

# COVID-19\_Nigeria

Bbosa Robert

4/30/2020

## Contents

Installing important packages	2
Loading libraries	2
Pulling the coronavirus data from John Hopkins repo	3
<a href="https://github.com/CSSEGISandData/COVID-19">https://github.com/CSSEGISandData/COVID-19</a>	3
Transforming the data from wide to long	9
Creating new data frame	9
Parsing the date	11
Aggregate the data to daily	12
Pulling death cases	12
Transforming the data from wide to long	18
Creating new data frame	18
Parsing the date	20
Aggregate the data to daily	20
Pulling recovered cases	21
Transforming the data from wide to long	27
Creating new data frame	27
Parsing the date	29
Aggregate the data to daily	29
Remove variables such as province.state, Lat and Long	30
Introduce a variable for cumulative totals of the cases called total_cases	30
filter out Nigeria	30
Export it in .csv and .xlsx formarts	31

## Installing important packages

```
install.packages("ggpubr") install.packages("hrbrthemes")
```

## Loading libraries

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.6.3
## -- Attaching packages ----- tidyverse 1.3.0 --
## v ggplot2 3.3.0      v purrr   0.3.3
## v tibble  2.1.3      v dplyr  0.8.4
## v tidyr   1.0.2      v stringr 1.4.0
## v readr   1.3.1      v forcats 0.5.0
## Warning: package 'dplyr' was built under R version 3.6.3
## Warning: package 'forcats' was built under R version 3.6.3
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
library(plotly)
```

```
## Warning: package 'plotly' was built under R version 3.6.3
##
## Attaching package: 'plotly'
##
## The following object is masked from 'package:ggplot2':
##
##     last_plot
##
## The following object is masked from 'package:stats':
##
##     filter
##
## The following object is masked from 'package:graphics':
##
##     layout
```

```
library(ggplot2)
library(dplyr)
library(viridis)
```

```
## Warning: package 'viridis' was built under R version 3.6.3
## Loading required package: viridisLite
```

```
library(patchwork)
```

```
## Warning: package 'patchwork' was built under R version 3.6.3
```

```
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 3.6.3
## Loading required package: magrittr
```

```
## Warning: package 'magrittr' was built under R version 3.6.3
##
## Attaching package: 'magrittr'
## The following object is masked from 'package:purrr':
##
##     set_names
## The following object is masked from 'package:tidyr':
##
##     extract
library(hrbrthemes)

## Warning: package 'hrbrthemes' was built under R version 3.6.3
## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.
##       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and
##       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

