

## Atividade Revisão

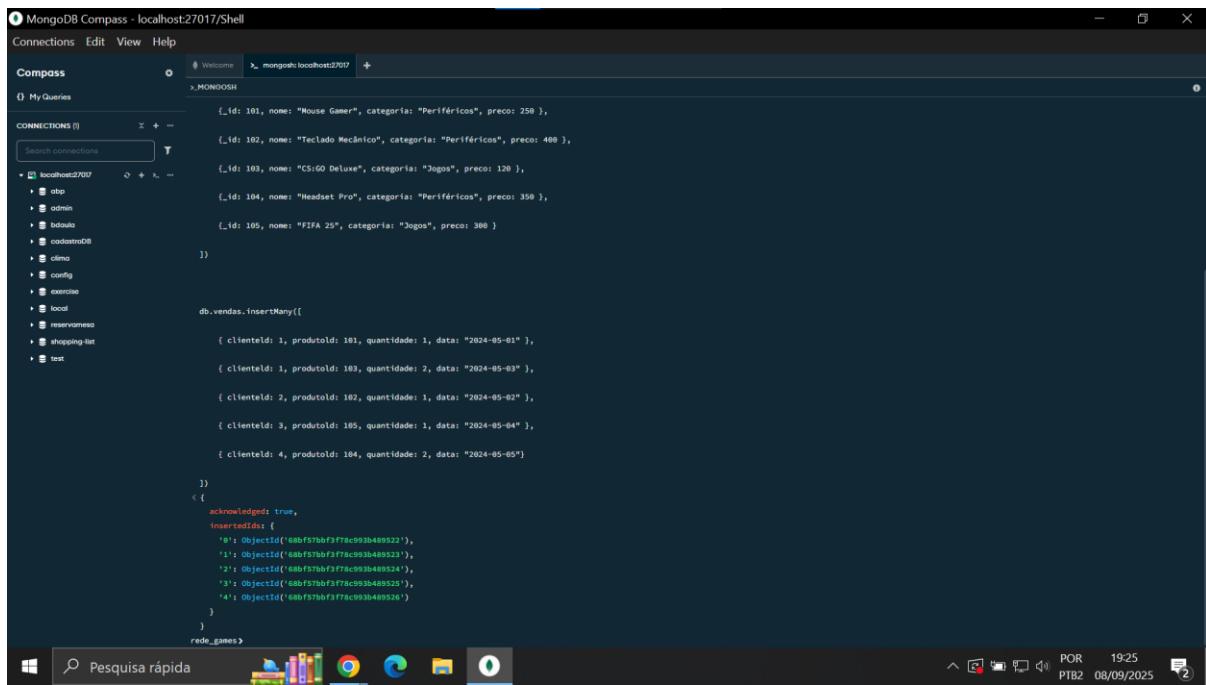
### Questão 1 – Criação do Banco e Coleções

```
use rede_games
```

```
db.clientes.insertMany([
    {_id: 1, nome: "Marcos", idade: 25, cidade: "Fortaleza" },
    {_id: 2, nome: "Julia", idade: 30, cidade: "São Paulo" },
    {_id: 3, nome: "Lucas", idade: 19, cidade: "Recife" },
    {_id: 4, nome: "Ana", idade: 27, cidade: "Fortaleza"}
])
```

```
db.produtos.insertMany([
    {_id: 101, nome: "Mouse Gamer", categoria: "Periféricos", preco: 250 },
    {_id: 102, nome: "Teclado Mecânico", categoria: "Periféricos", preco: 400 },
    {_id: 103, nome: "CS:GO Deluxe", categoria: "Jogos", preco: 120 },
    {_id: 104, nome: "Headset Pro", categoria: "Periféricos", preco: 350 },
    {_id: 105, nome: "FIFA 25", categoria: "Jogos", preco: 300 }
])
```

```
db.vendas.insertMany([
    { clientelid: 1, produtolid: 101, quantidade: 1, data: "2024-05-01" },
    { clientelid: 1, produtolid: 103, quantidade: 2, data: "2024-05-03" },
    { clientelid: 2, produtolid: 102, quantidade: 1, data: "2024-05-02" },
    { clientelid: 3, produtolid: 105, quantidade: 1, data: "2024-05-04" },
    { clientelid: 4, produtolid: 104, quantidade: 2, data: "2024-05-05" }
])
```



