



# **SAE 15 :** *Traitement de données*

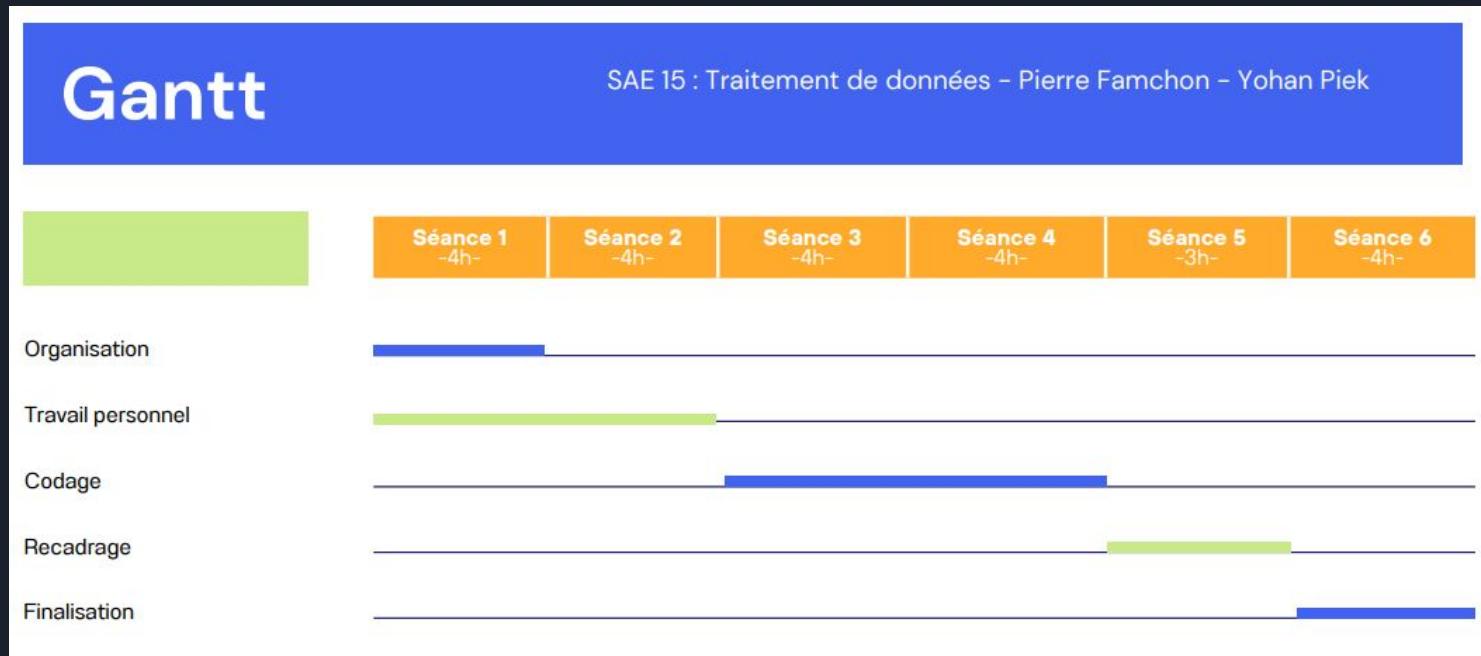
Pierre Famchon - Yohan Piek



## **Sommaire :**

- 1. Gant*
- 2. Matrice RACI*
- 3. Présentation*
- 4. Bash*
- 5. Python*
- 6. php*
- 7. Portfolio*

# 1.Gantt



## 2. Matrice RACI

Matrice RACI		
Taches	Pierre	Yohan
Gantt	R	I
RACI	R	C
Bash	C, I	R
Python	R	C, I
PHP	I	R
Rapport, Diapo	R	C, I

