

# Documentation

---

## Application de Gestion des Notes Étudiantes

---

Ce projet est une application Python conçue pour traiter les notes des étudiants. Elle offre une interface conviviale permettant de sélectionner un fichier de notes et de générer des fichiers de sortie contenant le traitement et l'analyse des données.

### Structure du Projet

```
application-notes-etudiantes
├── main.py                # Point d'entrée de l'application
├── ui.py                  # Code de l'interface utilisateur
├── backend.py             # Logique pour le traitement des données de notes
├── utils
│   └── file_utils.py      # Fonctions utilitaires pour les opérations sur les
fichiers
├── requirements.txt       # Dépendances du projet
└── README.md              # Documentation du projet
```

### Installation

#### 1. Cloner le dépôt :

```
git clone https://github.com/bkablaml1/app-student-grades.git
cd app-student-grades
```

#### 2. Créer un environnement virtuel (optionnel mais recommandé) :

```
python -m venv env
source env/bin/activate      # Sur Linux/Mac
env\Scripts\activate        # Sur Windows
```

#### 3. Installer les dépendances :

```
pip install -r requirements.txt
```

#### 4. Vérifier les dépendances installées : Les bibliothèques suivantes doivent être installées :

- `tk` : Pour l'interface graphique.
- `pandas` : Pour la manipulation des données.

- `statistics` : Pour les calculs statistiques.
- `pyinstaller` : Pour créer un exécutable.
- `matplotlib` : Pour générer des graphiques.

## Utilisation

### 1. Lancer l'application :

```
python main.py
```

## Fonction `lire_coefficients`

### Description

Lit les coefficients des matières depuis un fichier CSV.

### Algorithme

1. Ouvrir le fichier CSV contenant les coefficients.
2. Lire les lignes du fichier en tant que dictionnaire.
3. Parcourir chaque ligne :
  - Si la ligne correspond à celle des coefficients (`Libelle == 'Coeff'`), extraire les coefficients pour chaque matière.
  - Convertir les coefficients en entiers.
  - Si une conversion échoue, lever une erreur.
4. Retourner un dictionnaire contenant les matières et leurs coefficients.

---

## Fonction `process_grades`

### Description

Traite les données des étudiants, calcule les moyennes pondérées, les rangs et génère deux fichiers CSV :

- `traitement.csv` : Contient les notes, moyennes pondérées et rangs des étudiants.
- `analyse.csv` : Contient les statistiques par matière et pour la classe.

### Algorithme

1. Lire les coefficients des matières en appelant `lire_coefficients`.
2. Lire et traiter les données des étudiants :
  - Extraire les matières et les notes pour chaque étudiant.
  - Calculer la moyenne pondérée pour chaque étudiant.
3. Calculer les rangs pour chaque matière et pour la moyenne générale.
4. Calculer les statistiques par matière :
  - Moyenne, médiane, écart type, effectifs, etc.
5. Générer deux fichiers CSV :
  - `traitement.csv` : Contient les données des étudiants.

- `analyse.csv` : Contient les statistiques par matière et pour la classe.
- 

## Fonction `calculer_rangs`

### Description

Calcule les rangs pour une liste de moyennes ou de valeurs numériques.

### Algorithme

1. Associer chaque moyenne à son index original.
  2. Trier les moyennes par ordre décroissant.
  3. Parcourir les moyennes triées :
    - Identifier les moyennes égales (ex aequo).
    - Attribuer un rang formaté (ex : "1er", "2e ex").
  4. Retourner une liste contenant les rangs pour chaque moyenne.
- 

## Calcul des Statistiques par Matière

### Description

Effectué dans `process_grades` pour chaque matière.

### Algorithme

1. Pour chaque matière :
    - Extraire les notes valides.
    - Calculer :
      - La note minimale et maximale.
      - La moyenne.
      - La médiane.
      - L'écart type (si au moins deux notes valides).
      - Le nombre de notes supérieures à 10.
      - Le nombre de notes supérieures à la moyenne de la classe.
  2. Ajouter les statistiques dans un dictionnaire.
- 

## Génération des Fichiers CSV

### Description

Effectué dans `process_grades` pour produire les fichiers `traitement.csv` et `analyse.csv`.

### Algorithme

1. Ouvrir un fichier CSV en mode écriture.
2. Écrire l'en-tête (colonnes).
3. Parcourir les données des étudiants :

- Écrire une ligne pour chaque étudiant avec ses notes, moyennes et rangs.
4. Ajouter une ligne pour les moyennes de classe.
  5. Répéter pour le fichier des statistiques (`analyse.csv`).
- 

## Calcul des Statistiques Globales de la Classe

### Description

Effectué dans `process_grades` pour calculer les statistiques globales de la classe.