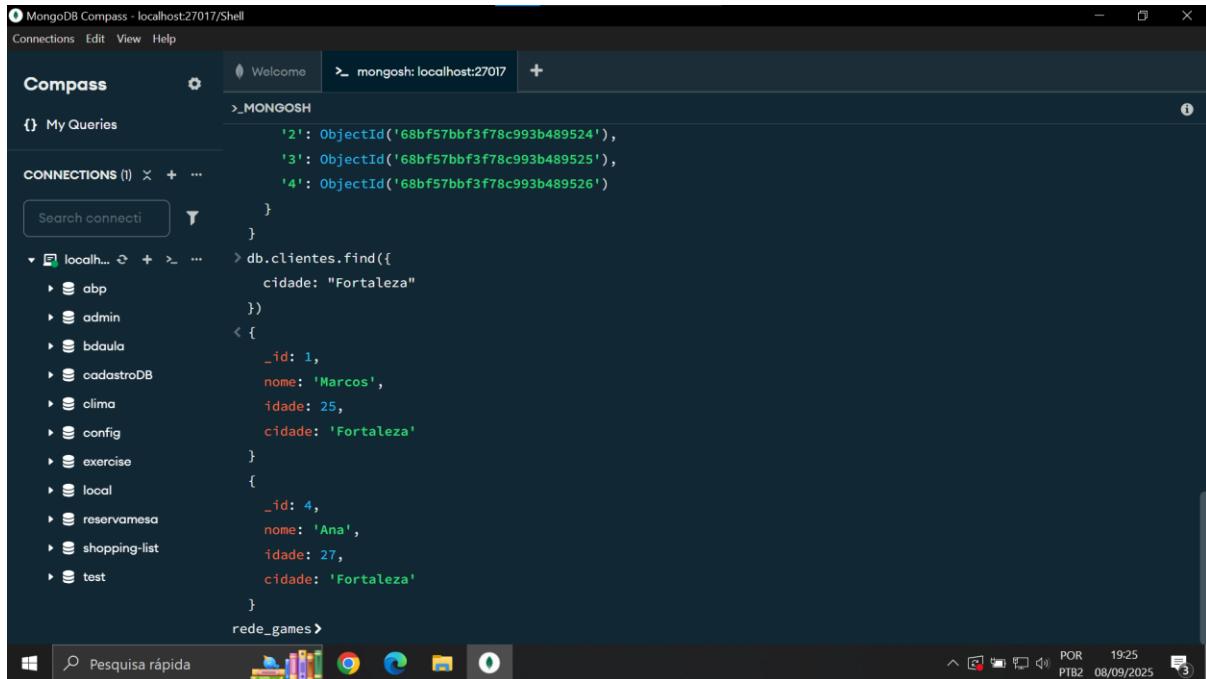
```
mongosh:localhost:27017> db.vendas.insertMany([
  { clientelid: 1, produtolid: 101, quantidade: 1, data: "2024-05-01" },
  { clientelid: 1, produtolid: 103, quantidade: 2, data: "2024-05-03" },
  { clientelid: 2, produtolid: 102, quantidade: 1, data: "2024-05-02" },
  { clientelid: 3, produtolid: 105, quantidade: 1, data: "2024-05-04" },
  { clientelid: 4, produtolid: 104, quantidade: 2, data: "2024-05-05" }
])
{
  acknowledged: true,
  insertedIds: [
    '1': ObjectId('68bf57bbf3f78c993b489521'),
    '1': ObjectId('68bf57bbf3f78c993b489523'),
    '2': ObjectId('68bf57bbf3f78c993b489524'),
    '3': ObjectId('68bf57bbf3f78c993b489525'),
    '4': ObjectId('68bf57bbf3f78c993b489526')
  ]
}
```

## Questão 2 – Consulta com Filtro

```
db.clientes.find({
```

```
  cidade: "Fortaleza"
```

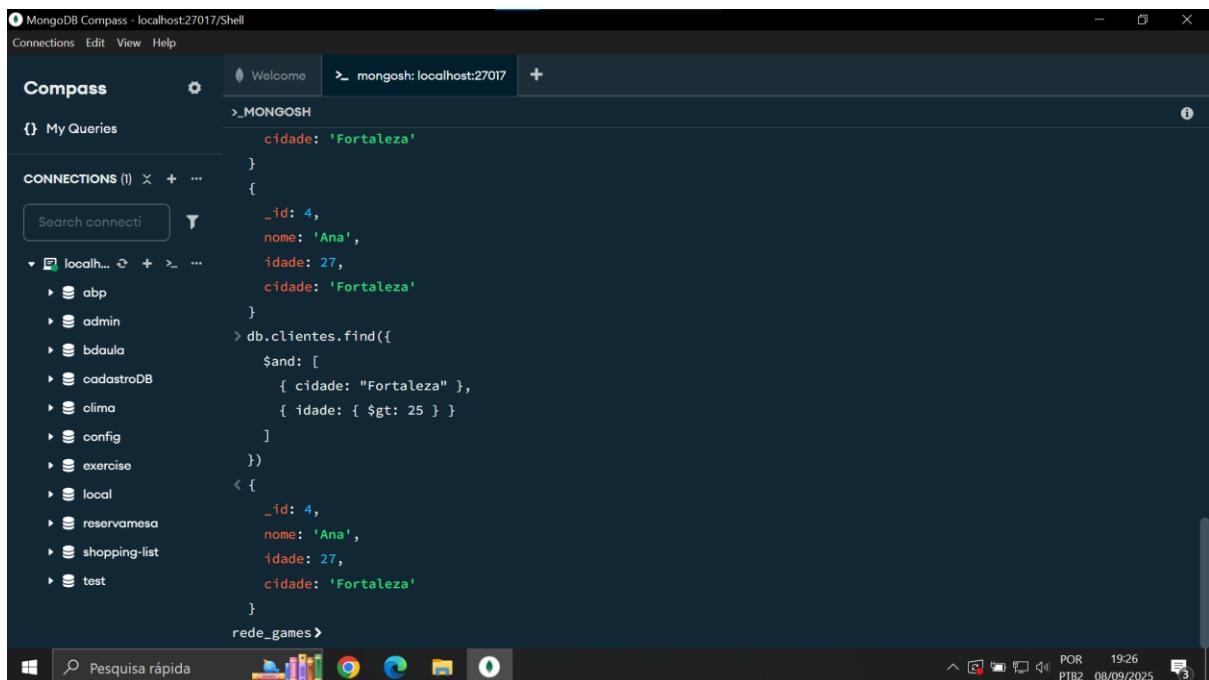
```
)
```



```
mongosh:localhost:27017> db.clientes.find({
  cidade: "Fortaleza"
})
{
  _id: 1,
  nome: 'Marcos',
  idade: 25,
  cidade: 'Fortaleza'
}
{
  _id: 4,
  nome: 'Ana',
  idade: 27,
  cidade: 'Fortaleza'
```

### Questão 3 – Consulta com \$and

```
db.clientes.find({  
    $and: [  
        { cidade: "Fortaleza" },  
        { idade: { $gt: 25 } }  
    ]  
})
```



