R Notebook

#QUESTION 3 The following code will download the data from the wiki page above and create a dataframe named top Cities  
  
#### Start Top Cities Code ####  
#library(reprex) -- I commented out the libraries I did not use  
library(tidyverse)

## ── Attaching packages ──────────────────────────────────────────────────────────────────────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.3.2 ✓ purrr 0.3.4  
## ✓ tibble 3.0.3 ✓ dplyr 1.0.2  
## ✓ tidyr 1.1.1 ✓ stringr 1.4.0  
## ✓ readr 1.3.1 ✓ forcats 0.5.0

## ── Conflicts ─────────────────────────────────────────────────────────────────────────────────────────────────────────── tidyverse\_conflicts() ──  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(rvest)

## Loading required package: xml2

##   
## Attaching package: 'rvest'

## The following object is masked from 'package:purrr':  
##   
## pluck

## The following object is masked from 'package:readr':  
##   
## guess\_encoding

#library(readr)  
#library(dplyr)  
  
topCities <- read\_html("https://en.wikipedia.org/wiki/List\_of\_cities\_proper\_by\_population")  
topCities <- html\_table(  
html\_node(  
topCities, "table.sortable"  
), header=TRUE, trim=F, fill = TRUE  
)[-1,]  
names(topCities) <- c(  
"City", "Country", "Image", "Population", "City.Def", "City.Population", "City.Area", "Metro.Pop",  
"Metro.Area", "Urban.Pop", "Urban.Area"  
)  
  
topCities$Image <- NULL  
  
## Old String Substitute (gsub)  
topCities$Population <- as.numeric(gsub(",", "", topCities$Population))  
  
## tidyverse String Substitute (str\_replace\_all)  
topCities$City<-str\_replace\_all(topCities$City, "\n", "")  
topCities$Country<-str\_replace\_all(topCities$Country, "\n", "")  
topCities$City.Def<-str\_replace\_all(topCities$City.Def, "\n", "")  
topCities$City.Area<-str\_replace\_all(topCities$City.Area, "\n", "")  
topCities$Metro.Pop<-str\_replace\_all(topCities$Metro.Pop, "\n", "")  
topCities$Metro.Area<-str\_replace\_all(topCities$Metro.Area, "\n", "")  
topCities$Urban.Pop<-str\_replace\_all(topCities$Urban.Pop, "\n", "")  
topCities$Urban.Area<-str\_replace\_all(topCities$Urban.Area, "\n", "")  
#### End Top Cities Code ####

# 3 a) Use the topCities dataframe and create the R code to find the 5 largest cities in India.   
  
indian\_cities <- filter(topCities, Country == "India")  
  
# small change for demo purposes  
head(indian\_cities, 5)

## [1] City Country Population City.Def   
## [5] City.Population City.Area Metro.Pop Metro.Area   
## [9] Urban.Pop Urban.Area   
## <0 rows> (or 0-length row.names)

My change

# After this both India and China will work  
topCities$Country<-str\_trim(topCities$Country)   
  
# now, look  
  
indian\_cities <- filter(topCities, Country == "India")  
  
# small change for demo purposes  
head(indian\_cities, 5)

## City Country Population City.Def City.Population  
## 1 Delhi India 28514000 National capital territory 16,753,235[16]  
## 2 Mumbai India 19980000 Municipality 12,478,447[24]  
## 3 Kolkata India 14681000 Municipality 4,496,694[38]  
## 4 Bangalore India 11440000 Municipality 8,443,675[49]  
## 5 Chennai India 10456000 Municipality 6,727,000[56]  
## City.Area Metro.Pop Metro.Area Urban.Pop Urban.Area  
## 1 1,484 29,000,000[17] 3,483[17] 29,617,000 2,232[f]  
## 2 603 24,400,000[25] 4,355[26] 23,355,000 944[i]  
## 3 205 14,035,959[39] 1,851[40] 17,560,000 1,351  
## 4 709 13,707,000 1,205  
## 5 426[57] 11,324,000 1,049

#3 b. Use the topCities dataframe and create the R code to find the 5 largest cities in China.   
china\_cities <- filter(topCities, Country == "China")  
  
head(china\_cities, 5)

## City Country Population City.Def City.Population City.Area  
## 1 Shanghai China 25582000 Municipality 24,183,000[18] 6,341  
## 2 Beijing China 19618000 Municipality 21,707,000[18] 16,411  
## 3 Chongqing China 14838000 Municipality 30,165,500[36] 82,403  
## 4 Tianjin China 13215000 Municipality 15,569,000[18] 11,920  
## 5 Guangzhou China 12638000 City (sub-provincial) 14,498,400[18] 7,434  
## Metro.Pop Metro.Area Urban.Pop Urban.Area  
## 1 N/A N/A 22,120,000 4,068[g]  
## 2 N/A N/A 19,433,000 4,172  
## 3 N/A N/A 7,739,000 1,537  
## 4 N/A N/A 10,800,000 2,813  
## 5 N/A N/A 20,902,000 4,342[n]

Please use the dplyr function arrange to put the data in order.