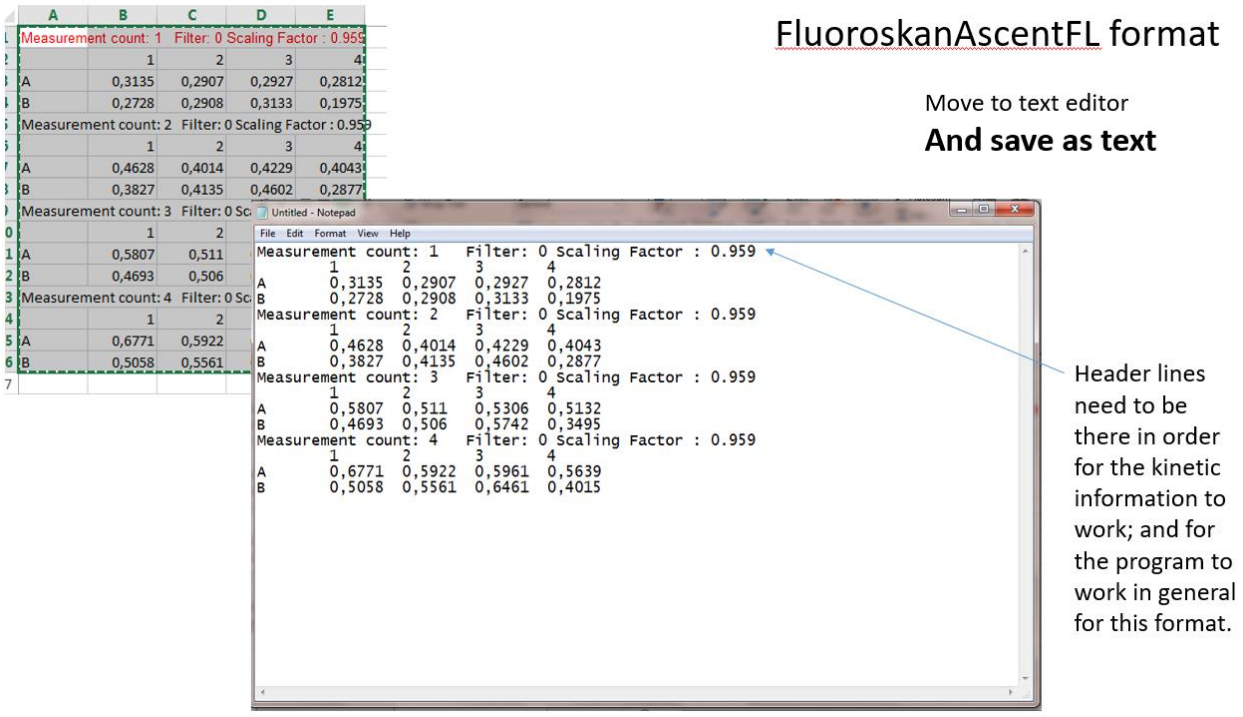



Figure 3. Preparing the FluoroSkanFL output.

The raw output file part containing the raw values is used as input for the program. A kinetic run will appear as above, whereas a single run will lack the line “Measurement Count: X  Filt…”. You need to add one to use the program for a single non-kinetic experiment. You can take from the template file. I left it like this since it is easy to add a line and I would have had to add another radio button, unnecessarily crowding the GUI. Keep in mind that the Tecan has a different header line from the FluoroSkanFL which you will see below. Observe that only the relevant data (highlighted) should be moved otherwise it will cause an error, do nonsense, or crash.

