

Premières requêtes MongoDB

Exercice 1

Créez une nouvelle base de données nommée `Devoir1` et vérifiez qu'elle est sélectionnée.

```
gescom> use Devoir1
switched to db Devoir1
Devoir1> db.createCollection('produits')
{ ok: 1 }
```

Exercice 2

Créez une nouvelle collection nommée `produits` et y insérer le document suivant:

- nom: Macbook Pro
- fabricant: Apple
- prix: 1299.99
- options:
 - Intel Core i5
 - Retina Display
 - Long life battery

```
Devoir1> db.produits.insert({nom:"Macbook Pro", fabricant:"Apple", prix:1299.99, options:[
Intel Core i5,"Retina Display","Long life battery"]})
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("61b0c75e8f452b218c002de0") }
}
Devoir1> db.produits.find()
[
  {
    _id: ObjectId("61b0c75e8f452b218c002de0"),
    nom: 'Macbook Pro',
    fabricant: 'Apple',
    prix: 1299.99,
    options: [ 'Intel Core i5', 'Retina Display', 'Long life battery' ]
  }
]
```

Rajoutez un autre document dans `produits`:

- nom: Macbook Air
- fabricant: Apple
- prix: 1099.99

- ultrabook: true
- options:
 - Intel Core i7
 - SSD
 - Long life battery

Et enfin un dernier document:

- nom: Thinkpad X230
- fabricant: Lenovo
- prix: 999.99
- ultrabook: true
- options:
 - Intel Core i5
 - SSD
 - Long life battery

```
Devoir1> db.produits.insert([{nom:"Macbook Air", fabricant:"Apple", prix:1099.99, ultrabook:true, options:["Intel Core i7","SSD","Long life battery"]},{nom:"Thinkpad X230", fabricant:"Lenovo", prix:999.99, ultrabook:true, options:["Intel Core i5", "SSD", "Longlife battery"]}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("61b0c95f8f452b218c002de1"),
    '1': ObjectId("61b0c95f8f452b218c002de2")
  }
}
```

Exercice 3

Effectuez les requêtes de lecture suivantes:

1. Récupérez tous les produits.

```

Devoir1> db.produits.find()
[
  {
    _id: ObjectId("61b0c75e8f452b218c002de0"),
    nom: 'Macbook Pro',
    fabricant: 'Apple',
    prix: 1299.99,
    options: [ 'Intel Core i5', 'Retina Display', 'Long life battery' ]
  },
  {
    _id: ObjectId("61b0c95f8f452b218c002de1"),
    nom: 'Macbook Air',
    fabricant: 'Apple',
    prix: 1099.99,
    ultrabook: true,
    options: [ 'Intel Core i7', 'SSD', 'Long life battery' ]
  },
  {
    _id: ObjectId("61b0c95f8f452b218c002de2"),
    nom: 'Thinkpad X230',
    fabricant: 'Lenovo',
    prix: 999.99,
    ultrabook: true,
    options: [ 'Intel Core i5', 'SSD', 'Longlife battery' ]
  }
]

```

2. Récupérez le premier produit

```

Devoir1> db.produits.find().limit(1)
[
  {
    _id: ObjectId("61b0c75e8f452b218c002de0"),
    nom: 'Macbook Pro',
    fabricant: 'Apple',
    prix: 1299.99,
    options: [ 'Intel Core i5', 'Retina Display', 'Long life battery' ]
  }
]

```

3. Trouvez l'id du Thinkpad et faites la requête pour récupérer ce produit avec son id.

```

Devoir1> db.produits.find({nom:/^Thinkpad/})
[
  {
    _id: ObjectId("61b0c95f8f452b218c002de2"),
    nom: 'Thinkpad X230',
    fabricant: 'Lenovo',
    prix: 999.99,
    ultrabook: true,
    options: [ 'Intel Core i5', 'SSD', 'Longlife battery' ]
  }
]
Devoir1> db.produits.find({_id:ObjectId("61b0c75e8f452b218c002de0")})
[
  {
    _id: ObjectId("61b0c75e8f452b218c002de0"),
    nom: 'Macbook Pro',
    fabricant: 'Apple',
    prix: 1299.99,
    options: [ 'Intel Core i5', 'Retina Display', 'Long life battery' ]
  }
]

```

- Récupérez les produits dont le prix est supérieur à 1200\$

```

Devoir1> db.produits.find({"prix":{$gt:1200}})
[
  {
    _id: ObjectId("61b0c75e8f452b218c002de0"),
    nom: 'Macbook Pro',
    fabricant: 'Apple',
    prix: 1299.99,
    options: [ 'Intel Core i5', 'Retina Display', 'Long life battery' ]
  }
]

