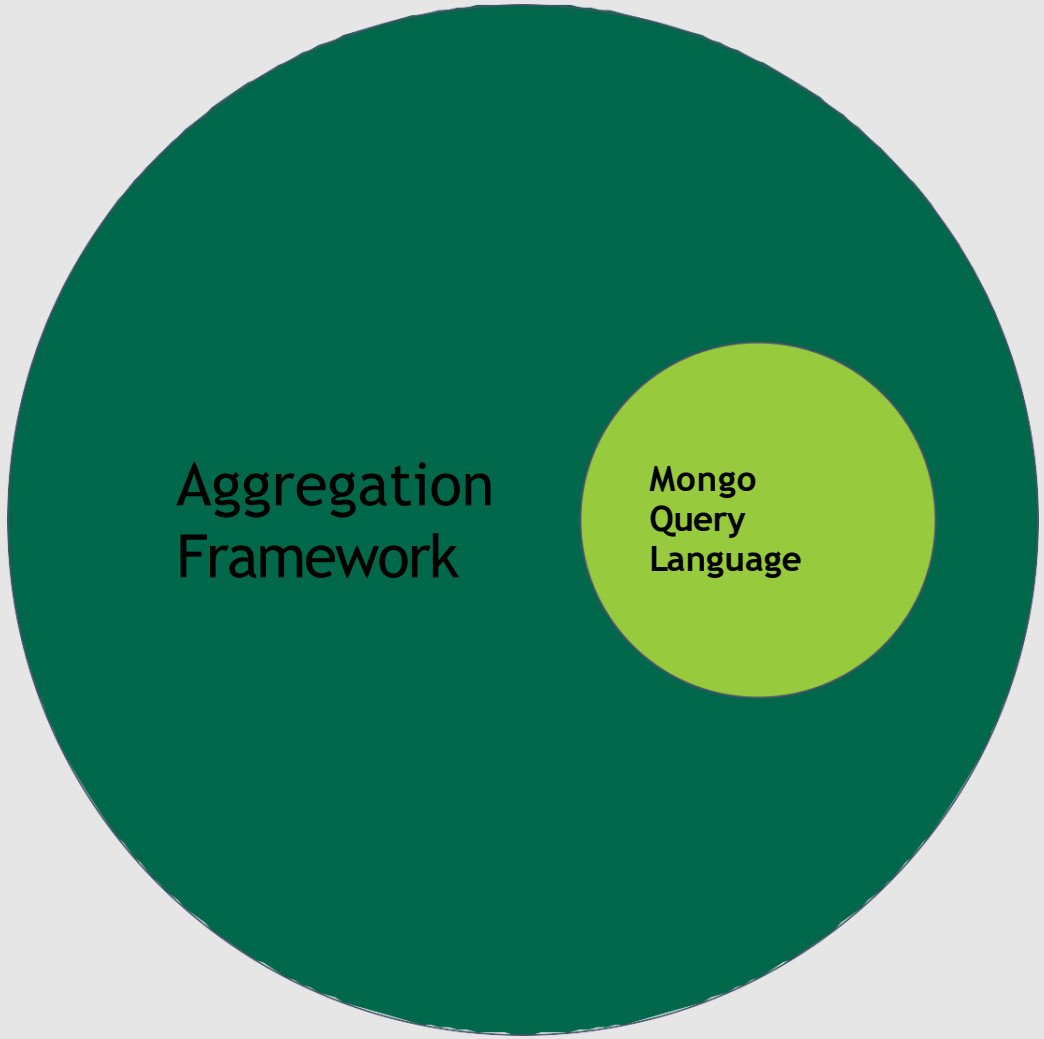
Introducción a MongoDB

Servicio Nacional de Aprendizaje SENA

Centreo de Servicios de Gestión Empresarial (CESGE)

