


| | |
|---|---|
|  <small>Se former autrement</small> | <h2 style="margin: 0;">EXAMEN</h2> <p style="margin: 5px 0;">Semestre : 1 <input type="checkbox"/> 2 <input checked="" type="checkbox"/></p> <p style="margin: 5px 0;">Session : Principale <input checked="" type="checkbox"/> Rattrapage <input type="checkbox"/></p> |
| <p>Module : BD Nosql</p> <p>Enseignante : Ines Slimene</p> <p>Classes : 4 TWIN</p> <p>Documents autorisés : OUI <input type="checkbox"/> NON <input checked="" type="checkbox"/></p> <p>Calculatrice autorisée : OUI <input type="checkbox"/> NON <input checked="" type="checkbox"/></p> <p>Date : 05/03/2018 Heure : 09:00</p> | |
| <p>Nombre de pages : 03</p> <p>Internet autorisée : OUI <input type="checkbox"/> NON <input checked="" type="checkbox"/></p> <p>Durée : 01H30</p> | |

Exercice 1 : (4 points)

On souhaite réaliser une base de données orientée documents pour gérer des cours et des étudiants, étant données les informations suivantes :

- Un cours est décrit par les attributs code, titre, description, crédits et prérequis.
- Les prérequis sont d'autres cours.
- Un étudiant est décrit par les attributs nom, prénom, adresse.
- Les adresses sont composées d'un numéro de rue, d'une rue, d'une ville et d'un code postal.
- Un étudiant suit plusieurs cours et les cours sont suivis par plusieurs étudiants.

1. Proposer un exemple JSON qui favorise les références sur l'imbrication.
2. Proposer un exemple JSON qui favorise l'imbrication sur les références. Est-il possible d'avoir une solution ne reposant que sur l'imbrication ?
3. Sachant que l'objectif de l'application est de visualiser une liste des étudiants avec les cours que chacun suit, et d'accéder aux détails des cours uniquement lorsque l'on clique sur son code ou son titre, proposer une solution adaptée à ce problème.

Exercice 2 : (6 points)

1. Quelle est la requête qui permet d'avoir le résultat suivant :

```
{ "title" : "P.S. I Love You" }
{ "title" : "Love Actually" }
{ "title" : "Shakespeare in Love" }
```

- a. `db.movies.find({}, {title:1, _id:0})`
- b. `db.movies.find({}, {title:1})`
- c. `db.movies.find({title:1})`

2. Quelle est la commande qui permet de visualiser l'index utilisé pour l'exécution de la requête :

- a. `db.movie.find({title:"The Matrix"}).explain(executionStats)`
- b. `db.movie.remove({title:"The Matrix"}).explain(executionStats)`
- c. `db.movie.explain(executionStats).find()`

3. La commande suivante permet de créer un index multiple pour la collection foo :

```
db.foo.createIndex( { a : 1, b : 1, c : 1 } )
```

Quelles sont les requêtes qui utilisent cet index

- a. `db.foo.find({b :3,c:4})`
- b. `db.foo.find({c:1}).sort({a :1,b:1})`
- c. `db.foo.find({c:4}).sort({a :-1,b:1})`

4. On a créé un index textuel sur le champ *title* de la collection *movies*. Quel(s) est(sont) le(s) résultat(s) de la requête suivante :

```
db.movies.find( { $text : { $search : "Big Lebowski" } } )
```

- a. `{title : « The big Lebowski »,star:"jeff bridges"}`
- b. `{title : «big», star:"Tom Hanks"}`
- c. `{title : « The big fish»,star:"Ewan McGregor"}`

5. Soit la requête suivante :

```
db.companies.aggregate( [  
  { $match: { "relationships.person": { $ne: null } } },  
  { $project: { relationships: 1, _id: 0 } },  
  { $unwind: "$relationships" },  
  { $group: { _id: "$relationships.person", count: { $sum: 1 }  
    } }, { $sort: { count: -1 } }  
] )
```

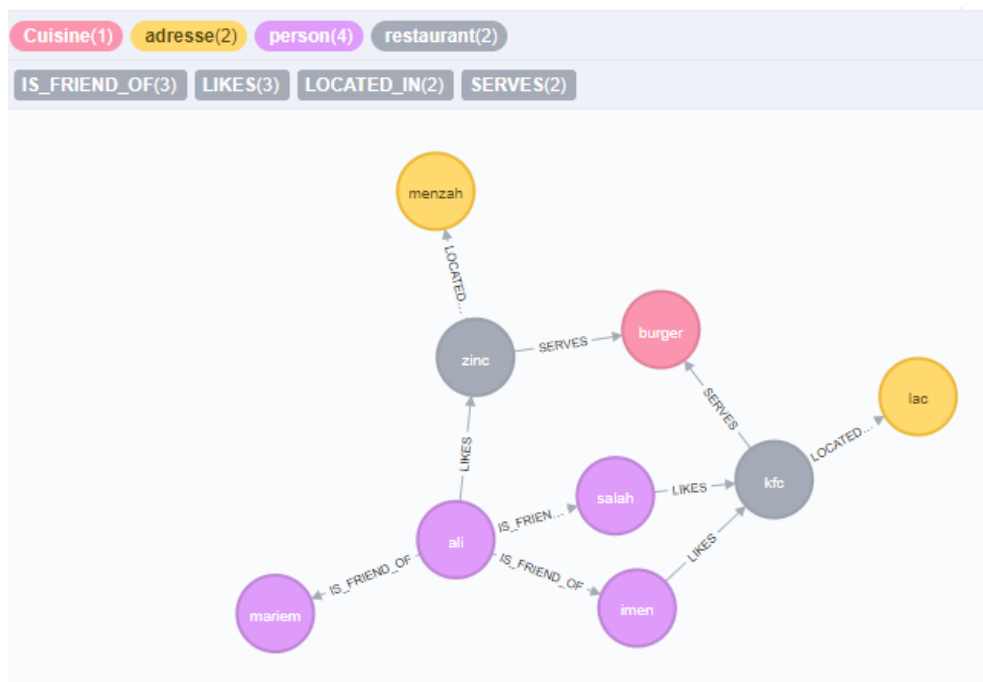
Cette requête permet de :

- a. Afficher le nombre de relations par personne ordonné dans l'ordre décroissant
- b. Afficher la somme de relations par personne ordonné dans l'ordre décroissant
- c. Afficher le nombre de relations nulles par personne dans l'ordre décroissant

6. Quels sont les nœuds qui peuvent participer aux élections du nœud primaire ?
 - a. Regular
 - b. Hidden
 - c. Passive
 - d. arbiter

Exercice 3 : (10 points)

1. Citer les avantages et les inconvénients de chaque type de base de données non relationnelles.
2. Les bases de données non relationnelles ne respectent pas les propriétés ACID des BDR. Pourquoi ?
3. Expliquer le principe de sharding et donner les étapes de sa configuration pour une collection mongodb donnée.
4. Quel est l'avantage de la création d'un index de type *sparse* ou *partialFilterExpression* ?
5. Quels sont les différents types de données gérées par redis ?
6. Est-ce que redis assure la durabilité des données ? expliquer.
7. Soit la base de données orientée graphe suivante :



- a. Quel est le résultat de la requête suivante :

```
$ match (:Cuisine{name:"burger"}) <-[:SERVES]-(:restaurant)-[:LOCATED_IN]->
(a:adresse) return a
```

- b. Ecrire la requête qui permet d'afficher les restaurants aimés par les amis de Ali.