

Bases de Datos 2023

Práctico IX: Modelado de Datos en MongoDB

Juan Cabral, Tadeo Cocucci,
Ramiro Demasi, Guido Ivetta

Parte I

Continuando con la base de datos mflix:

Agregar las siguientes reglas de validación usando JSON Schema. Luego de cada especificación testear que efectivamente las reglas de validación funcionen, intentando insertar 5 documentos válidos y 5 inválidos (por distintos motivos).

1. Especificar en la colección *users* las siguientes reglas de validación: El campo *name* (requerido) debe ser un string con un máximo de 30 caracteres, *email* (requerido) debe ser un string que matchee con la [expresión regular](#): `"^(.*)@(.*)\\.({2,4})$"`, *password* (requerido) debe ser un string con al menos 50 caracteres.
2. Obtener metadata de la colección *users* que garantice que las reglas de validación fueron correctamente aplicadas.
3. Especificar en la colección *theaters* las siguientes reglas de validación: El campo *theaterId* (requerido) debe ser un int y *location* (requerido) debe ser un object con:
 - a. un campo *address* (requerido) que sea un object con campos *street1*, *city*, *state* y *zipcode* todos de tipo string y requeridos
 - b. un campo *geo* (no requerido) que sea un object con un campo *type*, con valores posibles *"Point"* o *null* y *coordinates* que debe ser una lista de 2 doubles

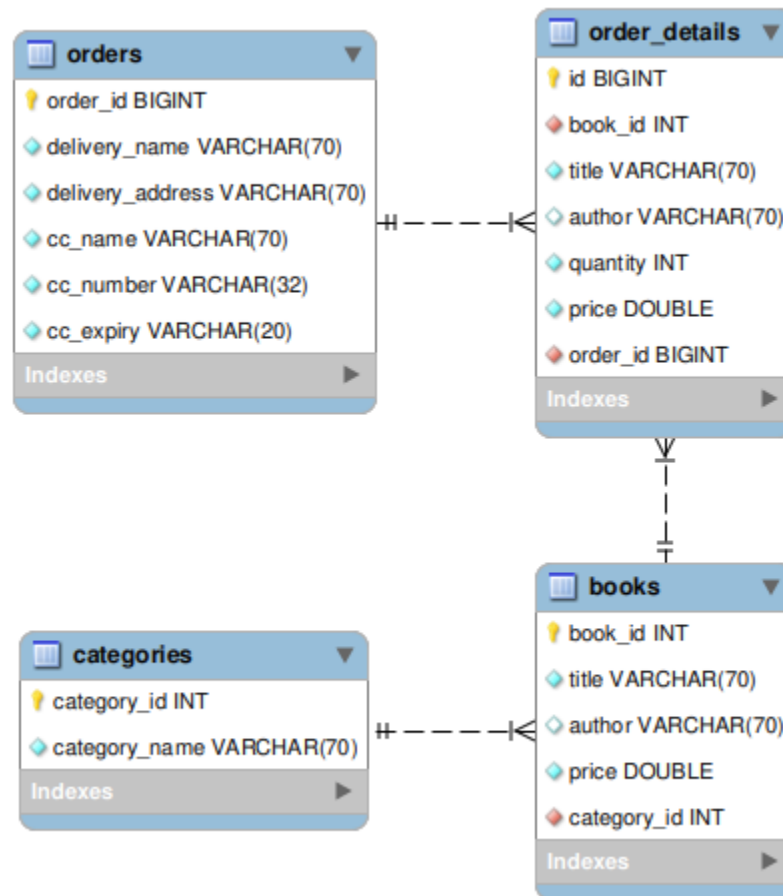
Por último, estas reglas de validación no deben prohibir la inserción o actualización de documentos que no las cumplan sino que solamente deben advertir.

4. Especificar en la colección *movies* las siguientes reglas de validación: El campo *title* (requerido) es de tipo string, *year* (requerido) int con mínimo en 1900 y máximo en 3000, y que tanto *cast*, *directors*, *countries*, como *genres* sean arrays de strings sin duplicados.
 - a. Hint: Usar el constructor `NumberInt()` para especificar valores enteros a la hora de insertar documentos. Recordar que mongo shell es un intérprete javascript y en javascript los literales numéricos son de tipo `Number` (double).
5. Crear una colección *userProfiles* con las siguientes reglas de validación: Tenga un campo *user_id* (requerido) de tipo `"objectId"`, un campo *language* (requerido) con alguno de los siguientes valores [*"English"*, *"Spanish"*, *"Portuguese"*] y un campo *favorite_genres* (no requerido) que sea un array de strings sin duplicados.

