Problem Set 1 Key

1. Open the dataset as a dataframe. This dataframe should have the following properties: a) The column names should match the column names in the original dataset. b) The row names should correspond to the variable ID in the original dataset.

```
Expends2002 <- read.csv(
"C:/Users/dl0ck/OneDrive/Spring2020/Political Data Science/Problem Sets/PS 1/Expends2002.csv",
na.strings="", stringsAsFactors=FALSE)
Expends2002 <- as.data.frame(Expends2002)
```

2. Change the variable name TransID to Useless.

```
colnames(Expends2002)
```

```
##  [1] "Cycle"        "ID"           "TransID"      "CRPFilerid"   "Recipcode"
##  [6] "Pacshort"     "CRPRecipname" "Expcode"      "Amount"       "Date"
## [11] "City"         "State"        "Zip"          "CmtelD_EF"    "Candid"
## [16] "Type"         "Descrip"      "PG"           "ElecOther"    "EntType"
## [21] "Source"
```

```
colnames(Expends2002)[3] <- "Useless"
colnames(Expends2002)
```

```
##  [1] "Cycle"        "ID"           "Useless"      "CRPFilerid"   "Recipcode"
##  [6] "Pacshort"     "CRPRecipname" "Expcode"      "Amount"       "Date"
## [11] "City"         "State"        "Zip"          "CmtelD_EF"    "Candid"
## [16] "Type"         "Descrip"      "PG"           "ElecOther"    "EntType"
## [21] "Source"
```

3. Remove the variables Useless, and Source from the dataframe

```
Expends2002 <- Expends2002[,-3]
colnames(Expends2002)
```

```
##  [1] "Cycle"        "ID"           "CRPFilerid"   "Recipcode"    "Pacshort"
##  [6] "CRPRecipname" "Expcode"      "Amount"       "Date"         "City"
## [11] "State"        "Zip"          "CmtelD_EF"    "Candid"       "Type"
## [16] "Descrip"      "PG"           "ElecOther"    "EntType"      "Source"
```

4. Change the variable EntType to a factor. How many levels does this variable have?

```
Expends2002$EntType <- as.factor(Expends2002$EntType)
```

5. The variable State contains several obvious errors, as it includes nonexistent state codes.

- Identify observations that have non-existent state codes.
- Write a script to recode these observations. Use the additional information in the dataset (candidate name, city, zip code) to correctly identify each state.

```
unique(Expends2002$State)
```

```
##  [1] "IL" "DC" "MA" "PA" "KY" "TN" "MI" "TX" "IA" "CA" "NJ" "MO" "NE" "MD" "VA"
## [16] "NV" "GA" "  " "MN" "NY" "UT" "SC" "AZ" "IN" "NH" "OH" "CO" "VT" "LA" "AK"
```

```
## [31] "DE" "AR" "OR" "NM" "WI" "AL" "NC" "OK" "ME" "WA" "CT" "FL" "KS" "RI" "WY"
## [46] "MS" "MT" "SD" "HI" "ND" "WV" "ID" "GU" "VI" "AS" "St" "ZZ" "LL"
```

```
wrongstate<-subset(Expends2002,State=="St" | State=="ZZ" | State=="LL")
Expends2002$State[Expends2002$State=="LL"] <- "IA"
Expends2002$State[Expends2002$State=="ZZ"] <- "VI"
Expends2002$State[Expends2002$State=="St"] <- "DC"
```

6. Remove all observations from the dataset where the variable State is missing. Report the number of observations after removing missing values.

```
Expends2002<-subset(Expends2002,State!="  ")
nrow(Expends2002)
```

```
## [1] 19912
```

7. Change the variable Zip into a numeric. Be sure to document what you do with missing cases. What is the mean of this variable?

```
Expends2002$Zip <- as.numeric(Expends2002$Zip)
```

```
## Warning: NAs introduced by coercion
```

```
#Missing cases become NA
mean(Expends2002$Zip, na.rm=T)
```

```
## [1] 48214902
```

8. Create new variables that contain the following information (you will be making several variables), and answer the questions:

+ The number of words in the Descrip variable. What is the median value of this new variable?

```
#Part 1
df <- NULL
df$descrip<- Expends2002$Descrip
df <- as.data.frame(df)
df$total <- sapply(df$descrip, function(x) length(unlist(strsplit(as.character(x), "\\W+"))))
sum(df$total, na.rm =T) #The number of words in the Descrip variable
```

```
## [1] 61164
```

```
median(df$total, na.rm=T) # The median number of words in each row
```

```
## [1] 2
```

+ A variable containing the numeric portion of CRPFilerid. This variable should be of length 8 for all observations. What is the number of unique values of this variable?

```
#Part 2
df$filerid <- Expends2002$CRPFilerid
df$filerid <- as.character(df$filerid)
df$filerid <- sapply(df$filerid, function(x) gsub("[^0-9.-]", "", x)) #get rid of all the leters
length(unique(df$filerid)) #number of unique values in this variable
```

```
## [1] 2243
```

+ A vector containing the first four digits of Zip. What is the most frequent value of this vector?

```
#Part 3
df$zip <- substr(as.character(Expends2002$Zip), 0,4)
```

```
#df$zip
names(which(max(table(df$zip))==table(df$zip)))
```

```
## [1] "2000"
```

+ A boolean indicating whether the Descrip variable contains the word "Communications" REGARDLESS OF CAPITALIZATION. Report the number of TRUE values in this boolean.

```
#Part 4
df$descrip <- as.character(df$descrip)
#df$descrip
#grepl("communications", df$descrip, ignore.case=T)
sum(grepl("communications", df$descrip, ignore.case=T)==T)
```

```
## [1] 9
```

+ A variable indicating that either CRPFilerid is "N" or that BOTH Amount is greater than 500 and Descrip is non-missing. Report the number of TRUE values.

```
#Part 5
length(as.character(Expends2002$CRPFilerid))
```

```
## [1] 19912
```

```
df$variable <- ifelse(grepl('N', as.character(Expends2002$CRPFilerid), ignore.case=T)==T, 1,0
| as.numeric(Expends2002$Amount >500) & Expends2002$Descrip!="")
#df$variable #Variable indicating whether variable passes or fails above specifications
#Expends2002[,c("CRPFilerid", "Amount", "Descrip")]
```

+ EXTRA CREDIT: A variable that provides the most commonletter in the Descrip variable.

```
#Part 6
df$descrip <- gsub(" ", "", df$descrip, fixed = T)
names(table(unlist(strsplit(df$descrip, split =
""))))[as.vector(table(unlist(strsplit(df$descrip, split =
""))))==max(table(unlist(strsplit(df$descrip, split = ""))))]
```

```
## [1] "e"
```

9. Write a script that subsets the data by state, and writes out a unique CSV file for each subset, where each file has a unique (and meaningful)name.

```
data <- by(Expends2002, Expends2002$State, function(x){Expends2002})
length(data)
```

```
## [1] 54
```

```
#lapply(1:length(data), function(i) write.csv(data[[i]], file =
#paste0(names(data[i]), ".csv"), row.names = FALSE))
```

‘‘‘