### Algorithme

1. Extraire les moyennes générales valides des étudiants.
  2. Calculer :
    - La moyenne générale de la classe.
    - La médiane des moyennes générales.
    - L'écart type des moyennes générales.
    - Le nombre de moyennes supérieures à 10.
    - Le nombre de moyennes supérieures à la moyenne générale de la classe.
  3. Ajouter ces statistiques dans un dictionnaire.
- 

## Fonction `select_files`

### Description

Permet à l'utilisateur de sélectionner les fichiers CSV de notes et de coefficients via une interface graphique.

### Algorithme

1. Ouvrir une boîte de dialogue pour sélectionner le fichier des notes (`notes_file`).
    - Si aucun fichier n'est sélectionné, afficher un message d'avertissement et arrêter l'exécution.
  2. Ouvrir une boîte de dialogue pour sélectionner le fichier des coefficients (`coeff_file`).
    - Si aucun fichier n'est sélectionné, afficher un message d'avertissement et arrêter l'exécution.
  3. Si les deux fichiers sont sélectionnés, activer le bouton pour générer les fichiers (`generate_button`).
- 

## Fonction `generate_files`

### Description

Génère les fichiers `traitement.csv` et `analyse.csv` à partir des fichiers de notes et de coefficients sélectionnés. Affiche une barre de progression pendant le traitement.

### Algorithme

1. Créer une barre de progression et la démarrer.
2. Appeler la fonction `process_grades` (du module `grades_processor`) pour traiter les fichiers de notes et de coefficients.

- Récupérer les chemins des fichiers générés (`traitement_file` et `analyse_file`).
- 3. Arrêter et détruire la barre de progression.
- 4. Afficher un message de succès avec les chemins des fichiers générés.
- 5. Activer le bouton pour afficher les graphiques (`graph_button`).
- 6. En cas d'erreur :
  - Arrêter et détruire la barre de progression.
  - Afficher un message d'erreur avec les détails de l'exception.

---

## Fonction `show_graphs`

### Description

Affiche des graphiques basés sur les données du fichier `analyse.csv`.

### Algorithme

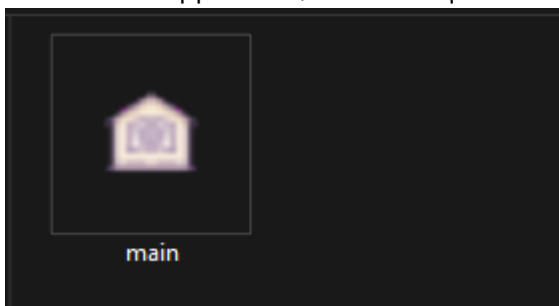
1. Ouvrir le fichier `analyse.csv` en mode lecture.
2. Lire les données du fichier en tant que dictionnaire (`csv.DictReader`).
3. Préparer les données pour les graphiques :
  - Extraire les matières, les moyennes de classe, les notes minimales et maximales.
  - Extraire les effectifs (`Effectif Moyenne>10` et `Effectif Moyenne > Moyenne Classe`).
4. Créer un graphique :
  - Utiliser un diagramme en barres pour les moyennes de classe.
  - Ajouter des courbes pour les notes minimales et maximales.
5. Créer un diagramme à bandes pour les effectifs :
  - Afficher les effectifs des étudiants ayant une moyenne > 10 et une moyenne > moyenne de la classe.
6. Configurer les graphiques :
  - Ajouter un titre, des étiquettes pour les axes, une légende et ajuster l'affichage.
7. Afficher les graphiques avec `matplotlib`.
8. En cas d'erreur, afficher un message d'erreur avec les détails de l'exception.

