```
---
title: "Straddle Screening Tool"
# Bassam Rizk
# YUid 303525
output: html_notebook
---

### Research Question
# What is the highest reachable consistency in screening stocks that will exhibit abnormal price volatility at a foreseeable market or stock event (dividends announcement…)?
# Candidate stocks should have their at-the-money options tree premiums exhibiting high correlation (positive for calls and negative for puts) with their underlying stocks’ prices.


### Methodology

#Data sources:
  #S&P 100 Stock history (10 years)
  #S&P stock dividends events
  #Option Tree historical data for at the market Call/Put Option (api on sprint 2)
  #Greeks historical data - specifically delta (api on sprint 2)


#Consolidate & clean data
  #Divide data into test & training sets
  #Using training data sets:
      #Isolate historical straddle opportunities during the last 10 years (where stock price had a 5% increase with similar increase in option prices)
  #Clustering for optimal straddle opportunities using multi variables (strike price in the option tree, Greeks, market depths, beta, volatility)
  #Using testing data set:
  #Test the top 10 selection combinations

library(quandl)
library(QuantTools)
library(quantmod)
library(derivmkts)
library(RND)
setDefaults(getSymbols.av, api.key="V7YC53BOMBUB28FJ")




####Downloading Data

#There weren't any API to filter S&P100 stocks of all listed stockes. So, I got a list of S&P 100 from Wikipedia
#Converted that list from table to csv using Excel / notepad (I tried doing that using R - it was too complicated...)
#Copy/ pasted that list into a getSymbols function from Quantmod.


getSymbols(c('AAPL', 'ABBV', 'ABT', 'ACN', 'ADBE', 'AGN', 'AIG', 'ALL', 'AMGN', 'AMZN', 'AXP', 'BA', 'BAC', 'BIIB', 'BK', 'BKNG', 'BLK', 'BMY', 'BRK.B', 'C', 'CAT', 'CELG', 'CHTR', 'CL', 'CMCSA', 'COF', 'COP', 'COST', 'CSCO', 'CVS',
'CVX', 'DD', 'DHR', 'DIS', 'DOW', 'DUK', 'EMR', 'EXC', 'F', 'FB', 'FDX', 'GD', 'GE', 'GILD', 'GM', 'GOOG', 'GOOGL', 'GS', 'HD', 'HON', 'IBM', 'INTC', 'JNJ', 'JPM', 'KHC', 'KMI', 'KO', 'LLY', 'LMT', 'LOW', 'MA', 'MCD', 'MDLZ', 'MDT',
'MET', 'MMM', 'MO', 'MRK', 'MS', 'MSFT', 'NEE', 'NFLX', 'NKE', 'NVDA', 'ORCL', 'OXY', 'PEP', 'PFE', 'PG', 'PM', 'PYPL', 'QCOM', 'RTN', 'SBUX', 'SLB', 'SO', 'SPG', 'T', 'TGT', 'TXN', 'UNH', 'UNP', 'UPS', 'USB', 'UTX', 'V', 'VZ', 'WBA',
'WFC', 'WMT', 'XOM'
))
#This worked!!!!!!
# I received in my environment 100 xts objects with history from 2007-01-03 to 2019-08-02



##MEASURE VOLATILITY - I needed to add a trailing volalitity measure of each stock for each time point.

#I added a column to measure volatility of each stock - I tried doing that in one function, it got too complicated.
AAPL$VOLA<- volatility(AAPL)
ABBV$VOLA<- volatility(ABBV)
ABT$VOLA<- volatility(ABT)
ACN$VOLA<- volatility(ACN)
ADBE$VOLA<- volatility(ADBE)
AGN$VOLA<- volatility(AGN)
AIG$VOLA<- volatility(AIG)
ALL$VOLA<- volatility(ALL)
AMGN$VOLA<- volatility(AMGN)
AMZN$VOLA<- volatility(AMZN)
AXP$VOLA<- volatility(AXP)
BA$VOLA<- volatility(BA)
BAC$VOLA<- volatility(BAC)
BIIB$VOLA<- volatility(BIIB)
BK$VOLA<- volatility(BK)
BKNG$VOLA<- volatility(BKNG)
BLK$VOLA<- volatility(BLK)
BMY$VOLA<- volatility(BMY)
C$VOLA<- volatility(C)
CAT$VOLA<- volatility(CAT)
CELG$VOLA<- volatility(CELG)
CHTR$VOLA<- volatility(CHTR)
CL$VOLA<- volatility(CL)
CMCSA$VOLA<- volatility(CMCSA)
COF$VOLA<- volatility(COF)
COP$VOLA<- volatility(COP)
COST$VOLA<- volatility(COST)
CSCO$VOLA<- volatility(CSCO)
CVS$VOLA<- volatility(CVS)
CVX$VOLA<- volatility(CVX)
DD$VOLA<- volatility(DD)
DHR$VOLA<- volatility(DHR)
DIS$VOLA<- volatility(DIS)
DOW$VOLA<- volatility(DOW)
DUK$VOLA<- volatility(DUK)
EMR$VOLA<- volatility(EMR)
EXC$VOLA<- volatility(EXC)
F$VOLA<- volatility(F)
FB$VOLA<- volatility(FB)
FDX$VOLA<- volatility(FDX)
GD$VOLA<- volatility(GD)
GE$VOLA<- volatility(GE)
GILD$VOLA<- volatility(GILD)
GM$VOLA<- volatility(GM)
GOOG$VOLA<- volatility(GOOG)
GOOGL$VOLA<- volatility(GOOGL)
GS$VOLA<- volatility(GS)
HD$VOLA<- volatility(HD)
HON$VOLA<- volatility(HON)
IBM$VOLA<- volatility(IBM)
INTC$VOLA<- volatility(INTC)
JNJ$VOLA<- volatility(JNJ)
JPM$VOLA<- volatility(JPM)
KHC$VOLA<- volatility(KHC)
KMI$VOLA<- volatility(KMI)
KO$VOLA<- volatility(KO)
LLY$VOLA<- volatility(LLY)
LMT$VOLA<- volatility(LMT)
LOW$VOLA<- volatility(LOW)
MA$VOLA<- volatility(MA)
MCD$VOLA<- volatility(MCD)
MDLZ$VOLA<- volatility(MDLZ)
MDT$VOLA<- volatility(MDT)
MET$VOLA<- volatility(MET)
MMM$VOLA<- volatility(MMM)
MO$VOLA<- volatility(MO)
MRK$VOLA<- volatility(MRK)
MS$VOLA<- volatility(MS)
MSFT$VOLA<- volatility(MSFT)
NEE$VOLA<- volatility(NEE)
NFLX$VOLA<- volatility(NFLX)
NKE$VOLA<- volatility(NKE)
NVDA$VOLA<- volatility(NVDA)
ORCL$VOLA<- volatility(ORCL)
OXY$VOLA<- volatility(OXY)
PEP$VOLA<- volatility(PEP)
PFE$VOLA<- volatility(PFE)
PG$VOLA<- volatility(PG)
PM$VOLA<- volatility(PM)
PYPL$VOLA<- volatility(PYPL)
QCOM$VOLA<- volatility(QCOM)
RTN$VOLA<- volatility(RTN)
SBUX$VOLA<- volatility(SBUX)
SLB$VOLA<- volatility(SLB)
SO$VOLA<- volatility(SO)
SPG$VOLA<- volatility(SPG)
T$VOLA<- volatility(T)
TGT$VOLA<- volatility(TGT)
TXN$VOLA<- volatility(TXN)
UNH$VOLA<- volatility(UNH)
UNP$VOLA<- volatility(UNP)
UPS$VOLA<- volatility(UPS)
USB$VOLA<- volatility(USB)
UTX$VOLA<- volatility(UTX)
V$VOLA<- volatility(V)
VZ$VOLA<- volatility(VZ)
WBA$VOLA<- volatility(WBA)
WFC$VOLA<- volatility(WFC)
WMT$VOLA<- volatility(WMT)
XOM$VOLA<- volatility(XOM)

#remove NAs FROM FIRST ROWS IN THE VOLATILITY COLUMN

AAPL[is.na(AAPL)] <-0
ABBV[is.na(ABBV)] <-0
ABT[is.na(ABT)] <-0
ACN[is.na(ACN)] <-0
ADBE[is.na(ADBE)] <-0
AGN[is.na(AGN)] <-0
AIG[is.na(AIG)] <-0
ALL[is.na(ALL)] <-0
```

```
AMGN[is.na(AMGN)] <-0
AMZN[is.na(AMZN)] <-0
AXP[is.na(AXP)] <-0
BA[is.na(BA)] <-0
BAC[is.na(BAC)] <-0
BIIB[is.na(BIIB)] <-0
BK[is.na(BK)] <-0
BKNG[is.na(BKNG)] <-0
BLK[is.na(BLK)] <-0
BMY[is.na(BMY)] <-0
C[is.na(C)] <-0
CAT[is.na(CAT)] <-0
CELG[is.na(CELG)] <-0
CHTR[is.na(CHTR)] <-0
CL[is.na(CL)] <-0
CMCSA[is.na(CMCSA)] <-0
COF[is.na(COF)] <-0
COP[is.na(COP)] <-0
COST[is.na(COST)] <-0
CSCO[is.na(CSCO)] <-0
CVS[is.na(CVS)] <-0
CVX[is.na(CVX)] <-0
DD[is.na(DD)] <-0
DHR[is.na(DHR)] <-0
DIS[is.na(DIS)] <-0
DOW[is.na(DOW)] <-0
DUK[is.na(DUK)] <-0
EMR[is.na(EMR)] <-0
EXC[is.na(EXC)] <-0
F[is.na(F)] <-0
FB[is.na(FB)] <-0
FDX[is.na(FDX)] <-0
GD[is.na(GD)] <-0
GE[is.na(GE)] <-0
GILD[is.na(GILD)] <-0
GM[is.na(GM)] <-0
GOOG[is.na(GOOG)] <-0
GOOGL[is.na(GOOGL)] <-0
GS[is.na(GS)] <-0
HD[is.na(HD)] <-0
HON[is.na(HON)] <-0
IBM[is.na(IBM)] <-0
INTC[is.na(INTC)] <-0
JNJ[is.na(JNJ)] <-0
JPM[is.na(JPM)] <-0
KHC[is.na(KHC)] <-0
KMI[is.na(KMI)] <-0
KO[is.na(KO)] <-0
LLY[is.na(LLY)] <-0
LMT[is.na(LMT)] <-0
LOW[is.na(LOW)] <-0
MA[is.na(MA)] <-0
MCD[is.na(MCD)] <-0
MDLZ[is.na(MDLZ)] <-0
MDT[is.na(MDT)] <-0
MET[is.na(MET)] <-0
MMM[is.na(MMM)] <-0
MO[is.na(MO)] <-0
MRK[is.na(MRK)] <-0
MS[is.na(MS)] <-0
MSFT[is.na(MSFT)] <-0
NEE[is.na(NEE)] <-0
NFLX[is.na(NFLX)] <-0
NKE[is.na(NKE)] <-0
NVDA[is.na(NVDA)] <-0
ORCL[is.na(ORCL)] <-0
OXY[is.na(OXY)] <-0
PEP[is.na(PEP)] <-0
PFE[is.na(PFE)] <-0
PG[is.na(PG)] <-0
PM[is.na(PM)] <-0
PYPL[is.na(PYPL)] <-0
QCOM[is.na(QCOM)] <-0
RTN[is.na(RTN)] <-0
SBUX[is.na(SBUX)] <-0
SLB[is.na(SLB)] <-0
SO[is.na(SO)] <-0
SPG[is.na(SPG)] <-0
T[is.na(T)] <-0
TGT[is.na(TGT)] <-0
TXN[is.na(TXN)] <-0
UNH[is.na(UNH)] <-0
UNP[is.na(UNP)] <-0
UPS[is.na(UPS)] <-0
USB[is.na(USB)] <-0
UTX[is.na(UTX)] <-0
V[is.na(V)] <-0
VZ[is.na(VZ)] <-0
WBA[is.na(WBA)] <-0
WFC[is.na(WFC)] <-0
WMT[is.na(WMT)] <-0
XOM[is.na(XOM)] <-0


## Add a column for ticker symbol - might be useful as we agreggate information
AAPL$Ticker <- NA
ABBV$Ticker <- NA
ABT$Ticker <- NA
ACN$Ticker <- NA
ADBE$Ticker <- NA
AGN$Ticker <- NA
AIG$Ticker <- NA
ALL$Ticker <- NA
AMGN$Ticker <- NA
AMZN$Ticker <- NA
AXP$Ticker <- NA
BA$Ticker <- NA
BAC$Ticker <- NA
BIIB$Ticker <- NA
BK$Ticker <- NA
BKNG$Ticker <- NA
BLK$Ticker <- NA
BMY$Ticker <- NA
C$Ticker <- NA
CAT$Ticker <- NA
CELG$Ticker <- NA
CHTR$Ticker <- NA
CL$Ticker <- NA
CMCSA$Ticker <- NA
COF$Ticker <- NA
COP$Ticker <- NA
COST$Ticker <- NA
CSCO$Ticker <- NA
CVS$Ticker <- NA
CVX$Ticker <- NA
DD$Ticker <- NA
DHR$Ticker <- NA
DIS$Ticker <- NA
DOW$Ticker <- NA
DUK$Ticker <- NA
EMR$Ticker <- NA
EXC$Ticker <- NA
F$Ticker <- NA
FB$Ticker <- NA
FDX$Ticker <- NA
GD$Ticker <- NA
GE$Ticker <- NA
GILD$Ticker <- NA
GM$Ticker <- NA
GOOG$Ticker <- NA
GOOGL$Ticker <- NA
GS$Ticker <- NA
HD$Ticker <- NA
HON$Ticker <- NA
IBM$Ticker <- NA
INTC$Ticker <- NA
JNJ$Ticker <- NA
JPM$Ticker <- NA
KHC$Ticker <- NA
KMI$Ticker <- NA
KO$Ticker <- NA
LLY$Ticker <- NA
LMT$Ticker <- NA
LOW$Ticker <- NA
MA$Ticker <- NA
MCD$Ticker <- NA
MDLZ$Ticker <- NA
MDT$Ticker <- NA
MET$Ticker <- NA
MMM$Ticker <- NA
MO$Ticker <- NA
MRK$Ticker <- NA
MS$Ticker <- NA
MSFT$Ticker <- NA
NEE$Ticker <- NA
NFLX$Ticker <- NA
NKE$Ticker <- NA
NVDA$Ticker <- NA
ORCL$Ticker <- NA
OXY$Ticker <- NA
PEP$Ticker <- NA
```

```
PFE$Ticker <- NA
PG$Ticker <- NA
PM$Ticker <- NA
PYPL$Ticker <- NA
QCOM$Ticker <- NA
RTN$Ticker <- NA
SBUX$Ticker <- NA
SLB$Ticker <- NA
SO$Ticker <- NA
SPG$Ticker <- NA
T$Ticker <- NA
TGT$Ticker <- NA
TXN$Ticker <- NA
UNH$Ticker <- NA
UNP$Ticker <- NA
UPS$Ticker <- NA
USB$Ticker <- NA
UTX$Ticker <- NA
V$Ticker <- NA
VZ$Ticker <- NA
WBA$Ticker <- NA
WFC$Ticker <- NA
WMT$Ticker <- NA
XOM$Ticker <- NA


#Replace "NA"s with ticker symbol of each stock
AAPL$Ticker[is.na(AAPL$Ticker)] <- "AAPL"
ABBV$Ticker[is.na(ABBV$Ticker)] <- "ABBV"
ABT$Ticker[is.na(ABT$Ticker)] <- "ABT"
ACN$Ticker[is.na(ACN$Ticker)] <- "ACN"
ADBE$Ticker[is.na(ADBE$Ticker)] <- "ADBE"
AGN$Ticker[is.na(AGN$Ticker)] <- "AGN"
AIG$Ticker[is.na(AIG$Ticker)] <- "AIG"
ALL$Ticker[is.na(ALL$Ticker)] <- "ALL"
AMGN$Ticker[is.na(AMGN$Ticker)] <- "AMGN"
AMZN$Ticker[is.na(AMZN$Ticker)] <- "AMZN"
AXP$Ticker[is.na(AXP$Ticker)] <- "AXP"
BA$Ticker[is.na(BA$Ticker)] <- "BA"
BAC$Ticker[is.na(BAC$Ticker)] <- "BAC"
BIIB$Ticker[is.na(BIIB$Ticker)] <- "BIIB"
BK$Ticker[is.na(BK$Ticker)] <- "BK"
BKNG$Ticker[is.na(BKNG$Ticker)] <- "BKNG"
BLK$Ticker[is.na(BLK$Ticker)] <- "BLK"
BMY$Ticker[is.na(BMY$Ticker)] <- "BMY"
C$Ticker[is.na(C$Ticker)] <- "C"
CAT$Ticker[is.na(CAT$Ticker)] <- "CAT"
CELG$Ticker[is.na(CELG$Ticker)] <- "CELG"
CHTR$Ticker[is.na(CHTR$Ticker)] <- "CHTR"
CL$Ticker[is.na(CL$Ticker)] <- "CL"
CMCSA$Ticker[is.na(CMCSA$Ticker)] <- "CMCSA"
COF$Ticker[is.na(COF$Ticker)] <- "COF"
COP$Ticker[is.na(COP$Ticker)] <- "COP"
COST$Ticker[is.na(COST$Ticker)] <- "COST"
CSCO$Ticker[is.na(CSCO$Ticker)] <- "CSCO"
CVS$Ticker[is.na(CVS$Ticker)] <- "CVS"
CVX$Ticker[is.na(CVX$Ticker)] <- "CVX"
DD$Ticker[is.na(DD$Ticker)] <- "DD"
DHR$Ticker[is.na(DHR$Ticker)] <- "DHR"
DIS$Ticker[is.na(DIS$Ticker)] <- "DIS"
DOW$Ticker[is.na(DOW$Ticker)] <- "DOW"
DUK$Ticker[is.na(DUK$Ticker)] <- "DUK"
EMR$Ticker[is.na(EMR$Ticker)] <- "EMR"
EXC$Ticker[is.na(EXC$Ticker)] <- "EXC"
F$Ticker[is.na(F$Ticker)] <- "F"
FB$Ticker[is.na(FB$Ticker)] <- "FB"
FDX$Ticker[is.na(FDX$Ticker)] <- "FDX"
GD$Ticker[is.na(GD$Ticker)] <- "GD"
GE$Ticker[is.na(GE$Ticker)] <- "GE"
GILD$Ticker[is.na(GILD$Ticker)] <- "GILD"
GM$Ticker[is.na(GM$Ticker)] <- "GM"
GOOG$Ticker[is.na(GOOG$Ticker)] <- "GOOG"
GOOGL$Ticker[is.na(GOOGL$Ticker)] <- "GOOGL"
GS$Ticker[is.na(GS$Ticker)] <- "GS"
HD$Ticker[is.na(HD$Ticker)] <- "HD"
HON$Ticker[is.na(HON$Ticker)] <- "HON"
IBM$Ticker[is.na(IBM$Ticker)] <- "IBM"
INTC$Ticker[is.na(INTC$Ticker)] <- "INTC"
JNJ$Ticker[is.na(JNJ$Ticker)] <- "JNJ"
JPM$Ticker[is.na(JPM$Ticker)] <- "JPM"
KHC$Ticker[is.na(KHC$Ticker)] <- "KHC"
KMI$Ticker[is.na(KMI$Ticker)] <- "KMI"
KO$Ticker[is.na(KO$Ticker)] <- "KO"
LLY$Ticker[is.na(LLY$Ticker)] <- "LLY"
LMT$Ticker[is.na(LMT$Ticker)] <- "LMT"
LOW$Ticker[is.na(LOW$Ticker)] <- "LOW"
MA$Ticker[is.na(MA$Ticker)] <- "MA"
MCD$Ticker[is.na(MCD$Ticker)] <- "MCD"
MDLZ$Ticker[is.na(MDLZ$Ticker)] <- "MDLZ"
MDT$Ticker[is.na(MDT$Ticker)] <- "MDT"
MET$Ticker[is.na(MET$Ticker)] <- "MET"
MMM$Ticker[is.na(MMM$Ticker)] <- "MMM"
MO$Ticker[is.na(MO$Ticker)] <- "MO"
MRK$Ticker[is.na(MRK$Ticker)] <- "MRK"
MS$Ticker[is.na(MS$Ticker)] <- "MS"
MSFT$Ticker[is.na(MSFT$Ticker)] <- "MSFT"
NEE$Ticker[is.na(NEE$Ticker)] <- "NEE"
NFLX$Ticker[is.na(NFLX$Ticker)] <- "NFLX"
NKE$Ticker[is.na(NKE$Ticker)] <- "NKE"
NVDA$Ticker[is.na(NVDA$Ticker)] <- "NVDA"
ORCL$Ticker[is.na(ORCL$Ticker)] <- "ORCL"
OXY$Ticker[is.na(OXY$Ticker)] <- "OXY"
PEP$Ticker[is.na(PEP$Ticker)] <- "PEP"
PFE$Ticker[is.na(PFE$Ticker)] <- "PFE"
PG$Ticker[is.na(PG$Ticker)] <- "PG"
PM$Ticker[is.na(PM$Ticker)] <- "PM"
PYPL$Ticker[is.na(PYPL$Ticker)] <- "PYPL"
QCOM$Ticker[is.na(QCOM$Ticker)] <- "QCOM"
RTN$Ticker[is.na(RTN$Ticker)] <- "RTN"
SBUX$Ticker[is.na(SBUX$Ticker)] <- "SBUX"
SLB$Ticker[is.na(SLB$Ticker)] <- "SLB"
SO$Ticker[is.na(SO$Ticker)] <- "SO"
SPG$Ticker[is.na(SPG$Ticker)] <- "SPG"
T$Ticker[is.na(T$Ticker)] <- "T"
TGT$Ticker[is.na(TGT$Ticker)] <- "TGT"
TXN$Ticker[is.na(TXN$Ticker)] <- "TXN"
UNH$Ticker[is.na(UNH$Ticker)] <- "UNH"
UNP$Ticker[is.na(UNP$Ticker)] <- "UNP"
UPS$Ticker[is.na(UPS$Ticker)] <- "UPS"
USB$Ticker[is.na(USB$Ticker)] <- "USB"
UTX$Ticker[is.na(UTX$Ticker)] <- "UTX"
V$Ticker[is.na(V$Ticker)] <- "V"
VZ$Ticker[is.na(VZ$Ticker)] <- "VZ"
WBA$Ticker[is.na(WBA$Ticker)] <- "WBA"
WFC$Ticker[is.na(WFC$Ticker)] <- "WFC"
WMT$Ticker[is.na(WMT$Ticker)] <- "WMT"
XOM$Ticker[is.na(XOM$Ticker)] <- "XOM"



