COURS DE *DATA SCIENCE*

1re Partie: Introduction à la science des données

Dr. Pegdwendé Nicolas Sawadogo

sawadogonicholas44@gmail.com

OBJECTIFS DU COURS

1. Vous donner un aperçu des enjeux du domaine de la science des données
   - Intelligence artificielle
   - Big data
2. Vous présenter les outils à connaitre pour être un *data scientist*
   - Anaconda, Spark, Hadoop,
   - Python, Map-reduce
3. Vous introduire aux techniques classiques de *machine learning*
   - Méthodes de classification et de regression
   - Clustering

PROGRAMME GÉNÉRAL DU COURS

1. Introduction à la science des données
   - Enjeux de la science des données: IA, Big Data
   - Techniques & technos pour les *data scientist: NoSQL, Hadoop, Spark, etc.*
2. Prétraitements et nettoyage des données
   - Sélection de variables, gestion des valeurs manquantes, etc.
   - Transformation des données, agrégations, recodage, etc.
3. Modèles d'apprentissage supervisé
4. Techniques d'apprentissage non supervisés
5. Visualisation de données

PROGRAMME GÉNÉRAL DU COURS

1. Introduction à la science des données
2. Prétraitements et nettoyage des données
3. Techniques d'apprentissage supervisé
   - Modèles de régression et de classification
   - Etapes d'application et d'évaluation
4. Techniques d'apprentissage non supervisés
   - Modèles de clustering
   - Application et évaluation d'un modèle de clustering
5. Visualisation de données
   - Conception d'un outil de data-viz avec Dash

DÉROULEMENT DU COURS

- Mardi 25 au vendredi 28
  - 8h - 10h30: cours théorique + présentation des TD
  - 11h – 13h: TD à préparer en autonomie
  - 13h30 – 15h30: correction des TD
- Cours du lundi 24 sauté: solution?
- Evaluation
  - Projet à réaliser en groupes
  - et/ou une session de QCM ?
- Les supports de cours vous seront transmis progressivement à la fin des séances.

PROGRAMME DE LA SÉANCE 1

1. Présentation du cours
2. Notions de *big data* et d'IA
3. Notion de *data science*
4. Rappels sur Python

NOTIONS DE BIG DATA ET D'IA | SECTION 2

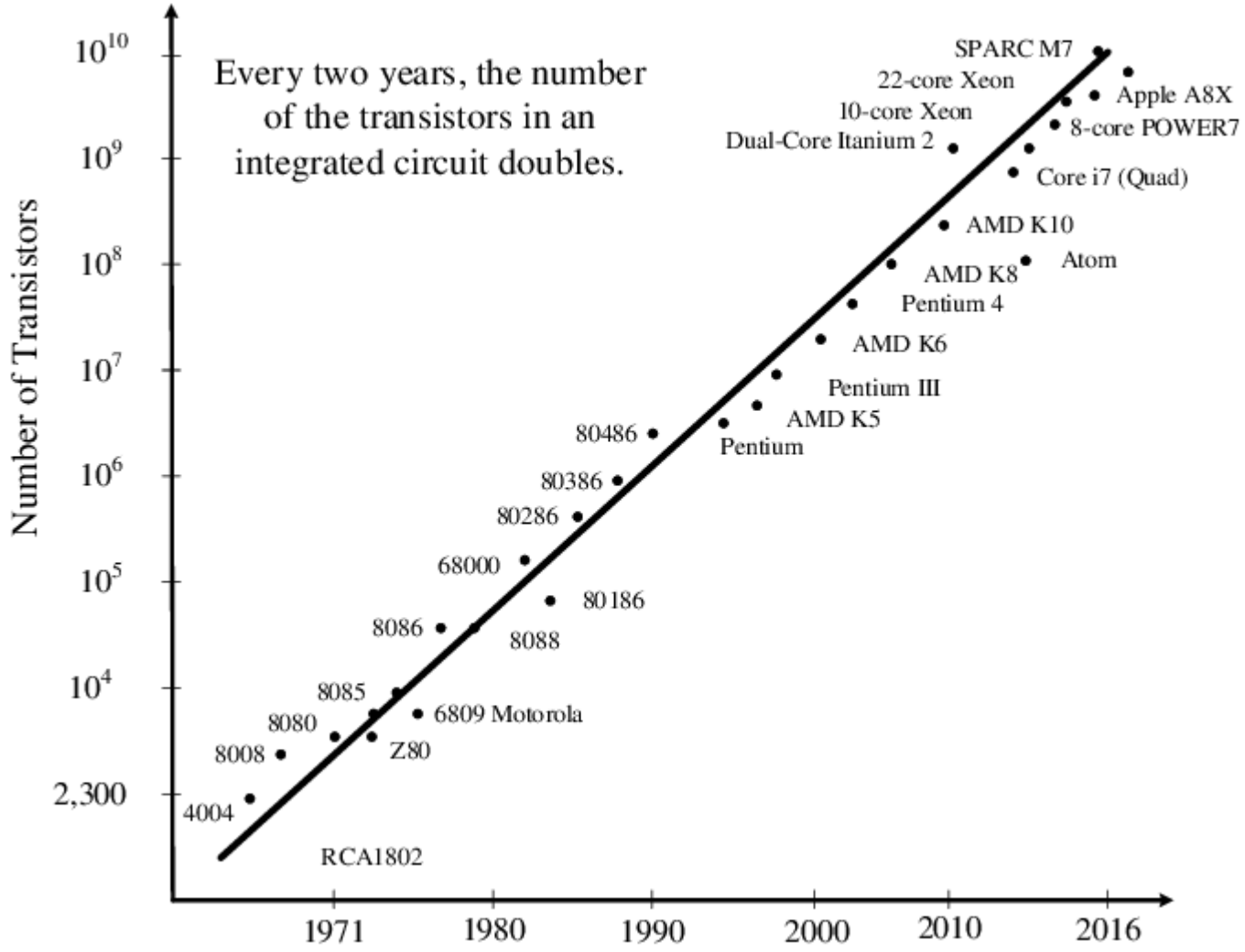
LOI DE MOORE

- Emise en 1965 par Gordon E. Moore (puis réajustée 10 ans plus tard)

