

INSTITUTO TECNOLÓGICO DEL PUTUMAYO

Bases de datos y almacenamiento masivo

Informe base de datos Nosql

Deisy Estefania Burbano e Ingry Nathaly Silva

Septiembre 2024

TABLA DE CONTENIDOS

Bases de datos y almacenamiento masivo	1
Resumen Ejecutivo.....	3
Introducción.....	4
<i>Contexto y Motivación.....</i>	<i>4</i>
<i>Alcance del Informe.....</i>	<i>4</i>
<i>Objetivos.....</i>	<i>4</i>
Metodología.....	5
<i>Herramientas Utilizadas</i>	<i>5</i>
<i>Procedimientos.....</i>	<i>5</i>
Desarrollo del Informe.....	11
<i>Descripción de la Base de Datos</i>	<i>11</i>
<i>Consultas NoSQL.....</i>	<i>13</i>
<i>Diseño de Base de Datos</i>	<i>21</i>
Análisis y Discusión	21
<i>Interpretación de Resultados</i>	<i>21</i>
Conclusiones	24
Recomendaciones	25
Referencias	26

Resumen Ejecutivo

Este informe presenta un análisis exhaustivo de una base de datos NoSQL en MongoDB, enfocándose en la gestión y consulta de datos relacionados con productos, proveedores y ventas. El informe detalla el esquema de la base de datos, las consultas realizadas para extraer y manipular datos, y el diseño de la base de datos. También se incluyen interpretaciones de los resultados obtenidos y recomendaciones para mejorar la gestión de datos en la base de datos.

Introducción

Contexto y Motivación

En el entorno actual, la gestión eficiente de datos es crucial para el éxito de las empresas. La base de datos NoSQL en MongoDB proporcionada contiene información sobre productos, proveedores, clientes y ventas en una tienda de moda retro. La capacidad de realizar consultas eficaces y manejar datos correctamente es vital para la toma de decisiones informadas.

Alcance del Informe

Este informe cubre los siguientes aspectos de NoSQL:

- Consultas: Ejecución y resultados de consultas para obtener, actualizar y analizar datos.
- Optimización: Mejoras en el manejo de datos y consultas.
- Diseño de Bases de Datos: Estructura y organización de la base de datos.

Objetivos

El objetivo del informe es:

- Analizar el esquema de la base de datos.
- Ejecutar y evaluar consultas para obtener información útil.
- Explicar el diseño de la base de datos y las consultas realizadas.
- Ofrecer recomendaciones para mejorar la gestión de datos.

Metodología

Link del repositorio GitHub: https://github.com/Nia-jd/DB_ALMACENAMIENTO_MASIVO_DB_NOSQL.git

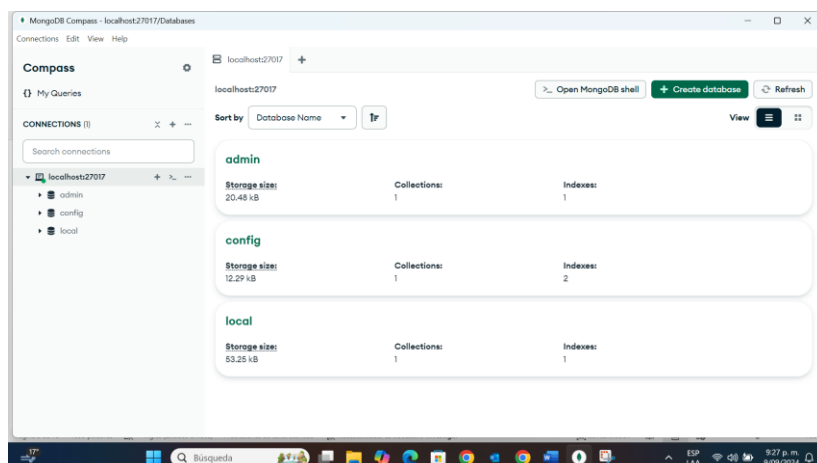
Herramientas Utilizadas

- MongoDB: Sistema de base de datos NoSQL utilizado para almacenar y gestionar los datos.
- MongoDB Shell: Herramienta de línea de comandos para ejecutar consultas y comandos.
- MongoDB Database Tools: Herramienta utilizada para exportar la base de datos.
- GitHub: Plataforma de desarrollo colaborativo basada en la web que utiliza el sistema de control de versiones Git, donde se subió el repositorio con la base de datos e información por medio de commits.

Procedimientos

- Creación de la base de datos: La base de datos se creó de la siguiente manera.

Una vez instalado el MongoDB, ingresamos a la aplicación y nos encontramos con la siguiente interfaz:



En la parte superior encontramos el siguiente botón:



Al presionar el botón nos aparecerá la siguiente ventana, aquí lo que tenemos que hacer es asignar el nombre de nuestra base de datos, crear una collection para poner iniciar y presionamos el botón Create Database.

localhost:27017 >_ Open MongoDB sh

Create Database

Database Name

SOLO_MODA_VINTAGE

Collection Name

> Additional preferences (e.g. Custom collation, Capped, Clustered collections)

i Before MongoDB can save your new database, a collection name must also be specified at the time of creation. [More Information](#)