R = Responsable

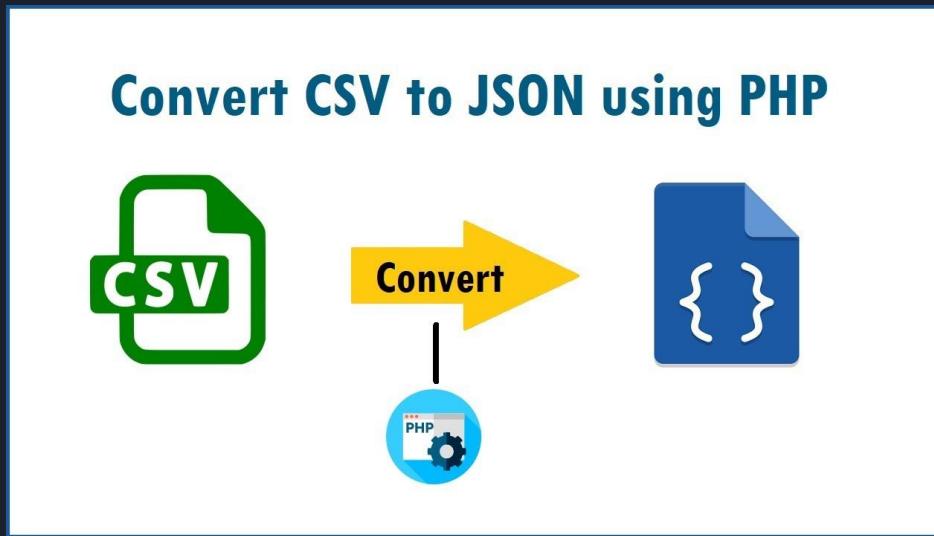
A = Autorité

C = Consultant

I = Informé

### 3.Présentation

Le but de cette sae 15, était de récupérer un fichier csv, puis de le convertir en fichier json, et l'afficher grâce à un script php



## 4.Bash

```
1 #!/usr/bin/env bash
2 #!/usr/bin/env node
3
4 # proxy de l'iut
5 export http_proxy=cache-etu.univ-artois.fr:3128
6 export https_proxy=cache-etu.univ-artois.fr:3128
7
8 # téléchargement du fichier csv et extraction du fichier
9 wget https://www.insee.fr/statistiques/fichier/4768335/etatcivil2019_nais2019_csv.zip
10 unzip etatcivil2019_nais2019_csv.zip
11
12 # sélection de la colonne qui nous intéresse
13 cut -d';' -f22 FD_NAIS_2019.csv > userdata.csv
14
15 # création du fichier json
16 touch userdata.json
17 python3 python.py
18
19 # suppression des fichiers inutiles
20 rm varmod_NAIS_2019.csv
21 rm -r etatcivil2019_nais2019_csv.zip
22 rm FD_NAIS_2019.csv
23 rm userdata.csv
24
25 # attribution des droits d'exécution
26 chmod +x python.py
27 chmod 777 index.php
28 chmod 777 userdata.json
29
30 # organisation des fichiers
31 mkdir ~/public_html
32 cp index.php ~/public_html
33 mv userdata.json ~/public_html
34
35 username=$(whoami)
36 firefox http://172.31.25.9/~$username/
```

*Ce script bash sert à récupérer le fichier csv, à le dézipper puis il trie et découpe le fichier csv, et finis en lançant notre script python.*