## Pulling the coronavirus data from John Hopkins repo

<https://github.com/CSSEGISandData/COVID-19>

#———— Pulling confirmed cases —————

```
conf_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/
csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_global.csv"

raw_conf <- read.csv(file = conf_url, stringsAsFactors = FALSE)

lapply(1:ncol(raw_conf), function(i){
  if(all(is.na(raw_conf[, i]))){
    raw_conf <- raw_conf[, -i]
    return(print(paste("Column", names(raw_conf)[i], "is missing", sep = " ")))
  } else {
    return(NULL)
  }
})

## [[1]]
## NULL
##
## [[2]]
## NULL
##
## [[3]]
## NULL
##
## [[4]]
## NULL
##
## [[5]]
## NULL
##
```

```
## [[6]]
## NULL
##
## [[7]]
## NULL
##
## [[8]]
## NULL
##
## [[9]]
## NULL
##
## [[10]]
## NULL
##
## [[11]]
## NULL
##
## [[12]]
## NULL
##
## [[13]]
## NULL
##
## [[14]]
## NULL
##
## [[15]]
## NULL
##
## [[16]]
## NULL
##
## [[17]]
## NULL
##
## [[18]]
## NULL
##
## [[19]]
## NULL
##
## [[20]]
## NULL
##
## [[21]]
## NULL
##
## [[22]]
## NULL
##
## [[23]]
## NULL
##
```

```
## [[24]]
## NULL
##
## [[25]]
## NULL
##
## [[26]]
## NULL
##
## [[27]]
## NULL
##
## [[28]]
## NULL
##
## [[29]]
## NULL
##
## [[30]]
## NULL
##
## [[31]]
## NULL
##
## [[32]]
## NULL
##
## [[33]]
## NULL
##
## [[34]]
## NULL
##
## [[35]]
## NULL
##
## [[36]]
## NULL
##
## [[37]]
## NULL
##
## [[38]]
## NULL
##
## [[39]]
## NULL
##
## [[40]]
## NULL
##
## [[41]]
## NULL
##
```

```
## [[42]]
## NULL
##
## [[43]]
## NULL
##
## [[44]]
## NULL
##
## [[45]]
## NULL
##
## [[46]]
## NULL
##
## [[47]]
## NULL
##
## [[48]]
## NULL
##
## [[49]]
## NULL
##
## [[50]]
## NULL
##
## [[51]]
## NULL
##
## [[52]]
## NULL
##
## [[53]]
## NULL
##
## [[54]]
## NULL
##
## [[55]]
## NULL
##
## [[56]]
## NULL
##
## [[57]]
## NULL
##
## [[58]]
## NULL
##
## [[59]]
## NULL
##
```

```
## [[60]]
## NULL
##
## [[61]]
## NULL
##
## [[62]]
## NULL
##
## [[63]]
## NULL
##
## [[64]]
## NULL
##
## [[65]]
## NULL
##
## [[66]]
## NULL
##
## [[67]]
## NULL
##
## [[68]]
## NULL
##
## [[69]]
## NULL
##
## [[70]]
## NULL
##
## [[71]]
## NULL
##
## [[72]]
## NULL
##
## [[73]]
## NULL
##
## [[74]]
## NULL
##
## [[75]]
## NULL
##
## [[76]]
## NULL
##
## [[77]]
## NULL
##
```

```
## [[78]]
## NULL
##
## [[79]]
## NULL
##
## [[80]]
## NULL
##
## [[81]]
## NULL
##
## [[82]]
## NULL
##
## [[83]]
## NULL
##
## [[84]]
## NULL
##
## [[85]]
## NULL
##
## [[86]]
## NULL
##
## [[87]]
## NULL
##
## [[88]]
## NULL
##
## [[89]]
## NULL
##
## [[90]]
## NULL
##
## [[91]]
## NULL
##
## [[92]]
## NULL
##
## [[93]]
## NULL
##
## [[94]]
## NULL
##
## [[95]]
## NULL
##
```



```
## [[96]]
## NULL
##
## [[97]]
## NULL
##
## [[98]]
## NULL
##
## [[99]]
## NULL
##
## [[100]]
## NULL
##
## [[101]]
## NULL
##
## [[102]]
## NULL
##
## [[103]]
## NULL
```

