

Introduction et Configuration

GitHub du projet : https://github.com/lclerc561/DWM_ParadigmeBDD.git

Ce TP a pour objectif de prendre en main une base de données NoSQL orientée documents : MongoDB. L'environnement a été déployé via Docker avec deux services :

1. Un conteneur MongoDB pour stocker les données.
2. Un conteneur PHP (avec l'extension mongodb et la librairie installée via Composer) pour exécuter les scripts.

Nous avons travaillé sur la base de données chopizza, composée de deux collections : produits et recettes. Les données ont été importées via l'utilitaire mongoimport ou MongoDB Compass. Lien

I. Requêtes simples sur Mongoshell

Nous avons d'abord interagi directement avec la base via le shell pour tester la syntaxe des requêtes (CRUD).

1. **Liste des produits** : Affichage simple de tous les documents de la collection produits.

`db.produits.find()`

```
chopizza> db.produits.find()
[
  {
    _id: ObjectId('6543ac35a08f42f1cc02e80d'),
    numero: 1,
    libelle: 'Margherita',
    description: 'Tomate, mozzarella, basilic',
    image: 'https://www.dominos.fr/ManagedAssets/FR/product/PZSO.png',
    categorie: 'Pizzas',
    tarifs: [
      { taille: 'grande', tarif: 11.99 },
      { taille: 'normale', tarif: 8.99 }
    ],
    recettes: [
      ObjectId('6543b503f0e418df120c79b8'),
      ObjectId('6543b503f0e418df120c79ba'),
      ObjectId('6543b503f0e418df120c79a2')
    ]
  },
  {
    _id: ObjectId('6543ac35a08f42f1cc02e811'),
    numero: 5,
    libelle: 'cola',
    description: 'Cola-Calo, producteur local',
    image: 'https://www.dominos.fr/ManagedAssets/FR/product/BOCO.png',
    categorie: 'Boissons',
    tarifs: [
      { taille: 'grande', tarif: 3.99 },
      { taille: 'normale', tarif: 2.99 }
    ]
  }
]
```

2. **Comptage** : Utilisation de la fonction `countDocuments()` pour vérifier le nombre d'entrées

`db.produits.countDocuments()`

```
chopizza> db.produits.countDocuments()  
11
```

3. **Tri décroissant** :

Nous avons trié les résultats sur le champ `numero` en ordre inverse (-1).

`db.produits.find().sort({ numero: -1 })`

```
chopizza> db.produits.find().sort({ numero: -1 })  
[  
  {  
    _id: ObjectId('695e662cb2aa66b7e2071fa2'),  
    numero: 999,  
    libelle: 'Pizza Speciale Dev',  
    categorie: 'Pizzas',  
    description: 'Une pizza créée par PHP',  
    tarifs: [ { taille: 'normale', tarif: 12.5 } ]  
  },  
  {  
    _id: ObjectId('6543ac35a08f42f1cc02e816'),  
    numero: 10,  
    libelle: 'panna cotta',  
    description: 'panna cotta, la meilleure de la place',  
    image: 'https://www.dominos.fr/ManagedAssets/FR/product/DEPA.png',  
    categorie: 'Desserts',  
    tarifs: [  
      { taille: 'grande', tarif: 5.99 },  
      { taille: 'normale', tarif: 4.99 }  
    ],  
    recettes: [  
      ObjectId('6543b503f0e418df120c79a7'),  
      ObjectId('6543b503f0e418df120c79b6'),  
      ObjectId('6543b503f0e418df120c79ac'),  
      ObjectId('6543b503f0e418df120c79b4'),  
      ObjectId('6543b503f0e418df120c79a2')  
    ]  
  },  
]
```

4. **Recherche par attribut simple** Récupération du document unique correspondant à la pizza "Margherita".

