



Rapport Projet de POO1

Ahmadou Bamba Mboup P32 379

Fallou Ndiaye P32 1290

Abdoulaye Thiaw Diouf P32 677

PLAN

1. Introduction

- Présentation générale du projet.
- Objectif du rapport.

2. Description des Fonctions de Hachage

- Liste des fonctions de hachage implémentées.
- Description de chaque fonction.

3. Justification des Choix des Fonctions de Hachage

- Critères de sélection des fonctions.
- Avantages et inconvénients de chaque fonction choisie.
- Comparaison avec d'autres fonctions possibles.

4. Tests et Résultats

- Méthodologie de test.
- Résultats des tests de performance.
- Analyse des résultats

5. Utilisation de L'application

- Fonctionnalités

6. Conclusion

- Résumé des points clés.
- Recommandations pour les améliorations futures.

1. Introduction

Dans le cadre du projet de programmation orientée objet pour l'année académique 2023-2024, plusieurs fonctionnalités ont été implémentées, notamment la connexion à la base de données et les opérations sur la table de hachage. Ce rapport vise à expliquer et justifier le choix des différentes fonctions de hachage utilisées dans notre projet.

2. Description des Fonctions de Hachage

Les fonctions de hachage implémentées dans ce projet sont les suivantes :

1. **Fonction de Hachage Division**
2. **Fonction de Hachage Multiplication**
3. **Fonction de Hachage Universelle**

2.1 Fonction de Hachage Division

La fonction de hachage par division utilise la méthode suivante : $h(k) = \mathbf{k} \bmod \mathbf{m}$ où \mathbf{k} est la clé et \mathbf{m} est un nombre premier.

2.2 Fonction de Hachage Multiplication

La fonction de hachage par division utilise la méthode suivante :

$H(k) = \text{partie-entiere de } [\mathbf{m} (\mathbf{kA} \bmod 1)]$

Où A est nombre rationnel (souvent $\mathbf{A} = (\text{sqrt}(5) - 1) / 2$)

2.3 Fonction de Hachage Universelle

$$h(k) = ((a \cdot k + b) \bmod p) \bmod m$$

où a et b sont choisis aléatoirement et p est un nombre premier supérieur à m .

3.

Justification des Choix des Fonctions de Hachage

3.1 Critères de Sélection

Les critères suivants ont été utilisés pour sélectionner les fonctions de hachage :

- **Efficacité** : La fonction doit être rapide à calculer.

- **Uniformité** : La fonction doit distribuer les clés de manière uniforme pour minimiser les collisions.
- **Simplicité** : La fonction doit être simple à implémenter et à comprendre.

3.2 Avantages et Inconvénients

Fonction de Hachage Division

- **Avantages** : Simple à implémenter, rapide.
- **Inconvénients** : Peut entraîner des clusters (regroupement de plusieurs clés différentes qui sont mappées au même indice dans la table de hachage.) si **m** n'est pas bien choisi.

Fonction de Hachage Multiplication

- **Avantages** : Bonne distribution des clés, moins de clusters.
- **Inconvénients** : Légèrement plus complexe à implémenter.

Fonction de Hachage Universelle

- **Avantages** : Offre une très bonne uniformité, minimise les collisions.
- **Inconvénients** : Nécessite un choix aléatoire de **a** et **b**, plus complexe.

3.3 Comparaison avec d'Autres Fonctions

Les fonctions choisies ont été comparées à d'autres méthodes comme la fonction de hachage par addition et la fonction de hachage par extraction, mais elles ont été jugées moins efficaces en termes de distribution et de performance.

4. Tests et Résultats

4.1 Méthodologie de Test

Les fonctions de hachage ont été testées en utilisant un jeu de données simulé pour évaluer leur performance en termes de temps de calcul et de distribution des clés.

4.2 Résultats des Tests

Les résultats montrent que :