# Track historical dividends & stock splits events for each stocl
SiDiAAPL<- get_yahoo_splits_and_dividends('AAPL','2007-01-03', '2019-08-02')
SiDiABBV<- get_yahoo_splits_and_dividends('ABBV','2007-01-03', '2019-08-02')
SiDiABT<- get_yahoo_splits_and_dividends('ABT','2007-01-03', '2019-08-02')
SiDiACN<- get_yahoo_splits_and_dividends('ACN','2007-01-03', '2019-08-02')
SiDiADBE<- get_yahoo_splits_and_dividends('ADBE','2007-01-03', '2019-08-02')
SiDiAGN<- get_yahoo_splits_and_dividends('AGN','2007-01-03', '2019-08-02')
SiDiAIG<- get_yahoo_splits_and_dividends('AIG','2007-01-03', '2019-08-02')
SiDiALL<- get_yahoo_splits_and_dividends('ALL','2007-01-03', '2019-08-02')
SiDiAMGN<- get_yahoo_splits_and_dividends('AMGN','2007-01-03', '2019-08-02')
SiDiAMZN<- get_yahoo_splits_and_dividends('AMZN','2007-01-03', '2019-08-02')
SiDiAXP<- get_yahoo_splits_and_dividends('AXP','2007-01-03', '2019-08-02')
SiDiBA<- get_yahoo_splits_and_dividends('BA','2007-01-03', '2019-08-02')
SiDiBAC<- get_yahoo_splits_and_dividends('BAC','2007-01-03', '2019-08-02')
SiDiBIIB<- get_yahoo_splits_and_dividends('BIIB','2007-01-03', '2019-08-02')
SiDiBK<- get_yahoo_splits_and_dividends('BK','2007-01-03', '2019-08-02')
SiDiBKNG<- get_yahoo_splits_and_dividends('BKNG','2007-01-03', '2019-08-02')
SiDiBLK<- get_yahoo_splits_and_dividends('BLK','2007-01-03', '2019-08-02')
SiDiBMY<- get_yahoo_splits_and_dividends('BMY','2007-01-03', '2019-08-02')
SiDiC<- get_yahoo_splits_and_dividends('C','2007-01-03', '2019-08-02')
SiDiCAT<- get_yahoo_splits_and_dividends('CAT','2007-01-03', '2019-08-02')
SiDiCELG<- get_yahoo_splits_and_dividends('CELG','2007-01-03', '2019-08-02')
SiDiCHTR<- get_yahoo_splits_and_dividends('CHTR','2007-01-03', '2019-08-02')
SiDiCL<- get_yahoo_splits_and_dividends('CL','2007-01-03', '2019-08-02')
SiDiCMCSA<- get_yahoo_splits_and_dividends('CMCSA','2007-01-03', '2019-08-02')
SiDiCOF<- get_yahoo_splits_and_dividends('COF','2007-01-03', '2019-08-02')
SiDiCOP<- get_yahoo_splits_and_dividends('COP','2007-01-03', '2019-08-02')
SiDiCOST<- get_yahoo_splits_and_dividends('COST','2007-01-03', '2019-08-02')
SiDiCSCO<- get_yahoo_splits_and_dividends('CSCO','2007-01-03', '2019-08-02')
SiDiCVS<- get_yahoo_splits_and_dividends('CVS','2007-01-03', '2019-08-02')
SiDiCVX<- get_yahoo_splits_and_dividends('CVX','2007-01-03', '2019-08-02')
SiDiDD<- get_yahoo_splits_and_dividends('DD','2007-01-03', '2019-08-02')
SiDiDHR<- get_yahoo_splits_and_dividends('DHR','2007-01-03', '2019-08-02')
SiDiDIS<- get_yahoo_splits_and_dividends('DIS','2007-01-03', '2019-08-02')
SiDiDOW<- get_yahoo_splits_and_dividends('DOW','2007-01-03', '2019-08-02')
SiDiDUK<- get_yahoo_splits_and_dividends('DUK','2007-01-03', '2019-08-02')
SiDiEMR<- get_yahoo_splits_and_dividends('EMR','2007-01-03', '2019-08-02')
SiDiEXC<- get_yahoo_splits_and_dividends('EXC','2007-01-03', '2019-08-02')
SiDiF<- get_yahoo_splits_and_dividends('F','2007-01-03', '2019-08-02')
SiDiFB<- get_yahoo_splits_and_dividends('FB','2007-01-03', '2019-08-02')
SiDiFDX<- get_yahoo_splits_and_dividends('FDX','2007-01-03', '2019-08-02')
```

```
SiDiGD<- get_yahoo_splits_and_dividends('GD','2007-01-03', '2019-08-02')
SiDiGE<- get_yahoo_splits_and_dividends('GE','2007-01-03', '2019-08-02')
SiDiGILD<- get_yahoo_splits_and_dividends('GILD','2007-01-03', '2019-08-02')
SiDiGM<- get_yahoo_splits_and_dividends('GM','2007-01-03', '2019-08-02')
SiDiGOOG<- get_yahoo_splits_and_dividends('GOOG','2007-01-03', '2019-08-02')
SiDiGOOGL<- get_yahoo_splits_and_dividends('GOOGL','2007-01-03', '2019-08-02')
SiDiGS<- get_yahoo_splits_and_dividends('GS','2007-01-03', '2019-08-02')
SiDiHD<- get_yahoo_splits_and_dividends('HD','2007-01-03', '2019-08-02')
SiDiHON<- get_yahoo_splits_and_dividends('HON','2007-01-03', '2019-08-02')
SiDiIBM<- get_yahoo_splits_and_dividends('IBM','2007-01-03', '2019-08-02')
SiDiINTC<- get_yahoo_splits_and_dividends('INTC','2007-01-03', '2019-08-02')
SiDiJNJ<- get_yahoo_splits_and_dividends('JNJ','2007-01-03', '2019-08-02')
SiDiJPM<- get_yahoo_splits_and_dividends('JPM','2007-01-03', '2019-08-02')
SiDiKHC<- get_yahoo_splits_and_dividends('KHC','2007-01-03', '2019-08-02')
SiDiKMI<- get_yahoo_splits_and_dividends('KMI','2007-01-03', '2019-08-02')
SiDiKO<- get_yahoo_splits_and_dividends('KO','2007-01-03', '2019-08-02')
SiDiLLY<- get_yahoo_splits_and_dividends('LLY','2007-01-03', '2019-08-02')
SiDiLMT<- get_yahoo_splits_and_dividends('LMT','2007-01-03', '2019-08-02')
SiDiLOW<- get_yahoo_splits_and_dividends('LOW','2007-01-03', '2019-08-02')
SiDiMA<- get_yahoo_splits_and_dividends('MA','2007-01-03', '2019-08-02')
SiDiMCD<- get_yahoo_splits_and_dividends('MCD','2007-01-03', '2019-08-02')
SiDiMDLZ<- get_yahoo_splits_and_dividends('MDLZ','2007-01-03', '2019-08-02')
SiDiMDT<- get_yahoo_splits_and_dividends('MDT','2007-01-03', '2019-08-02')
SiDiMET<- get_yahoo_splits_and_dividends('MET','2007-01-03', '2019-08-02')
SiDiMMM<- get_yahoo_splits_and_dividends('MMM','2007-01-03', '2019-08-02')
SiDiMO<- get_yahoo_splits_and_dividends('MO','2007-01-03', '2019-08-02')
SiDiMRK<- get_yahoo_splits_and_dividends('MRK','2007-01-03', '2019-08-02')
SiDiMS<- get_yahoo_splits_and_dividends('MS','2007-01-03', '2019-08-02')
SiDiMSFT<- get_yahoo_splits_and_dividends('MSFT','2007-01-03', '2019-08-02')
SiDiNEE<- get_yahoo_splits_and_dividends('NEE','2007-01-03', '2019-08-02')
SiDiNFLX<- get_yahoo_splits_and_dividends('NFLX','2007-01-03', '2019-08-02')
SiDiNKE<- get_yahoo_splits_and_dividends('NKE','2007-01-03', '2019-08-02')
SiDiNVDA<- get_yahoo_splits_and_dividends('NVDA','2007-01-03', '2019-08-02')
SiDiORCL<- get_yahoo_splits_and_dividends('ORCL','2007-01-03', '2019-08-02')
SiDiOXY<- get_yahoo_splits_and_dividends('OXY','2007-01-03', '2019-08-02')
SiDiPEP<- get_yahoo_splits_and_dividends('PEP','2007-01-03', '2019-08-02')
SiDiPFE<- get_yahoo_splits_and_dividends('PFE','2007-01-03', '2019-08-02')
SiDiPG<- get_yahoo_splits_and_dividends('PG','2007-01-03', '2019-08-02')
SiDiPM<- get_yahoo_splits_and_dividends('PM','2007-01-03', '2019-08-02')
SiDiPYPL<- get_yahoo_splits_and_dividends('PYPL','2007-01-03', '2019-08-02')
SiDiQCOM<- get_yahoo_splits_and_dividends('QCOM','2007-01-03', '2019-08-02')
SiDiRTN<- get_yahoo_splits_and_dividends('RTN','2007-01-03', '2019-08-02')
SiDiSBUX<- get_yahoo_splits_and_dividends('SBUX','2007-01-03', '2019-08-02')
SiDiSLB<- get_yahoo_splits_and_dividends('SLB','2007-01-03', '2019-08-02')
SiDiSO<- get_yahoo_splits_and_dividends('SO','2007-01-03', '2019-08-02')
SiDiSPG<- get_yahoo_splits_and_dividends('SPG','2007-01-03', '2019-08-02')
SiDiT<- get_yahoo_splits_and_dividends('T','2007-01-03', '2019-08-02')
SiDiTGT<- get_yahoo_splits_and_dividends('TGT','2007-01-03', '2019-08-02')
SiDiTXN<- get_yahoo_splits_and_dividends('TXN','2007-01-03', '2019-08-02')
SiDiUNH<- get_yahoo_splits_and_dividends('UNH','2007-01-03', '2019-08-02')
SiDiUNP<- get_yahoo_splits_and_dividends('UNP','2007-01-03', '2019-08-02')
SiDiUPS<- get_yahoo_splits_and_dividends('UPS','2007-01-03', '2019-08-02')
SiDiUSB<- get_yahoo_splits_and_dividends('USB','2007-01-03', '2019-08-02')
SiDiUTX<- get_yahoo_splits_and_dividends('UTX','2007-01-03', '2019-08-02')
SiDiV<- get_yahoo_splits_and_dividends('V','2007-01-03', '2019-08-02')
SiDiVZ<- get_yahoo_splits_and_dividends('VZ','2007-01-03', '2019-08-02')
SiDiWBA<- get_yahoo_splits_and_dividends('WBA','2007-01-03', '2019-08-02')
SiDiWFC<- get_yahoo_splits_and_dividends('WFC','2007-01-03', '2019-08-02')
SiDiWMT<- get_yahoo_splits_and_dividends('WMT','2007-01-03', '2019-08-02')
SiDiXOM<- get_yahoo_splits_and_dividends('XOM','2007-01-03', '2019-08-02')


##USING DATES MERGE dividends & split view with stock historical data
AAPL<- merge(AAPL, SiDiAAPL)
ABBV<- merge(ABBV, SiDiAAPL)
ABT<- merge(ABT, SiDiAAPL)
ACN<- merge(ACN, SiDiAAPL)
ADBE<- merge(ADBE, SiDiAAPL)
AGN<- merge(AGN, SiDiAAPL)
AIG<- merge(AIG, SiDiAAPL)
ALL<- merge(ALL, SiDiAAPL)
AMGN<- merge(AMGN, SiDiAAPL)
AMZN<- merge(AMZN, SiDiAAPL)
AXP<- merge(AXP, SiDiAAPL)
BA<- merge(BA, SiDiAAPL)
BAC<- merge(BAC, SiDiAAPL)
BIIB<- merge(BIIB, SiDiAAPL)
BK<- merge(BK, SiDiAAPL)
BKNG<- merge(BKNG, SiDiAAPL)
BLK<- merge(BLK, SiDiAAPL)
BMY<- merge(BMY, SiDiAAPL)
C<- merge(C, SiDiAAPL)
CAT<- merge(CAT, SiDiAAPL)
CELG<- merge(CELG, SiDiAAPL)
CHTR<- merge(CHTR, SiDiAAPL)
CL<- merge(CL, SiDiAAPL)
CMCSA<- merge(CMCSA, SiDiAAPL)
COF<- merge(COF, SiDiAAPL)
COP<- merge(COP, SiDiAAPL)
COST<- merge(COST, SiDiAAPL)
CSCO<- merge(CSCO, SiDiAAPL)
CVS<- merge(CVS, SiDiAAPL)
CVX<- merge(CVX, SiDiAAPL)
DD<- merge(DD, SiDiAAPL)
DHR<- merge(DHR, SiDiAAPL)
DIS<- merge(DIS, SiDiAAPL)
DOW<- merge(DOW, SiDiAAPL)
DUK<- merge(DUK, SiDiAAPL)
EMR<- merge(EMR, SiDiAAPL)
EXC<- merge(EXC, SiDiAAPL)
F<- merge(F, SiDiAAPL)
FB<- merge(FB, SiDiAAPL)
FDX<- merge(FDX, SiDiAAPL)
GD<- merge(GD, SiDiAAPL)
GE<- merge(GE, SiDiAAPL)
GILD<- merge(GILD, SiDiAAPL)
GM<- merge(GM, SiDiAAPL)
GOOG<- merge(GOOG, SiDiAAPL)
GOOGL<- merge(GOOGL, SiDiAAPL)
GS<- merge(GS, SiDiAAPL)
HD<- merge(HD, SiDiAAPL)
HON<- merge(HON, SiDiAAPL)
IBM<- merge(IBM, SiDiAAPL)
INTC<- merge(INTC, SiDiAAPL)
JNJ<- merge(JNJ, SiDiAAPL)
JPM<- merge(JPM, SiDiAAPL)
KHC<- merge(KHC, SiDiAAPL)
KMI<- merge(KMI, SiDiAAPL)
KO<- merge(KO, SiDiAAPL)
LLY<- merge(LLY, SiDiAAPL)
LMT<- merge(LMT, SiDiAAPL)
LOW<- merge(LOW, SiDiAAPL)
MA<- merge(MA, SiDiAAPL)
MCD<- merge(MCD, SiDiAAPL)
MDLZ<- merge(MDLZ, SiDiAAPL)
MDT<- merge(MDT, SiDiAAPL)
MET<- merge(MET, SiDiAAPL)
MMM<- merge(MMM, SiDiAAPL)
MO<- merge(MO, SiDiAAPL)
MRK<- merge(MRK, SiDiAAPL)
MS<- merge(MS, SiDiAAPL)
MSFT<- merge(MSFT, SiDiAAPL)
NEE<- merge(NEE, SiDiAAPL)
NFLX<- merge(NFLX, SiDiAAPL)
NKE<- merge(NKE, SiDiAAPL)
NVDA<- merge(NVDA, SiDiAAPL)
ORCL<- merge(ORCL, SiDiAAPL)
OXY<- merge(OXY, SiDiAAPL)
PEP<- merge(PEP, SiDiAAPL)
PFE<- merge(PFE, SiDiAAPL)
PG<- merge(PG, SiDiAAPL)
PM<- merge(PM, SiDiAAPL)
PYPL<- merge(PYPL, SiDiAAPL)
QCOM<- merge(QCOM, SiDiAAPL)
RTN<- merge(RTN, SiDiAAPL)
SBUX<- merge(SBUX, SiDiAAPL)
SLB<- merge(SLB, SiDiAAPL)
SO<- merge(SO, SiDiAAPL)
SPG<- merge(SPG, SiDiAAPL)
T<- merge(T, SiDiAAPL)
TGT<- merge(TGT, SiDiAAPL)
TXN<- merge(TXN, SiDiAAPL)
UNH<- merge(UNH, SiDiAAPL)
UNP<- merge(UNP, SiDiAAPL)
UPS<- merge(UPS, SiDiAAPL)
USB<- merge(USB, SiDiAAPL)
UTX<- merge(UTX, SiDiAAPL)
V<- merge(V, SiDiAAPL)
VZ<- merge(VZ, SiDiAAPL)
WBA<- merge(WBA, SiDiAAPL)
WFC<- merge(WFC, SiDiAAPL)
WMT<- merge(WMT, SiDiAAPL)
XOM<- merge(XOM, SiDiAAPL)



#REMOVE NAs FROM TICKER SYMBOLS (THIS WAS LOST IN THE MERGE)
AAPL$Ticker[is.na(AAPL$Ticker)] <- "AAPL"
ABBV$Ticker[is.na(ABBV$Ticker)] <- "ABBV"
ABT$Ticker[is.na(ABT$Ticker)] <- "ABT"
ACN$Ticker[is.na(ACN$Ticker)] <- "ACN"
```

```
ADBE$Ticker[is.na(ADBE$Ticker)] <- "ADBE"
AGN$Ticker[is.na(AGN$Ticker)] <- "AGN"
AIG$Ticker[is.na(AIG$Ticker)] <- "AIG"
ALL$Ticker[is.na(ALL$Ticker)] <- "ALL"
AMGN$Ticker[is.na(AMGN$Ticker)] <- "AMGN"
AMZN$Ticker[is.na(AMZN$Ticker)] <- "AMZN"
AXP$Ticker[is.na(AXP$Ticker)] <- "AXP"
BA$Ticker[is.na(BA$Ticker)] <- "BA"
BAC$Ticker[is.na(BAC$Ticker)] <- "BAC"
BIIB$Ticker[is.na(BIIB$Ticker)] <- "BIIB"
BK$Ticker[is.na(BK$Ticker)] <- "BK"
BKNG$Ticker[is.na(BKNG$Ticker)] <- "BKNG"
BLK$Ticker[is.na(BLK$Ticker)] <- "BLK"
BMY$Ticker[is.na(BMY$Ticker)] <- "BMY"
C$Ticker[is.na(C$Ticker)] <- "C"
CAT$Ticker[is.na(CAT$Ticker)] <- "CAT"
CELG$Ticker[is.na(CELG$Ticker)] <- "CELG"
CHTR$Ticker[is.na(CHTR$Ticker)] <- "CHTR"
CL$Ticker[is.na(CL$Ticker)] <- "CL"
CMCSA$Ticker[is.na(CMCSA$Ticker)] <- "CMCSA"
COF$Ticker[is.na(COF$Ticker)] <- "COF"
COP$Ticker[is.na(COP$Ticker)] <- "COP"
COST$Ticker[is.na(COST$Ticker)] <- "COST"
CSCO$Ticker[is.na(CSCO$Ticker)] <- "CSCO"
CVS$Ticker[is.na(CVS$Ticker)] <- "CVS"
CVX$Ticker[is.na(CVX$Ticker)] <- "CVX"
DD$Ticker[is.na(DD$Ticker)] <- "DD"
DHR$Ticker[is.na(DHR$Ticker)] <- "DHR"
DIS$Ticker[is.na(DIS$Ticker)] <- "DIS"
DOW$Ticker[is.na(DOW$Ticker)] <- "DOW"
DUK$Ticker[is.na(DUK$Ticker)] <- "DUK"
EMR$Ticker[is.na(EMR$Ticker)] <- "EMR"
EXC$Ticker[is.na(EXC$Ticker)] <- "EXC"
F$Ticker[is.na(F$Ticker)] <- "F"
FB$Ticker[is.na(FB$Ticker)] <- "FB"
FDX$Ticker[is.na(FDX$Ticker)] <- "FDX"
GD$Ticker[is.na(GD$Ticker)] <- "GD"
GE$Ticker[is.na(GE$Ticker)] <- "GE"
GILD$Ticker[is.na(GILD$Ticker)] <- "GILD"
GM$Ticker[is.na(GM$Ticker)] <- "GM"
GOOG$Ticker[is.na(GOOG$Ticker)] <- "GOOG"
GOOGL$Ticker[is.na(GOOGL$Ticker)] <- "GOOGL"
GS$Ticker[is.na(GS$Ticker)] <- "GS"
HD$Ticker[is.na(HD$Ticker)] <- "HD"
HON$Ticker[is.na(HON$Ticker)] <- "HON"
IBM$Ticker[is.na(IBM$Ticker)] <- "IBM"
INTC$Ticker[is.na(INTC$Ticker)] <- "INTC"
JNJ$Ticker[is.na(JNJ$Ticker)] <- "JNJ"
JPM$Ticker[is.na(JPM$Ticker)] <- "JPM"
KHC$Ticker[is.na(KHC$Ticker)] <- "KHC"
KMI$Ticker[is.na(KMI$Ticker)] <- "KMI"
KO$Ticker[is.na(KO$Ticker)] <- "KO"
LLY$Ticker[is.na(LLY$Ticker)] <- "LLY"
LMT$Ticker[is.na(LMT$Ticker)] <- "LMT"
LOW$Ticker[is.na(LOW$Ticker)] <- "LOW"
MA$Ticker[is.na(MA$Ticker)] <- "MA"
MCD$Ticker[is.na(MCD$Ticker)] <- "MCD"
MDLZ$Ticker[is.na(MDLZ$Ticker)] <- "MDLZ"
MDT$Ticker[is.na(MDT$Ticker)] <- "MDT"
MET$Ticker[is.na(MET$Ticker)] <- "MET"
MMM$Ticker[is.na(MMM$Ticker)] <- "MMM"
MO$Ticker[is.na(MO$Ticker)] <- "MO"
MRK$Ticker[is.na(MRK$Ticker)] <- "MRK"
MS$Ticker[is.na(MS$Ticker)] <- "MS"
MSFT$Ticker[is.na(MSFT$Ticker)] <- "MSFT"
NEE$Ticker[is.na(NEE$Ticker)] <- "NEE"
NFLX$Ticker[is.na(NFLX$Ticker)] <- "NFLX"
NKE$Ticker[is.na(NKE$Ticker)] <- "NKE"