### Questão 4 – Projeção

```
db.produtos.find(  
    { categoria: "Periféricos" },  
    { nome: 1, preco: 1, _id: 0 }  
)
```

The screenshot shows the MongoDB Compass interface. The top bar has tabs for 'Welcome' and 'mongosh: localhost:27017'. The main area is titled '\_MONGOSH' and contains the following code:

```
        nome: 'Ana',
        idade: 27,
        cidade: 'Fortaleza'
    }
} > db.produtos.find(
    { categoria: 'Periféricos' }, // Filtro
    { nome: 1, preco: 1, _id: 0 } // Projeção (1 para exibir, 0 para ocultar)
)
< {
    nome: 'Mouse Gamer',
    preco: 250
}
{
    nome: 'Teclado Mecânico',
    preco: 400
}
{
    nome: 'Headset Pro',
    preco: 350
}
rede_games>|
```

## Questão 5 – Consulta com \$gte e \$lte

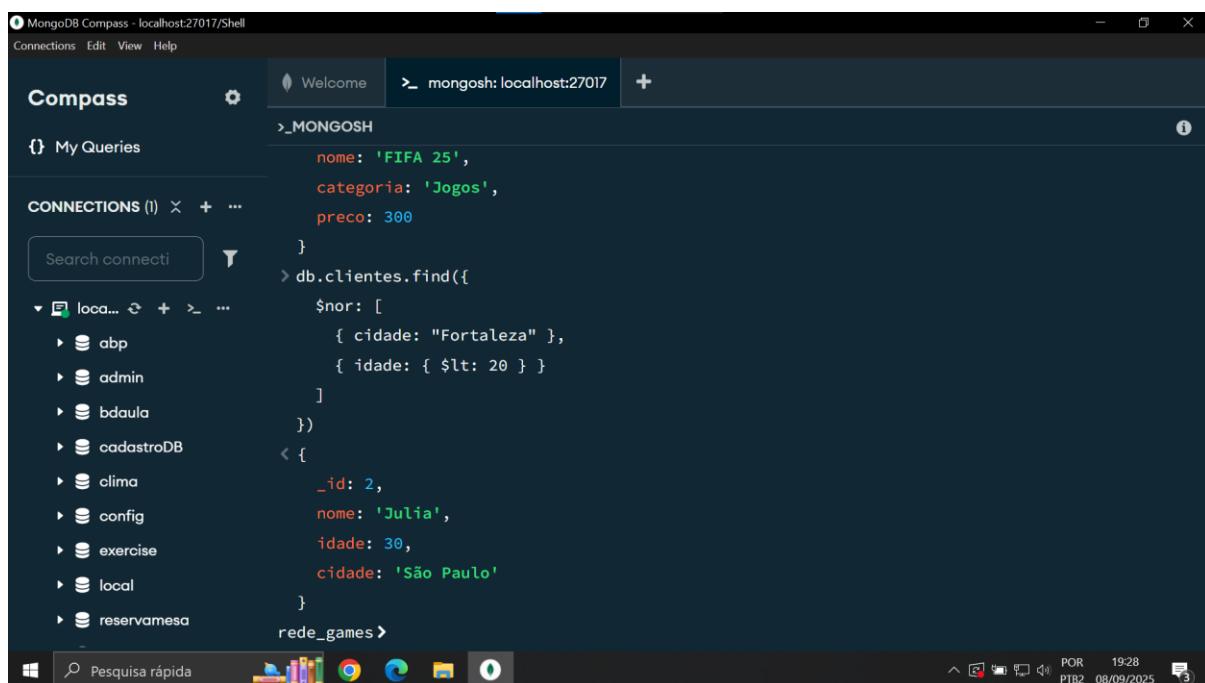
```
db.produtos.find({
    preco: { $gte: 200, $lte: 400 }
})
```

The screenshot shows the MongoDB Compass interface with the same shell tab open. The results of the query are displayed:

```
        _id: 101,
        nome: 'Mouse Gamer',
        categoria: 'Periféricos',
        preco: 250
}
{
    _id: 102,
    nome: 'Teclado Mecânico',
    categoria: 'Periféricos',
    preco: 400
}
{
    _id: 104,
    nome: 'Headset Pro',
    categoria: 'Periféricos',
    preco: 350
}
{
    _id: 105,
    nome: 'FIFA 25',
    categoria: 'Jogos',
    preco: 300
}
rede_games>|
```

Questão 6 – Consulta com \$nor e \$not

```
db.clientes.find({  
    $nor: [  
        { cidade: "Fortaleza" },  
        { idade: { $lt: 20 } }  
    ]  
})
```



The screenshot shows the MongoDB Compass interface. On the left, there's a sidebar with 'Compass' and 'My Queries'. Below it, under 'CONNECTIONS', there are several database connections listed. In the main area, the command line shows a query being run:

```
_MONGOSH  
  nome: 'FIFA 25',  
  categoria: 'Jogos',  
  preco: 300  
}  
db.clientes.find({  
    $nor: [  
        { cidade: "Fortaleza" },  
        { idade: { $lt: 20 } }  
    ]  
})  
< {  
  _id: 2,  
  nome: 'Julia',  
  idade: 30,  
  cidade: 'São Paulo'  
}  
rede_games >
```