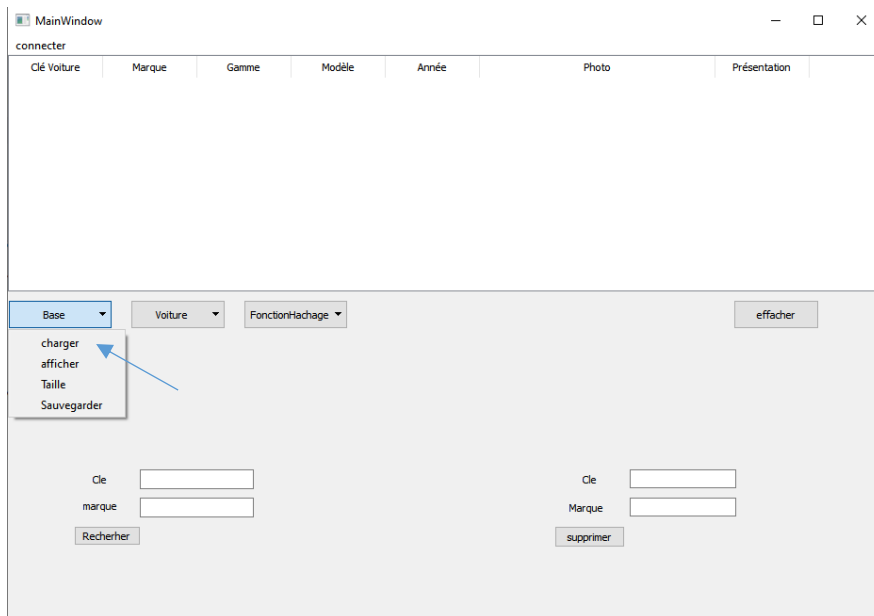
- La fonction de hachage par division est la plus rapide mais a tendance à créer des clusters.
- La fonction de hachage par multiplication offre une meilleure distribution avec un léger coût en temps.
- La fonction de hachage universelle fournit la meilleure uniformité au prix d'une complexité accrue.

4.3 Analyse des Résultats

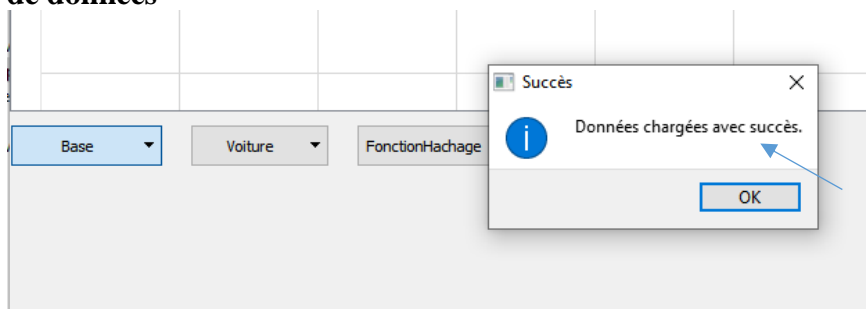
Les fonctions de hachage par multiplication et universelle ont démontré une meilleure performance en termes de distribution des clés, ce qui réduit les collisions et améliore l'efficacité globale.

5 Utilisation de L'application

Ci-dessous on a notre interface avec les différentes options

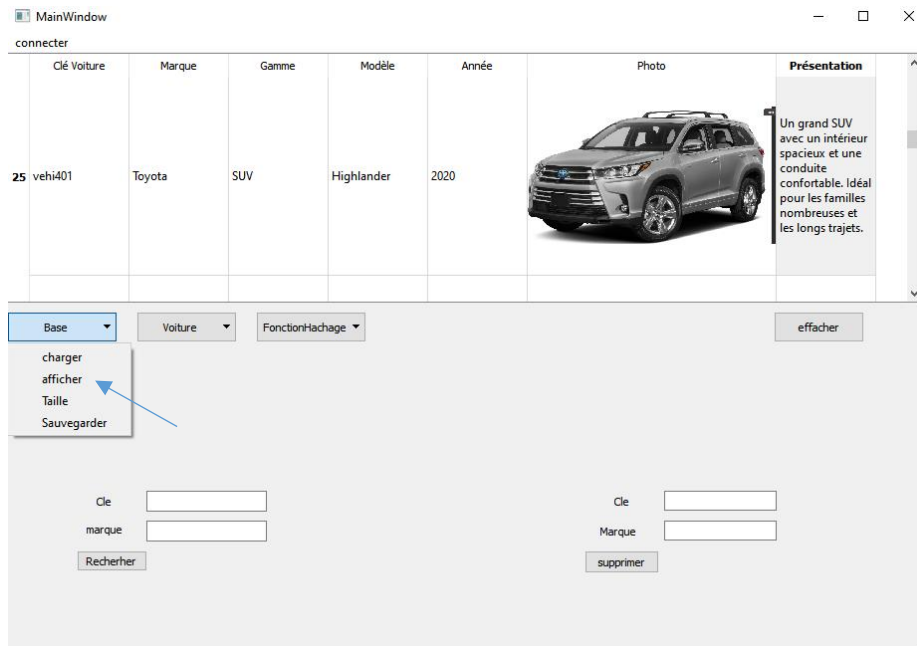


Le buttons charger nous permet ici de charger notre table depuis depuis la bases de données