`db.produits.findOne({ libelle: "Margherita" })`

```
chopizza> db.produits.findOne({ libelle: "Margherita" })
{
  _id: ObjectId('6543ac35a08f42f1cc02e80d'),
  numero: 1,
  libelle: 'Margherita',
  description: 'Tomate, mozzarella, basilic',
  image: 'https://www.dominos.fr/ManagedAssets/FR/product/PZSO.png',
  categorie: 'Pizzas',
  tarifs: [
    { taille: 'grande', tarif: 11.99 },
    { taille: 'normale', tarif: 8.99 }
  ],
  recettes: [
    ObjectId('6543b503f0e418df120c79b8'),
    ObjectId('6543b503f0e418df120c79ba'),
    ObjectId('6543b503f0e418df120c79a2')
  ]
}
```

5. **Filtrage par catégorie** Sélection de tous les produits appartenant à la catégorie "Boissons"

`db.produits.find({ categorie: "Boissons" })`

```
chopizza> db.produits.find({ categorie: "Boissons" })
[
  {
    _id: ObjectId('6543ac35a08f42f1cc02e811'),
    numero: 5,
    libelle: 'cola',
    description: 'Cola-Calo, producteur local',
    image: 'https://www.dominos.fr/ManagedAssets/FR/product/BOCO.png',
    categorie: 'Boissons',
    tarifs: [
      { taille: 'grande', tarif: 3.99 },
      { taille: 'normale', tarif: 2.99 }
    ],
    recettes: [
      ObjectId('6543b503f0e418df120c79b9'),
      ObjectId('6543b503f0e418df120c79a3'),
      ObjectId('6543b503f0e418df120c79be'),
      ObjectId('6543b503f0e418df120c79a5')
    ]
  },
  {
    _id: ObjectId('6543ac35a08f42f1cc02e812'),
    numero: 6,
    libelle: 'eau',
    description: 'eau plate, eau pétillante ou aromatisée',
    image: 'https://www.dominos.fr/ManagedAssets/FR/product/BOEA.png',
    categorie: 'Boissons',
    tarifs: [
      { taille: 'grande', tarif: 2.99 },
      { taille: 'normale', tarif: 1.99 }
    ]
  }
]
```

6. **Projection (Sélection de champs)** : Nous avons affiché uniquement les champs categorie, numero et libelle (en excluant l'ID par défaut).

`db.produits.find({}, { categorie: 1, numero: 1, libelle: 1, _id: 0 })`

```
chopizza> db.produits.find({}, { categorie: 1, numero: 1, libelle: 1, _id: 0 })
[
  { numero: 1, libelle: 'Margherita', categorie: 'Pizzas' },
  { numero: 5, libelle: 'cola', categorie: 'Boissons' },
  { numero: 6, libelle: 'eau', categorie: 'Boissons' },
  { numero: 4, libelle: 'Pepperoni', categorie: 'Pizzas' },
  { numero: 2, libelle: 'Reine', categorie: 'Pizzas' },
  { numero: 3, libelle: 'Savoyarde', categorie: 'Pizzas' },
  { numero: 8, libelle: 'salade tomate', categorie: 'Salades' },
  { numero: 9, libelle: 'tiramisu', categorie: 'Desserts' },
  { numero: 10, libelle: 'panna cotta', categorie: 'Desserts' },
  { numero: 7, libelle: 'salade verte', categorie: 'Salades' },
  { numero: 999, libelle: 'Pizza Speciale Dev', categorie: 'Pizzas' }
]
```

7. **Projection avec tableaux** : Même requête en ajoutant l'affichage du tableau tarifs.