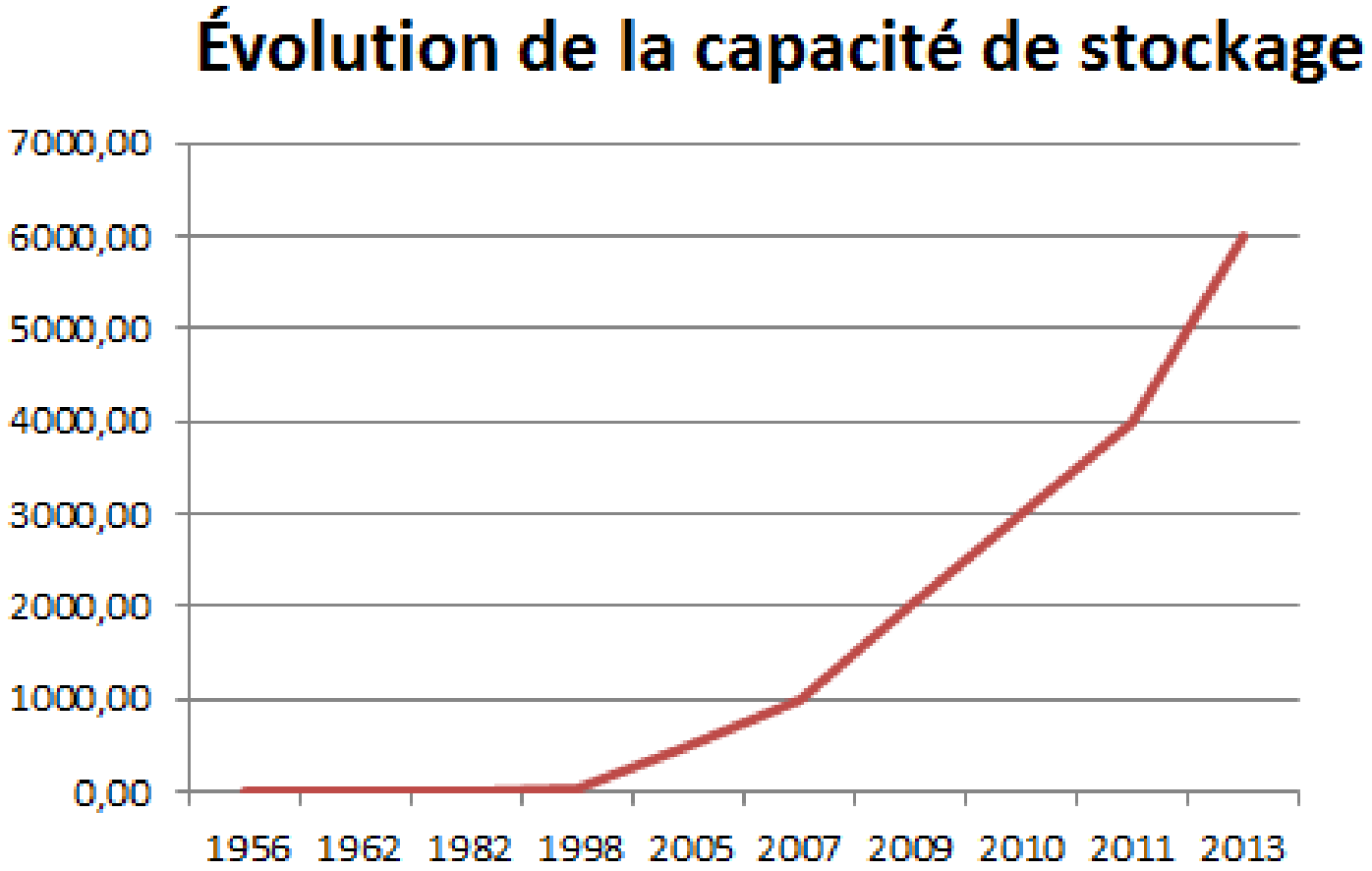
- Elle prédit le doublement du nombre de transitors présents sur une puce de microprocesseur tous les deux ans.

