

Práctica 2 – Arrays y cosas parecidas - Solución

1. Nos movemos al área de trabajo 2 (ctrl+Alt+ →)
2. Abrir un terminal (arriba a la derecha; al lado del símbolo de Chrome; si ya hay un terminal y queremos otro, dar botón derecho sobre el terminal y elegir “Ventana Nueva”)
3. Crear un directorio donde se grabarán los datos: `rm -rf datos; mkdir datos`
4. Iniciamos el servidor tecleando `mongod --dbpath datos`
5. Volvemos al área de trabajo 1 ((ctrl+Alt+ ←)
6. Ahora vamos a importar los datos que usaremos en la práctica:
 1. Bajar del [campus](#) el fichero
 2. Abrimos un nuevo terminal y escribimos

```
mongoimport --drop --db test --collection libros --file /home/hlocal/Descargas/libros.json
```

7. En el mismo terminal tecleamos `mongo` sin más...ya estamos en el shell de mongo. Podemos teclear `show collections` y debemos ver que la colección `libros` está. Y escribir `db.libros.findOne()` para mirar un poco la estructura de la colección.

- 1) Libros sin *opiniones* y con *precio* mayor de 12 euros (mostrar solo título y precio)

Solución:

```
db.libros.find({opiniones:{$exists:false}, precio:{$gt:12}}, {_id:0, titulo:1, precio:1})
```

- 2) Número de libros cuya clave *precio* es de tipo string

Solución

```
db.libros.find({precio:{$type:"string"}}).count()
```

- 3) Títulos de libros sobre las que no haya opinado Bertoldo

Solución

```
db.libros.find({"opiniones.usuario":{$ne:"bertoldo"}},{titulo:1,_id:0})  
db.libros.find({"opiniones.usuario":{$nin:["bertoldo"]},{titulo:1,_id:0})
```

- 4) Títulos de libros que sean de los géneros *romance* y *terror* a la vez

Solución

```
db.libros.find({"genero":{$all:["romance","terror"]},{titulo:1,_id:0})  
db.libros.find({$and:[{"genero":"romance"}, {"genero":"terror"}]},{titulo:1,_id:0})
```

- 5) Número de libros que sean del género *romance* pero no del género *terror*.

Solución

```
db.libros.find({$and:[{"genero":"romance"}, {"genero:{$ne:"terror"}}],{}).count()  
db.libros.find({"genero":{$in:["romance"],$nin:["terror"]},{_id:0, "titulo":1})  
db.libros.find({"genero":{$eq:"romance",$neq:"terror"}},{_id:0, "titulo":1})
```

- 6) Número de libros para los que la opinión (posición 0) tenga más de 5 *puntos*

Solución

```
db.libros.find({"opiniones.0.puntos":{$gt:5}}).count()
```

- 7) Número de libros que, o bien cuentan con al menos una opinión de herminia, o cuestan menos de 10 euros.

Solución

```
db.libros.find({$or:[{"opiniones.usuario":"herminia"}, {"precio:{$lt:10}]}).count()
```

- 8) Libros que tengan al menos 3 opiniones, y para cada uno de ellos mostrar solo el título y la 3ª opinión.

Solución

```
db.libros.find({"opiniones.2":{"$exists:true"}},{"_id":0,"titulo":1,"opiniones":{"$slice:[2,1]}}).pretty()
```

9) Para cada película de género romance, mostrar:

- El título
- La primera opinión de bertoldo para esa película en la que haya puesto una puntuación igual o superior a 8

Solución

```
db.libros.find({genero:"romance"},  
               {titulo:1, _id:0, opiniones:{$elemMatch:{usuario:"bertoldo",puntos:{$gte:8}}}})
```

10) Mostrar los títulos de libros que entre su última edición y la primera hayan pasado al menos 10 años

Solución

```
db.libros.find({$expr: {$gte: [{$subtract: ["$edicion.actual.year", "$edicion.primer"], 10]}},  
               {_id:0, titulo:1})
```