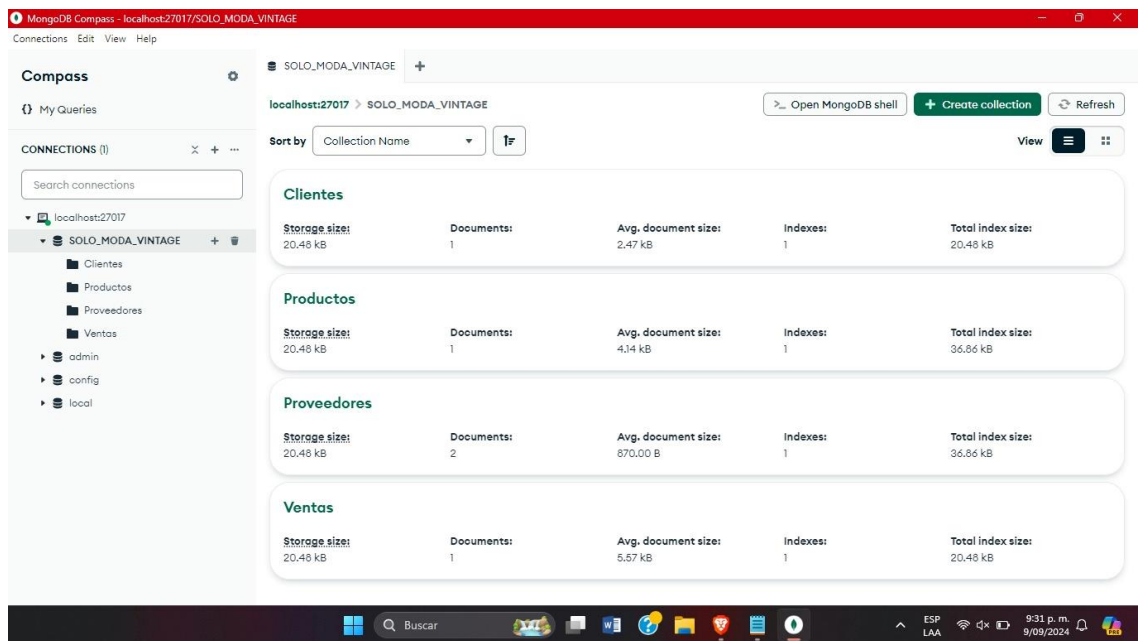
Cancel Create Database

universidad

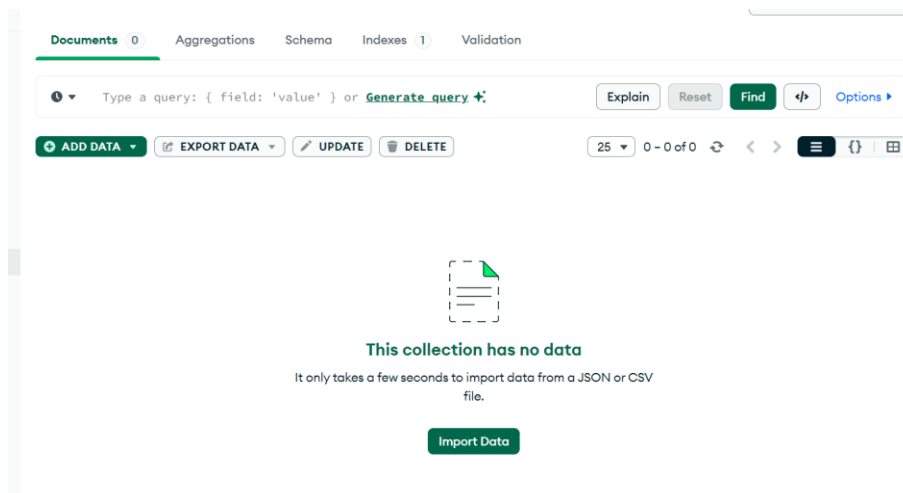
Luego vamos a poder visualizar nuestra base de datos con la collection creada y procedemos a crear las demás collecciones para poder llenar información, para esto en la parte superior nos aparecerá el siguiente botón que debemos presionar.

+ Create collection

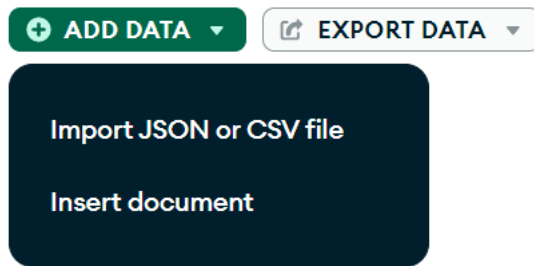
Una vez creadas las collecciones necesarias para nuestra base de datos nos vamos a encontrar con una interfaz como la que se muestra a continuación.



Para poder llenar datos en cada collection, vamos a presionar encima de ella lo cual nos llevará a la siguiente ventana.



Para llenar cada collection de datos presionamos el siguiente botón (ADD DATA) y elegimos la opción que queramos para llenar los datos de nuestra base de datos y procedemos a realizarlo.



- Instalación de MongoDB Database Tools: Instalación de herramientas necesarias para exportar la base de datos.

Para realizar esta instalación simplemente seguimos las instrucciones brindadas por nuestro docente.

1. [ps://www.mongodb.com/try/download/database-tools](https://www.mongodb.com/try/download/database-tools) ->Descargar Database Tools

2. Instalar Database Tools:

*Sigue las instrucciones de instalación.

* Si deseas que estas herramientas estén disponibles en cualquier terminal, añade la carpeta de las herramientas al PATH del sistema

* Buscar la carpeta de tools en mongodb e ingresar esta ruta en el PATH del sistema

