

## Challenge 2024-05-31 : Classement par moyennes

## Table des matières

Challenge 2024-05-31 : Classement par moyennes .....	1
1) Enoncé .....	3
2) Logigramme des modules .....	4
a. Logigramme simple du code .....	4
3) Variables du code .....	5
4) Code.....	6
b. constantes.py .....	6
c. main.py .....	7
d. moyennes.py .....	8
e. classement.py.....	9
1) Résulta .....	10

## 1) Enoncé

Ce challenge est un niveau débutant, mais si vous avez plus d'expérience, vous pouvez essayer de trouver de belles astuces pour un code propre, rapide et concis. Voici la liste des élèves et leurs notes respectives:

```
Tao      -> 18, 12, 3, 5, 19
Josette  -> 20, 2, 12, 18, 14
Patrick  -> 2, 4, 6, 18, 17
Pema     -> 3, 19, 15, 3, 12
Jean     -> 0, 9, 8, 8, 4
Bixente  -> 14, 20, 10, 12, 4
Paco     -> 16, 1, 1, 1, 20
Chuluun  -> 15, 6, 17, 20, 15
Marie    -> 16, 4, 16, 20, 12
Mohamed  -> 16, 19, 17, 6, 20
```

### Conditions :

Créez une fonction `moyenne_eleves()` qui retourne un dictionnaire contenant en clé le nom de l'élève et en valeur sa moyenne arrondie à l'entier le plus proche.

Créez une fonction `classement_eleves()` qui permettra d'afficher dans le terminal le classement complet de la classe avec la position de chaque élève par rapport aux autres.

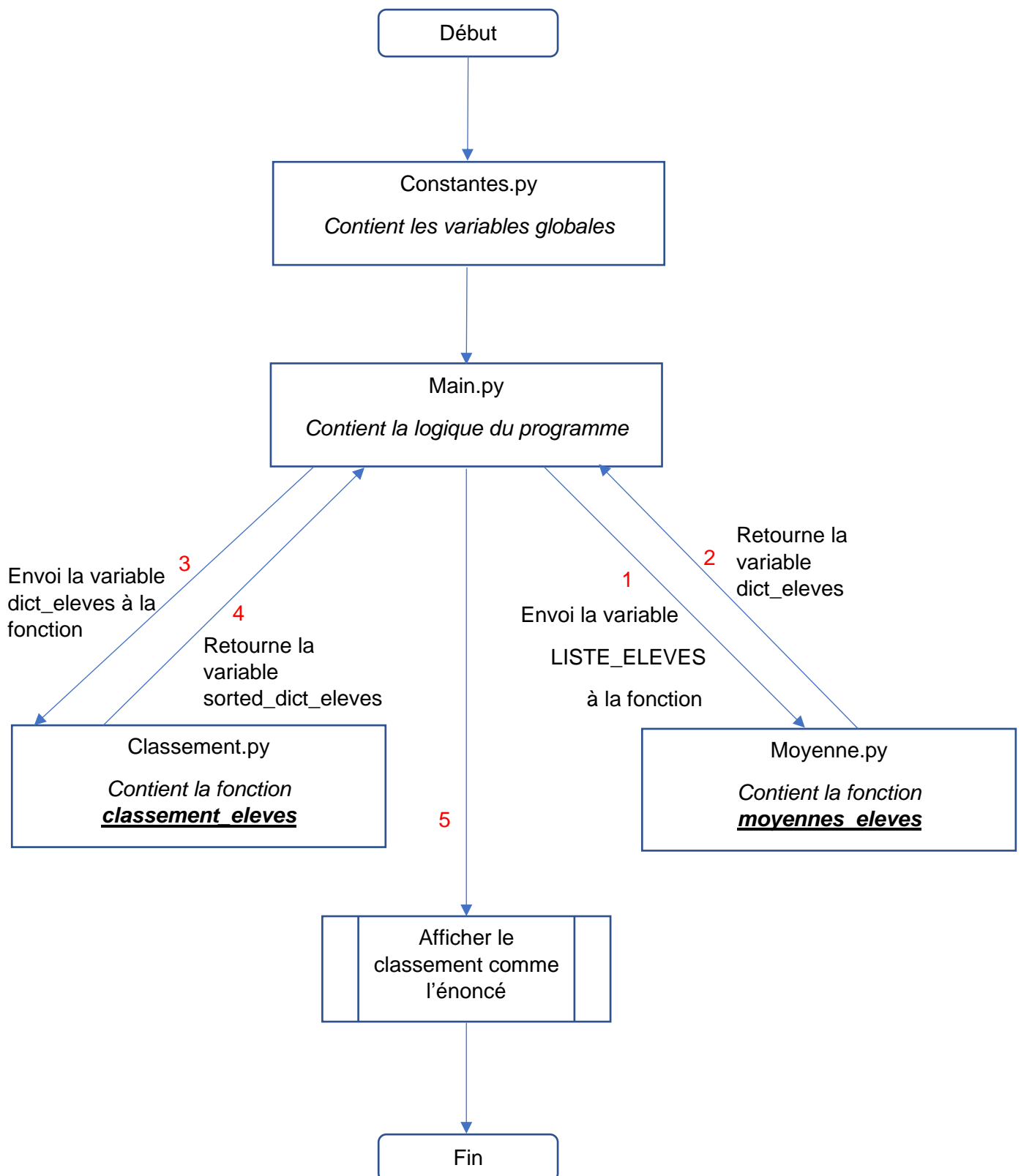
### Exemples :

```
{'Tao': 11, 'Josette': 13, 'Patrick': 9, 'Pema': 10, 'Jean': 6,
 'Bixente': 12, 'Paco': 8, 'Chuluun': 15, 'Marie': 14, 'Mohamed': 16}
```

```
1 : Mohamed avec une moyenne de 16/20
2 : Chuluun avec une moyenne de 15/20
3 : Marie avec une moyenne de 14/20
4 : Josette avec une moyenne de 13/20
5 : Bixente avec une moyenne de 12/20
6 : Tao avec une moyenne de 11/20
7 : Pema avec une moyenne de 10/20
8 : Patrick avec une moyenne de 9/20
9 : Paco avec une moyenne de 8/20
10 : Jean avec une moyenne de 6/20
```

