```
#For the set of instructions to serve as a guide to the code below please visit the links below
# https://brockdsl.github.io/Text-Analysis-with-R/
# https://github.com/BrockDSL/Text-Analysis-with-R

#In depth dive into Text analysis: https://programminghistorian.org/en/lessons/basic-text-processing-in-r

#     NOTE that Rstudio Cloud was used : rstudio.cloud

#Install packages
install.packages("tidyverse")
install.packages("tokenizers")

# Activate packages
library(tokenizers)
library(tidyverse)

#Create a variable called "text"
text <- paste("You will rejoice to hear that no disaster has accompanied the commencement of an enterprise which
you have regarded with such evil forebodings. I arrived here yesterday, and my first task is to assure my dear
sister of my welfare and increasing confidence in the success of my undertaking")

#Create a variable containing the above text tokenized into words
words <- tokenize_words(text)

#Pulling out the list
words <- words[[1]]

#Making your table
tab <- table(words)

#Turn your list of words into a data frame
tab <- data_frame(word = names(tab), count = as.numeric(tab))

#Arrange your data frame so the most common words are listed first
tab <- arrange(tab,desc(count))

#View your results
tab

#Find out how long your list of words is using the length function
length(words)



#Tokenize the paragraph in the “text” variable into sentences and pull out just the list
sentences <- tokenize_sentences(text)
sentences <- sentences[[1]]

#Tokenize your sentences into lists of words
sen_words <- tokenize_words(sentences)

#Finding the length of each of the two sentences now made into 2 lists
length(sen_words[[1]])
length(sen_words[[2]])


#Use the “sapply” function to find the length of each list of words
sapply(sen_words, length)


# Now analyzing a book
#Load in the full text of the book “Frankenstein”
text <-
paste(readLines("https://raw.githubusercontent.com/BrockDSL/R_for_Text_Analysis/master/frankenstein.txt"),collapse
 = "\n")
text

#Tokenize the book into words
words <- tokenize_words(text)

#Pulling out the list
words <- words[[1]]
```

```
#Making it into table
tab <- table(words)

#Turn it into a dataframe arranged by count
tab <- data_frame(word = names(tab), count = as.numeric(tab))

tab <- arrange(tab,desc(count))

tab

#Load in the word frequency dataset
wordfreq <- read_csv("https://raw.githubusercontent.com/BrockDSL/R_for_Text_Analysis/master/wordfrequency.csv")

wordfreq

#Join the two datasets together to get frequency values for each word in the book
tab <- inner_join(tab,wordfreq)

tab

#Filter your results to remove the stopwords. (Try out different frequency values to see more or less common
words)
filter(tab,frequency<0.01)
#filter(tab,frequency<0.00001)


#We can make function to do this us as well
#Make a function that takes in a variable containing text and outputs a dataframe filtered to remove stopwords
# In order to run this function,remember you need to run your packages.

top_words <- function(fulltext){
  words <- tokenize_words(fulltext)
  words <- words[[1]]
  tab <- table(words)
  tab <- data_frame(word = names(tab), count = as.numeric(tab))
  tab <- arrange(tab,desc(count))
  wordfreq <-
read_csv("https://raw.githubusercontent.com/BrockDSL/R_for_Text_Analysis/master/wordfrequency.csv")
  tab <- inner_join(tab,wordfreq)
  return(filter(tab, frequency<0.01))
}

# Try out your new function by running on the text variable
top_words(text) # text is just a variable
```