

Solutions pour le Dossier 3: Gérer les données (NoSQL)

Introduction

Ce document présente les solutions aux requêtes MongoDB demandées pour la gestion des données dans une base de données NoSQL, spécifiquement pour la collection terrains. Chaque solution est accompagnée d'une explication concise pour clarifier son objectif.

1 Afficher les notaires qui travaillent à « Rabat », triés par nom croissant

Cette requête utilise l'agrégation pour filtrer les documents où l'adresse du notaire est « Rabat », trie les résultats par nom croissant, et projette uniquement les champs pertinents du notaire.

```
1 db.terrains.aggregate([
2   { $match: { "notaire.Adresse": "Rabat" } },
3   { $sort: { "notaire.nom": 1 } },
4   { $project: {
5     "notaire.Numn": 1,
6     "notaire.nom": 1,
7     "notaire.prenom": 1,
8     "notaire.Adresse": 1,
9     _id: 0
10  }
11 }
12 ])
```

Listing 1: Requête MongoDB pour les notaires à Rabat

2 Afficher le nombre total des terrains pour chaque type

Cette requête groupe les terrains par leur type et compte le nombre total de terrains pour chaque catégorie.

```

1 db.terrains.aggregate([
2   { $group: {
3     _id: "$type",
4     total: { $sum: 1 }
5   }
6 }
7 ])
```

Listing 2: Requête MongoDB pour le nombre de terrains par type

3 Afficher le prix maximum de tous les terrains de type agricole

Cette requête filtre les terrains de type « agricole » et utilise une agrégation pour déterminer le prix maximum.

```

1 db.terrains.aggregate([
2   { $match: { type: "agricole" } },
3   { $group: {
4     _id: null,
5     prix_max: { $max: "$prix" }
6   }
7 }
8 ])
```

Listing 3: Requête MongoDB pour le prix maximum des terrains agricoles

4 Augmenter de 1% le prix de tous les terrains agricoles des propriétaires cin=P1 et cin=P4

Cette requête met à jour les documents pour augmenter de 1% le prix des terrains agricoles appartenant aux propriétaires avec les CIN « P1 » ou « P4 ».

```

1 db.terrains.updateMany(
2   {
3     type: "agricole",
4     cin: { $in: ["P1", "P4"] }
5   },
6   { $mul: { prix: 1.01 } }
7 )
```

Listing 4: Requête MongoDB pour augmenter le prix des terrains agricoles

5 Supprimer l'adresse et le téléphone du notaire numn=N2

Cette requête supprime les champs « Adresse » et « Tel » du notaire avec l'identifiant « N2 » dans tous les documents concernés.

```
1 db.terrains.updateMany(  
2   { "notaire.Numn": "N2" },  
3   { $unset: {  
4     "notaire.Adresse": "",  
5     "notaire.Tel": ""  
6   }  
7 }  
8 )
```

Listing 5: Requête MongoDB pour supprimer les champs du notaire N2

Note

Ces solutions supposent que les données sont stockées dans une collection MongoDB appelée `terrains`. Les requêtes peuvent nécessiter des ajustements en fonction de la structure exacte des documents dans la base de données.