3. Abrir una terminal de Power Shell y lanzar el siguiente comando

* `mongoexport --version`

5. Ruta para exportar colección `cd "D:\ITP\BASE DE DATOS Y ALMACENAMIENTO MASIVO\dbMasivo\backups"`

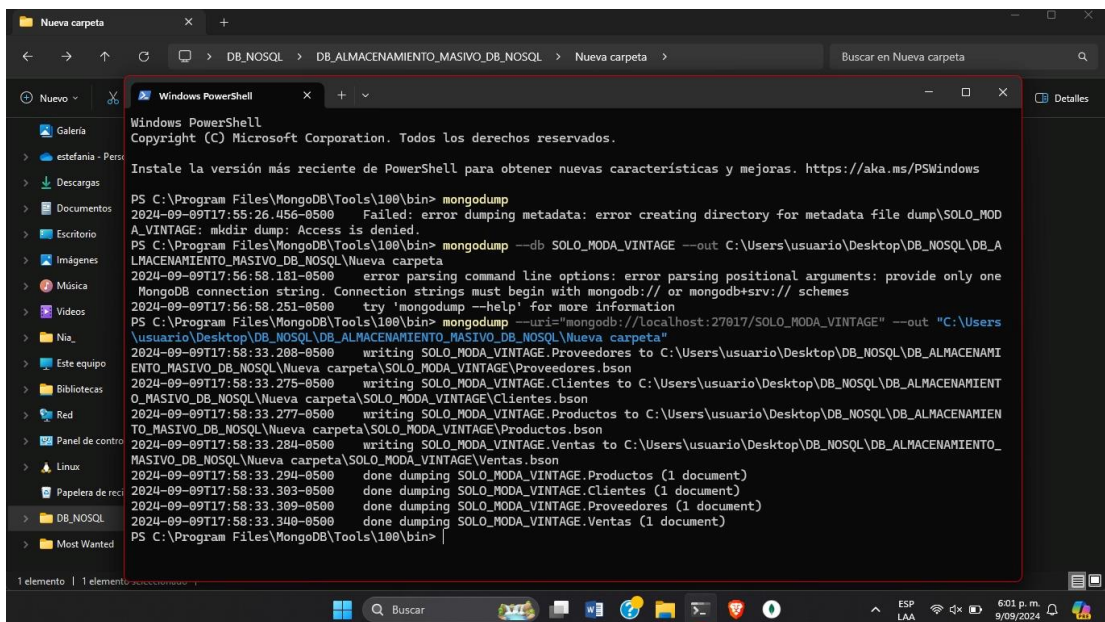
4. comando para exportar mongoexport --db miNuevaBaseDeDatos --collection books
--out books.json

response

2024-08-29T17:51:14.243-0500 connected to: mongodb://localhost/

2024-08-29T17:51:14.309-0500 exported 2 records

En la siguiente imagen se muestra la correcta exportación de nuestra base de datos después de haber realizado los pasos anteriores.



```

Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\Program Files\MongoDB\Tools\100\bin> mongoexport
2024-08-29T17:51:14.243-0500 Failed: error dumping metadata: error creating directory for metadata file dump\SOLO_MOD
A_VINTAGE: mkdir dump: Access is denied.
PS C:\Program Files\MongoDB\Tools\100\bin> mongoexport --db SOLO_MOD --collection SOLO_MOD_VINTAGE --out "C:\Users\usuario\Desktop\DB_NOSQL\DB_ALMACENAMIENTO_MASIVO_DB_NOSQL\Nueva carpeta\SOLO_MOD_VINTAGE.json"
2024-08-29T17:51:14.309-0500 exported 2 records
2024-08-29T17:51:14.340-0500
PS C:\Program Files\MongoDB\Tools\100\bin>
  
```

- Revisión del Esquema de la Base de Datos: Examinar las colecciones productos, proveedores, clientes y ventas para entender su estructura.
- Ejecución de Consultas: Realizar consultas para obtener datos específicos, actualizar registros y calcular estadísticas.
- Análisis de Resultados: Evaluar los resultados obtenidos para interpretar los datos.

- **Diseño de la Base de Datos:** Revisar la estructura y diseño de las colecciones para asegurar la eficiencia.

Desarrollo del Informe

Descripción de la Base de Datos

La base de datos contiene cuatro colecciones principales:

- **Productos:** Información sobre los artículos disponibles en la tienda.
- **Proveedores:** Datos de los proveedores que suministran los productos.
- **Clientes:** Información sobre los clientes de la tienda.
- **Ventas:** Registros de transacciones de ventas.

En la imagen se puede evidenciar lo mencionado anteriormente.

Collection Name	Storage size	Documents	Avg. document size	Indexes	Total index size
Clientes	20.48 kB	1	2.47 kB	1	20.48 kB
Productos	20.48 kB	1	4.14 kB	1	36.86 kB
Proveedores	20.48 kB	2	870.00 B	1	36.86 kB
Ventas	20.48 kB	1	5.57 kB	1	20.48 kB

Esquema de la Base de Datos

- **Productos:** nombre, categoria, precio, stock, descripcion, talla, proveedor_id

Type a query: { field: 'value' } or [Generate query](#) ↗

[ADD DATA](#)
[EXPORT DATA](#)
[UPDATE](#)
[DELETE](#)

```

_id: ObjectId('66df6901c79af50dfdc13c54')
▼ productos : Array (20)
  ▼ 0: Object
    nombre : "Vestido Vintage Floral"
    categoria : "Ropa"
    precio : 49.99
    stock : 20
    descripcion : "Vestido con estampado floral de los años 70."
    ▼ talla : Array (3)
      0: "S"
      1: "M"
      2: "L"
    proveedor_id : "prov1"
  ▼ 1: Object
    nombre : "Chaqueta de Cuero Retro"
    categoria : "Ropa"
    precio : 89.99
    stock : 10
    descripcion : "Chaqueta de cuero auténtica estilo retro."
    ▼ talla : Array (2)
      0: "M"
      1: "L"

```

- Proveedores: proveedor_id, nombre, contacto, telefono, dirección

[ADD DATA](#)
[EXPORT DATA](#)
[UPDATE](#)
[DELETE](#)

```

_id: ObjectId('66df6b6fc79af50dfdc13c5e')
▼ proveedores : Array (10)
  ▼ 0: Object
    proveedor_id : "prov1"
    nombre : "Moda Retro SA"
    contacto : "Carlos Mendoza"
    telefono : "555-2345"
    direccion : "Calle Moda 789, Ciudad"
  ▼ 1: Object
    proveedor_id : "prov2"
    nombre : "Vintage Leather"
    contacto : "Laura García"
    telefono : "555-6789"
    direccion : "Avenida Cuero 321, Ciudad"
  ▼ 2: Object
    proveedor_id : "prov3"
    nombre : "Retro Chic"
    contacto : "Javier Fernández"
    telefono : "555-9876"
    direccion : "Plaza Vintage 45, Ciudad"
  ▼ 3: Object
    proveedor_id : "prov4"
    nombre : "Old School Fabrics"
    contacto : "María López"