`db.produits.find({}, { categorie: 1, numero: 1, libelle: 1, tarifs: 1, _id: 0 })`

```
chopizza> db.produits.find({}, { categorie: 1, numero: 1, libelle: 1, tarifs: 1, _id: 0 })
[
  {
    numero: 1,
    libelle: 'Margherita',
    categorie: 'Pizzas',
    tarifs: [
      { taille: 'grande', tarif: 11.99 },
      { taille: 'normale', tarif: 8.99 }
    ]
  },
  {
    numero: 5,
    libelle: 'cola',
    categorie: 'Boissons',
    tarifs: [
      { taille: 'grande', tarif: 3.99 },
      { taille: 'normale', tarif: 2.99 }
    ]
  },
  {
    numero: 6,
    libelle: 'eau',
    categorie: 'Boissons',
    tarifs: [
      { taille: 'grande', tarif: 2.99 },
      { taille: 'normale', tarif: 1.99 }
    ]
  }
]
```

8. **Filtrage dans un tableau imbriqué (Dot notation)** : Nous avons cherché les produits ayant au moins un tarif inférieur à 8.0€ en utilisant la notation pointée "tarifs.tarif".

`db.produits.find({ "tarifs.tarif": { $lt: 8.0 } })`

```
chopizza> db.produits.find({ "tarifs.tarif": { $lt: 8.0 } })
[
  {
    _id: ObjectId('6543ac35a08f42f1cc02e811'),
    numero: 5,
    libelle: 'cola',
    description: 'Cola-Calo, producteur local',
    image: 'https://www.dominos.fr/ManagedAssets/FR/product/BOCO.png',
    categorie: 'Boissons',
    tarifs: [
      { taille: 'grande', tarif: 3.99 },
      { taille: 'normale', tarif: 2.99 }
    ],
    recettes: [
      ObjectId('6543b503f0e418df120c79b9'),
      ObjectId('6543b503f0e418df120c79a3'),
      ObjectId('6543b503f0e418df120c79be'),
      ObjectId('6543b503f0e418df120c79a5')
    ]
  },
  {
    _id: ObjectId('6543ac35a08f42f1cc02e812'),
    numero: 6,
    libelle: 'eau',
    description: 'eau plate, eau pétillante ou aromatisée',
    image: 'https://www.dominos.fr/ManagedAssets/FR/product/BOEA.png',
    categorie: 'Boissons',
    tarifs: [
      { taille: 'grande', tarif: 2.99 },
      { taille: 'normale', tarif: 1.99 }
    ],
  },
]
```

9. **Filtrage complexe (\$elemMatch)** Pour être plus précis, nous avons cherché les produits dont le tarif pour la taille "grande" spécifiquement est inférieur à 8.0€.

`db.produits.find({ tarifs: { $elemMatch: { taille: "grande", tarif: { $lt: 8.0 } } } })`

```
chopizza> db.produits.find({ tarifs: { $elemMatch: { taille: "grande", tarif: { $lt: 8.0 } } } })
[
  {
    _id: ObjectId('6543ac35a08f42f1cc02e811'),
    numero: 5,
    libelle: 'cola',
    description: 'Cola-Calo, producteur local',
    image: 'https://www.dominos.fr/ManagedAssets/FR/product/BOCO.png',
    categorie: 'Boissons',
    tarifs: [
      { taille: 'grande', tarif: 3.99 },
      { taille: 'normale', tarif: 2.99 }
    ],
    recettes: [
      ObjectId('6543b503f0e418df120c79b9'),
      ObjectId('6543b503f0e418df120c79a3'),
      ObjectId('6543b503f0e418df120c79be'),
      ObjectId('6543b503f0e418df120c79a5')
    ]
  },
  {
    _id: ObjectId('6543ac35a08f42f1cc02e812'),
    numero: 6,
    libelle: 'eau',
    description: 'eau plate, eau pétillante ou aromatisée',
    image: 'https://www.dominos.fr/ManagedAssets/FR/product/BOEA.png',
    categorie: 'Boissons',
    tarifs: [
      { taille: 'grande', tarif: 2.99 },
      { taille: 'normale', tarif: 1.99 }
    ],
  },
]
```

