AQUÍ NOMBRE DE LA ACTIVIDAD

**Presentado a:** Instructor Cesar Chacon

**Por Aprendiz:** Juan Daniel Madroñero

**Aprendices:** XXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXX

**Ficha:** 2770242

**Competencia:** Bases Datos

**Resultado de Aprendizaje:**

**Tecnólogo en Análisis y Desarrollo de Software**

**Servicio Nacional de Aprendizaje SENA**

**Centro de Teleinformática y Producción Industrial**

**Regional Cauca**

Popayán, día 26 de mes abril del año 2024

**Tabla de Contenido**

[**1.** **Actividad 1** 3](#_Toc165024456)

[**2.** **Actividad 2** 4](#_Toc165024457)

[**3.** **Actividad 3** 5](#_Toc165024458)

[**4.** **Bibliografía** 6](#_Toc165024459)

# **Actividad 1**

Crear una base de datos llamada INSTITUTO

a. Crear una colección llamada ALUMNOS con los siguientes atributos y reglas.

ATRIBUTOS

• identificacion: de tipo string, longitud mínima 5, longitud máxima 15

• nombres: de tipo string

• apellidos: de tipo string

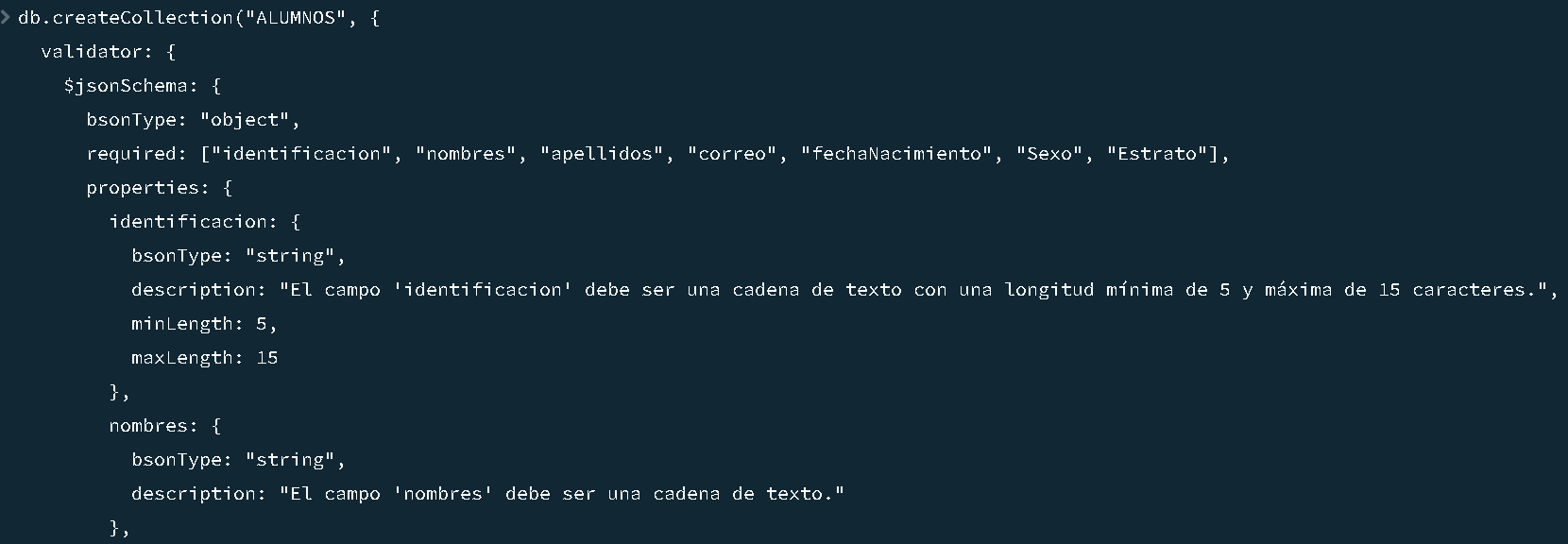
• correo: de tipo string y validar que tenga texto@texto.texto, o sea que sea una dirección

