# NoSQL - DIA2

**MANNAI Hasna (≈ 33%)**
**CHENIK Yassine (≈ 33%)**
**BOUCHIBA Emine (≈ 33%)**

# RAPPORT
# TP MongoDB

# TABLE DES MATIERES

# Importation du json dans MongoDB

## Dézipper le fichier JSON :

**Utilisation de WinRar pour dézipper le fichier et le mettre dans notre répertoire de travail.**

## Lancement de Docker :

**Une fois Docker lancé et le container Cassandra lancé :**

➔ **On lance le terminal Windows dans lequel on tape la ligne de commande suivante afin d'importer le fichier JSON dans Docker :**

- ✓ **Docker cp "C:\Users\emine\Documents\Ecole\A4\S8\Advanced_topics_in_NoSql_databases\TP_MongoDB\companies2.json" MongoDB:/**

➔ **On importe le fichier json dans le container MongoDB :**

- ✓ **mongoimport --db DB_Companies --collection TD --file companies2.json**
- ✓ *Ici, on utilise **"--file"** et non **"--jsonArray"** parce que le ficher JSON companies2.json ne comporte pas une liste dans laquelle est contenu nos objets JSON comme dans celui du TD. Ici, les objets sont inscrits seuls sans être dans une liste.*

➔ **On entre dans le terminal Cassandra :**

- ✓ **mongosh (dans le terminal Docker)**

➔ **On se met dans la database :**

- ✓ **USE DB_Companies;**

# Simple Queries

## 1. Get all the companies that have a category_code=”nanotech” :

➢ db.TD.find({"category_code":"nanotech"},{"category_code":1,"name":1})

✓ *Ici, on veut afficher les entreprises qui sont dans le domaine de la nanotechnologie en affichant le **"name"**, le **"category_code"** ainsi que l’**id**.*

✓ *Output :*

```
DB_Companies> db.TD.find({"category_code":"nanotech"},{"category_code":1,"name":1})
[
  {
    _id: ObjectId("63fb0623e48211a53b2643c9"),
    name: 'Nanosolar',
    category_code: 'nanotech'
  },
  {
    _id: ObjectId("63fb0623e48211a53b26695c"),
    name: 'Nantero',
    category_code: 'nanotech'
  },
  {
    _id: ObjectId("63fb0624e48211a53b2680f8"),
    name: 'QD Vision',
    category_code: 'nanotech'
  },
  {
    _id: ObjectId("63fb0624e48211a53b2682c8"),
    name: 'Pixelligent',
    category_code: 'nanotech'
  },
  {
    _id: ObjectId("63fb0624e48211a53b26874f"),
    name: 'Siluria Technologies',
    category_code: 'nanotech'
  }
]
```

# Simple Queries

## 2. Get all the Companies that founded in 2008 :

- db.TD.find({"founded_year":2008},{"founded_year":1,"founded_month":1,"founded_day":1,"name":1})

✓ *Ici, on veut afficher les entreprises qui ont été fondés en 2008. en affichant le **"name"**, le **"founded_year"**, le **"founded_month"**, le **"founded_day"** ainsi que l'**id**.*

✓ *Output :*

```
DB_Companies> db.TD.find({"founded_year":2008},{"founded_year":1,"founded_month":1,"founded_day":1,"name":1})
[
  {
    _id: ObjectId("63fb0623e48211a53b26438a"),
    name: 'BeliefNet',
    founded_year: 2008,
    founded_month: null,
    founded_day: null
  },
  {
    _id: ObjectId("63fb0623e48211a53b264402"),
    name: 'Fancast',
    founded_year: 2008,
    founded_month: 1,
    founded_day: 1
  },
  {
    _id: ObjectId("63fb0623e48211a53b26440e"),
    name: 'Webnode',
    founded_year: 2008,
    founded_month: 1,
    founded_day: 1
  },
  {
    _id: ObjectId("63fb0623e48211a53b264411"),
    name: 'Newspepper',
    founded_year: 2008,
    founded_month: null,
    founded_day: null
  },
  {
    _id: ObjectId("63fb0623e48211a53b264441"),
    name: 'MOLI',
    founded_year: 2008,
    founded_month: 1,
    founded_day: null
  },
```