## TECAN format

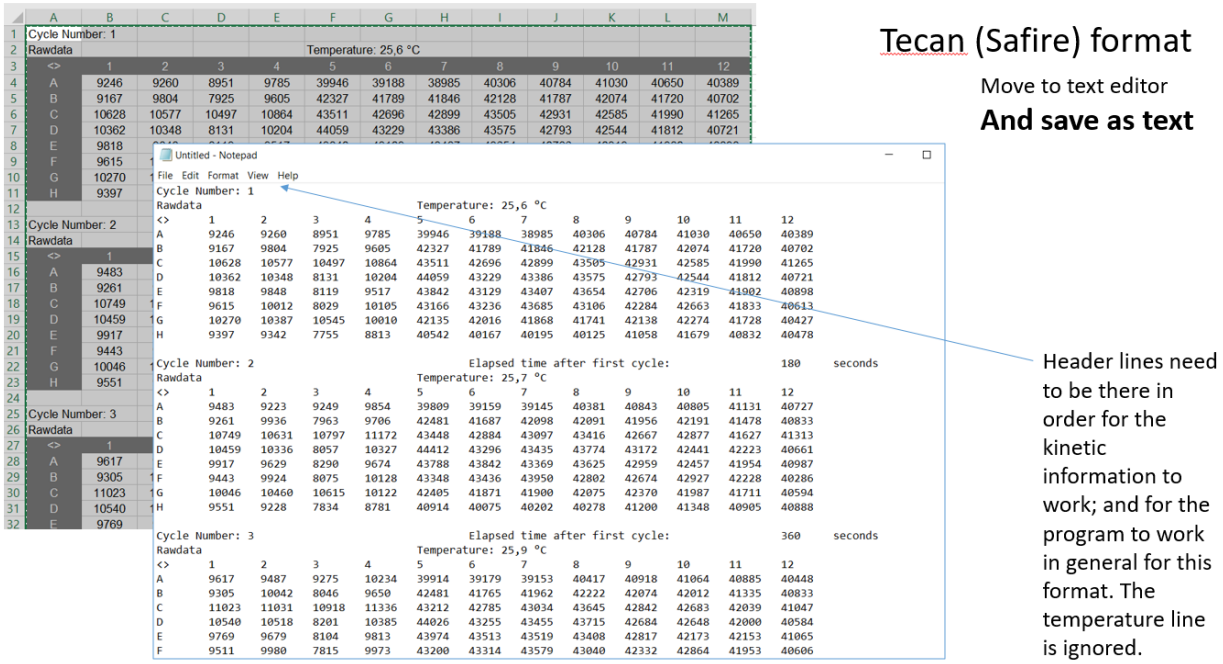



Figure 4. Preparing the TECAN Safire output.

The raw TECAN output file part containing the raw values is used as input for the program. A kinetic run will appear as above, whereas a single run will lack the line “Cycle Number: 1”. You need to add one to use the program for a single non-kinetic experiment. You can take from the template file. Observe that only the relevant data (highlighted) should be moved otherwise it will cause an error, do nonsense, or crash.

## WinSkan format

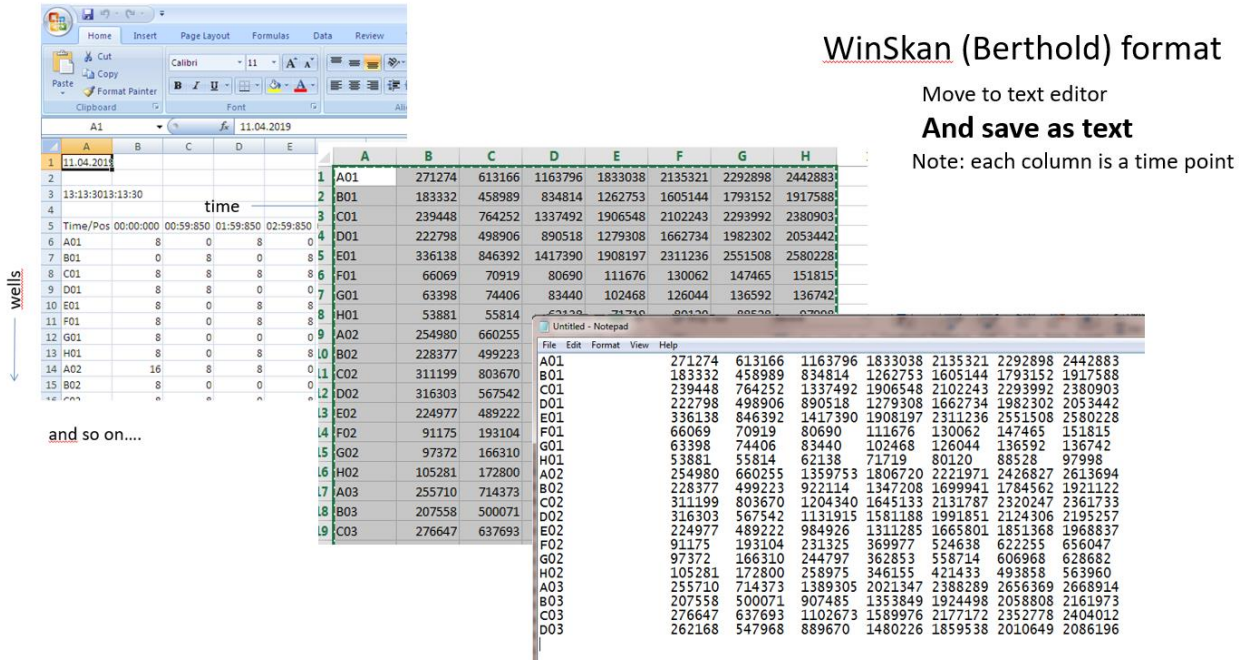



Figure 5. Preparing the WinSkan output.