## Transforming the data from wide to long

### Creating new data frame

```
library(tidyr)
library(dplyr)

df_conf <- raw_conf[, 1:4]

for(i in 5:ncol(raw_conf)){

  raw_conf[,i] <- as.integer(raw_conf[,i])
  # raw_conf[,i] <- ifelse(is.na(raw_conf[, i]), 0 , raw_conf[, i])
  print(names(raw_conf)[i])

  if(i == 5){
    df_conf[[names(raw_conf)[i]]] <- raw_conf[, i]
  } else {
    df_conf[[names(raw_conf)[i]]] <- raw_conf[, i] - raw_conf[, i - 1]
  }

}
```

```
## [1] "X1.22.20"
## [1] "X1.23.20"
## [1] "X1.24.20"
## [1] "X1.25.20"
## [1] "X1.26.20"
```

```
## [1] "X1.27.20"  
## [1] "X1.28.20"  
## [1] "X1.29.20"  
## [1] "X1.30.20"  
## [1] "X1.31.20"  
## [1] "X2.1.20"  
## [1] "X2.2.20"  
## [1] "X2.3.20"  
## [1] "X2.4.20"  
## [1] "X2.5.20"  
## [1] "X2.6.20"  
## [1] "X2.7.20"  
## [1] "X2.8.20"  
## [1] "X2.9.20"  
## [1] "X2.10.20"  
## [1] "X2.11.20"  
## [1] "X2.12.20"  
## [1] "X2.13.20"  
## [1] "X2.14.20"  
## [1] "X2.15.20"  
## [1] "X2.16.20"  
## [1] "X2.17.20"  
## [1] "X2.18.20"  
## [1] "X2.19.20"  
## [1] "X2.20.20"  
## [1] "X2.21.20"  
## [1] "X2.22.20"  
## [1] "X2.23.20"  
## [1] "X2.24.20"  
## [1] "X2.25.20"  
## [1] "X2.26.20"  
## [1] "X2.27.20"  
## [1] "X2.28.20"  
## [1] "X2.29.20"  
## [1] "X3.1.20"  
## [1] "X3.2.20"  
## [1] "X3.3.20"  
## [1] "X3.4.20"  
## [1] "X3.5.20"  
## [1] "X3.6.20"  
## [1] "X3.7.20"  
## [1] "X3.8.20"  
## [1] "X3.9.20"  
## [1] "X3.10.20"  
## [1] "X3.11.20"  
## [1] "X3.12.20"  
## [1] "X3.13.20"  
## [1] "X3.14.20"  
## [1] "X3.15.20"  
## [1] "X3.16.20"  
## [1] "X3.17.20"  
## [1] "X3.18.20"  
## [1] "X3.19.20"  
## [1] "X3.20.20"
```

```
## [1] "X3.21.20"
## [1] "X3.22.20"
## [1] "X3.23.20"
## [1] "X3.24.20"
## [1] "X3.25.20"
## [1] "X3.26.20"
## [1] "X3.27.20"
## [1] "X3.28.20"
## [1] "X3.29.20"
## [1] "X3.30.20"
## [1] "X3.31.20"
## [1] "X4.1.20"
## [1] "X4.2.20"
## [1] "X4.3.20"
## [1] "X4.4.20"
## [1] "X4.5.20"
## [1] "X4.6.20"
## [1] "X4.7.20"
## [1] "X4.8.20"
## [1] "X4.9.20"
## [1] "X4.10.20"
## [1] "X4.11.20"
## [1] "X4.12.20"
## [1] "X4.13.20"
## [1] "X4.14.20"
## [1] "X4.15.20"
## [1] "X4.16.20"
## [1] "X4.17.20"
## [1] "X4.18.20"
## [1] "X4.19.20"
## [1] "X4.20.20"
## [1] "X4.21.20"
## [1] "X4.22.20"
## [1] "X4.23.20"
## [1] "X4.24.20"
## [1] "X4.25.20"
## [1] "X4.26.20"
## [1] "X4.27.20"
## [1] "X4.28.20"
## [1] "X4.29.20"

df_conf1 <- df_conf %>% tidyr::pivot_longer(cols = dplyr::starts_with("X"),
                                             names_to = "date_temp",
                                             values_to = "cases_temp")
```

## Parsing the date

```
df_conf1$month <- sub("X", "", strsplit(df_conf1$date_temp, split = "\\.") %>%
  purrr::map_chr(~.x[1]) )

df_conf1$day <- strsplit(df_conf1$date_temp, split = "\\.") %>%
  purrr::map_chr(~.x[2])
```

```
df_conf1$date <- as.Date(paste("2020", df_conf1$month, df_conf1$day, sep = "-"))
```

## Aggregate the data to daily

```
df_conf2 <- df_conf1 %>%
  dplyr::group_by(Province.State, Country.Region, Lat, Long, date) %>%
  dplyr::summarise(cases = sum(cases_temp)) %>%
  dplyr::ungroup() %>%
  dplyr::mutate(type = "confirmed",
                Country.Region = trimws(Country.Region),
                Province.State = trimws(Province.State))
```

