

I LEARNED IT LAST NIGHT! EVERYTHING IS SO SIMPLE!

! HELLO WORLD IS JUST

print "Hello, world!"

I DUNNO... DYNAMIC TYPING? WHITESPACE?

COME JOIN US! PROGRAMMING IS FUN AGAIN! IT'S A WHOLE NEW WORLD UP HERE!

BUT HOW ARE YOU FLYING?

I JUST TYPED

import antigravity

THAT'S IT? !

... I ALSO SAMPLED EVERYTHING IN THE MEDICINE CABINET FOR COMPARISON.

! BUT I THINK THIS IS THE PYTHON.

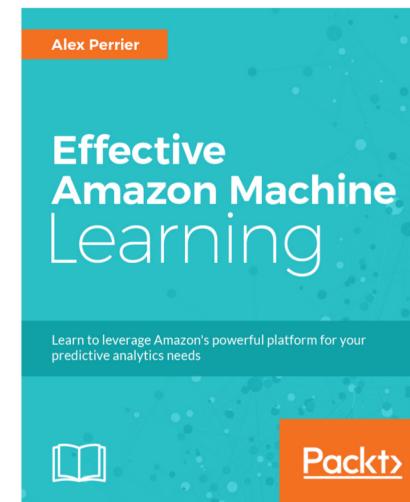
Présentations

Alexis Perrier

- Data Scientist
- 20+ ans de software

Connect:

- alexis.perrier@gmail.com
- twitter: @alexip
- www.linkedin.com/in/alexisperrier



But des 2 semaines

- Parler le python pour jongler avec les données
- Manipuler des données avec python

Programme des 2 semaines

- J1
 - Python
 - Installation
 - Notebook jupyter et éditeur
 - Variables et print
 - Import
 - Listes
 - String et concatenation
- J2
 - List comprehension
 - Fonctions et arguments
 - Dict
 - I/O fichiers
 - System: parcourir un répertoire
- J3
 - Pandas
 - File formats
 - Dataframes : subset, concat, merge, groupby, apply
- J4
 - API & web
 - Twitter API
 - http requests
 - wget
- J5
 - Projet: corpus twitter
- J6
 - NLP / TAL avec NLTK
- J7
 - Visualization
 - Web scraping avec BeautifulSoup
 - Git

