The "Live" Code of Lecture 7

```
########################################################################
# LECTURE 7: Giving better names to economic variables in SNB data  #
########################################################################

# This is the first of a series of scripts for Lecture 7.
# In this script, we "dress up" our SNB data and then save them
# To the hard disk directly in RData and csv format. So we can use them
# next time directly, without any dressing up.

rm(list = ls())

# Read the data as they come from the SNB website
#####################################################

# REPLACE THE WORKING DIRECTORY BELOW WITH THE ONE FOR YOUR DEVICE
setwd("D:/Dropbox/Mac&Surf/Programmierkurs Dropb/Data")



# Read files, adjust file names to your situation!
rawXrates = read.csv(file =
   "SNB Xrates downloaded clean noEmptyCol.csv")

rawAussen = read.csv(file =
    "SNB Aussenhandel downloaded clean noEmptyCol.csv")

# If you are having troubles with the data, use the following two lines

# load("rawXrates.RData")
# load("rawExpImp.RData"); rawAussen = rawExpImp; rm(rawExpImp)

# You get these data in RData format from the online script on
# https://binswanger.github.io/practicaldata_hs16/#Problems--What-if-some-numbers-are-automatically-con
# Put these files in your working directory

# Give better name to the "economic variables"
###############################################

# First for rawAussen

# Make a copy of rawAussen
dataAussen = rawAussen

#rename Do
oldNamesA_D0 = unique(as.character(dataAussen$D0))
  # use "as.character", so we really know what we are doing, factors are tricky
newNamesA_D0 = c("Einfuhr", "Ausfuhr", "Handelsbilanzsaldo")
#check
cbind(oldNamesA_D0, newNamesA_D0)
```

```
##      oldNamesA_D0 newNamesA_D0
```

```
## [1,] "E"          "Einfuhr"
## [2,] "A"          "Ausfuhr"
## [3,] "H"          "Handelsbilanzsaldo"
```

```
# # For developing the function
# df = dataAussen
# colInDf = "D0"
# oldNames = oldNamesA_D0
# newNames = newNamesA_D0

# A function for renaming, so we need to type this 3 times...
rename = function(df, colInDf, oldNames, newNames){
  # In case that the column where we want to change names is a factor
  # first convert the factor to character. Otherwise, it becomes a bit more tricky
  if (is.factor(df[[colInDf]])) {
    df[[colInDf]] = as.character(df[[colInDf]])
  }

  # Now rename
  for (i in 1:length(oldNames)){
    df[[colInDf]]   [df[[colInDf]] == oldNames[i]] = newNames[i]
    #dataAussen[["D0"]]  [dataAussen[["D0"]] == "E"]
    #dataAussen$D0   [dataAussen$D0 == "E"]
  }
  return(df)
}

dataAussen = rename(dataAussen, "D0", oldNamesA_D0, newNamesA_D0)

# rename D1

oldNamesA_D1 = unique(as.character(dataAussen$D1))
newNamesA_D1= c("Total", "Maschinen, Apparate und Elektronik",
              "Praezisionsinstrumente, Uhren, Bijouterie",
              "Chemikalien", "Textilien, Bekleidung, Schuhe",
              "Fahrzeuge", "Metalle", "Uhren", "Praezisionsinstrumente",
              "Praezisionsinstrumente, Uhren, Bijouterie")

# Check
cbind(oldNamesA_D1, newNamesA_D1)
```

```
##       oldNamesA_D1 newNamesA_D1
##  [1,] "T"          "Total"
##  [2,] "MAE"        "Maschinen, Apparate und Elektronik"
##  [3,] "PUB"        "Praezisionsinstrumente, Uhren, Bijouterie"
##  [4,] "C"          "Chemikalien"
##  [5,] "TBS0"       "Textilien, Bekleidung, Schuhe"
##  [6,] "F"          "Fahrzeuge"
##  [7,] "M"          "Metalle"
##  [8,] "U"          "Uhren"
##  [9,] "P"          "Praezisionsinstrumente"
## [10,] "TBS1"       "Praezisionsinstrumente, Uhren, Bijouterie"
```

```
dataAussen = rename(dataAussen, "D1", oldNamesA_D1, newNamesA_D1)

oldNamesA_D2 = unique(as.character(dataAussen$D2))
newNamesA_D2= c("Wert in Millionen Franken",