## Pulling death cases

```
death_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master
/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_deaths_global.csv"
```

```
raw_death <- read.csv(file = death_url, stringsAsFactors = FALSE, fill = FALSE)
```

```
lapply(1:ncol(raw_death), function(i){
  if(all(is.na(raw_death[, i]))){
    raw_death <- raw_death[, -i]
    return(print(paste("Column", names(raw_death)[i], "is missing", sep = " ")))
  } else {
    return(NULL)
  }
})
```

```
## [[1]]
## NULL
##
## [[2]]
## NULL
##
## [[3]]
## NULL
##
## [[4]]
## NULL
##
## [[5]]
## NULL
##
## [[6]]
## NULL
##
## [[7]]
## NULL
##
## [[8]]
## NULL
```

```
##  
## [[9]]  
## NULL  
##  
## [[10]]  
## NULL  
##  
## [[11]]  
## NULL  
##  
## [[12]]  
## NULL  
##  
## [[13]]  
## NULL  
##  
## [[14]]  
## NULL  
##  
## [[15]]  
## NULL  
##  
## [[16]]  
## NULL  
##  
## [[17]]  
## NULL  
##  
## [[18]]  
## NULL  
##  
## [[19]]  
## NULL  
##  
## [[20]]  
## NULL  
##  
## [[21]]  
## NULL  
##  
## [[22]]  
## NULL  
##  
## [[23]]  
## NULL  
##  
## [[24]]  
## NULL  
##  
## [[25]]  
## NULL  
##  
## [[26]]  
## NULL
```

```
##
## [[27]]
## NULL
##
## [[28]]
## NULL
##
## [[29]]
## NULL
##
## [[30]]
## NULL
##
## [[31]]
## NULL
##
## [[32]]
## NULL
##
## [[33]]
## NULL
##
## [[34]]
## NULL
##
## [[35]]
## NULL
##
## [[36]]
## NULL
##
## [[37]]
## NULL
##
## [[38]]
## NULL
##
## [[39]]
## NULL
##
## [[40]]
## NULL
##
## [[41]]
## NULL
##
## [[42]]
## NULL
##
## [[43]]
## NULL
##
## [[44]]
## NULL
```

```
##
## [[45]]
## NULL
##
## [[46]]
## NULL
##
## [[47]]
## NULL
##
## [[48]]
## NULL
##
## [[49]]
## NULL
##
## [[50]]
## NULL
##
## [[51]]
## NULL
##
## [[52]]
## NULL
##
## [[53]]
## NULL
##
## [[54]]
## NULL
##
## [[55]]
## NULL
##
## [[56]]
## NULL
##
## [[57]]
## NULL
##
## [[58]]
## NULL
##
## [[59]]
## NULL
##
## [[60]]
## NULL
##
## [[61]]
## NULL
##
## [[62]]
## NULL
```

```
##
## [[63]]
## NULL
##
## [[64]]
## NULL
##
## [[65]]
## NULL
##
## [[66]]
## NULL
##
## [[67]]
## NULL
##
## [[68]]
## NULL
##
## [[69]]
## NULL
##
## [[70]]
## NULL
##
## [[71]]
## NULL
##
## [[72]]
## NULL
##
## [[73]]
## NULL
##
## [[74]]
## NULL
##
## [[75]]
## NULL
##
## [[76]]
## NULL
##
## [[77]]
## NULL
##
## [[78]]
## NULL
##
## [[79]]
## NULL
##
## [[80]]
## NULL
```



```
##
## [[81]]
## NULL
##
## [[82]]
## NULL
##
## [[83]]
## NULL
##
## [[84]]
## NULL
##
## [[85]]
## NULL
##
## [[86]]
## NULL
##
## [[87]]
## NULL
##
## [[88]]
## NULL
##
## [[89]]
## NULL
##
## [[90]]
## NULL
##
## [[91]]
## NULL
##
## [[92]]
## NULL
##
## [[93]]
## NULL
##
## [[94]]
## NULL
##
## [[95]]
## NULL
##
## [[96]]
## NULL
##
## [[97]]
## NULL
##
## [[98]]
## NULL
```

```
##
## [[99]]
## NULL
##
## [[100]]
## NULL
##
## [[101]]
## NULL
##
## [[102]]
## NULL
##
## [[103]]
## NULL
```