- Cette loi s'est finalement généralisé à un doublement des capacités informatiques quelconques

LOI DE MOORE : CPU



LOI DE MOORE : STOCKAGE



VERS UN MONDE DE « TOUT STOCKER »

Le phénomène observé par la loi de Moore va entrainer une prépondérance de l'informatique

- Digitalisation croissante des entreprises
  - L'évolution des capacités de stockage entraine une basse des coûts:
  - Les entreprises vont se digitaliser davantage, et stocker plus
- Développement de l'internet
  - En 1989, le WWW est inventé
  - Emergence des réseaux sociaux (Facebook en 2004, Twitter et Youtube en 2006, etc).
  - internet des objets: capteurs

VERS UN MONDE DE « TOUT STOCKER »

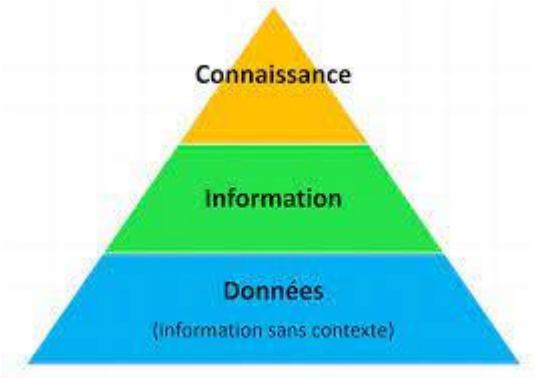
Le phénomène observé par la loi de Moore va entrainer une prépondérance de l'informatique

- Digitalisation croissante des entreprises
  - L'évolution des capacités de stockage entraine une basse des coûts:
  - Les entreprises vont se digitaliser davantage, et stocker plus
- Développement de l'internet
  - En 1989, le WWW est inventé
  - Emergence des réseaux sociaux (Facebook en 2004, Twitter et Youtube en 2006, etc).
  - internet des objets: capteurs

VERS UN MONDE DE « TOUT STOCKER »

Se posent alors de nouveaux problèmes

- Comment stocker toutes ces données ?
  - Il faut stocker intelligemment, pour retrouver facilement
  - Problème des données complexes: images, textes, vidéos
- Comment analyser toutes ces données ?
  - Les données sont la matière première de la connaissance
  - Comment raffiner les données pour en tirer des informations potables ?



NOTION DE BIG DATA : DÉFINITION

Ces données qui engendrent tant de problèmes sont connus sous la notion de *big data*

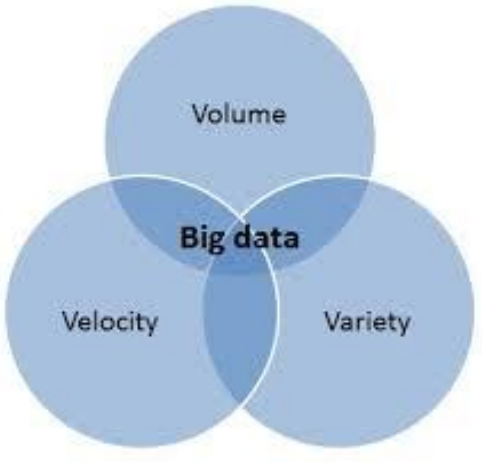
Définition

Les *big data*, ou mégadonnées (ou encore données massives), désignent des ensembles de **données** si **complexes** qu'elles **surpassent** les **capacités** des **outils informatiques classiques** pour leur traitement et leur exploitation.

NOTION DE BIG DATA: CARACTÉRISTIQUES

Les *big data* sont souvent identifiées par des caractéristiques en V

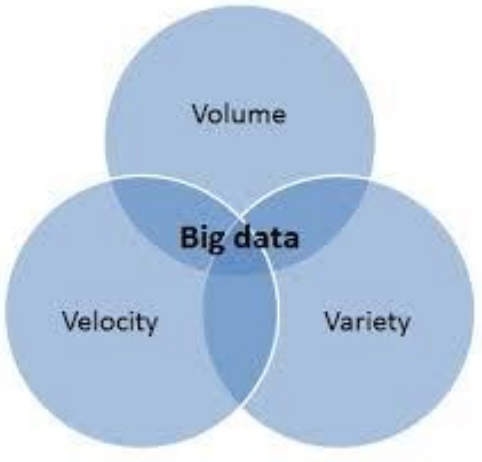
- Volume:
  - Chaque seconde, 253k textos sont échangés,
  - 18k vidéos sont visionnées sur Youtube,
  - 60k requêtes sont lancées sur Google,



NOTION DE BIG DATA: CARACTÉRISTIQUES

Les *big data* sont souvent identifiées par des caractéristiques en V

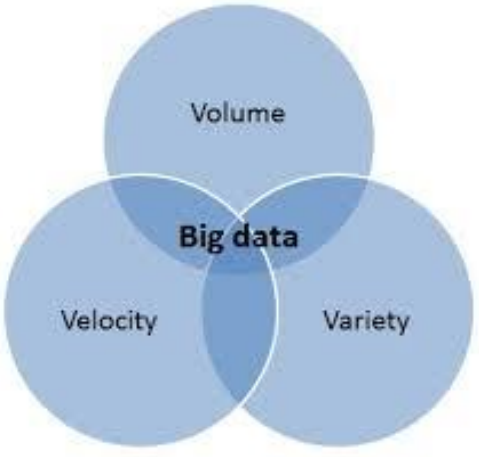
- Vélocité:
  - Données en flux, quasiment en temps réel
  - Données des réseaux sociaux, des capteurs
  - Vidéos surveillance, etc.



NOTION DE BIG DATA: CARACTÉRISTIQUES

Les *big data* sont souvent identifiées par des caractéristiques en V

- Variété:
  - Données structurées, semi-structurées et non-structurées
  - Seulement 20% de données structurées



SOLUTIONS POUR LES BIG DATA

- Outils/techniques de stockage adaptées
  - Entrepôts de données
  - SGBD NoSQL pour stocker les données complexes
  - Hadoop HDFS pour le stockage distribué
  - Solutions d'indexation
- Solutions de traitement
  - Informatique décisionnelle (*business intelligence)*
  - Intelligence artificielle: reconnaissance d'image, traitements du langage naturel, prédictions, systèmes de recommandations, détection de communautés, etc.

