

Contenido

PREGUNTAS DEL TALLER	2
MONGODB CONSOLA:	4
MONGODB COMPASS:.....	4
MONGODB BACKEND:.....	14

PREGUNTAS DEL TALLER

Taller de Repaso Bases de Datos No Relacionales

Teniendo en cuenta sus conocimientos previos sobre bases de datos NoSQL, realice la siguiente actividad:

Eres parte del equipo de desarrollo de una aplicación de gestión de inventarios y usuarios. Como desarrollador, te han asignado la tarea de crear y manejar bases de datos con MongoDB. En este taller, usarás MongoDB para:

1. Gestionar una base de datos de usuarios.
2. Administrar una colección de productos.
3. Realizar consultas, actualizaciones y análisis de datos.

Parte 1: Gestión de Usuarios

1. Crea una base de datos llamada tallerMongoDB.
2. Inserta la siguiente información en una colección llamada usuarios:
 - Juan Pérez, 30 años, correo: `juan.perez@urosario.com`
 - Ana López, 25 años, correo: `ana.lopez@urosario.com`
 - Luis Torres, 35 años, correo: `luis.torres@urosario.com`
3. Realiza las siguientes consultas:
 - Encuentra todos los usuarios.
 - Encuentra el usuario con nombre Ana López.
 - Encuentra todos los usuarios mayores o iguales a 30 años.
4. Actualiza los datos:
 - Cambia la edad de Juan Pérez a 31 años.
 - Añade el campo activo: true a todos los usuarios con edad mayor o igual a 30 años.
5. Elimina registros:
 - Elimina el usuario Luis Torres.
 - Elimina todos los usuarios menores de 30 años.

Parte 2: Gestión de Productos

1. Crea una nueva colección llamada productos e inserta al menos 5 productos con los siguientes campos:

- nombre: Nombre del producto.
- precio: Precio del producto.
- categoria: Categoría del producto (por ejemplo: “Electrónica”, “Hogar”, “Ropa”).

2. Realiza las siguientes consultas:

- Encuentra todos los productos con precio mayor a \$100.
- Ordena los productos por precio de manera descendente.

3. Actualiza los datos:

- Añade un campo en stock con valor true a todos los productos.
- Cambia el valor de en stock a false para los productos cuyo precio sea mayor a \$500.

4. Elimina registros:

- Elimina todos los productos cuyo precio sea menor a \$50.

Parte 3: Agregaciones y Análisis

1. Realiza una agregación para calcular:

- El precio promedio de los productos agrupados por categoría.

2. Crea una consulta que devuelva:

- La categoría con el mayor precio promedio.

Desafíos Adicionales

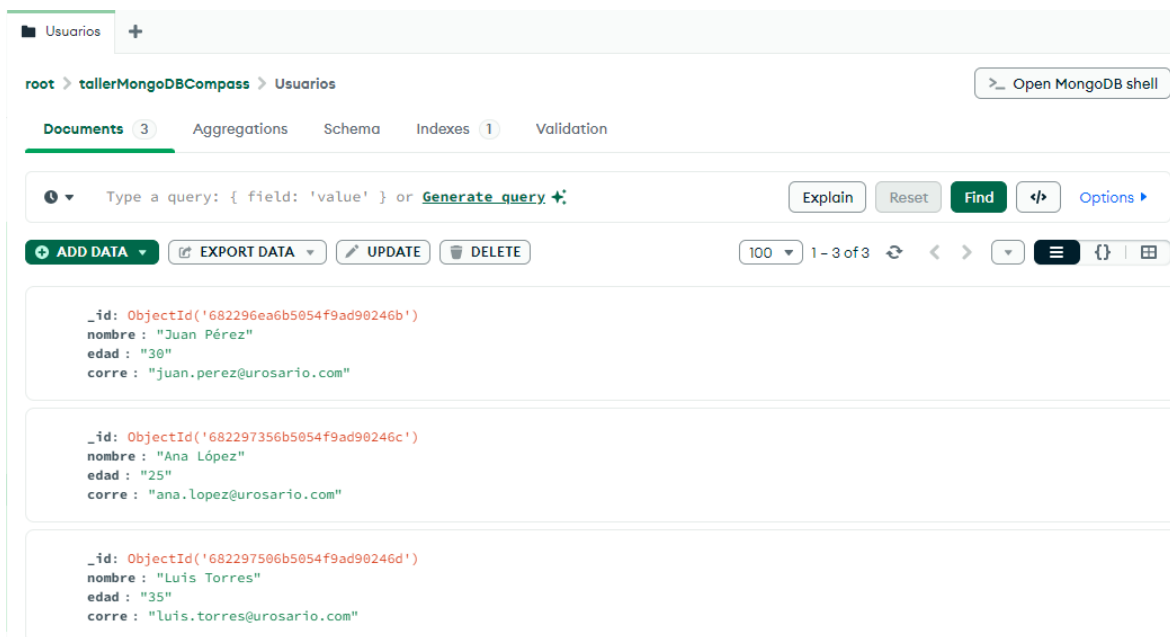
1. **Desafío 1:** Encuentra el número total de productos por categoría.
2. **Desafío 2:** Implementa un sistema de búsqueda en la colección productos que permita encontrar productos cuyo nombre contenga una palabra clave (por ejemplo, “smart”).
3. **Desafío 3:** Usa Node.js para automatizar la inserción de datos en las colecciones y realizar consultas.

