# Suivi des notes des étudiants

## Vue d'ensemble :

Dans ce projet, vous allez créer un **Système de suivi des notes des étudiants**. Ce système permettra aux enseignants d'entrer les noms des étudiants et leurs notes pour différentes matières, de calculer la moyenne de leurs notes et d'afficher les résultats. Cette version nécessite l'utilisation de plusieurs fichiers afin d'encourager l'organisation et la modularité.

## Exigences du projet :

**1. Fichiers à créer :**

* **main.py** - Le fichier principal qui exécute le programme et contient la logique de base.
* **student.py** - Contient une classe Student avec des méthodes pour gérer les données des étudiants (nom et notes).
* **grades.py** - Contient des fonctions auxiliaires pour les opérations liées aux notes (comme la validation des notes, le calcul des moyennes).
* **utils.py** - Contient des fonctions utilitaires pour des tâches comme la gestion des entrées et le formatage des sorties.

## Fichiers expliqués :

**1. student.py :**

* Ce fichier contient la classe Student avec les éléments suivants :
  + **Attributs :**
    - name : Une chaîne représentant le nom de l'étudiant.
    - grades : Une liste de valeurs numériques représentant les notes que l'étudiant a reçues.
  + **Méthodes :**
    - add\_grade(grade) : Ajoute une note à la liste des notes de l'étudiant.
    - calculate\_average() : Calcule et retourne la moyenne des notes.

**2. grades.py :**

* Ce fichier contient des fonctions pour travailler avec les notes :
  + **Fonctions :**
    - is\_valid\_grade(grade) : Retourne True si la note est un nombre valide (entre 0 et 100), sinon retourne False.
    - calculate\_average(grades) : Prend une liste de notes et retourne la moyenne.

**3. utils.py :**

* Ce fichier contient des fonctions utilitaires pour aider à l'interaction avec l'utilisateur :
  + **Fonctions :**
    - get\_student\_input() : Demande à l'utilisateur d'entrer le nom de l'étudiant et les notes, en s'assurant que l'entrée est valide.
    - display\_results(student) : Affiche le nom de l'étudiant, ses notes et sa moyenne de manière lisible.

**4. main.py :**

* C'est le fichier principal qui connecte tout :
  + **Entrée :**
    - Demande à l'utilisateur d'entrer le nom de l'étudiant et les notes.
    - Gère les entrées invalides en demandant de répéter l'entrée jusqu'à ce qu'elle soit correcte.
  + **Sortie :**
    - Affiche le nom de l'étudiant, ses notes et sa moyenne une fois que toutes les notes ont été entrées.

## Exemple d'interaction :

Entrez le nom de l'étudiant : Alice

Entrez la note pour Alice (entre 0 et 100, ou tapez 'done' pour finir) : 85

Entrez la note pour Alice (entre 0 et 100, ou tapez 'done' pour finir) : 90

Entrez la note pour Alice (entre 0 et 100, ou tapez 'done' pour finir) : done

Étudiant : Alice

Notes : [85, 90]

Moyenne des notes : 87,5

## Fonctionnalités supplémentaires (optionnelles) :

* Permettre à l'utilisateur d'entrer plusieurs étudiants.
* Ajouter la possibilité d'afficher les résultats de tous les étudiants en une seule fois après avoir saisi les notes de plusieurs étudiants.

## Critères d'évaluation :

* **Fonctionnalité (50%)** : Le système doit accepter correctement les noms et les notes des étudiants, calculer la moyenne et l'afficher.
* **Structure du code (30%)** : Le code doit être modulaire et organisé à travers les quatre fichiers.
* **Gestion des erreurs (20%)** : Le programme doit gérer les entrées invalides (par exemple, valeurs non numériques, notes hors de la plage) de manière appropriée.