10. Insertion de données : Nous avons ajouté un nouveau produit ("Pizza Shell") via insertOne.

```
db.produits.insertOne({ numero: 99, libelle: "Pizza Shell", categorie: "Pizzas", tarifs: [{  
taille: "normale", tarif: 10.0 } ] })
```

```
chopizza> db.produits.insertOne({ numero: 99, libelle: "Pizza Shell", categorie: "Pizzas", tarifs: [{ taille: "normale", tarif: 10.0 } ] })  
{  
  acknowledged: true,  
  insertedId: ObjectId('695e707bdc76948f1b8de666')  
}
```

11. Jointure (Agrégation) : Pour afficher les recettes liées au produit n°1, nous avons utilisé un pipeline d'agrégation avec \$lookup pour faire la jointure entre la collection produits et recettes.

```
db.produits.aggregate([ { $match: { numero: 1 } }, { $lookup: { from: "recettes",  
localField: "recettes", foreignField: "_id", as: "recettes_detail" } } ])
```

```
chopizza> db.produits.aggregate([ { $match: { numero: 1 } }, { $lookup: { from: "recettes", localField: "recettes", foreignField: "_id", as: "r  
ecettes_detail" } } ])  
[  
  {  
    _id: ObjectId('6543ac35a08f42f1cc02e80d'),  
    numero: 1,  
    libelle: 'Margherita',  
    description: 'Tomate, mozzarella, basilic',  
    image: 'https://www.dominos.fr/ManagedAssets/FR/product/PZSO.png',  
    categorie: 'Pizzas',  
    tarifs: [  
      { taille: 'grande', tarif: 11.99 },  
      { taille: 'normale', tarif: 8.99 }  
    ],  
    recettes: [  
      ObjectId('6543b503f0e418df120c79b8'),  
      ObjectId('6543b503f0e418df120c79ba'),  
      ObjectId('6543b503f0e418df120c79a2')  
    ],  
    recettes_detail: [  
      {  
        _id: ObjectId('6543b503f0e418df120c79a2'),  
        nom: 'pate a pizza',  
        ingredients: [  
          'lenticchie',  
          'champignons',  
          'pepperoni',  
          'vongole',  
          'creme fraiche',  
          'huile olive'
```

```
      ],  
      preparation: 20,  
      cuisson: 30,  
      etapes: {  
        '1': 'Blanditiis quidem eaque saepe expedita officia voluptas. Qui enim quos ut temporibus minima qui. Eius et vero eum nostrum omnis  
aut sit nulla. Ut architecto consequatur praesentium voluptate nesciunt. Eaque quas eius tempora voluptatem. Non ab minus beatae qui quam sed.  
Facilis ipsa et et rem minus. Pariatur voluptates quod eligendi nihil sit ut doloribus. Molestiae in ut vel facilis aut sed. Sed sequi et ipsa  
laborum qui sit. Voluptate laborum molestiae quaerat laborum nulla.',  
        '2': 'Et facere quasi voluptas maiores. Odit in quia aut quia. Consectetur error sit et ab qui amet. Dolores dolores et error facere  
aut. Iste vero eligendi tempore optio rerum et. Dolorem sed repellendus blanditiis tempora distinctio. Nulla debitis dolorum explicabo placeat  
nostrum qui odio. Dolorem rerum accusantium perspiciatis animi modi rerum eum sint.',  
        '3': 'Dolorem sunt aut iure qui corporis dolore. Dolorum doloribus et cum dolorem. Labore est nobis quibusdam similique aperiam ut ex  
pedita quibusdam. Temporibus et architecto voluptatem voluptatem odit esse nobis. Molestias autem beatae exercitationem nostrum delectus. Earum  
dolorem officiis illo dolorem. Dolore reiciendis aut architecto dolorum excepturi officiis.',  
        '4': 'Qui facere repellat ducimus voluptatem temporibus. Necessitatibus ab nihil consequatur voluptas. Aut et ullam aut sit. Animi a  
sit repellat. Fugiat sequi architecto quia error hic consequatur. Consequatur repudiandae est qui esse accusantium nam molestiae. Assumenda qui  
squam dolore officia eos voluptate explicabo occaecati rerum. Tempora quasi sunt eveniet accusamus. Non et nobis ea unde rem deleniti cupiditat  
e itaque.'  
      },  
      note: 5,  
      difficile: 'difficile'  
    },  
    {  
      _id: ObjectId('6543b503f0e418df120c79b8'),  
      nom: 'zuppa di pomodoro',  
      ingredients: [ 'poulet', 'lardons', 'parmesan', 'poireaux', 'ceci', 'riz' ],  
      preparation: 45,  
      cuisson: 45,  
      etapes: {
```