## Transforming the data from wide to long

### Creating new data frame

```
df_death <- raw_death[, 1:4]

for(i in 5:ncol(raw_death)){
  print(i)
  raw_death[,i] <- as.integer(raw_death[,i])
  raw_death[,i] <- ifelse(is.na(raw_death[, i]), 0 , raw_death[, i])

  if(i == 5){
    df_death[[names(raw_death)[i]]] <- raw_death[, i]
  } else {
    df_death[[names(raw_death)[i]]] <- raw_death[, i] - raw_death[, i - 1]
  }
}
```

```
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 9
## [1] 10
## [1] 11
## [1] 12
## [1] 13
## [1] 14
## [1] 15
## [1] 16
## [1] 17
## [1] 18
## [1] 19
## [1] 20
## [1] 21
## [1] 22
## [1] 23
```

```
## [1] 24
## [1] 25
## [1] 26
## [1] 27
## [1] 28
## [1] 29
## [1] 30
## [1] 31
## [1] 32
## [1] 33
## [1] 34
## [1] 35
## [1] 36
## [1] 37
## [1] 38
## [1] 39
## [1] 40
## [1] 41
## [1] 42
## [1] 43
## [1] 44
## [1] 45
## [1] 46
## [1] 47
## [1] 48
## [1] 49
## [1] 50
## [1] 51
## [1] 52
## [1] 53
## [1] 54
## [1] 55
## [1] 56
## [1] 57
## [1] 58
## [1] 59
## [1] 60
## [1] 61
## [1] 62
## [1] 63
## [1] 64
## [1] 65
## [1] 66
## [1] 67
## [1] 68
## [1] 69
## [1] 70
## [1] 71
## [1] 72
## [1] 73
## [1] 74
## [1] 75
## [1] 76
## [1] 77
```

```
## [1] 78
## [1] 79
## [1] 80
## [1] 81
## [1] 82
## [1] 83
## [1] 84
## [1] 85
## [1] 86
## [1] 87
## [1] 88
## [1] 89
## [1] 90
## [1] 91
## [1] 92
## [1] 93
## [1] 94
## [1] 95
## [1] 96
## [1] 97
## [1] 98
## [1] 99
## [1] 100
## [1] 101
## [1] 102
## [1] 103
```

```
df_death1 <- df_death %>% tidyr::pivot_longer(cols = dplyr::starts_with("X"),
                                              names_to = "date_temp",
                                              values_to = "cases_temp")
```

## Parsing the date

```
df_death1$month <- sub("X", "", strsplit(df_death1$date_temp, split = "\\.") %>%
  purrr::map_chr(~.x[1]) )

df_death1$day <- strsplit(df_death1$date_temp, split = "\\.") %>%
  purrr::map_chr(~.x[2])

df_death1$date <- as.Date(paste("2020", df_death1$month, df_death1$day, sep = "-"))
```

## Aggregate the data to daily

```
df_death2 <- df_death1 %>%
  dplyr::group_by(Province.State, Country.Region, Lat, Long, date) %>%
  dplyr::summarise(cases = sum(cases_temp)) %>%
  dplyr::ungroup() %>%
  dplyr::mutate(type = "death",
               Country.Region = trimws(Country.Region),
               Province.State = trimws(Province.State))
```

## Pulling recovered cases

```
raw_rec <- read.csv(file = "https://raw.githubusercontent.com/CSSEGISandData/COVID-19
/master/csse_covid_19_data/csse_covid_19_time_series
/time_series_covid19_recovered_global.csv", stringsAsFactors = FALSE, fill =FALSE)

lapply(1:ncol(raw_rec), function(i){
  if(all(is.na(raw_rec[, i]))){
    raw_rec <- raw_rec[, -i]
    return(print(paste("Column", names(raw_rec)[i], "is missing", sep = " ")))
  } else {
    return(NULL)
  }
})
```

```
## [[1]]
## NULL
##
## [[2]]
## NULL
##
## [[3]]
## NULL
##
## [[4]]
## NULL
##
## [[5]]
## NULL
##
## [[6]]
## NULL
##
## [[7]]
## NULL
##
## [[8]]
## NULL
##
## [[9]]
## NULL
##
## [[10]]
## NULL
##
## [[11]]
## NULL
##
## [[12]]
## NULL
##
## [[13]]
## NULL
##
```

```
## [[14]]
## NULL
##
## [[15]]
## NULL
##
## [[16]]
## NULL
##
## [[17]]
## NULL
##
## [[18]]
## NULL
##
## [[19]]
## NULL
##
## [[20]]
## NULL
##
## [[21]]
## NULL
##
## [[22]]
## NULL
##
## [[23]]
## NULL
##
## [[24]]
## NULL
##
## [[25]]
## NULL
##
## [[26]]
## NULL
##
## [[27]]
## NULL
##
## [[28]]
## NULL
##
## [[29]]
## NULL
##
## [[30]]
## NULL
##
## [[31]]
## NULL
##
```

```
## [[32]]
## NULL
##
## [[33]]
## NULL
##
## [[34]]
## NULL
##
## [[35]]
## NULL
##
## [[36]]
## NULL
##
## [[37]]
## NULL
##
## [[38]]
## NULL
##
## [[39]]
## NULL
##
## [[40]]
## NULL
##
## [[41]]
## NULL
##
## [[42]]
## NULL
##
## [[43]]
## NULL
##
## [[44]]
## NULL
##
## [[45]]
## NULL
##
## [[46]]
## NULL
##
## [[47]]
## NULL
##
## [[48]]
## NULL
##
## [[49]]
## NULL
##
```

```
## [[50]]
## NULL
##
## [[51]]
## NULL
##
## [[52]]
## NULL
##
## [[53]]
## NULL
##
## [[54]]
## NULL
##
## [[55]]
## NULL
##
## [[56]]
## NULL
##
## [[57]]
## NULL
##
## [[58]]
## NULL
##
## [[59]]
## NULL
##
## [[60]]
## NULL
##
## [[61]]
## NULL
##
## [[62]]
## NULL
##
## [[63]]
## NULL
##
## [[64]]
## NULL
##
## [[65]]
## NULL
##
## [[66]]
## NULL
##
## [[67]]
## NULL
##
```



```
## [[68]]
## NULL
##
## [[69]]
## NULL
##
## [[70]]
## NULL
##
## [[71]]
## NULL
##
## [[72]]
## NULL
##
## [[73]]
## NULL
##
## [[74]]
## NULL
##
## [[75]]
## NULL
##
## [[76]]
## NULL
##
## [[77]]
## NULL
##
## [[78]]
## NULL
##
## [[79]]
## NULL
##
## [[80]]
## NULL
##
## [[81]]
## NULL
##
## [[82]]
## NULL
##
## [[83]]
## NULL
##
## [[84]]
## NULL
##
## [[85]]
## NULL
##
```

```
## [[86]]
## NULL
##
## [[87]]
## NULL
##
## [[88]]
## NULL
##
## [[89]]
## NULL
##
## [[90]]
## NULL
##
## [[91]]
## NULL
##
## [[92]]
## NULL
##
## [[93]]
## NULL
##
## [[94]]
## NULL
##
## [[95]]
## NULL
##
## [[96]]
## NULL
##
## [[97]]
## NULL
##
## [[98]]
## NULL
##
## [[99]]
## NULL
##
## [[100]]
## NULL
##
## [[101]]
## NULL
##
## [[102]]
## NULL
##
## [[103]]
## NULL
```