The raw output file part of the WinSkan output containing the raw values is used as input for the program. Notice that you can copy just the data without the header information, which is in this case columns the different times. Therefore, you do not need to add anything for a single non-kinetic experiment. Observe that only the relevant data (highlighted) should be moved otherwise it will cause an error, do nonsense, or crash.

## Preparing the labels

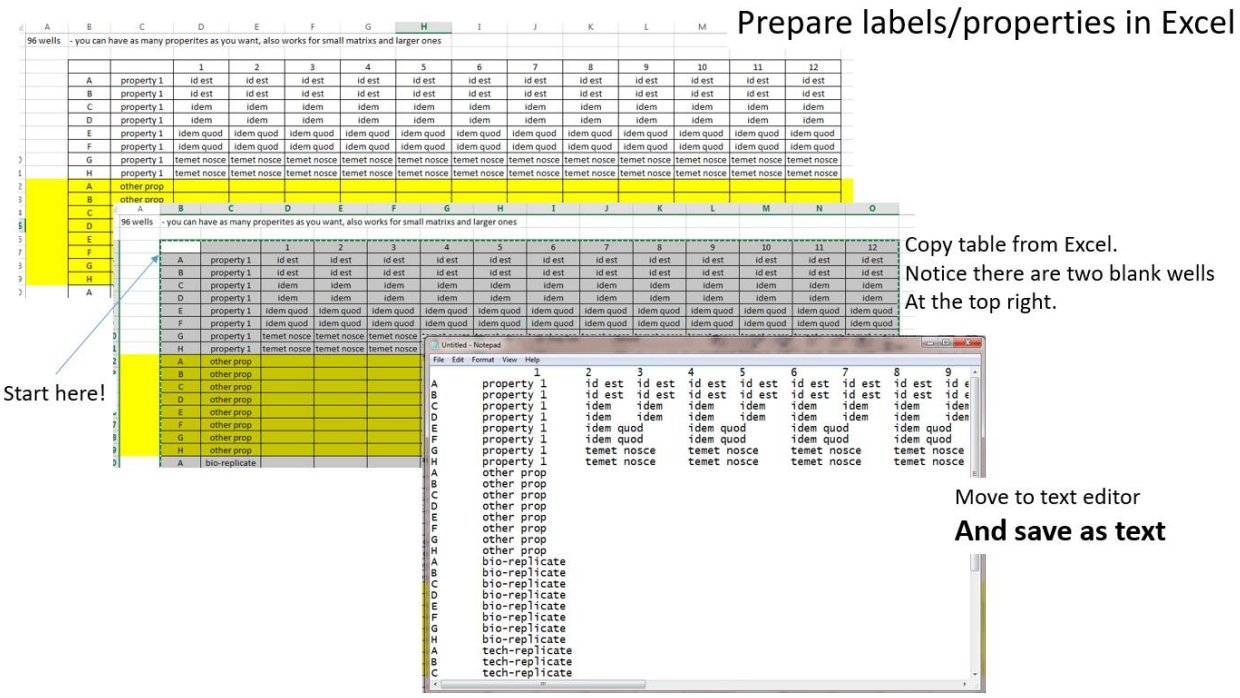



Figure 6. Preparing the labels input.

The format is above and in the template file. I guess the only special thing is that the top row has two tabs inside that have to be there before the column names appear. Make sure that you copy the entire label block, left justified and place it in a text file. If you have an alphabet other than English, you may need to save it in UTF-8 if you have characters that are not in 7-bit ASCII. I think it tolerates some gaps, but I haven't really tried to break it all the way. Each property type does not have to be right after each other; you'll see that in the TEMPLATE file. I think if you accidently made one twice with the same type(property), the program will only take the last one and discard the first.

## Putting it all together

After you have moved your data to your text files and have the labels as a text file as well then you are ready to drag and drop them into the program. You can also click the white text box space to being up a file browser if you prefer that. I like the drag and drop – which, on the PCs I have tested it on, needs to be activated by first gently moving the mouse over the text fields, without clicking, activates them. Now you drag and drop into the text boxes.