Les datasets

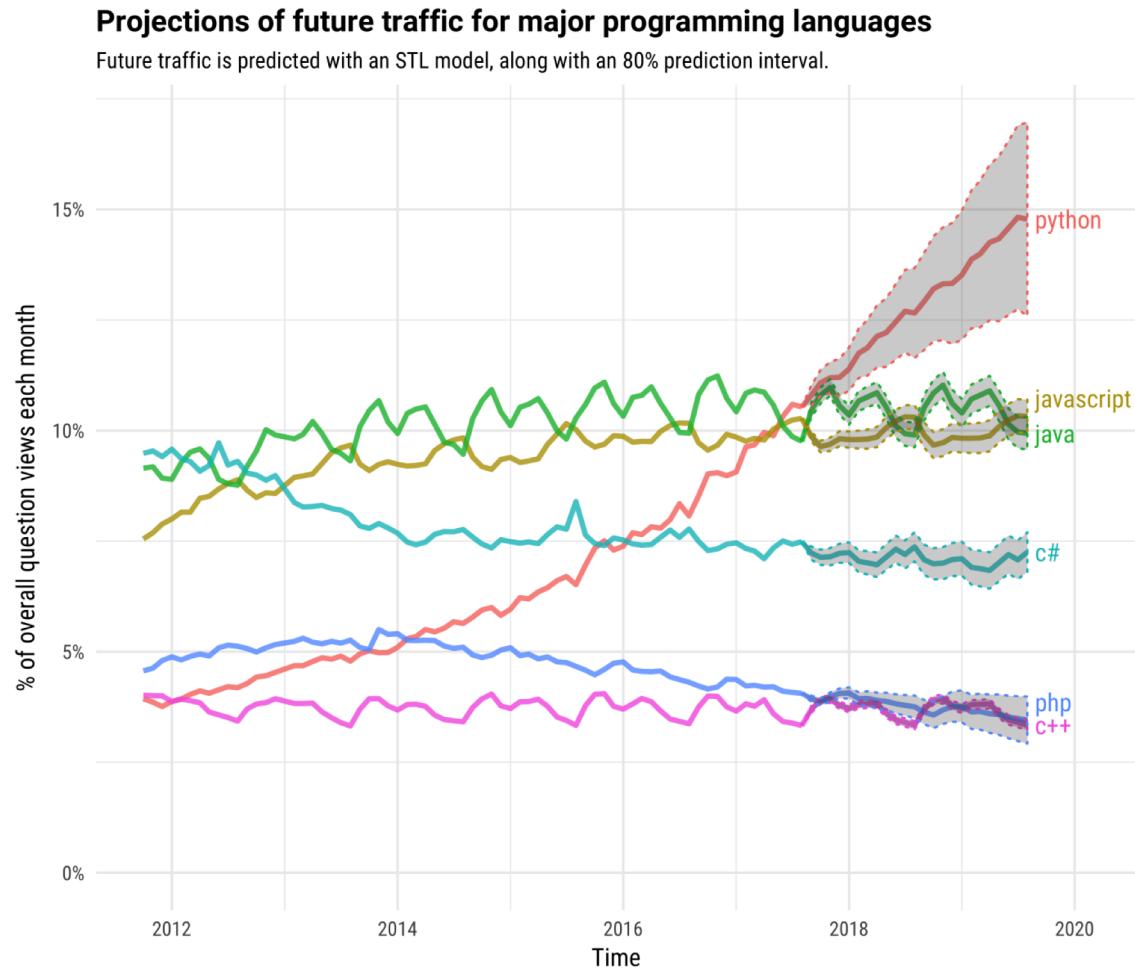
- Les arbres de paris (csv)
- Les news de la BBC (texte)
- Des corpus twitter (json)
- Assemblé nationale, les députés (csv)
- Fête de la musique 2018 (Json)
- Directives européennes (xml)

Python, python, python, pourquoi python ?

- Beaucoup d'applications: web, data science, scientific, ...
- Crée en 1991 par Guido von Rossum! 30 ans déjà!
- 130.000 packages et librairies
- Duck typing, pas de compilation, pas de ; ou de {}
- Indentation => le code est lisible
- Performances
- Mais il y a des surprises, des incohérences, des idioms, ...
- Python 2.7 ou python 3.6



Python, la croissance



<https://stackoverflow.blog/2017/09/06/incredible-growth-python/>

```
#!/usr/bin/python

import sys

for i in range(1,11):
    print i
    if i <= 5:
        print "AHHHHHHH"
    elif i >= 6 and i <= 9:
        print "WHERE ARE ALL THE BRACKETS??"
    else:
        print "HOW DO YOU PEOPLE READ THIS SYNTAX EASILY"
```

I wrote my first python program
today!

Jupyter - iPython

- <http://jupyter.org/>
- Executer du code dans le navigateur
- Multilingue
- Local ou cloud
 - \$ Jupyter notebook
 - AWS Sagemaker, Google datalab, Kaggle kernels
- A base de cellules
 - Documentation: **markdown et latex**
 - languages: python et R mais aussi julia, scala, ...
 - shell terminal
- Calcul et visualisation
- Partage et reproductibilité
- Seul soucis: le code est séquentiel et le séquencement des operations prete a confusion



Avez-vous python ?

- Dans un terminal :
 - `python --version`

```
[@:~]$ python --version
Python 3.6.2 :: Anaconda custom (x86_64)
[@:~]$ █
```

Installer Anaconda

- Distribution [Anaconda](#)
 - Jupyter notebooks et ipython
 - Dataframe : [pandas](#), dask
 - Dataviz : **matplotlib**, [plot.ly](#), [bokeh](#)
 - Texte : Gensim, NLTK, [Spacy.io](#)
 - Statistiques : **numpy**, **scipy**, **statsmodel**,
- Package manager **conda**
 - `conda install jupyter`
- Ou utiliser **pip**
 - `pip install jupyter`

Install Anaconda

- <https://www.anaconda.com/download/>
- Pour windows, linux ou macosx

Python 3.6 version *

 Download

[64-Bit Graphical Installer \(631 MB\)](#) 

