



## 5.4. Comparação Prática de Instruções em SQL (Oracle) x Instruções em NoSQL (MongoDB)

### 5.4.1 Criação, Alteração e Exclusão de Tabelas e Coleções

#### Criação

SQL em ORACLE	NoSQL em MongoDB
<pre>create table produto (codigo number(5), descricao varchar2(50), valor number(9,2), data_vencimento date);  insert into produto values(1,'Arroz Tio João',18.45, '10/04/2021');  insert into produto values(2,'Feijão Douradinho',7.50, '10/04/2021');</pre>	<pre>db.createCollection("produto")  db.produto.insertMany([ { codigo: 1, descricao: "Arroz Tio João", valor: 18.45, data_vencimento: "10/04/2021" }, { codigo: 2, descricao: "Feijão Douradinho", valor: 7.50, data_vencimento: "10/04/2021" }, ])</pre>

#### Alteração

SQL em ORACLE	NoSQL em MongoDB
<pre>alter table produto add estoque number(6,2);</pre>	<pre>db.produto.insert( { codigo: 3, descricao: "Arroz Tio João", valor: 18.45, data_vencimento: "10/04/2021", estoque: 35 } )</pre>

#### Exclusão

SQL em ORACLE	NoSQL em MongoDB
<pre>drop table produto;</pre>	<pre>db.produto.drop();</pre>



## Disciplina BANCO DE DADOS II

### 5.4.2. Inserção, Consulta e Alteração de Dados em Tabelas e em Coleções

#### Inserção

SQL em ORACLE	NoSQL em MongoDB
<pre>insert into produto values(4,'Café Cabloco',8.99, '10/04/2021',25);  insert into produto values(5,'Batata Bola',3.16, '10/04/2021',10);  insert into produto values(6,'Goiaba Vermelha',3.55, '10/04/2021',100);</pre>	<pre>db.produto.insertMany([ { codigo: 4, descricao: "Café Cabloco", valor: 8.99, data_vencimento: "10/04/2021", estoque: 25 }, { codigo: 5, descricao: "Batata Bola", valor: 3.16, data_vencimento: "10/04/2021", estoque: 10 }, ])  db.produto.insert( { codigo: 6, descricao: "Goiaba Vermelha", valor: 3.55, data_vencimento: "10/04/2021", estoque: 100 } )</pre>

#### Alteração

SQL em ORACLE	NoSQL em MongoDB
<pre>update produto set estoque = 0 where codigo&lt;5;  update produto set valor = valor * 1.50;  update produto set valor = valor + 50;</pre>	<pre><b>SET</b> - troca db.produto.updateMany( { codigo: { \$lt: 5 } }, { \$set: { estoque: 0 } } )  <b>MUL</b> = Multiplica db.produto.updateMany( { }, { \$mul: { valor: 1.50 } } )  <b>INC</b> = Soma db.produto.updateMany( { estoque: { \$eq: 0 } }, { \$inc: { valor: 50 } } )</pre>



## Disciplina BANCO DE DADOS II



### Exclusão

SQL em ORACLE	NoSQL em MongoDB
<code>delete from produto where codigo=5;</code>	<code>db.produto.deleteMany( { codigo: 5 } )</code>
<code>delete from produto;</code>	<code>db.produto.deleteMany( {} )</code>

### Consulta

SQL em ORACLE	NoSQL em MongoDB
<code>select *</code> <code>from produto;</code>	<code>db.produto.find()</code>
<code>select codigo, descricao</code> <code>from produto;</code>	<code>db.produto.find(</code> <code>{ },</code> <code>{ codigo: 1, descricao: 1 }</code> <code>)</code>
<code>select codigo, descricao, valor</code> <code>from produto</code> <code>where valor&gt;5;</code>	<code>db.produto.find(</code> <code>{ valor: { \$gt: 5 } },</code> <code>{ codigo: 1, descricao: 1, valor: 1 }</code> <code>)</code>
<code>select count(valor)</code> <code>from produto;</code>	<code>db.produto.count()</code>

### 5.4.3. Operações em MongoDB

#### 5.4.3.1. Inserir Documento

Se a coleção não existir atualmente, as operações de inserção criarão a coleção.

