

## Atividade Revisão

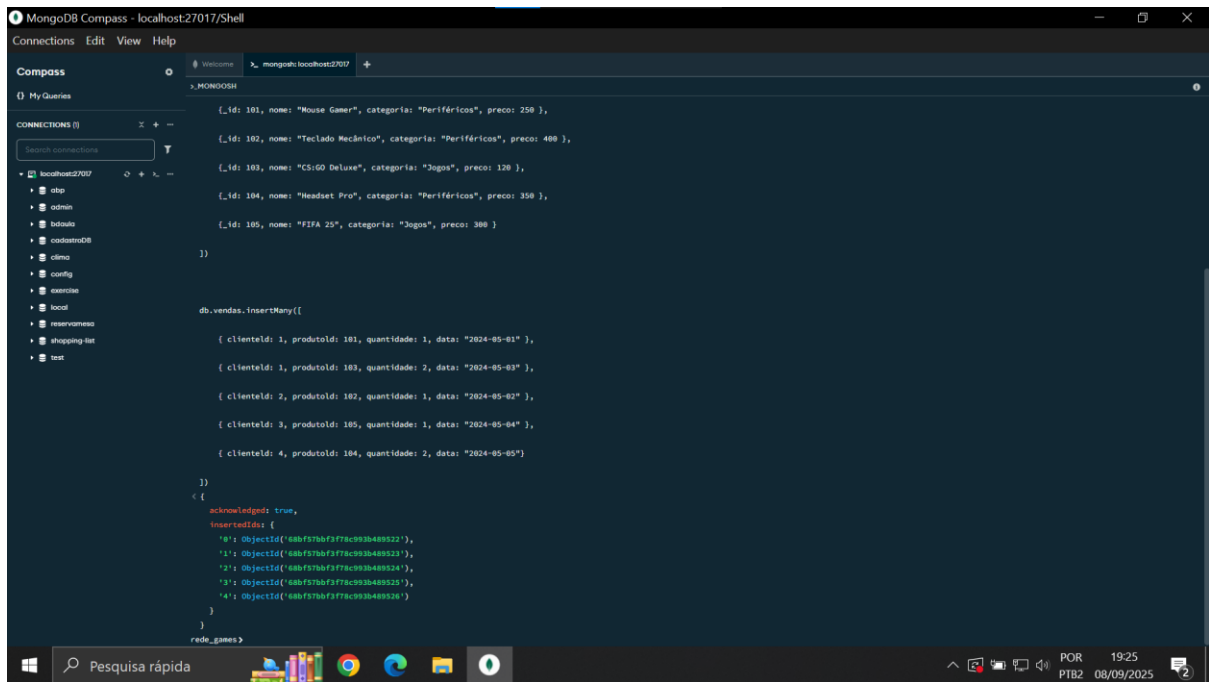
### Questão 1 – Criação do Banco e Coleções

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" }
])
```

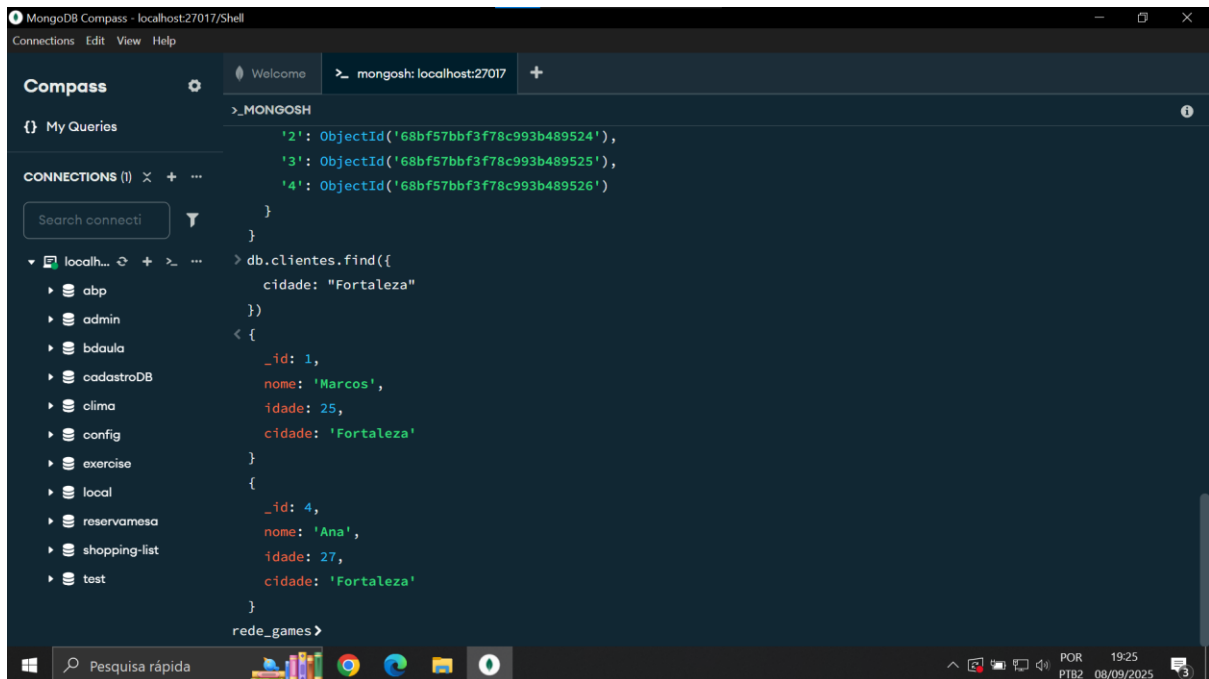
```
db.produtos.insertMany([
  { _id: 101, nome: "Mouse Gamer", categoria: "Periféricos", preco: 250 },
  { _id: 102, nome: "Teclado Mecânico", categoria: "Periféricos", preco: 400 },
  { _id: 103, nome: "CS:GO Deluxe", categoria: "Jogos", preco: 120 },
  { _id: 104, nome: "Headset Pro", categoria: "Periféricos",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