# 5.Python

```
1 # Importation des bibliothèques nécessaires
2 import csv
3 import json
4 import os
5
6 def csv_to_json(csvFilePath, jsonFilePath):
7     """
8         Fonction qui prend en entrée les chemins d'accès des fichiers CSV et JSON et effectue la conversion.
9     """
10    deces_compteur = {}                                # Création d'un dictionnaire pour stocker les comptages de deces par mois
11
12    with open(csvFilePath, encoding='utf-8') as csvf:   # Ouverture du fichier CSV en mode lecture
13        csvReader = csv.DictReader(csvf)                # Utilisation de DictReader pour convertir les lignes du fichier CSV en dictionnaires Python
14
15        for row in csvReader:                          # Boucle pour parcourir chaque ligne du fichier CSV convertie en dictionnaire Python
16            month = row['MNAIS']                         # Récupération de la valeur de la colonne "MNAIS" pour chaque ligne
17            if month in deces_compteur:                  # Incrémentation si le mois existe déjà dans le dictionnaire
18                deces_compteur[month] += 1
19            else:
20                deces_compteur[month] = 1
21
22 #####
23
24    data = []                                         # Création d'une liste pour stocker les données converties au format JSON
25
26    for month, count in deces_compteur.items():       # Boucle pour parcourir les entrées du dictionnaire deces_compteur
27        month_int = int(month)                        # Conversion de la valeur de "month" en entier
28        count_str = str(count)                        # Conversion de la valeur de "count" en chaîne de caractères
29        data.append({"y": count, "x": month_int, "label": count_str}) # Ajout d'une entrée à la liste "data".
30
31    data = sorted(data, key=lambda x: x["x"])          # Tri de la liste "data" selon la valeur de x
32
33    with open(jsonFilePath, 'w', encoding='utf-8') as jsonf: # Ouverture du fichier JSON en mode écriture
34        json.dump(data, jsonf, indent=2)                 # Ecriture des données de la liste "data" dans le fichier JSON avec une indentation de 2 espaces
35
36 #####
37
38 csvFilePath = r'userdata.csv'                      # Chemins d'accès des fichiers CSV et JSON
39 jsonFilePath = r'userdata.json'
40
41 csv_to_json(csvFilePath, jsonFilePath)             # Appel de la fonction pour convertir le fichier CSV en JSON
42
43 php_script_path = os.path.join("/Documents/sae15", "index.php") # Redirection vers la page PHP en utilisant le chemin absolu du fichier PHP
44 script_directory = os.path.dirname(os.path.abspath(__file__)) # Définition du répertoire de travail
45
46 os.chdir(script_directory)                        # Redirection vers la page PHP
47 os.system("php {}".format(php_script_path))
```

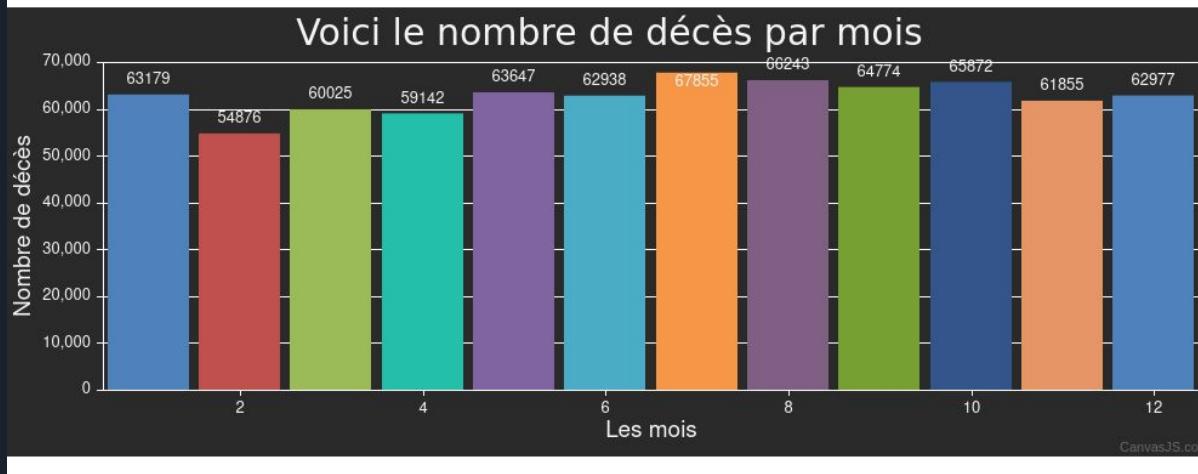
Ce script python sert à convertir le fichier csv en fichier json.

## 6. PHP

*Ce script php sert à afficher les données à l'origine se trouvant dans le fichier csv d'origine, qui a ensuite été trié, puis transformé en fichier json.*

*Le script était fourni, suffit de le modifier et de l'adapter.*

Le graphique ci-dessous représente le nombre de décès total par mois en 2019 :



## 7.Portfolio

	Ressources	Niveau d'acquisition
1	R1.06	Acquis
2	R1.07	Acquis
3	R1.08	Acquis
4	R1.09	Acquis
5	R1.10	Acquis
6	R1.11	Acquis
7	R1.15	Acquis
8		

*Nous devions faire un portfolio, contenant les compétences acquises, et les ressources acquises.*