NOTION D'INTELLIGENCE ARTIFICIELLE

Définition

L'intelligence artificielle désigne un ensemble de techniques mises en œuvre en vue de conférer à une machine des capacités similaires à l'intelligence humaine

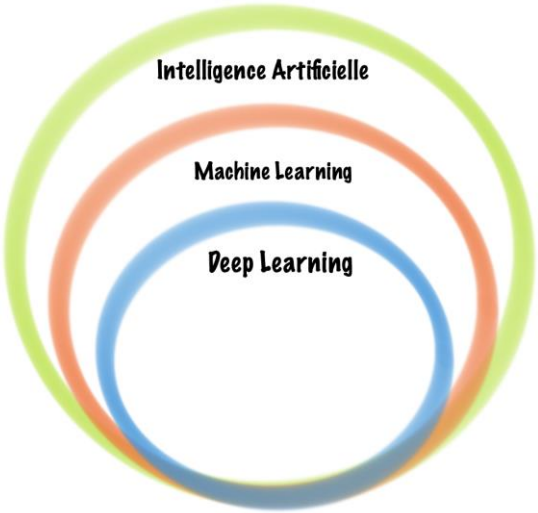
BRANCHES DE L'INTELLIGENCE ARTIFICIELLE

- Apprentissage statistique (machine learning)
- Traitement du langage naturel
- Systèmes de recommandation
- Systèmes experts et de raisonnement
- Reconnaissance d'images
- Recherche inversée, détection de motifs

NOTION DE MACHINE LEARNING

Définition

Le *machine learning* désigne un ensemble de techniques d'intelligence artificielle, qui se fondent sur des approches mathématiques et statistiques pour donner aux ordinateurs la capacité d'apprendre à partir des données.



NOTION DE MACHINE LEARNING

- Le *machine learning* inclue plusieurs disciplines:
  - Les statistiques et les probabilités
  - Les techniques d'optimisation
  - L'informatique

- Plusieurs domaines d'application:
  - Reconnaissance de motifs/d'images
  - Traitement automatique du langage/text mining
  - Systèmes de recommandation

N° de la question : 633

PETIT POINT D'ETAPE…

Parmi les options suivantes, lesquels relèvent du big data ?

A. Les statistiques collectées par les sondages ou par recensement, ou produites par l'INSD

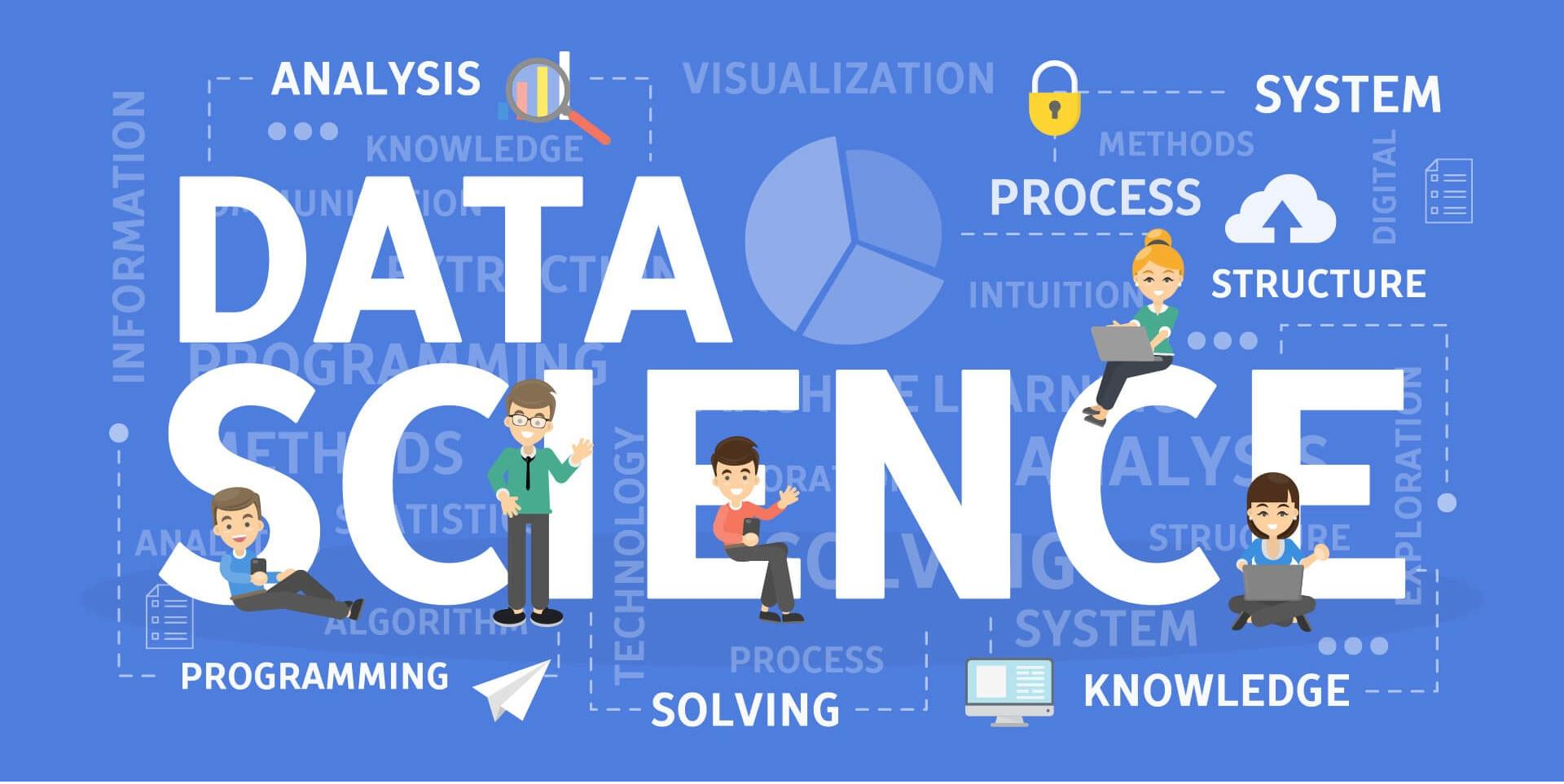
B. Les données de localisation GSM d'un opérateur de téléphonie sur une journée