# Simple Queries

## 3. Get the number of Companies whith more than 100 employees :

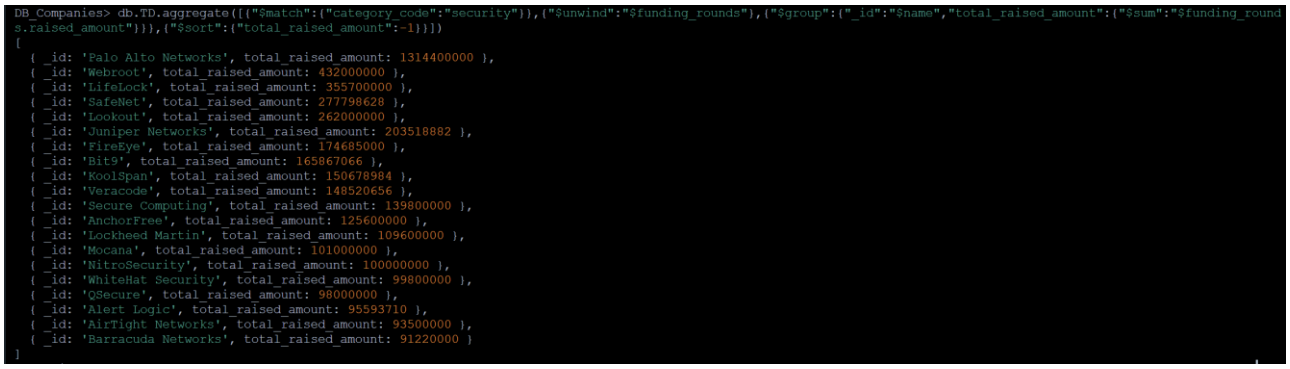
- db.TD.find({"number_of_employees":{"$gte":100}},{"name":1,"number_of_employees":1})
- ✓ *Ici, on veut afficher les entreprises qui ont plus de 100 employés en affichant le **"name"**, le **"number_of_employees"** ainsi que l'**id**.*
- ✓ *Output :*

```
DB_Companies> db.TD.find({"number_of_employees":{"$gte":100}},{"name":1,"number_of_employees":1})
[
  {
    _id: ObjectId("63fb0623e48211a53b264386"),
    name: 'GrubHub',
    number_of_employees: 250
  },
  {
    _id: ObjectId("63fb0623e48211a53b26438f"),
    name: 'Skyscanner',
    number_of_employees: 300
  },
  {
    _id: ObjectId("63fb0623e48211a53b264392"),
    name: 'Viadeo',
    number_of_employees: 400
  },
  {
    _id: ObjectId("63fb0623e48211a53b2643aa"),
    name: 'Match',
    number_of_employees: 350
  },
  {
    _id: ObjectId("63fb0623e48211a53b2643b4"),
    name: 'CDNetworks',
    number_of_employees: 350
  },
  {
    _id: ObjectId("63fb0623e48211a53b2643b7"),
    name: 'KAYAK',
    number_of_employees: 101
  },
```