MONGODB CONSOLA:

El Código del Script realizado por la Consola se encuentra en el GitHub

MONGODB COMPASS:

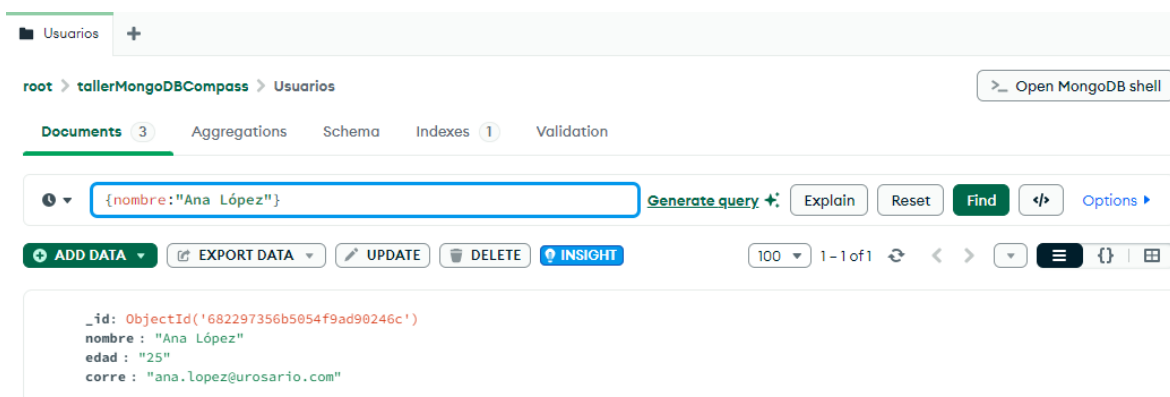
- Inserción de Usuarios y encuentro de todos los usuarios.



The screenshot shows the MongoDB Compass interface for the 'Usuarios' collection. The 'Documents' tab is active, displaying three documents. Each document contains the following fields: '_id' (ObjectId), 'nombre' (string), 'edad' (string), and 'corre' (string).

Document 1	Document 2	Document 3
<code>{ "_id": ObjectId("682296ea6b5054f9ad90246b"), "nombre": "Juan Pérez", "edad": "30", "corre": "juan.perez@urosario.com" }</code>	<code>{ "_id": ObjectId("682297356b5054f9ad90246c"), "nombre": "Ana López", "edad": "25", "corre": "ana.lopez@urosario.com" }</code>	<code>{ "_id": ObjectId("682297506b5054f9ad90246d"), "nombre": "Luis Torres", "edad": "35", "corre": "luis.torres@urosario.com" }</code>

- Consulta de Ana López.



The screenshot shows the MongoDB Compass interface for the 'Usuarios' collection. The 'Documents' tab is active, and a query filter is applied: `{nombre:"Ana López"}`. The query results show one document.

Document 1
<code>{ "_id": ObjectId("682297356b5054f9ad90246c"), "nombre": "Ana López", "edad": "25", "corre": "ana.lopez@urosario.com" }</code>

- Consulta de usuarios mayores o iguales a 30 años.



Query: `{ edad: { $gte: "30" } }`

Buttons: Generate query, Explain, Reset, Find, Options

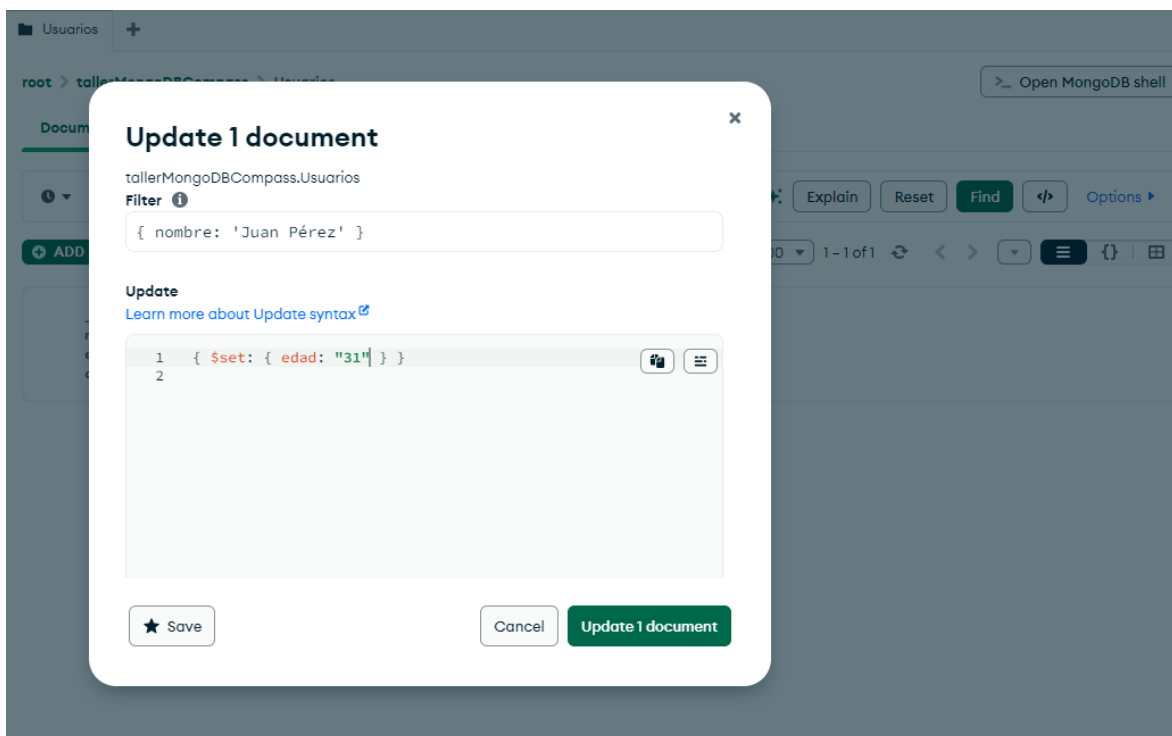
Actions: ADD DATA, EXPORT DATA, UPDATE, DELETE, INSIGHT

Documents:

```
{ "_id": ObjectId('682296ea6b5054f9ad90246b'),  
  "nombre": "Juan Pérez",  
  "edad": "30",  
  "corre": "juan.perez@urosario.com" }
```

```
{ "_id": ObjectId('682297506b5054f9ad90246d'),  
  "nombre": "Luis Torres",  
  "edad": "35",  
  "corre": "Luis.torres@urosario.com" }
```

- Cambiar la edad de Juan Pérez.