```

- Clientes: cliente_id, nombre, email, telefono, dirección

Documents 1 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#) ✨

ADD DATA EXPORT DATA UPDATE DELETE

```

_id: ObjectId('66df69f8c79af50dfdc13c59')
clientes: Array (15)
  0: Object
    nombre: "Ana García"
    email: "ana.garcia@example.com"
    telefono: "555-1234"
    direccion: "Calle Falsa 123, Ciudad"
    historial_compras: Array (empty)
  1: Object
    nombre: "Juan Pérez"
    email: "juan.perez@example.com"
    telefono: "555-5678"
    direccion: "Avenida Siempre Viva 456, Ciudad"
    historial_compras: Array (empty)
  2: Object
    nombre: "Laura Martínez"
    email: "laura.martinez@example.com"
    telefono: "555-8765"
    direccion: "Calle del Sol 789, Ciudad"
    historial_compras: Array (empty)
  3: Object
    nombre: "Carlos López"
    email: "carlos.lopez@example.com"
    telefono: "555-6543"

```

- Ventas: venta_id, cliente, producto, cantidad, precio_total, fecha, proveedor

ADD DATA EXPORT DATA UPDATE DELETE

```

_id: ObjectId('66df6ee3c79af50dfdc13c67')
ventas: Array (30)
  0: Object
    venta_id: "venta001"
    cliente: "Ana García"
    producto: "Vestido Vintage Floral"
    cantidad: 1
    precio_total: 49.99
    fecha: "2024-09-01"
    proveedor: "Moda Retro SA"
  1: Object
    venta_id: "venta002"
    cliente: "Juan Pérez"
    producto: "Chaqueta de Cuero Retro"
    cantidad: 1
    precio_total: 89.99
    fecha: "2024-09-02"
    proveedor: "Vintage Leather"
  2: Object
    venta_id: "venta003"
    cliente: "Laura Martínez"
    producto: "Camisa de Rayas Vintage"
    cantidad: 2
    precio_total: 59.98

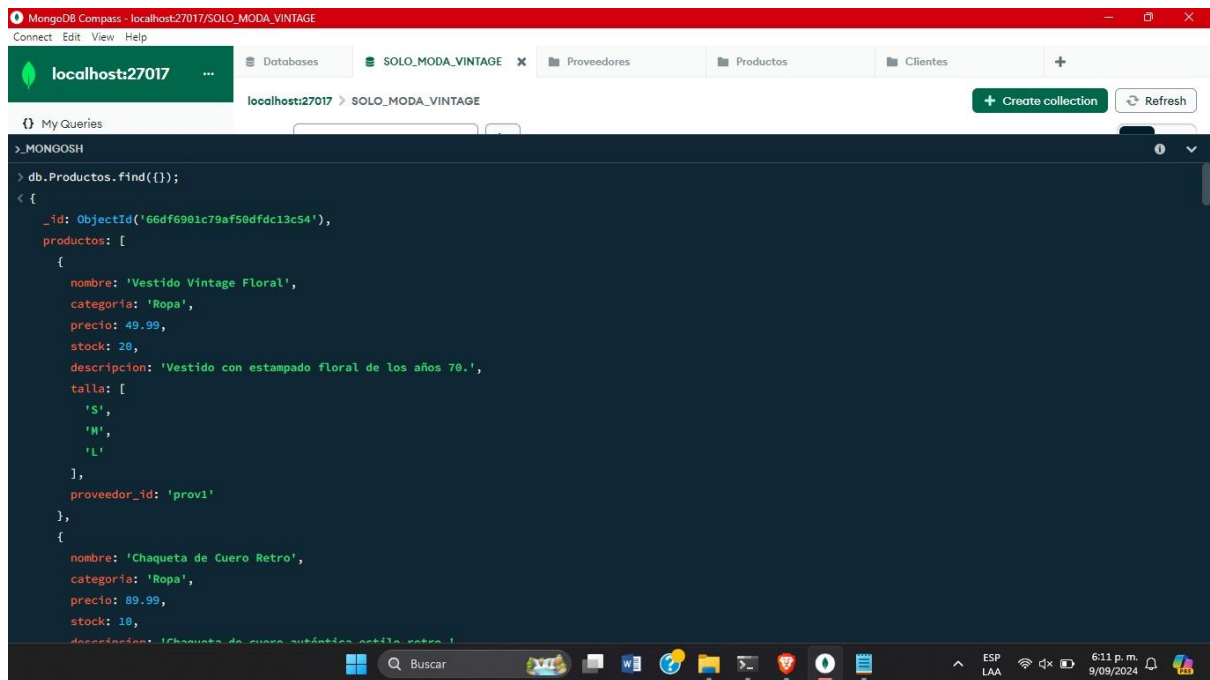
```

