

Getting messy-looking csv files in order

Data distribution platforms like the OECD or the Swiss National Bank put their files in a particular “format”. If the regional setting of your device corresponds to a different format, then the downloaded csv files may look pretty ugly, like so (or even uglier):

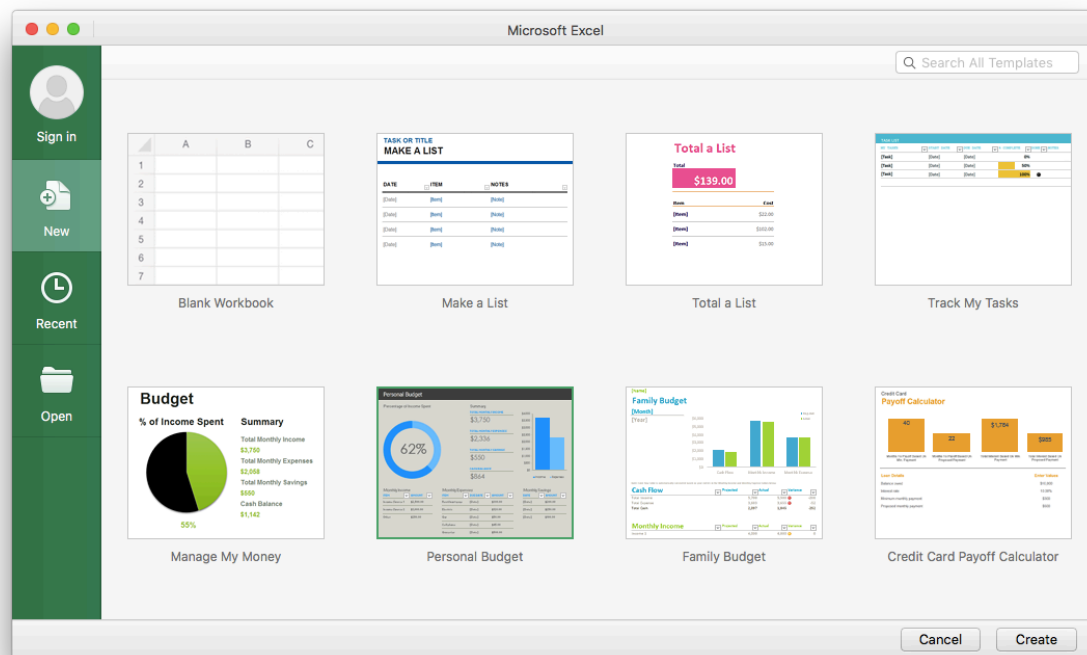
And when you try to read them into R, it looks like this:

```
> View(snbAussen)
> snbAussen = read.table("checkItOut.csv")
> View(snbAussen)
> snbAussen = read.table("snbAussenhandel.csv", sep = ";")
> View(snbAussen)
> View(snbAussen)
```