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



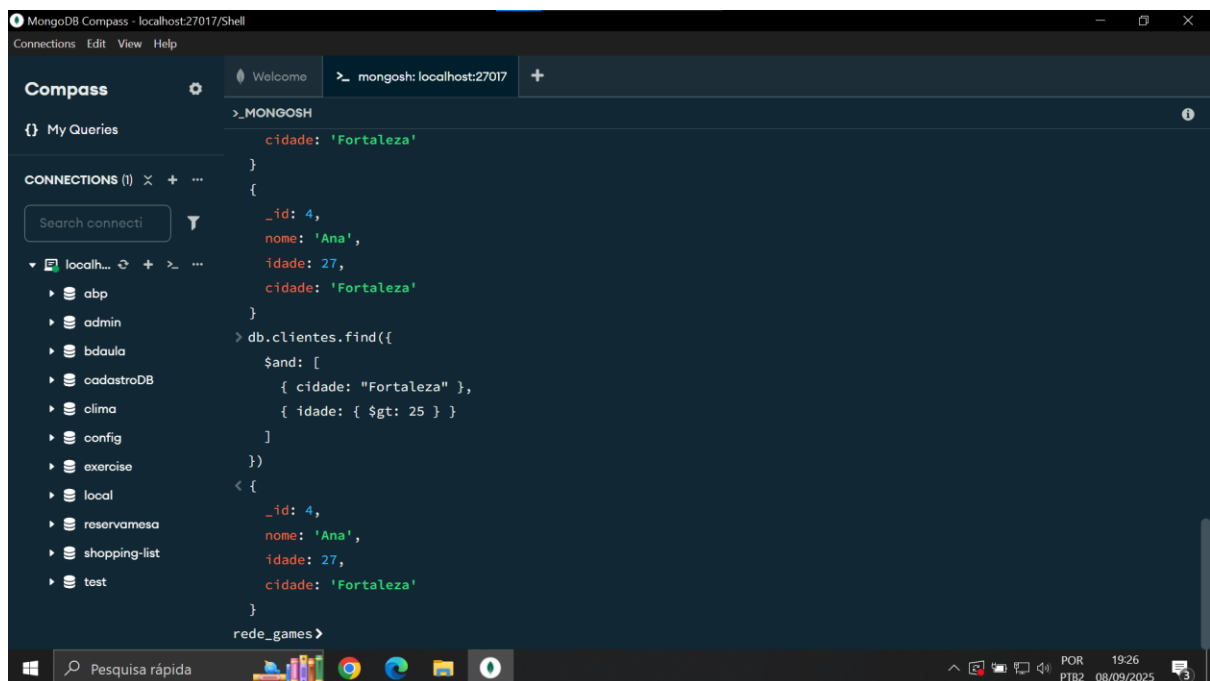
## Questão 2 – Consulta com Filtro

```
db.clientes.find({  
  cidade: "Fortaleza"  
})
```