# Simple Queries

## 4. Get all companies founded in May 2019 :

- db.TD.aggregate([{"$match":{"category_code":"security"}},{"$unwind":"$funding_rounds"},{"$group":{"_id":"$name","total_raised_amount":{"$sum":"$funding_rounds.raised_amount"}}},{"$sort":{"total_raised_amount":-1}}])
- ✓ *Ici nous cherchons à afficher le **montant total amassé** par les entreprises qui travaillent dans le **domaine de la sécurité**.*
- ✓ *Output :*



```
DB_Companies> db.TD.aggregate([{"$match":{"category_code":"security"}},{"$unwind":"$funding_rounds"},{"$group":{"_id":"$name","total_raised_amount":{"$sum":"$funding_rounds.raised_amount"}}},{"$sort":{"total_raised_amount":-1}}])
[
  { _id: 'Palo Alto Networks', total_raised_amount: 1314400000 },
  { _id: 'Webroot', total_raised_amount: 432000000 },
  { _id: 'LifeLock', total_raised_amount: 355700000 },
  { _id: 'SafeNet', total_raised_amount: 277798628 },
  { _id: 'Lookout', total_raised_amount: 262000000 },
  { _id: 'Juniper Networks', total_raised_amount: 203518882 },
  { _id: 'FireEye', total_raised_amount: 174685000 },
  { _id: 'Bit9', total_raised_amount: 165867066 },
  { _id: 'KoolSpan', total_raised_amount: 150678984 },
  { _id: 'Veracode', total_raised_amount: 148520656 },
  { _id: 'Secure Computing', total_raised_amount: 139800000 },
  { _id: 'AnchorFree', total_raised_amount: 125600000 },
  { _id: 'Lockheed Martin', total_raised_amount: 109600000 },
  { _id: 'Mocana', total_raised_amount: 101000000 },
  { _id: 'NitroSecurity', total_raised_amount: 100000000 },
  { _id: 'WhiteHat Security', total_raised_amount: 99800000 },
  { _id: 'QSecure', total_raised_amount: 98000000 },
  { _id: 'Alert Logic', total_raised_amount: 95593710 },
  { _id: 'AirTight Networks', total_raised_amount: 93500000 },
  { _id: 'Barracuda Networks', total_raised_amount: 91220000 }
]
```

## 5. Get total money raised for each domain :

- db.TD.aggregate([{"$match":{"category_code":{"$ne":null}}},{"$unwind":"$funding_rounds"},{"$group":{"_id":{"category_code":"$category_code","name":"$name"},"total_raised_amount":{"$sum":"$funding_rounds.raised_amount"}}},{"$group":{"_id":"$_id.category_code","total_raised_amount":{"$sum":{"$convert":{"input":"$total_raised_amount","to":"double"}}}}},{"$sort":{"total_raised_amount":-1}}])

- ✓ *Ici, nous cherchons à afficher le **montant total amassé** par les entreprises du **même domaine**.*
- ✓ *Output :*

```
DB_Companies> db.TD.aggregate([{"$match":{"category_code":{"$ne":null}}},{"$unwind":"$funding_rounds"},{"$group":{"_id":{"category_code":"$category_code","name":"$name"},"
total_raised_amount":{"$sum":"$funding_rounds.raised_amount"}}},{"$group":{"_id":"$_id.category_code","total_raised_amount":{"$sum":{"$convert":{"input":"$total_raised_amo
unt","to":"double"}}}}},{"$sort":{"total_raised_amount":-1}}])
[
  { _id: 'software', total_raised_amount: 33944457514 },
  { _id: 'cleantech', total_raised_amount: 31449030414 },
  { _id: 'mobile', total_raised_amount: 31296583972 },
  { _id: 'games_video', total_raised_amount: 27865408810 },
  { _id: 'biotech', total_raised_amount: 23580622906 },
  { _id: 'web', total_raised_amount: 21715309802 },
  { _id: 'enterprise', total_raised_amount: 16105137306 },
  { _id: 'hardware', total_raised_amount: 14601832496 },
  { _id: 'advertising', total_raised_amount: 12254157930.2 },
  { _id: 'network_hosting', total_raised_amount: 10808769886 },
  { _id: 'social', total_raised_amount: 9423242476 },
  { _id: 'ecommerce', total_raised_amount: 8925804782 },
  { _id: 'semiconductor', total_raised_amount: 8760223044 },
  { _id: 'security', total_raised_amount: 6428595218 },
  { _id: 'manufacturing', total_raised_amount: 5156004322 },
  { _id: 'public_relations', total_raised_amount: 5080730386 },
  { _id: 'search', total_raised_amount: 4742096616 },
  { _id: 'medical', total_raised_amount: 3123714274 },
  { _id: 'analytics', total_raised_amount: 3072373734 },
  { _id: 'finance', total_raised_amount: 2518553030 }
]
```