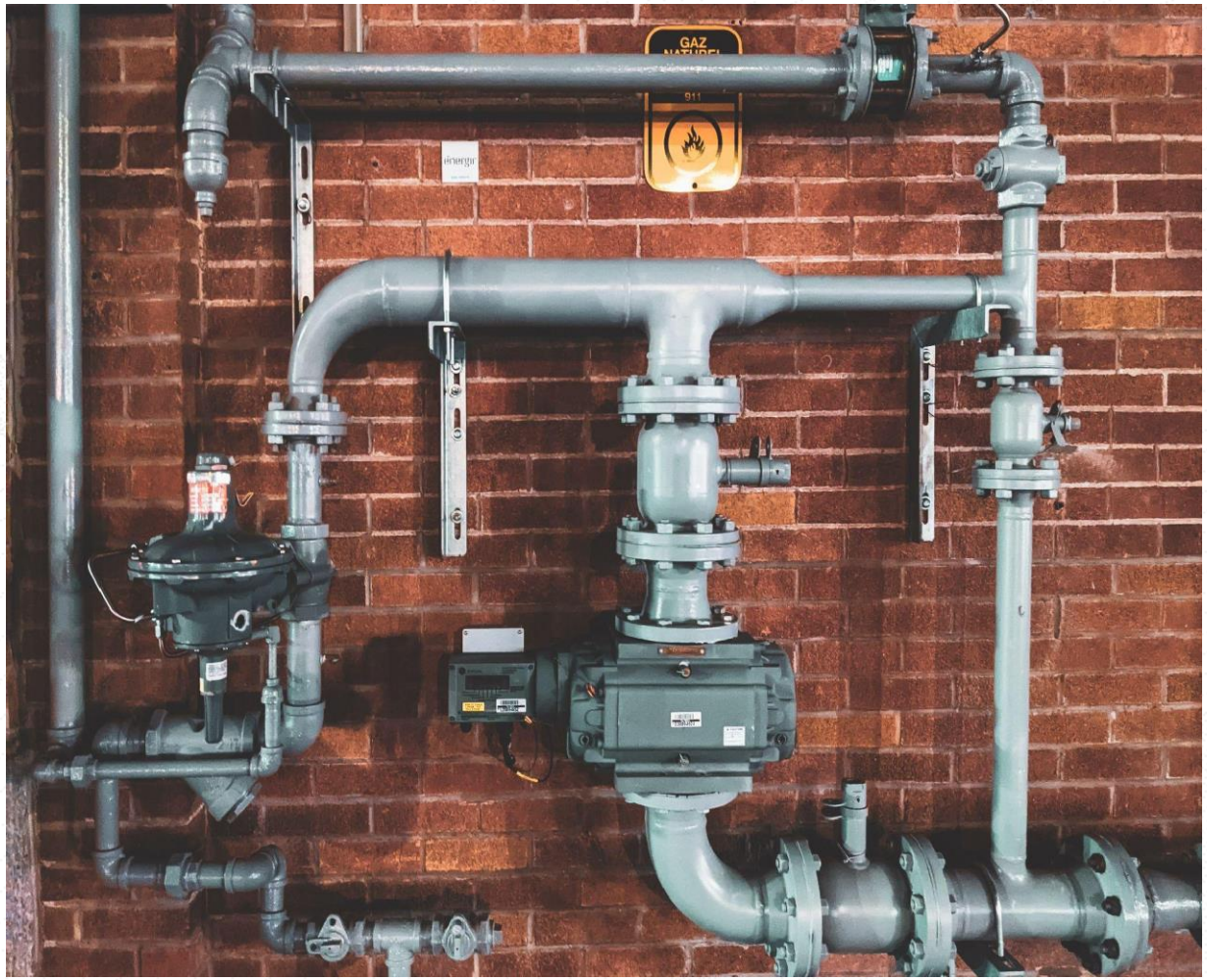
2023-I

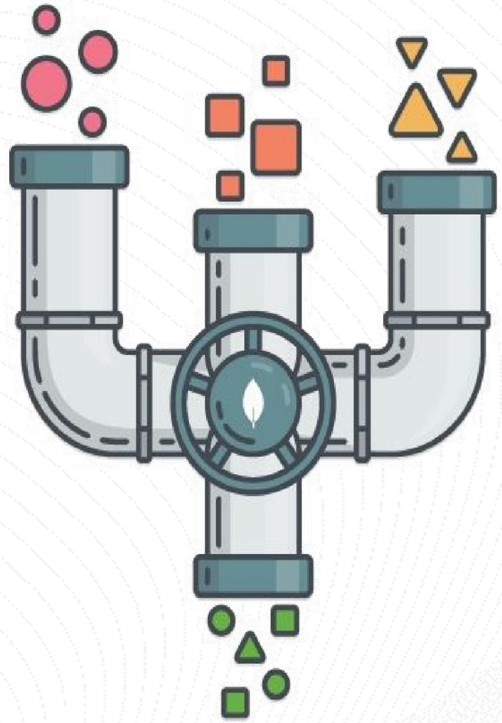