---

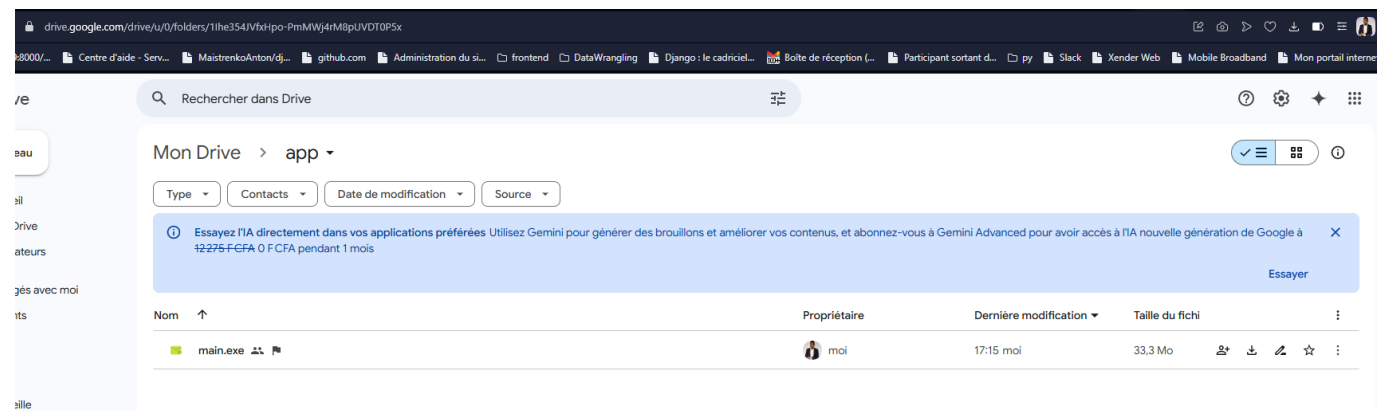
## Exemple de Flux d'Utilisation

### Lancement de l'application

Pour lancer l'application, double-cliquez sur le fichier `main` qui se trouve dans le dossier `dist`.



lien de telechargement du fichier `dist` [Télécharger le fichier](#)



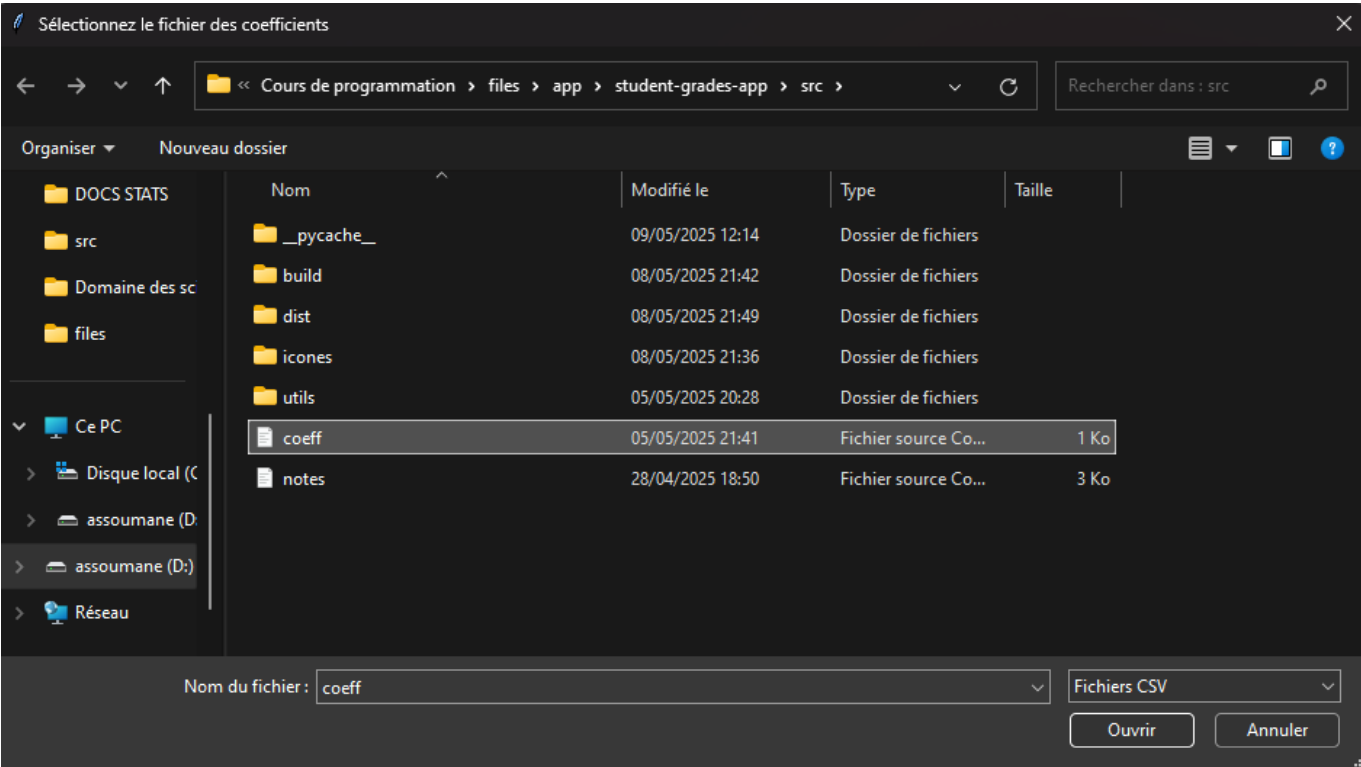
1. Sélection des fichiers

L'utilisateur clique sur le bouton "Sélectionner les fichiers de notes et de coefficients". Deux boîtes de dialogue s'ouvrent pour sélectionner les fichiers nécessaires.