```

- Récupérez le premier produit ayant le champ ultrabook à true

```

Devoir1> db.produits.find({"ultrabook":true}).limit(1)
[
  {
    _id: ObjectId("61b0c95f8f452b218c002de1"),
    nom: 'Macbook Air',
    fabricant: 'Apple',
    prix: 1099.99,
    ultrabook: true,
    options: [ 'Intel Core i7', 'SSD', 'Long life battery' ]
  }
]

```

Avec findOne :

```
Devoir1> db.produits.findOne({"ultrabook":true})
{
  _id: ObjectId("61b0c95f8f452b218c002de1"),
  nom: 'Macbook Air',
  fabricant: 'Apple',
  prix: 1099.99,
  ultrabook: true,
  options: [ 'Intel Core i7', 'SSD', 'Long life battery' ]
}
```

6. Récupérez le premier produit dont le nom contient Macbook

```
Devoir1> db.produits.findOne({"nom":/\bMacbook\b/})
{
  _id: ObjectId("61b0c75e8f452b218c002de0"),
  nom: 'Macbook Pro',
  fabricant: 'Apple',
  prix: 1299.99,
  options: [ 'Intel Core i5', 'Retina Display', 'Long life battery' ]
}
```

OU :

```
Devoir1> db.produits.find({"nom":{"$regex:/Macbook/}})
```

7. Récupérez les produits dont le nom commence par Macbook

```
Devoir1> db.produits.find({"nom":/^Macbook/})
[
  {
    _id: ObjectId("61b0c75e8f452b218c002de0"),
    nom: 'Macbook Pro',
    fabricant: 'Apple',
    prix: 1299.99,
    options: [ 'Intel Core i5', 'Retina Display', 'Long life battery' ]
  },
  {
    _id: ObjectId("61b0c95f8f452b218c002de1"),
    nom: 'Macbook Air',
    fabricant: 'Apple',
    prix: 1099.99,
    ultrabook: true,
    options: [ 'Intel Core i7', 'SSD', 'Long life battery' ]
  }
]
```

Exercice 4

1. Supprimez les deux produits dont le fabricant est Apple.

```
Devoir1> db.produits.remove({fabriquant:"Apple"})
DeprecationWarning: Collection.remove() is deprecated. Use deleteOne, deleteMany, findOneAndDelete, or bulkWrite.
{ acknowledged: true, deletedCount: 2 }
Devoir1> db.produits.find({fabriquant:"Apple"})
Devoir1>
```

2. Supprimez le Lenovo X230 en utilisant uniquement son id.

```
Devoir1> db.produits.find()
[
  {
    _id: ObjectId("61b0c95f8f452b218c002de2"),
    nom: 'Thinkpad X230',
    fabriquant: 'Lenovo',
    prix: 999.99,
    ultrabook: true,
    options: [ 'Intel Core i5', 'SSD', 'Longlife battery' ]
  }
]
Devoir1> db.produits.remove({_id:ObjectId("61b0c95f8f452b218c002de2")})
{ acknowledged: true, deletedCount: 1 }
Devoir1> db.produits.find()
Devoir1>
```

Exercice 5

Dans cet exercice, nous allons modéliser un système de facturation très simple.

Voici deux factures:

Facture numéro 10012A:

- Date de facture: 2013-07-04
- Client:
 - Nom: Alexandre Terrasa
 - Courriel: foobar@example.com
- Liste des produits (2 produits):
 - Code: MACBOOKAIR
 - Nom: Macbook Air
 - Prix: 999.99
 - Quantite: 1
 - Code: APPLESUPPORT
 - Nom: AppleCare 1 an
 - Prix: 149.99
 - Quantite: 1
- Total: 1149.98

Facture numéro 10013A:

- Date de facture: 2013-07-05
- Client:
 - Nom: Jacques Berger
 - Courriel: berger.jacques@uqam.ca
- Liste des produits (1 produit) :
 - Code: LENOVOX230
 - Nom: Lenovo Thinkpad X230
 - Prix: 899.99
 - Quantité: 1
- Total : 899.99

Effectuez les actions suivantes:

