#Load in Libraries

install.packages ("tidyverse")

install.packages ("pkgbuild")

install.packages ("gifski")

install.packages("gganimate")

install.packages("png")

install.packages("ggplot2")

install.packages("dplyr")

library(ggplot2)

library(pkgbuild)

library(gifski)

library(gganimate)

library(png)

library(tidyverse)

library(dplyr)

#Time Series Data#

### Pre plotting ###

# Check and set working directory

getwd()

setwd("/Users/kf/Documents/ ")#ensure here is updated to your file location

# Import WDI data

df <- read.csv("WDI\_Data\_Final.csv")

# Check structure of data

head(df)

str(df)

#Replace column names in column 5-29

colnames(df)[5:29]<-c(1998:2022)

#Remove inputs with ".." missing value to get it ready for plotting by replacing it with empty string

df == ".." #booleans will show TRUE for missing data, FALSE for cell with data

class(df == "..") #can observe it's a matrix

df[df == ".."] = "" #by passing the matrix in the dataset, the TRUE cells will be replaced with empty cell

#Check structure of data and change necessary variables from character to numeric

str(df)

df[5:ncol(df)] <- lapply(df[5:ncol(df)],as.numeric)

df <- df %>%

select(-Series.Code) #to "drop" series code from the df

#pipe together pivoting

# Use pivot\_longer to Collect the Variables Under Years as a Single Column

#Use pivot\_wider to extract out variables lumped in SeriesName

df %>%

pivot\_longer(cols = 4:ncol(df), names\_to = "Year", values\_to = "Values") %>%

pivot\_wider(names\_from = Series.Name, values\_from = "Values") -> df2

#Check structure of data and change necessary variables from character to numeric

str(df2) #year should all be numeric

df2[3] <- lapply(df2[3],as.numeric)

#Rename columns so it reads better

colnames(df2) <- c("Country", "CountryCode","Year", "GDPGrowth", "GDPPerCap","ChildMortality")

#Arrange data in ascending order

df2 <- arrange(df2, CountryCode, Year)

#Read in metadata to extract Region and Income Group

install.packages("readxl")

library(readxl)

meta\_data <- read\_excel("WDI\_Data\_Final\_Metadata.xlsx", sheet = "Country - Metadata")

meta\_data2 <- select(meta\_data,"Code","Income Group","Region")

#Merging meta\_data2 with df2

data\_merged <- merge(df2,meta\_data2, by.x="CountryCode", by.y="Code")

head(data\_merged)

names(data\_merged)

#Exporting clean dataset

write.csv(data\_merged, file="WDI\_clean\_Final.csv")

df3 <- read.csv("WDI\_clean\_Final.csv")

### Plot 1 - Track Singapore's GDP Growth YoY ###

# Create subset of Singapore, declare the data and aesthetics globally

sg <- subset(df3,Country %in% "Singapore")

sg.ggplot <- ggplot(data = sg, mapping = aes(x = Year, y = GDPGrowth))

sg.ggplot + geom\_point()

#Overlay a nonlinear trend and suppress confidence bonds

sg.ggplot + geom\_point() +

geom\_smooth(se = FALSE)

#Change up labels, add title and add theme

sg.ggplot + geom\_point() +

geom\_smooth(se = FALSE,color = "grey") +

geom\_path(aes(group = 1),color = "red")+

labs(x = "Year", y = "GDP Growth",

title = "Singapore's GDP Growth Overtime",

caption = "Source: WDI") +

theme\_light()

### Plot 2 - Correlation Between GDP Per Capitia and Child Mortality###

#Check if region has empty cells

summary(df3)

unique(df3$Region)

#Remove empty label from Region and check

df\_regionfiltered <- na.omit(df3)

unique(df\_regionfiltered$Region) #to make sure the empty label is removed

# Declare the data and aesthetics globally. Name it ggplot.2

ggplot.2 <- ggplot(data = df\_regionfiltered, mapping = aes(x = GDPPerCap, y = ChildMortality))

ggplot.2 + geom\_point()

#Overlay a linear trend

ggplot.2 + geom\_point() +

geom\_smooth(method = "lm", se = FALSE)