[32-Bit Graphical Installer \(506 MB\)](#)

Python 2.7 version *

 Download

[64-Bit Graphical Installer \(564 MB\)](#) 

[32-Bit Graphical Installer \(443 MB\)](#)

Verifier l'installation



- `python --version`
- `ipython -version`
- `jupyter notebook`

Demo jupyter notebooks

Demo python - notebook jupyter

- les arbres de Paris
- <https://opendata.paris.fr/explore/dataset/les-arbres/>
- Dans mon terminal :
- Jupyter notebook

2 éditeurs fortement conseillés



<https://atom.io/>



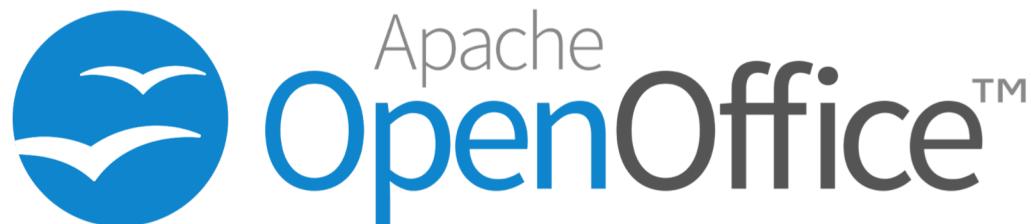
<https://www.sublimetext.com/>

Les bons outils font
les bons ouvriers



Lire des fichiers csv

- Bon outils



- A eviter



- Pas stable au niveau des séparateurs et des ""

Bases de python



LET'S DO THIS.

Variables

- Déclarer une variable

- a = "Hello"
- auteur = "Jacques Dutronc"
- x = 12.12344
- faux = False;
vrai = True

- Type de la variable

- `type(a)`

- String: "Bonjour", "Это приятно", "شمس"
- Float: 1.2, 3.14
- Int: 1,2,3,4, -1, -2
- Boolean: True, False

Comment nommer ses variables

- Bien :
 - nom_auteur = "V. Hugo"
 - nom_auteur_ = "V. Hugo"
 - nom_auteur_01 = "V. Hugo"
- pas bien :
 - NomAuteur = "V. Hugo"
 - lnom_auteur = "V. Hugo"
 - atr = "V. Hugo"
 - a:01 = "V. Hugo"
- Explicite
- En minuscule
- Mots séparés par un "_"



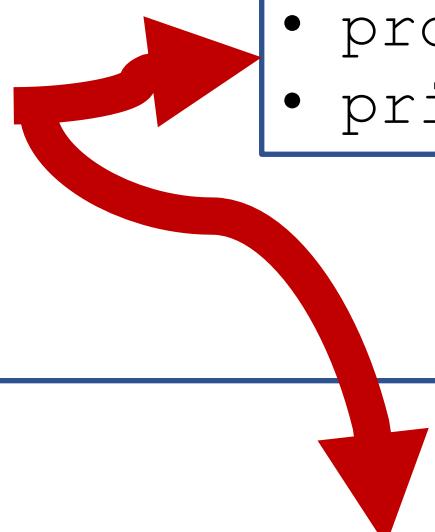
Printing

- `print("Hello world")`

Interpolation de strings

- `prof = "Bilel"`
- `print("Bonjour {}".format(prof))`

- `temp = 12`
- `nom = "Bilel"`
- `print("Bonjour {}, il fait {} degrés".format(nom, temp))`



A vous

- Print ("Hello world")
- Passer un nom en tant que variable pour afficher bonjour + le nom
- Afficher un texte avec 2 variables de types différents
 - Par ex: *Ce matin, j'ai mangé des {} et bu un {}*

Boolean et logique conditionnelle

- age_alexis = 38
- age_billel = 38
- age_alexis == age_billel

Qu'est ce que vous obtenez
pour chaque cas ?

- age_alexis = 38
- age_billel = "38"
- age_alexis == age_billel

- age_alexis = 38
- age_billel = "38"
- age_alexis == int(age_billel)

Boolean et logique conditionnelle

```
• if (age_alexis == age_billel) :  
    • print("c'est pas vrai")  
• elif(age_alexis < age_billel) :  
    • print("ça m'étonnerait!")  
• else:  
    • print("Ok mais ça me rajeunit pas tout ça")
```

=> Il n'y a pas de case ou switch en python

Question banco: que se passe t il si on compare un int a un str?