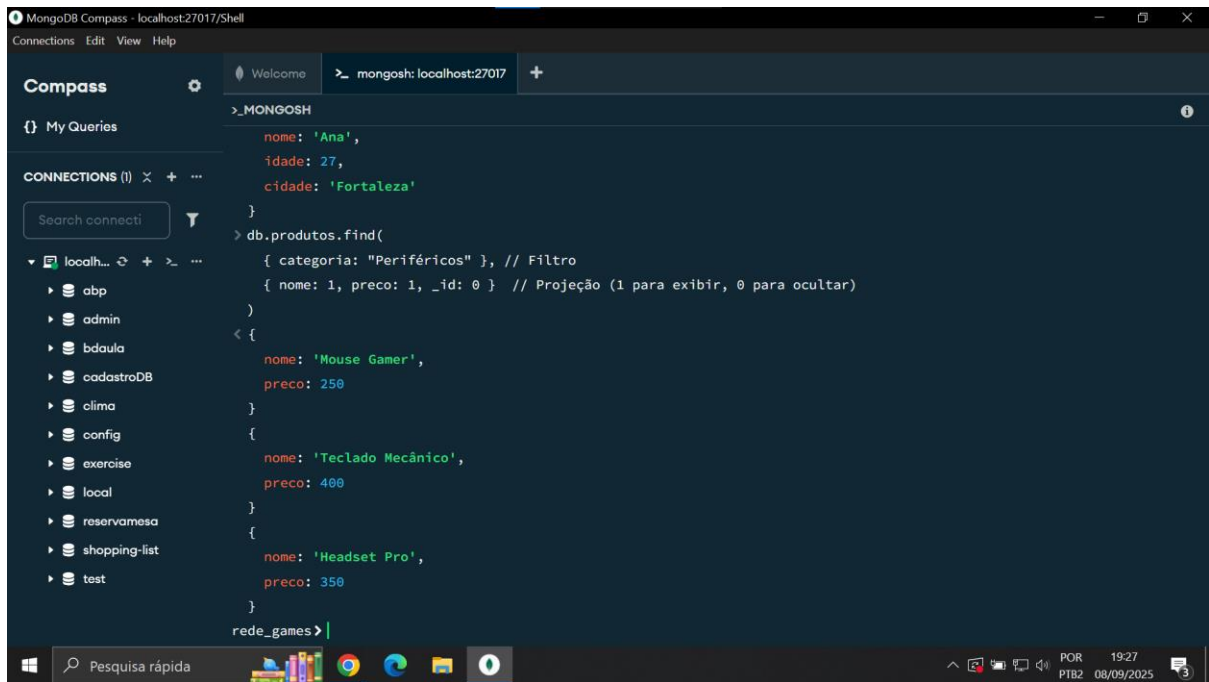
### Questão 3 – Consulta com \$and

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



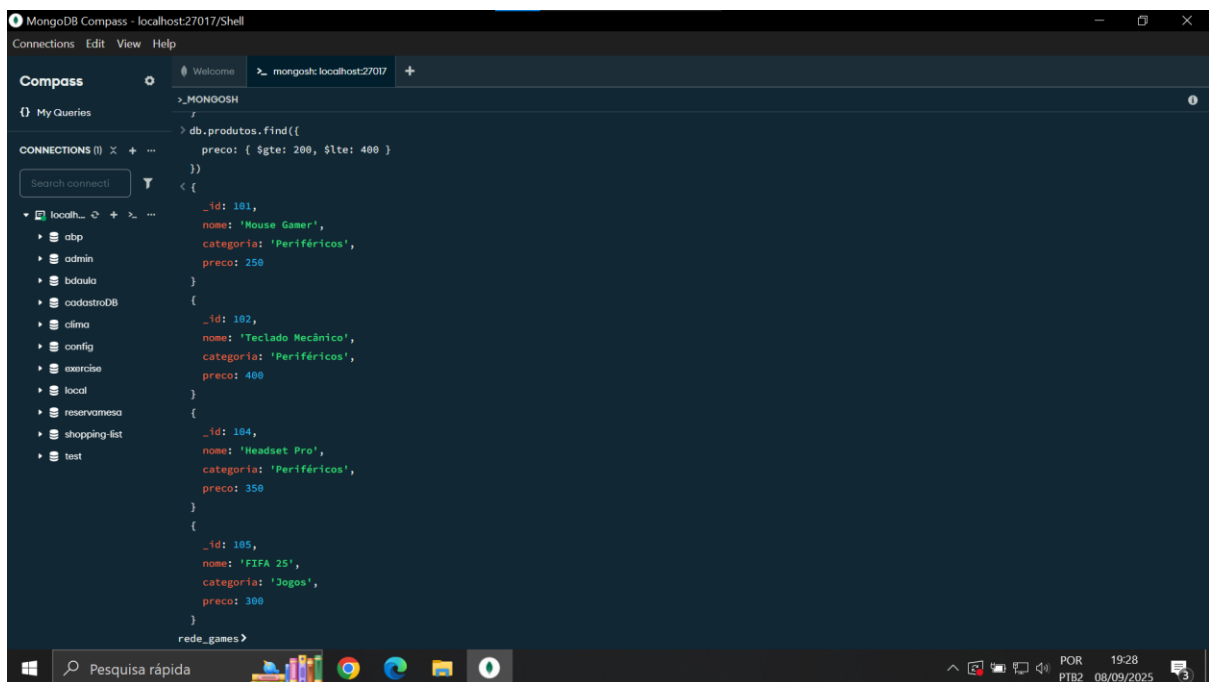
### Questão 4 – Projeção

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



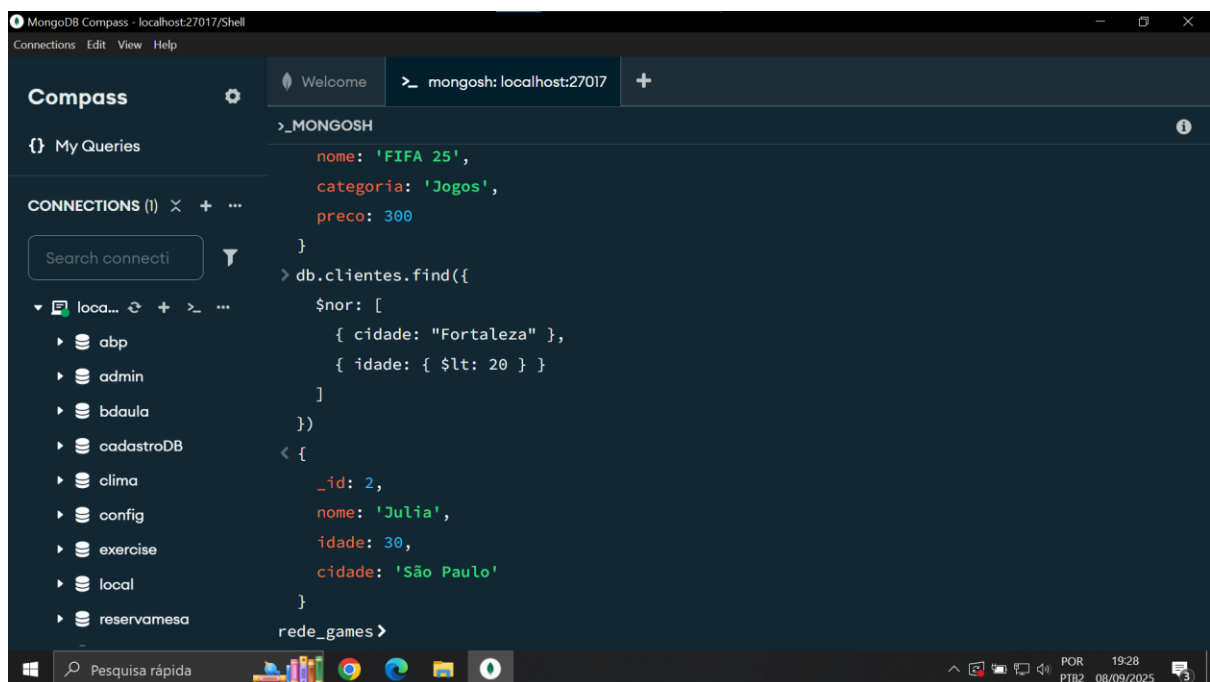
## Questão 5 – Consulta com \$gte e \$lte