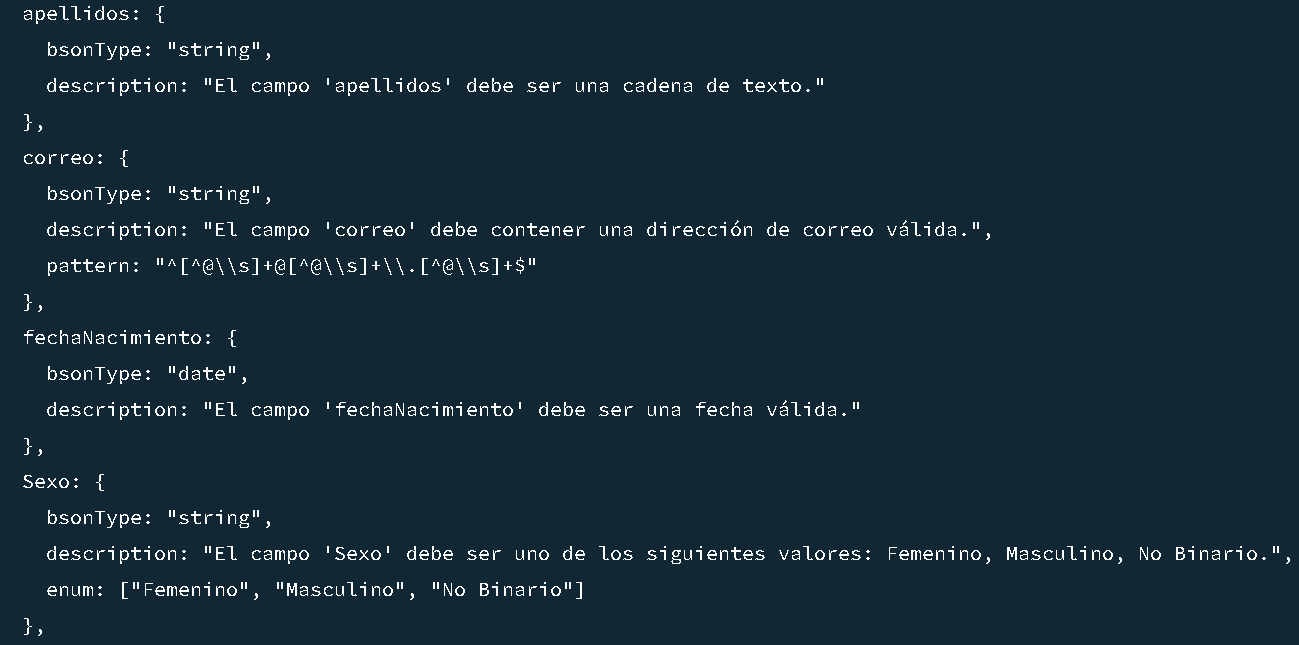
de correo valida (buscar cómo)

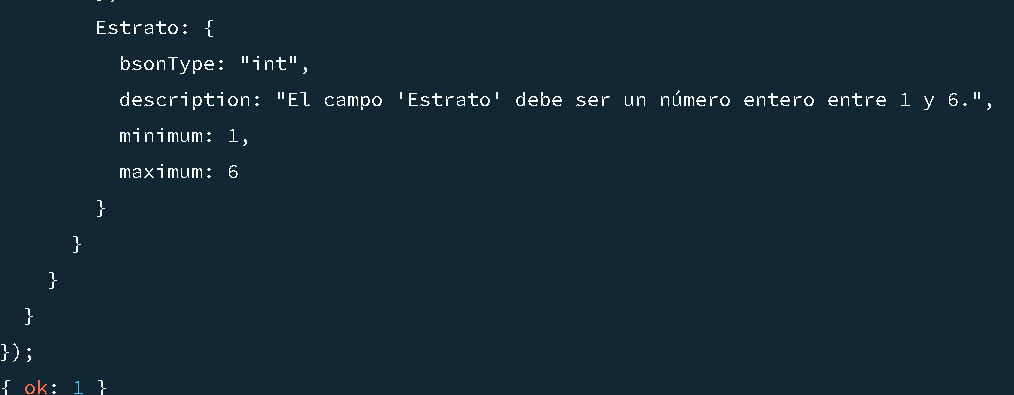
• fechaNacimiento: de tipo date

• Sexo: de tipo string, con las siguientes posibilidades: Femenino, Masculino, No Binario.

• Estrato: de tipo int, mínimo valor 1, máximo valor 6







# **Actividad 2**

En la base de datos TIENDAADSO creada en el material de apoyo crear la siguiente colección.

Sino ha creado la base de datos, crearla.

a. Crear una colección llamada PEDIDOS con los siguientes atributos y reglas.

ATRIBUTOS

• codigo: de tipo int

• cliente: de tipo string (nombre completo del cliente.

• direcciónCliente: tipo string

• fechaPedido: de tipo date

• fechaEntrega: de tipo date

• telefonoCliente: tipo string, longitud mínima 7 máxima 12

GFPI-F-135 V02

• correoCliente: de tipo string, validando que el formato sea una cuenta de correo.

• FormaEnvío: de tipo String, con las siguientes posibilidades: Aérea, Maritima o

Terrestre.