C. Les rapports de stages et mémoires (s'ils étaient) conservés en version électronique par la bibliothèque centrale de l'UJKZ

https://toreply.univ-lille.fr/

PROGRAMME DE LA SÉANCE 1

1. Présentation du cours
2. Notions de big data et d'IA
3. Notion de *data science*
4. Rappels sur Python



NOTION DE DATA SCIENCE | SECTION 3

METIER DE DATA SCIENTIST

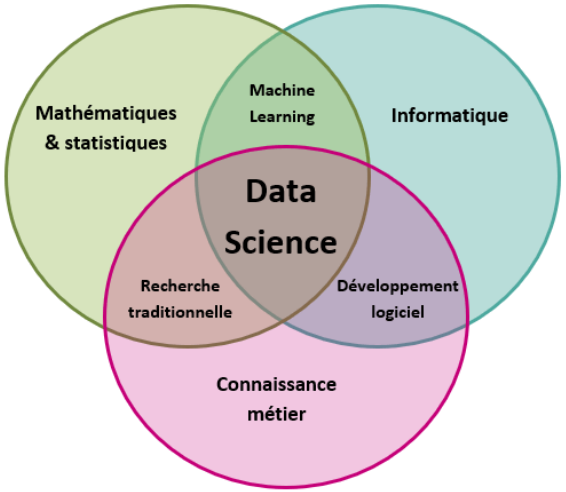
- Avec l'émergence des sources de données et et la démocratisation des techniques d'IA
  - des modèles prédictifs, basés sur les données sont adoptés
  - les modèles faits de règles métiers sont abandonnés

- Les compétences nécessaires à la mise en œuvre de ces nouveaux modèles étant:
  - de l'intelligence artificielle
  - de l'analyse statistique
  - des notions métier (marketing)
  - des notions informatiques
- Le métier de data scientist est né.

METIER DE DATA SCIENTIST

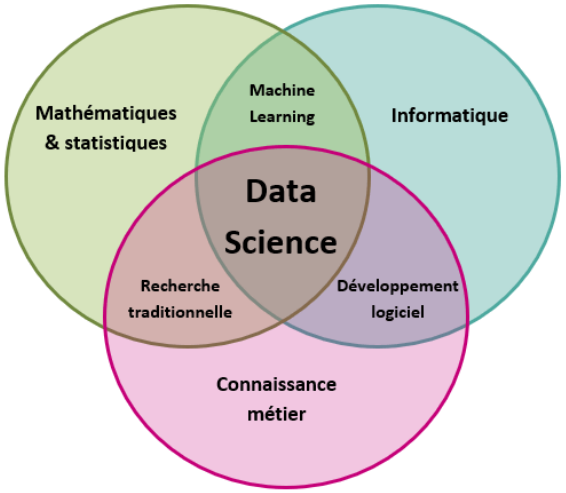
- Le data scientist est par définition un généraliste, qui touche à plusieurs disciplines

- Une définition très courante :
  - *I think of data scientists as knowing more about statistics than computer scientists and more about computer science than statisticians.*
  - Autrement dit, "informatique + statistiques" = *data science*
  - Mais l'aspect métier est également important



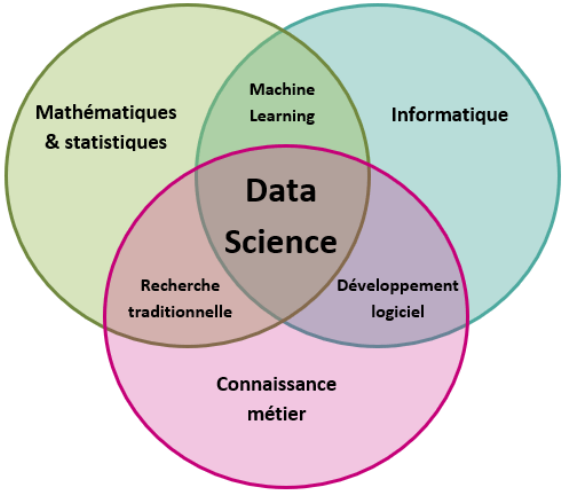
METIER DE DATA SCIENTIST

- Dimension mathématique/statistiques
  - Besoin de notions statistiques: tests statistiques, p-value, niveau de signification, probabilités, corrélations, …
  - Maîtrise des algo de prédiction et de clustering, leurs avantages et limites, interprétation des résultats, …