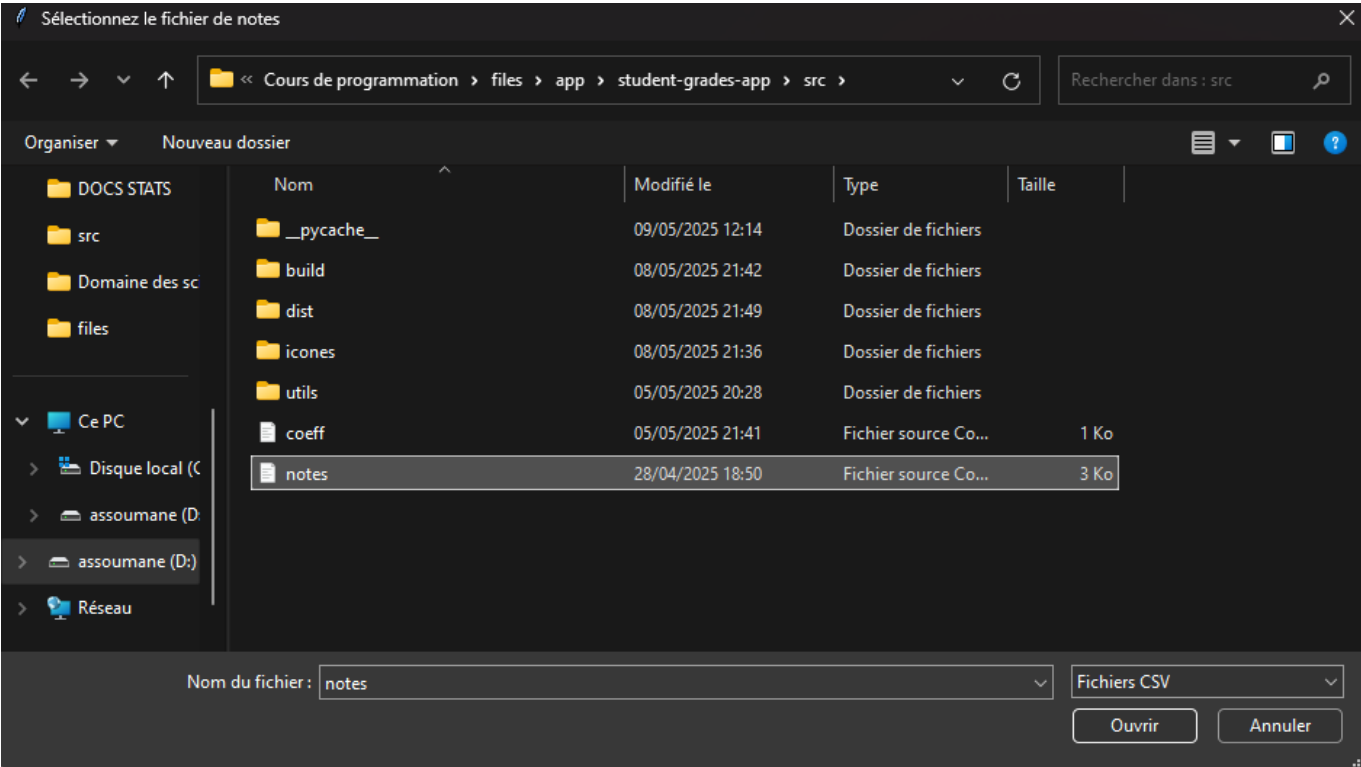
Étape 0 : Interface



Étape 1 : Sélection du fichier des coefficients



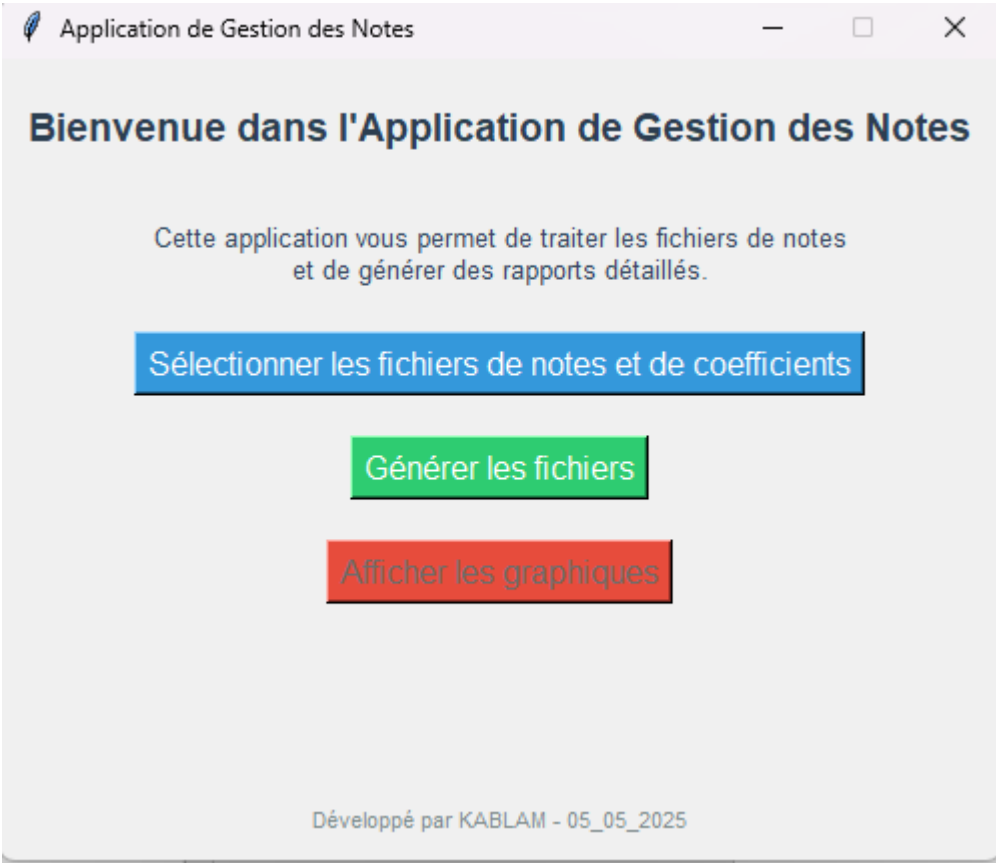
Étape 2 : Sélection du fichier des notes



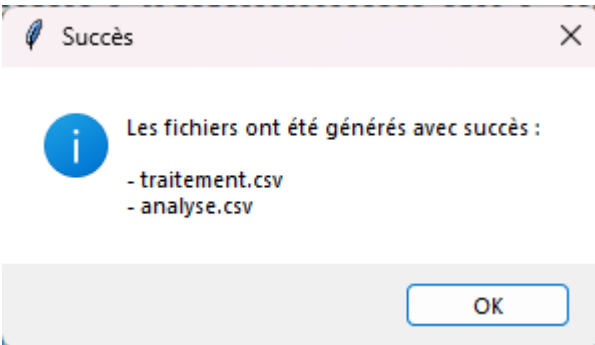
Une fois les fichiers sélectionnés, le bouton "Générer les fichiers" est activé.

2. Génération des fichiers

L'utilisateur clique sur le bouton "Générer les fichiers". Une barre de progression s'affiche pendant le traitement.





**Message de succès après la génération des fichiers**



Les fichiers générés sont :

- traitement.csv
- analyse.csv

**Fichiers générés dans le répertoire**

Nom	Modifié le	Type	Taille
 analyse	09/05/2025 14:06	Fichier source Comma Separated Values	1 Ko
 traitement	09/05/2025 14:06	Fichier source Comma Separated Values	7 Ko