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



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

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

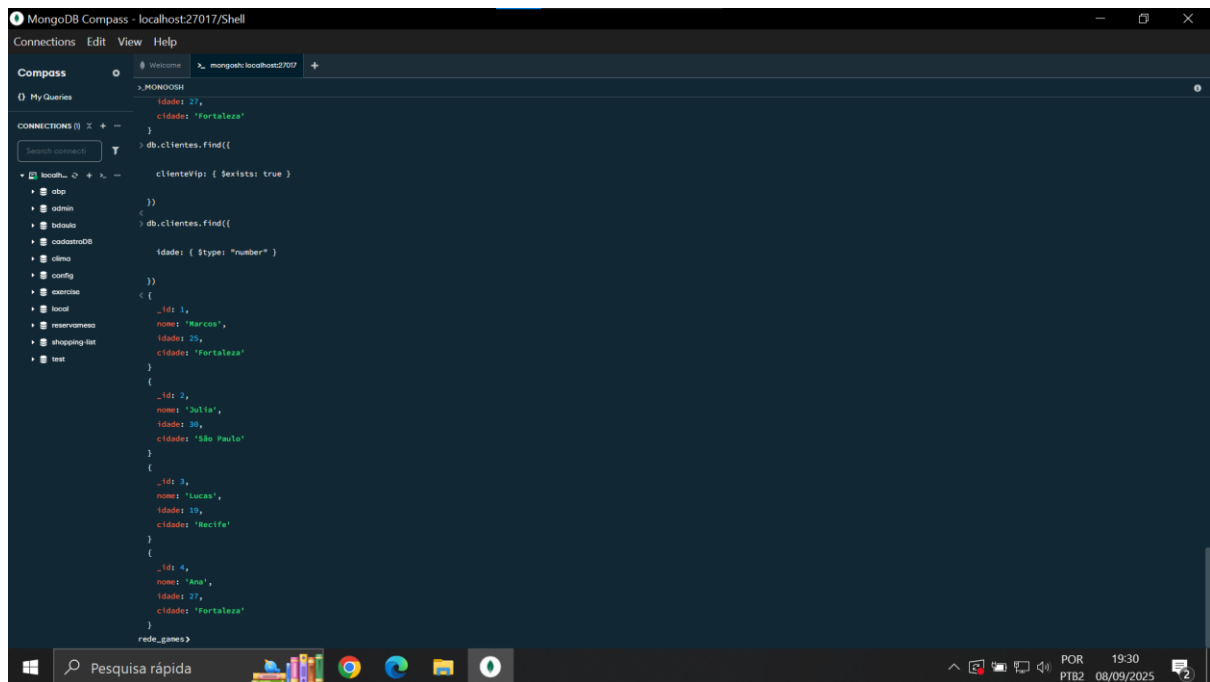


### Questão 7 – Consulta com \$exists e \$type

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

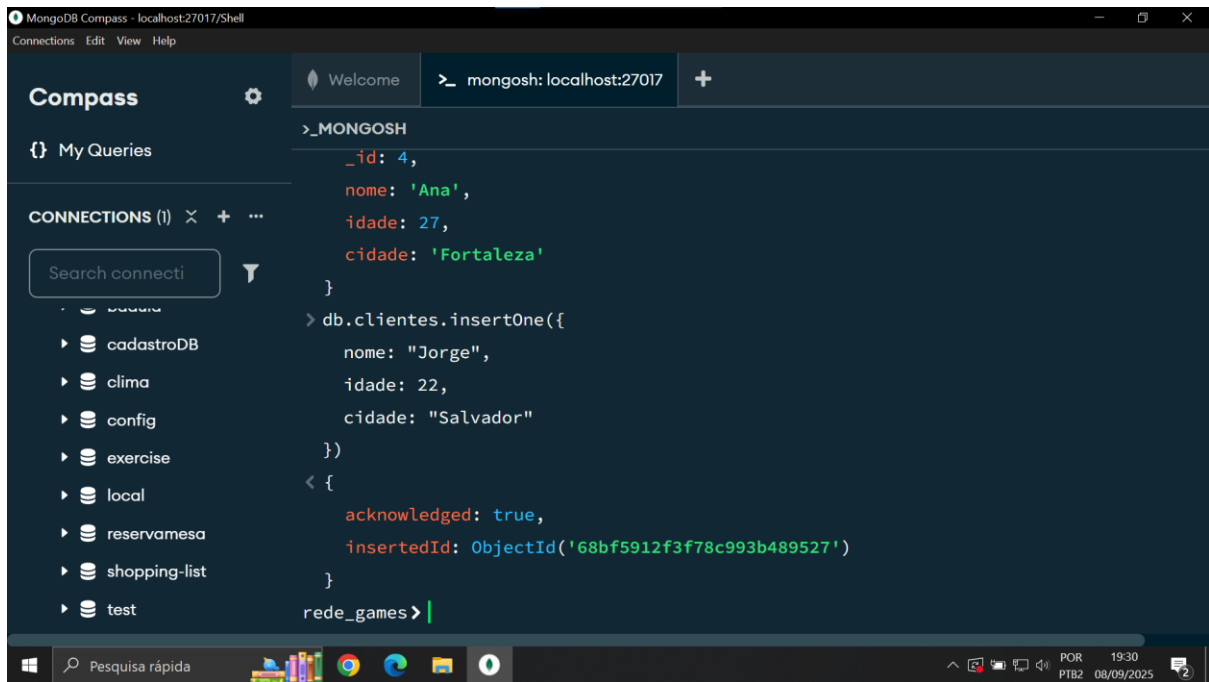
```
db.clientes.find({  
  
  idade: { $type: "number" }  
  
})
```

```
}}
```