NVDA$Ticker[is.na(NVDA$Ticker)] <- "NVDA"
ORCL$Ticker[is.na(ORCL$Ticker)] <- "ORCL"
OXY$Ticker[is.na(OXY$Ticker)] <- "OXY"
PEP$Ticker[is.na(PEP$Ticker)] <- "PEP"
PFE$Ticker[is.na(PFE$Ticker)] <- "PFE"
PG$Ticker[is.na(PG$Ticker)] <- "PG"
PM$Ticker[is.na(PM$Ticker)] <- "PM"
PYPL$Ticker[is.na(PYPL$Ticker)] <- "PYPL"
QCOM$Ticker[is.na(QCOM$Ticker)] <- "QCOM"
RTN$Ticker[is.na(RTN$Ticker)] <- "RTN"
SBUX$Ticker[is.na(SBUX$Ticker)] <- "SBUX"
SLB$Ticker[is.na(SLB$Ticker)] <- "SLB"
SO$Ticker[is.na(SO$Ticker)] <- "SO"
SPG$Ticker[is.na(SPG$Ticker)] <- "SPG"
T$Ticker[is.na(T$Ticker)] <- "T"
TGT$Ticker[is.na(TGT$Ticker)] <- "TGT"
TXN$Ticker[is.na(TXN$Ticker)] <- "TXN"
UNH$Ticker[is.na(UNH$Ticker)] <- "UNH"
UNP$Ticker[is.na(UNP$Ticker)] <- "UNP"
UPS$Ticker[is.na(UPS$Ticker)] <- "UPS"
USB$Ticker[is.na(USB$Ticker)] <- "USB"
UTX$Ticker[is.na(UTX$Ticker)] <- "UTX"
V$Ticker[is.na(V$Ticker)] <- "V"
VZ$Ticker[is.na(VZ$Ticker)] <- "VZ"
WBA$Ticker[is.na(WBA$Ticker)] <- "WBA"
WFC$Ticker[is.na(WFC$Ticker)] <- "WFC"
WMT$Ticker[is.na(WMT$Ticker)] <- "WMT"
XOM$Ticker[is.na(XOM$Ticker)] <- "XOM"


#REMOVE NAs FROM DIVIDEND VALUE

AAPL$value[is.na(AAPL$value)] <- 0
ABBV$value[is.na(ABBV$value)] <-  0
ABT$value[is.na(ABT$value)] <-  0
ACN$value[is.na(ACN$value)] <-  0
ADBE$value[is.na(ADBE$value)] <-  0
AGN$value[is.na(AGN$value)] <-  0
AIG$value[is.na(AIG$value)] <-  0
ALL$value[is.na(ALL$value)] <-  0
AMGN$value[is.na(AMGN$value)] <-  0
AMZN$value[is.na(AMZN$value)] <-  0
AXP$value[is.na(AXP$value)] <-  0
BA$value[is.na(BA$value)] <-  0
BAC$value[is.na(BAC$value)] <-  0
BIIB$value[is.na(BIIB$value)] <-  0
BK$value[is.na(BK$value)] <-  0
BKNG$value[is.na(BKNG$value)] <-  0
BLK$value[is.na(BLK$value)] <-  0
BMY$value[is.na(BMY$value)] <-  0
C$value[is.na(C$value)] <-  0
CAT$value[is.na(CAT$value)] <-  0
CELG$value[is.na(CELG$value)] <-  0
CHTR$value[is.na(CHTR$value)] <-  0
CL$value[is.na(CL$value)] <-  0
CMCSA$value[is.na(CMCSA$value)] <-  0
COF$value[is.na(COF$value)] <-  0
COP$value[is.na(COP$value)] <-  0
COST$value[is.na(COST$value)] <-  0
CSCO$value[is.na(CSCO$value)] <-  0
CVS$value[is.na(CVS$value)] <-  0
CVX$value[is.na(CVX$value)] <-  0
DD$value[is.na(DD$value)] <-  0
DHR$value[is.na(DHR$value)] <-  0
DIS$value[is.na(DIS$value)] <-  0
DOW$value[is.na(DOW$value)] <-  0
DUK$value[is.na(DUK$value)] <-  0
EMR$value[is.na(EMR$value)] <-  0
EXC$value[is.na(EXC$value)] <-  0
F$value[is.na(F$value)] <-  0
FB$value[is.na(FB$value)] <-  0
FDX$value[is.na(FDX$value)] <-  0
GD$value[is.na(GD$value)] <-  0
GE$value[is.na(GE$value)] <-  0
GILD$value[is.na(GILD$value)] <-  0
GM$value[is.na(GM$value)] <-  0
GOOG$value[is.na(GOOG$value)] <-  0
GOOGL$value[is.na(GOOGL$value)] <-  0
GS$value[is.na(GS$value)] <-  0
HD$value[is.na(HD$value)] <-  0
HON$value[is.na(HON$value)] <-  0
IBM$value[is.na(IBM$value)] <-  0
INTC$value[is.na(INTC$value)] <-  0
JNJ$value[is.na(JNJ$value)] <-  0
JPM$value[is.na(JPM$value)] <-  0
KHC$value[is.na(KHC$value)] <-  0
KMI$value[is.na(KMI$value)] <-  0
KO$value[is.na(KO$value)] <-  0
LLY$value[is.na(LLY$value)] <-  0
LMT$value[is.na(LMT$value)] <-  0
LOW$value[is.na(LOW$value)] <-  0
MA$value[is.na(MA$value)] <-  0
MCD$value[is.na(MCD$value)] <-  0
MDLZ$value[is.na(MDLZ$value)] <-  0
MDT$value[is.na(MDT$value)] <-  0
MET$value[is.na(MET$value)] <-  0
MMM$value[is.na(MMM$value)] <-  0
MO$value[is.na(MO$value)] <-  0
MRK$value[is.na(MRK$value)] <-  0
MS$value[is.na(MS$value)] <-  0
MSFT$value[is.na(MSFT$value)] <-  0
NEE$value[is.na(NEE$value)] <-  0
NFLX$value[is.na(NFLX$value)] <-  0
```

```
NKE$value[is.na(NKE$value)] <-  0
NVDA$value[is.na(NVDA$value)] <-  0
ORCL$value[is.na(ORCL$value)] <-  0
OXY$value[is.na(OXY$value)] <-  0
PEP$value[is.na(PEP$value)] <-  0
PFE$value[is.na(PFE$value)] <-  0
PG$value[is.na(PG$value)] <-  0
PM$value[is.na(PM$value)] <-  0
PYPL$value[is.na(PYPL$value)] <-  0
QCOM$value[is.na(QCOM$value)] <-  0
RTN$value[is.na(RTN$value)] <-  0
SBUX$value[is.na(SBUX$value)] <-  0
SLB$value[is.na(SLB$value)] <-  0
SO$value[is.na(SO$value)] <-  0
SPG$value[is.na(SPG$value)] <-  0
T$value[is.na(T$value)] <-  0
TGT$value[is.na(TGT$value)] <-  0
TXN$value[is.na(TXN$value)] <-  0
UNH$value[is.na(UNH$value)] <-  0
UNP$value[is.na(UNP$value)] <-  0
UPS$value[is.na(UPS$value)] <-  0
USB$value[is.na(USB$value)] <-  0
UTX$value[is.na(UTX$value)] <-  0
V$value[is.na(V$value)] <-  0
VZ$value[is.na(VZ$value)] <-  0
WBA$value[is.na(WBA$value)] <-  0
WFC$value[is.na(WFC$value)] <-  0
WMT$value[is.na(WMT$value)] <-  0
XOM$value[is.na(XOM$value)] <-  0


### Data Exploration

#Summary view of all stocks
summary (AAPL)
summary (ABBV)
summary (ABT)
summary (ACN)
summary (ADBE)
summary (AGN)
summary (AIG)
summary (ALL)
summary (AMGN)
summary (AMZN)
summary (AXP)
summary (BA)
summary (BAC)
summary (BIIB)
summary (BK)
summary (BKNG)
summary (BLK)
summary (BMY)
summary (C)
summary (CAT)
summary (CELG)
summary (CHTR)
summary (CL)
summary (CMCSA)
summary (COF)
summary (COP)
summary (COST)
summary (CSCO)
summary (CVS)
summary (CVX)
summary (DD)
summary (DHR)
summary (DIS)
summary (DOW)
summary (DUK)
summary (EMR)
summary (EXC)
summary (F)
summary (FB)
summary (FDX)
summary (GD)
summary (GE)
summary (GILD)
summary (GM)
summary (GOOG)
summary (GOOGL)
summary (GS)
summary (HD)
summary (HON)
summary (IBM)
summary (INTC)
summary (JNJ)
summary (JPM)
summary (KHC)
summary (KMI)
summary (KO)
summary (LLY)
summary (LMT)
summary (LOW)
summary (MA)
summary (MCD)
summary (MDLZ)
summary (MDT)
summary (MET)
summary (MMM)
summary (MO)
summary (MRK)
summary (MS)
summary (MSFT)
summary (NEE)
summary (NFLX)
summary (NKE)
summary (NVDA)
summary (ORCL)
summary (OXY)
summary (PEP)
summary (PFE)
summary (PG)
summary (PM)
summary (PYPL)
summary (QCOM)
summary (RTN)
summary (SBUX)
summary (SLB)
summary (SO)
summary (SPG)
summary (T)
summary (TGT)
summary (TXN)
summary (UNH)
summary (UNP)
summary (UPS)
summary (USB)
summary (UTX)
summary (V)
summary (VZ)
summary (WBA)
summary (WFC)
summary (WMT)
summary (XOM)


# Summary of aggregated SnP100 view
summary(SnP100)


####xts is being classified as factor - so will transfrom all the symbols xts into dataframes
#### than will transform all columns (except date & ticker) into numeric

#TRANSFORM INTO DATA FRAMES
AAPLfull<- data.frame(AAPL)
ABBVfull<- data.frame(ABBV)
ABTfull<- data.frame(ABT)
ACNfull<- data.frame(ACN)
ADBEfull<- data.frame(ADBE)
AGNfull<- data.frame(AGN)
AIGfull<- data.frame(AIG)
ALLfull<- data.frame(ALL)
AMGNfull<- data.frame(AMGN)
AMZNfull<- data.frame(AMZN)
AXPfull<- data.frame(AXP)
BAfull<- data.frame(BA)
BACfull<- data.frame(BAC)
BIIBfull<- data.frame(BIIB)
BKfull<- data.frame(BK)
BKNGfull<- data.frame(BKNG)
BLKfull<- data.frame(BLK)
BMYfull<- data.frame(BMY)
Cfull<- data.frame(C)
CATfull<- data.frame(CAT)
CELGfull<- data.frame(CELG)
CHTRfull<- data.frame(CHTR)
CLfull<- data.frame(CL)
CMCSAfull<- data.frame(CMCSA)
COFfull<- data.frame(COF)
COPfull<- data.frame(COP)
COSTfull<- data.frame(COST)
```

```
CSCOfull<- data.frame(CSCO)
CVSfull<- data.frame(CVS)
CVXfull<- data.frame(CVX)
DDfull<- data.frame(DD)
DHRfull<- data.frame(DHR)
DISfull<- data.frame(DIS)
DOWfull<- data.frame(DOW)
DUKfull<- data.frame(DUK)
EMRfull<- data.frame(EMR)
EXCfull<- data.frame(EXC)
Ffull<- data.frame(F)
FBfull<- data.frame(FB)
FDXfull<- data.frame(FDX)
GDfull<- data.frame(GD)
GEfull<- data.frame(GE)
GILDfull<- data.frame(GILD)
GMfull<- data.frame(GM)
GOOGfull<- data.frame(GOOG)
GOOGLfull<- data.frame(GOOGL)
GSfull<- data.frame(GS)
HDfull<- data.frame(HD)
HONfull<- data.frame(HON)
IBMfull<- data.frame(IBM)
INTCfull<- data.frame(INTC)
JNJfull<- data.frame(JNJ)
JPMfull<- data.frame(JPM)
KHCfull<- data.frame(KHC)
KMIfull<- data.frame(KMI)
KOfull<- data.frame(KO)
LLYfull<- data.frame(LLY)
LMTfull<- data.frame(LMT)
LOWfull<- data.frame(LOW)
MAfull<- data.frame(MA)
MCDfull<- data.frame(MCD)
MDLZfull<- data.frame(MDLZ)
MDTfull<- data.frame(MDT)
METfull<- data.frame(MET)
MMMfull<- data.frame(MMM)
MOfull<- data.frame(MO)
MRKfull<- data.frame(MRK)
MSfull<- data.frame(MS)
MSFTfull<- data.frame(MSFT)
NEEfull<- data.frame(NEE)
NFLXfull<- data.frame(NFLX)
NKEfull<- data.frame(NKE)
NVDAfull<- data.frame(NVDA)
ORCLfull<- data.frame(ORCL)
OXYfull<- data.frame(OXY)
PEPfull<- data.frame(PEP)
PFEfull<- data.frame(PFE)
PGfull<- data.frame(PG)
PMfull<- data.frame(PM)
PYPLfull<- data.frame(PYPL)
QCOMfull<- data.frame(QCOM)
RTNfull<- data.frame(RTN)
SBUXfull<- data.frame(SBUX)
SLBfull<- data.frame(SLB)
SOfull<- data.frame(SO)
SPGfull<- data.frame(SPG)
Tfull<- data.frame(T)
TGTfull<- data.frame(TGT)
TXNfull<- data.frame(TXN)
UNHfull<- data.frame(UNH)
UNPfull<- data.frame(UNP)
UPSfull<- data.frame(UPS)
USBfull<- data.frame(USB)
UTXfull<- data.frame(UTX)
Vfull<- data.frame(V)
VZfull<- data.frame(VZ)
WBAfull<- data.frame(WBA)
WFCfull<- data.frame(WFC)
WMTfull<- data.frame(WMT)
XOMfull<- data.frame(XOM)


#transfrom Volatility classification from factor into numeric
AAPLfull$VOLA<- as.numeric(as.character(AAPLfull$VOLA))
ABBVfull$VOLA<- as.numeric(as.character(ABBVfull$VOLA))
ABTfull$VOLA<- as.numeric(as.character(ABTfull$VOLA))
ACNfull$VOLA<- as.numeric(as.character(ACNfull$VOLA))
ADBEfull$VOLA<- as.numeric(as.character(ADBEfull$VOLA))
AGNfull$VOLA<- as.numeric(as.character(AGNfull$VOLA))
AIGfull$VOLA<- as.numeric(as.character(AIGfull$VOLA))
ALLfull$VOLA<- as.numeric(as.character(ALLfull$VOLA))
AMGNfull$VOLA<- as.numeric(as.character(AMGNfull$VOLA))
AMZNfull$VOLA<- as.numeric(as.character(AMZNfull$VOLA))
AXPfull$VOLA<- as.numeric(as.character(AXPfull$VOLA))
BAfull$VOLA<- as.numeric(as.character(BAfull$VOLA))
BACfull$VOLA<- as.numeric(as.character(BACfull$VOLA))
BIIBfull$VOLA<- as.numeric(as.character(BIIBfull$VOLA))
BKfull$VOLA<- as.numeric(as.character(BKfull$VOLA))
BKNGfull$VOLA<- as.numeric(as.character(BKNGfull$VOLA))
BLKfull$VOLA<- as.numeric(as.character(BLKfull$VOLA))
BMYfull$VOLA<- as.numeric(as.character(BMYfull$VOLA))
Cfull$VOLA<- as.numeric(as.character(Cfull$VOLA))
CATfull$VOLA<- as.numeric(as.character(CATfull$VOLA))
CELGfull$VOLA<- as.numeric(as.character(CELGfull$VOLA))
CHTRfull$VOLA<- as.numeric(as.character(CHTRfull$VOLA))
CLfull$VOLA<- as.numeric(as.character(CLfull$VOLA))
CMCSAfull$VOLA<- as.numeric(as.character(CMCSAfull$VOLA))
COFfull$VOLA<- as.numeric(as.character(COFfull$VOLA))
COPfull$VOLA<- as.numeric(as.character(COPfull$VOLA))
COSTfull$VOLA<- as.numeric(as.character(COSTfull$VOLA))
CSCOfull$VOLA<- as.numeric(as.character(CSCOfull$VOLA))
CVSfull$VOLA<- as.numeric(as.character(CVSfull$VOLA))
CVXfull$VOLA<- as.numeric(as.character(CVXfull$VOLA))
DDfull$VOLA<- as.numeric(as.character(DDfull$VOLA))
DHRfull$VOLA<- as.numeric(as.character(DHRfull$VOLA))
DISfull$VOLA<- as.numeric(as.character(DISfull$VOLA))
DOWfull$VOLA<- as.numeric(as.character(DOWfull$VOLA))
DUKfull$VOLA<- as.numeric(as.character(DUKfull$VOLA))
EMRfull$VOLA<- as.numeric(as.character(EMRfull$VOLA))
EXCfull$VOLA<- as.numeric(as.character(EXCfull$VOLA))
Ffull$VOLA<- as.numeric(as.character(Ffull$VOLA))
FBfull$VOLA<- as.numeric(as.character(FBfull$VOLA))
FDXfull$VOLA<- as.numeric(as.character(FDXfull$VOLA))
GDfull$VOLA<- as.numeric(as.character(GDfull$VOLA))
GEfull$VOLA<- as.numeric(as.character(GEfull$VOLA))
GILDfull$VOLA<- as.numeric(as.character(GILDfull$VOLA))
GMfull$VOLA<- as.numeric(as.character(GMfull$VOLA))
GOOGfull$VOLA<- as.numeric(as.character(GOOGfull$VOLA))
GOOGLfull$VOLA<- as.numeric(as.character(GOOGLfull$VOLA))
GSfull$VOLA<- as.numeric(as.character(GSfull$VOLA))
HDfull$VOLA<- as.numeric(as.character(HDfull$VOLA))
HONfull$VOLA<- as.numeric(as.character(HONfull$VOLA))
IBMfull$VOLA<- as.numeric(as.character(IBMfull$VOLA))
INTCfull$VOLA<- as.numeric(as.character(INTCfull$VOLA))
JNJfull$VOLA<- as.numeric(as.character(JNJfull$VOLA))
JPMfull$VOLA<- as.numeric(as.character(JPMfull$VOLA))
KHCfull$VOLA<- as.numeric(as.character(KHCfull$VOLA))
KMIfull$VOLA<- as.numeric(as.character(KMIfull$VOLA))
KOfull$VOLA<- as.numeric(as.character(KOfull$VOLA))
LLYfull$VOLA<- as.numeric(as.character(LLYfull$VOLA))
LMTfull$VOLA<- as.numeric(as.character(LMTfull$VOLA))
LOWfull$VOLA<- as.numeric(as.character(LOWfull$VOLA))
MAfull$VOLA<- as.numeric(as.character(MAfull$VOLA))
MCDfull$VOLA<- as.numeric(as.character(MCDfull$VOLA))
MDLZfull$VOLA<- as.numeric(as.character(MDLZfull$VOLA))
MDTfull$VOLA<- as.numeric(as.character(MDTfull$VOLA))
METfull$VOLA<- as.numeric(as.character(METfull$VOLA))
MMMfull$VOLA<- as.numeric(as.character(MMMfull$VOLA))
MOfull$VOLA<- as.numeric(as.character(MOfull$VOLA))
MRKfull$VOLA<- as.numeric(as.character(MRKfull$VOLA))
MSfull$VOLA<- as.numeric(as.character(MSfull$VOLA))
MSFTfull$VOLA<- as.numeric(as.character(MSFTfull$VOLA))
NEEfull$VOLA<- as.numeric(as.character(NEEfull$VOLA))
NFLXfull$VOLA<- as.numeric(as.character(NFLXfull$VOLA))
NKEfull$VOLA<- as.numeric(as.character(NKEfull$VOLA))
NVDAfull$VOLA<- as.numeric(as.character(NVDAfull$VOLA))
ORCLfull$VOLA<- as.numeric(as.character(ORCLfull$VOLA))
OXYfull$VOLA<- as.numeric(as.character(OXYfull$VOLA))
PEPfull$VOLA<- as.numeric(as.character(PEPfull$VOLA))
PFEfull$VOLA<- as.numeric(as.character(PFEfull$VOLA))
PGfull$VOLA<- as.numeric(as.character(PGfull$VOLA))
PMfull$VOLA<- as.numeric(as.character(PMfull$VOLA))
PYPLfull$VOLA<- as.numeric(as.character(PYPLfull$VOLA))