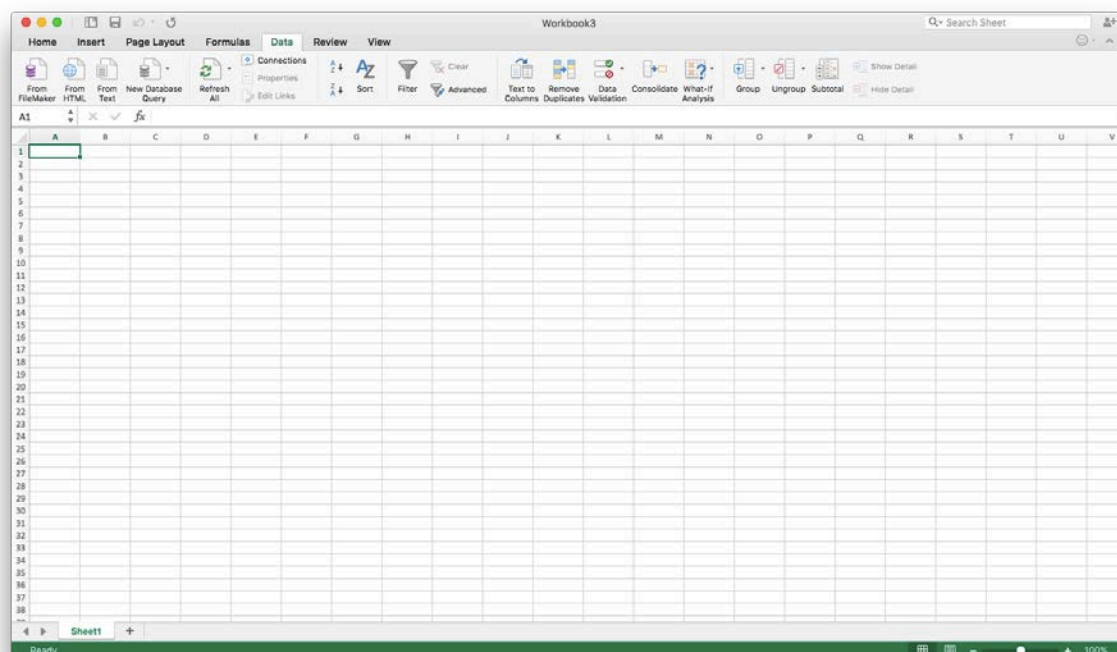
Sometimes, it helps to adjust the regional settings of your device (as in the case of my Windows 7 machine), but sometimes not (as in the case of my Mac). So what can you do? Below you find some screenshots that give you instructions how to get a clean-looking and workable csv file that you can read into R.

I first show this on a Mac (it works the same on Windows), and then I show you a different trick that sometimes works on a Windows machine

