

Introduction

"Comparaison entre MongoDB et SQL"

- Qu'est-ce qu'une base de données ?
- Une base de données est un système permettant de stocker, organiser et gérer les données.
- Types principaux de bases de données :
 - SQL (relationnelles) : Utilisent des tables structurées avec colonnes et lignes.
 - NoSQL (non relationnelles) : Conçues pour gérer des données non structurées ou semi-structurées.
- Focus de la présentation : Comparaison entre MongoDB (NoSQL) et SQL (relationnel).

Caractéristiques de SQL

"Bases de données relationnelles (SQL)"

- Stockage des données :

- Données organisées en tables avec des relations bien définies.
- Chaque table possède un schéma strict (colonnes et types de données).

- Langage utilisé : SQL (Structured Query Language).

- Exemples de systèmes : MySQL, PostgreSQL, Microsoft SQL Server, Oracle.

- Avantages :

- Bonne gestion des relations complexes entre données.
- Adapté aux transactions nécessitant une forte intégrité des données (ACID).

Caractéristiques de MongoDB

"Bases de données NoSQL (MongoDB)"

- Stockage des données :
 - Données organisées en documents JSON ou BSON.
 - Schéma flexible, idéal pour des données semi-structurées.
- Langage utilisé : Requêtes en JavaScript (MongoDB Query Language - MQL).
- Scalabilité : Scalabilité horizontale (ajout de serveurs pour gérer la charge).
- Avantages :
 - Performances optimales pour des données volumineuses et non structurées.
 - Adapté aux systèmes nécessitant des modifications fréquentes du schéma.

Comparaison entre MongoDB et SQL

- Structure des données : SQL utilise des tables rigides, MongoDB des documents flexibles.
- Schéma : SQL est rigide, MongoDB est adapté aux changements.
- Scalabilité : SQL s'agrandit verticalement, MongoDB horizontalement.
- Cas d'usage : SQL pour les relations complexes, MongoDB pour les données massives et dynamiques.

Conclusion

- Quand utiliser SQL ?
 - Pour des projets nécessitant une forte intégrité des données et des relations complexes.
 - Exemple : Systèmes financiers, gestion d'inventaires.
- Quand utiliser MongoDB ?
 - Pour des projets nécessitant flexibilité, scalabilité et rapidité avec des données non structurées.
 - Exemple : Applications temps réel, systèmes Big Data.