II. Requêtes en PHP

Dans cette seconde partie, nous avons utilisé la librairie PHP de MongoDB pour automatiser les requêtes et formater l'affichage HTML.

1. Liste simple Récupération et boucle sur les produits pour afficher Numéro, Catégorie et Libellé.

```
echo "<h3>1. Liste simple (Numéro, Catégorie, Libellé)</h3>";
$results = $collection->find(filter: [], options: [
    'projection' => [
        'numero' => 1,
        'categorie' => 1,
        'libelle' => 1,
        '_id' => 0
    ]
]);
foreach ($results as $p) {
    echo "N°{$p['numero']} - [{$p['categorie']}] {$p['libelle']}<br>";
}
```

2. Détail d'un produit Recherche d'un document spécifique (findOne) et affichage formaté de ses détails et de ses tarifs.

```
$p6 = $collection->findOne(
    filter: ['numero' => 6],
    options: [
        'projection' => [
            'libelle' => 1,
            'categorie' => 1,
            'description' => 1,
            'tarifs' => 1,
            '_id' => 0
        ]
    ]
);

if ($p6) {
    echo "<strong>{$p6['libelle']}</strong> ({$p6['categorie']})<br>";
    echo "<em>{$p6['description']}</em><br>";
    echo "Tarifs : ";
    foreach ($p6['tarifs'] as $t) {
        echo "{$t['taille']}={$t['tarif']}€ ";
    }
} else {
    echo "Produit 6 introuvable.";
}
```

3. Filtrage avancé Requête PHP pour trouver les produits dont le tarif "taille normale" est ≤ 3.0€.

```
$results = $collection->find(filter: [
    'tarifs' => [
        '$elemMatch' => [
            'taille' => 'normale',
            'tarif' => ['$lte' => 3.0]
        ]
    ]
]);
foreach ($results as $p) {
    echo "- {$p['libelle']}<br>";
}
```

4. Filtre sur la taille d'un tableau Utilisation de l'opérateur \$size pour trouver les produits associés exactement à 4 recettes.

```
$results = $collection->find(filter: [
    'recettes' => ['$size' => 4]
]);
foreach ($results as $p) {
    echo "- {$p['libelle']} (N°{$p['numero']})<br>";
}
```

5. Agrégation en PHP Reproduction de la jointure (Lookup) vue précédemment, mais exécutée depuis le script PHP pour afficher les recettes d'un produit.

```
$results = $collection->aggregate(pipeline: [
    ['$match' => ['numero' => 6]],
    ['$lookup' => [
        'from' => 'recettes',
        'localField' => 'recettes',
        'foreignField' => '_id',
        'as' => 'details_recettes'
    ]
]);
```

6. Fonction API JSON : Création d'une fonction `getProduitInfo` qui prend un numéro et une taille, et retourne les données au format JSON.

```
if (!function_exists('getProduitInfo')) {  
    2 references  
    function getProduitInfo($collection, $numero, $tailleDemande): array {  
        $produit = $collection->findOne(['numero' => $numero]);  
        if (!$produit) return ["erreur" => "Produit introuvable"];  
  
        $tarifTrouve = null;  
        foreach ($produit['tarifs'] as $t) {  
            if ($t['taille'] === $tailleDemande) {  
                $tarifTrouve = $t['tarif'];  
                break;  
            }  
        }  
  
        if ($tarifTrouve === null) return ["erreur" => "Taille inexistante"];  
  
        return [  
            'numero' => $produit['numero'],  
            'libelle' => $produit['libelle'],  
            'categorie' => $produit['categorie'],  
            'taille' => $tailleDemande,  
            'tarif' => $tarifTrouve  
        ];  
    }  
}  
  
$info = getProduitInfo(collection: $collection, numero: 2, tailleDemande: 'grande');
```