Boolean et logique conditionnelle

```
• if (age_alexis == age_billel) :  
    • print("c'est pas vrai")  
• elif(age_alexis < age_billel) :  
    • print("ça m'étonnerait!")  
• else:  
    • print("Ok mais ça me rajeunit pas tout ça")
```



Que se passe t il si on compare un int a un str?

A vous

- Etant donné 2 variables, un float et un string, écrire une série de conditions :
 - qui testent leurs égalités (==) et inégalités (<)
 - qui affiche un message dans les différents cas

Par exemple,

- meteo, prévision de pluie ou de soleil et prévision de température

Interlude

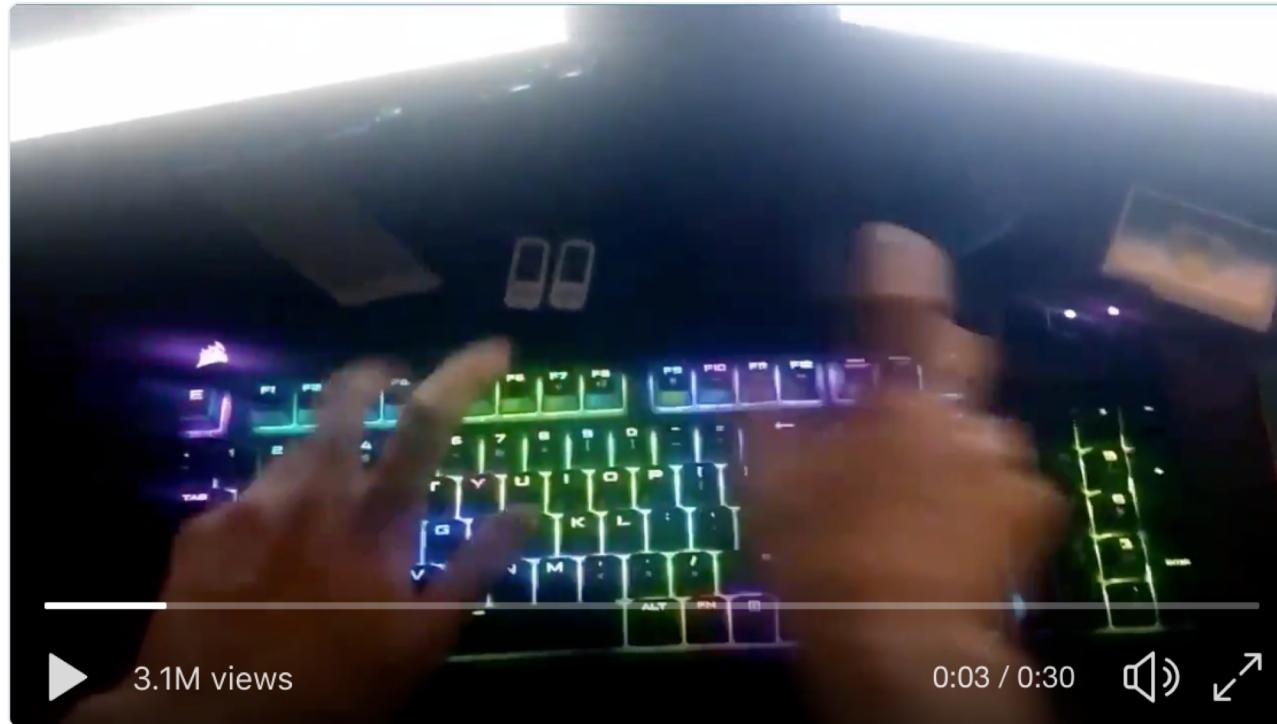


Dee Nadzree

@racoconn

Follow

what people think programming is vs. how it actually is



8:17 AM - 24 Feb 2018

93,136 Retweets 163,569 Likes

ND



1.7K

93K

164K



Stackoverflow: the real reason why google was invented!