1. Insérer les deux factures dans la base

```
Devoir1> db.createCollection('factures')
{ ok: 1 }
Devoir1> show collections
factures
produits
```

```
Devoir1> db.factures.insert({ _id: '10012A', date: new Date('2012-07-04'), client: {nom: "Alexandre Terrasa", courriel: "foobar@example.com"}, listeProduits: [{code: "MACBOOKAIR", nom: "Macbook Air", prix: 999.99, quantite: 1},{code:"APPLESUPPORT", nom: "AppleCare 1 an", prix: 149.99, quantite:1}], total: 1149.98})
{ acknowledged: true, insertedIds: { '0': '10012A' } }
```

```
Devoir1> db.factures.insert({ _id: '10013A', date: new Date('2013-07-05'), client: {nom: "Jacques Berger", courriel: "berger.jacques@uqam.ca"}, listeProduits: [{code:"LENOVOX230", nom: "Lenovo Thinkpad X230", prix: 899.99, quantite:1}], total: 899.99})
{ acknowledged: true, insertedIds: { '0': '10013A' } }
```

```

Devoir1> db.factures.find()
[
  {
    _id: '10012A',
    date: ISODate("2012-07-04T00:00:00.000Z"),
    client: { nom: 'Alexandre Terrasa', courriel: 'foobar@example.com' },
    listeProduits: [
      {
        code: 'MACBOOKAIR',
        nom: 'Macbook Air',
        prix: 999.99,
        quantite: 1
      },
      {
        code: 'APPLESUPPORT',
        nom: 'AppleCare 1 an',
        prix: 149.99,
        quantite: 1
      }
    ],
    total: 1149.98
  },
  {
    _id: '10013A',
    date: ISODate("2013-07-05T00:00:00.000Z"),
    client: { nom: 'Jacques Berger', courriel: 'berger.jacques@uquam.ca' },
    listeProduits: [
      {
        code: 'LENOVOX230',
        nom: 'Lenovo Thinkpad X230',
        prix: 899.99,
        quantite: 1
      }
    ],
    total: 899.99
  }
]

```

2. Récupérer la facture avec le numéro est 10013A

```

Devoir1> db.factures.find({_id: '10013A'})
[
  {
    _id: '10013A',
    date: ISODate("2013-07-05T00:00:00.000Z"),
    client: { nom: 'Jacques Berger', courriel: 'berger.jacques@uquam.ca' },
    listeProduits: [
      {
        code: 'LENOVOX230',
        nom: 'Lenovo Thinkpad X230',
        prix: 899.99,
        quantite: 1
      }
    ],
    total: 899.99
  }
]

```

3. Modifier la facture 10012A en changeant la date pour le 2013-07-03 et le courriel du contact pour alex@example.com


```

Devair1> db.factures.updateOne({_id:'10012A'},{$set:{date: new Date('2013-07-03'), "client.courriel": 'alex@example.com'}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
Devair1> db.factures.find({_id:'10012A'})
[
  {
    _id: '10012A',
    date: ISODate("2013-07-03T00:00:00.000Z"),
    client: { nom: 'Alexandre Terrasa', courriel: 'alex@example.com' },
    listeProduits: [
      {
        code: 'MACBOOKAIR',
        nom: 'Macbook Air',
        prix: 999.99,
        quantite: 1
      },
      {
        code: 'APPLESUPPORT',
        nom: 'AppleCare 1 an',
        prix: 149.99,
        quantite: 1
      }
    ],
    total: 1149.98
  }
]

```

4. Récupérer la facture avec le produit vendu ayant un code `LENOVOX230`

```

Devair1> db.factures.find({"listeProduits.code":'LENOVOX230'})
[
  {
    _id: '10013A',
    date: ISODate("2013-07-05T00:00:00.000Z"),
    client: { nom: 'Jacques Berger', courriel: 'berger.jacques@uquam.ca' },
    listeProduits: [
      {
        code: 'LENOVOX230',
        nom: 'Lenovo Thinkpad X230',
        prix: 899.99,
        quantite: 1
      }
    ],
    total: 899.99
  }
]

```

5. Supprimer la facture `10012A`

```
Devoir1> db.factures.remove({_id:'10012A'})
{ acknowledged: true, deletedCount: 1 }
Devoir1> db.factures.find()
[
  {
    _id: '10013A',
    date: ISODate("2013-07-05T00:00:00.000Z"),
    client: { nom: 'Jacques Berger', courriel: 'berger.jacques@uquam.ca' },
    listeProduits: [
      {
        code: 'LENOVOX230',
        nom: 'Lenovo Thinkpad X230',
        prix: 899.99,
        quantite: 1
      }
    ],
    total: 899.99
  }
]
```