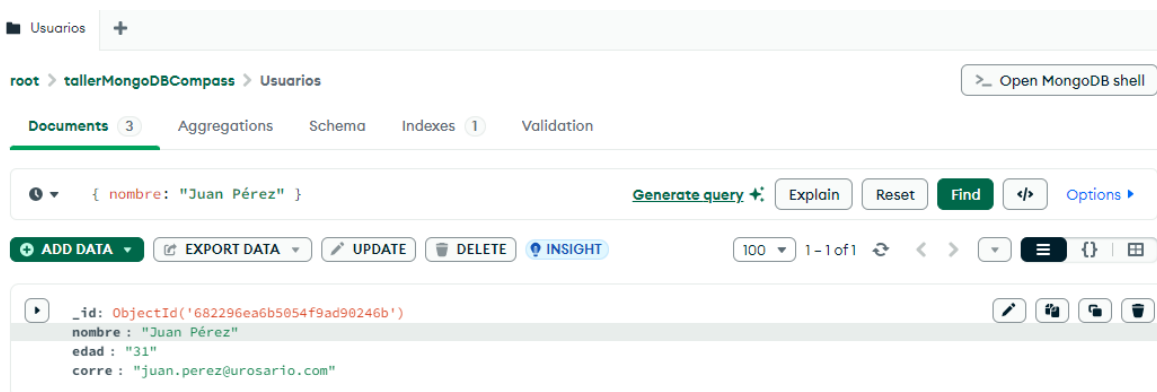
Update 1 document

tallerMongoDBCompass.Usuarios

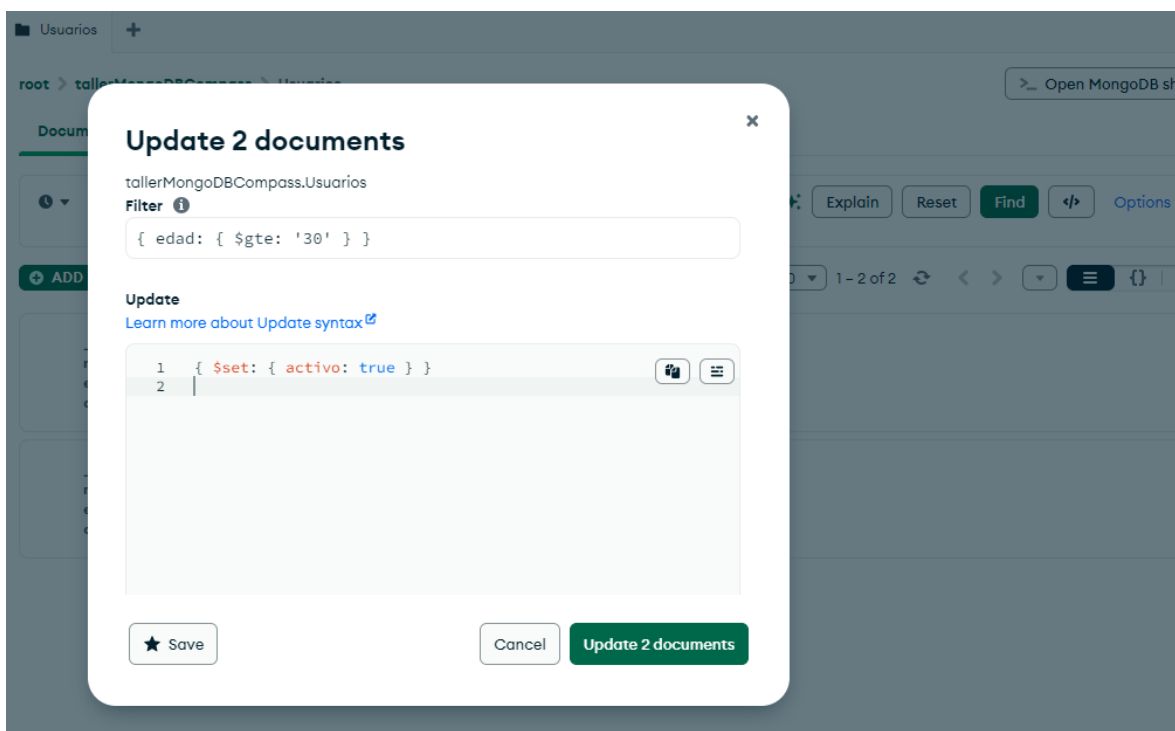
Filter: `{ nombre: 'Juan Pérez' }`

Update: `{ $set: { edad: '31' } }`

Buttons: Save, Cancel, Update 1 document



- Añadir el campo `activo:true` a todos los usuarios con edad mayor o igual a 30 años.