Aggregation Pipelines



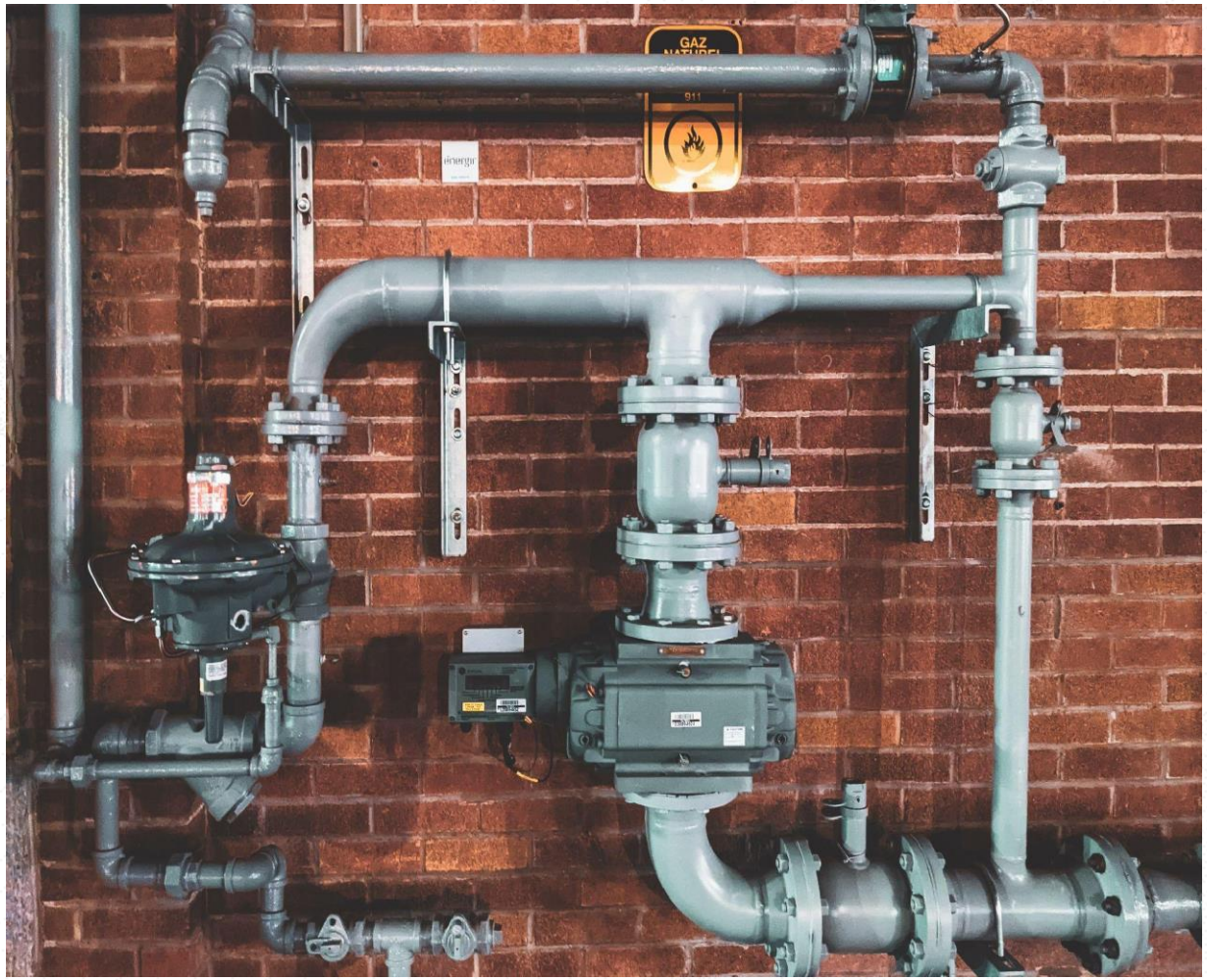
MongoDB.