QCOMfull$VOLA<- as.numeric(as.character(QCOMfull$VOLA))
RTNfull$VOLA<- as.numeric(as.character(RTNfull$VOLA))
SBUXfull$VOLA<- as.numeric(as.character(SBUXfull$VOLA))
SLBfull$VOLA<- as.numeric(as.character(SLBfull$VOLA))
SOfull$VOLA<- as.numeric(as.character(SOfull$VOLA))
SPGfull$VOLA<- as.numeric(as.character(SPGfull$VOLA))
Tfull$VOLA<- as.numeric(as.character(Tfull$VOLA))
TGTfull$VOLA<- as.numeric(as.character(TGTfull$VOLA))
TXNfull$VOLA<- as.numeric(as.character(TXNfull$VOLA))
UNHfull$VOLA<- as.numeric(as.character(UNHfull$VOLA))
UNPfull$VOLA<- as.numeric(as.character(UNPfull$VOLA))
UPSfull$VOLA<- as.numeric(as.character(UPSfull$VOLA))
USBfull$VOLA<- as.numeric(as.character(USBfull$VOLA))
UTXfull$VOLA<- as.numeric(as.character(UTXfull$VOLA))
Vfull$VOLA<- as.numeric(as.character(Vfull$VOLA))
```

```
VZfull$VOLA<- as.numeric(as.character(VZfull$VOLA))
WBAfull$VOLA<- as.numeric(as.character(WBAfull$VOLA))
WFCfull$VOLA<- as.numeric(as.character(WFCfull$VOLA))
WMTfull$VOLA<- as.numeric(as.character(WMTfull$VOLA))
XOMfull$VOLA<- as.numeric(as.character(XOMfull$VOLA))


#Transfrom Open column from factor to numeric
AAPLfull$AAPL.Open<- as.numeric(as.character(AAPLfull$AAPL.Open))
ABBVfull$ABBV.Open<- as.numeric(as.character(ABBVfull$ABBV.Open))
ABTfull$ABT.Open<- as.numeric(as.character(ABTfull$ABT.Open))
ACNfull$ACN.Open<- as.numeric(as.character(ACNfull$ACN.Open))
ADBEfull$ADBE.Open<- as.numeric(as.character(ADBEfull$ADBE.Open))
AGNfull$AGN.Open<- as.numeric(as.character(AGNfull$AGN.Open))
AIGfull$AIG.Open<- as.numeric(as.character(AIGfull$AIG.Open))
ALLfull$ALL.Open<- as.numeric(as.character(ALLfull$ALL.Open))
AMGNfull$AMGN.Open<- as.numeric(as.character(AMGNfull$AMGN.Open))
AMZNfull$AMZN.Open<- as.numeric(as.character(AMZNfull$AMZN.Open))
AXPfull$AXP.Open<- as.numeric(as.character(AXPfull$AXP.Open))
BAfull$BA.Open<- as.numeric(as.character(BAfull$BA.Open))
BACfull$BAC.Open<- as.numeric(as.character(BACfull$BAC.Open))
BIIBfull$BIIB.Open<- as.numeric(as.character(BIIBfull$BIIB.Open))
BKfull$BK.Open<- as.numeric(as.character(BKfull$BK.Open))
BKNGfull$BKNG.Open<- as.numeric(as.character(BKNGfull$BKNG.Open))
BLKfull$BLK.Open<- as.numeric(as.character(BLKfull$BLK.Open))
BMYfull$BMY.Open<- as.numeric(as.character(BMYfull$BMY.Open))
Cfull$C.Open<- as.numeric(as.character(Cfull$C.Open))
CATfull$CAT.Open<- as.numeric(as.character(CATfull$CAT.Open))
CELGfull$CELG.Open<- as.numeric(as.character(CELGfull$CELG.Open))
CHTRfull$CHTR.Open<- as.numeric(as.character(CHTRfull$CHTR.Open))
CLfull$CL.Open<- as.numeric(as.character(CLfull$CL.Open))
CMCSAfull$CMCSA.Open<- as.numeric(as.character(CMCSAfull$CMCSA.Open))
COFfull$COF.Open<- as.numeric(as.character(COFfull$COF.Open))
COPfull$COP.Open<- as.numeric(as.character(COPfull$COP.Open))
COSTfull$COST.Open<- as.numeric(as.character(COSTfull$COST.Open))
CSCOfull$CSCO.Open<- as.numeric(as.character(CSCOfull$CSCO.Open))
CVSfull$CVS.Open<- as.numeric(as.character(CVSfull$CVS.Open))
CVXfull$CVX.Open<- as.numeric(as.character(CVXfull$CVX.Open))
DDfull$DD.Open<- as.numeric(as.character(DDfull$DD.Open))
DHRfull$DHR.Open<- as.numeric(as.character(DHRfull$DHR.Open))
DISfull$DIS.Open<- as.numeric(as.character(DISfull$DIS.Open))
DOWfull$DOW.Open<- as.numeric(as.character(DOWfull$DOW.Open))
DUKfull$DUK.Open<- as.numeric(as.character(DUKfull$DUK.Open))
EMRfull$EMR.Open<- as.numeric(as.character(EMRfull$EMR.Open))
EXCfull$EXC.Open<- as.numeric(as.character(EXCfull$EXC.Open))
Ffull$F.Open<- as.numeric(as.character(Ffull$F.Open))
FBfull$FB.Open<- as.numeric(as.character(FBfull$FB.Open))
FDXfull$FDX.Open<- as.numeric(as.character(FDXfull$FDX.Open))
GDfull$GD.Open<- as.numeric(as.character(GDfull$GD.Open))
GEfull$GE.Open<- as.numeric(as.character(GEfull$GE.Open))
GILDfull$GILD.Open<- as.numeric(as.character(GILDfull$GILD.Open))
GMfull$GM.Open<- as.numeric(as.character(GMfull$GM.Open))
GOOGfull$GOOG.Open<- as.numeric(as.character(GOOGfull$GOOG.Open))
GOOGLfull$GOOGL.Open<- as.numeric(as.character(GOOGLfull$GOOGL.Open))
GSfull$GS.Open<- as.numeric(as.character(GSfull$GS.Open))
HDfull$HD.Open<- as.numeric(as.character(HDfull$HD.Open))
HONfull$HON.Open<- as.numeric(as.character(HONfull$HON.Open))
IBMfull$IBM.Open<- as.numeric(as.character(IBMfull$IBM.Open))
INTCfull$INTC.Open<- as.numeric(as.character(INTCfull$INTC.Open))
JNJfull$JNJ.Open<- as.numeric(as.character(JNJfull$JNJ.Open))
JPMfull$JPM.Open<- as.numeric(as.character(JPMfull$JPM.Open))
KHCfull$KHC.Open<- as.numeric(as.character(KHCfull$KHC.Open))
KMIfull$KMI.Open<- as.numeric(as.character(KMIfull$KMI.Open))
KOfull$KO.Open<- as.numeric(as.character(KOfull$KO.Open))
LLYfull$LLY.Open<- as.numeric(as.character(LLYfull$LLY.Open))
LMTfull$LMT.Open<- as.numeric(as.character(LMTfull$LMT.Open))
LOWfull$LOW.Open<- as.numeric(as.character(LOWfull$LOW.Open))
MAfull$MA.Open<- as.numeric(as.character(MAfull$MA.Open))
MCDfull$MCD.Open<- as.numeric(as.character(MCDfull$MCD.Open))
MDLZfull$MDLZ.Open<- as.numeric(as.character(MDLZfull$MDLZ.Open))
MDTfull$MDT.Open<- as.numeric(as.character(MDTfull$MDT.Open))
METfull$MET.Open<- as.numeric(as.character(METfull$MET.Open))
MMMfull$MMM.Open<- as.numeric(as.character(MMMfull$MMM.Open))
MOfull$MO.Open<- as.numeric(as.character(MOfull$MO.Open))
MRKfull$MRK.Open<- as.numeric(as.character(MRKfull$MRK.Open))
MSfull$MS.Open<- as.numeric(as.character(MSfull$MS.Open))
MSFTfull$MSFT.Open<- as.numeric(as.character(MSFTfull$MSFT.Open))
NEEfull$NEE.Open<- as.numeric(as.character(NEEfull$NEE.Open))
NFLXfull$NFLX.Open<- as.numeric(as.character(NFLXfull$NFLX.Open))
NKEfull$NKE.Open<- as.numeric(as.character(NKEfull$NKE.Open))
NVDAfull$NVDA.Open<- as.numeric(as.character(NVDAfull$NVDA.Open))
ORCLfull$ORCL.Open<- as.numeric(as.character(ORCLfull$ORCL.Open))
OXYfull$OXY.Open<- as.numeric(as.character(OXYfull$OXY.Open))
PEPfull$PEP.Open<- as.numeric(as.character(PEPfull$PEP.Open))
PFEfull$PFE.Open<- as.numeric(as.character(PFEfull$PFE.Open))
PGfull$PG.Open<- as.numeric(as.character(PGfull$PG.Open))
PMfull$PM.Open<- as.numeric(as.character(PMfull$PM.Open))
PYPLfull$PYPL.Open<- as.numeric(as.character(PYPLfull$PYPL.Open))
QCOMfull$QCOM.Open<- as.numeric(as.character(QCOMfull$QCOM.Open))
RTNfull$RTN.Open<- as.numeric(as.character(RTNfull$RTN.Open))
SBUXfull$SBUX.Open<- as.numeric(as.character(SBUXfull$SBUX.Open))
SLBfull$SLB.Open<- as.numeric(as.character(SLBfull$SLB.Open))
SOfull$SO.Open<- as.numeric(as.character(SOfull$SO.Open))
SPGfull$SPG.Open<- as.numeric(as.character(SPGfull$SPG.Open))
Tfull$T.Open<- as.numeric(as.character(Tfull$T.Open))
TGTfull$TGT.Open<- as.numeric(as.character(TGTfull$TGT.Open))
TXNfull$TXN.Open<- as.numeric(as.character(TXNfull$TXN.Open))
UNHfull$UNH.Open<- as.numeric(as.character(UNHfull$UNH.Open))
UNPfull$UNP.Open<- as.numeric(as.character(UNPfull$UNP.Open))
UPSfull$UPS.Open<- as.numeric(as.character(UPSfull$UPS.Open))
USBfull$USB.Open<- as.numeric(as.character(USBfull$USB.Open))
UTXfull$UTX.Open<- as.numeric(as.character(UTXfull$UTX.Open))
Vfull$V.Open<- as.numeric(as.character(Vfull$V.Open))
VZfull$VZ.Open<- as.numeric(as.character(VZfull$VZ.Open))
WBAfull$WBA.Open<- as.numeric(as.character(WBAfull$WBA.Open))
WFCfull$WFC.Open<- as.numeric(as.character(WFCfull$WFC.Open))
WMTfull$WMT.Open<- as.numeric(as.character(WMTfull$WMT.Open))
XOMfull$XOM.Open<- as.numeric(as.character(XOMfull$XOM.Open))


#transform High column from factor to numeric
AAPLfull$AAPL.High<- as.numeric(as.character(AAPLfull$AAPL.High))
ABBVfull$ABBV.High<- as.numeric(as.character(ABBVfull$ABBV.High))
ABTfull$ABT.High<- as.numeric(as.character(ABTfull$ABT.High))
ACNfull$ACN.High<- as.numeric(as.character(ACNfull$ACN.High))
ADBEfull$ADBE.High<- as.numeric(as.character(ADBEfull$ADBE.High))
AGNfull$AGN.High<- as.numeric(as.character(AGNfull$AGN.High))
AIGfull$AIG.High<- as.numeric(as.character(AIGfull$AIG.High))
ALLfull$ALL.High<- as.numeric(as.character(ALLfull$ALL.High))
AMGNfull$AMGN.High<- as.numeric(as.character(AMGNfull$AMGN.High))
AMZNfull$AMZN.High<- as.numeric(as.character(AMZNfull$AMZN.High))
AXPfull$AXP.High<- as.numeric(as.character(AXPfull$AXP.High))
BAfull$BA.High<- as.numeric(as.character(BAfull$BA.High))
BACfull$BAC.High<- as.numeric(as.character(BACfull$BAC.High))
BIIBfull$BIIB.High<- as.numeric(as.character(BIIBfull$BIIB.High))
BKfull$BK.High<- as.numeric(as.character(BKfull$BK.High))
BKNGfull$BKNG.High<- as.numeric(as.character(BKNGfull$BKNG.High))
BLKfull$BLK.High<- as.numeric(as.character(BLKfull$BLK.High))
BMYfull$BMY.High<- as.numeric(as.character(BMYfull$BMY.High))
Cfull$C.High<- as.numeric(as.character(Cfull$C.High))
CATfull$CAT.High<- as.numeric(as.character(CATfull$CAT.High))
CELGfull$CELG.High<- as.numeric(as.character(CELGfull$CELG.High))
CHTRfull$CHTR.High<- as.numeric(as.character(CHTRfull$CHTR.High))
CLfull$CL.High<- as.numeric(as.character(CLfull$CL.High))
CMCSAfull$CMCSA.High<- as.numeric(as.character(CMCSAfull$CMCSA.High))
COFfull$COF.High<- as.numeric(as.character(COFfull$COF.High))
COPfull$COP.High<- as.numeric(as.character(COPfull$COP.High))
COSTfull$COST.High<- as.numeric(as.character(COSTfull$COST.High))
CSCOfull$CSCO.High<- as.numeric(as.character(CSCOfull$CSCO.High))
CVSfull$CVS.High<- as.numeric(as.character(CVSfull$CVS.High))
CVXfull$CVX.High<- as.numeric(as.character(CVXfull$CVX.High))
DDfull$DD.High<- as.numeric(as.character(DDfull$DD.High))
DHRfull$DHR.High<- as.numeric(as.character(DHRfull$DHR.High))
DISfull$DIS.High<- as.numeric(as.character(DISfull$DIS.High))
DOWfull$DOW.High<- as.numeric(as.character(DOWfull$DOW.High))
DUKfull$DUK.High<- as.numeric(as.character(DUKfull$DUK.High))
EMRfull$EMR.High<- as.numeric(as.character(EMRfull$EMR.High))
EXCfull$EXC.High<- as.numeric(as.character(EXCfull$EXC.High))
Ffull$F.High<- as.numeric(as.character(Ffull$F.High))
FBfull$FB.High<- as.numeric(as.character(FBfull$FB.High))
FDXfull$FDX.High<- as.numeric(as.character(FDXfull$FDX.High))
GDfull$GD.High<- as.numeric(as.character(GDfull$GD.High))
GEfull$GE.High<- as.numeric(as.character(GEfull$GE.High))
GILDfull$GILD.High<- as.numeric(as.character(GILDfull$GILD.High))
GMfull$GM.High<- as.numeric(as.character(GMfull$GM.High))
GOOGfull$GOOG.High<- as.numeric(as.character(GOOGfull$GOOG.High))
GOOGLfull$GOOGL.High<- as.numeric(as.character(GOOGLfull$GOOGL.High))
GSfull$GS.High<- as.numeric(as.character(GSfull$GS.High))
HDfull$HD.High<- as.numeric(as.character(HDfull$HD.High))
HONfull$HON.High<- as.numeric(as.character(HONfull$HON.High))
IBMfull$IBM.High<- as.numeric(as.character(IBMfull$IBM.High))
INTCfull$INTC.High<- as.numeric(as.character(INTCfull$INTC.High))
JNJfull$JNJ.High<- as.numeric(as.character(JNJfull$JNJ.High))
JPMfull$JPM.High<- as.numeric(as.character(JPMfull$JPM.High))
KHCfull$KHC.High<- as.numeric(as.character(KHCfull$KHC.High))
KMIfull$KMI.High<- as.numeric(as.character(KMIfull$KMI.High))
KOfull$KO.High<- as.numeric(as.character(KOfull$KO.High))
LLYfull$LLY.High<- as.numeric(as.character(LLYfull$LLY.High))
LMTfull$LMT.High<- as.numeric(as.character(LMTfull$LMT.High))
LOWfull$LOW.High<- as.numeric(as.character(LOWfull$LOW.High))
MAfull$MA.High<- as.numeric(as.character(MAfull$MA.High))
```

```r
MCDfull$MCD.High<- as.numeric(as.character(MCDfull$MCD.High))
MDLZfull$MDLZ.High<- as.numeric(as.character(MDLZfull$MDLZ.High))
MDTfull$MDT.High<- as.numeric(as.character(MDTfull$MDT.High))
METfull$MET.High<- as.numeric(as.character(METfull$MET.High))
MMMfull$MMM.High<- as.numeric(as.character(MMMfull$MMM.High))
MOfull$MO.High<- as.numeric(as.character(MOfull$MO.High))
MRKfull$MRK.High<- as.numeric(as.character(MRKfull$MRK.High))
MSfull$MS.High<- as.numeric(as.character(MSfull$MS.High))
MSFTfull$MSFT.High<- as.numeric(as.character(MSFTfull$MSFT.High))
NEEfull$NEE.High<- as.numeric(as.character(NEEfull$NEE.High))
NFLXfull$NFLX.High<- as.numeric(as.character(NFLXfull$NFLX.High))
NKEfull$NKE.High<- as.numeric(as.character(NKEfull$NKE.High))
NVDAfull$NVDA.High<- as.numeric(as.character(NVDAfull$NVDA.High))
ORCLfull$ORCL.High<- as.numeric(as.character(ORCLfull$ORCL.High))
OXYfull$OXY.High<- as.numeric(as.character(OXYfull$OXY.High))
PEPfull$PEP.High<- as.numeric(as.character(PEPfull$PEP.High))
PFEfull$PFE.High<- as.numeric(as.character(PFEfull$PFE.High))
PGfull$PG.High<- as.numeric(as.character(PGfull$PG.High))
PMfull$PM.High<- as.numeric(as.character(PMfull$PM.High))
PYPLfull$PYPL.High<- as.numeric(as.character(PYPLfull$PYPL.High))
QCOMfull$QCOM.High<- as.numeric(as.character(QCOMfull$QCOM.High))
RTNfull$RTN.High<- as.numeric(as.character(RTNfull$RTN.High))
SBUXfull$SBUX.High<- as.numeric(as.character(SBUXfull$SBUX.High))
SLBfull$SLB.High<- as.numeric(as.character(SLBfull$SLB.High))
SOfull$SO.High<- as.numeric(as.character(SOfull$SO.High))
SPGfull$SPG.High<- as.numeric(as.character(SPGfull$SPG.High))
Tfull$T.High<- as.numeric(as.character(Tfull$T.High))
TGTfull$TGT.High<- as.numeric(as.character(TGTfull$TGT.High))
TXNfull$TXN.High<- as.numeric(as.character(TXNfull$TXN.High))
UNHfull$UNH.High<- as.numeric(as.character(UNHfull$UNH.High))
UNPfull$UNP.High<- as.numeric(as.character(UNPfull$UNP.High))
UPSfull$UPS.High<- as.numeric(as.character(UPSfull$UPS.High))
USBfull$USB.High<- as.numeric(as.character(USBfull$USB.High))
UTXfull$UTX.High<- as.numeric(as.character(UTXfull$UTX.High))
