# **Diaporama : Comparaison entre bases de données SQL et NoSQL**

## **Diapositive 1 : Titre**

**Titre :** Comparaison entre bases de données relationnelles (SQL) et non-relationnelles (NoSQL) **Sous-titre :** Analyse comparative pour [Nom du cas d’usage choisi]

## **Diapositive 2 : Objectifs de l’atelier**

* Comprendre les différences entre bases de données SQL et NoSQL.
* Analyser les avantages et inconvénients de chaque type.
* Fournir des recommandations pour [Nom du cas d’usage].

## **Diapositive 3 : Bases relationnelles (SQL)**

* **Caractéristiques principales :**
  + Schéma fixe avec tableaux et relations prédéfinis.
  + Langage SQL pour manipuler les données.
  + Conformité ACID (Atomicité, Cohérence, Isolation, Durabilité).
* **Avantages :**
  + Intégrité des données garantie.
  + Idéal pour des requêtes complexes (jointures, agrégations).
  + Outils bien établis (MySQL, PostgreSQL).
* **Inconvénients :**
  + Scalabilité verticale limitée.
  + Rigidité en cas de changement de schéma.

## **Diapositive 4 : Bases NoSQL**

* **Caractéristiques principales :**
  + Schéma flexible ou inexistant.
  + Types : Clé-valeur, colonnes, documents, graphes.
  + Conformité BASE (Disponibilité de base, État temporaire, Cohérence éventuelle).
* **Avantages :**
  + Scalabilité horizontale facile.
  + Adapté aux données non-structurées.
  + Performances élevées pour les lectures/écritures rapides.
* **Inconvénients :**
  + Cohérence des données non garantie.
  + Moins adapté pour des relations complexes.

## **Diapositive 5 : Cas d’usage choisi**

* **[Nom du cas d’usage]**
* **Exigences métier :**
  + Nature des données : [Structurées/semi-structurées/non-structurées].
  + Volume de données : [Petit/moyen/grand].
  + Performance : [Rapide/évolutive].
  + Changements de schéma : [Fréquents/rares].

## **Diapositive 6 : Analyse des bases relationnelles (SQL)**

* **Points forts :**
  + Gestion des relations complexes entre données.
  + Transactions ACID pour garantir la fiabilité.
  + Idéal pour des opérations complexes (ex : jointures).
* **Limites :**
  + Difficultés à gérer des volumes massifs.
  + Scalabilité verticale coûteuse.

## **Diapositive 7 : Analyse des bases NoSQL**

* **Points forts :**
  + Haute performance pour des écritures massives.
  + Facilité d’adaptation aux données non-structurées.
  + Scalabilité horizontale (ajout facile de serveurs).
* **Limites :**
  + Pas de support natif pour les relations complexes.
  + Cohérence BASE peut poser des problèmes.

## **Diapositive 8 : Comparaison**

|  |  |  |
| --- | --- | --- |
| **Critère** | **SQL** | **NoSQL** |
| **Schéma** | Fixe | Flexible ou absent |
| **Volume de données** | Moyen | Grand |
| **Relations complexes** | Excellent | Faible |
| **Scalabilité** | Verticale | Horizontale |
| **Performance** | Requêtes complexes | Écritures rapides |

## **Diapositive 9 : Recommandation finale**

* **Recommandation :** [SQL/NoSQL/Combinaison]
* **Justifications :**
  + Exigence 1 : [Détail].
  + Exigence 2 : [Détail].
  + …

## **Diapositive 10 : Conclusion**

* Les bases SQL et NoSQL ont chacune leurs forces.
* Le choix doit s’aligner sur les besoins métier spécifiques.
* [SQL/NoSQL] est recommandé pour [Cas d’usage].
* Questions ?