• cantidadProductos: de tipo int, mínimo 1

• estadoPedido: de tipo string, con las siguientes posibilidades: Enviado, Entregado,

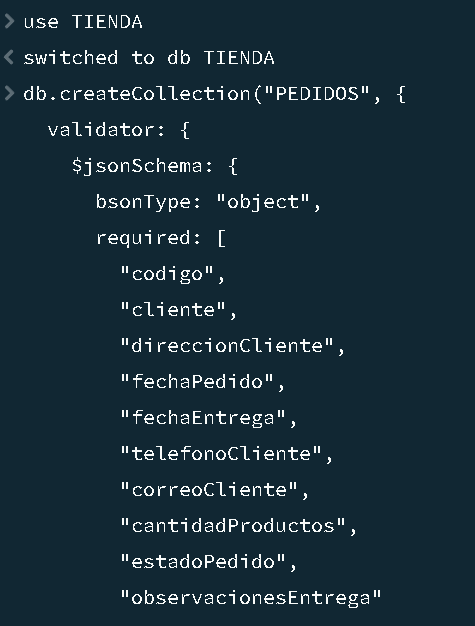
NoEntregado

• observacionesEntrega: de tipo string. Aquí se registra algunas observaciones de la

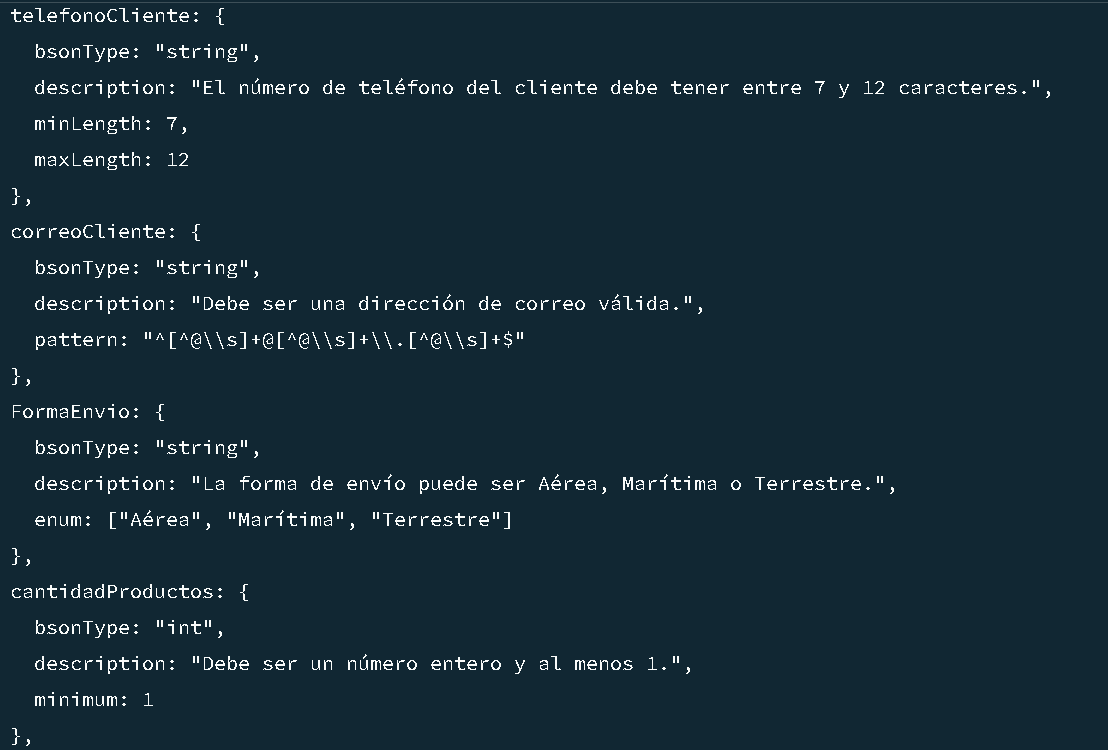
entrega o si no es posible entregarlo

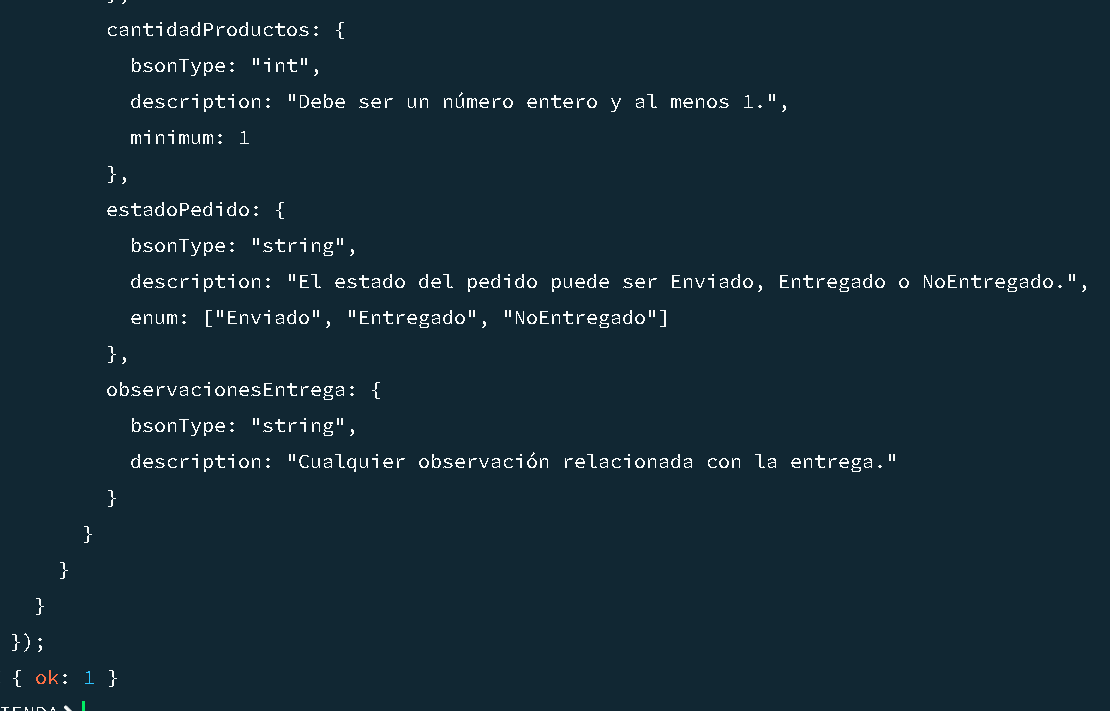
b. Todos los atributos son obligatorios, excepto la forma de envío.