Pipelines: Tuberías de datos





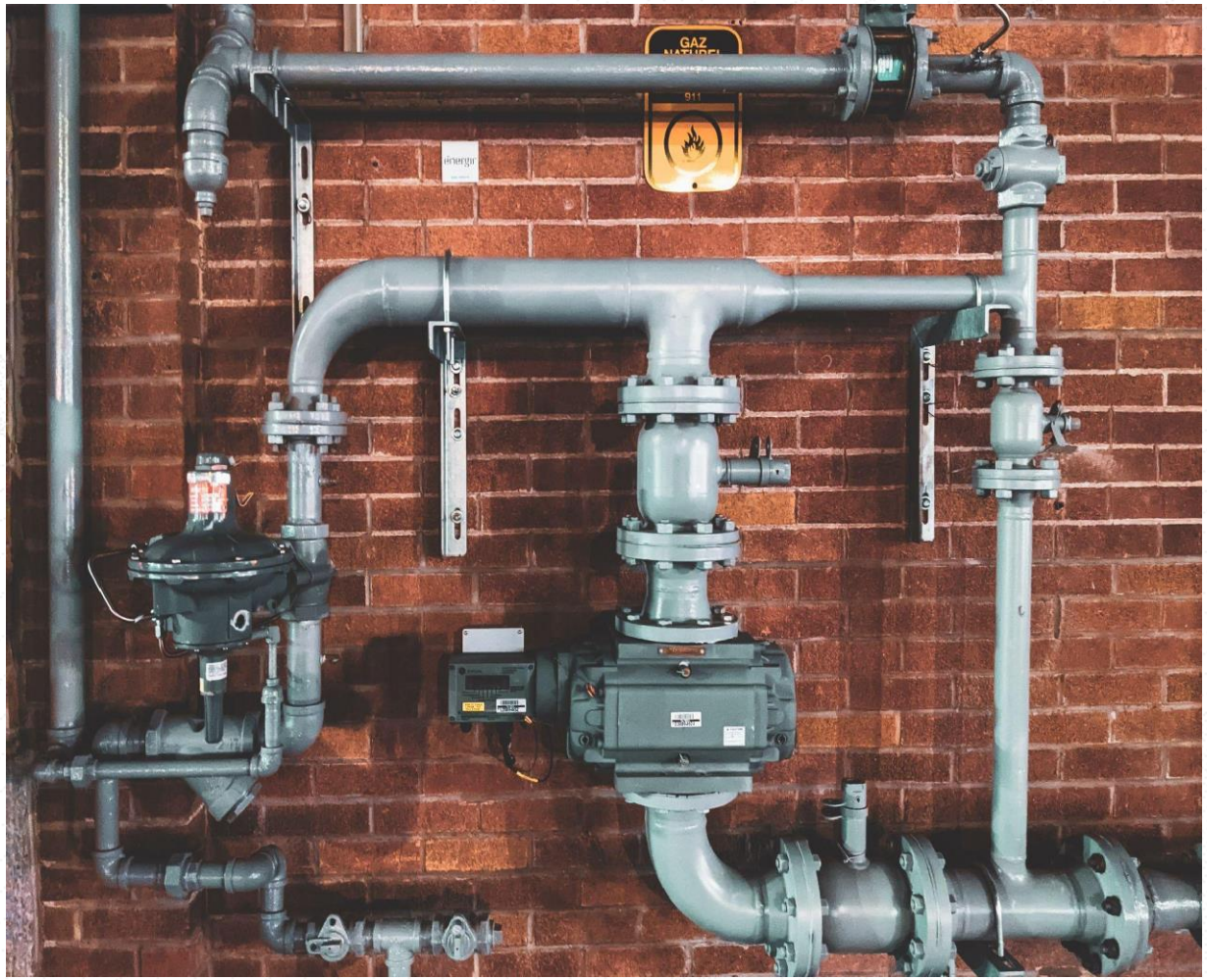
Pipelines: Tuberías de datos



Las operaciones de agregación nos permiten agrupar, clasificar, realizar cálculos, analizar datos y mucho más.

Las “tuberías” de agregación pueden tener una o más “etapas” (stages).
El orden de estas etapas es importante: Cada etapa actúa sobre los resultados de la etapa anterior.

Pipelines: Tuberías de datos

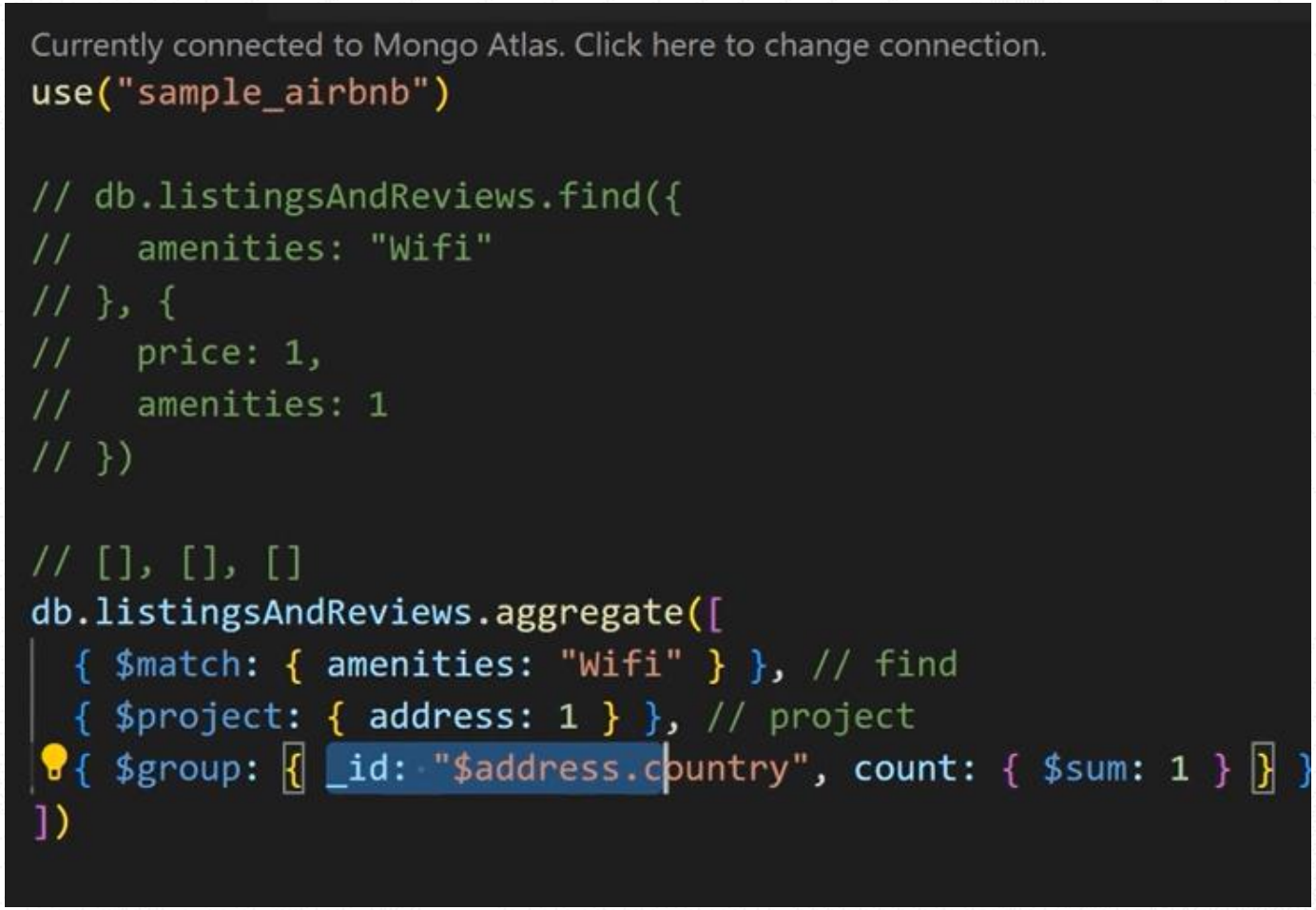


Documentación

https://www.mongodb.com/docs/manual/aggregation/

Ejemplo: find() = aggregate()

Ejemplo: find() = aggregate()



Aggregation $match

$match

- Esta etapa de agregación se comporta como un find(). Filtrará los documentos que coincidan con la consulta proporcionada.
- El uso temprano $match en el pipeline puede mejorar el rendimiento, ya que limita la cantidad de documentos que deben procesar las siguientes etapas.

```
db.listingsAndReviews.aggregate([
    {$match:{amenities:"Wifi"}},//find
    {$project:{price: true, amenities:true }}, //$projection
])
```

Aggregation $project

$project

- Pasa solo los campos especificados a la siguiente etapa de agregación.

```
//Concatenar
db.restaurants.aggregate(
  [
    {
      $match:{borough:"Brooklyn"}
    },
    {
      $match:{cuisine:"Mexican"}
    },
    {
      $project:{
          _id:0,name:1,grades:1,borough:1
        }
    }
  ]
)
```

Aggregation $group

$group

- Esta etapa de agregación agrupa los documentos según el _id proporcionado.
- Ojo! No confunda este _id con el _id de ObjectId proporcionado a cada documento.

```
//2. Cuenta el total de las propiedades de tipo casa
