

Implémentation d'un moteur de requêtes SQL simples

Bouzidi Belkacem - Elhouiti Chakib
Kezzoul Massili - Zeroual Ramzi
Yang Fei

Université de Montpellier

19 mai 2019

Sommaire

1 Introduction

Présentation du projet et organisation

Présentation du SQL et des bases de données

2 Modélisation

Structure de données

Structure de la requête

3 Implémentation

Les grandes lignes

Le développement

4 Démonstration

5 Conclusion

Perspective



UNIVERSITÉ
DE MONTPELLIER

1 Introduction

Présentation du projet et organisation

Présentation du SQL et des bases de données

2 Modélisation

3 Implémentation

4 Démonstration

5 Conclusion

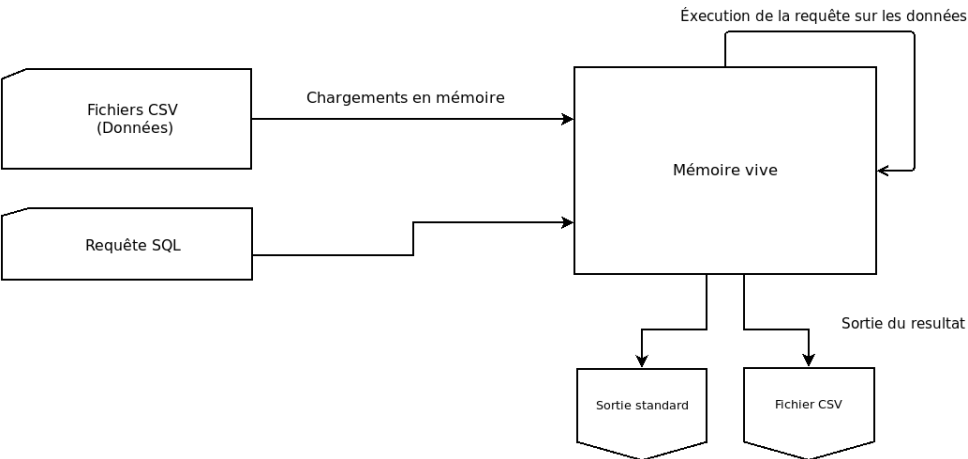
Introduction

Présentation du projet

- ▶ Projet choisi : conception et développement d'un moteur d'évaluation de requêtes SQL en mémoire vive.
- ▶ Forme des requêtes :
 - SELECT : Projection
 - FROM : Jointure
 - WHERE : Selection
- ▶ Fichiers pris en charge : un ou plusieurs fichiers CSV
 - Un fichier CSV est un fichier texte
 - Une Ligne du texte correspond à une ligne du tableau
 - Les virgules correspondent aux séparations entre les colonnes



Objectif de l'application



Introduction

Organisation

- ▶ Réunions :
 - Etudiants : trois à quatre fois par semaine
 - Encadrante : une fois par semaine
- ▶ Decoupage du projet :
 - Phase de modélisation
 - Phase de développement
 - Finalisation du projet
- ▶ Outils de collaboration : GitLab



UNIVERSITÉ
DE MONTPELLIER

Présentation

SQL et les bases de données

Base de données

Une base de données (en anglais database), permet de stocker et de manipuler des données brutes ou d'informations.

Système de gestion de base de données

Un SGBD est un logiciel système servant à stocker, à manipuler ou gérer, et à partager des informations dans une base de données, en garantissant la qualité, la pérennité et la confidentialité des informations, tout en cachant la complexité des opérations.

SGBD les plus utilisés :

- ▶ Oracle Database
- ▶ MySQL
- ▶ ...



UNIVERSITÉ
DE MONTPELLIER

Structured Query Languages

Interpréteur de requêtes SQL

Le SQL est un langage informatique normalisé servant à exploiter des bases de données relationnelles.

Les requêtes SQL considéré par notre programme sont sous cette forme :

```
SELECT nomAttribut1,...,nomAttribut2  
FROM nomTable1,..., nomTable2  
WHERE nomAttributX = nomAttributY OR ... AND nomAttributZ = ZZ
```



① Introduction

② **Modélisation**

Structure de données

Structure de la requête

③ Implémentation

④ Démonstration

⑤ Conclusion

Modélisation

Découpage de la conception en deux parties principales :