              "Veraenderung gegenueber dem Vorjahr in \u0025 - nominal",
              "Veraenderung gegenueber dem Vorjahr in \u0025 - real")

# "u0025" is UTF-8 coding

# Check
cbind(oldNamesA_D2, newNamesA_D2)
```

```
##      oldNamesA_D2 newNamesA_D2
## [1,] "WMF"        "Wert in Millionen Franken"
## [2,] "N"          "Veraenderung gegenueber dem Vorjahr in % - nominal"
## [3,] "R"          "Veraenderung gegenueber dem Vorjahr in % - real"
```

```
dataAussen = rename(dataAussen, "D2", oldNamesA_D2, newNamesA_D2)

# Now the exchange rates

# Copy the data to new data frame
# get rid of end-of-month values of exchange rates
dataXrates = rawXrates[rawXrates$D0 == "M0", ]

oldNamesX_D1 = unique(as.character(dataXrates$D1))

newNamesX_D1 = c(
    "CHF/Euro",
    "CHF/Britisches Pfund",
    "CHF/100 Daenische Kronen",
    "CHF/100 Norwegische Kronen",
    "CHF/100 Tschechische Kronen",
    "CHF/100 Ungarische Forint",
    "CHF/100 Polnische Zloty",
    "CHF/100 Rubel",
    "CHF/100 Schwedische Kronen",
    "CHF/100 Tuerkische Lira",
    "CHF/US Dollar",
    "CHF/Kanadische Dollar",
    "CHF/100 Argentinische Pesos",
    "CHF/100 Brasilianische Real",
    "CHF/100 Mexikanische Pesos",
    "CHF/Suedafrikanischer Rand",
    "CHF/100 Japanische Yen",
    "CHF/Australischer Dollar",
    "CHF/100 Chinesische Yuan",
    "CHF/100 Hongkong Dollars",
    "CHF/100 Koreanische Won",
    "CHF/100 Malaysische Ringgit",
    "CHF/Neuseeländischer Dollar",
    "CHF/100 Singapur-Dollars",
    "CHF/100 Thailaendische Baht",
```

```
    "CHF/Sonderziehungsrecht")


# Check
cbind(oldNamesX_D1, newNamesX_D1)
```

```
##       oldNamesX_D1 newNamesX_D1
##  [1,] "EUR1"       "CHF/Euro"
##  [2,] "GBP1"       "CHF/Britisches Pfund"
##  [3,] "DKK100"     "CHF/100 Daenische Kronen"
##  [4,] "NOK100"     "CHF/100 Norwegische Kronen"
##  [5,] "CZK100"     "CHF/100 Tschechische Kronen"
##  [6,] "HUF100"     "CHF/100 Ungarische Forint"
##  [7,] "PLN100"     "CHF/100 Polnische Zloty"
##  [8,] "RUB100"     "CHF/100 Rubel"
##  [9,] "SEK100"     "CHF/100 Schwedische Kronen"
## [10,] "TRY100"     "CHF/100 Tuerkische Lira"
## [11,] "USD1"       "CHF/US Dollar"
## [12,] "CAD1"       "CHF/Kanadische Dollar"
## [13,] "ARS100"     "CHF/100 Argentinische Pesos"
## [14,] "BRL100"     "CHF/100 Brasilianische Real"
## [15,] "MXN100"     "CHF/100 Mexikanische Pesos"
## [16,] "ZAR1"       "CHF/Suedafrikanischer Rand"
## [17,] "JPY100"     "CHF/100 Japanische Yen"
## [18,] "AUD1"       "CHF/Australischer Dollar"
## [19,] "CNY100"     "CHF/100 Chinesische Yuan"
## [20,] "HKD100"     "CHF/100 Hongkong Dollars"
## [21,] "KRW100"     "CHF/100 Koreanische Won"
## [22,] "MYR100"     "CHF/100 Malaysische Ringgit"
## [23,] "NZD1"       "CHF/Neuseeländischer Dollar"
## [24,] "SGD100"     "CHF/100 Singapur-Dollars"
## [25,] "THB100"     "CHF/100 Thailaendische Baht"
## [26,] "XDR1"       "CHF/Sonderziehungsrecht"
```

```
dataXrates = rename(dataXrates, "D1", oldNamesX_D1, newNamesX_D1)

# Create unique timeIDs
########################

library(stringr)

calcTimeID = function(df){

  df$year = as.numeric(
    substr(df$Date, start = 1, stop = 4)  )

  df$month = as.numeric(
    substr(df$Date, start = 6, stop = 7)  )

  df$timeID = df$year +
    (df$month-1)/12

  df$year = NULL
```

```
  df$month = NULL

  return(df)
}

dataAussen = calcTimeID(dataAussen)
dataXrates = calcTimeID(dataXrates)


# Save data in RData format
save(dataAussen, dataXrates,
     file = "dataforAnalysis.RData")


# Write data to csv (just for fun)

write.table(dataAussen, file = "dataAussen.csv",
            sep = ",", qmethod = "escape", na = "",
            row.names = FALSE)

write.table(dataXrates, file = "dataXrates.csv",
            sep = ",", qmethod = "escape", na = "",
            row.names = FALSE)
```