df.select \_dtypes(include=['float', 'int'])

import humanize

humanize.intcomma(12345)

'12,345'

or

n = 123123123

format(n, ",")

'123,123,123'

Be aware that `in` operator looks through index, not values!

False in pd.Series([True, True]).values

True as index contains 0 unless you append .values()

123 in pd.Series([123, 456]).values

False as index does not contain 123 unless you append .values()

if you want to check whether at least one series element is False or True:

(pd.Series([True, True]) == False).any()

False

text = "I love Mondays!"

text.ljust(30)

text.rjust(30, "\*")

http://text.center (30, "~")

animatplot - #python package for making interactive animated plots. Built on top of #matplotlib.

a, b = b,a works in one line due to tuple unpacking

To create a simple file sharing application go to your folder which you want to share and type

python -m http.server 8000

then go to your browser and type

127.0.0.1:8000

to open that folder in your browser

you can use this from other devices also if you are in the same network.

from collections import Counter

myList = [1,1,2,3,4,5,3,2,3,4,2,1,2,3]

print(Counter(myList)) try a|b or a&b

Counter({2: 4, 3: 4, 1: 3, 4: 2, 5: 1})

import itertools

itertools.permutations('ab')

[('a', 'b'), ('b', 'a')]

List, set and dict comprehensions

\_ calls the last executed expression

Python knows how to evaluate the truthfulness of all built-in data types. For example, an empty list evaluates to False, and a non-empty list evaluates to True.

print bool("five") # -> True

print bool({}) # -> False

print bool(0) # -> False

print bool(5) # -> True

print bool([]) # -> False

print bool([5]) # -> True

xrange generator is better than range static list

“stressed”[::-1]

In read\_csv, skiprows can be a fn

import termcolor

termcolor.cprint('Hey, this is important!!!', 'white', 'on\_red', attrs=['blink'])

# all letters: "new york".upper() 'NEW YORK'

# 1st letter: "new york".capitalize() 'New york'

# 1st letters: "new york".title() 'New York'

# 1st letters & smart: from titlecase import titlecase

titlecase('mchenry') 'McHenry'

Add leading zeros to your strings using zfill()

<http://www.fileformat.info/info/unicode/category/index.htm>

31 categories in 7 groups

import getpass # no echo

p = getpass.getpass("Answer to a very secret question:")

You can write #pandas objects directly to compressed format (gzip, bz2, zip, xz).

http://df.to \_csv("data.csv.gz", compression='gzip’)