Après chargement nous avons le message ci-dessus

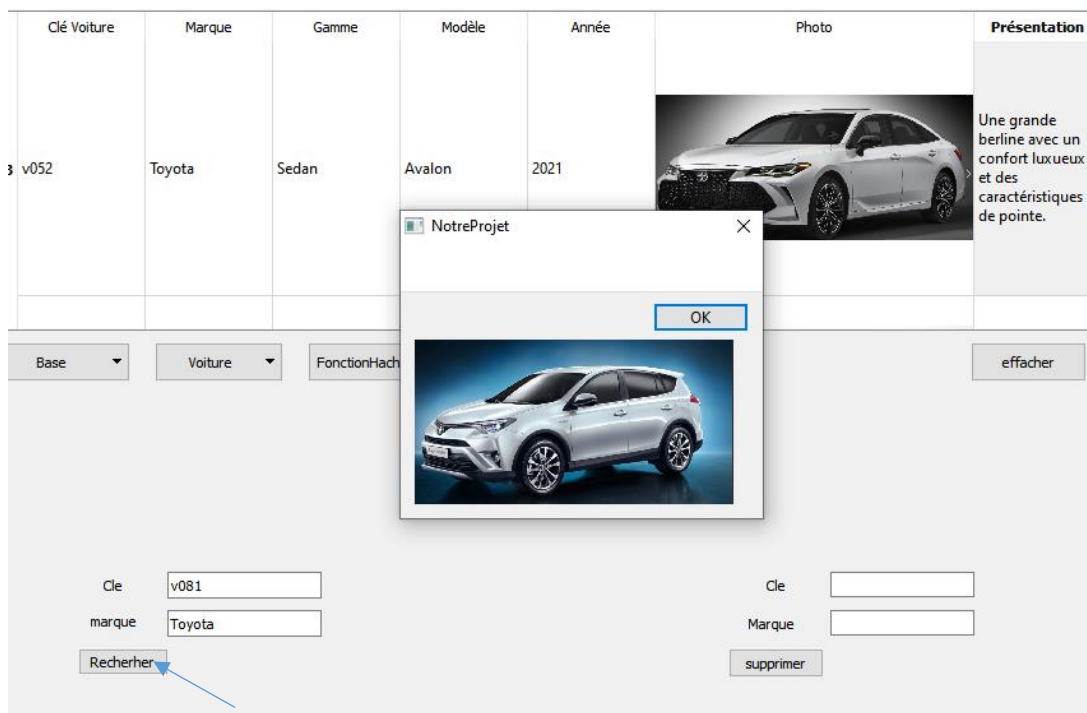
Maintenant après avoir fini de charger notre table on peut afficher toute les informations qu'on a chargé dans la table