Consultas NoSQL

Consultas Realizadas

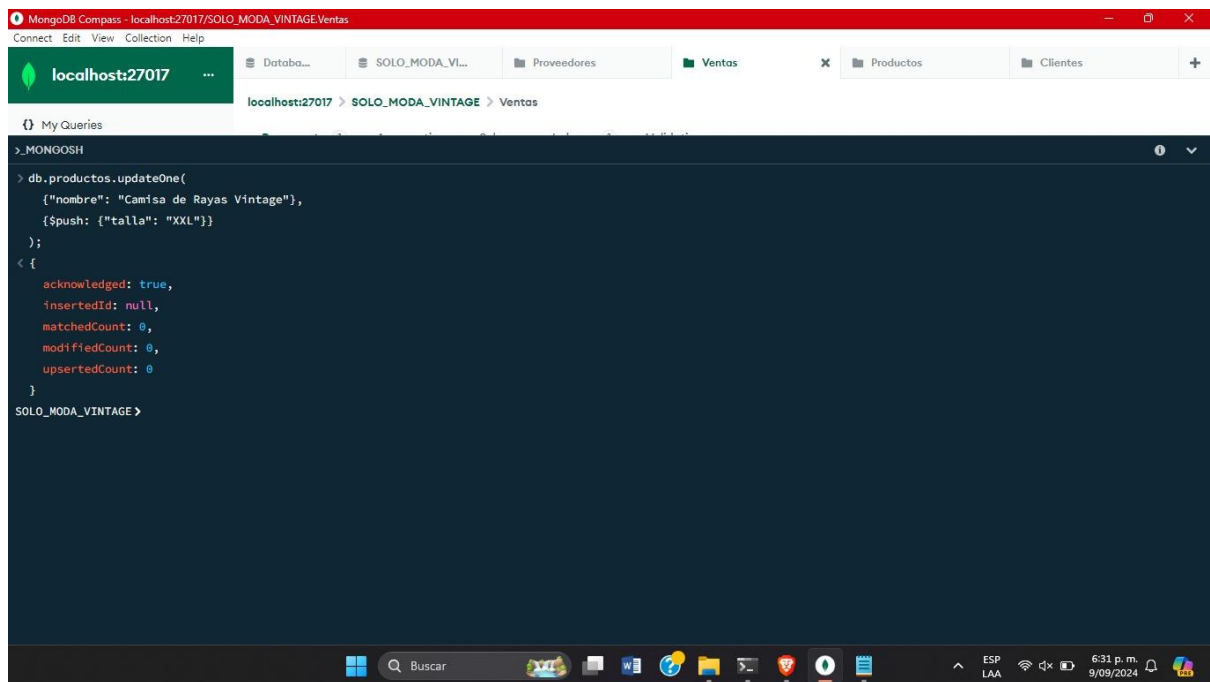
1. Obtener todos los productos:

```
db.productos.find({});
```



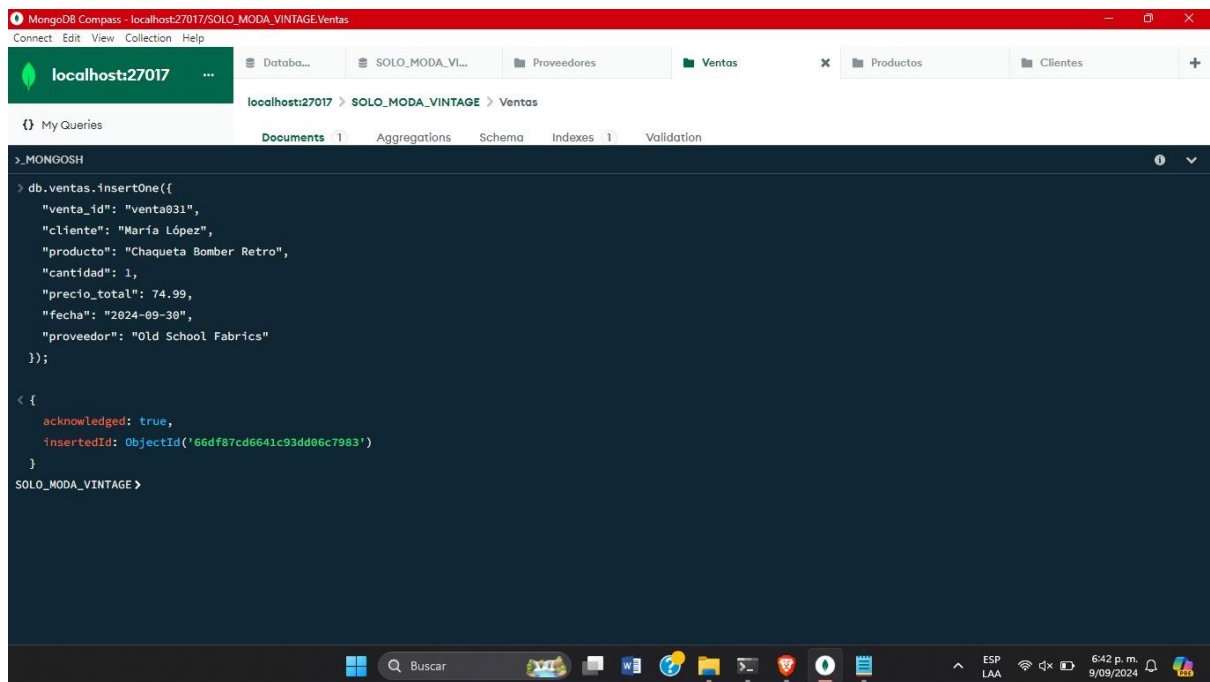
2. Agregar una nueva talla a un producto:

```
db.productos.updateOne(
  {"nombre": "Camisa de Rayas Vintage"},
  {$push: {"talla": "XXL"}}
);
```



3. Insertar una nueva venta:

```
db.ventas.insertOne({
  "venta_id": "venta031",
  "cliente": "María López",
  "producto": "Chaqueta Bomber Retro",
  "cantidad": 1,
  "precio_total": 74.99,
  "fecha": "2024-09-30",
  "proveedor": "Old School Fabrics"
});
```