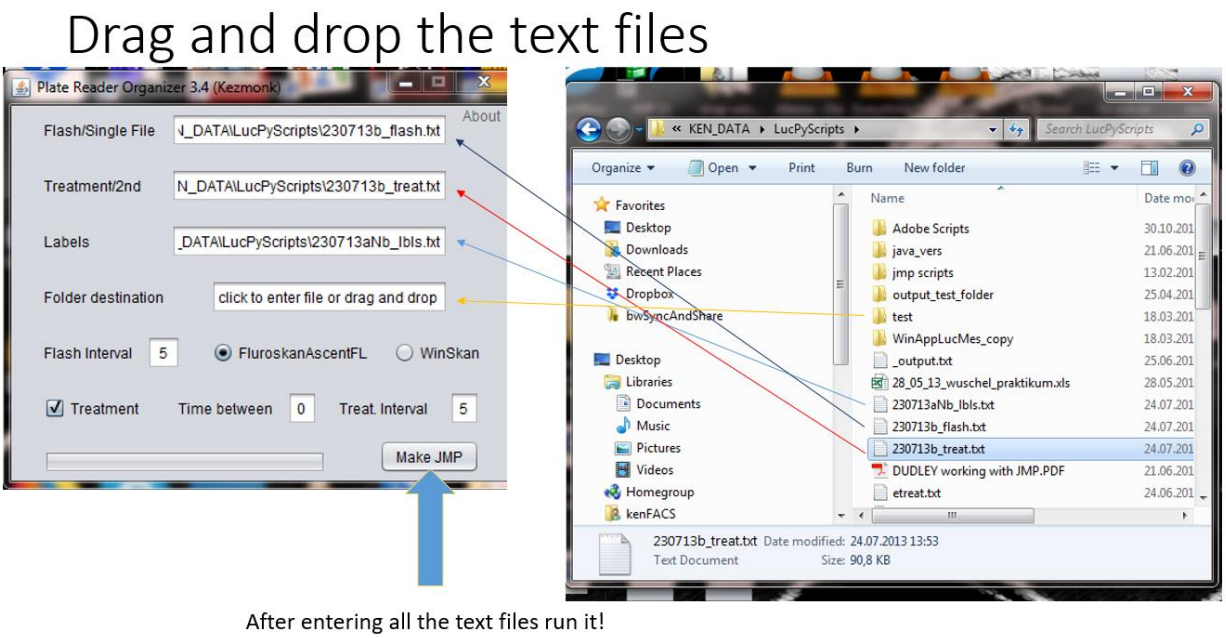
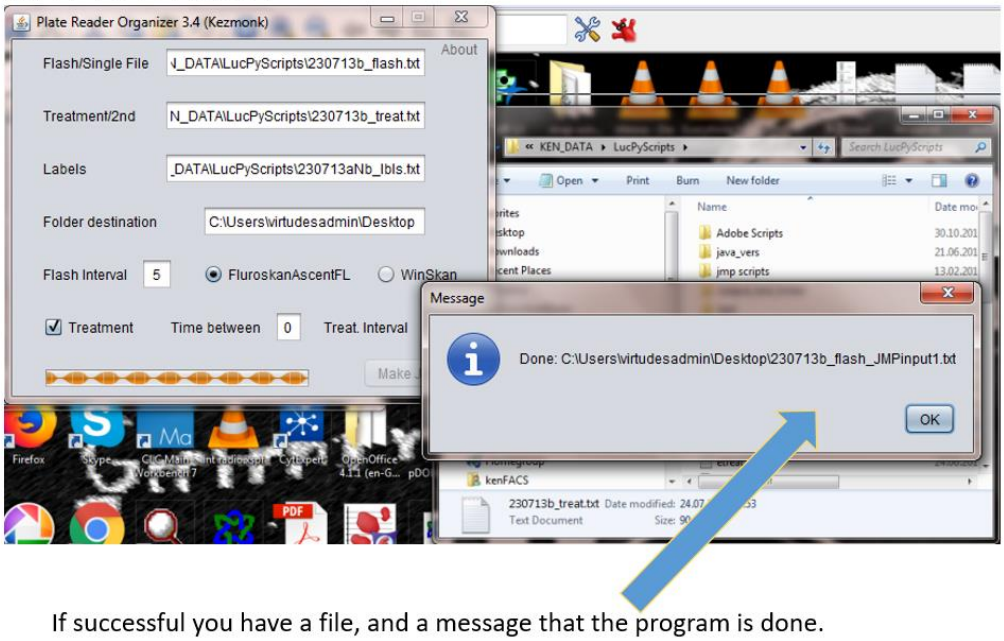



Figure 7. Drag and dropping text files. Remember to mouse over the text boxes first and then you can drag and drop into them, including the output folder you want to use.

Click “Make JMP”: if you did it all correctly, you’re done. Open in JMP by drag-drop or import, double check the column formats.




If successful you have a file, and a message that the program is done.

Figure 8. Executing the program.

Once you have all the text file paths entered into the program, run it. The text files are dynamically read so if you are running it over a network server it can sometimes be slow if you have a slow connection; it is not the program but saving it over the network server. Better to run it locally and then transfer the data when it is done. You get a message when the script is done.