**Titre**: Introduction à SQL et MongoDB

* **SQL** (Structured Query Language):
  + Type: Base de données relationnelle.
  + Structure: Utilise des tables avec des lignes et des colonnes.
  + Modèle de données: Schéma fixe et structuré.
  + Exemples: MySQL, PostgreSQL, Oracle Database, SQL Server.
* **MongoDB**:
  + Type: Base de données NoSQL orientée document.
  + Structure: Utilise des collections et des documents JSON (BSON).
  + Modèle de données: Schéma flexible et non structuré.
  + Exemples: MongoDB, CouchDB, Elasticsearch (pour les documents).

**Titre**: Fonctionnalités de SQL

* **Schéma rigide**:
  + Les données doivent correspondre à un schéma prédéfini.
  + Les relations sont établies via des clés étrangères.
* **Transactions ACID**:
  + Atomicité, Cohérence, Isolation, Durabilité.
  + Garanties fortes pour les transactions de base de données.
* **Puissante capacité de requête**:
  + Utilise SQL pour des opérations complexes (JOINs, GROUP BY, etc.).
  + Adapté aux analyses de données et aux rapports complexes.
* **Écosystème mature**:
  + Outils de gestion robustes et larges options de sécurité.
  + Support pour les procédures stockées, les déclencheurs et les vues.

**Titre**: Fonctionnalités de MongoDB

* **Schéma flexible**:
  + Documents JSON sans schéma strict.
  + Facile à adapter et à changer le modèle de données.
* **Hautement évolutif**:
  + Sharding horizontal pour une mise à l'échelle facile.
  + Particulièrement efficace pour les grandes quantités de données non structurées.
* **Requêtes JSON natives**:
  + Requêtes flexibles via des formats JSON.
  + Supporte les index complexes et les requêtes agrégées.
* **Optimisé pour les données non structurées**:
  + Stockage facile des documents variés (JSON).
  + Idéal pour les applications qui requièrent une adaptation rapide.

**Titre**: Comparaison entre SQL et MongoDB

| **Aspect** | **SQL** | **MongoDB** |
| --- | --- | --- |
| **Modèle de Données** | Schéma fixe (tables, lignes, colonnes) | Schéma flexible (documents JSON) |
| **Transactions** | ACID complet, transactions complexes | Transactions ACID (à partir de MongoDB 4.0) mais moins complexes |
| **Scalabilité** | Scalabilité verticale (ajouter plus de puissance à un serveur unique) | Scalabilité horizontale (sharding) |
| **Requêtes** | SQL pour des requêtes complexes et agrégations | JSON natif, agrégation flexible |
| **Utilisation typique** | Applications avec des relations complexes et la normalisation des données | Applications modernes nécessitant de la flexibilité et de la performance pour les données semi-structurées |

**Titre**: Quand utiliser SQL ou MongoDB?

* **Quand utiliser SQL**:
  + Lorsque la cohérence des transactions est cruciale (ex. : systèmes financiers).
  + Pour des modèles de données bien définis avec des relations complexes.
  + Lorsque des requêtes SQL complexes et des rapports sont nécessaires.
* **Quand utiliser MongoDB**:
  + Pour des applications avec des données non structurées ou semi-structurées.
  + Lorsque la flexibilité du modèle de données est essentielle (par ex., développement agile).
  + Pour les applications nécessitant une mise à l'échelle horizontale et une haute disponibilité (ex. : grandes applications web).