## Transforming the data from wide to long

### Creating new data frame

```
df_rec <- raw_rec[, 1:4]

for(i in 5:ncol(raw_rec)){
  print(i)
  raw_rec[,i] <- as.integer(raw_rec[,i])
  raw_rec[,i] <- ifelse(is.na(raw_rec[, i]), 0 , raw_rec[, i])

  if(i == 5){
    df_rec[[names(raw_rec)[i]]] <- raw_rec[, i]
  } else {
    df_rec[[names(raw_rec)[i]]] <- raw_rec[, i] - raw_rec[, i - 1]
  }
}
```

```
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 9
## [1] 10
## [1] 11
## [1] 12
## [1] 13
## [1] 14
## [1] 15
## [1] 16
## [1] 17
## [1] 18
## [1] 19
## [1] 20
## [1] 21
## [1] 22
## [1] 23
## [1] 24
## [1] 25
## [1] 26
## [1] 27
## [1] 28
## [1] 29
## [1] 30
## [1] 31
## [1] 32
## [1] 33
## [1] 34
## [1] 35
## [1] 36
## [1] 37
## [1] 38
## [1] 39
## [1] 40
```

```
## [1] 41
## [1] 42
## [1] 43
## [1] 44
## [1] 45
## [1] 46
## [1] 47
## [1] 48
## [1] 49
## [1] 50
## [1] 51
## [1] 52
## [1] 53
## [1] 54
## [1] 55
## [1] 56
## [1] 57
## [1] 58
## [1] 59
## [1] 60
## [1] 61
## [1] 62
## [1] 63
## [1] 64
## [1] 65
## [1] 66
## [1] 67
## [1] 68
## [1] 69
## [1] 70
## [1] 71
## [1] 72
## [1] 73
## [1] 74
## [1] 75
## [1] 76
## [1] 77
## [1] 78
## [1] 79
## [1] 80
## [1] 81
## [1] 82
## [1] 83
## [1] 84
## [1] 85
## [1] 86
## [1] 87
## [1] 88
## [1] 89
## [1] 90
## [1] 91
## [1] 92
## [1] 93
## [1] 94
```

```
## [1] 95
## [1] 96
## [1] 97
## [1] 98
## [1] 99
## [1] 100
## [1] 101
## [1] 102
## [1] 103

df_rec1 <- df_rec %>% tidyr::pivot_longer(cols = dplyr::starts_with("X"),
                                         names_to = "date_temp",
                                         values_to = "cases_temp")
```