db.listingsAndReviews.aggregate([
  {
    $match: { property_type : "House" }
  },
  {
    $count: "totalHouse"
  }
])
```

Aggregation $limit

$limit

- Esta etapa de agregación limita el número de documentos que pasan a la siguiente etapa.
- Es distinto del $project porque limita la cantidad de documentos que se muestran, pero no la cantidad de campos.

```
// 9.Muestra una pelicula
use("sample_mflix")
db.movies.aggregate([ { $limit: 1 } ])
```

Aggregation $sort

$sort

- Esta etapa de agregación agrupa todos los documentos en el orden de clasificación especificado.
- Ojo! El orden de tus etapas importa: Cada etapa sólo actúa sobre los documentos que le proporcionan las etapas anteriores.

```
use("sample_restaurants")
db.restaurants.aggregate([
    {
        $match:{borough:"Brooklyn"}
    },
    {
        $match:{cuisine:"Mexican"}
    },
    {
        $project:{_id:0,name:1,"address.zipcode":1}
    },
    {
        $sort:{name:1}
    }
])
```

Aggregation $count

$count

- Esta etapa de agregación cuenta la cantidad total de documentos pasados de la etapa anterior.

```
//2. Cuenta el total de las propiedades de tipo casa
db.listingsAndReviews.aggregate([
  {
    $match: { property_type : "House" }
  },
  {
    $count: "totalHouse"
  }
])
```