python write to file



All

Videos

Books

Shopping

Images

More

Settings

Tools

About 224,000,000 results (0.40 seconds)

Python File Write - W3Schools

https://www.w3schools.com/python/python_file_write.asp ▾

Write to an Existing File. Example. Open the file "demofile.txt" and append content to the file: f = open("demofile.txt", "a") Example. Open the file "demofile.txt" and overwrite the content: f = open("demofile.txt", "w") Example. Create a file called "myfile.txt": Example. Create a new file if it does not exist:

Reading and Writing Files in Python - Python for Beginners

<https://www.pythonforbeginners.com/files/reading-and-writing-files-in-python> ▾

Jul 8, 2013 - The first thing you'll need to do is use Python's built-in open function to get a file object. The open function opens a file. It's simple. When you use the open function, it returns something called a file object.

python - Correct way to write line to file? - Stack Overflow

<https://stackoverflow.com/questions/6159900/correct-way-to-write-line-to-file> ▾

9 answers

Nov 13, 2017 - f = open('myfile', 'w') f.write('hi there\n') # python will convert \n to ... It is good practice to use the 'with' keyword when dealing with file objects.

[python: writing in a file](#)

4 answers Feb 20, 2018

[Python : write text to file line by line](#)

4 answers Nov 7, 2016

[Python write to file without overwriting current txt](#)

1 answer Mar 13, 2016

[How to redirect 'print' output to a file using python?](#)

