

TP 2 – Gestion de notes (moyenne, min, max)

Objectif du TP

Créer un programme permettant de **lire, analyser et afficher des statistiques sur une série de notes** contenues dans un fichier texte. L'objectif est de **manipuler les fichiers, les listes et les boucles**, tout en appliquant une logique algorithmique simple.

Remarque si vous souhaitez vérifier qu'un fichier existe, utilisez le code suivant :

```
import os

if os.path.exists("data/notes.txt"):
    print("Le fichier existe.")
else:
    print("Le fichier n'existe pas.")
```

Consignes

1. Le programme lit un fichier `notes.txt` contenant une série de notes (une par ligne). Créez ce fichier. Exemple :

```
notes
11
12
15
9
14
```

2. Le programme ignore la première ligne (`notes`) et convertit les valeurs restantes en entiers.
3. Il calcule :
 - la **moyenne** des notes,
 - la **note minimale**,
 - la **note maximale**.

4. Il affiche les résultats de manière claire :

```
Moyenne : 12.2
Note minimale : 9
Note maximale : 15
```

5. Le programme doit gérer les erreurs possibles :

- fichier manquant
- données non numériques

6. Le code doit être structuré à l'aide de fonctions :

- `read_notes(filename)`
 - `compute_stats(notes)`
 - `display_results(stats)`
-

Éléments attendus

- Lecture et traitement d'un fichier texte avec `open()`
 - Boucle `for` pour parcourir les lignes
 - Gestion d'exceptions avec `try / except`
 - Fonctions claires et bien nommées
 - Utilisation de structures de données adaptées (`list`, `dict`)
-

Améliorations possibles (bonus)

- Ajouter la possibilité de **saisir le nom du fichier** à analyser.
 - Sauvegarder les résultats dans un **nouveau fichier texte** (`resultats.txt`).
 - Afficher les notes **triées par ordre croissant et décroissant**.
-

Objectif pédagogique

- Pratiquer la **manipulation de fichiers** en lecture et écriture.
- Appliquer les **boucles, conditions et fonctions** dans un programme complet.
- Introduire la **gestion d'erreurs et la validation de données**.