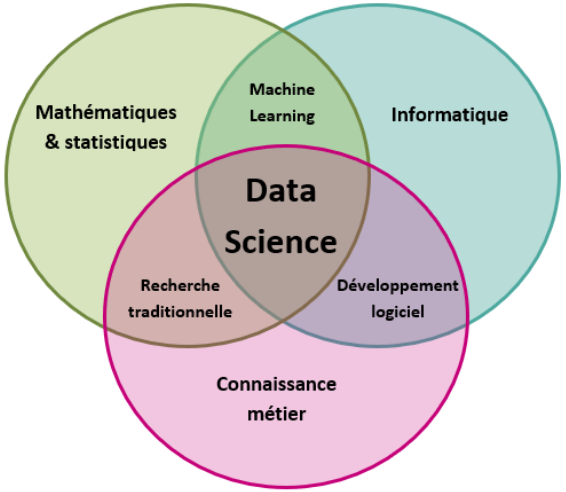
METIER DE DATA SCIENTIST

- Dimension informatique
  - Besoin d'interagir avec des bases de données, de rechercher/scrapper des données complémentaires, …
  - Besoin de programmer les algos de machine learning de façon efficace, en assurant le passage à l'échelle.
  - Besoin de créer des visualisations sur les données

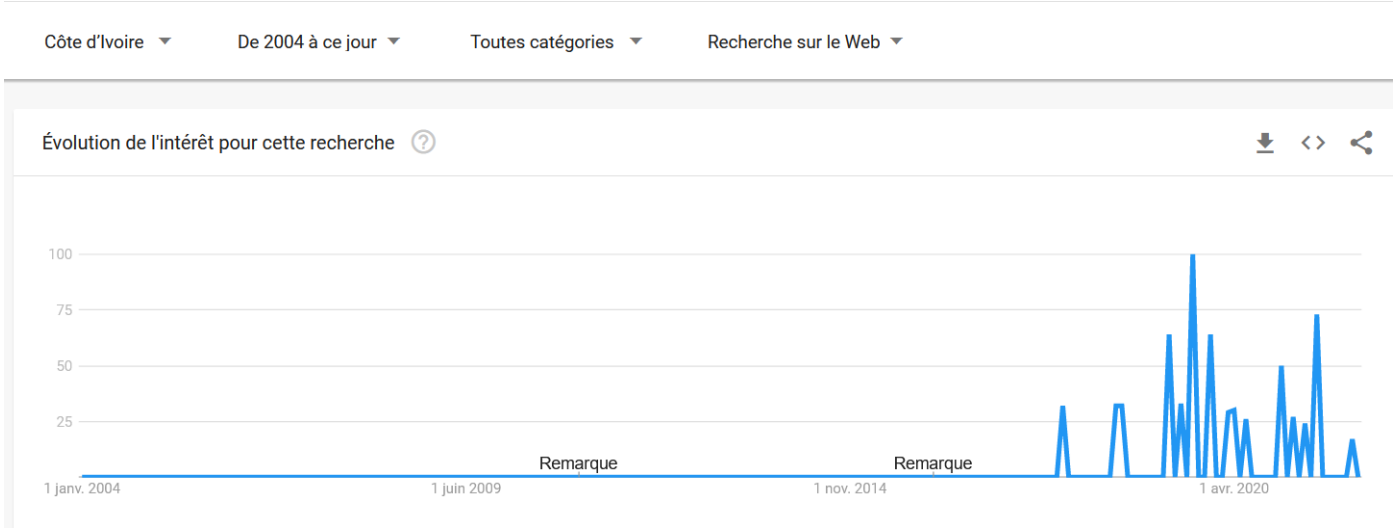


METIER DE DATA SCIENTIST

- Dimension métier
  - Besoin de connaitre le sens que portent les données, pour anticiper des besoins ou problèmes d'analyse.
  - Besoin de comprendre les échelles utilisées et les subtilités, connaître les règles métiers.
  - Être un bon communiquant pour présenter les résultats



METIER DE DATA SCIENTIST



TRAVAIL DU DATA SCIENTIST

1. Recueil du besoin métier
   - Il faut traduire le besoin exprimé par les équipes métiers en besoin d'analyse
   - Identifier les variables explicatives et la variable cible
   - Connaitre les critères de performances attendus
2. Collecte des données
   - Identifier les données pertinentes pour l'analyse
   - Trouver les données internes ou externes, les compléter
   - S'assurer des droits sur les données
3. Préparation des données
   - Nettoyer les données, gérer les valeurs aberrantes ou manquantes
   - Harmoniser les unités
   - Structurer les données non/mal structurées

TRAVAIL DU DATA SCIENTIST

4. Modélisation
   - Identifier le modèle approprié au type de données et au besoin d'analyse
   - Comparer plusieurs modèles en termes de performances, d'interpretabilité, de rapidité, etc.
5. Visualisation
   - Avoir connaissance des caractéristiques des données, des variables
   - Pouvoir présenter du concret aux métiers
6. Optimisation
   - Améliorer la qualité avec par exemple une sélection de variables.
   - Assurer le passage à l'échelle
7. Déploiement
   - Mettre en production le modèle proposé

PETIT POINT D'ETAPE…

La data science est plus proche des statistiques que de l'informatique:

A. Vrai

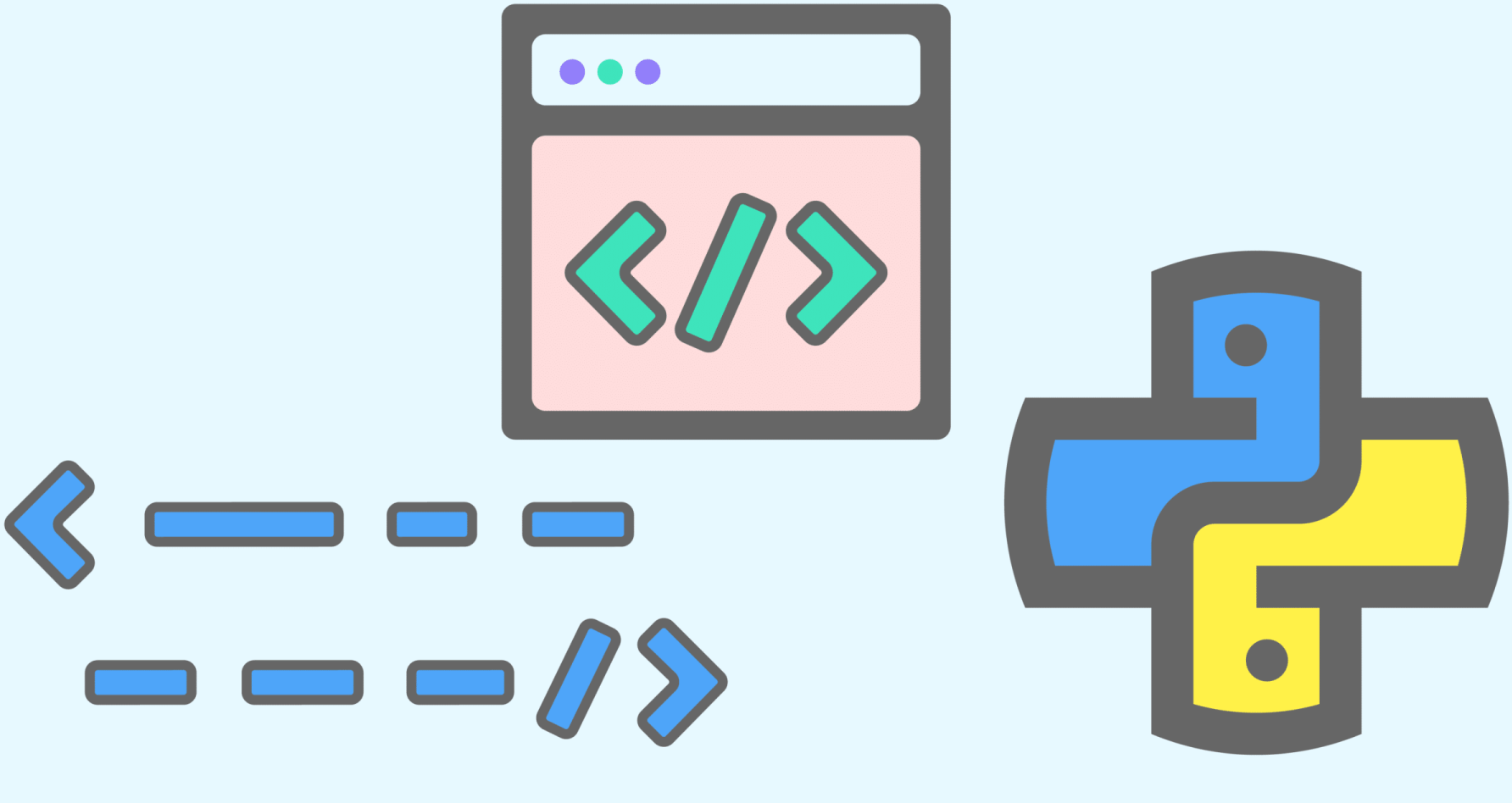
B. Faux

N° de la question : 491

https://toreply.univ-lille.fr/

PROGRAMME DE LA SÉANCE 1

1. Présentation du cours
2. Notions de big data et d'IA
3. Notion de data science
4. Rappels sur Python



RAPPELS SUR PYTHON | SECTION 4

POURQUOI PYTHON ?

- Python est à ce jour le 3è langage le plus populaire
  - Derrière Java et le langage C
- Un des principaux langages de data science
  - Avec R, Scala et Java
- Il offre de nombreuses bibliothèques
  - Nltk, gensim
  - Scikit-learn, pandas, numpy
- Utilisé dans plusieurs domaines
  - Nettoyage des données, dtatistiques descriptives
  - Clustering
  - Visualisation
  - Programmation web

PRÉSENTATION DE PYTHON

- Python est un langage créé en 1991
  - Open source
  - Langage de haut-niveau
- Python est multi-plateformes
  - Utilisable sur Mac, Linux ou Windows
  - Fonctionne grâce à un interpréteur de commandes

TYPES DE DONNÉES DE BASE SOUS PYTHON

- Données numériques (nombreuses)
  - Entiers (int):
    1, 5, 25, 1542, …
  - Flottants (float):
    2.5, 11.0, $3.25^{e}7$
- Chaines de caractères (str)
  - « Bonjour », « lundi », « 1 », …
- Booléens (bool)
  - True/False

Sous Python, le typage est implicite. Le type d'une variable est automatiquement détécté

OPÉRATEURS DE BASE SOUS PYTHON

symbole	types	exemples
+	entier, réel chaîne de caractères	6+4 == 10 "a" + "b"== "ab"
-	entier, réel	6-4 == 2
*	entier réel chaîne de caractères	6*4 == 24 1.2 * 1 == 1.2 3 * "s" == "sss"
**	entier, réel	12**2 == 144
/	entier réel	6/4 == 1 (*) 6./4 == 1.5
//	entier, réel	6//4 == 1
%	entier, réel	6%4 == 2

