# Ejercicios de Aggregation Framework

## Ejercicio 1: Contar productos por categoría

Enunciado: Obtén el número total de productos disponibles en cada categoría.

db.ventas.aggregate([

{$group:{\_id: "$categoria", totalProductos: {$sum:"$cantidad"}}}

])

## Ejercicio 2: Productos con precio mayor a 1000

Enunciado: Lista los productos que tienen un precio mayor a 1000.

db.ventas.aggregate([{$match:{precio:{$gt:1000}}}])

## Ejercicio 3: Calcular el valor total de cada producto en stock

Enunciado: Calcula el valor total de cada producto en stock (precio \* cantidad) y

muéstralo en un nuevo campo llamado valor\_total.

db.ventas.aggregate([

{$addFields:{valor\_total:{$multiply:["$precio", "$cantidad"]}}}

])

## Ejercicio 4: Obtener los 3 productos más vendidos

Enunciado: Muestra los 3 productos con mayor cantidad vendida en orden

descendente.

db.ventas.aggregate([{$sort:{cantidad:-1}},{$limit:3}])

## Ejercicio 5: Ingresos totales por categoría

Enunciado: Calcula los ingresos totales (precio \* cantidad) agrupados por categoría.

db.ventas.aggregate([

{$group:{\_id: "$categoria", ingresos\_totales:{$sum:{$multiply:["$precio","$cantidad"]}}}}

])

## Ejercicio 6: Calcular el promedio de precios por categoría

Enunciado: Calcula el precio promedio de los productos agrupados por categoría.

db.ventas.aggregate([

{$group:{\_id: "$categoria", precio\_medio:{$avg:"$precio"}}}

])

## 

## Ejercicio 7: Identificar la categoría con mayores ventas totales

Enunciado: Determina cuál es la categoría que ha generado mayores ingresos totales.

db.ventas.aggregate([

{$group:{\_id: "$categoria",

ingresos\_totales:{$sum:{$multiply:["$precio", "$cantidad"]}}}},

{$sort:{ingresos\_totales:-1}},

{$limit:1}

])

## Ejercicio 8: Calcular el producto más vendido en términos de cantidad

Enunciado: Identifica cuál es el producto con mayor cantidad vendida.

db.ventas.aggregate([{$sort:{cantidad:-1}},{$limit:1}])

## Ejercicio 9: Encontrar el precio más alto y más bajo por categoría

Enunciado: Obtén el precio máximo y mínimo de los productos dentro de cada

categoría.

db.ventas.aggregate([

{$group:{\_id: "$categoria", maximo:{$max:"$precio"}, minimo:{$min:"$precio"}}}

])

## 

## Ejercicio 10: Generar un ranking de ventas por producto

Enunciado: Crea un ranking de productos basado en las ventas totales (precio \*

cantidad), incluyendo la posición de cada producto.

db.ventas.aggregate([

{$addFields:{ventas\_totales:{$multiply:["$precio", "$cantidad"]}}},

{$sort:{ventas\_totales:-1}},

{$group: {\_id: null,productos: { $push: { producto: "$producto", ventas\_totales: "$ventas\_totales" }}}},

{$unwind:{path: "$productos", includeArrayIndex: "posicion"}},

{$project:{\_id: 0, producto: "$productos.producto", ventas\_totales: "$productos.ventas\_totales", posicion: { $add: ["$posicion",1]}}}

])