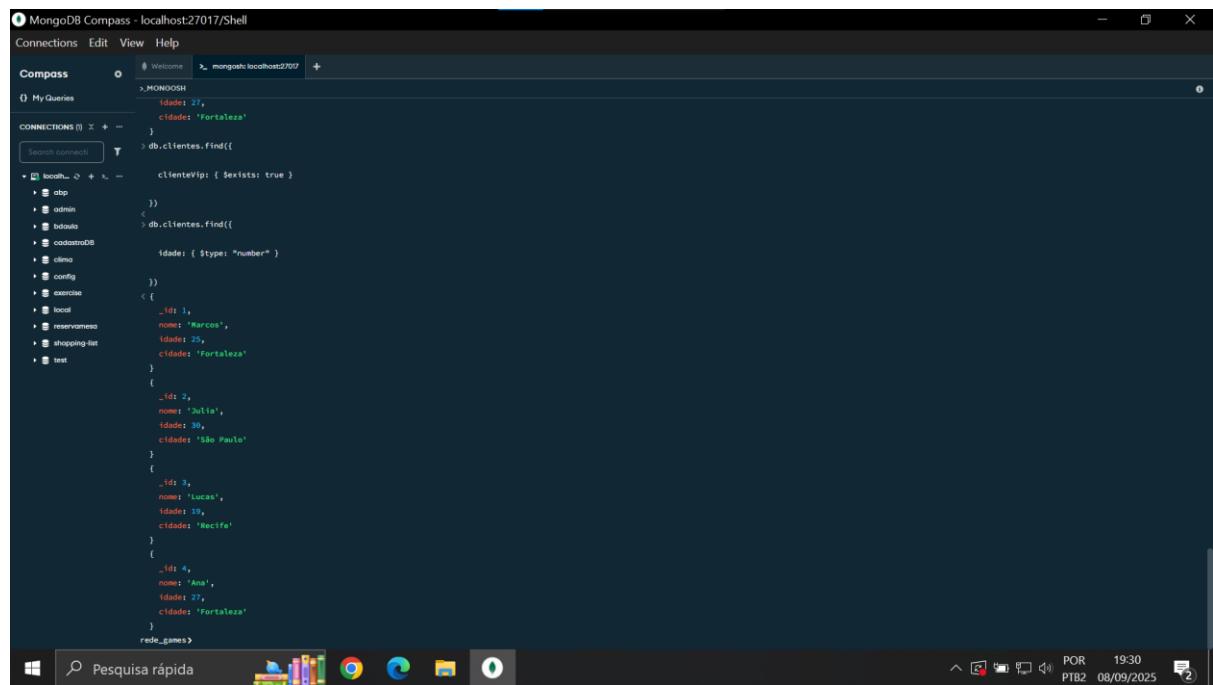
The results of the query are displayed below the command line, showing one document that matches the query criteria.

Questão 7 – Consulta com \$exists e \$type

```
db.clientes.find({  
    clienteVip: { $exists: true }  
})
```

```
db.clientes.find({  
    idade: { $type: "number" }  
})
```

})



The screenshot shows the MongoDB Compass interface with the title "MongoDB Compass - localhost:27017/Shell". The left sidebar lists databases: "Compass", "My Queries", and "CONNECTIONS". The main area displays a MongoDB shell command:

```
> db.clientes.find({  
  nome: "Jorge",  
  idade: 22,  
  cidade: "Salvador"  
})
```

The results show four documents:

- Object ID: 1, nome: "Jorge", idade: 22, cidade: "Salvador"
- Object ID: 2, nome: "Julia", idade: 20, cidade: "São Paulo"
- Object ID: 3, nome: "Lucas", idade: 19, cidade: "Recife"
- Object ID: 4, nome: "Ana", idade: 27, cidade: "Fortaleza"

The bottom status bar shows: POR 19:30 PTB2 08/09/2025.

## Questão 8 – Inserção de Novo Cliente

```
db.clientes.insertOne({  
  
  nome: "Jorge",  
  
  idade: 22,  
  
  cidade: "Salvador"  
})
```

```

MongoDB Compass - localhost:27017/Shell
Connections Edit View Help

Compass
  Connections (i) + ...
  My Queries
  Search connecti

  Welcome mongosh: localhost:27017 +
>_MONGOSH
  _id: 4,
  nome: 'Ana',
  idade: 27,
  cidade: 'Fortaleza'
}
> db.clientes.insertOne({
  nome: "Jorge",
  idade: 22,
  cidade: "Salvador"
})
< {
  acknowledged: true,
  insertedId: ObjectId('68bf5912f3f78c993b489527')
}

```

## Questão 9 – Atualização

```

db.clientes.updateOne(
  { nome: "Marcos" },
  { $set: { cidade: "Natal" } }
)

```

```

MongoDB Compass - localhost:27017/Shell
Connections Edit View Help

Compass
  Connections (i) + ...
  My Queries
  Search connecti

  Welcome mongosh: localhost:27017 +
>_MONGOSH
  acknowledged: true,
  insertedId: ObjectId('68bf5912f3f78c993b489527')
}
> db.clientes.updateOne(
  { nome: "Marcos" },
  { $set: { cidade: "Natal" } }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

## Questão 10 – Atualização com \$set e \$inc

```

db.clientes.updateOne(
  { nome: "Lucas" },
  {
    $set: { status: "ativo" },
    $inc: { idade: 1 }
  }
)

```

The screenshot shows the MongoDB Compass interface. The top bar displays 'MongoDB Compass - localhost:27017/Shell' and the connection details 'mongosh:localhost:27017'. The left sidebar lists databases: 'bbddia', 'cadastroDB', 'clima', 'config', 'exercise', 'local', 'reservamesa', 'shopping-list', and 'test'. The main area shows the execution of a query:

```

> db.clientes.updateOne(
  { nome: "Lucas" },
  {
    $set: { status: "ativo" },
    $inc: { idade: 1 }
  }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