## 6. Get all companies that starts with an “F” :

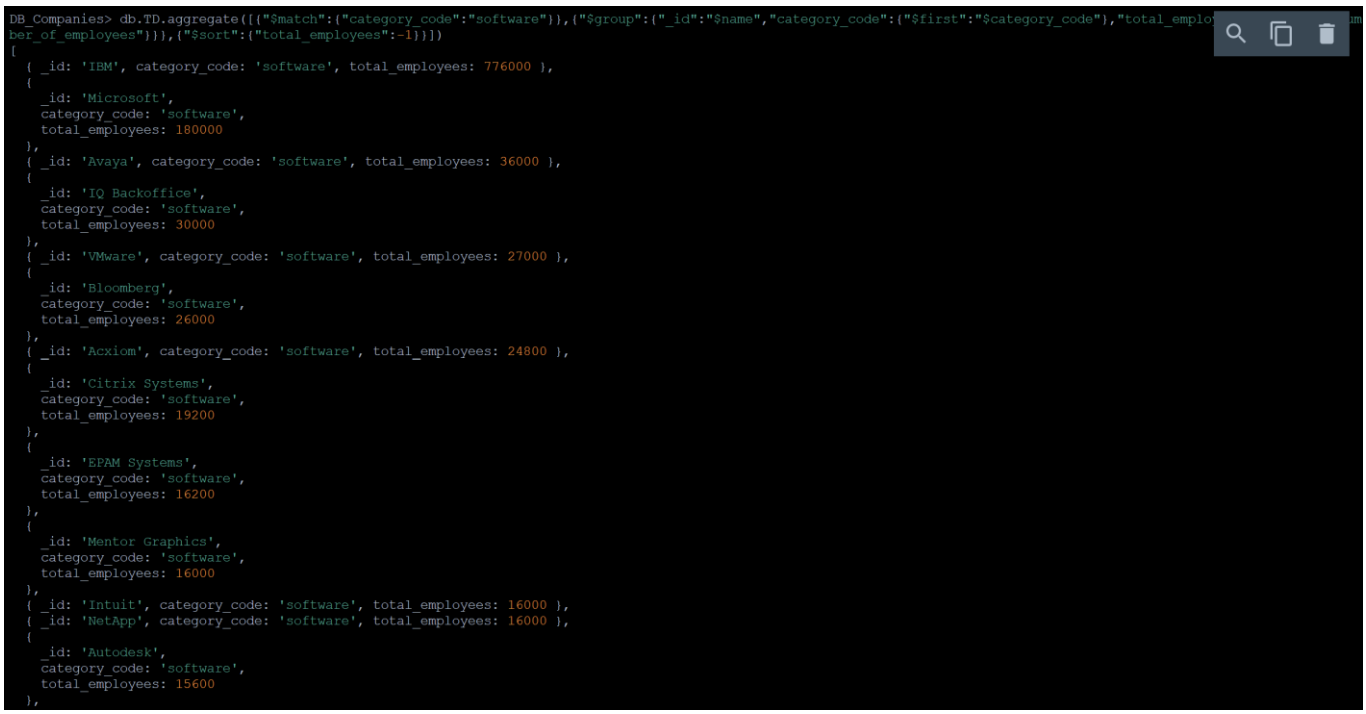
- db.TD.find({"name":{"$regex":"^F"}},{"name":1})
- *Ici, on cherche à afficher toutes les entreprises qui commencent par la lettre F. Pour ce faire, on utilise l’option **$regex** qui nous permet de reconnaître un motif dans une chaîne de caractères et on lui donne comme argument **‘^F’**. De et on affiche les noms des entreprises correspondantes.*
- *Output :*

```
DB_Companies> db.TD.find({"name":{"$regex":"^F"}},{"name":1})
[
  { _id: ObjectId("63fb0623e48211a53b264375"), name: 'Fraud Sciences' },
  { _id: ObjectId("63fb0623e48211a53b26437b"), name: 'FastBooking' },
  { _id: ObjectId("63fb0623e48211a53b264394"), name: 'FFWD Wheels' },
  { _id: ObjectId("63fb0623e48211a53b2643a7"), name: 'Flixwagon' },
  { _id: ObjectId("63fb0623e48211a53b2643c8"), name: 'Fotolia' },
  { _id: ObjectId("63fb0623e48211a53b2643d5"), name: 'Fyreball' },
  { _id: ObjectId("63fb0623e48211a53b264402"), name: 'Fancast' },
  { _id: ObjectId("63fb0623e48211a53b264408"), name: 'FamCorner' },
  { _id: ObjectId("63fb0623e48211a53b26440c"), name: 'FriendsAbroad' },
  { _id: ObjectId("63fb0623e48211a53b264427"), name: 'Frevvo' },
  { _id: ObjectId("63fb0623e48211a53b264429"), name: 'Freepath' },
  { _id: ObjectId("63fb0623e48211a53b26445d"), name: 'FotoFlexer' },
  { _id: ObjectId("63fb0623e48211a53b26445f"), name: 'First30Days' },
  { _id: ObjectId("63fb0623e48211a53b26446d"), name: 'FutonMedia' },
  { _id: ObjectId("63fb0623e48211a53b264494"), name: 'FiveLimes' },
  { _id: ObjectId("63fb0623e48211a53b2644a0"), name: 'FUPEI' },
  { _id: ObjectId("63fb0623e48211a53b2644e3"), name: 'Feedmap' },
  { _id: ObjectId("63fb0623e48211a53b264524"), name: 'Flypaper' },
  { _id: ObjectId("63fb0623e48211a53b264529"), name: 'Fat Goose' },
  { _id: ObjectId("63fb0623e48211a53b264540"), name: 'Fluc' }
]
```

