**NO SQL, Not Only SQL – partie 2**









































**Exercices :**

|  |
| --- |
| **SQL** |
| commande (cde\_id, cli\_id, date, status, prix)  ligne\_cde (cde\_id, prod\_id, qte, prix)  produit (prod\_id, lib) |
| **MongoDB** |
| Collection « commandes »  { cli\_id: "durand01", cde\_date : ISODate("2019-01-02"), status: 'A', prix : 50,  produits: [ { lib : "xxx", qte: 12, prix: 1 }, { lib : "yyy", qte: 19, prix : 2 } ] }  { cli\_id: "castor02", cde\_date : ISODate("2019-01-02"), status: 'B', prix : 47,  produits: [ { lib : "zzz", qte: 8, prix: 5 }, { lib : "xxx", qte: 7, prix : 1 } ] }  { cli\_id: "durand01", cde\_date : ISODate("2019-01-04"), status: 'A', prix : 105,  produits: [ { lib : "aaa", qte: 25, prix: 3 }, { lib : "yyy", qte: 10, prix : 2 }, { lib : "zzz", qte: 2, prix : 5 } ] }  { cli\_id: "durand01", cde\_date : ISODate("2019-01-04"), status: 'A', prix : 15,  produits: [ { lib : "xxx", qte: 15, prix: 1 } ] } |

Ecrire les requêtes MongoDB en utilisant l’agrégation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **MongoDB** | | | **Description** | | |
| db.commandes.aggregate( [  { $group: { \_id: null, count: { $sum: 1 } } } ] ) | | Nombre de commandes | | | |
| db.commandes.aggregate( [  { $group: { \_id: null, total: { $sum: "$prix" } } } ] ) | | Total des commandes | | | |
| db.commandes.aggregate( [  { $group: { \_id: "$cli\_id", total: { $sum: "$prix" } } } ] ) | | Total des commandes par client | | | |
| db.commandes.aggregate( [  { $group: { \_id: "$cli\_id", total: { $sum: "$prix" } } }  **,** { $sort: { total: **-**1 } } ] ) | | Total des commandes par client dans l’ordre décroissant | | | |
| db.commandes.aggregate( [  { $group: { \_id: { \_id: "$cli\_id",  cde\_date: { $dateToString:  { format: "%Y/%m/%d", date: "$cde\_date" } }  }  **,** total: { $sum: "$prix" }  } }  **,** { $sort: { \_id :1 } }  ] ) | | | Total des commandes par client et date par ordre croissant de n° client et date de commande | | |
| db.commandes.aggregate( [  { $group: { \_id: "$cli\_id", count: { $sum: 1 } } }  , { $match: { count: { $gt: 1 } } }  ] ) | | Nombre de commandes pour chaque client ayant passé au moins 2 commandes | | | |
| db.commandes.aggregate( [  { $group: { \_id: {  \_id: "$cli\_id",  cde\_date: { $dateToString:  { format: "%Y/%m/%d", date: "$cde\_date" }  }  }  , total: { $sum: "$prix" }  } }  , { $match: { total: { $gt: 100 } } }  ] ) | | | | Total des commandes « prix » par client et date pour un total dépassant 100 euros | |
| db.commandes.aggregate( [  { $match: { status: 'A' } },  { $group: { \_id: "$cli\_id", total: { $sum: "$prix" } } }  ] ) | Total des commandes avec un statut A | | | | |
| db.commandes.aggregate( [  { $match: { status: 'A' } },  { $group: { \_id: "$cli\_id", total: { $sum: "$prix" } } }  , { $match: { total: { $gt: 100 } } }  ] ) | Total des commandes avec un statut A avec un total dépassant 100 euros | | | | |
| db.commandes.aggregate( [  { $unwind: "$produits" },  { $group: { \_id: "$cli\_id", qte: { $sum: "$produits.qte" } } }  ] ) | | | Total des quantités commandées par client | | |
| db.commandes.aggregate( [  { $group: { \_id: {  \_id: "$cli\_id",  cde\_date: { $dateToString:  { format: "%Y/%m/%d", date: "$cde\_date" }  }  }  }  }  ,{  $group: {  \_id: "$\_id.cde\_date",  nb: { $sum: 1 }  }  } ] ) | | | | | Nombre de client différents qui ont commandé pour chaque jour où il y a eu une ou plusieurs commandes |

**Cas pratique**

**Base utilisée « publis »**

Exemples de document :

{

"\_id" : ObjectId("5b87df2c3b6c3a27dcc951ff"), "type" : "Article", "title" : "SmartKom-Public.",

"pages" : { "start" : 471,"end" : 492 },

"year" : 2006, "booktitle" : "SmartKom",

"url" : "db/series/cogtech/54023732.html#HorndaschRR06",

"authors" : [

"Axel Horndasch",

"Horst Rapp",

"Hans Rittger"

]

}

Exprimez des requêtes simples pour les recherches suivantes :

1. Trier les publications de “Toru Ishida” par titre de livre et par page de début
2. Projeter le résultat sur le titre de la publication, et les pages
3. Compter le nombre de ses publications
4. Compter le nombre de publications depuis 2011 et par type
5. Compter le nombre de publications par auteur et trier le résultat par ordre croissant
6. Pour chaque livre, donner le titre et le nombre de ses auteurs dans l’ordre décroissant du nombre d’auteurs par livre.
7. Donner le titre des livres où il y a plusieurs auteurs dans l’ordre décroissant du nombre d’auteurs par livre.
8. Donner le nombre de livres où il y a plusieurs auteurs.
9. Donner le nombre de livres pour chaque nombre d’auteurs (6171 livres ont été écrits par un seul auteur, 2764 livres par deux auteurs, etc).
10. Donner les pourcentages du nombre de livres pour chaque nombre d’auteurs par rapport au nombre total de livres.

