c. A todos los atributos colocarles una descripción.





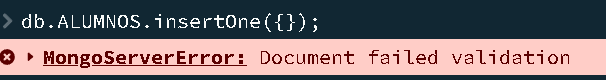




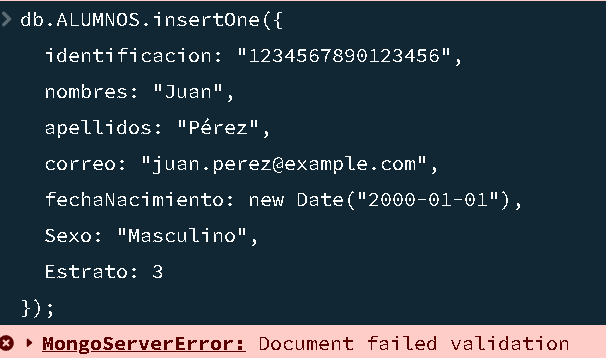
# **Actividad 3**

Realizar las siguientes actividades en las colecciones para validar las reglas:

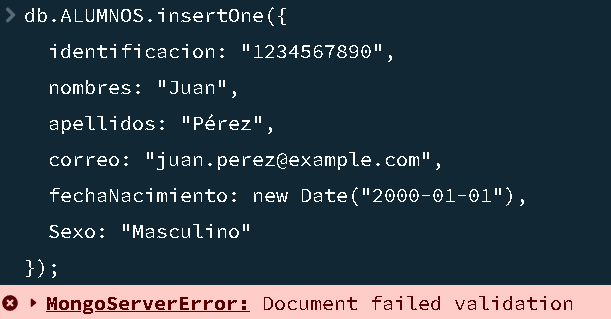
a. Intente insertar un alumno sin datos



b. Intente insertar un alumno que no cumpla con la longitud de la identificación



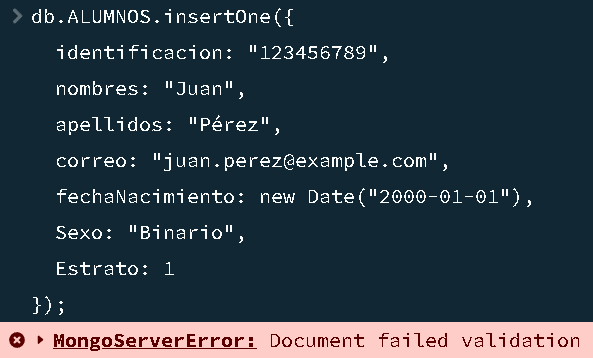
c. Intente insertar un alumno que le falte el campo estrato

