**Living Document – Rstudio Markdown File for Data Demons Final Project**

Let’s use this document to create, edit, and share our coding discoveries for the final project. A key part of the project will be working from the same code, and thus, seeing the same dataset.

Feel free to make changes via GitHub, so we can track them and see different versions as needed.

\*\*\*\*\*\*\*\*\*\*\*\*

---

title: "MGT-585-final-project"

author: "Jeff Rolling"

date: "6/26/2020"

output: html\_document

---

```{r setup, include=FALSE}

knitr::opts\_chunk$set(echo = TRUE)

```

### Reading Multiple Files ---------

The code below will import all .csv files within my working directory and bind them into one large dataset.

```{r}

library(tidyverse)

library(dplyr) #includes data wrangling abilities including renaming columns to be consistent.

library(readr)

library(skimr)

library(janitor) #changes all column names to lowercase and adds underscores.

```

```{r}

export\_reviews <- list.files(pattern = "\*.csv") %>%

lapply(read.csv) %>%

bind\_rows #when row binding, columns are matched by name and any missing columns will be filled with NA.

head(export\_reviews)

```

### Cleaning the Data ---------

Here's a place to begin cleaning up the data for analysis.

```{r}

# using a pipe to perform multiple cleaning steps at once

cleanExports <- export\_reviews %>%

clean\_names() %>% # makes column names lowercase and adds underscores

rename(app\_name = i\_app\_name) # when renaming columns, use newName = oldName

cleanExports$app\_name <- as.factor(cleanExports$app\_name) # converts this column into factor/categorical data.

cleanExports$app\_store <- as.factor(cleanExports$app\_store)

cleanExports$app <- as.factor(cleanExports$app)

cleanExports$store <- as.factor(cleanExports$store)

cleanExports$country <- as.factor(cleanExports$country)

cleanExports$device <- as.factor(cleanExports$device)

cleanExports$app\_id <- as.factor(cleanExports$app\_id)

cleanExports$review\_id <- as.factor(cleanExports$review\_id)

cleanExports$version <- as.factor(cleanExports$version)

cleanExports$emotion <- as.factor(cleanExports$emotion)

```

```{r}

cleanExports$app\_name\_app\_store\_app\_store\_app\_id\_review\_id\_country\_version\_rating\_date\_author\_subject\_body\_translated\_subject\_translated\_body\_emotion\_device <- NULL # deleting column

cleanExports$device <- NULL # device column is empty for all .csv files

cleanExports$unique\_id <- seq\_len(nrow(cleanExports)) # creating a unique ID for each row

names(cleanExports)

dim(cleanExports)

```

### Exploring the Data ---------

Here's a place to begin exploring the data.

```{r}

dim(cleanExports)

class(cleanExports)

summary(cleanExports)

```

```{r}

# This code lets you view the 34,831 NA's in dataset. Upon a closer look, it appears that this is junk data because all columns have NA. Could also be a problem in the bind\_row command that merged the dataset.

# Device column appears empty for all .csv files. Recommend deleting this column.

# summary() will give you the top 5 categories and counts. Other categories are included under (Other).

appName\_na <- cleanExports %>%

filter(is.na(app\_name))

# Trying to use code to see if the ‘app\_name’ and ‘app’ columns are identical info.

appName\_appStore <- select(cleanExports, app\_name, app)

appName\_appStore$compare <- appName\_appStore$app\_name != appName\_appStore$app

(appName\_appStore)

dim(appName\_appStore)

table(appName\_appStore$compare)

```