Aggregation $lookup

Ejemplo $lookup

- https://www.mongodb.com/docs/manual/reference/operator/aggregation/lookup/

```
//10. Une los comentarios con las peliculas
// muestra solo un comentario
use("sample_mflix")
db.comments.aggregate([
  {
    $lookup: {
      from: "movies",
      localField: "movie_id",
      foreignField: "_id",
      as: "movie_details",
    },
  },
  {
    $limit: 1
  }
])
```

$lookup

- Esta etapa de agregación realiza una unión izquierda (left join) a una colección en la misma base de datos.
- Hay cuatro campos obligatorios:
  1. **from:** La colección que se usará para la búsqueda en la misma base de datos
  2. **localField:** el campo de la colección principal que se puede utilizar como identificador único en la colección from.
  3. **foreignField:** el campo de la colección from que se puede utilizar como identificador único en la colección principal.
  4. **as:** El nombre del nuevo campo que contendrá los documentos coincidentes de la colección from.

Aggregation $addfields

$addfields

- Esta etapa de agregación agrega nuevos campos a los documentos.

```
use("sample_mflix")
db.comments.aggregate([
  {
    $lookup: {
      from: "movies",
      localField: "movie_id",
      foreignField: "_id",
      as: "movie_details",
    },
  },
  {
    $limit: 1
  }
])
```

Aggregation $unwind

Ejemplo $unwind

- https://www.mongodb.com/docs/manual/reference/operator/aggregation/unwind/

```
db.restaurants.aggregate( [
    {
        $unwind: { path: "$grades" }
    },
    {
        $limit: 4
    }
])
```

Aggregation $map

Ejemplo $map

- https://www.mongodb.com/docs/manual/reference/operator/aggregation/map/

Definition

`$map`

Applies an expression to each item in an array and returns an array with the applied results.

The `$map` expression has the following syntax:

```
{ $map: { input: <expression>, as: <string>, in: <expression> } }
```

Ejemplo $map

- https://www.mongodb.com/docs/manual/reference/operator/aggregation/map/

```
//$map

db.restaurants.aggregate([
  {
    $match:{borough:"Brooklyn"}
  }
  ,
   {
      $project: {
          adjustedGrades: {
            $map: {
              input: "$grades.score",
              as: "grade",
              in: { $divide: [ "$$grade", 5 ] }
            }
          }
      }
   }
])
```

Aggregation $out

Ejercicio 1

- Usando la colección listingsAndReviews de sample_airbnb, encuentre mediante el uso de agregaciones, cuál es la propiedad con mayor número de servicios (amenities) de la colección.
- Usando la colección listingsAndReviews de sample_airbnb, encuentre mediante el uso de agregaciones, el número de locaciones que tienen conexión a Internet, sea desde Wifi o desde cable (Ethernet). Nota. Revise el campo amenities (“facilidades”)
- Usando la colección listingsAndReviews de sample_airbnb, encuentre mediante el uso de agregaciones, todas las locaciones que tengan 50 o más comentarios, que la valoración sea mayor o igual a 80, que cuenten con conexión a Internet vía cable y estén ubicadas en Brazil.
- Usando la colección listingsAndReviews de sample_airbnb, muestre el costo promedio de una habitación en cada país para las propiedades de tipo casa.

Ejercicio 1

- Usando las colecciones users y comments de la sample_mflix, construye un pipeline que genere como resultados el correo y la contraseña de cada persona que realizó un comentario.
- Usando la base de datos de sample_mflix, obtener el número de películas que hay de cada género por país. Un ejemplo de salida, en formato de tabla, podría ser:

país	género	películas
USA	Short	10
USA	Drama	20
…	…	…

G R A C I A S

Línea de atención al ciudadano: 01 8000 910270
Línea de atención al empresario: 01 8000 910682

@SENAComunica

www.sena.edu.co