OPERATEURS DE BASE SOUS PYTHON

- Opérateur d'affectation
  - « = », équivaut à ← sous R
- Opérateurs logiques
  - or: OU logique
  - and: ET logique
  - not: NON logique
- Opérateurs de comparaison
  - <, > : Strictement inférieur, strictement supérieur
  - <=, >=: Inférieur ou égal, supérieur ou égal
  - ==, != : Egal, différent

TYPES DE DONNÉES COMPLEXES: CHAINES DE CARACTÈRES

- Elles expriment des informations alphanumériques et se définissent par des cottes simples ou doubles

- Opérations sur les chaines:

```
"5"*6                          # 555555
"data" + "science"             # "datascience"
"mercredi"[4:8]                # "redi"
"mercredi".upper()             # "MERCREDI"
```

TYPES DE DONNÉES COMPLEXES: LISTES

- Une liste est une collection d'objets qui peuvent être de tous types
- Opérations de base sur les listes:

  - Création d'une liste avec des elts:

    ```
    liste1 = [7,"deux"]; liste2 = ["un",1,"deux",2]; ...
    ```

  - Création d'une liste vide:

    ```
    liste3 = []; liste4 = list()
    ```

  - Accès aux elts d'une liste:

    ```
    liste1[0] #7; liste2[-1] #2;  liste2[-3] #1
    ```

- Opérations avancées sur les listes

  - Ajout d'éléments:

    ```
    liste1.append("trois") # [7,"deux","trois"]
    liste1.extend(["quatre","cinq"]) #[7,"deux","trois", ...
    ```

  - Tri par ordre croissant:

    ```
    liste.sort()
    ```

  - Taille, Minimum, Maximum d'une liste:

    ```
    len(liste), min(liste), max(liste)
    ```

TYPES DE DONNÉES COMPLEXES: DICTIONNAIRES

- Un dictionnaire est une liste de couples clé-valeur. Chaque clé est unique
- Opérations de base sur les dicos:

  - Création d'un dico avec des elts:

    ```
    dico1 = {"BIBD":24, "SISE":26}
    ```

  - Création d'un dico vide:

    ```
    dico2 = {}; dico3 = dict()
    ```

  - Accès aux elts d'un dico:

    ```
    dico1["BIBD"] ou dico1.get("BIBD") #24
    ```

- Opérations avancées sur les dicos

  - Obtenir toutes les clés / toutes les valeurs:

    ```
    dico.keys() / dico.values()
    ```

  - Taille, plus petite clé, plus grande clé:

    ```
    len(dico), min(dico), max(dico)
    ```

STRUCTURES DE CONTRÔLE: CONDITION

L'exécution conditionnelle permet d'instruire le programme d'exécuter une suite d'instruction ou une autre selon les circonstances

- if simple

```
if age >= 18:
    print("majeur")
```

- if - else

```
if age >= 18:
    print("majeur")
else:
    print("mineur")
```

- if - elif - else

```
if age < 10:
    print("enfant")
elif age < 20:
    print("ado")
else:
    print("adulte")
```

STRUCTURES DE CONTRÔLE: RÉPÉTITIONS

Les boucles permettent de répéter plusieurs fois un ensemble d'instructions. Il faut toujours une condition d'arrêt

- Boucle while: exécute tant que…

```
i = 0
while i < 5:
    i = i + 1
    print(i)
```

- Boucle for: parcourt une liste

```
for jour in ["lundi", "mardi", "mercredi"]:
    print(jour) #lundi, mardi, ...
```

FONCTIONS: CRÉATION

- Une fonction se crée via le mot clé « def »

```
def addition(x, y):
        return x + y

def multiplication(x, y):
        return x * y
```

- Les arguments peuvent être optionnels

```
def operation(x=0, y=10):
        return x + y

def operation(x=1, y):
        return x * y
```

FONCTIONS: UTILISATION

- Utiliser l'ordre des paramètres

```
def puissance(x, y)
    return x**y

puissance(2, 3) #8
puissance(3, 2) #9
```

- Utiliser les noms des paramètres

```
def puissance(x=1, y=1)
    return x**y

puissance(y=3) #1
puissance(x=2) #2
puissance(y=3, x=2) #8
```

MODULES

- Un module est un fichier qui regroupe un ensemble de fonctions
  - Le module est créé sous la forme d'un fichier .py
  - Le nom du fichier est par défaut celui du module
  - On peut charger le module complet, ou des éléments individuels.

- Création de module

```
(fichier operations.py)

    def add(x, y)
        return x+y

    def mult(x, y)
        return x*y
```

- Chargement et utilisation

```
import operations
operations.add(5,3)     #8
operations.mult(5,3)    #15

from operations import add
add(5,3)    #8
```

FICHIERS

Python permet d'interagir avec des fichiers stockés dans le filesystem

- Ecriture dans un fichier

```
fichier = open("sauvegarde.txt", "w")
fichier.write(texte)
fichier.close()
```

- Lecture depuis un fichier

```
fichier = open("sauvegarde.txt", "r")
texte = fichier.read()
fichier.close()
```

PETIT POINT D'ETAPE…

QUELQUES RESSOURCES

- https://tutorial.djangogirls.org/fr/python_introduction/
- https://courspython.com/introduction-python.html
- http://eric.univ-lyon2.fr/~ricco/cours/slides/PA%20-%20intro%20python%20-%20bases%20algorithmiques.pdf
- https://perso.limsi.fr/pointal/_media/python:cours:courspython3.pdf
- https://www.datacamp.com/learn-python-with-anaconda