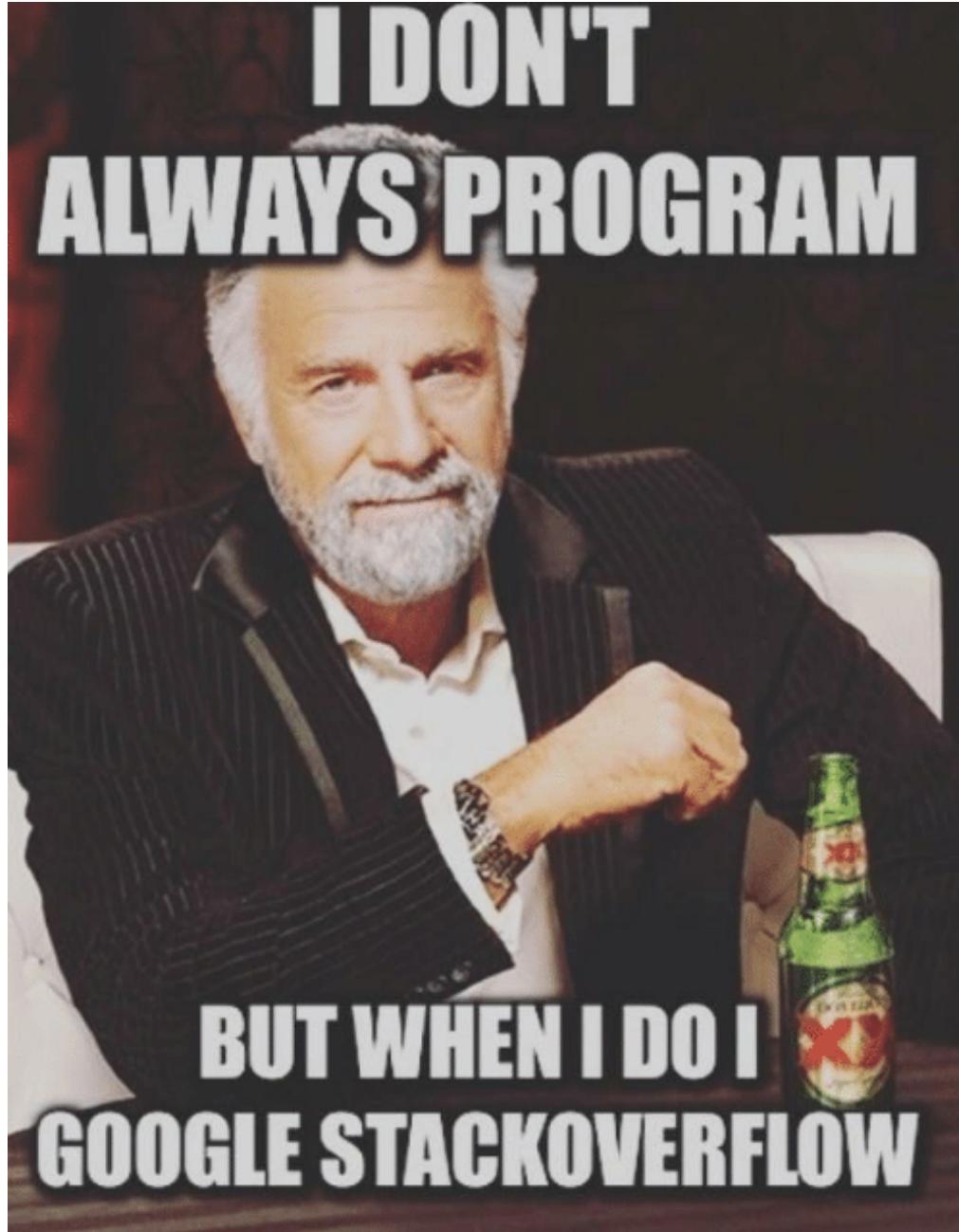
10 answers Aug 22, 2011

[More results from stackoverflow.com](#)

Reading and Writing Files in Python (article) - DataCamp

<https://www.datacamp.com/community/tutorials/reading-writing-files-python> ▾

Apr 11, 2018 - You can use this handle to read from or write to a file. Python treats the file as an object, which has its own attributes and methods.



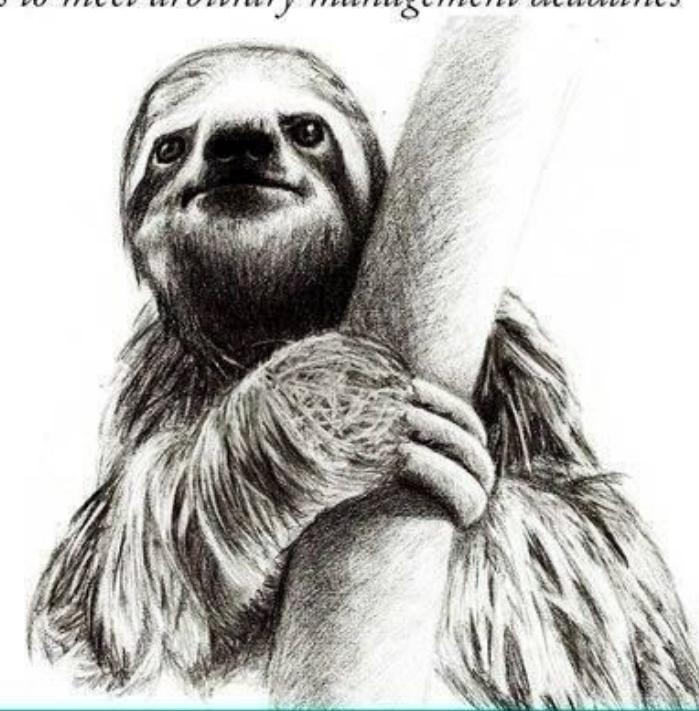
Cutting corners to meet arbitrary management deadlines

Essential

Copying and Pasting from Stack Overflow

O'REILLY®

The Practical Developer
@ThePracticalDev





Listes et for loop

- `equipe = ['Amelie', 'Jean', 'Bruno', 'John', 'Nathalie']`
 - `type(equipe)`
- Compter
 - Combien d'éléments: `len(equipe)`
- Afficher
 - `print(equipe)`
 - `for prenom in equipe :`
 - `print(prenom)`
 - `print(len(prenom))`
- **prenom est la variable dans la boucle**
- => Afficher prenom apres la boucle

Listes: indexation et slicing

- `equipe = ['Amelie', 'Jean', 'Bruno', 'John', 'Nathalie']`

- Premier element:
 - `print(equipe[?])`
 - L'indexation de la liste démarre à ????

Super
important!

