

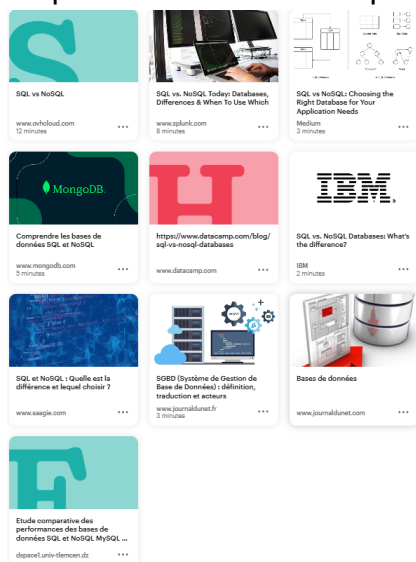
I. Première partie

J'ai choisi le sujet "Les différences entre les bases de données SQL et NoSQL" car il me permet de comprendre les évolutions du métier de développeur, notamment sur la manière de stocker les données et qu'il pourra m'aider à mieux définir les besoins techniques dans mon futur métier.

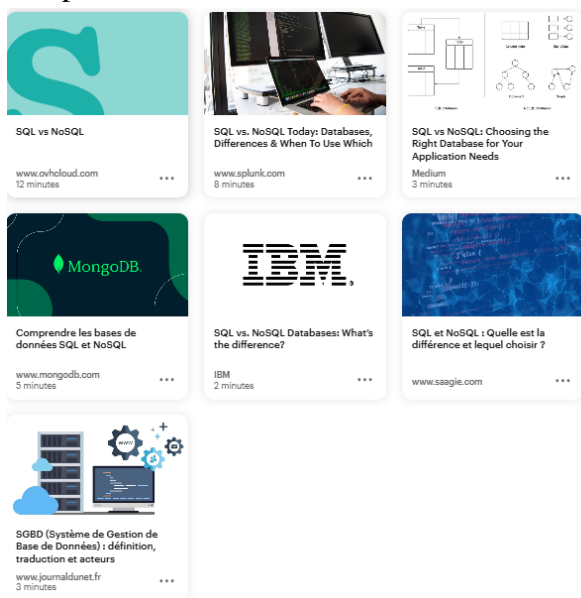
Cette veille a pour but de mettre en lumière les différences entre les bases de données SQL et NoSQL pour comprendre leurs utilisations et spécificités afin de mieux les utiliser dans le monde professionnel.

Pour rédiger cette veille, je me suis aidé de Pocket et Google Alerts.

La phase de collecte m'a permis de trouver 10 articles :



Mais seulement 7 ont été retenus parce qu'ils délimitent le sujet que j'encadre et se complètent :



Ces articles proviennent notamment des Google Alerts ci-dessous mais aussi de recherches personnelles :

Mes alertes (4)

diferencias sql and nosql

les différences entre Sql et NoSql

sql or nosql which is better

sql vs nosql

II. Deuxième partie

Une base de données est un système de stockage d'informations qui permet de gérer, organiser et manipuler des données. Elle permet de stocker de grandes quantités de données et d'offrir un accès rapide à ces dernières en fonction des besoins. Les bases de données sont utilisées pour des applications, des systèmes de gestion d'entreprise, des applications mobiles ou les sites web.

Les bases de données SQL et NoSQL sont deux types fondamentaux de systèmes de gestion de bases de données (SGBD) qui se distinguent principalement par leur manière d'organiser, de stocker et de récupérer les données.

1. SQL (Structured Query Language) :

SQL (Structured Query Language) est un langage standard qui permet d'interagir avec des bases de données relationnelles. Les bases de données SQL suivent un modèle relationnel, où les données sont organisées en tables, qui sont des ensembles de lignes (enregistrements) et de colonnes (attributs). Chaque table a une structure bien définie avec des relations entre les différentes tables.

Caractéristiques principales des bases de données SQL :

- Les données sont regroupées en tables liées entre elles par des clés primaires et des clés étrangères.
- Les tables ont un schéma fixe, ce qui signifie que les types de données et les structures doivent être définis à l'avance.

Exemples de bases de données SQL :

- MySQL
- PostgreSQL
- MS SQL Server
- Oracle Database

2. NoSQL (Not Only SQL) :

NoSQL est un terme générique désignant des systèmes de gestion de bases de données qui ne suivent pas le modèle relationnel classique. Il n'y a donc pas de tableau avec des lignes et des colonnes définies. Les schémas sont donc absents ou flexibles. Cela permet de regrouper des données ayant des structures différentes (document ou JSON).

Caractéristiques principales des bases de données NoSQL :

- NoSQL inclut plusieurs types de bases de données, comme les bases de données documentaires, clé-valeur, colonne ou graphes. Par exemple, MongoDB (document), Oracle NoSQL (JSON).
- Les bases de données NoSQL permettent un schéma flexible, ce qui signifie que les structures des données peuvent évoluer sans contrainte rigide, adaptées à des données semi-structurées ou non structurées.

Exemples de bases de données NoSQL :

- MongoDB
- Apache Cassandra
- Oracle NoSQL
- CouchDB

Différences principales entre SQL et NoSQL :

Critère	SQL (Relationnel)	NoSQL (Non-relationnel)
Modèle de données	Relationnel (tables, lignes, colonnes)	Non-relationnel (documents, clé-valeur, etc.)
Schéma	Schéma rigide et pré-déterminé	Schéma flexible ou sans schéma
Transactions	ACID (Atomicité, Cohérence, Isolation, Durabilité)	BASE (Basically Available, Soft state, Eventually consistent)
Langage de requête	SQL (Structured Query Language)	Souvent un langage spécifique à chaque base
Types de données	Données structurées	Données semi-structurées ou non structurées
Scalabilité	Scalabilité verticale (augmentation de la capacité d'un seul serveur)	Scalabilité horizontale (ajout de serveurs)

Quand utiliser SQL ou NoSQL ?

SQL est idéal pour des applications nécessitant une structure stricte, des transactions complexes et une forte intégrité des données, comme les applications bancaires, les systèmes de gestion d'entreprises ou les logiciels d'analyse de données (Power BI).

NoSQL est souvent choisi pour des applications à grande échelle nécessitant une haute disponibilité, une flexibilité du schéma ou un volume de données très large, comme les réseaux sociaux (Facebook), les applications IoT, ou les systèmes de recommandation.