

Em um documento de texto, coloque seu nome e matrícula, transcreva os enunciados, desenvolva *individualmente* as respostas dos exercícios 1 e 2, e entregue o arquivo em formato PDF no link de envio da APS localizado no card "AV2" no AVA, até o dia **06/11/2024**.

*Leia as instruções antes de cada exercício.*

### **Exercício de referência: Sistema de Gestão de Pedidos e Produtos**

**(Valor: Sem ponto)**

#### **INSTRUÇÕES:**

- Esse exercício não vale ponto, sua finalidade é ajudar o aluno no estudo do assunto e no desenvolvimento dos exercícios.

#### **Descrição do caso de uso**

Você está desenvolvendo um sistema de gestão de pedidos para uma pequena loja online. O sistema armazena pedidos de clientes, e cada pedido contém informações sobre os produtos que foram comprados. O objetivo é manter um registro das compras, com a possibilidade de consultar e atualizar informações tanto dos pedidos quanto dos produtos.

#### **1. Elaborar o Modelo Integrado**

No modelo integrado, os detalhes dos produtos serão incorporados diretamente dentro dos documentos de pedidos. Ou seja, os produtos serão armazenados junto com cada pedido.

##### **1.1. Coleção "pedidos" (Modelo Integrado)**

```
[
  {
    "_id": ObjectId("650a6b1a56e2e63a606e29a1"),
    "nome_cliente": "Alice",
    "data_pedido": "2024-08-20",
    "produtos": [
      { "nome": "Laptop", "preco": 1500, "quantidade": 1 },
      { "nome": "Mouse", "preco": 50, "quantidade": 2 }
    ]
  },
]
```

```
{
  "_id": ObjectId("650a6b1a56e2e63a606e29a2"),
  "nome_cliente": "Bob",
  "data_pedido": "2024-08-21",
  "produtos": [
    { "nome": "Teclado", "preco": 100, "quantidade": 1 }
  ]
}
```

## 2. Elaborar o Modelo Normalizado

No modelo normalizado, os pedidos e produtos são armazenados em coleções separadas. Os pedidos referenciam os produtos por seus IDs.

### 2.1. Coleção pedidos (Modelo Normalizado)

```
[
  {
    "_id": ObjectId("650a6b1a56e2e63a606e29a1"),
    "nome_cliente": "Alice",
    "data_pedido": "2024-08-20",
    "ids_produtos": [
      ObjectId("650a6b1a56e2e63a606e29b1"),
      ObjectId("650a6b1a56e2e63a606e29b2")
    ]
  },
  {
    "_id": ObjectId("650a6b1a56e2e63a606e29a2"),
    "nome_cliente": "Bob",
    "data_pedido": "2024-08-21",
    "ids_produtos": [
      ObjectId("650a6b1a56e2e63a606e29b3")
    ]
  }
]
```

### 2.2. Coleção produtos (Modelo Normalizado)

```
[
  {
    "_id": ObjectId("650a6b1a56e2e63a606e29b1"),
    "nome": "Laptop",
    "preco": 1500
  },
  {
    "_id": ObjectId("650a6b1a56e2e63a606e29b2"),
    "nome": "Mouse",
    "preco": 50
  },
  {
    "_id": ObjectId("650a6b1a56e2e63a606e29b3"),
    "nome": "Teclado",
  }
]
```

```
    "preço": 100
  }
]
```

### 3. Realizar consultas

#### 3.1. Consulta 1

##### 3.1.1. Serve para os dois Modelos

Buscar todos os pedidos de "Alice".

```
db.collection.find({ "nome_cliente": "Alice" })
```

#### 3.2. Consulta 2

##### 3.2.1. Modelo Integrado

Buscar pedidos que contêm o produto "Mouse".

```
db.collection.find({ "produtos.nome": "Mouse" })
```