Modelado de datos en MongoDB

- Identificar los distintos tipos de relaciones (One-To-One, One-To-Many) en las colecciones *movies* y *comments*. Determinar si se usó documentos anidados o referencias en cada relación y justificar la razón.
- Dado el diagrama de la base de datos **shop** junto con las queries más importantes.

Diagrama



Queries

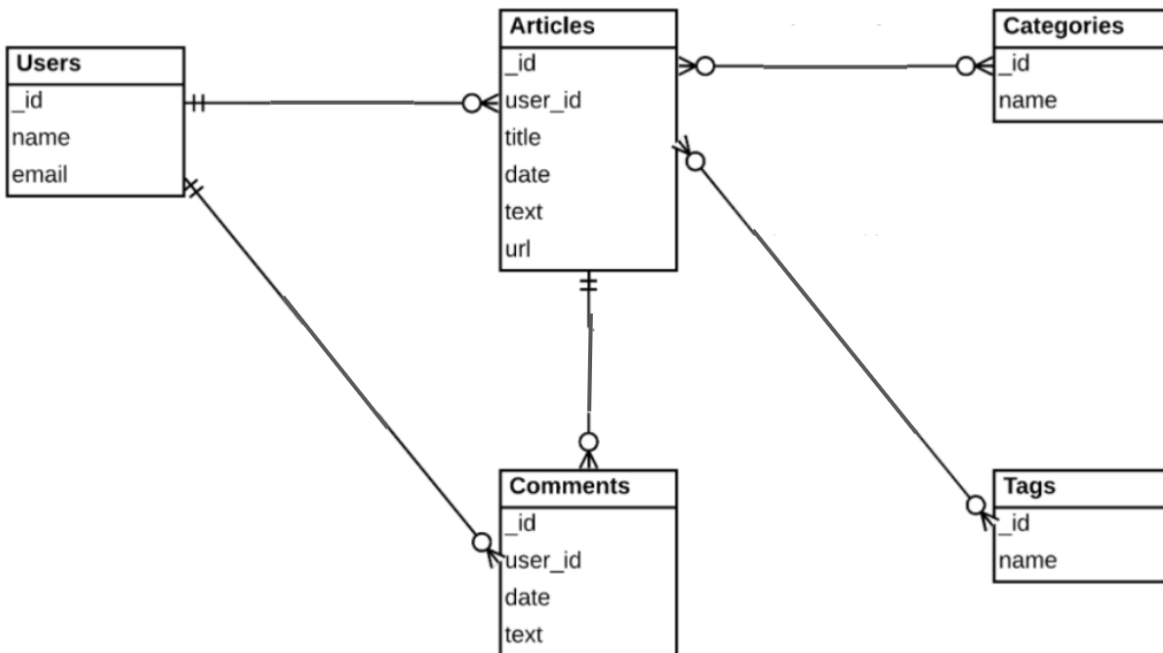
- Listar el id, título, y precio de los libros y sus categorías de un autor en particular
- Cantidad de libros por categorías
- Listar el nombre y dirección entrega y el monto total ($\text{quantity} * \text{price}$) de sus pedidos para un `order_id` dado.

Debe crear el modelo de datos en mongodb aplicando las estrategias "Modelo de datos anidados" y Referencias. El modelo de datos debe permitir responder las queries de manera eficiente.

Inserte algunos documentos para las colecciones del modelo de datos. Opcionalmente puede especificar una regla de validación de esquemas para las colecciones.

Se provee el archivo shop.tar.gz que contiene algunos datos que puede usar como ejemplo para los inserts en mongodb.

8. Dado el siguiente diagrama que representa los datos de un blog de artículos



Se pide

- Crear 3 modelos de datos distintos en mongodb aplicando solo las estrategias “Modelo de datos anidados” y Referencias (es decir, sin considerar queries).
- Crear un modelo de datos en mongodb aplicando las estrategias “Modelo de datos anidados” y Referencias y considerando las siguientes queries.
 - Listar título y url, tags y categorías de los artículos dado un user_id
 - Listar título, url y comentarios que se realizaron en un rango de fechas.
 - Listar nombre y email dado un id de usuario

Inserte algunos documentos para las colecciones del modelo de datos. Opcionalmente puede especificar una regla de validación de esquemas para las colecciones..