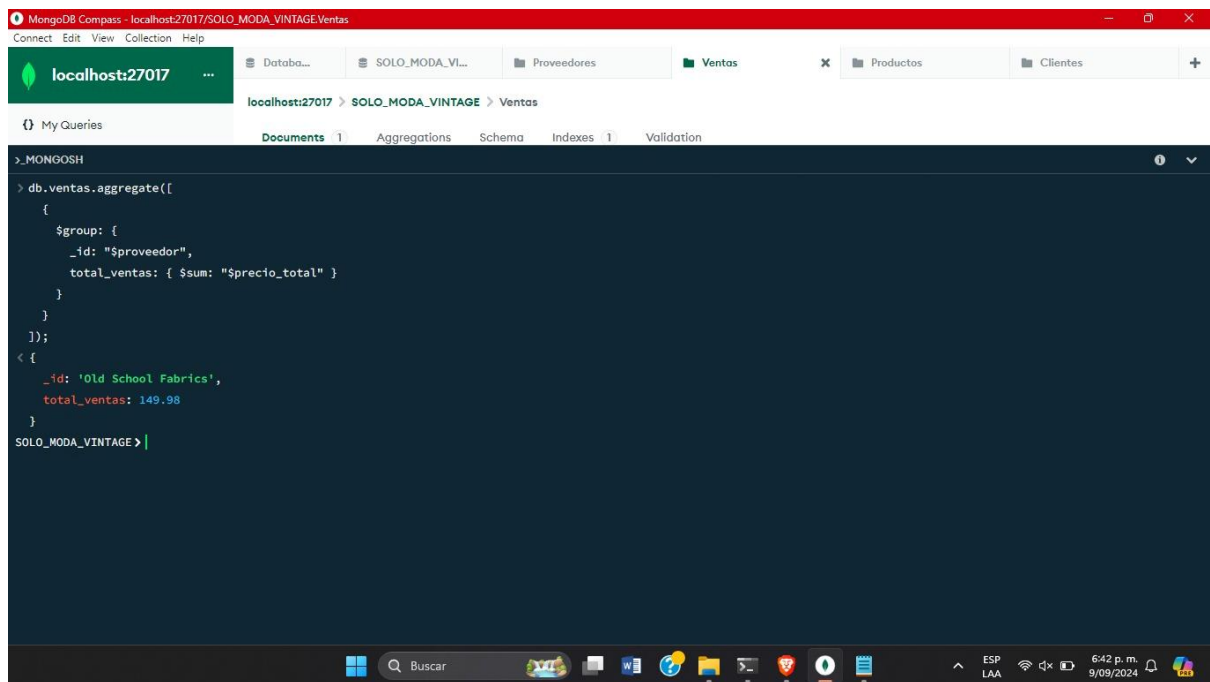


4. Total de ventas por proveedor:

```

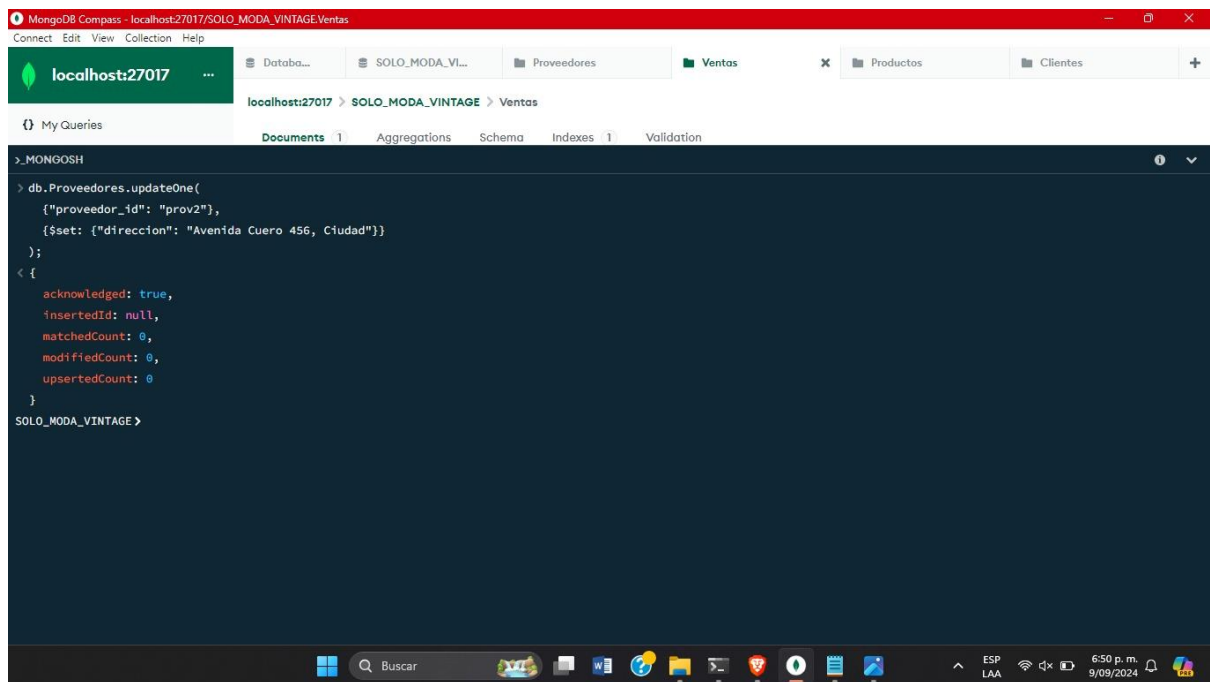
db.ventas.aggregate([
  {
    $group: {
      _id: "$proveedor",
      total_ventas: { $sum: "$precio_total" }
    }
  }
]);