##### 3.2.2. Modelo Normalizado

Buscar pedidos que referenciam o produto "Mouse". Para isso, você precisaria primeiro obter o `ObjectId` do produto "Mouse" na coleção produtos.

```
db.produtos.find({ "nome": "Mouse" })
```

Resultado: `ObjectId("650a6b1a56e2e63a606e29b2")`

Depois, busque os pedidos que contêm este ID:

```
db.pedidos.find({ "ids_produtos":
ObjectId("650a6b1a56e2e63a606e29b2") })
```

### 4. Realizar atualizações

#### 4.1. Atualização 1

##### 4.1.1. Serve para Ambos os Modelos, mudar o nome da coleção no modelo normalizado.

Atualizar o nome do cliente "Alice" para "Alice Cooper".

```
db.collection.update(  
  { "nome_cliente": "Alice" },  
  { $set: { "nome_cliente": "Alice Cooper" } }  
)
```

## 4.2. Atualização 2

### 4.2.1. Modelo Integrado

Atualizar o preço do "Mouse" para 55.

```
db.collection.update(  
  { "produtos.nome": "Mouse" },  
  { $set: { "produtos.preco": 55 } }  
)
```

### 4.2.2. Modelo Normalizado:

Atualizar o preço do "Mouse" para 55.

```
db.produtos.update(  
  { "nome": "Mouse" },  
  { $set: { "preco": 55 } }  
)
```

## 5. Realizar Consultas de Agregação

### 5.1. Agregação 1

#### 5.1.1. Modelo Integrado

Calcular o valor total de cada pedido (soma dos produtos).

```
db.collection.aggregate([  
  { $unwind: "$produtos" },  
  { $group: { _id: "$_id", totalAmount: { $sum: { $multiply:  
    ["$produtos.preco", "$produtos.quantidade"] } } } }  
])
```

#### 5.1.2. Modelo Normalizado

Calcular o valor total de cada pedido (soma dos produtos).

```
db.pedidos.aggregate([
  { $lookup: { from: "produtos", localField: "ids_produtos",
    foreignField: "_id", as: "product_details" } },
  { $unwind: "$product_details" },
  { $group: { _id: "$_id", totalAmount: { $sum:
    "$product_details.preco" } } }
])
```

## 5.2. Agregação 2

### 5.2.1. Modelo Integrado

Contar quantos pedidos têm mais de 1 produto.

```
db.collection.aggregate([
  { $project: { productCount: { $size: "$produtos" } } },
  { $match: { productCount: { $gt: 1 } } }
])
```

### 5.2.2. Modelo Normalizado

Contar quantos pedidos têm mais de 1 produto.

```
db.pedidos.aggregate([
  { $project: { productCount: { $size: "$ids_produtos" } } },
  { $match: { productCount: { $gt: 1 } } }
])
```

## Observações

1. *Modelo Integrado:* As consultas e atualizações que envolvem os detalhes dos produtos são realizadas diretamente dentro dos documentos de pedidos. Isso torna o acesso mais rápido, mas a atualização de produtos em todos os pedidos pode ser mais complexa, pois requer a atualização de vários documentos.
2. *Modelo Normalizado:* No modelo normalizado, os produtos são referenciados por IDs. As consultas que envolvem produtos exigem o uso do operador `$lookup` para buscar os detalhes dos produtos. O operador `$lookup` é usado para **fazer uma junção (join)** entre duas coleções no MongoDB.

## Exercício 1: Sistema de Gestão de Biblioteca

(Valor: 1,0 pt.)

### INSTRUÇÕES:

- As respostas de cada subtópico deverão ser entregues na forma **de texto ou na forma de capturas legíveis das telas (*printscreens*)** do Mongo Playground.

### Descrição do caso de uso

Você está desenvolvendo um sistema de gestão para uma biblioteca. O sistema precisa armazenar informações sobre livros e empréstimos. Um livro pode ser emprestado várias vezes, e o objetivo é registrar quais livros foram emprestados e quem são os responsáveis por esses empréstimos.