Vfull$V.High<- as.numeric(as.character(Vfull$V.High))
VZfull$VZ.High<- as.numeric(as.character(VZfull$VZ.High))
WBAfull$WBA.High<- as.numeric(as.character(WBAfull$WBA.High))
WFCfull$WFC.High<- as.numeric(as.character(WFCfull$WFC.High))
WMTfull$WMT.High<- as.numeric(as.character(WMTfull$WMT.High))
XOMfull$XOM.High<- as.numeric(as.character(XOMfull$XOM.High))


#Transfrom Low Column classification from factor into numeric
AAPLfull$AAPL.Low<- as.numeric(as.character(AAPLfull$AAPL.Low))
ABBVfull$ABBV.Low<- as.numeric(as.character(ABBVfull$ABBV.Low))
ABTfull$ABT.Low<- as.numeric(as.character(ABTfull$ABT.Low))
ACNfull$ACN.Low<- as.numeric(as.character(ACNfull$ACN.Low))
ADBEfull$ADBE.Low<- as.numeric(as.character(ADBEfull$ADBE.Low))
AGNfull$AGN.Low<- as.numeric(as.character(AGNfull$AGN.Low))
AIGfull$AIG.Low<- as.numeric(as.character(AIGfull$AIG.Low))
ALLfull$ALL.Low<- as.numeric(as.character(ALLfull$ALL.Low))
AMGNfull$AMGN.Low<- as.numeric(as.character(AMGNfull$AMGN.Low))
AMZNfull$AMZN.Low<- as.numeric(as.character(AMZNfull$AMZN.Low))
AXPfull$AXP.Low<- as.numeric(as.character(AXPfull$AXP.Low))
BAfull$BA.Low<- as.numeric(as.character(BAfull$BA.Low))
BACfull$BAC.Low<- as.numeric(as.character(BACfull$BAC.Low))
BIIBfull$BIIB.Low<- as.numeric(as.character(BIIBfull$BIIB.Low))
BKfull$BK.Low<- as.numeric(as.character(BKfull$BK.Low))
BKNGfull$BKNG.Low<- as.numeric(as.character(BKNGfull$BKNG.Low))
BLKfull$BLK.Low<- as.numeric(as.character(BLKfull$BLK.Low))
BMYfull$BMY.Low<- as.numeric(as.character(BMYfull$BMY.Low))
Cfull$C.Low<- as.numeric(as.character(Cfull$C.Low))
CATfull$CAT.Low<- as.numeric(as.character(CATfull$CAT.Low))
CELGfull$CELG.Low<- as.numeric(as.character(CELGfull$CELG.Low))
CHTRfull$CHTR.Low<- as.numeric(as.character(CHTRfull$CHTR.Low))
CLfull$CL.Low<- as.numeric(as.character(CLfull$CL.Low))
CMCSAfull$CMCSA.Low<- as.numeric(as.character(CMCSAfull$CMCSA.Low))
COFfull$COF.Low<- as.numeric(as.character(COFfull$COF.Low))
COPfull$COP.Low<- as.numeric(as.character(COPfull$COP.Low))
COSTfull$COST.Low<- as.numeric(as.character(COSTfull$COST.Low))
CSCOfull$CSCO.Low<- as.numeric(as.character(CSCOfull$CSCO.Low))
CVSfull$CVS.Low<- as.numeric(as.character(CVSfull$CVS.Low))
CVXfull$CVX.Low<- as.numeric(as.character(CVXfull$CVX.Low))
DDfull$DD.Low<- as.numeric(as.character(DDfull$DD.Low))
DHRfull$DHR.Low<- as.numeric(as.character(DHRfull$DHR.Low))
DISfull$DIS.Low<- as.numeric(as.character(DISfull$DIS.Low))
DOWfull$DOW.Low<- as.numeric(as.character(DOWfull$DOW.Low))
DUKfull$DUK.Low<- as.numeric(as.character(DUKfull$DUK.Low))
EMRfull$EMR.Low<- as.numeric(as.character(EMRfull$EMR.Low))
EXCfull$EXC.Low<- as.numeric(as.character(EXCfull$EXC.Low))
Ffull$F.Low<- as.numeric(as.character(Ffull$F.Low))
FBfull$FB.Low<- as.numeric(as.character(FBfull$FB.Low))
FDXfull$FDX.Low<- as.numeric(as.character(FDXfull$FDX.Low))
GDfull$GD.Low<- as.numeric(as.character(GDfull$GD.Low))
GEfull$GE.Low<- as.numeric(as.character(GEfull$GE.Low))
GILDfull$GILD.Low<- as.numeric(as.character(GILDfull$GILD.Low))
GMfull$GM.Low<- as.numeric(as.character(GMfull$GM.Low))
GOOGfull$GOOG.Low<- as.numeric(as.character(GOOGfull$GOOG.Low))
GOOGLfull$GOOGL.Low<- as.numeric(as.character(GOOGLfull$GOOGL.Low))
GSfull$GS.Low<- as.numeric(as.character(GSfull$GS.Low))
HDfull$HD.Low<- as.numeric(as.character(HDfull$HD.Low))
HONfull$HON.Low<- as.numeric(as.character(HONfull$HON.Low))
IBMfull$IBM.Low<- as.numeric(as.character(IBMfull$IBM.Low))
INTCfull$INTC.Low<- as.numeric(as.character(INTCfull$INTC.Low))
JNJfull$JNJ.Low<- as.numeric(as.character(JNJfull$JNJ.Low))
JPMfull$JPM.Low<- as.numeric(as.character(JPMfull$JPM.Low))
KHCfull$KHC.Low<- as.numeric(as.character(KHCfull$KHC.Low))
KMIfull$KMI.Low<- as.numeric(as.character(KMIfull$KMI.Low))
KOfull$KO.Low<- as.numeric(as.character(KOfull$KO.Low))
LLYfull$LLY.Low<- as.numeric(as.character(LLYfull$LLY.Low))
LMTfull$LMT.Low<- as.numeric(as.character(LMTfull$LMT.Low))
LOWfull$LOW.Low<- as.numeric(as.character(LOWfull$LOW.Low))
MAfull$MA.Low<- as.numeric(as.character(MAfull$MA.Low))
MCDfull$MCD.Low<- as.numeric(as.character(MCDfull$MCD.Low))
MDLZfull$MDLZ.Low<- as.numeric(as.character(MDLZfull$MDLZ.Low))
MDTfull$MDT.Low<- as.numeric(as.character(MDTfull$MDT.Low))
METfull$MET.Low<- as.numeric(as.character(METfull$MET.Low))
MMMfull$MMM.Low<- as.numeric(as.character(MMMfull$MMM.Low))
MOfull$MO.Low<- as.numeric(as.character(MOfull$MO.Low))
MRKfull$MRK.Low<- as.numeric(as.character(MRKfull$MRK.Low))
MSfull$MS.Low<- as.numeric(as.character(MSfull$MS.Low))
MSFTfull$MSFT.Low<- as.numeric(as.character(MSFTfull$MSFT.Low))
NEEfull$NEE.Low<- as.numeric(as.character(NEEfull$NEE.Low))
NFLXfull$NFLX.Low<- as.numeric(as.character(NFLXfull$NFLX.Low))
NKEfull$NKE.Low<- as.numeric(as.character(NKEfull$NKE.Low))
NVDAfull$NVDA.Low<- as.numeric(as.character(NVDAfull$NVDA.Low))
ORCLfull$ORCL.Low<- as.numeric(as.character(ORCLfull$ORCL.Low))
OXYfull$OXY.Low<- as.numeric(as.character(OXYfull$OXY.Low))
PEPfull$PEP.Low<- as.numeric(as.character(PEPfull$PEP.Low))
PFEfull$PFE.Low<- as.numeric(as.character(PFEfull$PFE.Low))
PGfull$PG.Low<- as.numeric(as.character(PGfull$PG.Low))
PMfull$PM.Low<- as.numeric(as.character(PMfull$PM.Low))
PYPLfull$PYPL.Low<- as.numeric(as.character(PYPLfull$PYPL.Low))
QCOMfull$QCOM.Low<- as.numeric(as.character(QCOMfull$QCOM.Low))
RTNfull$RTN.Low<- as.numeric(as.character(RTNfull$RTN.Low))
SBUXfull$SBUX.Low<- as.numeric(as.character(SBUXfull$SBUX.Low))
SLBfull$SLB.Low<- as.numeric(as.character(SLBfull$SLB.Low))
SOfull$SO.Low<- as.numeric(as.character(SOfull$SO.Low))
SPGfull$SPG.Low<- as.numeric(as.character(SPGfull$SPG.Low))
Tfull$T.Low<- as.numeric(as.character(Tfull$T.Low))
TGTfull$TGT.Low<- as.numeric(as.character(TGTfull$TGT.Low))
TXNfull$TXN.Low<- as.numeric(as.character(TXNfull$TXN.Low))
UNHfull$UNH.Low<- as.numeric(as.character(UNHfull$UNH.Low))
UNPfull$UNP.Low<- as.numeric(as.character(UNPfull$UNP.Low))
UPSfull$UPS.Low<- as.numeric(as.character(UPSfull$UPS.Low))
USBfull$USB.Low<- as.numeric(as.character(USBfull$USB.Low))
UTXfull$UTX.Low<- as.numeric(as.character(UTXfull$UTX.Low))
Vfull$V.Low<- as.numeric(as.character(Vfull$V.Low))
VZfull$VZ.Low<- as.numeric(as.character(VZfull$VZ.Low))
WBAfull$WBA.Low<- as.numeric(as.character(WBAfull$WBA.Low))
WFCfull$WFC.Low<- as.numeric(as.character(WFCfull$WFC.Low))
WMTfull$WMT.Low<- as.numeric(as.character(WMTfull$WMT.Low))
XOMfull$XOM.Low<- as.numeric(as.character(XOMfull$XOM.Low))


# Transfrom Close column classification from factor into numeric
AAPLfull$AAPL.Close<- as.numeric(as.character(AAPLfull$AAPL.Close))
ABBVfull$ABBV.Close<- as.numeric(as.character(ABBVfull$ABBV.Close))
ABTfull$ABT.Close<- as.numeric(as.character(ABTfull$ABT.Close))
ACNfull$ACN.Close<- as.numeric(as.character(ACNfull$ACN.Close))
ADBEfull$ADBE.Close<- as.numeric(as.character(ADBEfull$ADBE.Close))
AGNfull$AGN.Close<- as.numeric(as.character(AGNfull$AGN.Close))
AIGfull$AIG.Close<- as.numeric(as.character(AIGfull$AIG.Close))
ALLfull$ALL.Close<- as.numeric(as.character(ALLfull$ALL.Close))
AMGNfull$AMGN.Close<- as.numeric(as.character(AMGNfull$AMGN.Close))
AMZNfull$AMZN.Close<- as.numeric(as.character(AMZNfull$AMZN.Close))
AXPfull$AXP.Close<- as.numeric(as.character(AXPfull$AXP.Close))
BAfull$BA.Close<- as.numeric(as.character(BAfull$BA.Close))
BACfull$BAC.Close<- as.numeric(as.character(BACfull$BAC.Close))
BIIBfull$BIIB.Close<- as.numeric(as.character(BIIBfull$BIIB.Close))
BKfull$BK.Close<- as.numeric(as.character(BKfull$BK.Close))
BKNGfull$BKNG.Close<- as.numeric(as.character(BKNGfull$BKNG.Close))
BLKfull$BLK.Close<- as.numeric(as.character(BLKfull$BLK.Close))
BMYfull$BMY.Close<- as.numeric(as.character(BMYfull$BMY.Close))
Cfull$C.Close<- as.numeric(as.character(Cfull$C.Close))
CATfull$CAT.Close<- as.numeric(as.character(CATfull$CAT.Close))
CELGfull$CELG.Close<- as.numeric(as.character(CELGfull$CELG.Close))
CHTRfull$CHTR.Close<- as.numeric(as.character(CHTRfull$CHTR.Close))
CLfull$CL.Close<- as.numeric(as.character(CLfull$CL.Close))
CMCSAfull$CMCSA.Close<- as.numeric(as.character(CMCSAfull$CMCSA.Close))
COFfull$COF.Close<- as.numeric(as.character(COFfull$COF.Close))
```

```
COPfull$COP.Close<- as.numeric(as.character(COPfull$COP.Close))
COSTfull$COST.Close<- as.numeric(as.character(COSTfull$COST.Close))
CSCOfull$CSCO.Close<- as.numeric(as.character(CSCOfull$CSCO.Close))
CVSfull$CVS.Close<- as.numeric(as.character(CVSfull$CVS.Close))
CVXfull$CVX.Close<- as.numeric(as.character(CVXfull$CVX.Close))
DDfull$DD.Close<- as.numeric(as.character(DDfull$DD.Close))
DHRfull$DHR.Close<- as.numeric(as.character(DHRfull$DHR.Close))
DISfull$DIS.Close<- as.numeric(as.character(DISfull$DIS.Close))
DOWfull$DOW.Close<- as.numeric(as.character(DOWfull$DOW.Close))
DUKfull$DUK.Close<- as.numeric(as.character(DUKfull$DUK.Close))
EMRfull$EMR.Close<- as.numeric(as.character(EMRfull$EMR.Close))
EXCfull$EXC.Close<- as.numeric(as.character(EXCfull$EXC.Close))
Ffull$F.Close<- as.numeric(as.character(Ffull$F.Close))
FBfull$FB.Close<- as.numeric(as.character(FBfull$FB.Close))
FDXfull$FDX.Close<- as.numeric(as.character(FDXfull$FDX.Close))
GDfull$GD.Close<- as.numeric(as.character(GDfull$GD.Close))
GEfull$GE.Close<- as.numeric(as.character(GEfull$GE.Close))
GILDfull$GILD.Close<- as.numeric(as.character(GILDfull$GILD.Close))
GMfull$GM.Close<- as.numeric(as.character(GMfull$GM.Close))
GOOGfull$GOOG.Close<- as.numeric(as.character(GOOGfull$GOOG.Close))
GOOGLfull$GOOGL.Close<- as.numeric(as.character(GOOGLfull$GOOGL.Close))
GSfull$GS.Close<- as.numeric(as.character(GSfull$GS.Close))
HDfull$HD.Close<- as.numeric(as.character(HDfull$HD.Close))
HONfull$HON.Close<- as.numeric(as.character(HONfull$HON.Close))
IBMfull$IBM.Close<- as.numeric(as.character(IBMfull$IBM.Close))
INTCfull$INTC.Close<- as.numeric(as.character(INTCfull$INTC.Close))
JNJfull$JNJ.Close<- as.numeric(as.character(JNJfull$JNJ.Close))
JPMfull$JPM.Close<- as.numeric(as.character(JPMfull$JPM.Close))
KHCfull$KHC.Close<- as.numeric(as.character(KHCfull$KHC.Close))
KMIfull$KMI.Close<- as.numeric(as.character(KMIfull$KMI.Close))
KOfull$KO.Close<- as.numeric(as.character(KOfull$KO.Close))
LLYfull$LLY.Close<- as.numeric(as.character(LLYfull$LLY.Close))
LMTfull$LMT.Close<- as.numeric(as.character(LMTfull$LMT.Close))
LOWfull$LOW.Close<- as.numeric(as.character(LOWfull$LOW.Close))
MAfull$MA.Close<- as.numeric(as.character(MAfull$MA.Close))
MCDfull$MCD.Close<- as.numeric(as.character(MCDfull$MCD.Close))
MDLZfull$MDLZ.Close<- as.numeric(as.character(MDLZfull$MDLZ.Close))
MDTfull$MDT.Close<- as.numeric(as.character(MDTfull$MDT.Close))
METfull$MET.Close<- as.numeric(as.character(METfull$MET.Close))
MMMfull$MMM.Close<- as.numeric(as.character(MMMfull$MMM.Close))
MOfull$MO.Close<- as.numeric(as.character(MOfull$MO.Close))
MRKfull$MRK.Close<- as.numeric(as.character(MRKfull$MRK.Close))
MSfull$MS.Close<- as.numeric(as.character(MSfull$MS.Close))
MSFTfull$MSFT.Close<- as.numeric(as.character(MSFTfull$MSFT.Close))
NEEfull$NEE.Close<- as.numeric(as.character(NEEfull$NEE.Close))
NFLXfull$NFLX.Close<- as.numeric(as.character(NFLXfull$NFLX.Close))
NKEfull$NKE.Close<- as.numeric(as.character(NKEfull$NKE.Close))
NVDAfull$NVDA.Close<- as.numeric(as.character(NVDAfull$NVDA.Close))
ORCLfull$ORCL.Close<- as.numeric(as.character(ORCLfull$ORCL.Close))
OXYfull$OXY.Close<- as.numeric(as.character(OXYfull$OXY.Close))
PEPfull$PEP.Close<- as.numeric(as.character(PEPfull$PEP.Close))
PFEfull$PFE.Close<- as.numeric(as.character(PFEfull$PFE.Close))
PGfull$PG.Close<- as.numeric(as.character(PGfull$PG.Close))
PMfull$PM.Close<- as.numeric(as.character(PMfull$PM.Close))
PYPLfull$PYPL.Close<- as.numeric(as.character(PYPLfull$PYPL.Close))
QCOMfull$QCOM.Close<- as.numeric(as.character(QCOMfull$QCOM.Close))
RTNfull$RTN.Close<- as.numeric(as.character(RTNfull$RTN.Close))
SBUXfull$SBUX.Close<- as.numeric(as.character(SBUXfull$SBUX.Close))
SLBfull$SLB.Close<- as.numeric(as.character(SLBfull$SLB.Close))
SOfull$SO.Close<- as.numeric(as.character(SOfull$SO.Close))
SPGfull$SPG.Close<- as.numeric(as.character(SPGfull$SPG.Close))
Tfull$T.Close<- as.numeric(as.character(Tfull$T.Close))
TGTfull$TGT.Close<- as.numeric(as.character(TGTfull$TGT.Close))
TXNfull$TXN.Close<- as.numeric(as.character(TXNfull$TXN.Close))
UNHfull$UNH.Close<- as.numeric(as.character(UNHfull$UNH.Close))
UNPfull$UNP.Close<- as.numeric(as.character(UNPfull$UNP.Close))
UPSfull$UPS.Close<- as.numeric(as.character(UPSfull$UPS.Close))
USBfull$USB.Close<- as.numeric(as.character(USBfull$USB.Close))
UTXfull$UTX.Close<- as.numeric(as.character(UTXfull$UTX.Close))
Vfull$V.Close<- as.numeric(as.character(Vfull$V.Close))
VZfull$VZ.Close<- as.numeric(as.character(VZfull$VZ.Close))
WBAfull$WBA.Close<- as.numeric(as.character(WBAfull$WBA.Close))
WFCfull$WFC.Close<- as.numeric(as.character(WFCfull$WFC.Close))
WMTfull$WMT.Close<- as.numeric(as.character(WMTfull$WMT.Close))
XOMfull$XOM.Close<- as.numeric(as.character(XOMfull$XOM.Close))


# Transfrom Volume column classification from factor into numeric
AAPLfull$AAPL.Volume<- as.numeric(as.character(AAPLfull$AAPL.Volume))
ABBVfull$ABBV.Volume<- as.numeric(as.character(ABBVfull$ABBV.Volume))
ABTfull$ABT.Volume<- as.numeric(as.character(ABTfull$ABT.Volume))
ACNfull$ACN.Volume<- as.numeric(as.character(ACNfull$ACN.Volume))