Universidad del Rosario
Ana María Triviño Monje
Taller Base de datos NoSQL

The screenshot shows the MongoDB Compass interface for the 'Usuarios' collection. The 'Documents' tab is active, displaying two documents. The first document is for Juan Pérez, and the second is for Luis Torres. The interface includes a query bar with a filter, buttons for 'Generate query', 'Explain', 'Reset', 'Find', and 'Options'. Below the query bar are buttons for 'ADD DATA', 'EXPORT DATA', 'UPDATE', 'DELETE', and 'INSIGHT'. The document list shows the following data:

Document
<pre>{ "_id": ObjectId("682296ea6b5054f9ad90246b"), "nombre": "Juan Pérez", "edad": "31", "corre": "juan.perez@urosario.com", "activo": true }</pre>
<pre>{ "_id": ObjectId("682297506b5054f9ad90246d"), "nombre": "Luis Torres", "edad": "35", "corre": "luis.torres@urosario.com", "activo": true }</pre>

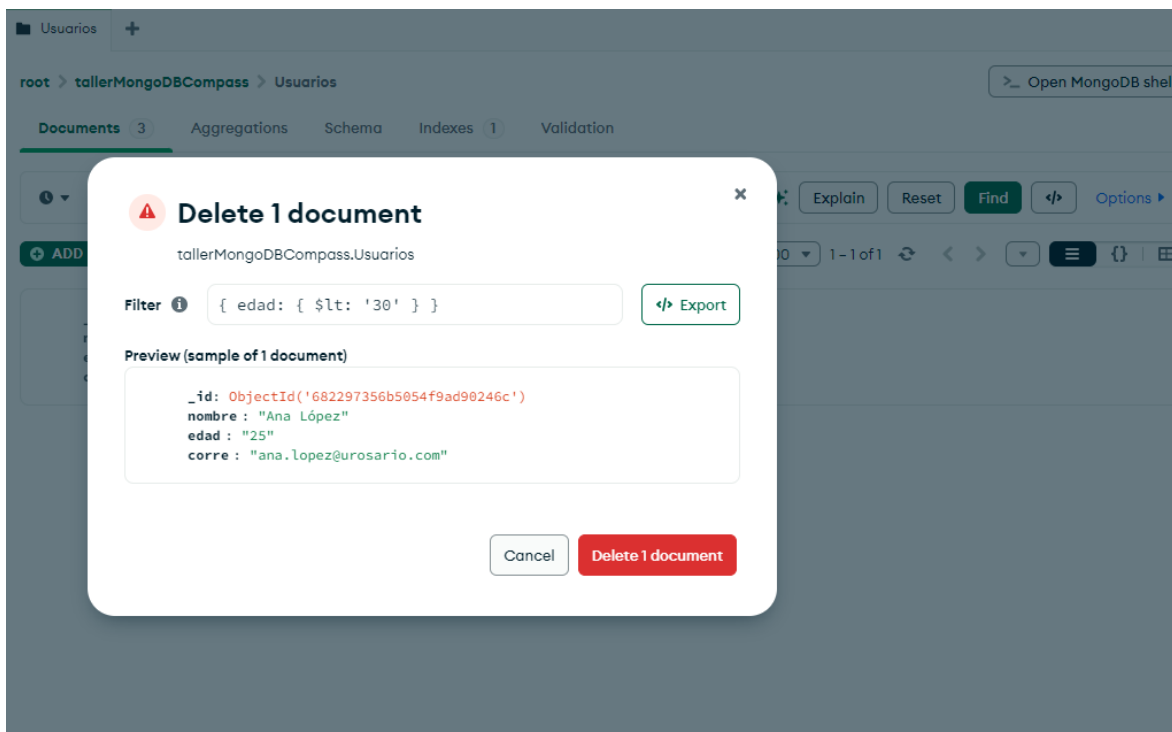
- Eliminar el usuario Luis Torres

The screenshot shows the MongoDB Compass interface with a 'Delete 1 document' dialog box open. The dialog box has a title bar with a red warning icon and a close button. It contains the following information:

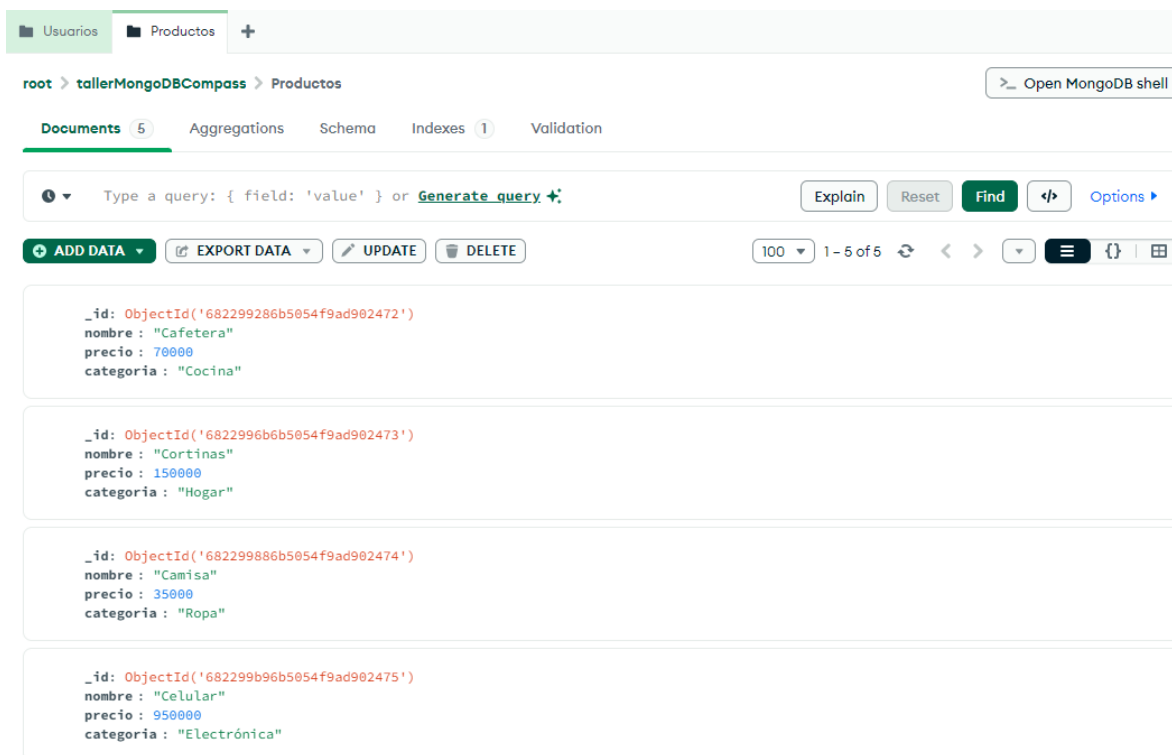
- Delete 1 document**
- Collection: tallerMongoDBCompass.Usuarios
- Filter: { nombre: 'Luis Torres' }
- Export button: </> Export
- Preview (sample of 1 document):

```
{
  "_id": ObjectId("682297506b5054f9ad90246d"),
  "nombre": "Luis Torres",
  "edad": "35",
  "corre": "luis.torres@urosario.com",
  "activo": true
}
```
- Buttons: Cancel, Delete 1 document