## 2) Logigramme des modules

### a. Logigramme simple du code



### 3) Variables du code

Modules	Nom de la variable	Type	Commentaires
constantes.py	LISTE_ELEVES	list	Liste contenant le nom des élèves ainsi que leurs notes
main.py moyennes.py	dict_eleves	Dict	Dictionnaire contenant le nom et la moyenne des élèves
main.py	sorted_dict_eleves	list	Liste classant par ordre décroissant la moyenne de chaque élève avec leur nom

#### 4) Code

##### b. constantes.py

```
# constantes.py
# But:
# Contient les constantes du code
# -----
# Date de création: 2024-06-01
# Date de dernière modification: 2024-06-01
# -----
# version: 1.0
#-----

LISTE_ELEVES=[
    ["Tao", [18, 12, 3, 5, 19]],
    ["Josette", [20, 2, 12, 18, 14]],
    ["Patrick", [2, 4, 6, 18, 17]],
    ["Pema", [3, 19, 15, 3, 12]],
    ["Jean", [0, 9, 8, 8, 4]],
    ["Bixente", [14, 20, 10, 12, 4]],
    ["Paco", [16, 1, 1, 1, 20]],
    ["Chuluun", [15, 6, 17, 20, 15]],
    ["Marie", [16, 4, 16, 20, 12]],
    ["Mohamed", [16, 19, 17, 6, 20]]
]

"""
Tao      -> 18, 12, 3, 5, 19
Josette  -> 20, 2, 12, 18, 14
Patrick  -> 2, 4, 6, 18, 17
Pema     -> 3, 19, 15, 3, 12
Jean     -> 0, 9, 8, 8, 4
Bixente  -> 14, 20, 10, 12, 4
Paco     -> 16, 1, 1, 1, 20
Chuluun  -> 15, 6, 17, 20, 15
Marie    -> 16, 4, 16, 20, 12
Mohamed  -> 16, 19, 17, 6, 20
"""
```

### c. main.py

```
# main.py
# But:
# Contient la logique du programme
# -----
# Date de création: 2024-06-01
# Date de dernière modification: 2024-06-01
# -----
# version: 1.0
#-----

# Appelle des modules externes

# Appelle des modules internes
from constantes import LISTE_ELEVES
from moyenne import moyenne_eleves
from classement import classement_eleves

if __name__ == '__main__':

    dict_eleves = moyenne_eleves(LISTE_ELEVES)
    sorted_dict_eleves = classement_eleves(dict_eleves)
    for index_numb, infos_eleves in enumerate(sorted_dict_eleves):
#Afficher le classement des élèves
        print(f"{index_numb+1}: {infos_eleves[0]} avec une moyenne de
{infos_eleves[1]}/20")
```

d. moyennes.py

```
# # moyenne.py
# But:
# Contient la fonction moyenne_eleves
# => Renvoyer la valeur moyenne de chaque élève sous forme de dico
# -----
# Date de création: 2024-06-01
# Date de dernière modification: 2024-06-01
# -----
# version: 1.0
#-----

def moyenne_eleves(LISTE_ELEVES):
    dict_eleves={}
    for eleves in LISTE_ELEVES:
        moyenne_float = sum(eleves[1])/len(eleves[1])
        moyenne = int(round(moyenne_float, 0)) # Retourne la moyenne
        # arrondi à l'entier le + proche
        dict_eleves[eleves[0]] = moyenne # Sauvegarde de la moyenne avec
        # l'élève associé
    return dict_eleves
```



e. classement.py

```
# classement.py
# But:
# Contient la fonction classement_eleves
# => Renvoyer le classement des élèves par rapport à leur moyenne sous
forme de dico
# -----
# Date de création: 2024-06-01
# Date de dernière modification: 2024-06-01
# -----
# version: 1.0
#-----

def classement_eleves(dict_eleves):
    return sorted(dict_eleves.items(), key=lambda x:x[1], reverse=True)
#Trie par ordre décroissant
```

## 1) Résultat

```
1: Mohamed avec une moyenne de 16/20
2: Chuluun avec une moyenne de 15/20
3: Marie avec une moyenne de 14/20
4: Josette avec une moyenne de 13/20
5: Bixente avec une moyenne de 12/20
6: Tao avec une moyenne de 11/20
7: Pema avec une moyenne de 10/20
8: Patrick avec une moyenne de 9/20
9: Paco avec une moyenne de 8/20
10: Jean avec une moyenne de 6/20

Process finished with exit code 0
```