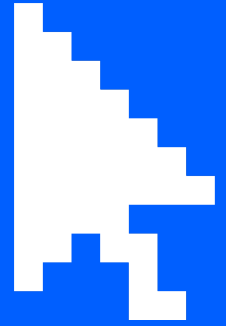


# La Plateforme\_Tracker

Prenez le contrôle du parcours éducatif avec La Plateforme\_Tracker



## Contexte

---

Vous êtes un développeur débutant qui souhaite se perfectionner en programmation et en gestion de bases de données.

Vous décidez de créer un projet de système de gestion d'étudiants pour mettre en pratique vos compétences. L'objectif est de développer une application Java qui permettra de gérer les informations des étudiants, telles que leur nom, prénom, âge, et notes.

## Projet

---

Créez une base de données **Postgres** pour stocker les informations des étudiants. Vous pouvez utiliser une table "**student**" avec les colonnes suivantes : "**id**", "**first\_name**", "**last\_name**", "**age**", et "**grade**".

Ajouter d'autres informations si nécessaire.

Votre programme doit permettre les actions suivantes :

- **Ajouter un étudiant** : permet à l'utilisateur de saisir les informations d'un nouvel étudiant (nom, prénom, âge, notes) et les enregistre dans la base de données.



- **Modifier un étudiant** : permet à l'utilisateur de rechercher un étudiant par son ID et de modifier ses informations (nom, prénom, âge, notes) dans la base de données.
- **Supprimer un étudiant** : permet à l'utilisateur de rechercher un étudiant par son ID et de le supprimer de la base de données.
- **Afficher tous les étudiants** : récupère tous les étudiants de la base de données et les affiche à l'utilisateur.
- **Rechercher un étudiant** : permet à l'utilisateur de rechercher un étudiant par son ID et d'afficher ses informations.

Pour ce projet, vous devez utiliser des requêtes SQL pour interagir avec la base de données. Vous devez utiliser des paramètres préparés pour éviter les problèmes de sécurité liés à l'injection SQL.

Créez une classe pour gérer la connexion à la base de données. Cette classe devrait inclure des méthodes pour établir la connexion, exécuter des requêtes et fermer la connexion.

Assurez-vous de gérer les exceptions liées à la connexion à la base de données et aux requêtes SQL.

Testez votre programme en utilisant différentes fonctionnalités pour vous assurer de son bon fonctionnement. Vous pouvez également simuler des scénarios d'erreur pour vérifier la robustesse de votre système.

Documentez votre code de manière appropriée, en ajoutant des commentaires et des explications pour faciliter la compréhension.

Vous pouvez utiliser la bibliothèque JDBC pour vous connecter à la base de données à partir de votre application Java.

Votre programme devra respecter à la lettre les informations fournies dans cet énoncé. N'hésitez pas à demander de l'aide si vous rencontrez des



difficultés lors de la réalisation du projet.

Comme vous êtes un développeur attentionné, vous voulez rendre plus facile d'utilisation votre programme. Pour ce faire

Pour ce faire, ajouter les fonctionnalités suivantes :

- **Tri des étudiants** : Ajoutez la possibilité de trier la liste des étudiants par nom, prénom, âge ou moyenne des notes.
- **Recherche avancée** : Implémentez une fonctionnalité de recherche avancée permettant de rechercher des étudiants selon des critères spécifiques tels que l'âge, la moyenne des notes, etc.
- **Statistiques** : Calculez et affichez des statistiques sur les étudiants, comme la moyenne des notes de la classe, le nombre d'étudiants par tranche d'âge, etc.
- **Import/Export de données** : Ajoutez la possibilité d'importer et d'exporter les données des étudiants à partir de fichiers CSV, XML ou JSON.
- **Pagination** : Si vous avez une grande quantité d'étudiants, ajoutez une pagination pour afficher les étudiants par lots plutôt que tous en une seule fois.
- **Gestion des erreurs améliorée** : Améliorez la gestion des erreurs en affichant des messages d'erreur plus clairs et en proposant des options de récupération en cas d'erreur de saisie.
- **Système d'authentification** : Ajoutez un système d'authentification pour sécuriser l'accès au système de gestion des étudiants en demandant un nom d'utilisateur et un mot de passe.
- **Export des résultats** : Permettez aux utilisateurs d'exporter les résultats des recherches ou des statistiques dans un fichier CSV, PDF ou HTML.
- **Sauvegarde automatique** : Implémentez une fonctionnalité de sauvegarde automatique des données des étudiants à intervalles réguliers pour éviter toute perte de données en cas de panne du système.



Pour aller jusqu'au bout de votre projet, n'hésitez pas à le mettre en production. Plusieurs services gratuits existent pour héberger votre backend et le rendre accessible partout !

## Aller plus loin

---

Pour pousser votre projet jusqu'au bout, pensez à le mettre en **production**. Vous pouvez le faire **gratuitement** grâce à plusieurs services d'hébergement disponibles en ligne.

## Compétences visées

---

- Installer et configurer son environnement de travail en fonction du projet.
- Développer des interfaces utilisateur
- Développer des composants métier
- Contribuer à la gestion d'un projet informatique
- Définir l'architecture logicielle d'une application
- Concevoir et mettre en place une base de données relationnelle
- Développer des composants d'accès aux données SQL et NoSQL

## Rendu

---

Votre travail est évalué en présentation sans support. Le projet est à rendre sur <https://github.com/prenom-nom/LaplateformeTracker>



## Base de connaissances

---

- [GIT](#)
- [GIT cheat sheet](#)
- [GIT travailler en équipe](#)
- [Trello](#)
- [JDBC](#)
- [JAVA FX](#)
- [POSTGRES](#)
- [Render](#)