

Atividade 4: Implementar operações do sistema de vocês que usa múltiplos stages de aggregation pipeline

Análise de Clientes por Faixa Etária (para grupo):

Grupos de (0-18, 19-25, 26-35, 36-45, 46-55,...) anos, dentro desses grupos temos quantidade de clientes, gastos médios, e top produtos consumidos.

Dentro do MongoDB

```
([{ $bucket: {
  groupBy: "$idade",
  boundaries: [0, 18, 25, 35, 45, 55, 65, 100],
  default: "Outros",
  output: {
    count: { $sum: 1 },
    totalGasto: { $sum: "$ultima_compra.valor" },
    produtosPopulares: { $push: "$ultima_compra.produto" } } },
{ $project: { _id: 0,
  faixaEtaria: "$_id",
  quantidadeClientes: "$count",
  gastoMedio: { $round: [ { $divide: [ "$totalGasto", "$count" ] }, 2 ] },
  topProdutos: { $slice: [ {
    $reduce: {
      input: "$produtosPopulares",
      initialValue: [],
      in: { $concatArrays: [ "$$value", ["$$this"] ] }, 5 } ] },
{ $sort: { faixaEtaria: 1 } } ] )}]
```

The screenshot displays the MongoDB Aggregation Pipeline Output. On the left, the aggregation pipeline is shown in a code editor. On the right, the output is displayed in a table format, showing a sample of 5 documents. The output is organized into three sections, each representing a different age group (faixaEtaria).

faixaEtaria : 18

- quantidadeClientes : 4
- gastoMedio : 247.25
- topProdutos : Array (4)
 - 0: "Base L'Oréal"
 - 1: "Paleta MAC"
 - 2: "Batom MAC"
 - 3: "Bruma Natura"

faixaEtaria : 25

- quantidadeClientes : 6
- gastoMedio : 238.72
- topProdutos : Array (5)
 - 0: "Base MAC"
 - 1: "Base Natura"
 - 2: "Base Avon"
 - 3: "Máscara Avon"
 - 4: "Paleta Natura"

faixaEtaria : 35

Ativar o Windows
Acesse Configurações para ativar o Windows

Segmentação Recencia, Frequência, Valor monetário (cluster):

Recencia: Refere-se ao tempo desde a última compra do cliente;

Frequencia: Mede a regularidade com que um cliente realiza compras;

Valor monetário: Avalia o valor total gasto pelo cliente;

```
([{ $addFields: {
  recencia: { $dateDiff: {
    startDate: { $toDate: "$ultima_compra.data" },
    endDate: new Date(),
    unit: "day" } },
  ValorMonetario: "$ultima_compra.valor" } },
{ $bucket: {
  groupBy: "$recencia",
  boundaries: [0, 30, 90, 180, 365],
  default: "Inativo",
  output: {
    count: { $sum: 1 },
    valorMedio: { $avg: "$valorMonetario" },
    idadeMedia: { $avg: "$idade" } } } },
{ $project: {
  _id: 0,
  categoriaRecencia: "$_id",
  totalClientes: "$count",
  valorMedioGasto: { $round: ["$valorMedio", 2] },
  idadeMedia: { $round: ["$idadeMedia", 1] } } },
{ $sort: { categoriaRecencia: 1 } } ])
```

The screenshot displays a MongoDB Aggregation Pipeline interface. On the left, a query is written in a code editor. On the right, the 'PIPELINE OUTPUT' section shows a sample of 3 documents. The output is a table with 4 columns: categoriaRecencia, totalClientes, valorMedioGasto, and idadeMedia. The data is grouped by recencia (0, 30, 90, 180, 365) and sorted by categoriaRecencia (1).

categoriaRecencia	totalClientes	valorMedioGasto	idadeMedia
0	2	378.09	48.5
30	11	271.45	34.7
90	7	230.49	34.6

Análise de Produtos Mais Vendidos:

Quantidade de vendas, valor total, nome do produto, valor médio.

```
([{ $group: {
  _id: "$ultima_compra.produto",
  totalVendas: { $sum: 1 },
  valorTotal: { $sum: "$ultima_compra.valor" },
  clientes: { $push: "$nome" } } },
{ $project: {
```

```

    _id: 0,
    produto: "$_id",
    totalVendas: 1,
    valorTotal: 1,
    valorMedio: { $round: [{ $divide: ["$valorTotal", "$totalVendas"] }, 2] },
    exemploClientes: { $slice: ["$clientes", 3] } }},
{ $sort: { totalVendas: -1 } },
{ $limit: 10 })

```

```

1  ({$group: {
2    _id: "$ultima_compra.produto",
3    totalVendas: { $sum: 1 },
4    valorTotal: { $sum: "$ultima_compra.valor" },
5    clientes: { $push: "$nome" } }},
6  {$project: {
7    _id: 0,
8    produto: "$_id",
9    totalVendas: 1,
10   valorTotal: 1,
11   valorMedio: { $round: [{ $divide: ["$valorTotal", "$totalVendas"] }, 2] },
12   exemploClientes: { $slice: ["$clientes", 3] } }},
13   { $sort: { totalVendas: -1 } },
14   { $limit: 10 })
15

```

PIPELINE OUTPUT

Sample of 10 documents

totalVendas : 3

valorTotal : 531.61

produto : "Paleta Natura"

valorMedio : 177.2

exemploClientes : Array (3)

0: "Ana Silva"

1: "Carlos Santos"

2: "Carlos Silva"

totalVendas : 2

valorTotal : 832.77

produto : "Base Avon"

valorMedio : 416.38

exemploClientes : Array (2)

0: "Carlos Silva"

1: "Ana Santos"

totalVendas : 2

valorTotal : 499.7899999999999

Cientes com Maior Valor de Compra:

Pesquisa o Id, nome, idade, produto e valor e data da última compra

```

([{$sort:
{ "ultima_compra.valor": -1 }},
{$limit: 10},
{$project: {
  _id: 0,
  id: 1,
  nome: 1,
  idade: 1,
  produto: "$ultima_compra.produto",
  valorCompra: "$ultima_compra.valor",
  dataCompra: "$ultima_compra.data" } }])

```

Sort

Limit

Project

Generate aggregation

Explain

Export

Run

Options

Untitled - modified

SAVE

CREATE NEW

EXPORT TO LANGUAGE

PREVIEW

STAGES

TEXT

WIZARD

```

1  {({$sort:
2  { "ultima_compra.valor": -1 }},
3  {$limit: 10},
4  {$project: {
5    _id: 0,
6    id: 1,
7    nome: 1,
8    idade: 1,
9    produto: "$ultima_compra.produto",
10   valorCompra: "$ultima_compra.valor",
11   dataCompra: "$ultima_compra.data"}}})
12

```

PIPELINE OUTPUT

Sample of 10 documents

OUTPUT OPTIONS

id : "C1015"

nome : "Juliana Santos"

idade : 19

produto : "Bruma Natura"

valorCompra : 499.37

dataCompra : "2025-07-08"

id : "C1006"

nome : "Pedro Santos"

idade : 35

produto : "Paleta MAC"

valorCompra : 434.46

dataCompra : "2025-06-26"

id : "C1007"

nome : "Carlos Silva"

idade : 34

produto : "Base Avon"

valorCompra : 430.67

dataCompra : "2025-06-03"

Análise por idade:

Consta as análises por idade das faixas etárias (20-29, 30-39, 40-49, 50-59), mostrando sua variedade de produtos e o valor médio comprado.

```

([{$addFields: {
  faixaEtaria: {$switch: {
    branches: [
      { case: { $lt: ["$idade", 20] }, then: "Menor que 20" },
      { case: { $lt: ["$idade", 30] }, then: "20-29" },
      { case: { $lt: ["$idade", 40] }, then: "30-39" },
      { case: { $lt: ["$idade", 50] }, then: "40-49" },
      { case: { $lt: ["$idade", 60] }, then: "50-59" } },
      default: "60+" } } } },
{$group: {
  _id: "$faixaEtaria",
  totalClientes: { $sum: 1 },
  valorMedioCompra: { $avg: "$ultima_compra.valor" },
  produtosDiferentes: { $addToSet: "$ultima_compra.produto" } } },
{$project: {
  _id: 0,
  faixaEtaria: "$_id",
  totalClientes: 1,
  valorMedioCompra: { $round: ["$valorMedioCompra", 2] },
  variedadeProdutos: { $size: "$produtosDiferentes" } } },
{ $sort: { faixaEtaria: 1 } } ]])

```

Untitled - modified
 SAVE
+ CREATE NEW
EXPORT TO LANGUAGE
PREVIEW
STAGES
TEXT
WIZARD

```

1  ([{$addFields: {
2    faixaEtaria: {$switch: {
3      branches: [
4        { case: { $lt: ["$idade", 20] }, then: "Menor que 20" },
5        { case: { $lt: ["$idade", 30] }, then: "20-29" },
6        { case: { $lt: ["$idade", 40] }, then: "30-39" },
7        { case: { $lt: ["$idade", 50] }, then: "40-49" },
8        { case: { $lt: ["$idade", 60] }, then: "50-59" }}
9      default: "60+"}}},
10   {$group: {
11     _id: "$faixaEtaria",
12     totalClientes: { $sum: 1 },
13     valorMedioCompra: { $avg: "$ultima_compra.valor" },
14     produtosDiferentes: { $addToSet: "$ultima_compra.produto" }
15   },
16   {$project: {
17     _id: 0,
18     faixaEtaria: "$_id",
19     totalClientes: 1,
20     valorMedioCompra: { $round: ["$valorMedioCompra", 2] },
21     variedadeProdutos: { $size: "$produtosDiferentes" } } }
22   },
23   {$sort: { faixaEtaria: 1 } } } ]

```

PIPELINE OUTPUT
OUTPUT OPTIONS

Sample of 5 documents

totalClientes : 4
 faixaEtaria : "20-29"
 valorMedioCompra : 239.99
 variedadeProdutos : 4

totalClientes : 6
 faixaEtaria : "30-39"
 valorMedioCompra : 282.92
 variedadeProdutos : 6

totalClientes : 5
 faixaEtaria : "40-49"
 valorMedioCompra : 277.29
 variedadeProdutos : 4

Comandos:

\$match	Filtra documentos (como WHERE no SQL)
\$group	Agrupa documentos por campo(s) especificado(s)
\$project	Seleciona/renomeia/remove campos (como SELECT no SQL)
\$sort	Ordena os documentos
\$limit	Limita o número de documentos
\$skip	Pula um número de documentos
\$lookup	Faz join entre coleções (como JOIN no SQL)
\$unwind	Desconstrói um array em vários documentos
\$addFields	Adiciona novos campos
\$count	Conta documentos
\$sum	Soma valores
\$avg	Calcula média
\$max	Encontra valor máximo
\$min	Encontra valor mínimo
\$push	Cria array com valores do grupo
\$first	Pega o primeiro valor do grupo
\$last	Pega o último valor do grupo
\$concat	Concatena strings
\$toUpper	Converte para maiúsculas
\$toLower	Converte para minúsculas
\$substr	Extraí parte de uma string
\$dateToString	Formata data como string
\$month	Extraí mês de uma data
\$year	Extraí ano de uma data
\$cond	Operador condicional (IF-THEN-ELSE)
\$switch	Condiciona com múltiplos casos
\$bucket	Agrupa em intervalos pré-definidos
\$bucketAuto	Agrupa em intervalos automaticamente calculados
\$facet	Executa múltiplos pipelines no mesmo estágio
\$out	Grava resultados em uma coleção