
Revisão para a Prova I – Banco de Dados Não Relacional – DSM – Profa. Lucineide

Tema: RedeGames – Rede de Lojas de Jogos e eSports

Banco de Dados: rede_games

Questões

Questão 1 – Criação do Banco e Coleções

A empresa RedeGames iniciou a organização do seu banco de dados. O banco **rede_games** com as coleções *clientes*, *produtos* e *vendas*, inserindo os documentos são fornecidos no script inicial.

Resolução esperada: execute o script inicial fornecido.

Script Inicial do Banco de Dados

use rede_games

// Coleção: clientes

```
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" }  
  
])
```

// Coleção: produtos

```
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 },  
  
])
```

Revisão para a Prova I – Banco de Dados Não Relacional – DSM – Profa. Lucineide

```
{_id: 104, nome: "Headset Pro", categoria: "Periféricos", preco: 350 },  
{_id: 105, nome: "FIFA 25", categoria: "Jogos", preco: 300 }  
}  
  
// Coleção: vendas  
  
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

Liste todos os clientes que moram em **Fortaleza**.

Questão 3 – Consulta com \$and

Liste todos os clientes que moram em **Fortaleza** E possuem idade maior que 25 anos.

Questão 4 – Projeção

Mostre apenas o *nome* e o *preco* dos produtos da categoria **Periféricos**.

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

Liste todos os produtos cujo *preço* esteja entre **200 e 400 reais** (inclusive).

Revisão para a Prova I – Banco de Dados Não Relacional – DSM – Profa. Lucineide**Questão 6 – Consulta com \$nor e \$not**

Liste todos os clientes que **não moram em Fortaleza** e **não possuem idade inferior a 20 anos**.

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

Verifique se há clientes com o campo *clienteVip*. Depois, liste apenas os documentos em que o campo *idade* seja do tipo **number**.

Questão 8 – Inserção de Novo Cliente

Adicione um novo *cliente* chamado **Jorge**, 22 anos, morador de Salvador.

Questão 9 – Atualização

O cliente **Marcos** mudou-se para **Natal**. Atualize sua cidade.

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

Atualize o cliente chamado **Lucas** para incluir o campo status: "ativo". Em seguida, incremente sua idade em +1 ano.

Questão 11 – Atualização com replaceOne

Substitua completamente o documento da cliente **Julia** por um novo documento:

```
{ nome: "Julia", idade: 31, cidade: "Rio de Janeiro", clienteVip: true }
```

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

Renomeie o campo *preco* para valor em todos os produtos. Depois, remova o campo *categoria* dos documentos.

Questão 13 – Exclusão

Remova o produto **FIFA 25** do catálogo.

Revisão para a Prova I – Banco de Dados Não Relacional – DSM – Profa. Lucineide

Questão 14 – Operadores Lógicos

Liste os *clientes* que são de **São Paulo** OU de **Recife**.

Questão 15 – Agregação com \$lookup

Liste as *vendas* com os dados completos do produto comprado.

Questão 16 – Agrupamento

Calcule o total de produtos *vendidos* por *categoria*.

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

Liste as *vendas* e calcule o **total de produtos vendidos por categoria**.

Questão 18 – Paginação

Liste os *produtos* ordenados pelo *preço* em ordem *decrescente*, exibindo apenas os **2 produtos mais caros**.

Questão 19 – Paginação com skip

Liste os *produtos* ordenados pelo *valor* (antes chamado de *preço*) em ordem *decrescente*, exibindo apenas do **3º** ao **5º produto mais caro**.