ADBEfull$ADBE.Volume<- as.numeric(as.character(ADBEfull$ADBE.Volume))
AGNfull$AGN.Volume<- as.numeric(as.character(AGNfull$AGN.Volume))
AIGfull$AIG.Volume<- as.numeric(as.character(AIGfull$AIG.Volume))
ALLfull$ALL.Volume<- as.numeric(as.character(ALLfull$ALL.Volume))
AMGNfull$AMGN.Volume<- as.numeric(as.character(AMGNfull$AMGN.Volume))
AMZNfull$AMZN.Volume<- as.numeric(as.character(AMZNfull$AMZN.Volume))
AXPfull$AXP.Volume<- as.numeric(as.character(AXPfull$AXP.Volume))
BAfull$BA.Volume<- as.numeric(as.character(BAfull$BA.Volume))
BACfull$BAC.Volume<- as.numeric(as.character(BACfull$BAC.Volume))
BIIBfull$BIIB.Volume<- as.numeric(as.character(BIIBfull$BIIB.Volume))
BKfull$BK.Volume<- as.numeric(as.character(BKfull$BK.Volume))
BKNGfull$BKNG.Volume<- as.numeric(as.character(BKNGfull$BKNG.Volume))
BLKfull$BLK.Volume<- as.numeric(as.character(BLKfull$BLK.Volume))
BMYfull$BMY.Volume<- as.numeric(as.character(BMYfull$BMY.Volume))
Cfull$C.Volume<- as.numeric(as.character(Cfull$C.Volume))
CATfull$CAT.Volume<- as.numeric(as.character(CATfull$CAT.Volume))
CELGfull$CELG.Volume<- as.numeric(as.character(CELGfull$CELG.Volume))
CHTRfull$CHTR.Volume<- as.numeric(as.character(CHTRfull$CHTR.Volume))
CLfull$CL.Volume<- as.numeric(as.character(CLfull$CL.Volume))
CMCSAfull$CMCSA.Volume<- as.numeric(as.character(CMCSAfull$CMCSA.Volume))
COFfull$COF.Volume<- as.numeric(as.character(COFfull$COF.Volume))
COPfull$COP.Volume<- as.numeric(as.character(COPfull$COP.Volume))
COSTfull$COST.Volume<- as.numeric(as.character(COSTfull$COST.Volume))
CSCOfull$CSCO.Volume<- as.numeric(as.character(CSCOfull$CSCO.Volume))
CVSfull$CVS.Volume<- as.numeric(as.character(CVSfull$CVS.Volume))
CVXfull$CVX.Volume<- as.numeric(as.character(CVXfull$CVX.Volume))
DDfull$DD.Volume<- as.numeric(as.character(DDfull$DD.Volume))
DHRfull$DHR.Volume<- as.numeric(as.character(DHRfull$DHR.Volume))
DISfull$DIS.Volume<- as.numeric(as.character(DISfull$DIS.Volume))
DOWfull$DOW.Volume<- as.numeric(as.character(DOWfull$DOW.Volume))
DUKfull$DUK.Volume<- as.numeric(as.character(DUKfull$DUK.Volume))
EMRfull$EMR.Volume<- as.numeric(as.character(EMRfull$EMR.Volume))
EXCfull$EXC.Volume<- as.numeric(as.character(EXCfull$EXC.Volume))
Ffull$F.Volume<- as.numeric(as.character(Ffull$F.Volume))
FBfull$FB.Volume<- as.numeric(as.character(FBfull$FB.Volume))
FDXfull$FDX.Volume<- as.numeric(as.character(FDXfull$FDX.Volume))
GDfull$GD.Volume<- as.numeric(as.character(GDfull$GD.Volume))
GEfull$GE.Volume<- as.numeric(as.character(GEfull$GE.Volume))
GILDfull$GILD.Volume<- as.numeric(as.character(GILDfull$GILD.Volume))
GMfull$GM.Volume<- as.numeric(as.character(GMfull$GM.Volume))
GOOGfull$GOOG.Volume<- as.numeric(as.character(GOOGfull$GOOG.Volume))
GOOGLfull$GOOGL.Volume<- as.numeric(as.character(GOOGLfull$GOOGL.Volume))
GSfull$GS.Volume<- as.numeric(as.character(GSfull$GS.Volume))
HDfull$HD.Volume<- as.numeric(as.character(HDfull$HD.Volume))
HONfull$HON.Volume<- as.numeric(as.character(HONfull$HON.Volume))
IBMfull$IBM.Volume<- as.numeric(as.character(IBMfull$IBM.Volume))
INTCfull$INTC.Volume<- as.numeric(as.character(INTCfull$INTC.Volume))
JNJfull$JNJ.Volume<- as.numeric(as.character(JNJfull$JNJ.Volume))
JPMfull$JPM.Volume<- as.numeric(as.character(JPMfull$JPM.Volume))
KHCfull$KHC.Volume<- as.numeric(as.character(KHCfull$KHC.Volume))
KMIfull$KMI.Volume<- as.numeric(as.character(KMIfull$KMI.Volume))
KOfull$KO.Volume<- as.numeric(as.character(KOfull$KO.Volume))
LLYfull$LLY.Volume<- as.numeric(as.character(LLYfull$LLY.Volume))
LMTfull$LMT.Volume<- as.numeric(as.character(LMTfull$LMT.Volume))
LOWfull$LOW.Volume<- as.numeric(as.character(LOWfull$LOW.Volume))
MAfull$MA.Volume<- as.numeric(as.character(MAfull$MA.Volume))
MCDfull$MCD.Volume<- as.numeric(as.character(MCDfull$MCD.Volume))
MDLZfull$MDLZ.Volume<- as.numeric(as.character(MDLZfull$MDLZ.Volume))
MDTfull$MDT.Volume<- as.numeric(as.character(MDTfull$MDT.Volume))
METfull$MET.Volume<- as.numeric(as.character(METfull$MET.Volume))
MMMfull$MMM.Volume<- as.numeric(as.character(MMMfull$MMM.Volume))
MOfull$MO.Volume<- as.numeric(as.character(MOfull$MO.Volume))
MRKfull$MRK.Volume<- as.numeric(as.character(MRKfull$MRK.Volume))
MSfull$MS.Volume<- as.numeric(as.character(MSfull$MS.Volume))
MSFTfull$MSFT.Volume<- as.numeric(as.character(MSFTfull$MSFT.Volume))
NEEfull$NEE.Volume<- as.numeric(as.character(NEEfull$NEE.Volume))
NFLXfull$NFLX.Volume<- as.numeric(as.character(NFLXfull$NFLX.Volume))
NKEfull$NKE.Volume<- as.numeric(as.character(NKEfull$NKE.Volume))
NVDAfull$NVDA.Volume<- as.numeric(as.character(NVDAfull$NVDA.Volume))
ORCLfull$ORCL.Volume<- as.numeric(as.character(ORCLfull$ORCL.Volume))
OXYfull$OXY.Volume<- as.numeric(as.character(OXYfull$OXY.Volume))
PEPfull$PEP.Volume<- as.numeric(as.character(PEPfull$PEP.Volume))
PFEfull$PFE.Volume<- as.numeric(as.character(PFEfull$PFE.Volume))
PGfull$PG.Volume<- as.numeric(as.character(PGfull$PG.Volume))
PMfull$PM.Volume<- as.numeric(as.character(PMfull$PM.Volume))
PYPLfull$PYPL.Volume<- as.numeric(as.character(PYPLfull$PYPL.Volume))
QCOMfull$QCOM.Volume<- as.numeric(as.character(QCOMfull$QCOM.Volume))
RTNfull$RTN.Volume<- as.numeric(as.character(RTNfull$RTN.Volume))
SBUXfull$SBUX.Volume<- as.numeric(as.character(SBUXfull$SBUX.Volume))
SLBfull$SLB.Volume<- as.numeric(as.character(SLBfull$SLB.Volume))
SOfull$SO.Volume<- as.numeric(as.character(SOfull$SO.Volume))
SPGfull$SPG.Volume<- as.numeric(as.character(SPGfull$SPG.Volume))
Tfull$T.Volume<- as.numeric(as.character(Tfull$T.Volume))
TGTfull$TGT.Volume<- as.numeric(as.character(TGTfull$TGT.Volume))
TXNfull$TXN.Volume<- as.numeric(as.character(TXNfull$TXN.Volume))
UNHfull$UNH.Volume<- as.numeric(as.character(UNHfull$UNH.Volume))
UNPfull$UNP.Volume<- as.numeric(as.character(UNPfull$UNP.Volume))
UPSfull$UPS.Volume<- as.numeric(as.character(UPSfull$UPS.Volume))
USBfull$USB.Volume<- as.numeric(as.character(USBfull$USB.Volume))
```

```r
UTXfull$UTX.Volume<- as.numeric(as.character(UTXfull$UTX.Volume))
Vfull$V.Volume<- as.numeric(as.character(Vfull$V.Volume))
VZfull$VZ.Volume<- as.numeric(as.character(VZfull$VZ.Volume))
WBAfull$WBA.Volume<- as.numeric(as.character(WBAfull$WBA.Volume))
WFCfull$WFC.Volume<- as.numeric(as.character(WFCfull$WFC.Volume))
WMTfull$WMT.Volume<- as.numeric(as.character(WMTfull$WMT.Volume))
XOMfull$XOM.Volume<- as.numeric(as.character(XOMfull$XOM.Volume))

#Transfrom Adjusted column classification from factor into numeric
AAPLfull$AAPL.Adjusted<- as.numeric(as.character(AAPLfull$AAPL.Adjusted))
ABBVfull$ABBV.Adjusted<- as.numeric(as.character(ABBVfull$ABBV.Adjusted))
ABTfull$ABT.Adjusted<- as.numeric(as.character(ABTfull$ABT.Adjusted))
ACNfull$ACN.Adjusted<- as.numeric(as.character(ACNfull$ACN.Adjusted))
ADBEfull$ADBE.Adjusted<- as.numeric(as.character(ADBEfull$ADBE.Adjusted))
AGNfull$AGN.Adjusted<- as.numeric(as.character(AGNfull$AGN.Adjusted))
AIGfull$AIG.Adjusted<- as.numeric(as.character(AIGfull$AIG.Adjusted))
ALLfull$ALL.Adjusted<- as.numeric(as.character(ALLfull$ALL.Adjusted))
AMGNfull$AMGN.Adjusted<- as.numeric(as.character(AMGNfull$AMGN.Adjusted))
AMZNfull$AMZN.Adjusted<- as.numeric(as.character(AMZNfull$AMZN.Adjusted))
AXPfull$AXP.Adjusted<- as.numeric(as.character(AXPfull$AXP.Adjusted))
BAfull$BA.Adjusted<- as.numeric(as.character(BAfull$BA.Adjusted))
BACfull$BAC.Adjusted<- as.numeric(as.character(BACfull$BAC.Adjusted))
BIIBfull$BIIB.Adjusted<- as.numeric(as.character(BIIBfull$BIIB.Adjusted))
BKfull$BK.Adjusted<- as.numeric(as.character(BKfull$BK.Adjusted))
BKNGfull$BKNG.Adjusted<- as.numeric(as.character(BKNGfull$BKNG.Adjusted))
BLKfull$BLK.Adjusted<- as.numeric(as.character(BLKfull$BLK.Adjusted))
BMYfull$BMY.Adjusted<- as.numeric(as.character(BMYfull$BMY.Adjusted))
Cfull$C.Adjusted<- as.numeric(as.character(Cfull$C.Adjusted))
CATfull$CAT.Adjusted<- as.numeric(as.character(CATfull$CAT.Adjusted))
CELGfull$CELG.Adjusted<- as.numeric(as.character(CELGfull$CELG.Adjusted))
CHTRfull$CHTR.Adjusted<- as.numeric(as.character(CHTRfull$CHTR.Adjusted))
CLfull$CL.Adjusted<- as.numeric(as.character(CLfull$CL.Adjusted))
CMCSAfull$CMCSA.Adjusted<- as.numeric(as.character(CMCSAfull$CMCSA.Adjusted))
COFfull$COF.Adjusted<- as.numeric(as.character(COFfull$COF.Adjusted))
COPfull$COP.Adjusted<- as.numeric(as.character(COPfull$COP.Adjusted))
COSTfull$COST.Adjusted<- as.numeric(as.character(COSTfull$COST.Adjusted))
CSCOfull$CSCO.Adjusted<- as.numeric(as.character(CSCOfull$CSCO.Adjusted))
CVSfull$CVS.Adjusted<- as.numeric(as.character(CVSfull$CVS.Adjusted))
CVXfull$CVX.Adjusted<- as.numeric(as.character(CVXfull$CVX.Adjusted))
DDfull$DD.Adjusted<- as.numeric(as.character(DDfull$DD.Adjusted))
DHRfull$DHR.Adjusted<- as.numeric(as.character(DHRfull$DHR.Adjusted))
DISfull$DIS.Adjusted<- as.numeric(as.character(DISfull$DIS.Adjusted))
DOWfull$DOW.Adjusted<- as.numeric(as.character(DOWfull$DOW.Adjusted))
DUKfull$DUK.Adjusted<- as.numeric(as.character(DUKfull$DUK.Adjusted))
EMRfull$EMR.Adjusted<- as.numeric(as.character(EMRfull$EMR.Adjusted))
EXCfull$EXC.Adjusted<- as.numeric(as.character(EXCfull$EXC.Adjusted))
Ffull$F.Adjusted<- as.numeric(as.character(Ffull$F.Adjusted))
FBfull$FB.Adjusted<- as.numeric(as.character(FBfull$FB.Adjusted))
FDXfull$FDX.Adjusted<- as.numeric(as.character(FDXfull$FDX.Adjusted))
GDfull$GD.Adjusted<- as.numeric(as.character(GDfull$GD.Adjusted))
GEfull$GE.Adjusted<- as.numeric(as.character(GEfull$GE.Adjusted))
GILDfull$GILD.Adjusted<- as.numeric(as.character(GILDfull$GILD.Adjusted))
GMfull$GM.Adjusted<- as.numeric(as.character(GMfull$GM.Adjusted))
GOOGfull$GOOG.Adjusted<- as.numeric(as.character(GOOGfull$GOOG.Adjusted))
GOOGLfull$GOOGL.Adjusted<- as.numeric(as.character(GOOGLfull$GOOGL.Adjusted))
GSfull$GS.Adjusted<- as.numeric(as.character(GSfull$GS.Adjusted))
HDfull$HD.Adjusted<- as.numeric(as.character(HDfull$HD.Adjusted))
HONfull$HON.Adjusted<- as.numeric(as.character(HONfull$HON.Adjusted))
IBMfull$IBM.Adjusted<- as.numeric(as.character(IBMfull$IBM.Adjusted))
INTCfull$INTC.Adjusted<- as.numeric(as.character(INTCfull$INTC.Adjusted))
JNJfull$JNJ.Adjusted<- as.numeric(as.character(JNJfull$JNJ.Adjusted))
JPMfull$JPM.Adjusted<- as.numeric(as.character(JPMfull$JPM.Adjusted))
KHCfull$KHC.Adjusted<- as.numeric(as.character(KHCfull$KHC.Adjusted))
KMIfull$KMI.Adjusted<- as.numeric(as.character(KMIfull$KMI.Adjusted))
KOfull$KO.Adjusted<- as.numeric(as.character(KOfull$KO.Adjusted))
LLYfull$LLY.Adjusted<- as.numeric(as.character(LLYfull$LLY.Adjusted))
LMTfull$LMT.Adjusted<- as.numeric(as.character(LMTfull$LMT.Adjusted))
LOWfull$LOW.Adjusted<- as.numeric(as.character(LOWfull$LOW.Adjusted))
MAfull$MA.Adjusted<- as.numeric(as.character(MAfull$MA.Adjusted))
MCDfull$MCD.Adjusted<- as.numeric(as.character(MCDfull$MCD.Adjusted))
MDLZfull$MDLZ.Adjusted<- as.numeric(as.character(MDLZfull$MDLZ.Adjusted))
MDTfull$MDT.Adjusted<- as.numeric(as.character(MDTfull$MDT.Adjusted))
METfull$MET.Adjusted<- as.numeric(as.character(METfull$MET.Adjusted))
MMMfull$MMM.Adjusted<- as.numeric(as.character(MMMfull$MMM.Adjusted))
MOfull$MO.Adjusted<- as.numeric(as.character(MOfull$MO.Adjusted))
MRKfull$MRK.Adjusted<- as.numeric(as.character(MRKfull$MRK.Adjusted))
MSfull$MS.Adjusted<- as.numeric(as.character(MSfull$MS.Adjusted))
MSFTfull$MSFT.Adjusted<- as.numeric(as.character(MSFTfull$MSFT.Adjusted))
NEEfull$NEE.Adjusted<- as.numeric(as.character(NEEfull$NEE.Adjusted))
NFLXfull$NFLX.Adjusted<- as.numeric(as.character(NFLXfull$NFLX.Adjusted))
NKEfull$NKE.Adjusted<- as.numeric(as.character(NKEfull$NKE.Adjusted))
NVDAfull$NVDA.Adjusted<- as.numeric(as.character(NVDAfull$NVDA.Adjusted))
ORCLfull$ORCL.Adjusted<- as.numeric(as.character(ORCLfull$ORCL.Adjusted))
OXYfull$OXY.Adjusted<- as.numeric(as.character(OXYfull$OXY.Adjusted))
PEPfull$PEP.Adjusted<- as.numeric(as.character(PEPfull$PEP.Adjusted))
PFEfull$PFE.Adjusted<- as.numeric(as.character(PFEfull$PFE.Adjusted))
PGfull$PG.Adjusted<- as.numeric(as.character(PGfull$PG.Adjusted))
PMfull$PM.Adjusted<- as.numeric(as.character(PMfull$PM.Adjusted))
PYPLfull$PYPL.Adjusted<- as.numeric(as.character(PYPLfull$PYPL.Adjusted))
QCOMfull$QCOM.Adjusted<- as.numeric(as.character(QCOMfull$QCOM.Adjusted))
RTNfull$RTN.Adjusted<- as.numeric(as.character(RTNfull$RTN.Adjusted))
SBUXfull$SBUX.Adjusted<- as.numeric(as.character(SBUXfull$SBUX.Adjusted))
SLBfull$SLB.Adjusted<- as.numeric(as.character(SLBfull$SLB.Adjusted))
SOfull$SO.Adjusted<- as.numeric(as.character(SOfull$SO.Adjusted))
SPGfull$SPG.Adjusted<- as.numeric(as.character(SPGfull$SPG.Adjusted))
Tfull$T.Adjusted<- as.numeric(as.character(Tfull$T.Adjusted))
TGTfull$TGT.Adjusted<- as.numeric(as.character(TGTfull$TGT.Adjusted))
TXNfull$TXN.Adjusted<- as.numeric(as.character(TXNfull$TXN.Adjusted))
UNHfull$UNH.Adjusted<- as.numeric(as.character(UNHfull$UNH.Adjusted))
UNPfull$UNP.Adjusted<- as.numeric(as.character(UNPfull$UNP.Adjusted))
UPSfull$UPS.Adjusted<- as.numeric(as.character(UPSfull$UPS.Adjusted))
USBfull$USB.Adjusted<- as.numeric(as.character(USBfull$USB.Adjusted))
UTXfull$UTX.Adjusted<- as.numeric(as.character(UTXfull$UTX.Adjusted))
Vfull$V.Adjusted<- as.numeric(as.character(Vfull$V.Adjusted))
VZfull$VZ.Adjusted<- as.numeric(as.character(VZfull$VZ.Adjusted))
WBAfull$WBA.Adjusted<- as.numeric(as.character(WBAfull$WBA.Adjusted))