```

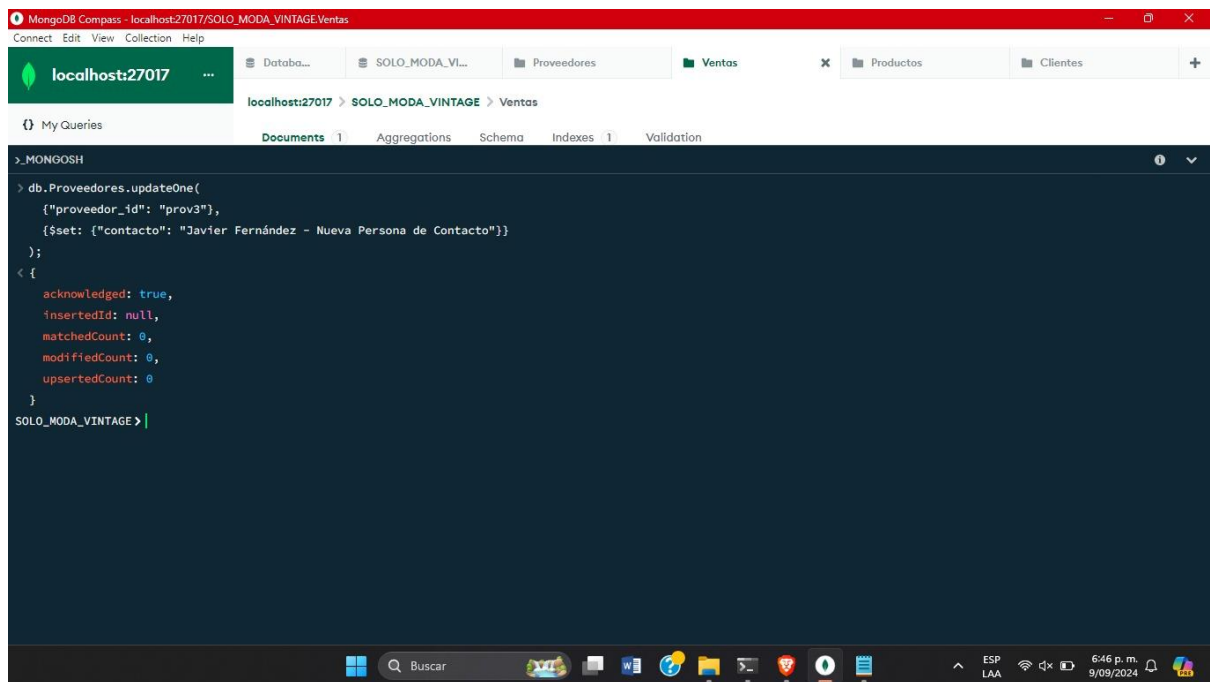
5. Actualizar la dirección de un proveedor:

```
db.proveedores.updateOne(
  {"proveedor_id": "prov2"},
  {$set: {"direccion": "Avenida Cuero 456, Ciudad"}}
);
```



6. Agregar un nuevo contacto a un proveedor:

```
db.proveedores.updateOne(
  {"proveedor_id": "prov3"},
  {$set: {"contacto": "Javier Fernández - Nueva Persona de Contacto"}}
);
```

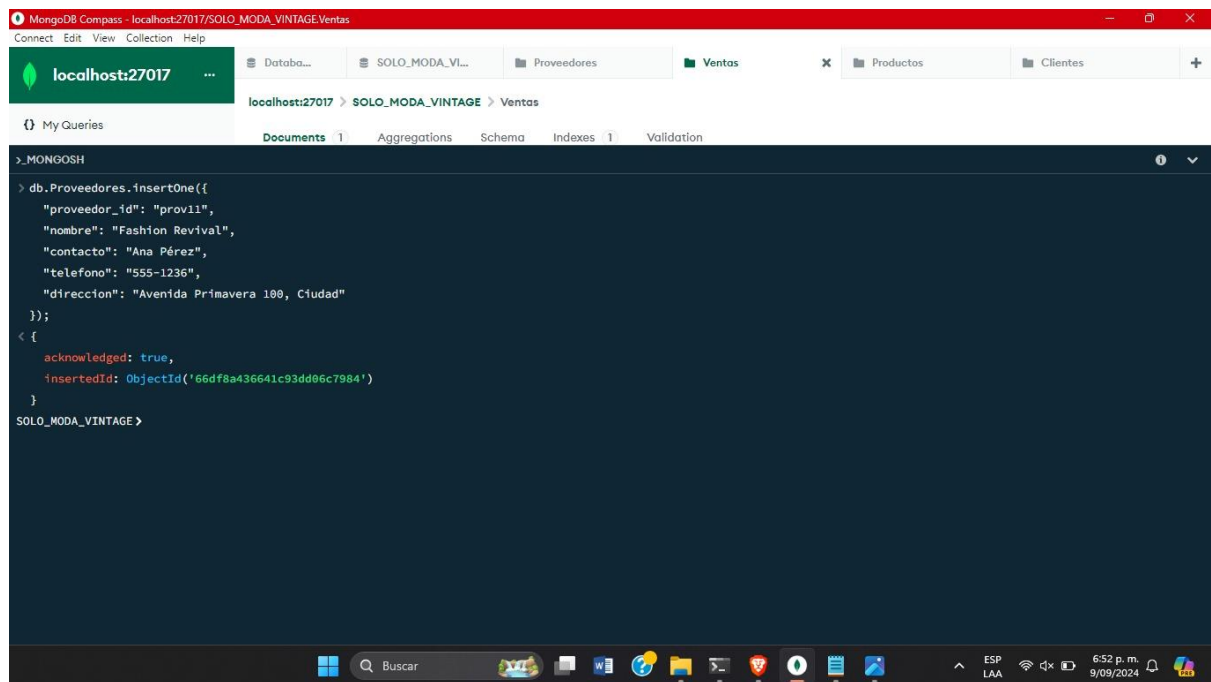


7. Insertar un nuevo proveedor:

```

db.proveedores.insertOne({
  "proveedor_id": "prov11",
  "nombre": "Fashion Revival",
  "contacto": "Ana Pérez",
  "telefono": "555-1236",
  "direccion": "Avenida Primavera 100, Ciudad"
});

```



Resultados de Consultas

- Productos: La consulta `find({})` devuelve todos los productos en la base de datos.
- Nueva Talla: Se agregó la talla "XXL" al producto "Camisa de Rayas Vintage".
- Nueva Venta: Se registró una venta de "Chaqueta Bomber Retro" el 30 de septiembre de 2024.
- Total de Ventas por Proveedor: Se obtiene el total de ventas acumuladas por cada proveedor.
- Dirección Actualizada: La dirección del proveedor "Vintage Leather" se actualizó.
- Nuevo Contacto: Se actualizó el contacto del proveedor "Retro Chic".
- Nuevo Proveedor: Se insertó un nuevo proveedor, "Fashion Revival".