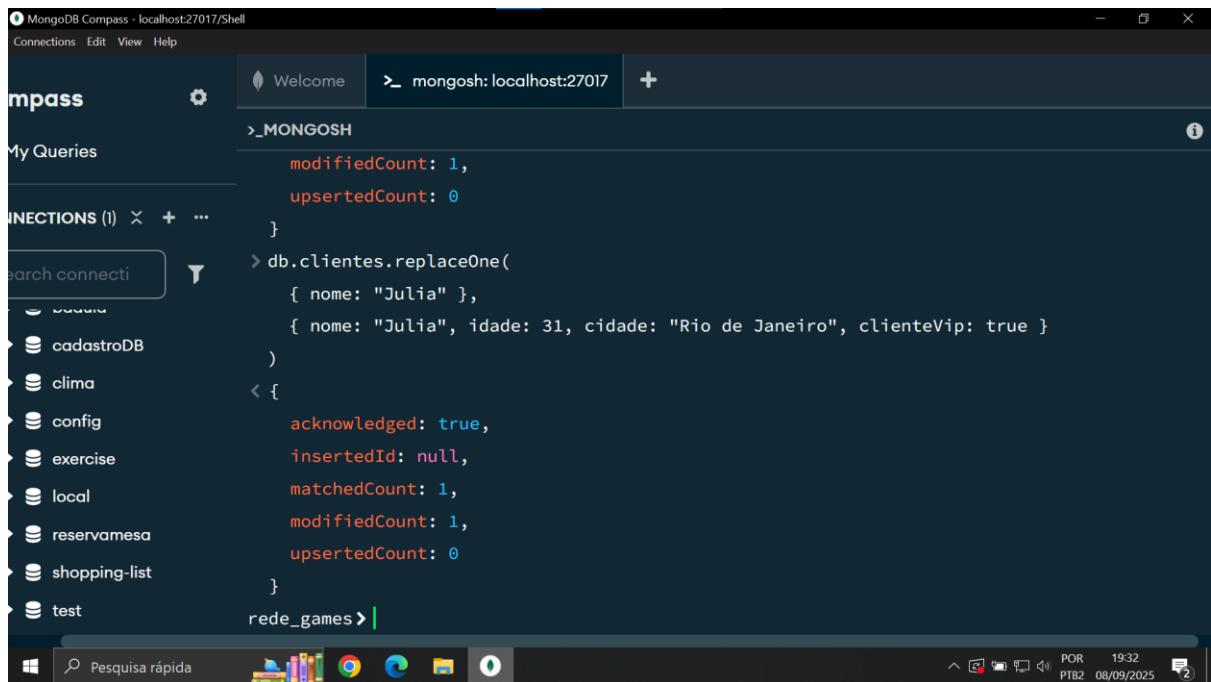
The results indicate that one document was matched, one was modified, and no new documents were inserted.

### Questão 11 – Atualização com replaceOne

```

db.clientes.replaceOne(
  { nome: "Julia" },
  { nome: "Julia", idade: 31, cidade: "Rio de Janeiro", clienteVip: true }
)

```



A screenshot of the MongoDB Compass interface. The title bar says "MongoDB Compass - localhost:27017/Shell". The left sidebar shows databases: bdaula, cadastroDB, clima, config, exercise, local, reservamesa, shopping-list, and test. The main panel shows the command line: >\_MONGOSH, followed by a successful updateOne operation on the db.clientes collection. The output shows modifiedCount: 1 and upsertedCount: 0.

```
_MONGOSH
    modifiedCount: 1,
    upsertedCount: 0
}
> db.clientes.replaceOne(
    { nome: "Julia" },
    { nome: "Julia", idade: 31, cidade: "Rio de Janeiro", clienteVIP: true }
)
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 1,
    modifiedCount: 1,
    upsertedCount: 0
}
```

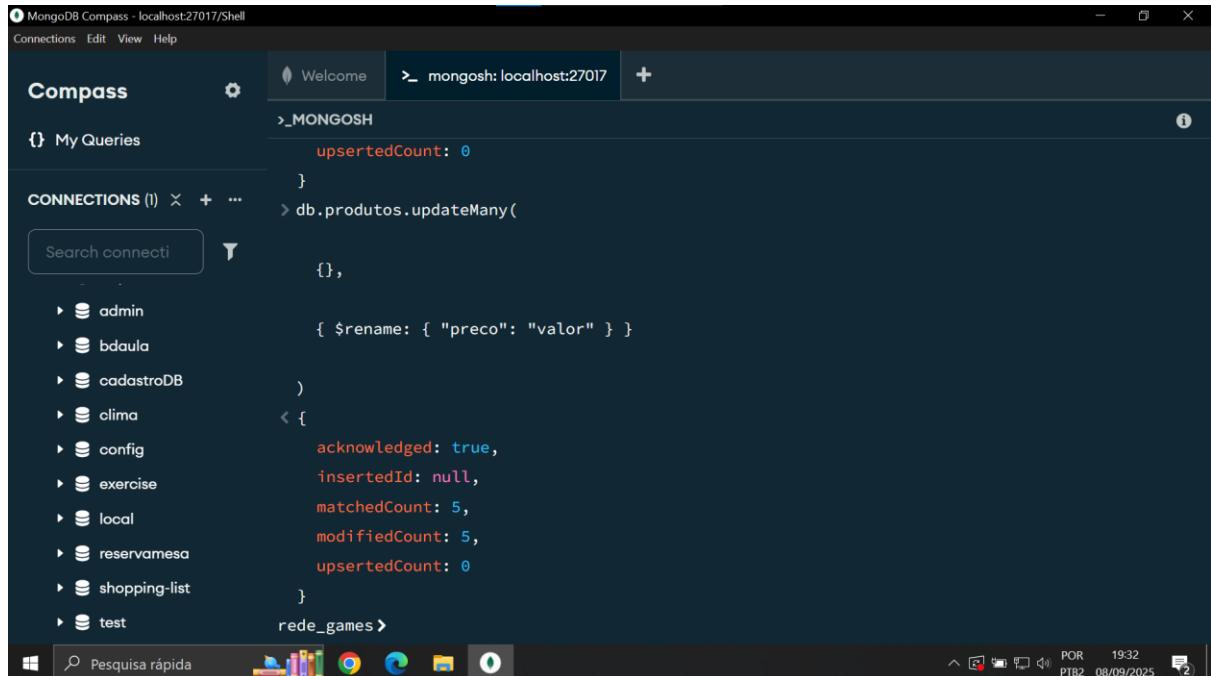
## Questão 12 – Atualização com \$rename e \$unset

```
db.produtos.updateMany(
```

```
{},
```

```
 { $rename: { "preco": "valor" } }
```

```
)
```