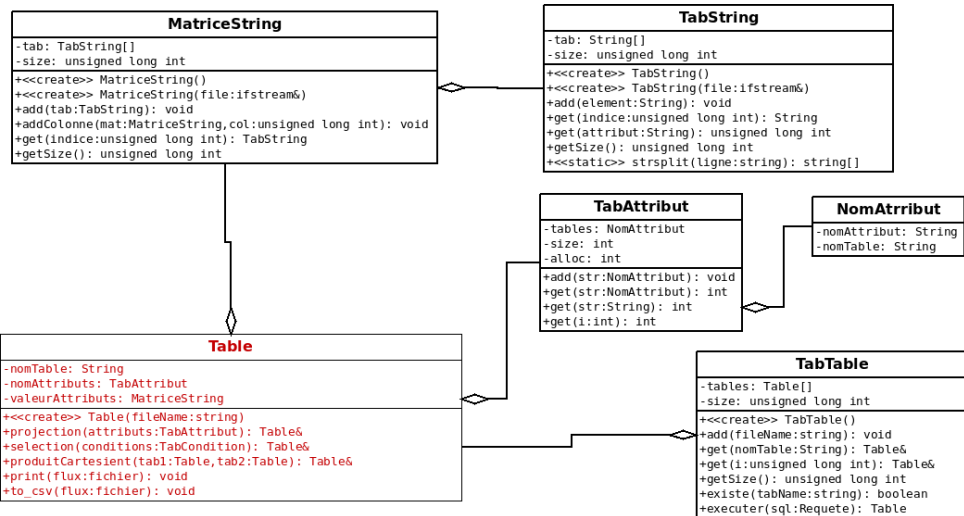
- ▶ Structuration des données
- ▶ Structure de la requête SQL

choix de l'approche

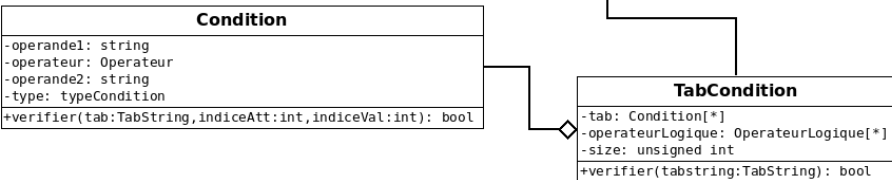
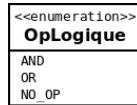
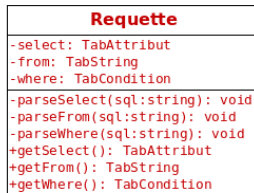
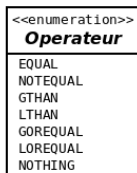
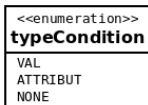
Nous avons choisis de modéliser ces deux sous parties, en utilisant l'approche orientée objet qui est parfaitement adaptée à notre problème.



Structuration des données



Structure de la requête



① Introduction

② Modélisation

③ Implémentation

Les grandes lignes

Le développement

④ Démonstration

⑤ Conclusion

Implémentation

Les grandes lignes de l'implémentation :

- ▶ Choix du C++ comme langage de programmation,
- ▶ Répartition du développement en trois parties principales :
 - Chargement des données,
 - Interprétation et exécution de la requête,
 - Restitution des données.
- ▶ Utilisation du programme,
- ▶ Phases de tests.



UNIVERSITÉ
DE MONTPELLIER

Développement

Chargement des données

- ▶ Implémentation des classes,
- ▶ Interpréter un fichier CSV,
- ▶ Fonction strsplit,
- ▶ Parser toutes les lignes du fichiers CSV,



Développement

La requête

Interpretation

Nous nous sommes occupés à trouver une manière de découper la requête afin de stocker chaque partie dans l'attribut correspondant.

- ▶ Select,
- ▶ From,
- ▶ Where.

Exécution

Notre application exécute la requête en trois étapes consécutives et complémentaires pour effectuer le traitement nécessaire.

- ▶ Le produit cartésien,
- ▶ La sélection,
- ▶ La projection.

- ① Introduction
- ② Modélisation
- ③ Implémentation
- ④ Démonstration**
- ⑤ Conclusion

- ① Introduction
- ② Modélisation
- ③ Implémentation
- ④ Démonstration
- ⑤ Conclusion**
Perspective

Perspective

Voici quelques idées afin d'améliorer le programme :

- ▶ Optimisation en mémoire :
 - Projection partielle
 - Exécuter les conditions de sélection sur chaque tables individuellement
 - Écrire le résultat du produit cartésien dans un fichier temporaire
- ▶ Opérateur :
 - LIKE
 - BETWEEN
 - IN
- ▶ Commandes :
 - GROUP BY
 - HAVING
 - AS
- ▶ Fonctions d'agrégation :
 - MAX et MIN
 - COUNT
 - SUM
 - AVG



UNIVERSITÉ
DE MONTPELLIER