- Eliminar todos los usuarios menores de 30 años.



- Creación de productos e inserción de 5 productos



- Consulta de todos los productos con precio mayor a 100,000.

The screenshot shows the MongoDB Compass interface. The top navigation bar has 'Usuarios' and 'Productos' tabs. The breadcrumb trail is 'root > tallerMongoDBCompass > Productos'. The 'Documents' tab is selected, showing 5 documents. A query filter is applied: `{ precio: { $gt: 100000 } }`. The results show two documents:

```
{
  "_id": ObjectId('6822996b6b5054f9ad902473'),
  "nombre": "Cortinas",
  "precio": 150000,
  "categoria": "Hogar"
}
```

```
{
  "_id": ObjectId('6822996b6b5054f9ad902475'),
  "nombre": "Celular",
  "precio": 950000,
  "categoria": "Electrónica"
}
```

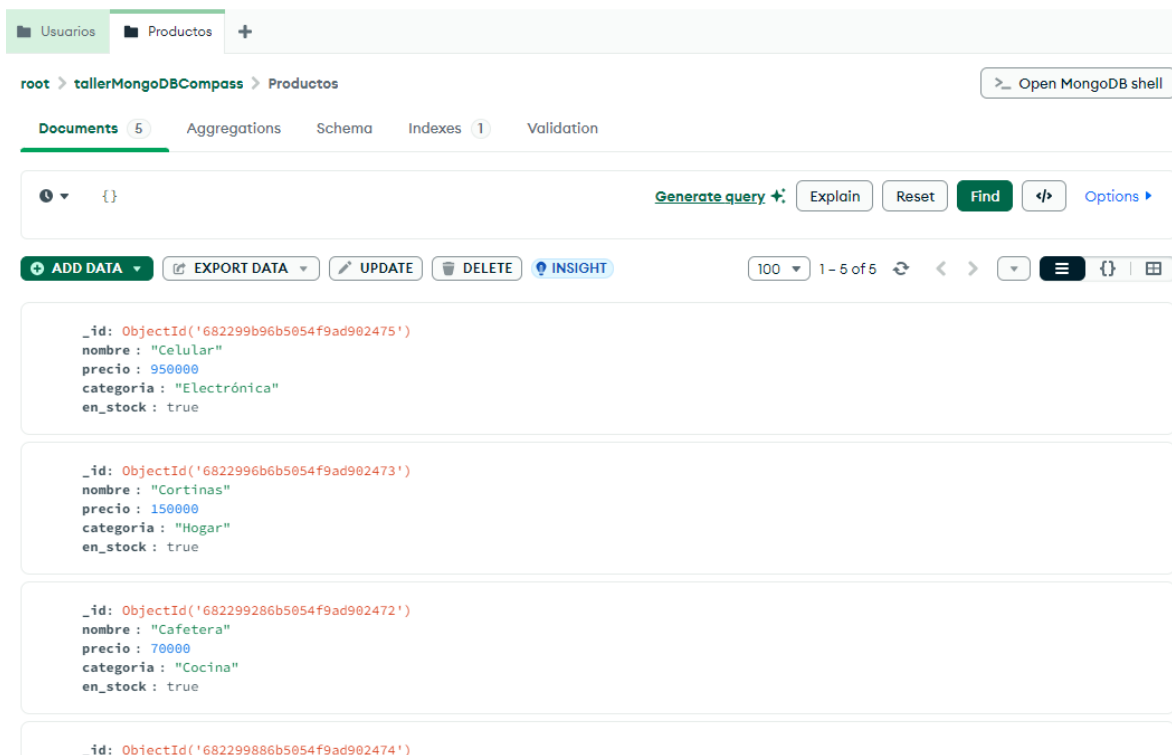
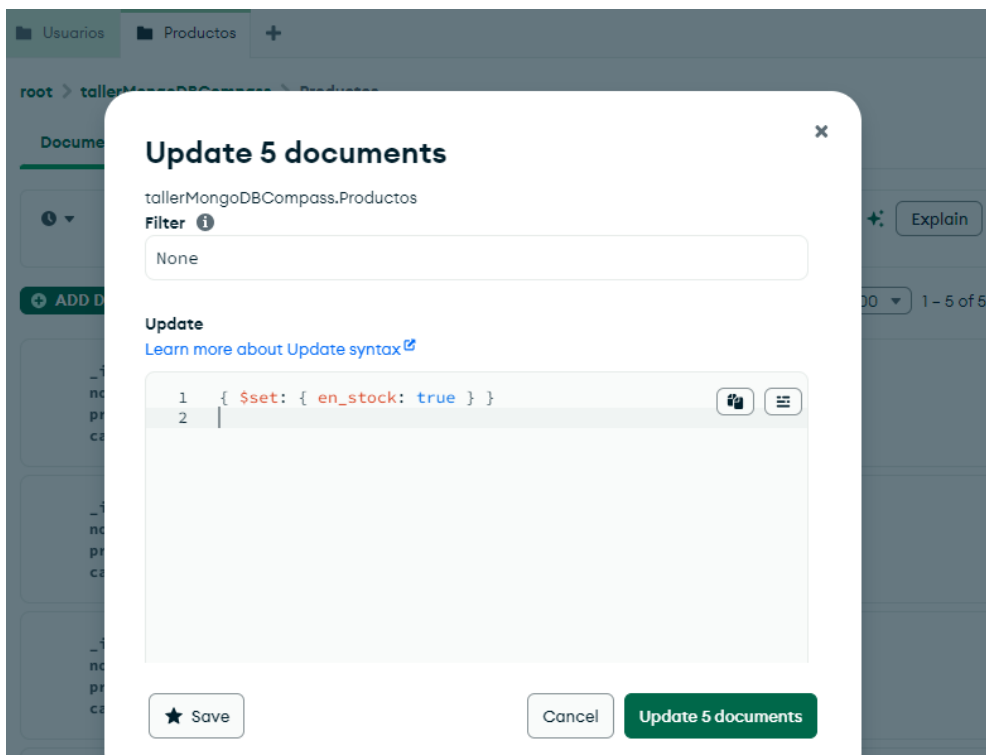
- Ordenar los productos por precio de manera descendente