Explicación de Consultas

- Consulta `find({})`: Recupera todos los documentos de la colección productos.
- Consulta `updateOne` con `$push`: Añade una nueva talla al array talla de un producto específico.
- Consulta `insertOne`: Inserta un nuevo documento en la colección ventas.
- Consulta `aggregate` con `$group`: Calcula el total de ventas por proveedor.
- Consulta `updateOne` con `$set`: Actualiza un campo específico de un documento.
- Consulta `insertOne` para proveedores: Añade un nuevo proveedor a la colección.

Diseño de Base de Datos

Consideraciones de Diseño

- Claves Primarias: `proveedor_id` en proveedores, `cliente_id` en clientes, `producto` en productos, y `venta_id` en ventas son únicas para cada documento.
- Definición de Campos: Los campos son definidos para capturar la información esencial sobre productos, proveedores, clientes y ventas, permitiendo consultas y actualizaciones eficientes.

Análisis y Discusión

Interpretación de Resultados

Los resultados obtenidos de las consultas reflejan un manejo adecuado de la información en la base de datos. La adición de nuevas tallas, ventas y proveedores muestra la flexibilidad de MongoDB para manejar cambios y actualizaciones en los datos. La consulta de ventas por proveedor permite evaluar el rendimiento de los proveedores y tomar decisiones informadas sobre futuras compras o negociaciones.

- **Recuperación y Gestión de Datos:** La consulta `find({})` demostró la capacidad de MongoDB para devolver grandes volúmenes de información de manera eficiente. Esta funcionalidad es fundamental para la visualización general del inventario y permite a los administradores de la base de datos obtener rápidamente un estado actualizado de todos los productos disponibles.
- **Actualización Dinámica de Documentos:** Las consultas `updateOne` y `$push` demostraron la flexibilidad para modificar documentos existentes. La posibilidad de agregar tallas adicionales a productos ya registrados, sin necesidad de modificar el esquema predefinido, destaca la ventaja de MongoDB frente a bases de datos relacionales, permitiendo una gestión más ágil y adaptable a los cambios del inventario.
- **Inserción de Nuevos Documentos:** La consulta `insertOne` para las ventas y proveedores muestra la capacidad de MongoDB para incorporar nuevos registros en tiempo real, lo que permite a la tienda registrar de forma inmediata nuevas transacciones y proveedores. Esto es crucial para mantener una base de datos actualizada que refleje el comportamiento del negocio al instante.
- **Análisis de Rendimiento por Proveedor:** La consulta agregada con `$group` para calcular el total de ventas por proveedor ofrece un poderoso análisis para la toma de decisiones. Al obtener esta información, el negocio puede identificar fácilmente cuáles proveedores generan más ingresos y establecer estrategias para fortalecer o

renegociar relaciones comerciales. Esta funcionalidad no solo mejora la toma de decisiones, sino que también optimiza la gestión del inventario y las finanzas.

- **Actualización de Contactos y Proveedores:** Las actualizaciones a los proveedores (como cambiar direcciones y contactos) evidencian la capacidad de la base de datos para manejar datos empresariales que cambian frecuentemente, como la información de contacto de proveedores. Esto asegura que los datos se mantengan precisos, lo que es esencial para operaciones eficientes y relaciones comerciales fluidas.
- **Consultas Complejas:** La combinación de consultas de agregación y modificación de documentos realza la capacidad de MongoDB para manejar tanto consultas simples como más complejas, proporcionando una visión detallada de la operación empresarial. Las consultas como el cálculo de ventas acumuladas por proveedor permiten realizar análisis estratégicos que contribuyen a mejorar la eficiencia operativa y la toma de decisiones informada.

Conclusiones

- **Eficiencia en Consultas:** MongoDB permite ejecutar consultas complejas de manera eficiente, proporcionando datos cruciales para la toma de decisiones.
- **Flexibilidad:** La capacidad de actualizar documentos y agregar nuevos registros facilita el mantenimiento de la base de datos.
- **Diseño Eficiente:** El diseño de la base de datos es adecuado para el tipo de datos manejados y permite una gestión eficiente de la información.

Recomendaciones

- **Revisión Periódica:** Realizar revisiones periódicas del esquema de la base de datos para asegurar que se ajusta a las necesidades cambiantes.
- **Optimización de Consultas:** Considerar la creación de índices para mejorar el rendimiento de las consultas más frecuentes.
- **Documentación Actualizada:** Mantener una documentación actualizada sobre la estructura y el contenido de la base de datos para facilitar el mantenimiento y las actualizaciones.

Referencias

Chodorow, K. (2019). MongoDB: The Definitive Guide: Powerful and Scalable Data Storage (3rd ed.). O'Reilly Media.

MongoDB Inc. (2023). MongoDB Database Tools. Recuperado de <https://www.mongodb.com/docs/database-tools/>

Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management (6th ed.). Pearson.

Welling, L., & Thomson, L. (2017). PHP and MySQL Web Development (5th ed.). Addison-Wesley.

Fowler, M. (2002). Patterns of Enterprise Application Architecture. Addison-Wesley.

Codd, E. F. (1970). A Relational Model of Data for Large Shared Data Banks. Communications of the ACM, 13(6), 377–387.
<https://doi.org/10.1145/362384.362685>

García-Molina, H., Ullman, J. D., & Widom, J. (2009). Database Systems: The Complete Book (2nd ed.). Pearson.