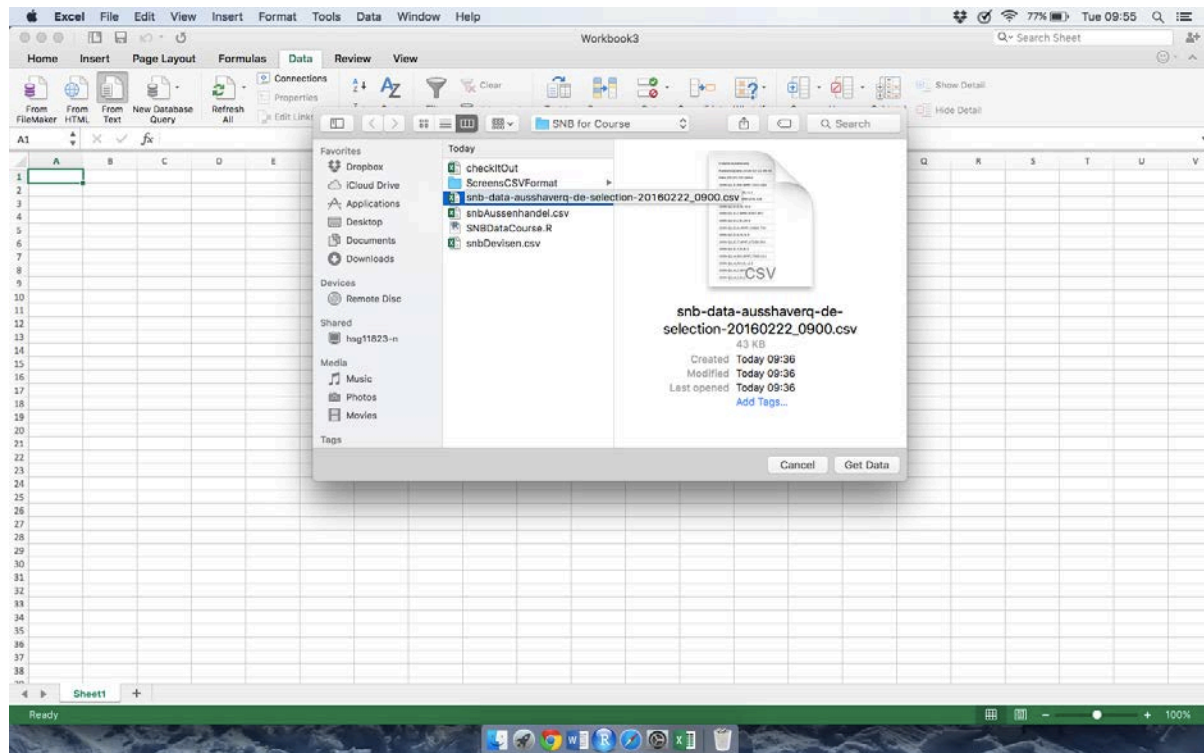
Open Excel



Open a new empty Workbook and go to the Data tab.



Click “From Text”



This opens the important wizard... Then just follow the steps you see on the below screens.

Text Import Wizard - Step 1 of 3

The Text Wizard has determined that your data is Delimited.

If this is correct, choose Next, or choose the Data Type that best describes your data.

- ☒ Delimited - Characters such as commas or tabs separate each field.
☐ Fixed width - Fields are aligned in columns with spaces between each field.

Start import at row:

File origin:

Preview of selected data:

Preview of file /Users/J.../snb-data-ausshaverq-de-selection-20160222_0900.csv.

```
1 "CubeId";"ausshaverq"
2 "PublishingDate";"2016-02-22 09:00"
3
4 "Date";"D0";"D1";"D2";"Value"
5 "1999-Q1";"E";"RH";"WMF";"7503.585"
6 "1999-Q1";"E";"RH";"R";"-1.1"
7 "1999-Q1";"E";"E";"WMF";"676.358"
8 "1999-Q1";"E";"E";"R";"-6.4"
9 "1999-Q1";"E";"I";"WMF";"8397.907"
```

Cancel

< Back

Next >

Finish

In the case of the Swiss National Bank data, the solution is most likely “semicolon”. I sometimes encounter the situation that I additionally need to check “Other” and insert double quotes in the field next to other (this leads to a lot of empty columns, but it is easy to delete them in R). When downloading data from another source (say from the OECD), you may need a different trick. Some willingness to play around is required here... Just check it out what works.

Text Import Wizard - Step 2 of 3

This screen lets you set the delimiters your data contains.

Delimiters

☐ Tab
 ☐ Treat consecutive delimiters as one

☒ Semicolon
 Text qualifier: "

☐ Comma

☐ Space

☐ Other:

Preview of selected data:

CubeId	ausshaverq			
PublishingDate	2016-02-22 09:00			
Date	D0	D1	D2	Value
1999-Q1	E	RH	WMF	7503.585
1999-Q1	E	RH	R	-1.1
1999-Q1	E	E	WMF	676.358
1999-Q1	E	E	R	-6.4
1999-Q1	E	I	WMF	8397.907

Text Import Wizard - Step 3 of 3

This screen lets you select each column and set the Data Format.

Column data format

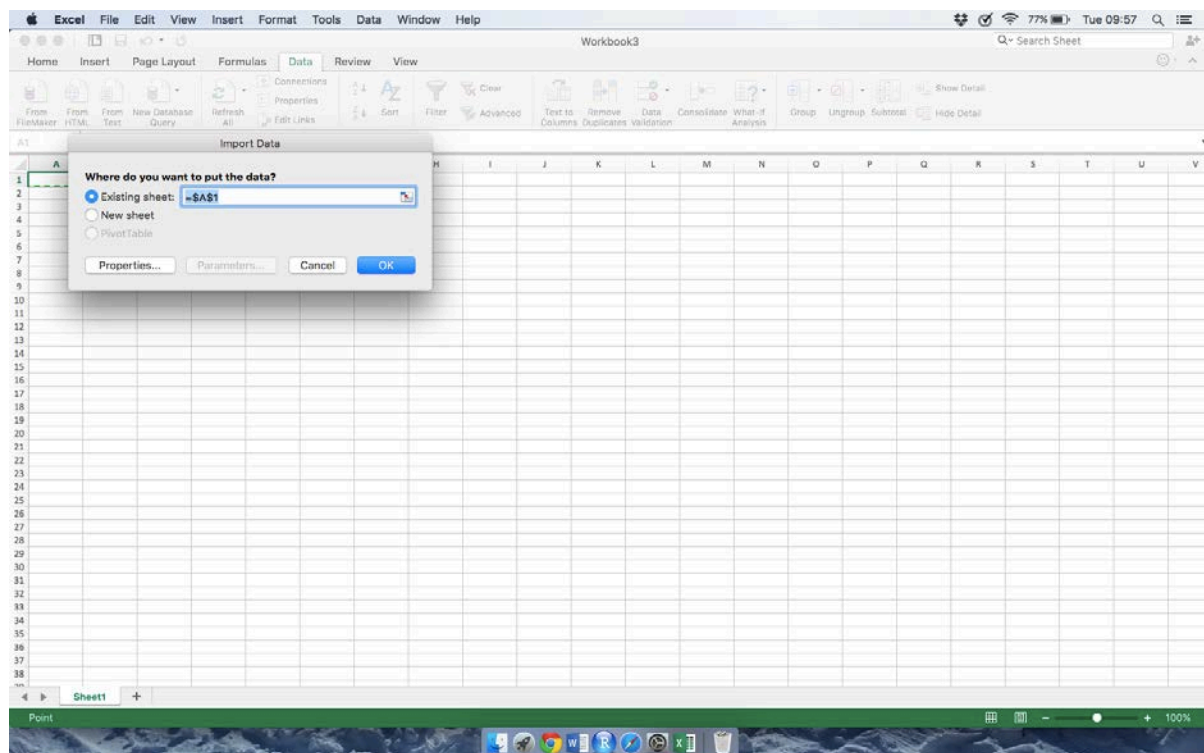
☒ General
☐ Text
☐ Date: DMY v
☐ Do not import column (Skip)

Advanced...

Preview of selected data:

General	General	General	General	General
CubeId	ausshaverg			
PublishingDate	2016-02-22 09:00			
Date	D0	D1	D2	Value
1999-01	E	RH	WMF	7503.585
1999-01	E	RH	R	-1.1
1999-01	E	E	WMF	676.358
1999-01	E	E	R	-6.4
1999-01	E	I	WMF	8397.907

Cancel
< Back
Next >
Finish



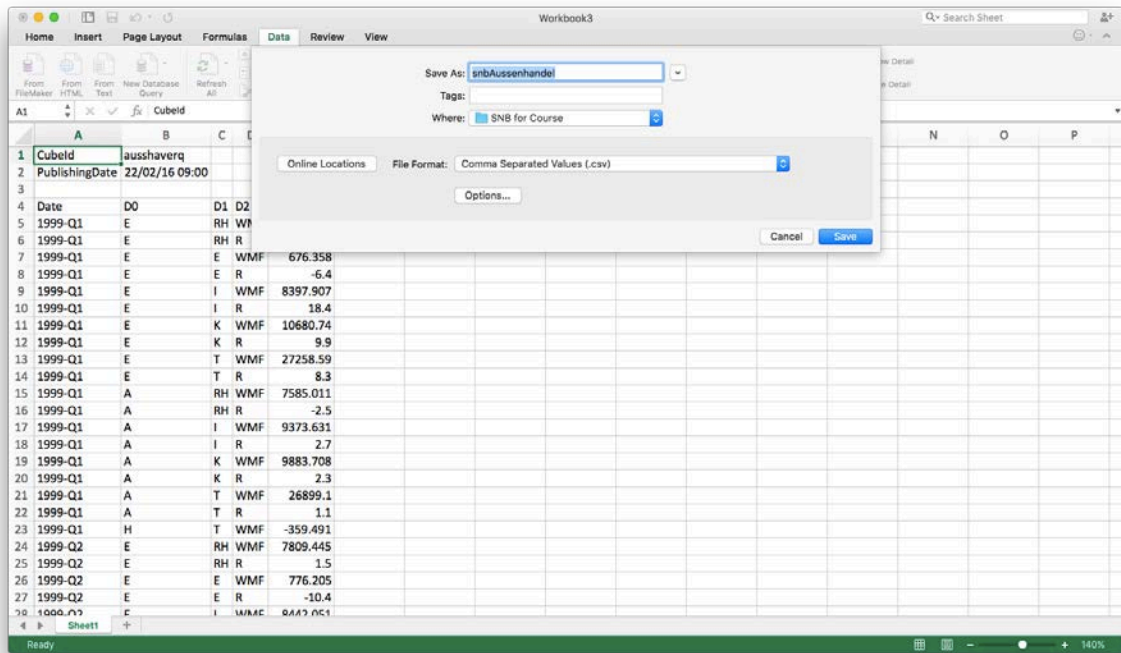
Now it looks neat...