The screenshot shows the MongoDB Compass interface. The top navigation bar has 'Usuarios' and 'Productos' tabs. The breadcrumb trail is 'root > tallerMongoDBCompass > Productos'. The 'Documents' tab is selected, showing 5 documents. A query filter is applied: `{ field: 'value' }` or `Generate query`. The 'Sort' field is set to `"precio": -1`. The results show two documents:

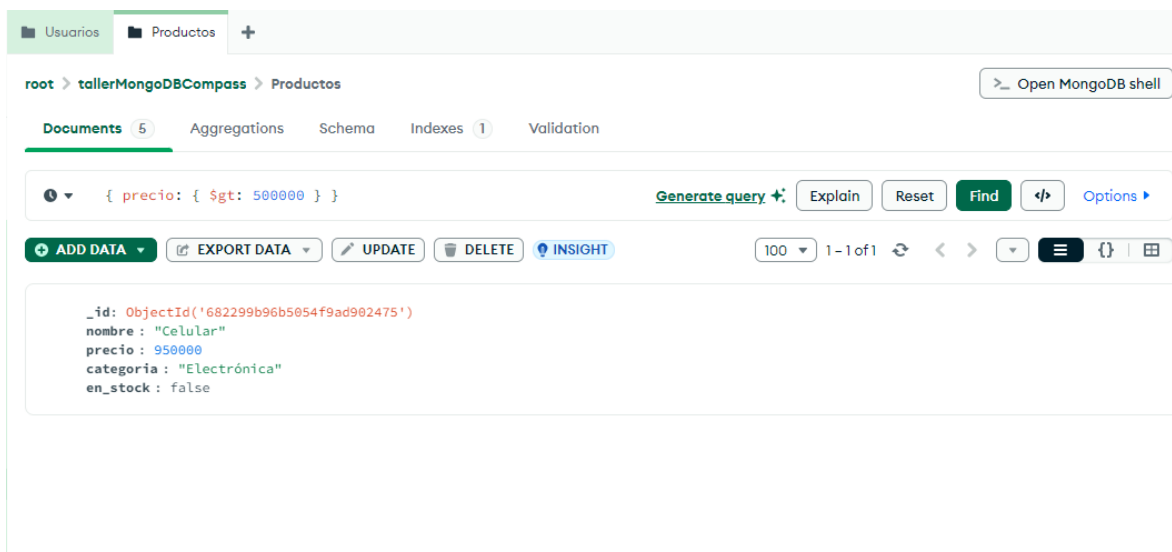
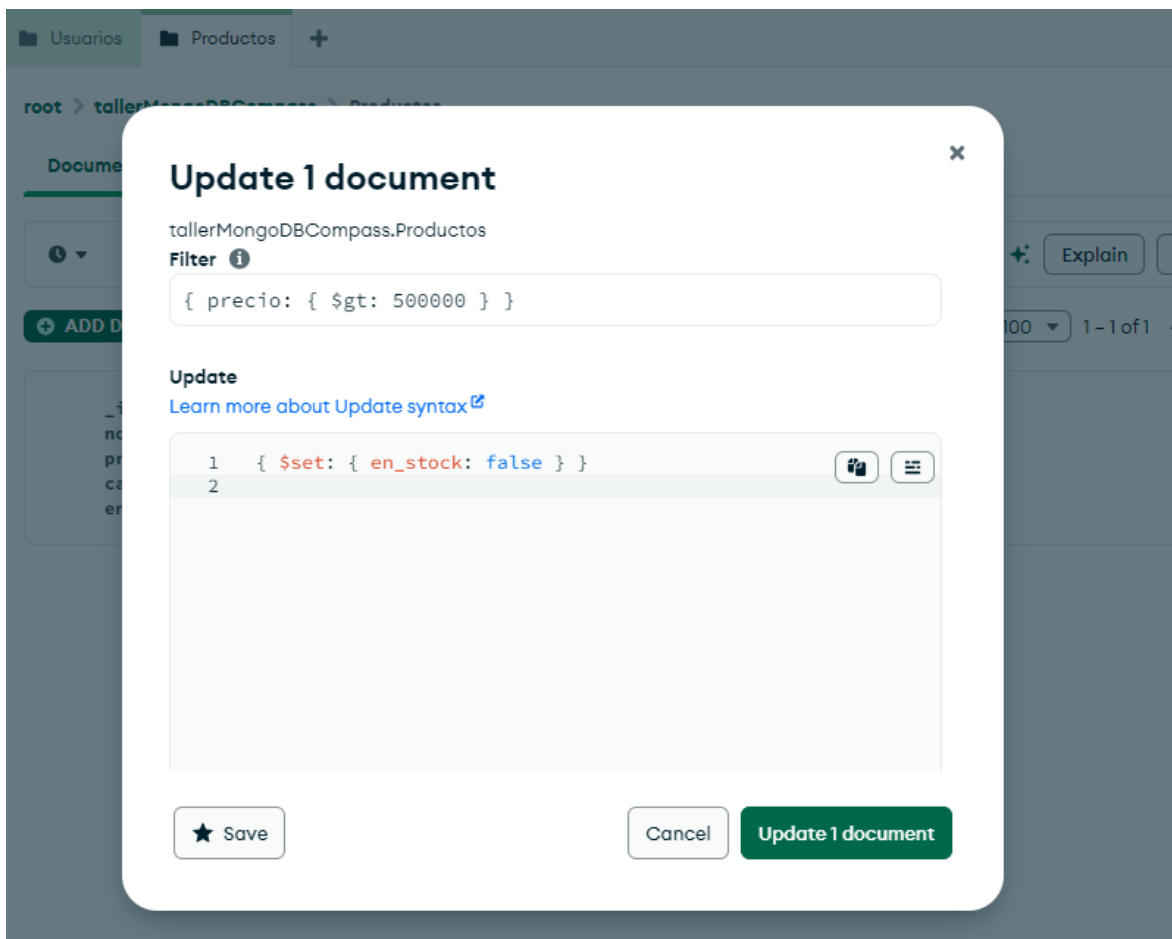
```
{
  "_id": ObjectId('6822996b6b5054f9ad902475'),
  "nombre": "Celular",
  "precio": 950000,
  "categoria": "Electrónica"
}
```

```
{
  "_id": ObjectId('6822996b6b5054f9ad902473'),
  "nombre": "Cortinas",
  "precio": 150000,
  "categoria": "Hogar"
}
```

