

*** ===== - Apartado 2.- =====

**** a) De cada Tema:

Consulta qué temas tengo:

```
db.aficiones.distinct("Tema")
```

**** 1. Obtener los componentes mejor valorados (puntuaciones >= 9)

--- Saco los nombres:

```
db.aficiones.find({"Puntuacion" :{$gte: 9}}, {"Nombre":1, _id:0});
```

.

**** 2. Calcula cuanto te gastarías si vas a todos los temas mejor valorados.

----- da el nombre del Tema, la suma y cuántos

Para cada Tema

```
db.aficiones.aggregate([
  { $match: { "Puntuacion":{$gte:9} } },
  { $group: { _id: "$Tema",
    total: { $sum: "$Precio" },
    count: { $sum: 1 } } },
  { $sort: { _id:1 } } //clasifico por _id ascendente
    // (no pedido())
])
```

----- da solo un total

```
db.aficiones.aggregate([
  { $match: {"Puntuacion": {$gt:9}} },
  { $group: { _id: null, PrecioTotal : { $sum : "$Precio" } } }
])
```

.

**** 3. Obtener los componentes valorados para cada uno de los valores (10,9,8,7,6 y 5) por separado

```
db.aficiones.aggregate(
  { $match : { $or : [ { Puntuacion : { $eq : 10 } },
    { Puntuacion : { $eq : 9 } },
    { Puntuacion : { $eq : 8 } },
    { Puntuacion : { $eq : 7 } },
    { Puntuacion : { $eq : 6 } } ] } },
  { $group : { _id : "$Puntuacion", componente : { $push : "$Tema" } } }
  )
```

.

**** 4. Lista de Apodos con esa afición

```
{ $group : { _id : "$Tema",
  Apodos: { $addToSet : "$Apodo" }
}
```

- **** b) (NO ENTRA en EXAMEN) Lista los Apodos, Componente y Tema en los que coincide al menos un componente (el nombre) del mismo Tema.
- **** c) (NO ENTRA en EXAMEN) Repite la búsqueda anterior para puntuaciones intermedias: más de 4 y menos de 9.
- **** d) Describe al menos cuatro consultas interesantes para ti, escribe el código y ejecútalas:

--> las que quieras

- **** e) Obtener todos los nombres de componentes de tu colección clasificados por tema. Para imprimirlos de una manera ordenada, usa un cursor que llama a una función sin nombre (definida dentro del cursor). Esa función imprime cada ítem que, previamente hemos transformado a json (así le damos un formato más legible)

```
db.aficiones.aggregate(
  { $group: {
    _id: { tema: "$Tema" },
    count: { $sum: 1 },
    docs: { $push: "$Nombre" }
  } }).forEach( function(myDoc) { print(myDoc);});
```

- **** f) Rebaja un 10% al precio de todos los componentes peor valorados (puntuación < 7). Y en la misma actualización añades el atributo Descuento a todas las aficiones que según la puntuación, si es buena de menos % de descuento.

```
db.aficiones.find().forEach( function (myDoc) {
  var descuento = myDoc.Precio * 0.10;
  var porcentaje = (myDoc.Puntuacion/10) * 10 ;

  if (myDoc.Puntuacion < 7 ){
    myDoc.Precio = myDoc.Precio - descuento;
  }
  myDoc.Descuento = porcentaje;
  db.Aficiones.save(myDoc);
});
```

- **** g) (NO ENTRA en EXAMEN) Crea una colección PorNivel donde vas a crear cuatro niveles de calidad

**** ===== - Apartado 3.- CAPPED =====

Queremos mantener en una colección superGuai, los 5 mejores componentes de la colección Aficiones. Para ello hacemos lo siguiente:

.

**** a) Crear dicha colección

```
db.createCollection("superGuai",{capped:true, size: 100000,max:5, autoIndexId:true});
```

.

**** b) (NO ENTRA en EXAMEN)Crea las operaciones necesarias para poner los 5 mejores elementos de acuerdo al criterio de calidad del apartado

**** c) (NO ENTRA en EXAMEN)Inserta un elemento a mano.

**** d) (NO ENTRA en EXAMEN)Lista todos los componentes para comprobar que mantiene los último cinco introducidos

**** ===== - Apartado 4.- EXTRA des/normalizar =====

- DECISION: Separan en diferentes colecciones o integrar en una sola colección ??

- Ver las transparencias de la teoría

.