The screenshot shows a Microsoft Excel spreadsheet titled 'Workbook3'. The 'Data' tab is selected. The spreadsheet contains the following data:

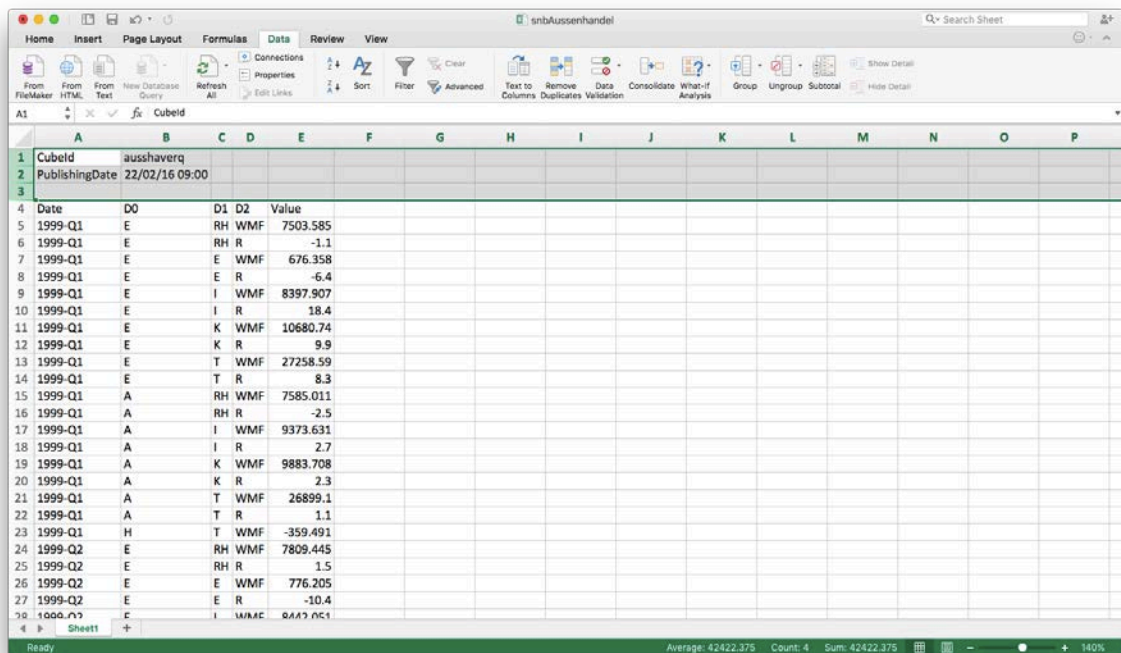
Date	D0	D1	D2	Value
1999-Q1	E	RH	WMF	7503.585
1999-Q1	E	RH	R	-1.1
1999-Q1	E	E	WMF	676.358
1999-Q1	E	E	R	-6.4
1999-Q1	E	I	WMF	8397.907
1999-Q1	E	I	R	18.4
1999-Q1	E	K	WMF	10680.74
1999-Q1	E	K	R	9.9
1999-Q1	E	T	WMF	27258.59
1999-Q1	E	T	R	8.3
1999-Q1	A	RH	WMF	7585.011
1999-Q1	A	RH	R	-2.5
1999-Q1	A	I	WMF	9373.631
1999-Q1	A	I	R	2.7
1999-Q1	A	K	WMF	9883.708
1999-Q1	A	K	R	2.3
1999-Q1	A	T	WMF	26899.1
1999-Q1	A	T	R	1.1
1999-Q1	H	T	WMF	-359.491
1999-Q2	E	RH	WMF	7809.445
1999-Q2	E	RH	R	1.5
1999-Q2	E	E	WMF	776.205
1999-Q2	E	E	R	-10.4

Save in csv format (we need this format for getting the data into R)

The screenshot shows the same Microsoft Excel spreadsheet as above, but with the 'File' menu open. The 'Save As...' option is selected, and a submenu is visible showing options like 'Restore', 'Import...', 'Reduce File Size...', 'Share', 'Restrict Permissions', 'Passwords...', 'Page Setup...', 'Print Area', 'Print...', and 'Properties...'. The spreadsheet data remains the same.



For reading the data into R, we need to delete the first three rows, as they would not make sense for R.



The screenshot shows an Excel spreadsheet with the following data (rows 4-27):

Date	D0	D1	D2	Value
1999-Q1	E	RH	WMF	7503.585
1999-Q1	E	RH	R	-1.1
1999-Q1	E	E	WMF	676.358
1999-Q1	E	E	R	-6.4
1999-Q1	E	I	WMF	8397.907
1999-Q1	E	I	R	18.4
1999-Q1	E	K	WMF	10680.74
1999-Q1	E	K	R	9.9
1999-Q1	E	T	WMF	27258.59
1999-Q1	E	T	R	8.3
1999-Q1	A	RH	WMF	7585.011
1999-Q1	A	RH	R	-2.5
1999-Q1	A	I	WMF	9373.631
1999-Q1	A	I	R	2.7
1999-Q1	A	K	WMF	9883.708
1999-Q1	A	K	R	2.3
1999-Q1	A	T	WMF	26899.1
1999-Q1	A	T	R	1.1
1999-Q1	H	T	WMF	-359.491
1999-Q2	E	RH	WMF	7809.445
1999-Q2	E	RH	R	1.5
1999-Q2	E	E	WMF	776.205
1999-Q2	E	E	R	-10.4
1999-Q2	E	I	WMF	8447.051

We'll read the data into R in class, I'll show you the details. But this is how it looks like. Finally, it's neat!

The RStudio screenshot shows the following data in the Environment pane:

Date	D0	D1	D2	Value
1999-Q1	E	RH	WMF	7503.585
1999-Q1	E	RH	R	-1.100
1999-Q1	E	E	WMF	676.358
1999-Q1	E	E	R	-6.400
1999-Q1	E	I	WMF	8397.907
1999-Q1	E	I	R	18.400
1999-Q1	E	K	WMF	10680.741
1999-Q1	E	K	R	9.900
1999-Q1	E	T	WMF	27258.591
1999-Q1	E	T	R	8.300
1999-Q1	A	RH	WMF	7585.011
1999-Q1	A	RH	R	-2.500
1999-Q1	A	I	WMF	9373.631
1999-Q1	A	I	R	2.700
1999-Q1	A	K	WMF	9883.708
1999-Q1	A	K	R	2.300
1999-Q1	A	T	WMF	26899.100
1999-Q1	A	T	R	1.100
1999-Q1	H	T	WMF	-359.491

The console shows the following R commands and output:

```
> View(snbAussen)
> snbAussen = read.table("checkItOut.csv", sep = ",")
> snbAussen = read.table("snbAussenhandel.csv", sep = ",")
> View(snbAussen)
> View(snbAussen)
> snbAussen = read.table("snbAussenhandel.csv", sep = ",")
> snbAussen = read.table("snbAussenhandel.csv", header = TRUE, sep = ",")
>
```

A different trick that works on my windows machine

Go to the start button and click on Control Panel (in Windows 10, just type “control panel” in the search field)

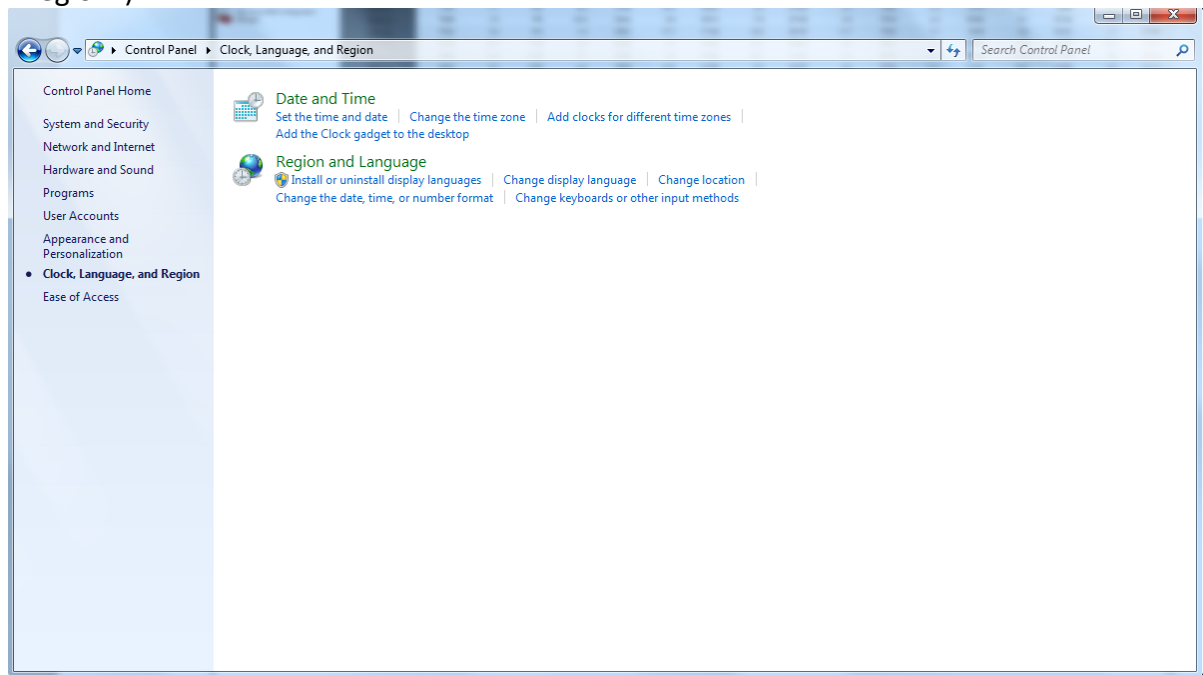
The screenshot shows the SNB BNS website interface. The main content area displays 'Aussenhandel nach Verwendungszweck' with a table of trade data. The table is organized into columns for different categories of goods and services, including 'Einfuhr', 'Ausfuhr', and 'Total'. The data is presented in a grid format with multiple rows and columns. A Windows Start menu is overlaid on the left side of the screen, showing various applications and the 'Control Panel' option highlighted. The Start menu also displays a search bar and a list of installed programs.

Einfuhr		Ausfuhr		Total	
Wert in Millionen Franken	Veränderung gegenüber dem Vorjahr in %	Wert in Millionen Franken	Veränderung gegenüber dem Vorjahr in %	Wert in Millionen Franken	Veränderung gegenüber dem Vorjahr in %
7504	-1.1	676	-6.4	8368	18.4
7809	1.5	776	-10.4	8442	9.8
7622	3.4	913	-1.4	8501	17.7
8340	12.5	1148	-2.8	9252	2.3
9146	11.8	1140	-7.1	9210	1.6
9125	9.7	1378	-4.4	9838	13.0
8803	9.0	1886	-5.2	9297	5.2
9017	3.4	1877	3.4	10636	9.0