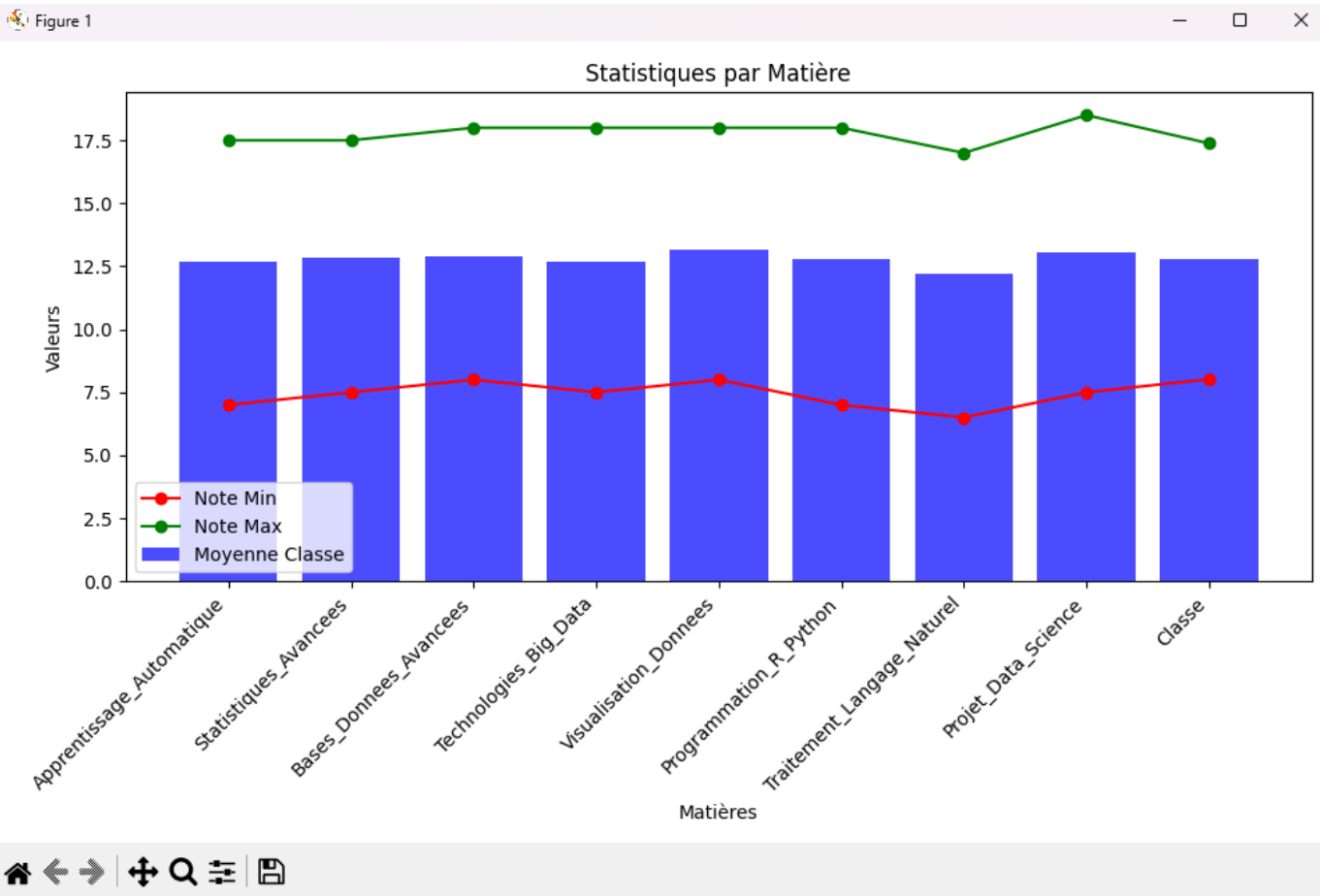
**3. Affichage des graphiques**



L'utilisateur clique sur le bouton **"Afficher les graphiques"**. Les graphiques des statistiques par matière sont affichés dans une nouvelle fenêtre.