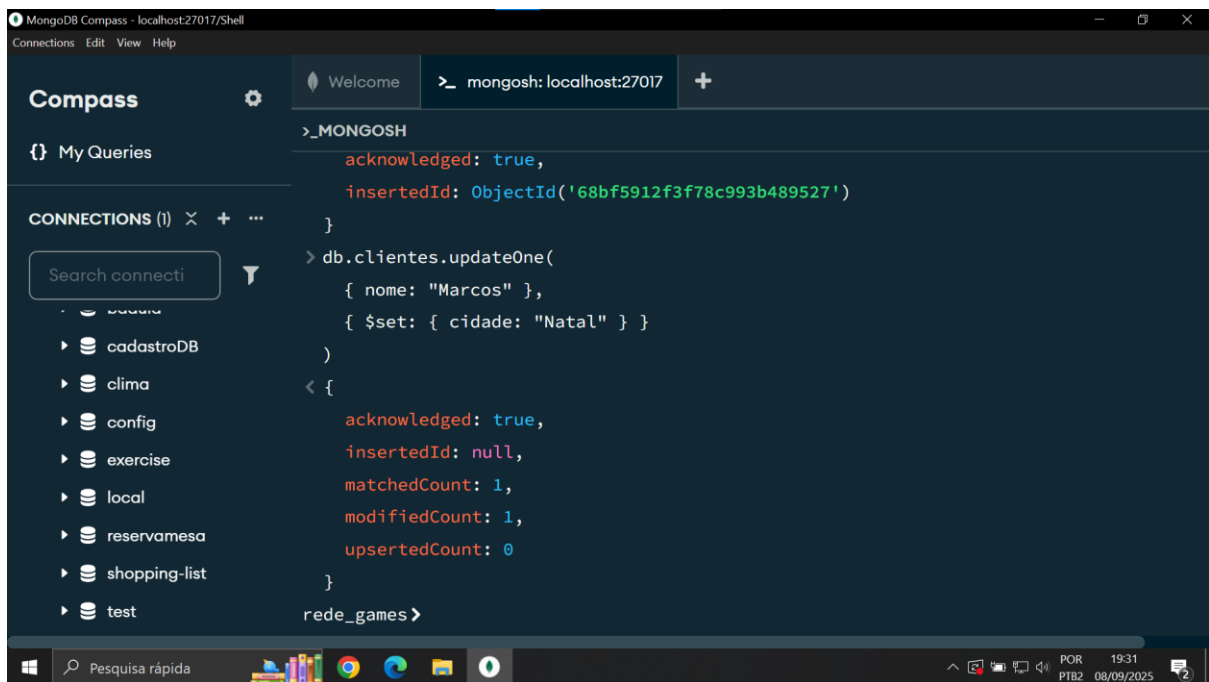
## Questão 8 – Inserção de Novo Cliente

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



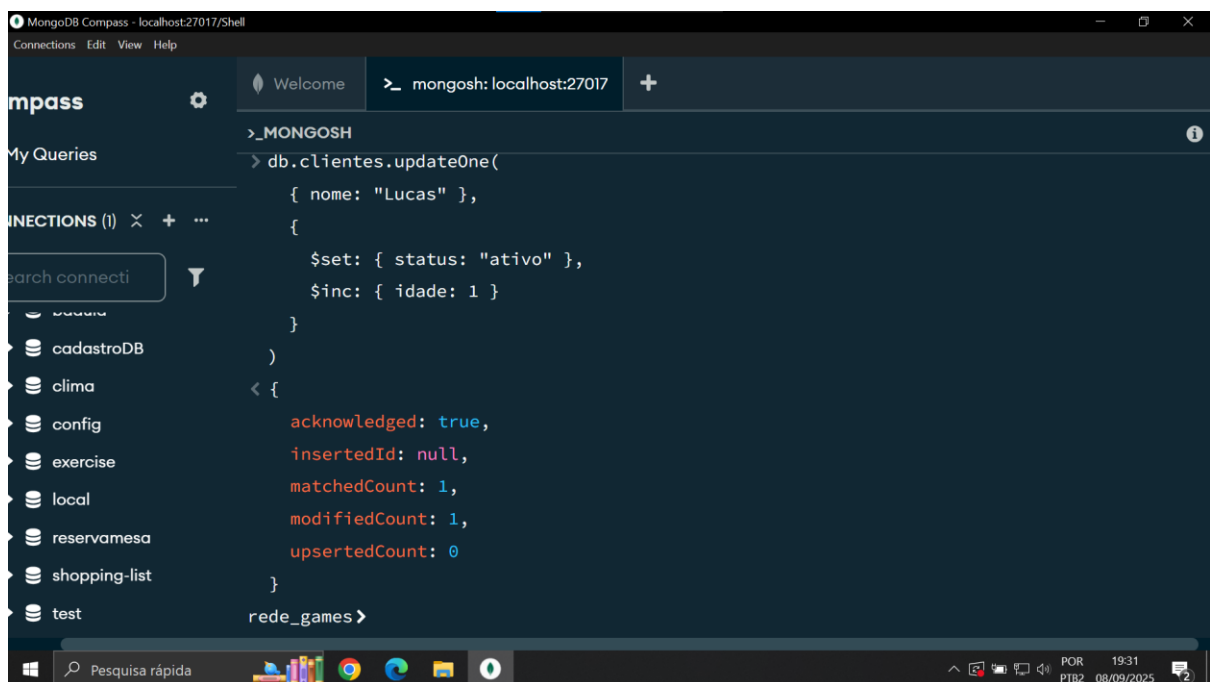
## Questão 9 – Atualização

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



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

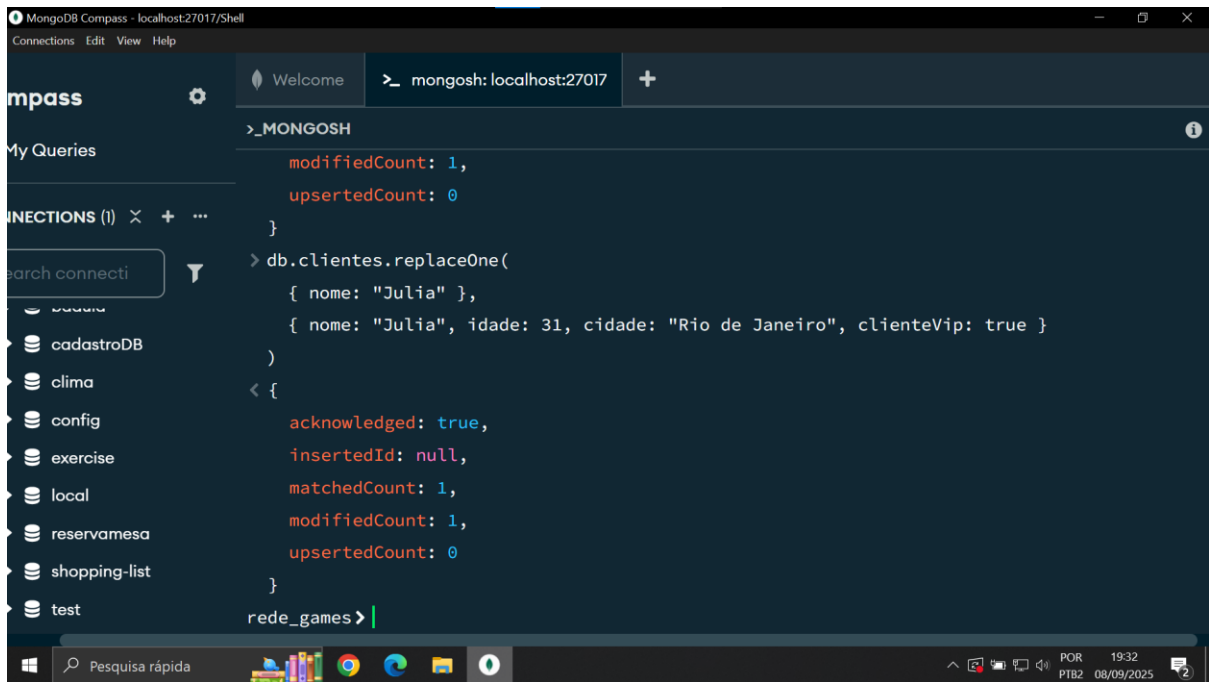
```
db.clientes.updateOne(
  { nome: "Lucas" },
  {
    $set: { status: "ativo" },
    $inc: { idade: 1 }
  }
)
```



## Questão 11 – Atualização com replaceOne

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



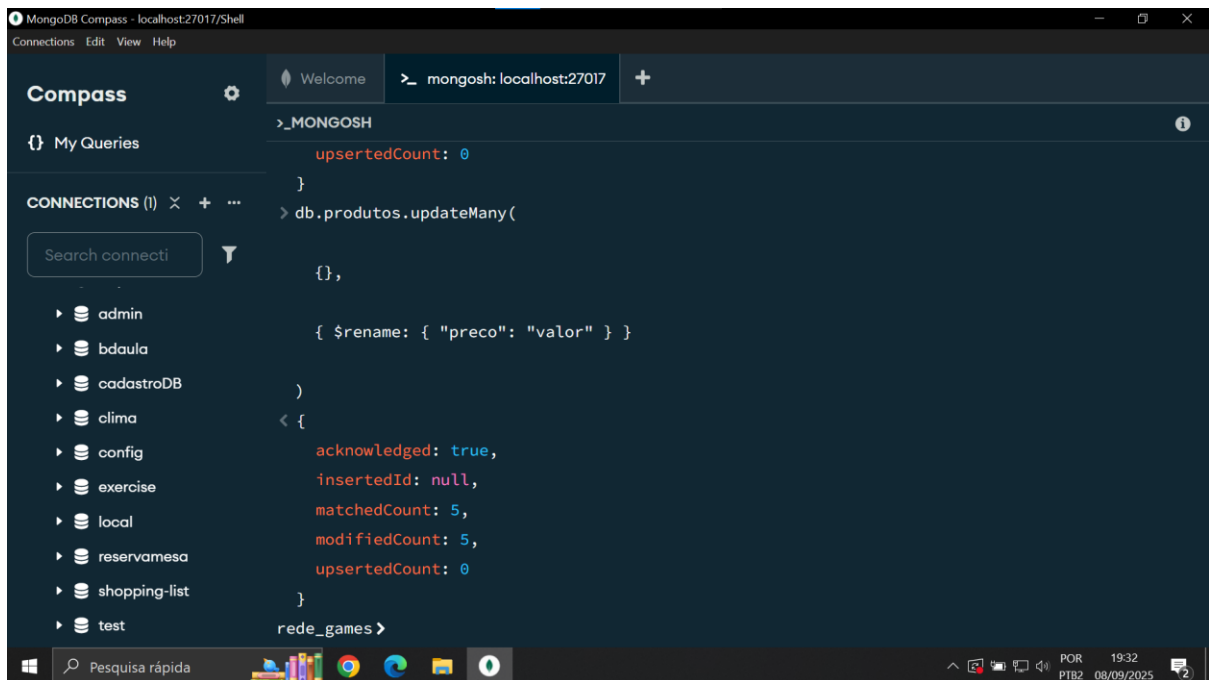


Questão 12 – Atualização com \$rename e \$unset

```

db.produtos.updateMany(
  {},
  { $rename: { "preco": "valor" } }
)