A screenshot of the MongoDB Compass interface. The title bar says "MongoDB Compass - localhost:27017/Shell". The left sidebar shows databases: admin, bdaula, cadastroDB, clima, config, exercise, local, reservamesa, shopping-list, and test. The main panel shows the command line: >\_MONGOSH, followed by a successful updateMany operation on the db.produtos collection. The output shows upsertedCount: 0.

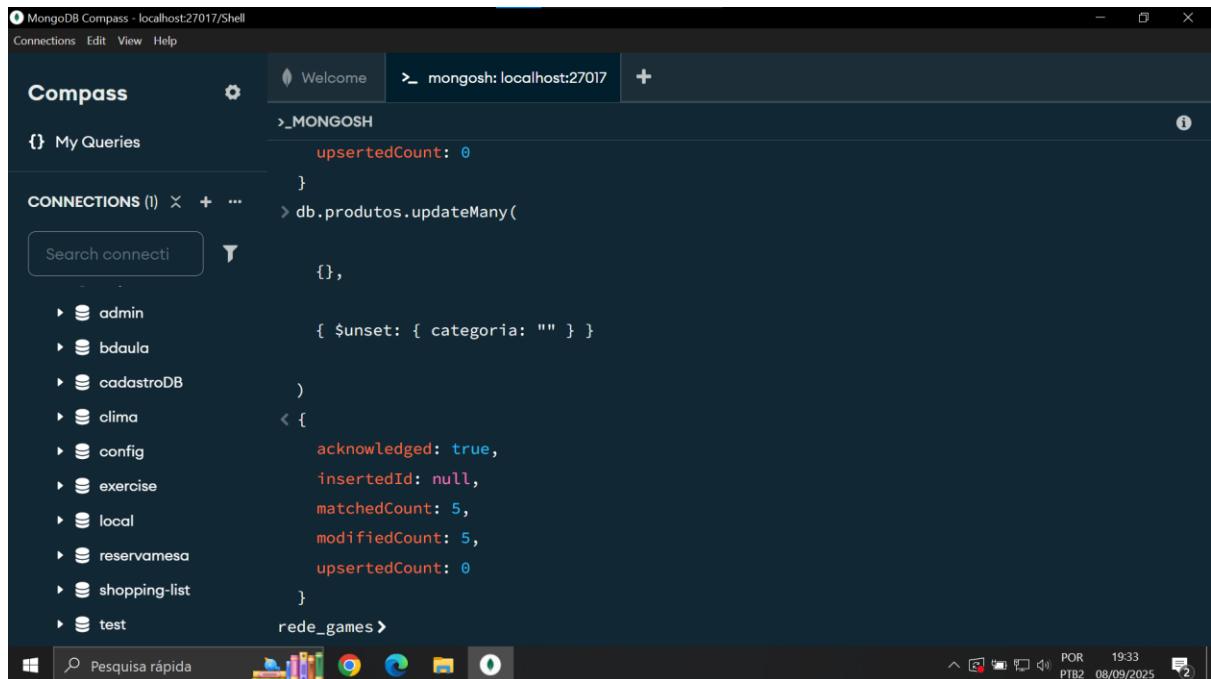
```
_MONGOSH
    upsertedCount: 0
}
> db.produtos.updateMany(
    {},
    { $rename: { "preco": "valor" } }
)
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 5,
    modifiedCount: 5,
    upsertedCount: 0
}
```

```
db.produtos.updateMany(
```

```
{},
```

```
{ $unset: { categoria: "" } }
```

```
)
```

