**DS8006: Lab 2 “Twitter Basic Text Analysis”**
**Student’s name: NAJLIS, BERNARDO (#500744793)**

**1. Briefly explain what libraries your script uses and why? List main functions of each library included in your script.**

*The script uses the following libraries:*

| *Library* | *Description* | *Function Used* |
|---|---|---|
| ***twitteR*** | *R based Twitter client* | *setup_twitter_oauth (), userTimeline(), twListToDF()* |
| ***httr*** | *Dependency for twitteR* | |
| ***rjson*** | *Dependency for twitter* | |
| ***tm*** | *Text mining package* | |
| ***lubridate*** | *Dependency for twitteR* | |
| ***stringr*** | *String operations* | *perl()* |
| ***wordcloud*** | *Chart wordclouds* | *wordcloud()* |

**2. In your own words, explain what tdmCreator function does? How would you improve and in a what way?**

***tdmCreator()*** *transforms words into their stems and removes stop words. An idea for improvement is to add a parameter that allow for additional use-case specific stop words to be removed. These stop words would be specific to the context of the analysis being done.*

**3. Briefly explain how you completed each of the given tasks:**

**Task 1:**
*Added additional gsub() calls to the textScrubber() function, one per “noise” word.*

**Task 2:**
*Added an additional call to gsub() using regular expressions to remove words that are of length 1 or 2.*

**Task 3:**
*Added additional calls into textScrubber(). First replace ‘#’ into a reserved character, to keep it safe from all other scrubbing functions, then replace it back. In subsequent steps, I used a perl regular expression to remove all words that do not start with ‘#’ and then do some list manipulation to return an array of character strings with the hashtag terms.*

**Task 4:**
*Just by using the wordcloud() function over the scrubbed text. I chose not to use the stemmed version to keep core words for this context from being altered (like “Hillary” being stemmed into “Hillari”).*

**4. What was the most challenging part of this lab?**

*The most challenging part was in Task 3 to find a regular expression and other functions to keep only the hashtags.*