

## Agregación

1. Como en la práctica anterior, importamos los datos con

```
$ mongoimport --drop --collection presupuesto --file presupuesto.json
```

Deben importarse 4.931 registros.

2. Ejecutamos mongosh y ya estamos en el shell de mongo. Podemos teclear `show collections` y debemos ver que la colección presupuesto está. Y escribir `db.presupuesto.findOne()` para revisar la estructura de la colección.

El conjunto de datos se corresponde con el presupuesto de gastos de una comunidad autónoma en el año 2024. Cada documento representa una partida presupuestaria, la unidad mínima en la que se divide el presupuesto. Escribe consultas de agregación para obtener la siguiente información:

1. La media de las cuantías iniciales (`cuantia.inicial`) de todas las partidas presupuestarias. Devuelve el resultado como un simple número.

2. Las cuantías iniciales y actuales acumuladas por cada `centro_presupuestario` con el formato

```
{ _id: 'AG.VIVIENDA SOCIAL', inicial: 182012480, actual: 182008477.35 }  
{ _id: 'CÁMARA DE CUENTAS', inicial: 9521166, actual: 9594031.26 } /* etc. */
```

3. La misma consulta pero agrupando por capítulo de gasto. El capítulo de gasto es la decena de millar de la clave `id` del subconcepto (1 significa gastos de personal, 2 gastos corrientes, etc.). El formato de cada documento de la agrupación debe ser

```
{ _id: 3, inicial: 961978141, actual: 853197376.88 }
```

y ordénalos por número de capítulo creciente.

4. Otra vez la misma consulta, pero agrupando por ambos conceptos a la vez (es decir, queremos saber el gasto de cada centro presupuestario en cada capítulo).

5. La cuantía actual máxima de entre las partidas en cuyo subconcepto aparezca la subcadena COMPLUTENSE. Devuelve el resultado como un simple número.

6. (1.5 puntos) El presupuesto ejecutado (`ejecucion.dispuesto`) cada mes del año entre todas las partidas. Muestra la salida ordenada por gasto decreciente donde cada documento tiene el formato

```
{ _id: 6, gasto: 1248916534.75 }
```

No aparecerá el mes de diciembre porque los datos no están disponibles.

7. (2 puntos) Queremos hacer recuento del número de partidas en intervalos disjuntos según su presupuesto inicial. El primer intervalo es `[0, 10.000)` y el resto terminan sucesivamente en un millón, 10 millones, mil millones y el último no está acotado. La salida esperada es:

```
{ _id: 0, numero: 1403 }  
{ _id: 10000, numero: 2771 }  
{ _id: 1000000, numero: 580 }  
{ _id: 100000000, numero: 175 }  
{ _id: '>10^9', numero: 2 }
```

**8.** (1.5 puntos) Los cinco programas con menor gasto autorizado (`cuantia.autorizado`) sobre la cuantía inicial en términos porcentuales. Cada objeto del resultado tendrá la forma

```
{ programa: { id: '929N', nombre: 'GESTIÓN CENTRALIZADA' }, gastado: 8.96 }
```

y han de aparecer ordenados por porcentaje creciente, que estará redondeado a dos cifras decimales.