Data used in the App About our World

# Introduction

The raw data used in this app are listed below and are just a few of many more that can be found on the [World Banks](http://data.worldbank.org/) webpage.

The data was downloaded from World Banks webpage during the period 2014-12-17 to 2015-01-15 and placed in a folder called "rawData" .

The name of the data files were kept the same as they were found at World Banks webpage. See the section for "Loading data into R" below.

Two data files were created as the result of the code described below and used running the app:  
dataMelt.csv  
countries.csv

Two additional data files used in the app were created by hand:  
indicators.csv  
years.csv

For the app these four data files were placed in a folder called "data"

# Raw Data

The data downloaded from the World Bank was found via the following links.

[Population, total](http://data.worldbank.org/indicator/SP.POP.TOTL)  
[Arable land (hectares per person)](http://data.worldbank.org/indicator/AG.LND.ARBL.HA.PC)  
[GNI per capita based on purchasing power parity (PPP)](http://data.worldbank.org/indicator/NY.GNP.PCAP.PP.CD)  
[Improved water source, rural (% of rural population with access)](http://data.worldbank.org/indicator/SH.H2O.SAFE.RU.ZS)  
[Improved water source, urban (% of urban population with access)](http://data.worldbank.org/indicator/SH.H2O.SAFE.UR.ZS)  
[Access to electricity (% of population)](http://data.worldbank.org/indicator/EG.ELC.ACCS.ZS)  
[Internet users (per 100 people)](http://data.worldbank.org/indicator/IT.NET.USER.P2)  
[Motor vehicles (per 1,000 people)](http://data.worldbank.org/indicator/IS.VEH.NVEH.P3)  
[Fertility rate, total (births per woman)](http://data.worldbank.org/indicator/SP.DYN.TFRT.IN)  
[Mortality rate, under-5 (per 1,000 live births)](http://data.worldbank.org/indicator/SH.DYN.MORT)

# Code

library(plyr)  
library(reshape2)

#### Loading data into R

totPop<-read.csv("rawData/sp.pop.totl\_Indicator\_en\_csv\_v2.csv",skip=2,header=TRUE, stringsAsFactors = FALSE)  
metadataTotPop<-read.csv("rawData/Metadata\_Country\_sp.pop.totl\_Indicator\_en\_csv\_v2.csv", header=TRUE,stringsAsFactors = FALSE)

agrRurDev<-read.csv("rawData/ag.lnd.arbl.ha.pc\_Indicator\_en\_csv\_v2.csv",skip=2,header=TRUE, stringsAsFactors = FALSE)  
metadataAgrRurDev<-read.csv("rawData/Metadata\_Country\_ag.lnd.arbl.ha.pc\_Indicator\_en\_csv\_v2.csv", header=TRUE,stringsAsFactors = FALSE)

accToEl<-read.csv("rawData/eg.elc.accs.zs\_Indicator\_en\_csv\_v2.csv",skip=2,header=TRUE, stringsAsFactors = FALSE)  
metadataAccToEl<-read.csv("rawData/Metadata\_Country\_eg.elc.accs.zs\_Indicator\_en\_csv\_v2.csv", header=TRUE,stringsAsFactors = FALSE)

intUse<-read.csv("rawData/it.net.user.p2\_Indicator\_en\_csv\_v2.csv",skip=2,header=TRUE, stringsAsFactors = FALSE)  
metadataIntUse<-read.csv("rawData/Metadata\_Country\_it.net.user.p2\_Indicator\_en\_csv\_v2.csv", header=TRUE,stringsAsFactors = FALSE)

h2oRU<-read.csv("rawData/sh.h2o.safe.ru.zs\_Indicator\_en\_csv\_v2.csv",skip=2,header=TRUE, stringsAsFactors = FALSE)  
metadataH2oRU<-read.csv("rawData/Metadata\_Country\_sh.h2o.safe.ru.zs\_Indicator\_en\_csv\_v2.csv", header=TRUE,stringsAsFactors = FALSE)

h2oUR<-read.csv("rawData/sh.h2o.safe.ur.zs\_Indicator\_en\_csv\_v2.csv",skip=2,header=TRUE, stringsAsFactors = FALSE)  
metadataH2oUR<-read.csv("rawData/Metadata\_Country\_sh.h2o.safe.ur.zs\_Indicator\_en\_csv\_v2.csv", header=TRUE,stringsAsFactors = FALSE)