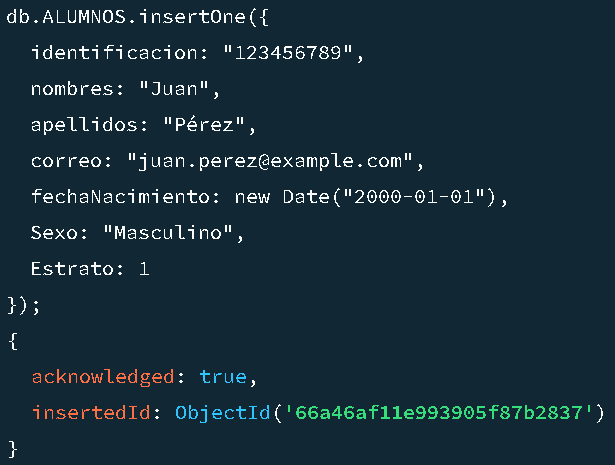


d. Intente insertar un alumno con estrato 7



e. Intente insertar un alumno con sexo Binario

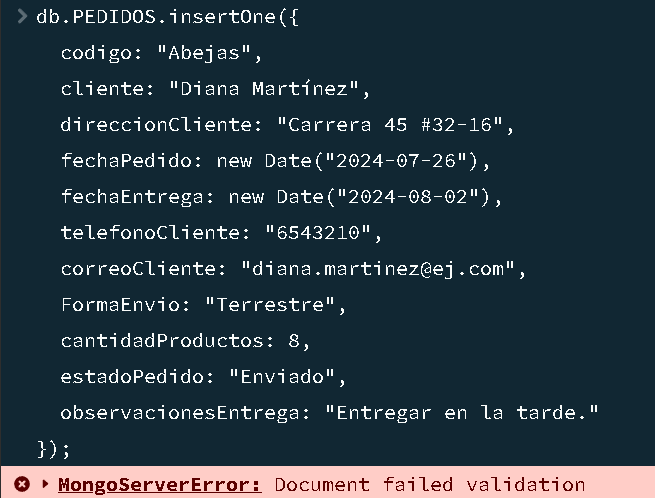


f. Inserte dos alumno que cumplan con todas las reglas. 

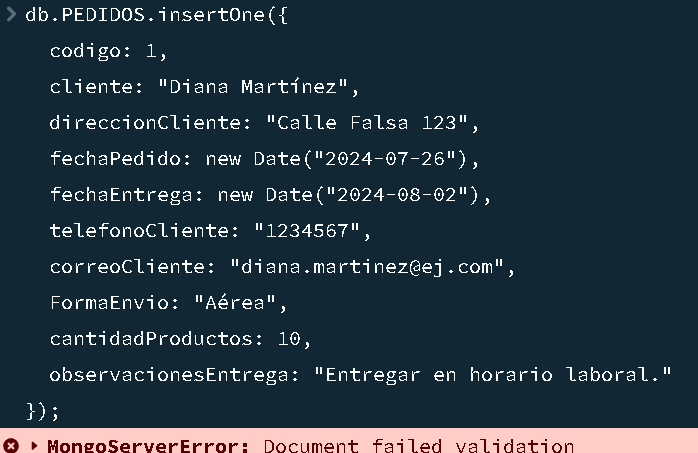
g. Insertar un pedido sin datos



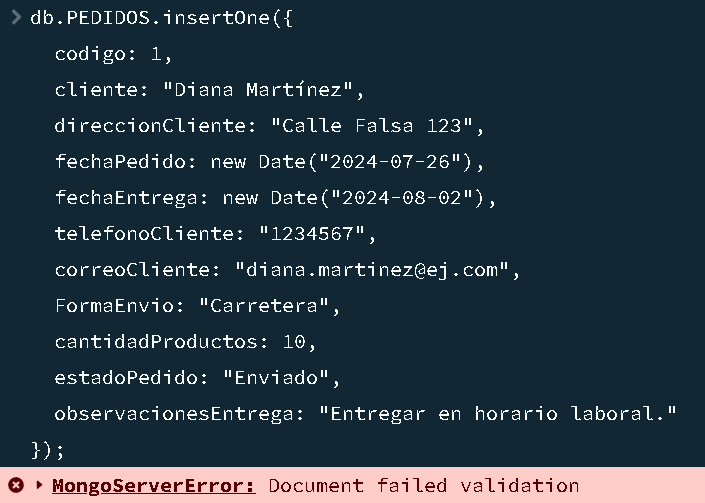
h. Insertar un pedido con el código en formato string



i. Insertar un pedido donde le falte el campo de estadoPedido



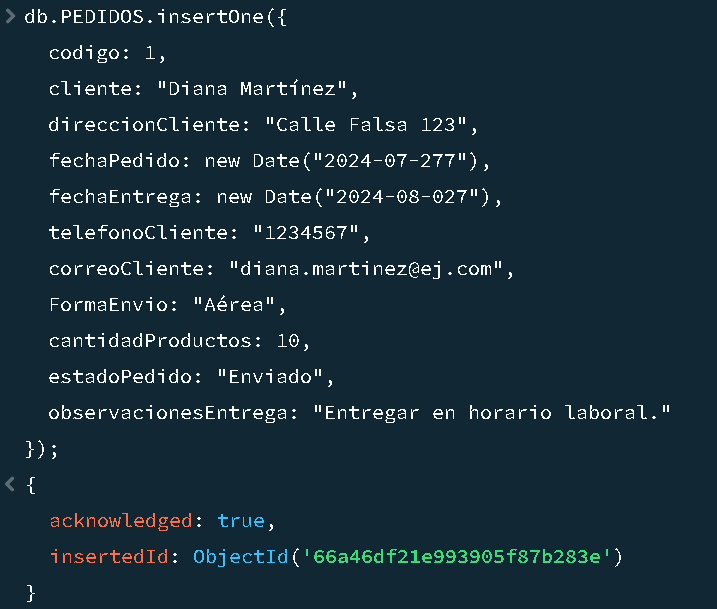
j. Insertar un pedido con la forma de Envío “Carretera”



k. Insertar un pedido con el formato de fecha incorrecto

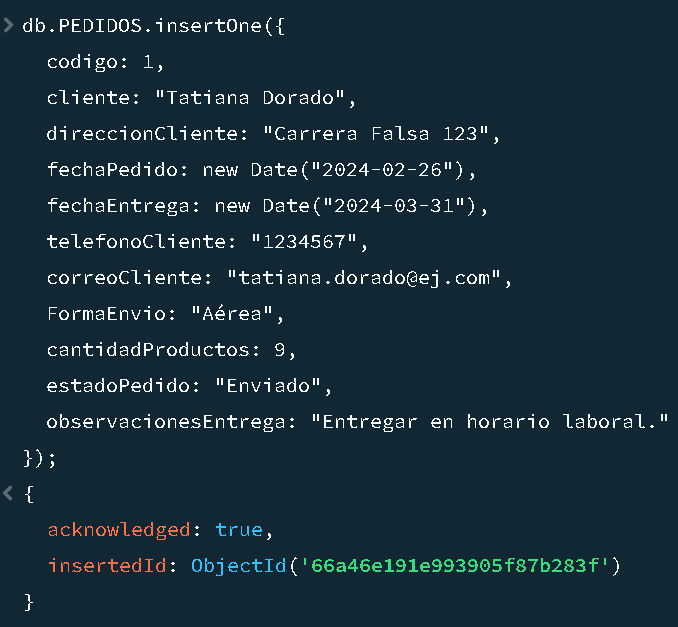


l. Insertar dos pedidos que cumplan con las reglas