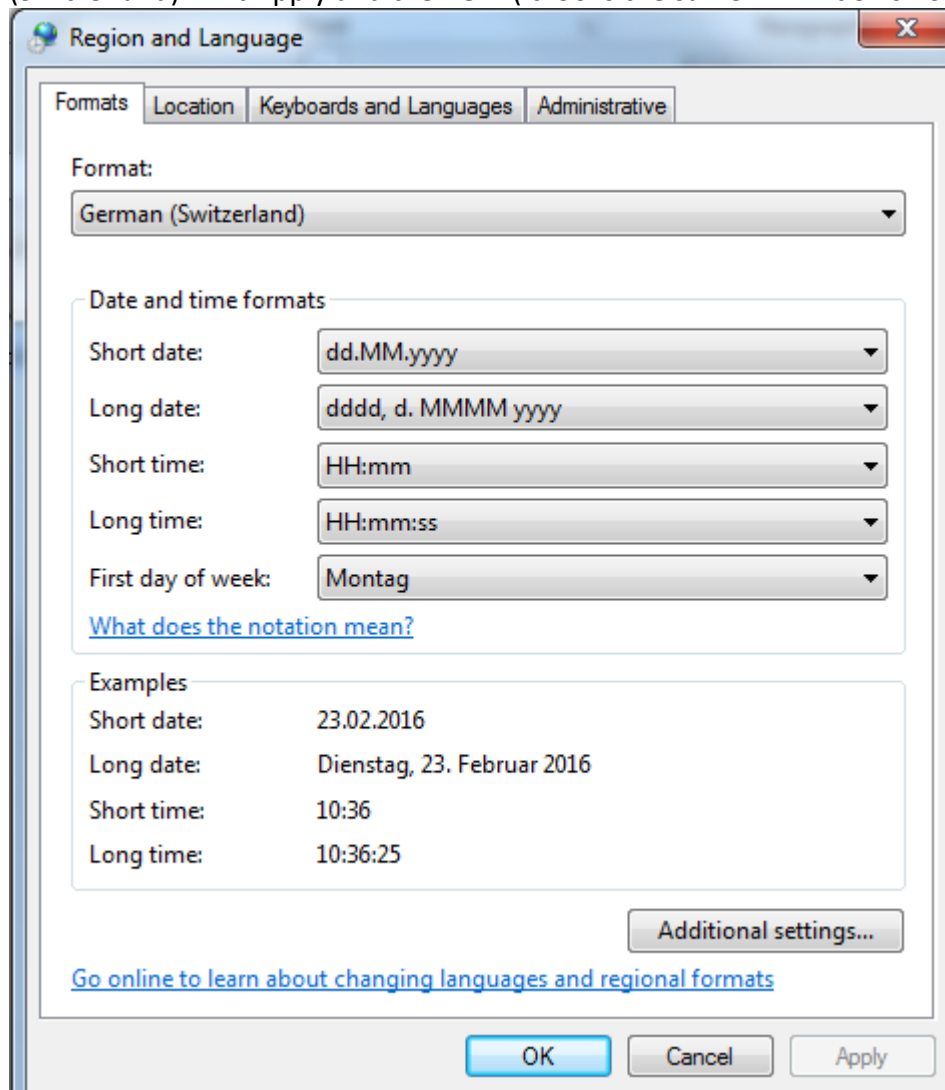
You get the next screen (probably); otherwise, navigate there using the navigation bar. Click on “Clock, Language, and Region” (it looks very similar in Windows 10).

The screenshot shows the Windows Control Panel interface. The 'Clock, Language, and Region' settings are highlighted. The settings include options for 'Change keyboards or other input methods', 'Change display language', and 'Optimize visual display'. The interface is clean and modern, with a search bar at the top and a list of settings on the left.

You'll get the following screen. Click on "Region and Language" (in Windows 10, click on "Region")



For a successful download of data from the Swiss National Bank, choose “German (Switzerland)”. Hit Apply and then OK. (It looks the same in Windows 10.)



On my Windows 7 or 10 machines, that’s all I have to do get neat csv files from the SNB.

If you want to download data from the OECD and you do so with Swiss regional settings, it will again look ugly. Then you will have to set your regional settings to, for instance, “English (United States)”.