## Parsing the date

```
df_rec1$month <- sub("X", "", strsplit(df_rec1$date_temp, split = "\\.") %>%
  purrr::map_chr(~.x[1]) )

df_rec1$day <- strsplit(df_rec1$date_temp, split = "\\.") %>%
  purrr::map_chr(~.x[2])

df_rec1$date <- as.Date(paste("2020", df_rec1$month, df_rec1$day, sep = "-"))
```

## Aggregate the data to daily

```
df_rec2 <- df_rec1 %>%
  dplyr::group_by(Province.State, Country.Region, Lat, Long, date) %>%
  dplyr::summarise(cases = sum(cases_temp)) %>%
  dplyr::ungroup() %>%
  dplyr::mutate(type = "recovered",
               Country.Region = trimws(Country.Region),
               Province.State = trimws(Province.State))

#----- Aggregate all cases -----

coronavirus <- dplyr::bind_rows(df_conf2, df_death2, df_rec2) %>% as.data.frame()

head(coronavirus)
```

##	Province.State	Country.Region	Lat	Long	date	cases	type
## 1		Afghanistan	33	65	2020-01-22	0	confirmed
## 2		Afghanistan	33	65	2020-01-23	0	confirmed
## 3		Afghanistan	33	65	2020-01-24	0	confirmed
## 4		Afghanistan	33	65	2020-01-25	0	confirmed
## 5		Afghanistan	33	65	2020-01-26	0	confirmed
## 6		Afghanistan	33	65	2020-01-27	0	confirmed

## Remove variables such as province.state, Lat and Long

```
corona_global <-coronavirus %>%  
  select( -Province.State, -Lat, -Long) %>%  
  rename(country = Country.Region)      #rename the variable country.region to country  
  
head(corona_global)
```

```
##      country      date cases      type  
## 1 Afghanistan 2020-01-22      0 confirmed  
## 2 Afghanistan 2020-01-23      0 confirmed  
## 3 Afghanistan 2020-01-24      0 confirmed  
## 4 Afghanistan 2020-01-25      0 confirmed  
## 5 Afghanistan 2020-01-26      0 confirmed  
## 6 Afghanistan 2020-01-27      0 confirmed
```

## Introduce a variable for cummulative totals of the cases called total\_cases

```
corona_global_total <- corona_global %>%  
  group_by(country, type ) %>%  
  mutate(total_cases = cumsum(cases))  
head(corona_global_total)
```

```
## # A tibble: 6 x 5  
## # Groups:   country, type [1]  
##   country      date      cases type      total_cases  
##   <chr>      <date>    <int> <chr>    <int>  
## 1 Afghanistan 2020-01-22      0 confirmed      0  
## 2 Afghanistan 2020-01-23      0 confirmed      0  
## 3 Afghanistan 2020-01-24      0 confirmed      0  
## 4 Afghanistan 2020-01-25      0 confirmed      0  
## 5 Afghanistan 2020-01-26      0 confirmed      0  
## 6 Afghanistan 2020-01-27      0 confirmed      0
```

## filter out Nigeria

```
COVID_19_Nigeria <- corona_global_total %>%  
  select(-cases) %>%  
  filter(country!="Nigeria")  
  
head(COVID_19_Nigeria)
```

```
## # A tibble: 6 x 4  
## # Groups:   country, type [1]  
##   country date      type      total_cases  
##   <chr>   <date>    <chr>    <int>  
## 1 Nigeria 2020-01-22 confirmed      0  
## 2 Nigeria 2020-01-23 confirmed      0  
## 3 Nigeria 2020-01-24 confirmed      0  
## 4 Nigeria 2020-01-25 confirmed      0
```

```
## 5 Nigeria 2020-01-26 confirmed      0
## 6 Nigeria 2020-01-27 confirmed      0
```

## Export it in .csv and .xlsx formarts

```
writexl::write_xlsx(x = COVID_19_Nigeria, path = "C:/Users/uganda/Documents/COVID-19/COVID-19_Data/COVID_19_Nigeria.xlsx")
write.csv(COVID_19_Nigeria, "C:/Users/uganda/Documents/COVID-19/COVID-19_Data/COVID_19_Nigeria.csv", row.names = FALSE)
```