Le résultat sur navigateur :

Partie 2 : Requêtes en PHP

1. Liste simple (Numéro, Catégorie, Libellé)

N°1 - [Pizzas] Margherita
N°5 - [Boissons] cola
N°6 - [Boissons] eau
N°4 - [Pizzas] Pepperoni
N°2 - [Pizzas] Reine
N°3 - [Pizzas] Savoyarde
N°8 - [Salades] salade tomate
N°9 - [Desserts] tiramisu
N°10 - [Desserts] panna cotta
N°7 - [Salades] salade verte
N°999 - [Pizzas] Pizza Speciale Dev
N°99 - [Pizzas] Pizza Shell
N°105 - [Pizzas] Pizza test
N°105 - [Pizzas] Pizza test

2. Détail du produit N°6

eau (Boissons)

eau plate, eau pétillante ou aromatisée

Tarifs : grande=2.99€ normale=1.99€

3. Produits pas chers (taille normale <= 3.0€)

- cola
- eau
- Pizza test
- Pizza test

4. Produits ayant exactement 4 recettes

- cola (N°5)
- Pepperoni (N°4)
- salade tomate (N°8)
- tiramisu (N°9)

5. Recettes du produit N°6

Produit : eau

Recettes associées :

- pate a pizza (Diff: difficile)
- tiramisu (Diff: moyen)
- pizza napolitaine (Diff: expert)
- spaghetti alle vongole (Diff: expert)
- minestra di farro (Diff: facile)

6. Fonction PHP et JSON

```
{  
  "numero": 2,  
  "libelle": "Reine",  
  "categorie": "Pizzas",  
  "taille": "grande",  
  "tarif": 12.99  
}
```

III. Application Web "Chopizza"

Nous avons intégré ces appels dans une mini-interface web.

1. Navigation par catégorie : L'interface permet de filtrer les produits via des boutons (Boissons, Desserts, Pizzas, etc.). Le tableau affiche dynamiquement les libellés, descriptions et la liste des tarifs.

Chopizza

1. Choisir une catégorie :

Tout voir Boissons Desserts Pizzas Salades

2. Liste des produits :

N°	Libellé	Description	Tarifs
1	Margherita Pizzas	Tomate, mozzarella, basilic	grande : 11.99 € normale : 8.99 €
2	Reine Pizzas	Tomate, mozzarella, jambon, champignons	normale : 9.99 € grande : 12.99 €
3	Savoyarde Pizzas	Tomate, mozzarella, jambon, reblochon, pommes de terre, oignons	grande : 13.99 € normale : 10.99 €
4	Pepperoni Pizzas	Tomate, mozzarella, pepperoni	grande : 12.99 € normale : 9.99 €
99	Pizza Shell Pizzas	power pizza	normale : 10 €
105	Pizza test Pizzas	pitie que ça marche	normale : 2 €

2. Formulaire d'ajout : Un formulaire permet de créer un nouveau produit en base. Les champs Catégorie et Taille sont proposés via des listes déroulantes pour éviter les erreurs de saisie.

3. Ajouter un produit

Numéro
106

Libellé
Pizza 4 fromage

Catégorie
Pizzas

Description
Tomate, mozzarella, chèvre, langre, bleu

Taille
Géant

Prix
12

Enregistrer le produit

3. Validation : Après soumission, le produit est inséré en base et un message de confirmation s'affiche.

Produit ajouté avec succès !

106	Pizza 4 fromage Pizzas	Tomate, mozzarella, chèvre, langre, bleu	geant : 12 €
-----	----------------------------------	--	--------------