# Complex Queries

## 1. Get the number of employees of each Company that have as category_code =”software” :

- db.TD.aggregate([{"$match":{"category_code":"software"}},{"$group":{"_id":"$name","category_code":{"$first":"$category_code"},"total_employees":{"$sum":"$number_of_employees"}}},{"$sort":{"total_employees":-1}}])
- *Ici, on cherche à avoir le **nombre d'employés** travaillant au sein d’entreprises dans le domaine du **Software** et on affiche **le nom de l’entreprise**, sa **category_code** et le **nombre d’employés**.*
- *Output :*



# Complex Queries

## 2. Get the number of products grouped by companies name :

- db.TD.aggregate([{"$group":{"_id":"$name","total_products":{"$sum":{"$size":"$products"}}}},{"$sort":{"total_products":-1}}])
- *Ici, on veut afficher le nombre de produits provenant de chacune des entreprises.*
- *Output :*

```
DB_Companies> db.TD.aggregate([{"$group":{"_id":"$name","total_products":{"$sum":{"$size":"$products"}}}},{"$sort":{"total_products":-1}}])
[
  { _id: 'The Teaching Company', total_products: 304 },
  { _id: 'Google', total_products: 182 },
  { _id: 'Chillingo', total_products: 136 },
  { _id: 'Microsoft', total_products: 102 },
  { _id: 'Yahoo!', total_products: 96 },
  { _id: 'i-Jet Media', total_products: 94 },
  { _id: 'Amazon', total_products: 82 },
  { _id: 'PlayFirst', total_products: 68 },
  { _id: 'Logic Voice', total_products: 64 },
  { _id: 'Yandex', total_products: 62 },
  { _id: 'Apple', total_products: 60 },
  { _id: 'modu', total_products: 52 },
  { _id: 'Webroot', total_products: 52 },
  { _id: 'AMP Consulting', total_products: 48 },
  { _id: 'Tintash', total_products: 48 },
  { _id: 'Dream Cheeky', total_products: 40 },
  { _id: 'SuccessFactors', total_products: 38 },
  { _id: 'Telenav', total_products: 38 },
  { _id: 'Mentor Graphics', total_products: 36 },
  { _id: 'Concentric Sky', total_products: 36 }
]
```

# Hard Queries

## 1. Get the total amount of money raised by companies founded in each year, sorted in descending order of the total amount raised :

- db.TD.aggregate([{"$match":{"founded_year":{"$ne":null}}},{"$unwind":"$funding_rounds"},{"$group":{"_id":{"year":"$founded_year","name":"$name"},"total_raised_amount":{"$sum":"$funding_rounds.raised_amount"}}},{"$group":{"_id":"$_id.year","total_raised_amount":{"$sum":{"$toDouble":"$total_raised_amount"}}}},{"$sort":{"total_raised_amount":-1}}])

✓ *Ici, on cherche à obtenir le montant total d'argent collecté par les entreprises fondées chaque année, trié par ordre décroissant du montant total collecté.*

✓ *Output :*