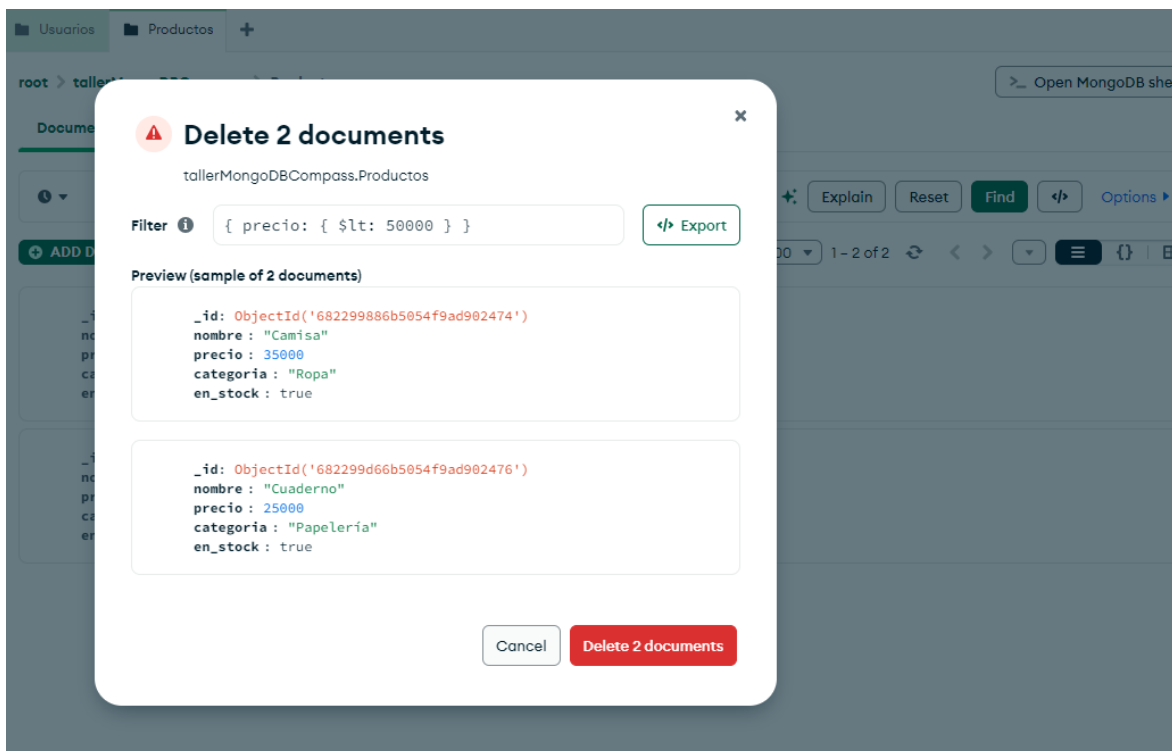
- Agregar un campo de stock con valor true a todos los productos



