We want to create a MongoDB database for a news magazine. If we think in the same way that we did using relational model a database for that may have 3 entities:

● Users. Fields:

○ \_id

○ nickname

○ dateOfBirth

○ email

● Articles. Fields:

○ \_id

○ title

○ body

○ Tags ([ ])

○ author

● Comments. Fields:

○ \_id

○ text

○ author

A user may create articles (an article belongs to a single user). Users may also add comments to articles (a comment belongs to a single article and it has a single author).

If we think in the same way that we did using relational model, this database will have 3 tables. But with MongoDB it’s a good idea to insert comments inside their article (because comments belong to a single article). The reason is that, usually, when you get an article you want its comments also… With MongoDB you think nearer to the implementation. You’ll understand this next year :)

So, create using MongoDB a database called “magazine” with two collections: "users" and "articles". Remember that the comments are a field inside an "article" document…

Finally, join all the information of the users joined with their articles (use $lookup).

use magazine

db.users.insertOne({

\_id: (5sfg1sdfhtr65h13)

nickname: “Alicia Tugores”

dateOfbirth: new Date “02-10-1998”

email: “[aliciatugores1098@gmail.com](mailto:aliciatugores1098@gmail.com)”

})

db.articles.insertOne({

\_id: (3sgbj64g8dsfg56)

tittle: Érase una vez

body: “Éste es el cuerpo”

tags:[“#estoesliteratura”,”pasionporelarte”]

autor:”Alicia Tugores”

})

db.comments.insertOne({

\_id: (3sgbj64g8dsfg56)

text: “no sé que estoy haciendo”

autor: “Alicia Tugores”

})

db.users.agreggate({$lookup:

{

From:”articles”

LocalField:”nickname”

ForeignField:”autor”

as: esto es una prueba

}})