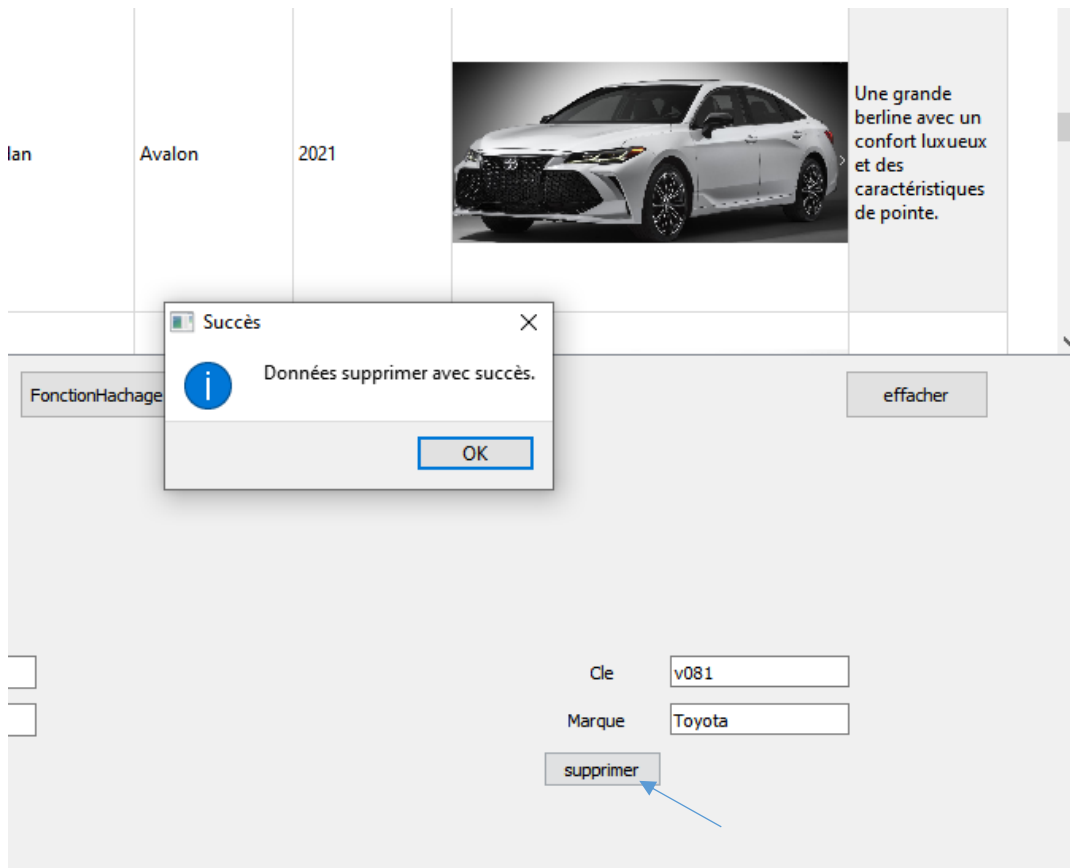
Encore pour les fonctionnalités, on a les options sauvegarder , taille(qui afficher le nombre d'éléments chargés) , rechercher et supprimer

On peut effectuer une recherche, on suppose que la clé de recherche c'est la matricule de la voiture.

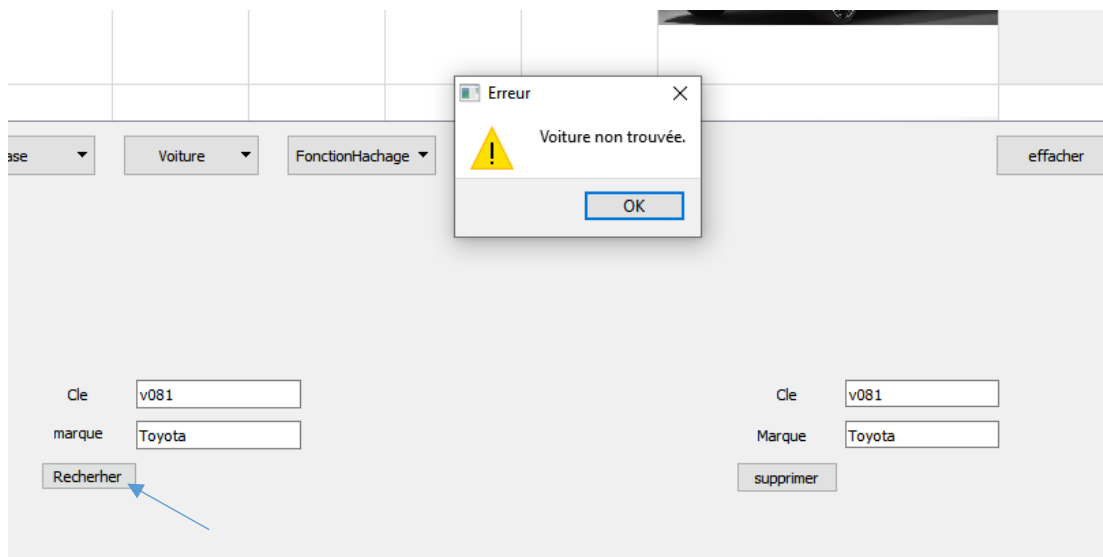
Voici le résultat de notre recherche



On Test maintenant la suppression



Maintenant on verifie si la voiture supprimer est encore dans la table



6. Conclusion

Ce rapport a détaillé les choix des fonctions de hachage utilisées dans le projet en se basant sur des critères de performance, d'uniformité et de simplicité. Les résultats des tests confirment que les fonctions de hachage par multiplication et universelle sont

préférables pour une utilisation dans des applications où la distribution uniforme des clés est cruciale.

Pour les fonctionnalités de l'application , ces dernières sont susceptibles d'être améliorées