**Gestion Des données - MongoDB**

Exercice 1

Créez une base de données qui porte le nom de votre groupe « exemple (DEV202)

Importez le fichier **restaurants.json** dans une collection qui porte le même nom :

Regardez la requette de chaque questions, ecrivez dans la zone de texte ce qu’elle est sensée faire.

Executez la en utilisant le mongosh en ligne de commande ou sur mongodbcompass puis corrigez votre explication

**Exemple :**

**db.restaurants.find( { "borough" : "Brooklyn" } )**

Cette requête permet d’afficher la liste des restaurants qui se trouvent dans le quartier « Brooklyn »……………………………………………………

………………………………………………………………………

**NB : ne pas faire copier/coller, retappez la requête pour mémoriser la syntaxe et faire/corriger les erreurs.**

**db.restaurants.find( { "borough" : "Brooklyn" } ).count()**

permet d'afficher le nombre des restaurants qui se trouvent dans le quartier Brooklyn

**db.restaurants.find(**

**{ "borough" : "Brooklyn",**

**"cuisine" : "Italian" }**

**)**

Permet d'afficher la liste des restaurants du quartier brooklyn qui preparent la cuisine italienne

**db.restaurants.find(**

**{ "borough" : "Brooklyn",**

**"cuisine" : "Italian",**

**"address.street" : "5 Avenue" }**

**)**

Permet d'afficher la liste des restaurants du quartier brooklyn qui preparent la cuisine italienne et qui se trouvent l'adresse à "5 Avenue"

**db.restaurants.find(**

**{ "borough" : "Brooklyn",**

**"cuisine" : "Italian",**

**"address.street" : "5 Avenue",**

**"name" : /pizza/i }**

**)**

Permet d'afficher la liste des restaurants du quartier brooklyn qui preparent la cuisine italienne et qui se trouvent à l'adresse "5 Avenue" et dont le nom contient le mot pizza "majuscule" ou "miniscule"

Projection

-------------------

**db.getCollection('restaurants').find(**

**{"borough":"Brooklyn",**

**"cuisine":"Italian",**

**"name":/pizza/i,**

**"address.street" : "5 Avenue"},**

**{"name":1})**

Permet d'afficher les noms et les ids des restaurants du quartier brooklyn qui preparent la cuisine italienne et qui se trouvent à l'adresse "5 Avenue" et dont le nom contient le mot pizza "majuscule" ou "miniscule"

regex

--------------------

**db.getCollection('restaurants').find(**

**{"borough":"Brooklyn",**

**"cuisine":"Italian",**

**"name":/pizza/i,**

**"address.street" : "5 Avenue"},**

**{"name" : 1,**

**"grades.score" : 1}**

**)**

Permet d'afficher les noms , les scors et les ids des restaurants du quartier brooklyn qui preparent la cuisine italienne et qui se trouvent à l'adresse "5 Avenue" et dont le nom contient le mot pizza "majuscule" ou "miniscule"

**opérateurs**

**db.getCollection('restaurants').find(**

**{"borough":"Manhattan",**

**"grades.score":{$lt : 10}**

**},**

**{"name":1,"grades.score":1, "\_id":0})**

Permet d'afficher les noms, les scores des restaurants de manhattan qui ont au moins un score inférieur à 10

**db.getCollection('restaurants').find(**

**{"borough":"Manhattan",**

**"grades.score":{**

**$lt:10,**

**$not:{$gte:10}**

**}**

**},**

**{"name":1,"grades.score":1, "\_id":0})**

Permet d'afficher les noms, les scores des restaurants de manhattan qui ont au moins un score inférieur à 10 et qui n'ont aucun score supérieur ou égale à 10

**db.restaurants.find({**

**"grades.grade" : "C",**

**"grades.score" : {$lt : 40}**

**},**

**{"Grades.grade":1, "grades.score":1}**

**);**

Permet d'afficher les ids, les grads et les scores des restaurants qui ont au moins un grade C et qui ont au moins un score inférieur à 40

Permet d'afficher les ids~~, les grads~~ et les scores des restaurants qui ont au moins un grade C et qui ont au moins un score inférieur à 40

**db.restaurants.find({**

**"grades.grade" : "C",**

**"grades.score" : {$lt : 40}**

**},**

**{"grades.grade":1, "grades.score":1}**

**);**

Permet d'afficher les ids, les grads et les scores des restaurants qui ont au moins un grade C et qui ont au moins un score inférieur à 40 (séparés)

**db.restaurants.find({**

**"grades" : {**

**$elemMatch : {**

**"grade" : "C",**

**"score" : {$lt :40}**

**}**

**}**

**},**

**{"grades.grade" : 1,"grades.score" : 1}**

**);**

permet d'afficher les ids, les grades et les scores des restaurant qui ont le grade C et le score inférieur à 40 (ensemble)

**db.restaurants.find({**

**"grades.0.grade":"C"**

**},**

**{"name":1, "borough":1, "\_id":0}**

**);**

Permet d'afficher les nom, les quartiers des restaurants qui ont le premier grade 0

**db.restaurants.distinct("borough")**

permet d'afficher un tableau qui contient les noms des quartiers.