WFCfull$WFC.Adjusted<- as.numeric(as.character(WFCfull$WFC.Adjusted))
WMTfull$WMT.Adjusted<- as.numeric(as.character(WMTfull$WMT.Adjusted))
XOMfull$XOM.Adjusted<- as.numeric(as.character(XOMfull$XOM.Adjusted))

#Transform dividend value column's classification from factor to numeric
AAPLfull$value<- as.numeric(as.character(AAPLfull$value))
ABBVfull$value<- as.numeric(as.character(ABBVfull$value))
ABTfull$value<- as.numeric(as.character(ABTfull$value))
ACNfull$value<- as.numeric(as.character(ACNfull$value))
ADBEfull$value<- as.numeric(as.character(ADBEfull$value))
AGNfull$value<- as.numeric(as.character(AGNfull$value))
AIGfull$value<- as.numeric(as.character(AIGfull$value))
ALLfull$value<- as.numeric(as.character(ALLfull$value))
AMGNfull$value<- as.numeric(as.character(AMGNfull$value))
AMZNfull$value<- as.numeric(as.character(AMZNfull$value))
AXPfull$value<- as.numeric(as.character(AXPfull$value))
BAfull$value<- as.numeric(as.character(BAfull$value))
BACfull$value<- as.numeric(as.character(BACfull$value))
BIIBfull$value<- as.numeric(as.character(BIIBfull$value))
BKfull$value<- as.numeric(as.character(BKfull$value))
BKNGfull$value<- as.numeric(as.character(BKNGfull$value))
BLKfull$value<- as.numeric(as.character(BLKfull$value))
BMYfull$value<- as.numeric(as.character(BMYfull$value))
Cfull$value<- as.numeric(as.character(Cfull$value))
CATfull$value<- as.numeric(as.character(CATfull$value))
CELGfull$value<- as.numeric(as.character(CELGfull$value))
CHTRfull$value<- as.numeric(as.character(CHTRfull$value))
CLfull$value<- as.numeric(as.character(CLfull$value))
CMCSAfull$value<- as.numeric(as.character(CMCSAfull$value))
COFfull$value<- as.numeric(as.character(COFfull$value))
COPfull$value<- as.numeric(as.character(COPfull$value))
COSTfull$value<- as.numeric(as.character(COSTfull$value))
CSCOfull$value<- as.numeric(as.character(CSCOfull$value))
CVSfull$value<- as.numeric(as.character(CVSfull$value))
CVXfull$value<- as.numeric(as.character(CVXfull$value))
DDfull$value<- as.numeric(as.character(DDfull$value))
DHRfull$value<- as.numeric(as.character(DHRfull$value))
DISfull$value<- as.numeric(as.character(DISfull$value))
DOWfull$value<- as.numeric(as.character(DOWfull$value))
DUKfull$value<- as.numeric(as.character(DUKfull$value))
EMRfull$value<- as.numeric(as.character(EMRfull$value))
EXCfull$value<- as.numeric(as.character(EXCfull$value))
Ffull$value<- as.numeric(as.character(Ffull$value))
FBfull$value<- as.numeric(as.character(FBfull$value))
FDXfull$value<- as.numeric(as.character(FDXfull$value))
GDfull$value<- as.numeric(as.character(GDfull$value))
GEfull$value<- as.numeric(as.character(GEfull$value))
GILDfull$value<- as.numeric(as.character(GILDfull$value))
GMfull$value<- as.numeric(as.character(GMfull$value))
GOOGfull$value<- as.numeric(as.character(GOOGfull$value))
GOOGLfull$value<- as.numeric(as.character(GOOGLfull$value))
GSfull$value<- as.numeric(as.character(GSfull$value))
HDfull$value<- as.numeric(as.character(HDfull$value))
HONfull$value<- as.numeric(as.character(HONfull$value))
IBMfull$value<- as.numeric(as.character(IBMfull$value))
INTCfull$value<- as.numeric(as.character(INTCfull$value))
JNJfull$value<- as.numeric(as.character(JNJfull$value))
JPMfull$value<- as.numeric(as.character(JPMfull$value))
KHCfull$value<- as.numeric(as.character(KHCfull$value))
KMIfull$value<- as.numeric(as.character(KMIfull$value))
KOfull$value<- as.numeric(as.character(KOfull$value))
LLYfull$value<- as.numeric(as.character(LLYfull$value))
LMTfull$value<- as.numeric(as.character(LMTfull$value))
LOWfull$value<- as.numeric(as.character(LOWfull$value))
MAfull$value<- as.numeric(as.character(MAfull$value))
```

```
MCDfull$value<- as.numeric(as.character(MCDfull$value))
MDLZfull$value<- as.numeric(as.character(MDLZfull$value))
MDTfull$value<- as.numeric(as.character(MDTfull$value))
METfull$value<- as.numeric(as.character(METfull$value))
MMMfull$value<- as.numeric(as.character(MMMfull$value))
MOfull$value<- as.numeric(as.character(MOfull$value))
MRKfull$value<- as.numeric(as.character(MRKfull$value))
MSfull$value<- as.numeric(as.character(MSfull$value))
MSFTfull$value<- as.numeric(as.character(MSFTfull$value))
NEEfull$value<- as.numeric(as.character(NEEfull$value))
NFLXfull$value<- as.numeric(as.character(NFLXfull$value))
NKEfull$value<- as.numeric(as.character(NKEfull$value))
NVDAfull$value<- as.numeric(as.character(NVDAfull$value))
ORCLfull$value<- as.numeric(as.character(ORCLfull$value))
OXYfull$value<- as.numeric(as.character(OXYfull$value))
PEPfull$value<- as.numeric(as.character(PEPfull$value))
PFEfull$value<- as.numeric(as.character(PFEfull$value))
PGfull$value<- as.numeric(as.character(PGfull$value))
PMfull$value<- as.numeric(as.character(PMfull$value))
PYPLfull$value<- as.numeric(as.character(PYPLfull$value))
QCOMfull$value<- as.numeric(as.character(QCOMfull$value))
RTNfull$value<- as.numeric(as.character(RTNfull$value))
SBUXfull$value<- as.numeric(as.character(SBUXfull$value))
SLBfull$value<- as.numeric(as.character(SLBfull$value))
SOfull$value<- as.numeric(as.character(SOfull$value))
SPGfull$value<- as.numeric(as.character(SPGfull$value))
Tfull$value<- as.numeric(as.character(Tfull$value))
TGTfull$value<- as.numeric(as.character(TGTfull$value))
TXNfull$value<- as.numeric(as.character(TXNfull$value))
UNHfull$value<- as.numeric(as.character(UNHfull$value))
UNPfull$value<- as.numeric(as.character(UNPfull$value))
UPSfull$value<- as.numeric(as.character(UPSfull$value))
USBfull$value<- as.numeric(as.character(USBfull$value))
UTXfull$value<- as.numeric(as.character(UTXfull$value))
Vfull$value<- as.numeric(as.character(Vfull$value))
VZfull$value<- as.numeric(as.character(VZfull$value))
WBAfull$value<- as.numeric(as.character(WBAfull$value))
WFCfull$value<- as.numeric(as.character(WFCfull$value))
WMTfull$value<- as.numeric(as.character(WMTfull$value))
XOMfull$value<- as.numeric(as.character(XOMfull$value))

## Standardize nomanclature of header
# This is done so we could merge the all data frames into 1 large one
setnames(AAPLfull, old=c("AAPL.Open", "AAPL.High", "AAPL.Low", "AAPL.Close", "AAPL.Volume", "AAPL.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(ABBVfull, old=c("ABBV.Open", "ABBV.High", "ABBV.Low", "ABBV.Close", "ABBV.Volume", "ABBV.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(ABTfull, old=c("ABT.Open", "ABT.High", "ABT.Low", "ABT.Close", "ABT.Volume", "ABT.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(ACNfull, old=c("ACN.Open", "ACN.High", "ACN.Low", "ACN.Close", "ACN.Volume", "ACN.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(ADBEfull, old=c("ADBE.Open", "ADBE.High", "ADBE.Low", "ADBE.Close", "ADBE.Volume", "ADBE.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(AGNfull, old=c("AGN.Open", "AGN.High", "AGN.Low", "AGN.Close", "AGN.Volume", "AGN.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(AIGfull, old=c("AIG.Open", "AIG.High", "AIG.Low", "AIG.Close", "AIG.Volume", "AIG.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(ALLfull, old=c("ALL.Open", "ALL.High", "ALL.Low", "ALL.Close", "ALL.Volume", "ALL.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(AMGNfull, old=c("AMGN.Open", "AMGN.High", "AMGN.Low", "AMGN.Close", "AMGN.Volume", "AMGN.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(AMZNfull, old=c("AMZN.Open", "AMZN.High", "AMZN.Low", "AMZN.Close", "AMZN.Volume", "AMZN.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(AXPfull, old=c("AXP.Open", "AXP.High", "AXP.Low", "AXP.Close", "AXP.Volume", "AXP.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(BAfull, old=c("BA.Open", "BA.High", "BA.Low", "BA.Close", "BA.Volume", "BA.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(BACfull, old=c("BAC.Open", "BAC.High", "BAC.Low", "BAC.Close", "BAC.Volume", "BAC.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(BIIBfull, old=c("BIIB.Open", "BIIB.High", "BIIB.Low", "BIIB.Close", "BIIB.Volume", "BIIB.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(BKfull, old=c("BK.Open", "BK.High", "BK.Low", "BK.Close", "BK.Volume", "BK.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(BKNGfull, old=c("BKNG.Open", "BKNG.High", "BKNG.Low", "BKNG.Close", "BKNG.Volume", "BKNG.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(BLKfull, old=c("BLK.Open", "BLK.High", "BLK.Low", "BLK.Close", "BLK.Volume", "BLK.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(BMYfull, old=c("BMY.Open", "BMY.High", "BMY.Low", "BMY.Close", "BMY.Volume", "BMY.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(Cfull, old=c("C.Open", "C.High", "C.Low", "C.Close", "C.Volume", "C.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(CATfull, old=c("CAT.Open", "CAT.High", "CAT.Low", "CAT.Close", "CAT.Volume", "CAT.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(CELGfull, old=c("CELG.Open", "CELG.High", "CELG.Low", "CELG.Close", "CELG.Volume", "CELG.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(CHTRfull, old=c("CHTR.Open", "CHTR.High", "CHTR.Low", "CHTR.Close", "CHTR.Volume", "CHTR.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(CLfull, old=c("CL.Open", "CL.High", "CL.Low", "CL.Close", "CL.Volume", "CL.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(CMCSAfull, old=c("CMCSA.Open", "CMCSA.High", "CMCSA.Low", "CMCSA.Close", "CMCSA.Volume", "CMCSA.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol",
"Dividends"))
setnames(COFfull, old=c("COF.Open", "COF.High", "COF.Low", "COF.Close", "COF.Volume", "COF.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(COPfull, old=c("COP.Open", "COP.High", "COP.Low", "COP.Close", "COP.Volume", "COP.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(COSTfull, old=c("COST.Open", "COST.High", "COST.Low", "COST.Close", "COST.Volume", "COST.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(CSCOfull, old=c("CSCO.Open", "CSCO.High", "CSCO.Low", "CSCO.Close", "CSCO.Volume", "CSCO.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(CVSfull, old=c("CVS.Open", "CVS.High", "CVS.Low", "CVS.Close", "CVS.Volume", "CVS.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(CVXfull, old=c("CVX.Open", "CVX.High", "CVX.Low", "CVX.Close", "CVX.Volume", "CVX.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(DDfull, old=c("DD.Open", "DD.High", "DD.Low", "DD.Close", "DD.Volume", "DD.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(DHRfull, old=c("DHR.Open", "DHR.High", "DHR.Low", "DHR.Close", "DHR.Volume", "DHR.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(DISfull, old=c("DIS.Open", "DIS.High", "DIS.Low", "DIS.Close", "DIS.Volume", "DIS.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(DOWfull, old=c("DOW.Open", "DOW.High", "DOW.Low", "DOW.Close", "DOW.Volume", "DOW.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(DUKfull, old=c("DUK.Open", "DUK.High", "DUK.Low", "DUK.Close", "DUK.Volume", "DUK.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(EMRfull, old=c("EMR.Open", "EMR.High", "EMR.Low", "EMR.Close", "EMR.Volume", "EMR.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(EXCfull, old=c("EXC.Open", "EXC.High", "EXC.Low", "EXC.Close", "EXC.Volume", "EXC.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(Ffull, old=c("F.Open", "F.High", "F.Low", "F.Close", "F.Volume", "F.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(FBfull, old=c("FB.Open", "FB.High", "FB.Low", "FB.Close", "FB.Volume", "FB.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(FDXfull, old=c("FDX.Open", "FDX.High", "FDX.Low", "FDX.Close", "FDX.Volume", "FDX.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(GDfull, old=c("GD.Open", "GD.High", "GD.Low", "GD.Close", "GD.Volume", "GD.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(GEfull, old=c("GE.Open", "GE.High", "GE.Low", "GE.Close", "GE.Volume", "GE.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(GILDfull, old=c("GILD.Open", "GILD.High", "GILD.Low", "GILD.Close", "GILD.Volume", "GILD.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(GMfull, old=c("GM.Open", "GM.High", "GM.Low", "GM.Close", "GM.Volume", "GM.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(GOOGfull, old=c("GOOG.Open", "GOOG.High", "GOOG.Low", "GOOG.Close", "GOOG.Volume", "GOOG.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(GOOGLfull, old=c("GOOGL.Open", "GOOGL.High", "GOOGL.Low", "GOOGL.Close", "GOOGL.Volume", "GOOGL.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol",
"Dividends"))