```



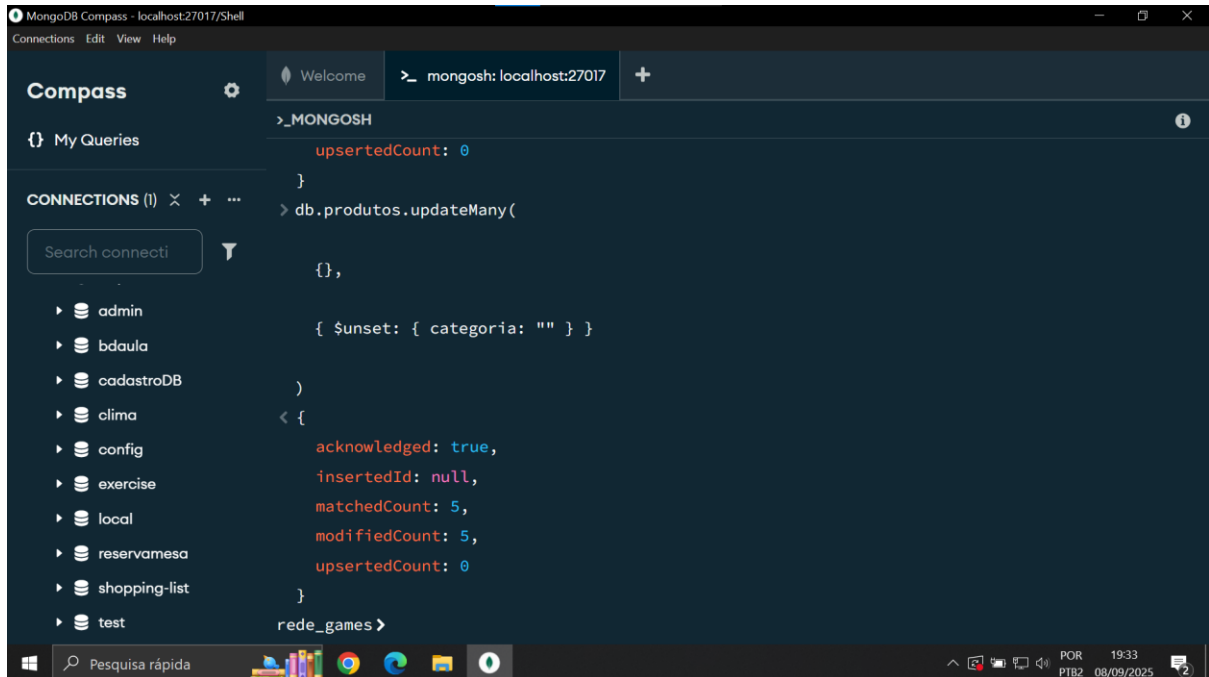
```

db.produtos.updateMany(
  {},

```

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

```
)
```

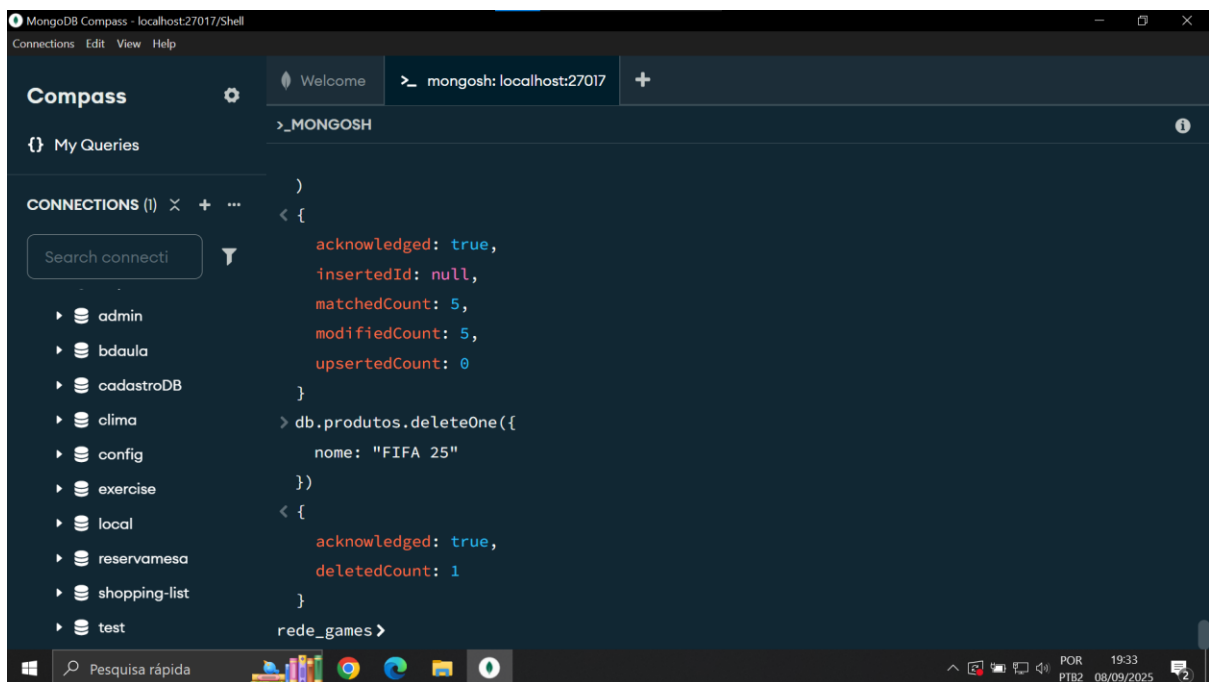


### Questão 13 – Exclusão

```
db.produtos.deleteOne({
```

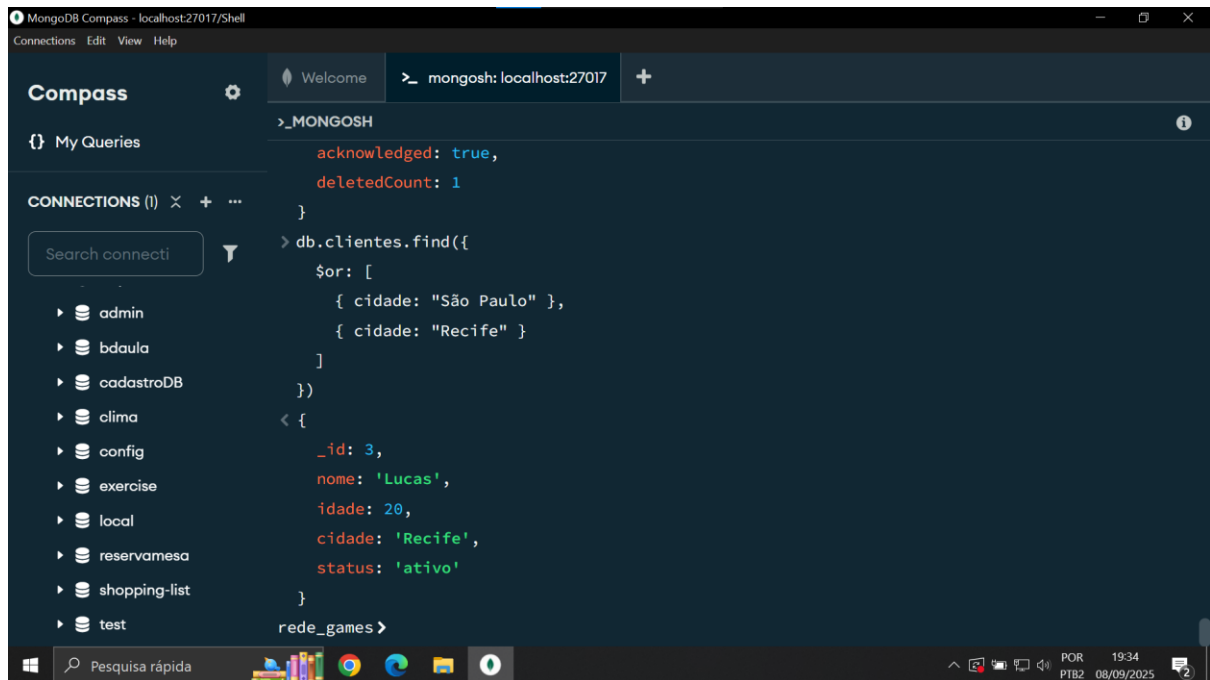
```
  nome: "FIFA 25"
```

```
})
```



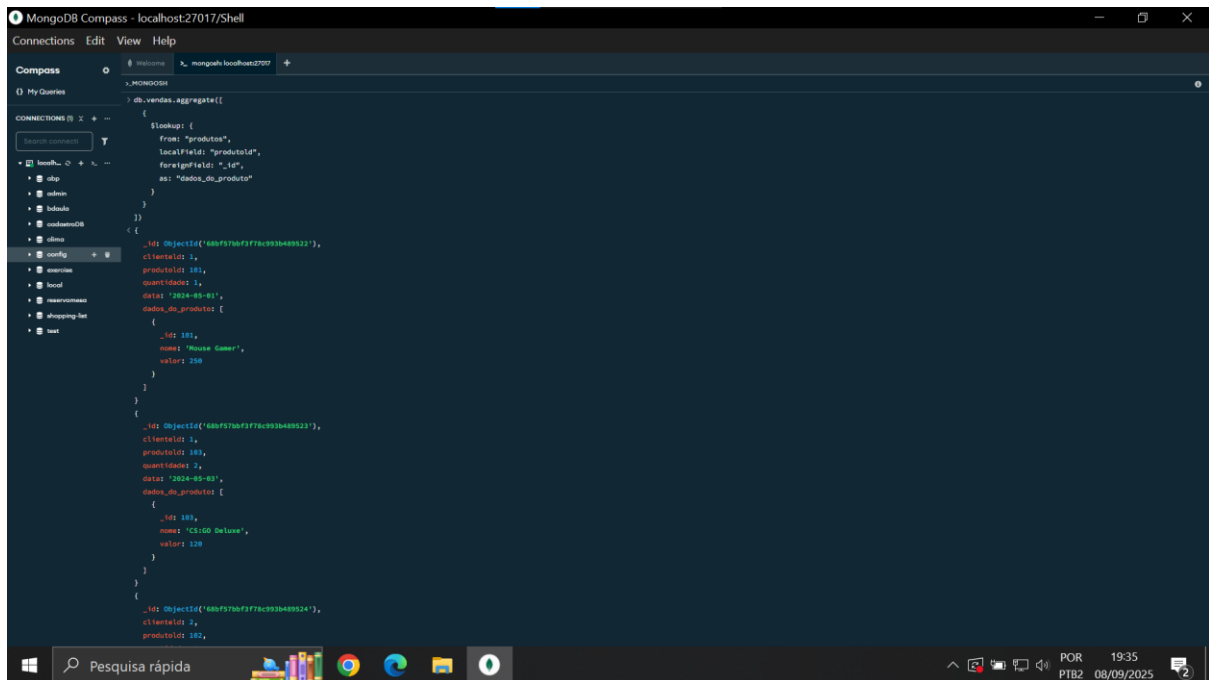
### Questão 14 – Operadores Lógicos

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