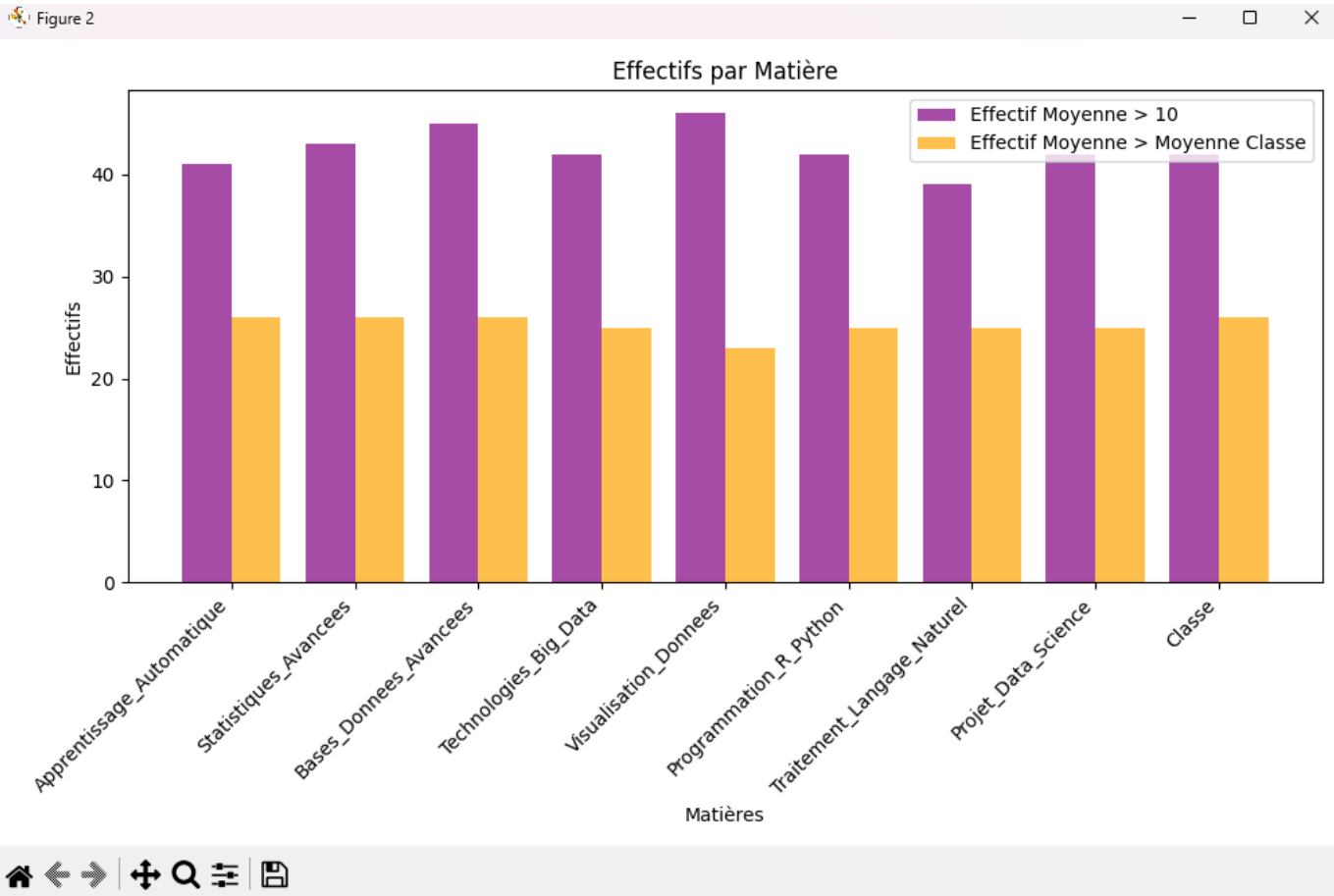


Graphique 1 : Statistiques par matière



- **Barres bleues** : Moyenne de la classe par matière.
- **Points rouges** : Notes minimales par matière.
- **Points verts** : Notes maximales par matière.

Graphique 2 : Effectifs par matière



- **Barres violettes** : Effectifs des étudiants ayant une moyenne > 10.
- **Barres oranges** : Effectifs des étudiants ayant une moyenne > moyenne de la classe.

#### 4. Fichiers générés

Les fichiers générés contiennent les données suivantes :

Exemple de fichier `traitement.csv`

 Edit File









	A	B	C	D	E	F	G	H
1	ID_Etudiant	Nom	Prenom	Apprentissage_Automatique_Note	Apprentissage_Automatique_Rang	Statistiques_Avancees_Note	Statistiques_Avancees_Rang	Bases_Donnees_Avancees_Note
2	1	Koné	Adama	14.50	14e ex	12.00	30e ex	15.50
3	2	Goli	Lucie	11.00	34e ex	13.50	21e ex	10.00
4	3	Kouamé	Koffi	8.50	46e ex	9.00	46e ex	11.50
5	4	Lou Nan	Irié	16.00	6e ex	15.50	9e ex	14.00
6	5	Diarra	Moussa	12.50	27e ex	11.00	36e ex	9.00
7	6	Mireille	Ekra	13.00	24e ex	14.00	17e ex	16.00
8	7	N'Guessan	Yao	9.00	44e ex	10.50	39e ex	8.00
9	8	Bamba	Mariam	17.50	1er ex	16.00	5e ex	18.00
10	9	Michel	Bleu	10.00	39e ex	12.50	27e ex	13.00
11	10	Sangaré	Bintou	15.00	11e ex	14.50	15e ex	12.00
12	11	Leonard	Zadi	7.00	50e	8.50	49e	10.00
13	12	Konan	Akissi	14.00	17e ex	13.00	24e ex	11.50
14	13	Doumbia	Seydou	11.50	32e ex	10.00	41e ex	14.00
15	14	Koffi	Amoin	16.50	5e	17.00	2e	15.00
16	15	Attemené	Zoro	9.50	42e ex	11.50	34e ex	12.50
17	16	Pierre	Blé	13.50	21e ex	12.00	30e ex	13.50
18	17	Zran	Louise	10.50	37e ex	9.50	44e ex	11.00
19	18	Diomandé	Assita	15.50	9e ex	16.50	3e ex	17.00
20	19	Kassi	Kouadio	12.00	29e ex	13.50	21e ex	10.50