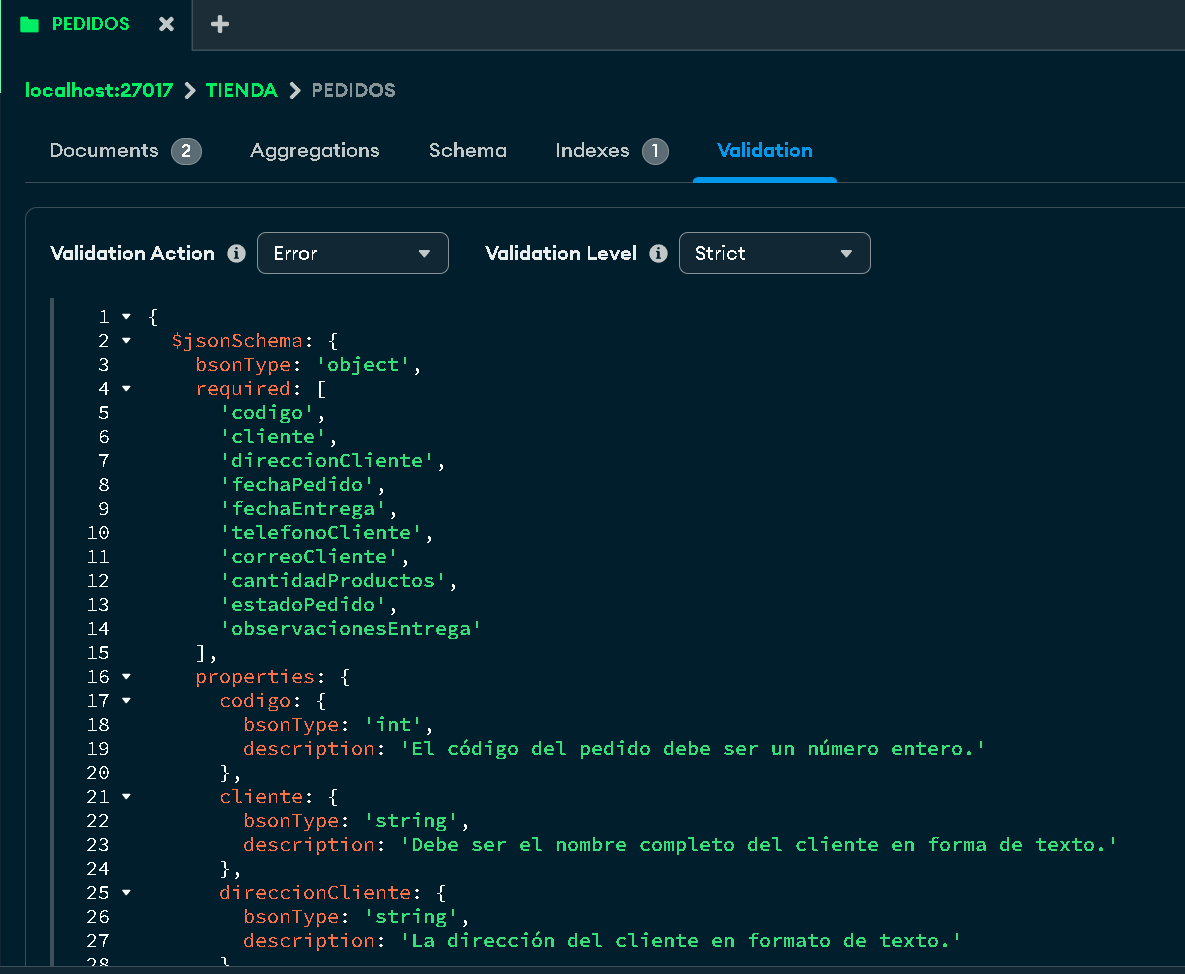


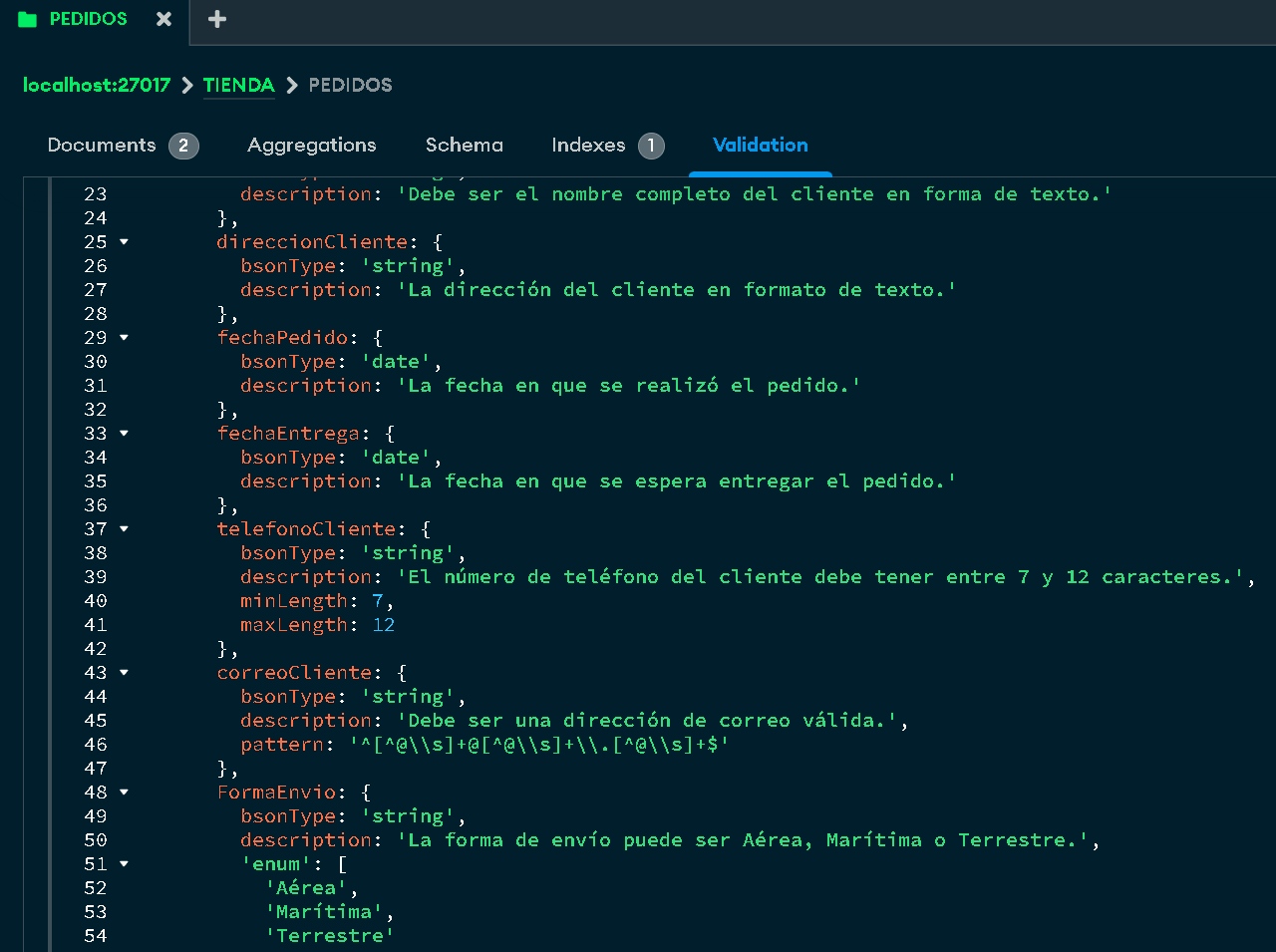
m. Ejecutar el comando que permite ver las reglas de validación de las colecciones