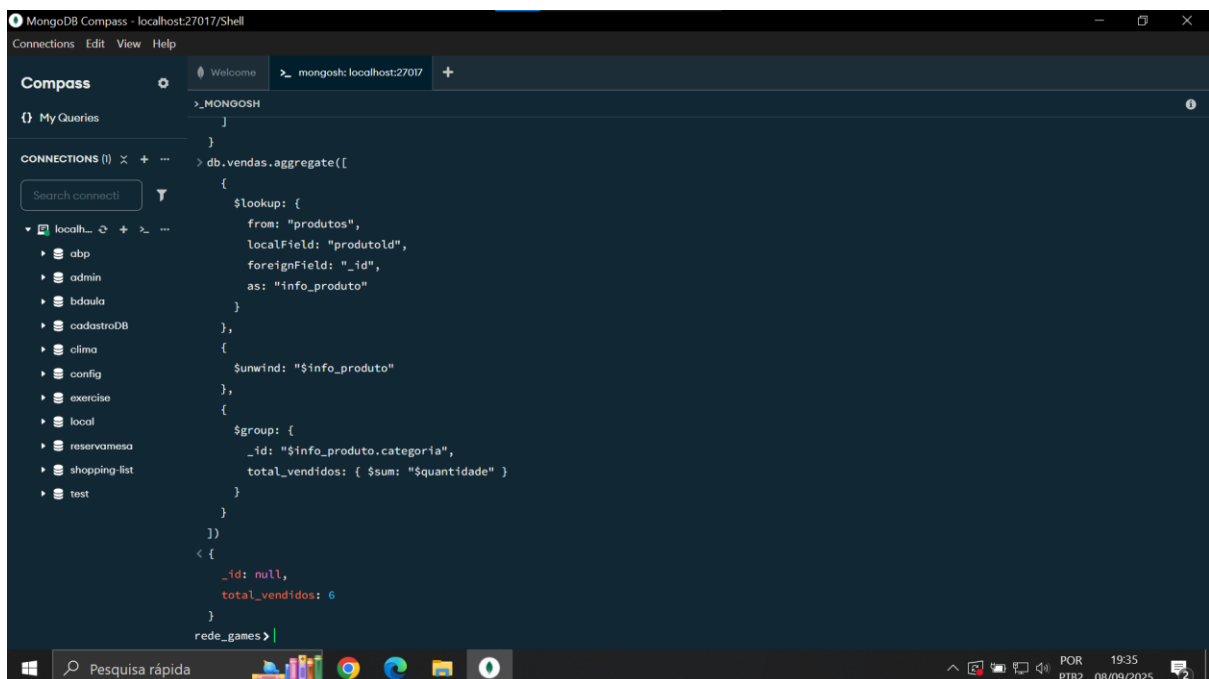
### Questão 15 – Agregação com \$lookup

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



## Questão 16 – Agrupamento

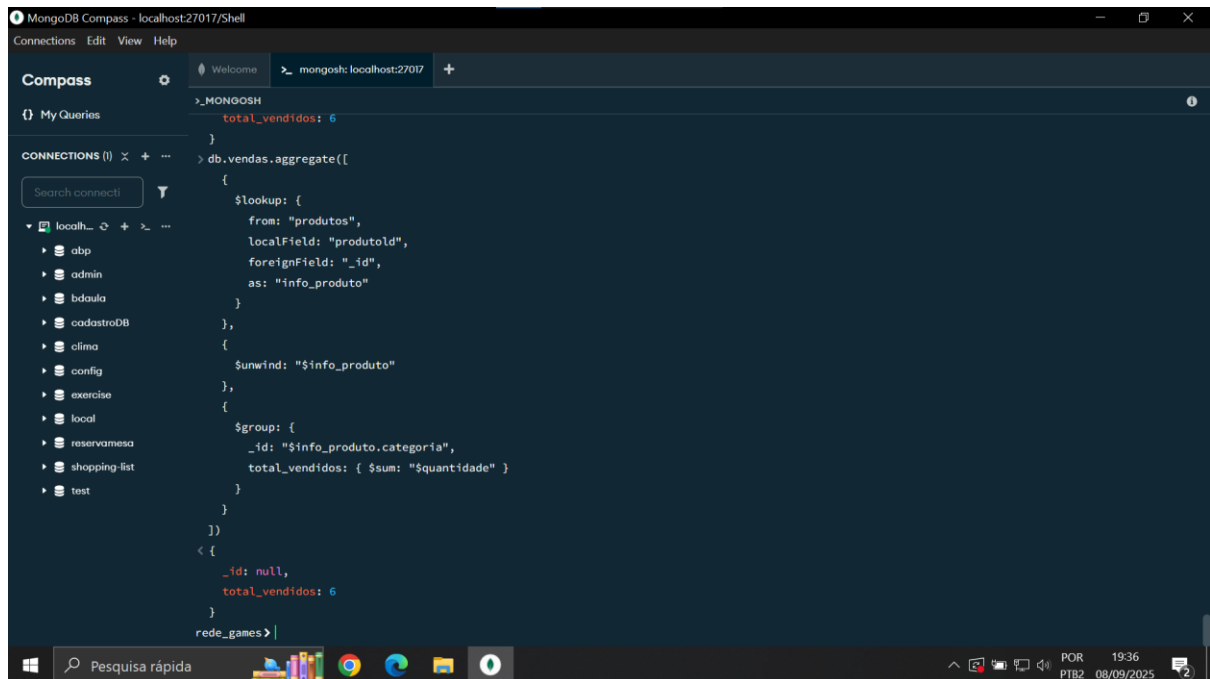
db.vendas.aggregate([ { \$lookup: { from: "produtos", localField: "produtoId", foreignField: "\_id", as: "info\_produto" } }, { \$unwind: "\$info\_produto" }, { \$group: { \_id: "\$info\_produto.categoria", total\_vendidos: { \$sum: "\$quantidade" } } } ])