gnpPC<-read.csv("rawData/ny.gnp.pcap.pp.cd\_Indicator\_en\_csv\_v2.csv",skip=2,header=TRUE, stringsAsFactors = FALSE)  
metadataGnpPC<-read.csv("rawData/Metadata\_Country\_ny.gnp.pcap.pp.cd\_Indicator\_en\_csv\_v2.csv", header=TRUE,stringsAsFactors = FALSE)

ferRate<-read.csv("rawData/sp.dyn.tfrt.in\_Indicator\_en\_csv\_v2.csv",skip=2,header=TRUE, stringsAsFactors = FALSE)  
metadataFerRate<-read.csv("rawData/Metadata\_Country\_sp.dyn.tfrt.in\_Indicator\_en\_csv\_v2.csv", header=TRUE,stringsAsFactors = FALSE)

motVeh<-read.csv("rawData/is.veh.nveh.p3\_Indicator\_en\_csv\_v2.csv",skip=2,header=TRUE, stringsAsFactors = FALSE)  
metadataMotVeh<-read.csv("rawData/Metadata\_Country\_is.veh.nveh.p3\_Indicator\_en\_csv\_v2.csv", header=TRUE,stringsAsFactors = FALSE)

mortBaby<-read.csv("rawData/sh.dyn.mort\_Indicator\_en\_csv\_v2.csv",skip=2,header=TRUE, stringsAsFactors = FALSE)  
metadataMortBaby<-read.csv("rawData/Metadata\_Country\_sh.dyn.mort\_Indicator\_en\_csv\_v2.csv", header=TRUE,stringsAsFactors = FALSE)

#### Merge data with metadata

totalPopulation<-merge(metadataTotPop[,c(2:3,5)],totPop,by.x="Country.Code",by.y="Country.Code",all=TRUE)  
arableLand<-merge(metadataAgrRurDev[,c(2:3,5)],agrRurDev[,-5],by.x="Country.Code",by.y="Country.Code",all=TRUE)  
accessToElectricity<-merge(metadataAccToEl[,c(2:3,5)],accToEl,by.x="Country.Code",by.y="Country.Code",all=TRUE)  
internetUse<-merge(metadataIntUse[,c(2:3,5)],intUse,by.x="Country.Code",by.y="Country.Code",all=TRUE) freshWaterRU<-merge(metadataH2oRU[,c(2:3,5)],h2oRU[,-5],by.x="Country.Code",by.y="Country.Code",all=TRUE)  
freshWaterUR<-merge(metadataH2oUR[,c(2:3,5)],h2oUR[,-5],by.x="Country.Code",by.y="Country.Code",all=TRUE)  
gnpPerCapita<-merge(metadataGnpPC[,c(2:3,5)],gnpPC[,-5],by.x="Country.Code",by.y="Country.Code",all=TRUE)  
fertilityRate<-merge(metadataFerRate[,c(2:3,5)],ferRate,by.x="Country.Code",by.y="Country.Code",all=TRUE)  
motorVehicle<-merge(metadataMotVeh[,c(2:3,5)],motVeh[,-5],by.x="Country.Code",by.y="Country.Code",all=TRUE)  
mortalityBaby<-merge(metadataMortBaby[,c(2:3,5)],mortBaby[,-5],by.x="Country.Code",by.y="Country.Code",all=TRUE)

#### Merge data

data0<-rbind(totalPopulation,arableLand,accessToElectricity,internetUse,freshWaterRU,freshWaterUR,gnpPerCapita,fertilityRate,motorVehicle,mortalityBaby)

#### CleanData

data1<-data0[,1:58]  
indWLD <- data1[,1] %in% "WLD"  
data1[indWLD,2]<-"World"  
indEUU <- data1[,1] %in% "EUU"  
data1[indEUU,2]<-"European Union"  
data2<-data1[which(data1$Region!=""),]  
data3<-data2[which(data2$Region!="NA"),]

#### Change variable names

colNames0<-colnames(data3)  
colNames1<-gsub(".","",colNames0,fixed=TRUE)  
colNames2<-gsub("X","",colNames1,fixed=TRUE)  
colnames(data3)<-colNames2  
countryNames<-data.frame(Country=data3[1:214,4],stringsAsFactors=FALSE)  
library(plyr)  
countries<-arrange(countryNames,Country) write.csv(countries,"countries.csv")

#### reshaping and save as csv-file

library(reshape2)  
dataMelt<-melt(data3,id=c(1:6),measure.vars=c(7:58))  
write.csv(dataMelt,"dataMelt.csv")