setnames(GSfull, old=c("GS.Open", "GS.High", "GS.Low", "GS.Close", "GS.Volume", "GS.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(HDfull, old=c("HD.Open", "HD.High", "HD.Low", "HD.Close", "HD.Volume", "HD.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(HONfull, old=c("HON.Open", "HON.High", "HON.Low", "HON.Close", "HON.Volume", "HON.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(IBMfull, old=c("IBM.Open", "IBM.High", "IBM.Low", "IBM.Close", "IBM.Volume", "IBM.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(INTCfull, old=c("INTC.Open", "INTC.High", "INTC.Low", "INTC.Close", "INTC.Volume", "INTC.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(JNJfull, old=c("JNJ.Open", "JNJ.High", "JNJ.Low", "JNJ.Close", "JNJ.Volume", "JNJ.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(JPMfull, old=c("JPM.Open", "JPM.High", "JPM.Low", "JPM.Close", "JPM.Volume", "JPM.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(KHCfull, old=c("KHC.Open", "KHC.High", "KHC.Low", "KHC.Close", "KHC.Volume", "KHC.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(KMIfull, old=c("KMI.Open", "KMI.High", "KMI.Low", "KMI.Close", "KMI.Volume", "KMI.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(KOfull, old=c("KO.Open", "KO.High", "KO.Low", "KO.Close", "KO.Volume", "KO.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(LLYfull, old=c("LLY.Open", "LLY.High", "LLY.Low", "LLY.Close", "LLY.Volume", "LLY.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(LMTfull, old=c("LMT.Open", "LMT.High", "LMT.Low", "LMT.Close", "LMT.Volume", "LMT.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(LOWfull, old=c("LOW.Open", "LOW.High", "LOW.Low", "LOW.Close", "LOW.Volume", "LOW.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(MAfull, old=c("MA.Open", "MA.High", "MA.Low", "MA.Close", "MA.Volume", "MA.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(MCDfull, old=c("MCD.Open", "MCD.High", "MCD.Low", "MCD.Close", "MCD.Volume", "MCD.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(MDLZfull, old=c("MDLZ.Open", "MDLZ.High", "MDLZ.Low", "MDLZ.Close", "MDLZ.Volume", "MDLZ.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(MDTfull, old=c("MDT.Open", "MDT.High", "MDT.Low", "MDT.Close", "MDT.Volume", "MDT.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(METfull, old=c("MET.Open", "MET.High", "MET.Low", "MET.Close", "MET.Volume", "MET.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(MMMfull, old=c("MMM.Open", "MMM.High", "MMM.Low", "MMM.Close", "MMM.Volume", "MMM.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(MOfull, old=c("MO.Open", "MO.High", "MO.Low", "MO.Close", "MO.Volume", "MO.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(MRKfull, old=c("MRK.Open", "MRK.High", "MRK.Low", "MRK.Close", "MRK.Volume", "MRK.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(MSfull, old=c("MS.Open", "MS.High", "MS.Low", "MS.Close", "MS.Volume", "MS.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(MSFTfull, old=c("MSFT.Open", "MSFT.High", "MSFT.Low", "MSFT.Close", "MSFT.Volume", "MSFT.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(NEEfull, old=c("NEE.Open", "NEE.High", "NEE.Low", "NEE.Close", "NEE.Volume", "NEE.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(NFLXfull, old=c("NFLX.Open", "NFLX.High", "NFLX.Low", "NFLX.Close", "NFLX.Volume", "NFLX.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(NKEfull, old=c("NKE.Open", "NKE.High", "NKE.Low", "NKE.Close", "NKE.Volume", "NKE.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(NVDAfull, old=c("NVDA.Open", "NVDA.High", "NVDA.Low", "NVDA.Close", "NVDA.Volume", "NVDA.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(ORCLfull, old=c("ORCL.Open", "ORCL.High", "ORCL.Low", "ORCL.Close", "ORCL.Volume", "ORCL.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(OXYfull, old=c("OXY.Open", "OXY.High", "OXY.Low", "OXY.Close", "OXY.Volume", "OXY.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(PEPfull, old=c("PEP.Open", "PEP.High", "PEP.Low", "PEP.Close", "PEP.Volume", "PEP.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(PFEfull, old=c("PFE.Open", "PFE.High", "PFE.Low", "PFE.Close", "PFE.Volume", "PFE.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(PGfull, old=c("PG.Open", "PG.High", "PG.Low", "PG.Close", "PG.Volume", "PG.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(PMfull, old=c("PM.Open", "PM.High", "PM.Low", "PM.Close", "PM.Volume", "PM.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(PYPLfull, old=c("PYPL.Open", "PYPL.High", "PYPL.Low", "PYPL.Close", "PYPL.Volume", "PYPL.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(QCOMfull, old=c("QCOM.Open", "QCOM.High", "QCOM.Low", "QCOM.Close", "QCOM.Volume", "QCOM.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(RTNfull, old=c("RTN.Open", "RTN.High", "RTN.Low", "RTN.Close", "RTN.Volume", "RTN.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(SBUXfull, old=c("SBUX.Open", "SBUX.High", "SBUX.Low", "SBUX.Close", "SBUX.Volume", "SBUX.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(SLBfull, old=c("SLB.Open", "SLB.High", "SLB.Low", "SLB.Close", "SLB.Volume", "SLB.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(SOfull, old=c("SO.Open", "SO.High", "SO.Low", "SO.Close", "SO.Volume", "SO.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(SPGfull, old=c("SPG.Open", "SPG.High", "SPG.Low", "SPG.Close", "SPG.Volume", "SPG.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(Tfull, old=c("T.Open", "T.High", "T.Low", "T.Close", "T.Volume", "T.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(TGTfull, old=c("TGT.Open", "TGT.High", "TGT.Low", "TGT.Close", "TGT.Volume", "TGT.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(TXNfull, old=c("TXN.Open", "TXN.High", "TXN.Low", "TXN.Close", "TXN.Volume", "TXN.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(UNHfull, old=c("UNH.Open", "UNH.High", "UNH.Low", "UNH.Close", "UNH.Volume", "UNH.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(UNPfull, old=c("UNP.Open", "UNP.High", "UNP.Low", "UNP.Close", "UNP.Volume", "UNP.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(UPSfull, old=c("UPS.Open", "UPS.High", "UPS.Low", "UPS.Close", "UPS.Volume", "UPS.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(USBfull, old=c("USB.Open", "USB.High", "USB.Low", "USB.Close", "USB.Volume", "USB.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(UTXfull, old=c("UTX.Open", "UTX.High", "UTX.Low", "UTX.Close", "UTX.Volume", "UTX.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(Vfull, old=c("V.Open", "V.High", "V.Low", "V.Close", "V.Volume", "V.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(VZfull, old=c("VZ.Open", "VZ.High", "VZ.Low", "VZ.Close", "VZ.Volume", "VZ.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(WBAfull, old=c("WBA.Open", "WBA.High", "WBA.Low", "WBA.Close", "WBA.Volume", "WBA.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(WFCfull, old=c("WFC.Open", "WFC.High", "WFC.Low", "WFC.Close", "WFC.Volume", "WFC.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(WMTfull, old=c("WMT.Open", "WMT.High", "WMT.Low", "WMT.Close", "WMT.Volume", "WMT.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))
setnames(XOMfull, old=c("XOM.Open", "XOM.High", "XOM.Low", "XOM.Close", "XOM.Volume", "XOM.Adjusted", "VOLA", "Ticker", "value"), new=c("open", "High", "Low", "Close", "Volume", "Adjusted", "Volatility", "Symbol", "Dividends"))




## Merge all in one data frame
SnP100full<- rbind(AAPLfull, ABBVfull, ABTfull, ACNfull, ADBEfull, AGNfull, AIGfull, ALLfull, AMGNfull, AMZNfull, AXPfull, BAfull, BACfull, BIIBfull, BKfull, BKNGfull, BLKfull, BMYfull, Cfull, CATfull, CELGfull, CHTRfull, CLfull,
CMCSAfull, COFfull, COPfull, COSTfull, CSCOfull, CVSfull, CVXfull, DDfull, DHRfull, DISfull, DOWfull, DUKfull, EMRfull, EXCfull, Ffull, FBfull, FDXfull, GDfull, GEfull, GILDfull, GMfull, GOOGfull, GOOGLfull, GSfull, HDfull, HONfull,
IBMfull, INTCfull, JNJfull, JPMfull, KHCfull, KMIfull, KOfull, LLYfull, LMTfull, LOWfull, MAfull, MCDfull, MDLZfull, MDTfull, METfull, MMMfull, MOfull, MRKfull, MSfull, MSFTfull, NEEfull, NFLXfull, NKEfull, NVDAfull, ORCLfull, OXYfull,
PEPfull, PFEfull, PGfull, PMfull, PYPLfull, QCOMfull, RTNfull, SBUXfull, SLBfull, SOfull, SPGfull, Tfull, TGTfull, TXNfull, UNHfull, UNPfull, UPSfull, USBfull, UTXfull, Vfull, VZfull, WBAfull, WFCfull, WMTfull, XOMfull)
#successfully merged all data sets and added ticker

summary(SnP100full)

### CLEAN-UP - delete all initial symbols and call upon them again - that will enable me to call on the generic stock charts from fnancial R libraries

AAPL<- NULL
ABBV<- NULL
ABT<- NULL
ACN<- NULL
ADBE<- NULL
AGN<- NULL
AIG<- NULL
ALL<- NULL
AMGN<- NULL
AMZN<- NULL
AXP<- NULL
```

```
BA<- NULL
BAC<- NULL
BIIB<- NULL
BK<- NULL
BKNG<- NULL
BLK<- NULL
BMY<- NULL
C<- NULL
CAT<- NULL
CELG<- NULL
CHTR<- NULL
CL<- NULL
CMCSA<- NULL
COF<- NULL
COP<- NULL
COST<- NULL
CSCO<- NULL
CVS<- NULL
CVX<- NULL
DD<- NULL
DHR<- NULL
DIS<- NULL
DOW<- NULL
DUK<- NULL
EMR<- NULL
EXC<- NULL
F<- NULL
FB<- NULL
FDX<- NULL
GD<- NULL
GE<- NULL
GILD<- NULL
GM<- NULL
GOOG<- NULL
GOOGL<- NULL
GS<- NULL
HD<- NULL
HON<- NULL
IBM<- NULL
INTC<- NULL
JNJ<- NULL
JPM<- NULL
KHC<- NULL
KMI<- NULL
KO<- NULL
LLY<- NULL
LMT<- NULL
LOW<- NULL
MA<- NULL
MCD<- NULL
MDLZ<- NULL
MDT<- NULL
MET<- NULL
MMM<- NULL
MO<- NULL
MRK<- NULL
MS<- NULL
MSFT<- NULL
NEE<- NULL
NFLX<- NULL
NKE<- NULL
NVDA<- NULL
ORCL<- NULL
OXY<- NULL
PEP<- NULL
PFE<- NULL
PG<- NULL
PM<- NULL
PYPL<- NULL
QCOM<- NULL
RTN<- NULL
SBUX<- NULL
SLB<- NULL
SO<- NULL
SPG<- NULL
T<- NULL
TGT<- NULL
TXN<- NULL
UNH<- NULL
UNP<- NULL
UPS<- NULL
USB<- NULL
UTX<- NULL
V<- NULL
VZ<- NULL
WBA<- NULL
WFC<- NULL
WMT<- NULL
XOM<- NULL


#call them back
getSymbols(c('AAPL', 'ABBV', 'ABT', 'ACN', 'ADBE', 'AGN', 'AIG', 'ALL', 'AMGN', 'AMZN', 'AXP', 'BA', 'BAC', 'BIIB', 'BK', 'BKNG', 'BLK', 'BMY', 'BRK.B', 'C', 'CAT', 'CELG', 'CHTR', 'CL', 'CMCSA', 'COF', 'COP', 'COST', 'CSCO', 'CVS',
'CVX', 'DD', 'DHR', 'DIS', 'DOW', 'DUK', 'EMR', 'EXC', 'F', 'FB', 'FDX', 'GD', 'GE', 'GILD', 'GM', 'GOOG', 'GOOGL', 'GS', 'HD', 'HON', 'IBM', 'INTC', 'JNJ', 'JPM', 'KHC', 'KMI', 'KO', 'LLY', 'LMT', 'LOW', 'MA', 'MCD', 'MDLZ', 'MDT',
'MET', 'MMM', 'MO', 'MRK', 'MS', 'MSFT', 'NEE', 'NFLX', 'NKE', 'NVDA', 'ORCL', 'OXY', 'PEP', 'PFE', 'PG', 'PM', 'PYPL', 'QCOM', 'RTN', 'SBUX', 'SLB', 'SO', 'SPG', 'T', 'TGT', 'TXN', 'UNH', 'UNP', 'UPS', 'USB', 'UTX', 'V', 'VZ', 'WBA',
'WFC', 'WMT', 'XOM'
))


### Charts

candleChart(GOOG, subset= "last 4 months",multi.col=TRUE,theme="white", addMACD(fast = 12, slow = 26, signal = 9, type = "EMA"))


barChart(c(AAPL, ABBV, ABT, ACN, ADBE, AGN, AIG, ALL, AMGN, AMZN, AXP, BA, BAC, BIIB, BK, BKNG, BLK, BMY, C, CAT, CELG, CHTR, CL, CMCSA, COF, COP, COST, CSCO, CVS, CVX, DD, DHR, DIS, DOW, DUK, EMR, EXC, F, FB, FDX, GD, GE, GILD, GM,
GOOG, GOOGL, GS, HD, HON, IBM, INTC, JNJ, JPM, KHC, KMI, KO, LLY, LMT, LOW, MA, MCD, MDLZ, MDT, MET, MMM, MO, MRK, MS, MSFT, NEE, NFLX, NKE, NVDA, ORCL, OXY, PEP, PFE, PG, PM, PYPL, QCOM, RTN, SBUX, SLB, SO, SPG, T, TGT, TXN, UNH, UNP,
UPS, USB, UTX, V, VZ, WBA, WFC, WMT, XOM), subset="last 4 months" ,multi.col=TRUE,theme="white")










#Chart

barChart(AAPL,multi.col=TRUE,theme="white")
# Specify lookup parameters, and save for future sessions.


candleChart(GOOG, subset= "last 4 months",multi.col=TRUE,theme="white", addMACD(fast = 12, slow = 26, signal = 9, type = "EMA"))
#This worked before the data cleanup...

barChart(c(AAPL, ABBV, ABT, ACN, ADBE, AGN, AIG, ALL, AMGN, AMZN, AXP, BA, BAC, BIIB, BK, BKNG, BLK, BMY, C, CAT, CELG, CHTR, CL, CMCSA, COF, COP, COST, CSCO, CVS, CVX, DD, DHR, DIS, DOW, DUK, EMR, EXC, F, FB, FDX, GD, GE, GILD, GM,
GOOG, GOOGL, GS, HD, HON, IBM, INTC, JNJ, JPM, KHC, KMI, KO, LLY, LMT, LOW, MA, MCD, MDLZ, MDT, MET, MMM, MO, MRK, MS, MSFT, NEE, NFLX, NKE, NVDA, ORCL, OXY, PEP, PFE, PG, PM, PYPL, QCOM, RTN, SBUX, SLB, SO, SPG, T, TGT, TXN, UNH, UNP,
UPS, USB, UTX, V, VZ, WBA, WFC, WMT, XOM), subset="last 4 months" ,multi.col=TRUE,theme="white")
#This worked before the data cleanup...

chart_Series(c(AAPL$VOLA, ABBV$VOLA, ABT$VOLA, ACN$VOLA, ADBE$VOLA, AGN$VOLA, AIG$VOLA, ALL$VOLA, AMGN$VOLA, AMZN$VOLA, AXP$VOLA, BA$VOLA, BAC$VOLA, BIIB$VOLA, BK$VOLA, BKNG$VOLA, BLK$VOLA, BMY$VOLA, C$VOLA, CAT$VOLA, CELG$VOLA,
CHTR$VOLA, CL$VOLA, CMCSA$VOLA, COF$VOLA, COP$VOLA, COST$VOLA, CSCO$VOLA, CVS$VOLA, CVX$VOLA, DD$VOLA, DHR$VOLA, DIS$VOLA, DOW$VOLA, DUK$VOLA, EMR$VOLA, EXC$VOLA, F$VOLA, FB$VOLA, FDX$VOLA, GD$VOLA, GE$VOLA, GILD$VOLA, GM$VOLA,
GOOG$VOLA, GOOGL$VOLA, GS$VOLA, HD$VOLA, HON$VOLA, IBM$VOLA, INTC$VOLA, JNJ$VOLA, JPM$VOLA, KHC$VOLA, KMI$VOLA, KO$VOLA, LLY$VOLA, LMT$VOLA, LOW$VOLA, MA$VOLA, MCD$VOLA, MDLZ$VOLA, MDT$VOLA, MET$VOLA, MMM$VOLA, MO$VOLA, MRK$VOLA,
MS$VOLA, MSFT$VOLA, NEE$VOLA, NFLX$VOLA, NKE$VOLA, NVDA$VOLA, ORCL$VOLA, OXY$VOLA, PEP$VOLA, PFE$VOLA, PG$VOLA, PM$VOLA, PYPL$VOLA, QCOM$VOLA, RTN$VOLA, SBUX$VOLA, SLB$VOLA, SO$VOLA, SPG$VOLA, T$VOLA, TGT$VOLA, TXN$VOLA, UNH$VOLA,
UNP$VOLA, UPS$VOLA, USB$VOLA, UTX$VOLA, V$VOLA, VZ$VOLA, WBA$VOLA, WFC$VOLA, WMT$VOLA, XOM$VOLA))
#didn't work

VolAvg <- mean(c(AAPL$VOLA, ABBV$VOLA, ABT$VOLA, ACN$VOLA, ADBE$VOLA, AGN$VOLA, AIG$VOLA, ALL$VOLA, AMGN$VOLA, AMZN$VOLA, AXP$VOLA, BA$VOLA, BAC$VOLA, BIIB$VOLA, BK$VOLA, BKNG$VOLA, BLK$VOLA, BMY$VOLA, C$VOLA, CAT$VOLA, CELG$VOLA,
CHTR$VOLA, CL$VOLA, CMCSA$VOLA, COF$VOLA, COP$VOLA, COST$VOLA, CSCO$VOLA, CVS$VOLA, CVX$VOLA, DD$VOLA, DHR$VOLA, DIS$VOLA, DOW$VOLA, DUK$VOLA, EMR$VOLA, EXC$VOLA, F$VOLA, FB$VOLA, FDX$VOLA, GD$VOLA, GE$VOLA, GILD$VOLA, GM$VOLA,
GOOG$VOLA, GOOGL$VOLA, GS$VOLA, HD$VOLA, HON$VOLA, IBM$VOLA, INTC$VOLA, JNJ$VOLA, JPM$VOLA, KHC$VOLA, KMI$VOLA, KO$VOLA, LLY$VOLA, LMT$VOLA, LOW$VOLA, MA$VOLA, MCD$VOLA, MDLZ$VOLA, MDT$VOLA, MET$VOLA, MMM$VOLA, MO$VOLA, MRK$VOLA,
MS$VOLA, MSFT$VOLA, NEE$VOLA, NFLX$VOLA, NKE$VOLA, NVDA$VOLA, ORCL$VOLA, OXY$VOLA, PEP$VOLA, PFE$VOLA, PG$VOLA, PM$VOLA, PYPL$VOLA, QCOM$VOLA, RTN$VOLA, SBUX$VOLA, SLB$VOLA, SO$VOLA, SPG$VOLA, T$VOLA, TGT$VOLA, TXN$VOLA, UNH$VOLA,
UNP$VOLA, UPS$VOLA, USB$VOLA, UTX$VOLA, V$VOLA, VZ$VOLA, WBA$VOLA, WFC$VOLA, WMT$VOLA, XOM$VOLA))
```