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

db.vendas.aggregate([ // Junta a coleção 'vendas' com a 'produtos' { \$lookup: { from: "produtos", localField: "produtoId", 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 } } ])

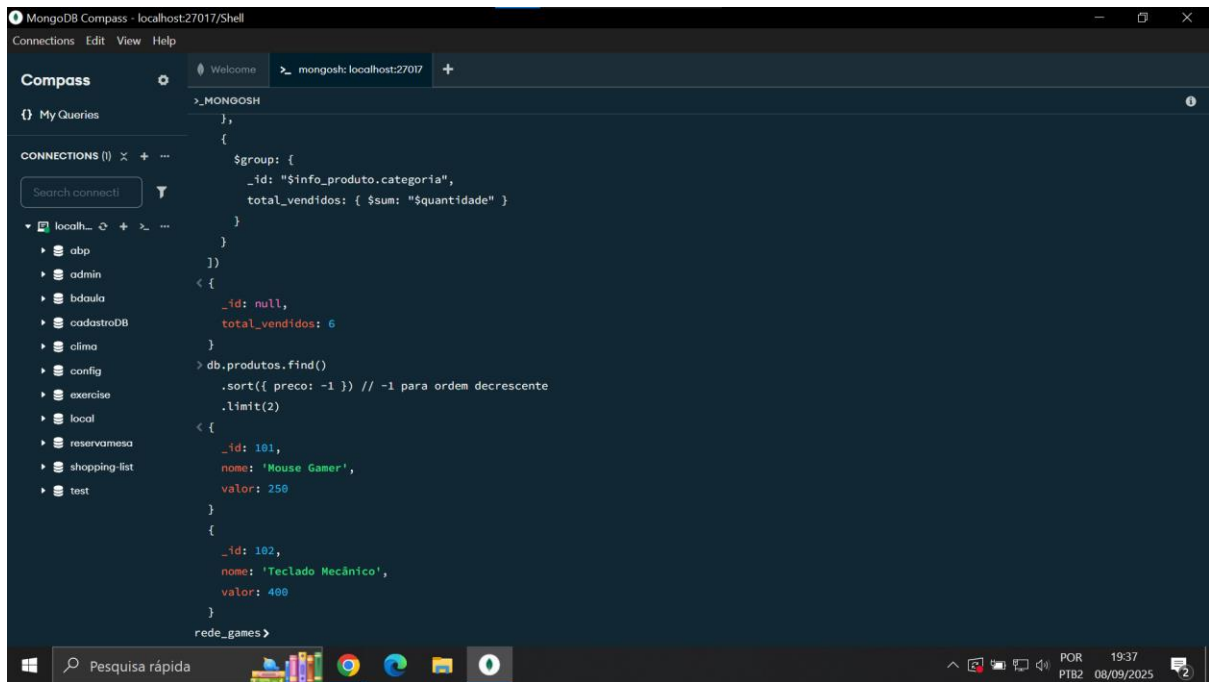


## Questão 18 – Paginação

db.produtos.find()

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

.limit(2)



## 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)