-

permet d'afficher un tableau qui contient les différents noms des quartiers.

**db.restaurants.distinct("grades.grade");**

permet d'afficher un tableau qui contient les différents grades.

aggregation

----------------

**db.restaurants.aggregate([**

**{**

**$group: {**

**\_id: {**

**cuisine: "$cuisine",**

**borough: "$borough"**

**}**

**}**

**},**

**{**

**"$project": {**

**\_id: 0,**

**cuisine: "$\_id.cuisine",**

**borough: "$\_id.borough"**

**}**

**},**

**{ $sort:**

**{**

**borough:1,**

**cuisine:1**

**}**

**}**

**])**

**db.restaurants.aggregate( [**

**{ $match : {**

**"grades.0.grade":"C"**

**}},**

**{ $project : {**

**"name":1, "borough":1, "\_id":0**

**}}**

**] )**

Permet d'afficher la liste des cuisines et des quartiers triés par quartiers ascendant et par cuisine ascendant

Permet **d'afficher** la liste des cuisines par quartier triés par quartiers ascendant et par cuisine ascendant

………………………………………………………………………

………………………………………………………………………

………………………………………………………………………

**varMatch = { $match : { "grades.0.grade":"C"} };**

**varProject = { $project : {"name":1, "borough":1, "\_id":0}};**

**db.restaurants.aggregate( [ varMatch, varProject ] );**

Permet d'afficher les noms et les quartiers des restaurants qui ont le premier grade égale à C "majuscule"

**varSort = { $sort : {"name":1} };**

**db.restaurants.aggregate( [ varMatch, varProject, varSort ] );**

Permet d'afficher les noms et les quartiers des restaurants qui ont le premier grade égale à C "majuscule" triés par nom

**varGroup = { $group : {"\_id" : null, "total" : {$sum : 1} } };**

**db.restaurants.aggregate( [ varMatch, varGroup ] );**

Affiche le total des restaurants qui ont au moins un grade C dans les grades

Affiche le nombre des restaurants qui ont le premier grade égale C "majuscule" dans les grades

**db.restaurants.count({"grades.0.grade":"C"})**

Affiche le nombre des restaurants qui ont le premier grade égale C "majuscule" dans les grades (Obsolète – Deprecated)

**db.restaurants.find({"grades.0.grade":"C"}).count()**

Affiche le nombre des restaurants qui ont le premier grade égale C "majuscule" dans les grades

**varGroup2 = { $group : {"\_id" : "borough", "total" : {$sum : 1} } };**

**db.restaurants.aggregate( [ varMatch, varGroup2 ] );**

permet d'afficher le nombre de restaurants qui ont le premier grade C "majuscule" avec un id qui contient la valeur borough

**varGroup2 = { $group : {"\_id" : "$borough", "total" : {$sum : 1} } };**

**db.restaurants.aggregate( [ varMatch, varGroup2 ] );**

Permet d'afficher le nombre des restaurant qui ont le premier grade égale à "C" majuscule par quartier

**varGroup4 = { $group : {"\_id" : "$borough", "moyenne" : {$avg : "$grades.score"} } };**

**varSort2 = { $sort : { "moyenne" : -1 } }**

**db.restaurants.aggregate( [ varGroup4, varSort2 ] );**

~~permet d'afficher la moyenne des scores des restaurants par quartier dans l'ordre décroissant~~

la fonction aggrégate ne peut pas acceder directement au contenu du tableau des grades

**varUnwind = {$unwind : "$grades"}**

**varGroup4 = { $group : {"\_id" : "$borough", "moyenne" : {$avg : "$grades.score"} } };**

**varSort2 = { $sort : { "moyenne" : -1 } }**

**db.restaurants.aggregate( [ varUnwind, varGroup4, varSort2 ] );**

permet d'afficher la moyenne des scores des restaurants par quartier dans l'ordre décroissant