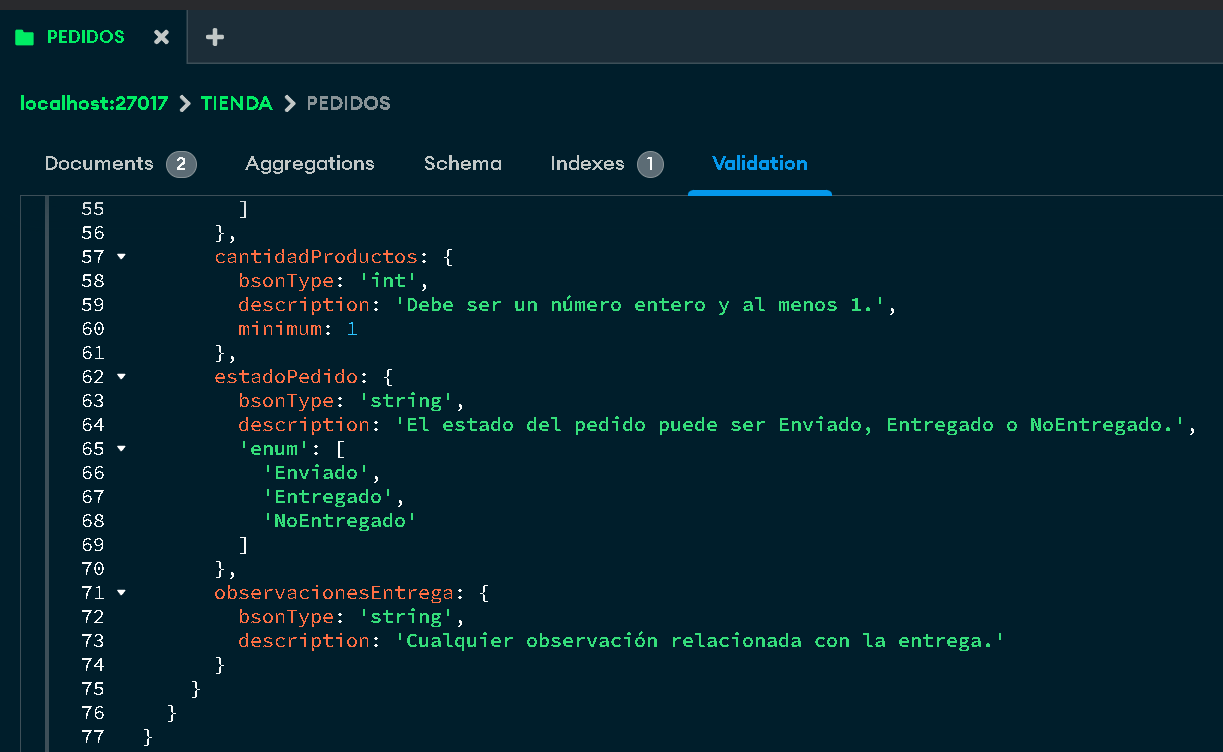
creadas.



n. Imprima un pantallazo desde la interfaz visual de MongoDBCompass donde se puedan

ver las reglas de validación 

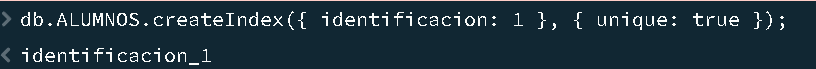




# **Actividad 4**

Buscar como puede crear campos únicos en MongoDB y realizar las siguientes actividades:

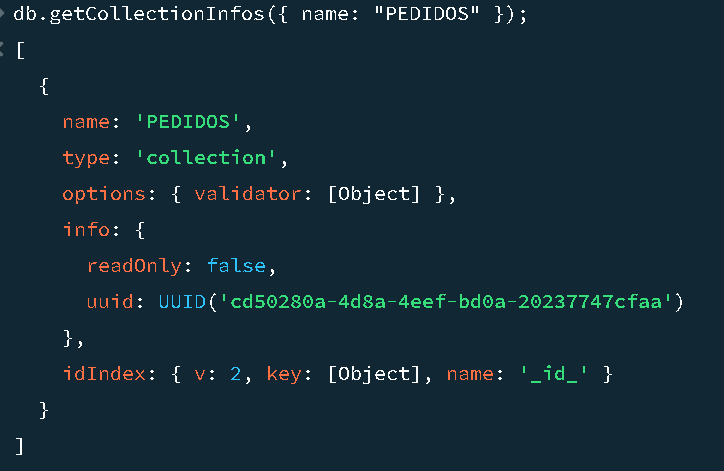
a. En la colección ALUMNOS hacer que el atributo identificación quede como único.



b. En la colección PEDIDOS, hacer que el atributo código quede como único.

c. Inserte un alumno con una identificación de alumno ya existente para validar si le permite

insertarlo.

d. Inserte un Pedido con un código de pedido existente para validar si permite insertarlo.

# **Bibliografía**

Relacionar aquí la bibliografía utilizada.

* Piñeiro Gómez, José Manuel. Diseño de bases de datos relacionales. Editorial Paraninfo, 2014. 164 páginas.
* Bases de datos, sitio web: <https://josejuansanchez.org/bd/>