Traitement_Langage_Naturel_Note	Traitement_Langage_Naturel_Rang	Projet_Data_Science_Note	Projet_Data_Science_Rang	Moyenne_Generale	Rang_General
10.00	37e ex	15.00	14e ex	13.47	22e
14.00	15e ex	12.00	30e ex	11.65	33e
8.00	46e ex	10.50	38e ex	9.88	42e ex
15.00	9e ex	17.50	4e	15.70	8e
13.50	19e ex	11.00	36e ex	11.78	32e
12.00	26e ex	13.00	26e ex	14.10	18e
11.00	31e ex	9.50	43e ex	9.57	44e
16.50	4e	18.00	2e ex	17.05	3e
9.50	40e ex	12.50	28e ex	11.23	35e
17.00	1er ex	15.50	11e ex	15.03	12e ex
7.50	48e	8.00	48e ex	8.42	49e
12.50	24e ex	14.50	18e ex	13.23	25e
10.00	37e ex	11.50	33e ex	11.40	34e
14.00	15e ex	16.00	9e ex	15.95	7e
11.50	29e ex	10.00	41e ex	10.72	38e
13.00	22e ex	13.50	24e ex	13.38	23e
8.50	45e	9.00	45e ex	9.43	45e
15.50	7e ex	17.00	5e ex	16.03	6e
11.00	31e ex	12.00	30e ex	12.13	29e
14.50	12e ex	15.00	14e ex	14.15	17e

34	33	Tuo	Leslie	8.50	46e ex	9.00	46e ex	8.50	49e
35	34	Dadié	Samuelle	17.00	3e ex	17.50	1er	17.50	2e
36	35	Assi	Bertin	10.50	37e ex	12.00	30e ex	11.00	36e ex
37	36	Sanogo	Maimouna	15.50	9e ex	14.00	17e ex	13.00	23e ex
38	37	Adia	Luc	12.00	29e ex	13.00	24e ex	14.00	16e ex
39	38	Didot	Marc	14.00	17e ex	12.50	27e ex	15.00	12e ex
40	39	Gbami	Serge	9.00	44e ex	10.00	41e ex	10.50	40e ex
41	40	Augustine	Gouzilé	16.00	6e ex	16.50	3e ex	16.50	5e
42	41	Sarako	Aline	13.00	24e ex	11.00	36e ex	12.00	29e ex
43	42	Lucie	Hé	14.50	14e ex	15.50	9e ex	14.00	16e ex
44	43	Kamagate	Gaoussou	10.00	39e ex	9.00	46e ex	11.50	33e ex
45	44	Kilina	Estelle	15.00	11e ex	16.00	5e ex	15.00	12e ex
46	45	Adeline	Selé	11.50	32e ex	12.50	27e ex	13.50	21e ex
47	46	Laetitia	Yao	13.50	21e ex	14.00	17e ex	12.00	29e ex
48	47	Albert	Amon	8.00	48e ex	9.50	44e ex	10.00	42e ex
49	48	Basile	Tano	17.50	1er ex	16.00	5e ex	17.00	3e ex
50	49	Fofana	Bintou	12.00	29e ex	13.00	24e ex	11.00	36e ex
51	50	Jules	Gueu	14.00	17e ex	15.00	11e ex	16.00	6e ex
52		Classe		12.67	-	12.83	-	12.92	-

Exemple de fichier analyse.csv

student-grades-app > src > analyse.csv

Edit File

	A	B	C	D	E	F	G	H
1	Matière	Note min	Note Max	Moyenne Classe	Médiane	Écart Type	Effectif Moyenne>10	Effectif Moyenne > Moyenne Classe
2	Apprentissage_Automatique	7.00	17.50	12.67	13.00	2.80	41	26
3	Statistiques_Avancees	7.50	17.50	12.83	13.00	2.54	43	26
4	Bases_Donnees_Avancees	8.00	18.00	12.92	13.00	2.57	45	26
5	Technologies_Big_Data	7.50	18.00	12.70	12.75	2.79	42	25
6	Visualisation_Donnees	8.00	18.00	13.14	13.00	2.62	46	23
7	Programmation_R_Python	7.00	18.00	12.77	12.75	2.84	42	25
8	Traitement_Langage_Naturel	6.50	17.00	12.23	12.25	2.86	39	25
9	Projet_Data_Science	7.50	18.50	13.08	13.25	2.97	42	25
10	Classe	8.02	17.38	12.81	12.81	2.57	42	26

Interface utilisateur

Voici l'interface principale de l'application :