- Plusieurs elements avec un *range*:
 - `print(equipe[1:3])`
- Que font les lignes suivantes ?
 - `print(equipe[:3])`
 - `print(equipe[:])`
 - `print(equipe[4:])`
 - `print(equipe[5])`
 - `print(equipe[-1])`
 - `print(equipe[-2:])`

Listes: les fonctions

- `equipe = ['Amelie', 'Jean', 'Bruno', 'John', 'Nathalie']`

- Functions applicable a une liste
 - `len(equipe)`
 - `sorted(equipe)`

In [12]: `equipe.`

```
append()  count()  insert()  reverse()
clear()   extend()  pop()    sort()
copy()    index()  remove()
```

A vous

- `equipe = ['Amelie', 'Jean', 'Bruno', 'John', 'Nathalie']`

- Tester les fonctions suivantes:

- `equipe.sort()`
- `equipe.count()`
- `equipe.reverse()`
- `equipe.index()`

- Quelle différence entre

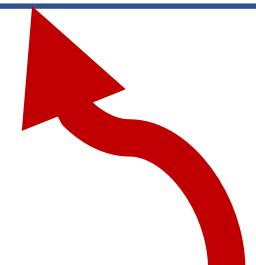
- `equipe.sort()`
- `sorted(equipe)`

- 2 façons de vider une liste ?

- pour que `equipe` devienne une liste vide `[]`

- Pour voir la doc d'une fonction, ajouter ?

- `?len`
- `?equipe.count()`



Super utile!



Comment tester qu'une liste soit vide?

Immutable types vs mutable type

```
In [32]: a = [1,2]
```

- Numbers, strings, tuples, frozensets, booleans, None, are immutable
- Lists, dictionaries, sets, bytearrays, are mutable

```
In [33]: b = a
```

```
In [34]: b
```

```
Out[34]: [1, 2]
```

```
In [35]: a[0] = 5
```

```
In [36]: b
```

```
Out[36]: [5, 2]
```



Pattern: transformer une liste

- Une liste
- Transformer les éléments de la liste et les mettre dans une autre liste
 - Compter la longueur des mots, supprimer la ponctuation
 - Mettre tous les mots en minuscule, capitaliser
 - Compter, mesurer, ...

```
• my_list = ['a', 'b', 'c', 'd', ...]
• new_list = []
• for elt in my_list:
    • resultat = fonction(elt)
    • new_list.append(resultat)
• print(new_list)
```

A vous

- legumes = ['tomate', 'salsifi', 'courgette', 'concombre', 'haricot vert', 'brocoli', 'oignon', 'petit pois']
- A partir de la liste legumes, construire une autre liste qui pour chaque légume
 - Compte le nombre de lettre
 - Remplace la lettre 'e' par la lettre 'u'
 - Mets en majuscule la premiere lettre
 - Supprime les legumes commençant par 'c' de la liste initiale

- Comment compter toutes les voyelles en utilisant 2 boucles imbriquées ?

Concaténation: + pour liste et string

- Ajouter 2 listes:
 - jeune = ['trap', 'rap']
 - vieux = ['disco', 'punk', 'ska', 'jazz', 'funk', 'rap', 'hip-hop', 'rock', 'soul']
 - musique = jeune + vieux
- Ajouter un élément à une string avec append
- Ajouter plusieurs listes pour former une nouvelle liste avec +

Concaténation: + pour liste et string

- Ajouter 2 strings:

- premier = 'il fait beau'
- second = 'et chaud'
- print(premier + second)
- print(premier + ' ' + second)

Problème au niveau de la jonction des 2 strings!
Il fait **beauet** chaud



- '<separator>' . **join**(list) ou separator = ',', '-',

- legumes = ['tomates', 'salsifis', 'courgettes',
'concombres', 'haricots vert', 'brocolis', 'oignons']

- Comment faire pour afficher:

- *J'aime les tomates, salsifis, courgettes, ..., brocolis, petits pois*

- Avec join:

- print("J'aime les {}".format(', '.join(legumes)))

Split: pour tokenizer

- `texte = "Mesdames, messieurs. Voie C. Attention au passage d'un train, éloignez vous de la bordure du quai s'il vous plaît."`
- Combien de mots ?
 - Utiliser `texte.split(' ')`

Mais comment enlever la ponctuation ?

Utiliser le module [string](#) et la fonction `punctuation`

```
import string  
string.punctuation
```

Wrap up

Resources

- Think python
- Python the hard way
- www.python-course.eu
- Datacamp