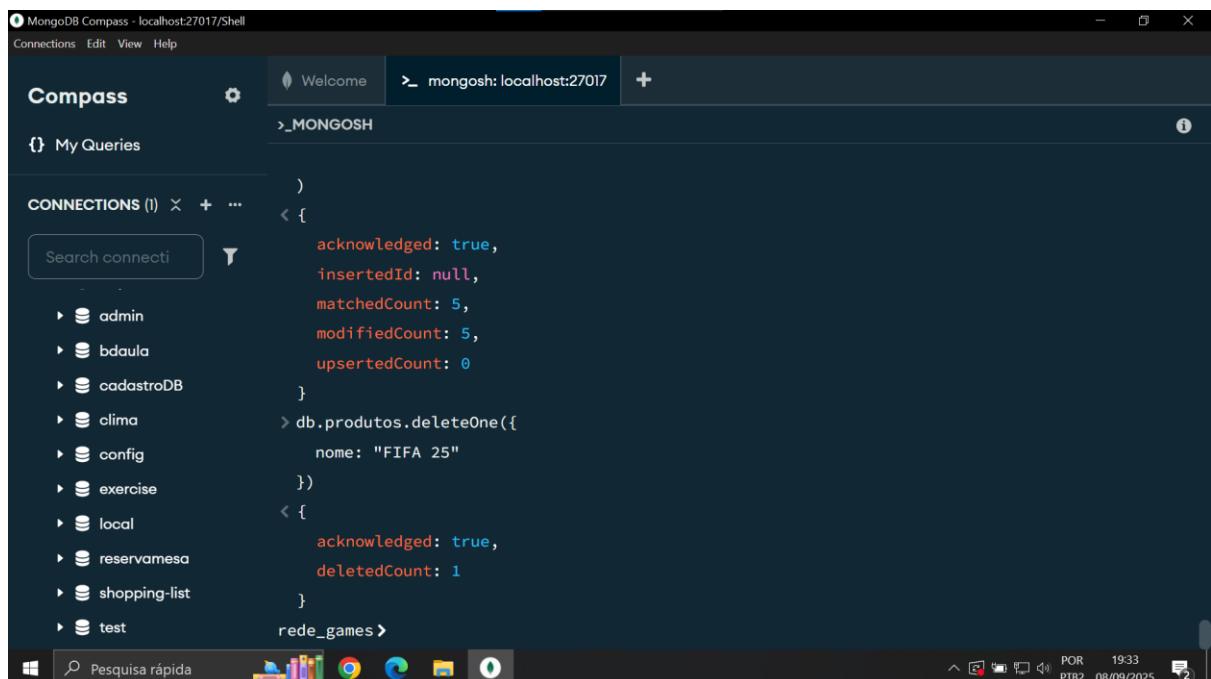


The screenshot shows the MongoDB Compass interface connected to 'mongosh: localhost:27017'. In the left sidebar, there's a 'CONNECTIONS' section listing various databases: admin, bdaula, cadastroDB, clima, config, exercise, local, reservamesa, shopping-list, and test. The main query editor window contains the following code:

```
>_MONGOSH
    upsertedCount: 0
}
} )
> db.produtos.updateMany(
    {},
    {
        $unset: { categoria: "" }
    }
)
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 5,
    modifiedCount: 5,
    upsertedCount: 0
}
rede_games >
```

### Questão 13 – Exclusão

```
db.produtos.deleteOne({
    nome: "FIFA 25"
})
```



The screenshot shows the MongoDB Compass interface connected to 'mongosh: localhost:27017'. In the left sidebar, there's a 'CONNECTIONS' section listing various databases: admin, bdaula, cadastroDB, clima, config, exercise, local, reservamesa, shopping-list, and test. The main query editor window contains the following code:

```
>_MONGOSH
    )
< {
    acknowledged: true,
    insertedId: null,
    matchedCount: 5,
    modifiedCount: 5,
    upsertedCount: 0
}
> db.produtos.deleteOne({
    nome: "FIFA 25"
})
< {
    acknowledged: true,
    deletedCount: 1
}
rede_games >
```

### Questão 14 – Operadores Lógicos

```

db.clientes.find({
  $or: [
    { cidade: "São Paulo" },
    { cidade: "Recife" }
  ]
})

```

The screenshot shows the MongoDB Compass interface. On the left, there's a sidebar titled 'Compass' with sections for 'My Queries' and 'CONNECTIONS'. The main area is a code editor with the following content:

```

>_MONGOSH
      acknowledged: true,
      deletedCount: 1
}
> db.clientes.find({
  $or: [
    { cidade: "São Paulo" },
    { cidade: "Recife" }
  ]
}
< {
  _id: 3,
  nome: 'Lucas',
  idade: 20,
  cidade: 'Recife',
  status: 'ativo'
}

```

The result pane shows the document found by the query. At the bottom, the operating system taskbar is visible with icons for search, file explorer, browser, and system status.

### Questão 15 – Agregação com \$lookup

```

db.vendas.aggregate([
  { $lookup: { from: "produtos", localField: "produtoid",
  foreignField: "_id", as: "dados_do_produto" } }
])

```

```

db.vendas.aggregate([
  {
    $lookup: {
      from: "produtos",
      localField: "produtold",
      foreignField: "_id",
      as: "detalhe_da_produto"
    }
  },
  {
    $unwind: {
      path: "$detalhe_da_produto"
    }
  },
  {
    $group: {
      _id: ObjectId("64bf57abf3f7fc993b489521"),
      clientelid: 1,
      produtoid: 1,
      quantidadeid: 1,
      data: "2024-05-01",
      detalhe_da_produto: [
        {
          _id: 101,
          nome: "Mouse Gamer",
          valor: 250
        }
      ]
    }
  },
  {
    _id: ObjectId("64bf57abf3f7fc993b489523"),
    clientelid: 1,
    produtoid: 103,
    quantidadeid: 2,
    data: "2024-05-01",
    detalhe_da_produto: [
      {
        _id: 103,
        nome: "CS:GO Deluxe",
        valor: 120
      }
    ]
  },
  {
    _id: ObjectId("64bf57abf3f7fc993b489524"),
    clientelid: 1,
    produtoid: 102,
  }
])

```

## Questão 16 – Agrupamento

```

db.vendas.aggregate([
  { $lookup: { from: "produtos", localField: "produtold",
  foreignField: "_id", as: "info_produto" } },
  { $unwind: "$info_produto" },
  { $group: { _id: "$info_produto.categoria",
  total_vendidos: { $sum: "$quantidade" } } }
])

```

```

db.vendas.aggregate([
  {
    $lookup: {
      from: "produtos",
      localField: "produtold",
      foreignField: "_id",
      as: "info_produto"
    }
  },
  {
    $unwind: "$info_produto"
  },
  {
    $group: {
      _id: "$info_produto.categoria",
      total_vendidos: { $sum: "$quantidade" }
    }
  }
])
< {
  _id: null,
  total_vendidos: 6
}
rede_games>

```

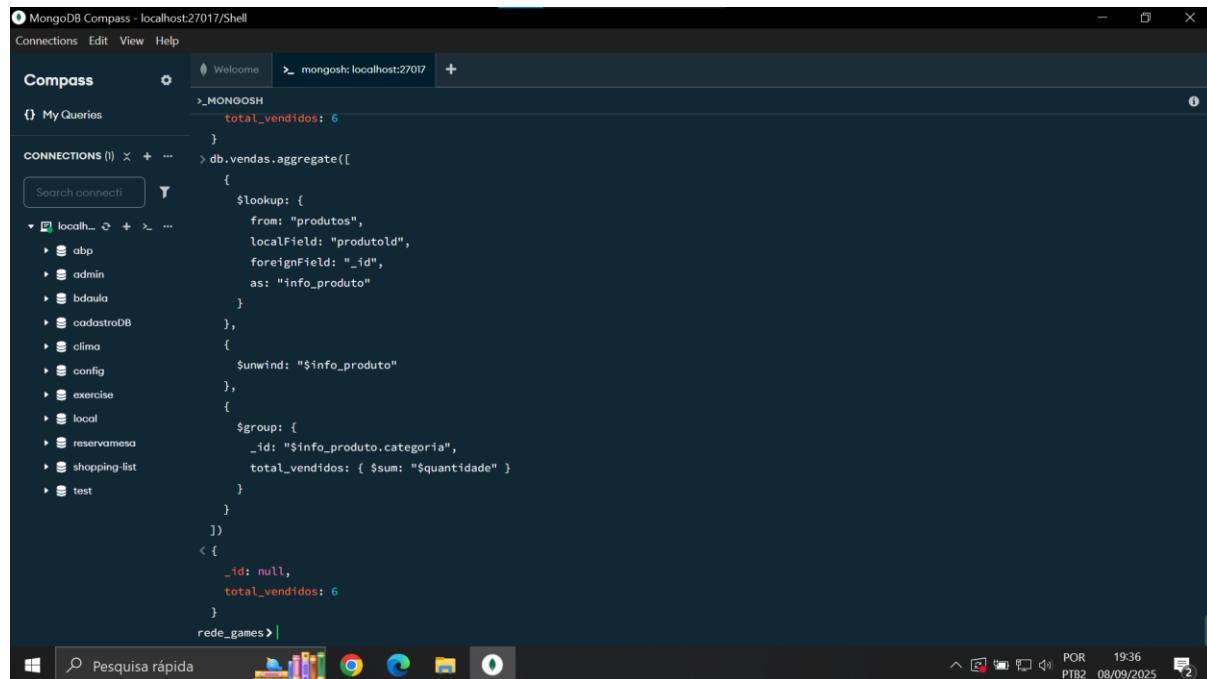
## Questão 17 – Agregação com \$lookup e \$group

```

db.vendas.aggregate([
  // Junta a coleção 'vendas' com a 'produtos' {
    $lookup: { from: "produtos",
      localField: "produtold",
      foreignField: "_id",
      as: "info_produto" }
  }, //
  // Desconstrói o array 'info_produto' para que cada documento represente uma venda
])

```

```
com um produto { $unwind: "$info_produto" }, // Agrupa os documentos pela categoria
do produto e soma as quantidades { $group: { _id: "$info_produto.categoria", // Agrupa
por categoria total_vendidos: { $sum: "$quantidade" } // Soma a quantidade de cada
venda no grupo } } ])
```



The screenshot shows the MongoDB Compass interface connected to the 'mongosh:localhost:27017' shell. The query being run is an aggregation pipeline:

```
>_MONGOSH
      total_vendidos: 6
    }
  > db.vendas.aggregate([
    {
      $lookup: {
        from: "produtos",
        localField: "produtolid",
        foreignField: "_id",
        as: "info_produto"
      }
    },
    {
      $unwind: "$info_produto"
    },
    {
      $group: {
        _id: "$info_produto.categoria",
        total_vendidos: { $sum: "$quantidade" }
      }
    }
  ])
< {
  _id: null,
  total_vendidos: 6
}
rede_games> |
```

## Questão 18 – Paginação

```
db.produtos.find()
```

```
.sort({ preco: -1 }) // -1 para ordem decrescente
```

```
.limit(2)
```

```

>_MONGOSH
    },
    {
      $group: {
        _id: "$info_produto.categoria",
        total_vendidos: { $sum: "$quantidade" }
      }
    }
  ]
)
< {
  _id: null,
  total_vendidos: 6
}
> db.produtos.find()
  .sort({ preco: -1 }) // -1 para ordem decrescente
  .limit(2)
< {
  _id: 101,
  nome: 'Mouse Gamer',
  valor: 250
}
{
  _id: 102,
  nome: 'Teclado Mecânico',
  valor: 400
}
>
rede_games>

```

## Questão 19 – Paginação com skip

`db.produtos.find()`

`.sort({ valor: -1 })` // Ordena do maior para o menor valor

`.skip(2)` // Pula os 2 primeiros (os mais caros)

`.limit(3)` // Exibe os 3 seguintes (o 3º, 4º e 5º)

```

>_MONGOSH
  .limit(2)
< {
  _id: 101,
  nome: 'Mouse Gamer',
  valor: 250
}
{
  _id: 102,
  nome: 'Teclado Mecânico',
  valor: 400
}
> db.produtos.find()
  .sort({ valor: -1 }) // Ordena do maior para o menor valor
  .skip(2)           // Pula os 2 primeiros (os mais caros)
  .limit(3)          // Exibe os 3 seguintes (o 3º, 4º e 5º)
< {
  _id: 101,
  nome: 'Mouse Gamer',
  valor: 250
}
{
  _id: 103,
  nome: 'CS:GO Deluxe',
  valor: 120
}
rede_games>

```