#### 1. Elaborar o Modelo Integrado

No modelo integrado, os detalhes dos livros estarão armazenados diretamente dentro dos documentos de empréstimos.

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

#### 2. Elaborar o Modelo Normalizado

No modelo normalizado, os empréstimos e livros são armazenados em coleções separadas, e os empréstimos referenciam os livros por seus IDs.

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

#### 3. Realizar consultas

##### 3.1. Consulta 1

3.1.1. Serve para os dois Modelos

Buscar todos os empréstimos de "Alice".

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

##### 3.1.2. Modelo Integrado

Buscar empréstimos que contêm o livro "1984".

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

### **3.2. Consulta 2**

#### **3.2.1. Modelo Normalizado**

Buscar empréstimos que referenciam o livro "1984".

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

### **4. Realizar Atualizações**

#### **4.1. Atualização 1**

##### **4.1.1. Serve para os dois Modelos**

Atualizar o nome do responsável "Alice" para "Alice Cooper".

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

##### **4.1.2. Modelo Integrado**

Atualizar o autor de "1984" para "G. Orwell".

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

#### **4.2. Atualização 2**

##### **4.2.1. Modelo Normalizado**

Atualizar o autor de "1984" para "G. Orwell"

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

### **5. Realizar Consultas de Agregação**

#### **5.1. Agregação 1**

##### **5.1.1. Modelo Integrado**

Contar quantos livros foram emprestados em cada empréstimo.

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

### 5.1.2. Modelo Normalizado

Contar quantos livros foram emprestados em cada empréstimo.

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

## 5.2. Agregação 2

### 5.2.1. Modelo Integrado

Listar os empréstimos que contêm mais de 1 livro.

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

### 5.2.2. Modelo Normalizado

Listar os empréstimos que contêm mais de 1 livro.

(Responda na forma de texto **ou** na forma de captura legível da tela (*printscreen*) do Mongo Playground).

## Exercício 2: Sistema de Gerenciamento de Consultas Médicas

(Valor: 1,0 pt.)

### INSTRUÇÕES:

- Substituir **<seunome>** pelo seu nome real;
- As respostas de cada subtópico deverão ser entregues **na forma de capturas legíveis das telas (*printscreens*) do Mongo Playground.**

### Descrição do caso de uso

Você está desenvolvendo um sistema para gerenciar consultas médicas em uma clínica. O sistema precisa armazenar informações sobre médicos e consultas. Cada consulta é associada a um médico, e o sistema deve permitir consultar e atualizar dados das consultas e dos médicos.

### 1. Elaborar o Modelo Integrado

(Responda na forma de captura legível da tela (*printscreen*) do Mongo Playground).

### 2. Elaborar o Modelo Normalizado



(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).

### 3. Realizar Consultas

- Buscar todas as consultas do paciente "Alice" (para os dois modelos)  
(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).
- Buscar consultas associadas ao médico "<seunome>" (Modelo Integrado)  
(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).
- Buscar consultas associadas ao médico "<seunome>" (Modelo Normalizado)  
(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).

### 4. Realizar Atualizações

- Atualizar a data da consulta de "Alice" para "15-10-2024" (para os dois modelos)  
(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).
- Atualizar a especialidade do "<seunome>" para "Cardiologista" (Modelo Integrado)  
(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).
- Atualizar a especialidade do "<seunome>" para "Cardiologista" (Modelo Normalizado)  
(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).

### 5. Realizar Consultas de Agregação

- Contar quantas consultas cada médico realizou (Modelo Integrado)  
(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).
- Contar quantas consultas cada médico realizou (Modelo Normalizado)  
(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).
- Listar consultas realizadas por "Dr. Smith" (Modelo Integrado)  
(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).
- Listar consultas realizadas por "Dr. Smith" (Modelo Normalizado)  
(Responda na forma captura legível da tela (*printscreen*) do Mongo Playground).