#Adjust x-scale so it's easier to comprehend and looks cleaner

ggplot.2 + geom\_point() +

geom\_smooth(method = "lm", se = FALSE) +

scale\_x\_log10(labels = scales::dollar)

#Use Region as color aesthetic

ggplot.2+ geom\_point(aes(group = Region, color = Region))+

geom\_smooth(method = "lm", se = FALSE, color = "red") +

scale\_x\_log10(labels = scales::dollar)

#Add title, clean up axis labels and add theme

ggplot.2+ geom\_point(aes(group = Region, color = Region))+

geom\_smooth(method = "lm", se = FALSE, color = "red") +

scale\_x\_log10(labels = scales::dollar) +

labs(x = "GDP Per Capita", y = "Child Mortality",

title = "Economic Growth & Child Mortality",

subtitle = "Data Points Are Country-Years",

caption = "Source: WDI") +

theme\_bw()

### Plot 3 - North America's GDP Growth Over Time by Country ###

# Create a subset for North America

na.subset <- subset(df3, Region %in% "North America")

# Declare the data and aesthetics globally. Name it ggplot.3

ggplot.3 <- ggplot(na.subset, mapping = aes(x = Year, y = GDPGrowth, color = Country))

#pass it through segment, add labels, theme

ggplot.3 + geom\_line() + geom\_point(size = 3) +

labs(x = "Year", y = "GDP Growth", title = "North America's GDP Growth Over Time by Country", caption = "Source: WDI")+

theme\_bw()

### Animate Plot 2 above#

#Label the whole plot to final.plot2

final.plot2 <-ggplot.2+ geom\_point(aes(color = Region))+

geom\_smooth(method = "lm", se = FALSE, color = "red") +

scale\_x\_log10(labels = scales::dollar) +

labs(x = "GDP Per Capita", y = "Child Mortality",

title = "Economic Growth & Child Mortality",

subtitle = "Data Points Are Country-Years",

caption = "Source: WDI") +

theme\_bw()

#Install and use the libraries:

install.packages ("tidyverse")

install.packages ("ggimage")

install.packages ("gifski")

install.packages("gganimate")

install.packages("transformr")

install.packages("ggplot2")

library(tidyverse)

library(ggimage)

library(gifski)

library(gganimate)

library(transformr)

library(ggplot2)

#Add final.plot2 to wrap by Region, declare axis, labels and add transitioned time

final.plot2 + facet\_wrap(~Region, nrow = 4) +

theme(title = element\_text(size=18),

axis.text.x = element\_text(size = 10), # x-axis grid text

axis.text.y = element\_text(size = 10), # y-axis grid text

axis.title = element\_text(size = 16),

strip.text = element\_text(size = 10), #adjust facet label size

legend.position = "none") + #to remove legends

labs(title = 'Year: {frame\_time}', x = 'GDP Per Capita', y = 'Child Mortality', caption = 'Source: WDI') +

transition\_time(Year) +

ease\_aes('linear') #to show how the linear line changes

#Save the code chunk as plot2.anim

plot2.anim <- final.plot2 + facet\_wrap(~Region, nrow = 4) +

theme(title = element\_text(size=18),

axis.text.x = element\_text(size = 10), # x-axis grid text

axis.text.y = element\_text(size = 10), # y-axis grid text

axis.title = element\_text(size = 16),

strip.text = element\_text(size = 10), #adjust facet label size

legend.position = "none") + #to remove legends

labs(title = 'Year: {frame\_time}', x = 'GDP Per Capita', y = 'Child Mortality', caption = 'Source: WDI') +

transition\_time(Year) +

ease\_aes('linear')

plot2.anim

#Render the animated plot and save it

animate(plot2.anim, fps=5, renderer = gifski\_renderer())

anim\_save("plot2\_1c\_final.gif", animation = plot2.anim)

#Check North America's lowest and highest child mortality, while ignoring the NA records

NA.subset <- subset(df\_regionfiltered, Region %in% "North America")

min(NA.subset$ChildMortality, na.rm = T)

max(NA.subset$ChildMortality, na.rm = T)

#Check North America's GDP Per Cap mean

mean(NA.subset$GDPPerCap, na.rm = T)