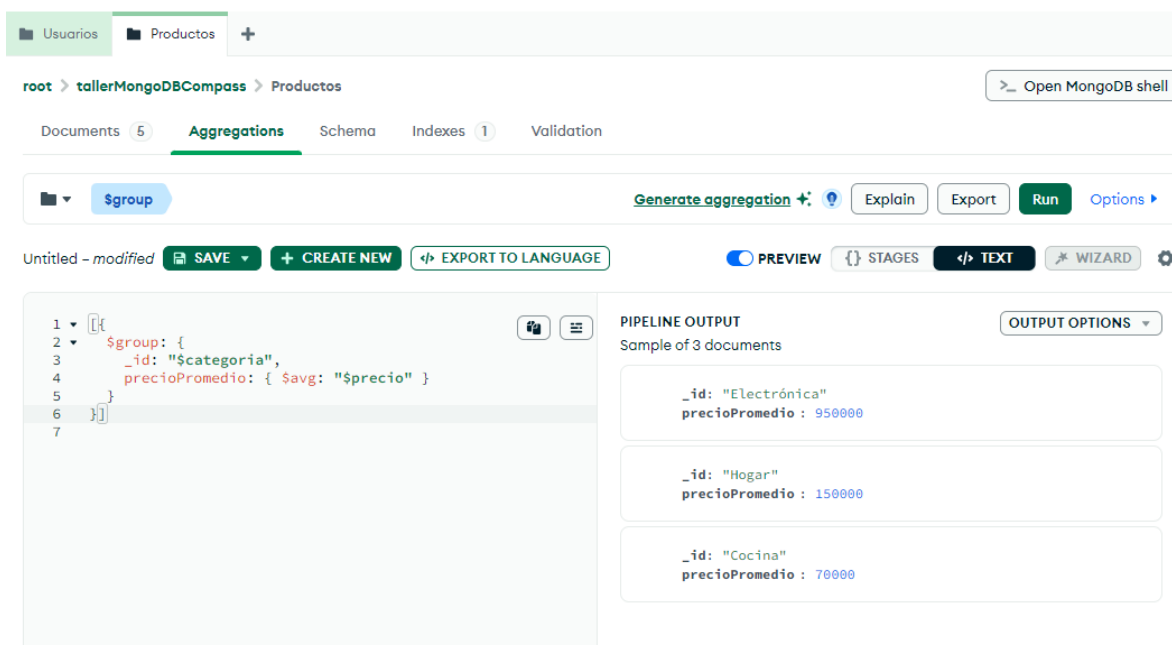
- Cambiar el valor de stock a false para los productos cuyo precio sea mayor a 500000



- Eliminar los productos cuyo precio sea menor a 50,000



- Realizar una agregación para consultar el promedio de los precios de los productos por categoria



- Realizar una agregación para consular la categoría con el mayor precio promedio

The screenshot shows the MongoDB Compass interface. The 'Productos' collection is selected. The 'Aggregations' tab is active. The pipeline consists of three stages: \$group, \$sort, and \$limit. The \$group stage groups by '_id: "\$categoria"' and calculates the average price 'precioPromedio: { \$avg: "\$precio" }'. The \$sort stage sorts by 'precioPromedio: -1'. The \$limit stage limits the results to 1. The pipeline output shows a single document for the 'Electrónica' category with an average price of 950000.

```
1 [{
2   $group: {
3     _id: "$categoria",
4     precioPromedio: { $avg: "$precio" }
5   },
6 },
7 {
8   $sort: { precioPromedio: -1 }
9 },
10 {
11   $limit: 1
12 }
13 ]
```

PIPELINE OUTPUT
Sample of 1 document

```
{
  "_id": "Electrónica",
  "precioPromedio": 950000
}
```

- Encontrar el número total de productos por categoría

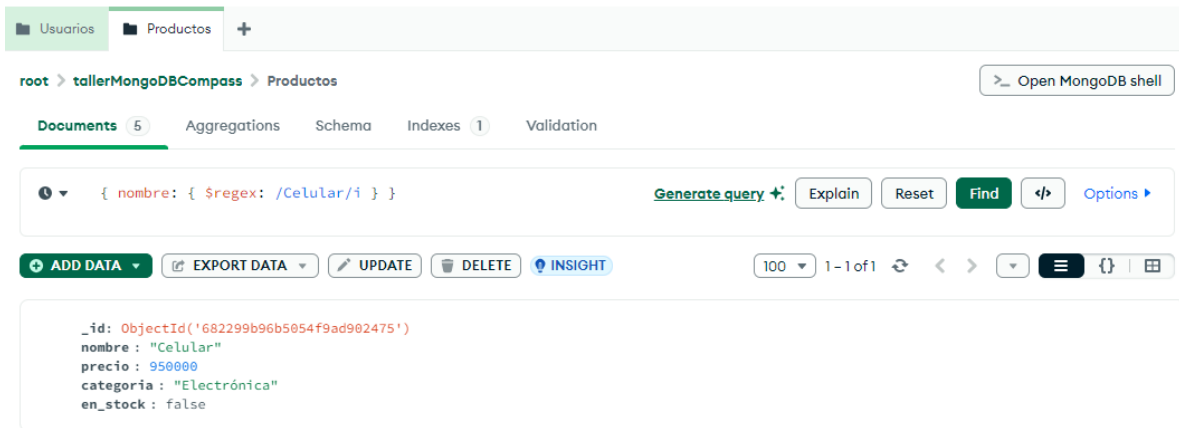
The screenshot shows the MongoDB Compass interface. The 'Productos' collection is selected. The 'Aggregations' tab is active. The pipeline consists of one stage: \$group. The \$group stage groups by '_id: "\$categoria"' and calculates the total number of products 'totalProductos: { \$sum: 1 }'. The pipeline output shows three documents for the categories 'Electrónica', 'Hogar', and 'Cocina', each with a total of 1 product.

```
1 [{
2   $group: {
3     _id: "$categoria",
4     totalProductos: { $sum: 1 }
5   },
6 },
7 ]
```

PIPELINE OUTPUT
Sample of 3 documents

```
{
  "_id": "Electrónica",
  "totalProductos": 1
}
{
  "_id": "Hogar",
  "totalProductos": 1
}
{
  "_id": "Cocina",
  "totalProductos": 1
}
```

- Implementar un sistema de búsqueda en la colección productos que permite encontrar productos cuyo nombre contenga una palabra clave



MONGODB BACKEND: