

MORE ON STRINGS  
IS TWITTER  
SUCCESSFUL?

# ANNOUNCEMENT


- HW 3 Due this Thursday!
- HW 3 Bonus question: comment
- Please come to office hours!

# SOCIAL IMPACT OF TWITTER



<http://techcrunch.com/2013/01/15/twitters-social-impact-cant-be-measured-but-its-the-pulse-of-the-planet/>



**Barack Obama** 

@BarackObama

 Follow

Four more years.

9:16 PM - 6 Nov 2012



770,825



338,422



**Donald J. Trump** ✓

@realDonaldTrump

 Follow

Remember, I am the only one who is self-funding my campaign. All of the other candidates are bought and paid for by special interests!

9:21 AM - 8 Mar 2016

  3,335  8,783



**Donald J. Trump**

@realDonaldTrump

Follow

Boycott all Apple products until such time as Apple gives cellphone info to authorities regarding radical Islamic terrorist couple from Cal

RETWEETS

**4,489**

LIKES

**10,991**



1:38 PM - 19 Feb 2016







**Donald J. Trump** ✓

@realDonaldTrump

 Follow

I use both iPhone & Samsung. If Apple doesn't give info to authorities on the terrorists I'll only be using Samsung until they give info.

2:32 PM - 19 Feb 2016

  3,699  9,786



**CNBC Now** 

@CNBCnow

 Follow

Sr. Apple exec, says Trump's call for Apple boycott puts the company in standing with other good people he has criticized - Reuters

3:53 PM - 19 Feb 2016

  379  432

Pay attention to the timeline...



# TWITTER REVOLUTIONS

- 2011 Egyptian revolution
- 2010–2011 Tunisian protests
- 2009–2010 Iranian election protests
- 2009 Moldova civil unrest



Anti-government protesters celebrate in Tahrir Square in downtown Cairo Friday. Fireworks burst, and Egypt exploded with joy and tears of relief after pro-democracy protesters brought down President Hosni Mubarak with a momentous march on his palaces and state TV. *Emilio Morenatti/AP*

IS TWITTER PROFITABLE?

## TWITTER 2016

- ❑ 2013, IPO \$44.90 to \$60
- ❑ Next facebook with 1 billion user with ads profit?
- ❑ Stalled at 300 million users in 2015
- ❑ 4 million new users recent quarter: \$22
- ❑ Fired CEO in June 2015
- ❑ Oct 2015, 300 layoffs
- ❑ Finally profitable the last quarter: \$7 million profit
- ❑ Not sustainable profit without user growth

Jan 2, 2016: <http://www.thenational.ae/business/technology/twitter-turns-a-profit-but-it-still-resists-change>

## IN COMPARISON...

- ❑ Facebook: 1.5 billion monthly active users; used by 70% of grownups online users in the USA
- ❑ Twitter: smaller than 25%, less than Instagram and Pinterest
- ❑ Media echo chamber: Media, Celebrities, Marketers
- ❑ No meaningful 2-way communication
- ❑ 1 billion signed-up & 300 million active: 70% left the service
- ❑ Worst kind of online commenting...
- ❑ Unfiltered, unorganized feed: not content rich

Jan 2, 2016: <http://www.thenational.ae/business/technology/twitter-turns-a-profit-but-it-still-resists-change>

IS TWITTER DYING?  
CLASS PROJECT



# IS TWITTER DYING?

- ❑ Edit button?
- ❑ 10K vs 140 character limit? (Jan, 2016)
- ❑ 1 billion registered – 250 million active

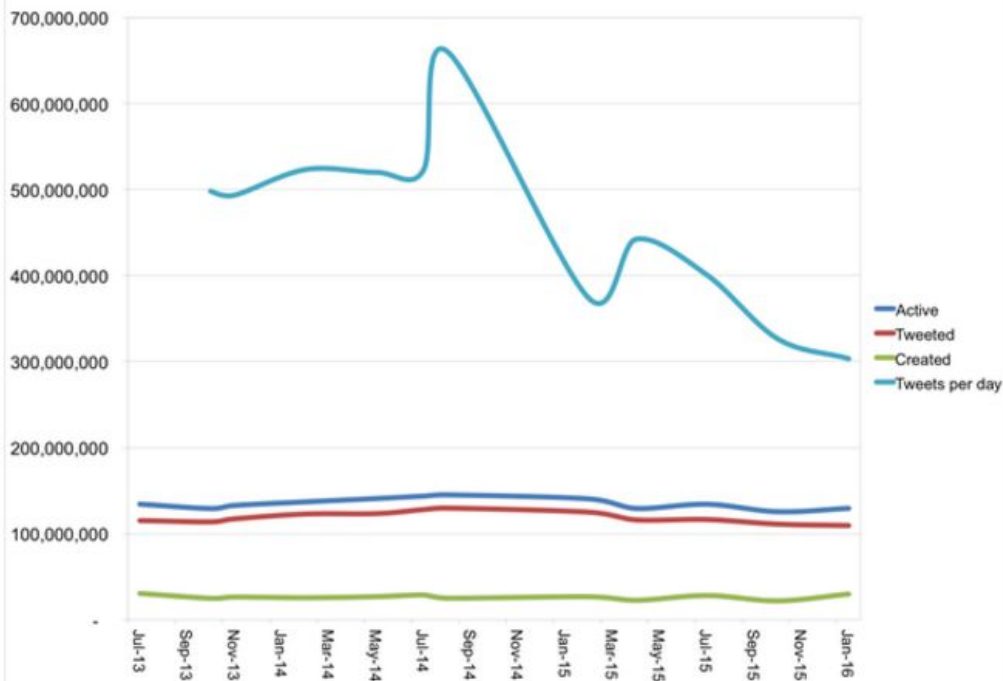
## #RIPTwitter

Feb 6, 2016 <http://www.newstatesman.com/science-tech/social-media/2016/02/why-twitter-dying-ten-tweets>

<http://www.newstatesman.com/sci-tech/2015/06/are-you-one-twitters-millions-ghost-users-could-be-why>



MAIN POINT:  
# OF TWEETS IN  
SERIOUS DECLINE



**Jim Edwards** ✓

@Jim\_Edwards

Follow

Leaked Twitter API data shows the number of tweets is in serious decline [read.bi/1KndiRx](http://read.bi/1KndiRx)

4:14 AM - 2 Feb 2016



159



92

# HOW TO KEEP CURRENT USERS ENGAGED/TWEETING?

For most people:

1. Twitter is too hard to use
2. Tweeting is scary: less confident users
3. Twitter feels lonely: talking into the winds

Core users in the Media Industry

Casual users: fast moving timeline, lack of curation

Messaging to engage less confident users?

# WHY TWITTER IS STILL ALIVE?

Competition, competition!

No replacement service yet

# STRINGS IN PYTHON

Lecture Notes modeled after: <http://www.openbookproject.net/thinkcs/python/english2e/ch07.html>

<http://www.pythontutor.com/index.html>

# Strings Manipulations: Creation, Accessing, length

```
my_word = "Hello World!"
```

```
his_word = "Let's have some fun with Python!"
```

```
print my_word
```

```
print his_word
```



Hello World!

Let's have some fun with Python!

```
my_word = "Hello World!"  
my_letter = my_word[0]
```

```
print my_word  
print my_letter
```

Hello World!

H

```
my_word = "Hello World!"
```

```
print len(my_word)
```

```
print my_word[2:5]
```

```
print len(my_word[2:5])
```

12  
110  
3

**Strings** are **immutable**: can't change a character in the string



```
greeting = "Hello"  
print greeting  
greeting[0] = 'w'      # ERROR!
```

Hello

TypeError: 'str' object does not support item assignment

```
greeting = "Hello"  
new_greeting = 'W'+ ' '+greeting  
print greeting  
print new_greeting
```

Hello  
W Hello

The **in** operator tests if one string is a **substring** of another.

```
print 'p' in 'apple'  
print 'i' in 'apple'  
print 'ap' in 'apple'  
print 'app' in 'apple'  
print 'ab' in 'apple'
```

```
print 'a' in 'a'  
print 'apple' in 'apple'
```

True  
False  
True  
True  
False  
True  
True



```
def remove_vowels(s):
    vowels = "aeiouAEIOU"
    s_without_vowels = ""
    for letter in s:
        if letter not in vowels:
            s_without_vowels += letter
    return s_without_vowels

user_input=raw_input('Please enter your word:')
print remove_vowels(user_input)
```

Please enter your word:banana  
bnn

#Eureka traversal: as soon as we find what we are looking for,  
#we can cry Eureka! and stop looking

```
def find(string, ch):  
    index = 0  
    while index < len(string):  
        if string[index] == ch:  
            return index  
        index += 1  
    return -1
```

```
user_input=raw_input('please enter your word:')  
print find(user_input, 'a')
```

please enter your word:banana

1

# Counting in String

```
fruit = "banana"  
count = 0  
for char in fruit:  
    if char == 'a':  
        count += 1  
print count
```

3

```
my_word = "Hello World!"
```

```
print my_word.count('l')
```

```
print my_word.count('o')
```



3

2

```
my_word = "Hello World!"
```

```
print my_word.find("H")
```

```
print my_word.index("World")
```

0  
6

# String Slicing: a review

Review: <http://www.pythonforbeginners.com/basics/string-manipulation-in-python>

```
my_word = "Hello World!"
```

```
print my_word
```

```
print my_word[0]
```

```
#get one char of the word
```

```
print my_word[0:1]
```

```
#get one char of the word (same as above)
```

```
print my_word[0:3]
```

```
#get the first three char
```

```
print my_word[:3]
```

```
#get the first three char
```

```
print my_word[-3:]
```

```
#get the last three char
```

```
print my_word[3:]
```

```
#get all but the three first char
```

```
print my_word[:-3]
```

```
#get all but the three last character
```

Hello World!

H

H

Hel

Hel

ld!

lo World!

Hello Wor

```
my_word = "Hello World!"
```

```
start = 1
```

```
end = 5
```

```
print my_word
```

```
print my_word[start:end]    # items start through end-1
```

```
print my_word[start:]      # items start through the rest of the list
```

```
print my_word[:end]        # items from the beginning through end-1
```

```
print my_word[:]           # a copy of the whole list
```

Hello World!

ello

ello World!

Hello

Hello World!



Want more string  
manipulations?

<http://www.pythonforbeginners.com/basics/string-manipulation-in-python>

```
my_word = "Hello World!"
```

```
# SPLIT
```

```
print my_word.split(' ') # Split on whitespace
```

```
# Startswith / Endswith
```

```
print my_word.startswith("H")
```

```
print my_word.endswith("!")
```

```
print my_word.endswith("d")
```

```
['Hello', 'World!']
```

```
True
```

```
True
```

```
False
```

```
my_word = "Hello World!"
```

```
# REPEATS
```

```
print my_word*3    # prints 3 times
```

```
print '.'*10    # print 10 dots
```

```
# REPLACING
```

```
print my_word.replace("Hello", "Hey")
```

Hello World!Hello World!Hello World!

.....

Hey World!

```
my_word = "Hello World!"
```

```
# CHANGE TO UPPER/LOWER CASE
```

```
print my_word.upper()
```

```
print my_word.lower()
```

```
print my_word.title()
```

```
print my_word.capitalize()
```

```
print my_word.swapcase()
```

HELLO WORLD!

hello world!

Hello World!

Hello world!

hELLO wORLD!

```
my_word = "Hello World!"
```

```
# REVERSING
```

```
print my_word[::-1]
```

```
begin = 1
```

```
end = 10
```

```
step = 2
```

```
print my_word[begin:end:step]
```

```
# TAKE HOME: figure out what the following mean by looking
```

```
# into Python manual or Googling
```

```
print ''.join(reversed(my_word))
```



!dlrow olleH

el ol

!dlrow olleH

```
# Strip off newline characters from end of the string
```

```
my_word = " Hello World! "
```

```
#strip() #removes from both ends
```

```
#lstrip() #removes leading characters (Left-strip)
```

```
#rstrip() #removes trailing characters (Right-strip)
```

```
print my_word
```

```
print my_word.strip()
```

```
print my_word.lstrip()
```

```
print my_word.rstrip()
```

Hello World!  
Hello World!  
Hello World!  
Hello World!

```
word = "Hello World"
```

```
print word.isalnum()    #check if all char are numbers  
print word.isalpha()   #check if all char in the string are alphabetic  
print word.isdigit()   #test if string contains digits  
print word.istitle()   #test if string contains title words  
print word.isupper()   #test if string contains upper case  
print word.islower()   #test if string contains lower case  
print word.isspace()   #test if string contains spaces  
print word.endswith('d') #test if string ends with a d  
print word.startswith('H') #test if string starts with H
```

False

False

False

True

False

False

False

True

True

COMING UP NEXT:  
TWITTER PROJECT  
INFORMATION  
SECURITY



**THANKS!**

---

**Any questions?**

You can find me at  
beiwang@sci.utah.edu

<http://www.sci.utah.edu/~beiwang/teaching/cs1060.html>

## CREDITS

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by [SlidesCarnival](#)
- Photographs by [Unsplash](#)