#### Insira um único documento

`db.collection.insertOne()` insere um *único documento* em uma coleção.



## Disciplina BANCO DE DADOS II

O exemplo a seguir insere um novo documento na coleção. Se o documento não especificar um campo, o MongoDB adicionará o campo com um valor ObjectId ao novo documento.

```
db.inventory.insertOne(  
  { item: "canvas", qty: 100, tags: ["cotton"], size: { h: 28, w: 35.5, uom: "cm" } }  
)
```

### Insira vários documentos

`db.collection.insertMany()` pode inserir *vários documentos* em uma coleção. Passe uma série de documentos para o método.

O exemplo a seguir insere três novos documentos na coleção. Se os documentos não especificarem um campo, o MongoDB adicionará o campo com um valor ObjectId a cada documento.

```
db.inventory.insertMany([  
  { item: "journal", qty: 25, tags: ["blank", "red"], size: { h: 14, w: 21, uom: "cm" } },  
  { item: "mat", qty: 85, tags: ["gray"], size: { h: 27.9, w: 35.5, uom: "cm" } },  
  { item: "mousepad", qty: 25, tags: ["gel", "blue"], size: { h: 19, w: 22.85, uom: "cm" } }  
)
```

### 5.4.3.2. Atualizar Documento

#### Atualiza um único documento

O método `db.collection.updateOne()` atualizar o *primeiro* documento onde for igual: `inventoryitem"paper"`

```
db.inventory.updateOne(  
  { item: "paper" },  
  {  
    $set: { "size.uom": "cm", status: "P" },  
    $currentDate: { lastModified: true }  
  }  
)
```

#### Atualiza vários documentos

O exemplo a seguir usa o método `db.collection.updateMany()` na coleção para atualizar todos os documentos onde é menor que: `inventory qty 50`



## Disciplina BANCO DE DADOS II

```
db.inventory.updateMany(  
  { "qty": { $lt: 50 } },  
  {  
    $set: { "size.uom": "in", status: "P" },  
    $currentDate: { lastModified: true }  
  }  
)
```

### Substitua um documento

O exemplo a seguir substitui o *primeiro* documento da coleção onde: inventory item: "paper"

```
db.inventory.replaceOne(  
  { item: "paper" },  
  { item: "paper", instock: [ { warehouse: "A", qty: 60 }, { warehouse: "B", qty: 40 } ] }  
)
```

### 5.4.3.3. Excluir Documento

#### Exclua todos os documentos

O exemplo a seguir exclui *todos* os documentos da coleção: inventory

```
db.inventory.deleteMany({})
```

#### Exclua todos os documentos que correspondam a uma condição

O exemplo a seguir remove todos os documentos da coleção em que o campo é igual: inventory status "A"

```
db.inventory.deleteMany({ status : "A" })
```

#### Exclua apenas um o documento que corresponda a uma condição

O exemplo a seguir exclui o *primeiro* documento onde está: status "D"

```
db.inventory.deleteOne( { status: "D" } )
```



## Disciplina BANCO DE DADOS II

### 5.4.3.4. Selecionar Documento

#### Selecione todos os documentos de uma coleção

```
db.inventory.find( {} )
```

#### Especificar condição de igualdade para selecionar documentos

O exemplo a seguir seleciona a partir da coleção todos os documentos em que os iguais: inventory status "D"

```
db.inventory.find( { status: "D" } )
```

O exemplo a seguir recupera todos os documentos da coleção em que iguais ou: inventory status "A" "D"

```
db.inventory.find( { status: { $in: [ "A", "D" ] } } )
```

#### Especificar condições AND

O exemplo a seguir recupera todos os documentos da coleção em que os iguais e menores que (\$lt): inventory status "A" qty 30

```
db.inventory.find( { status: "A", qty: { $lt: 30 } } )
```

#### Especificar condições OR

O exemplo a seguir recupera todos os documentos da coleção em que os iguais ou menores que (\$lt): status "A" qty 30

```
db.inventory.find( { $or: [ { status: "A" }, { qty: { $lt: 30 } } ] } )
```

#### Especificar condições AND e OR

No exemplo a seguir, o documento de consulta composto seleciona todos os documentos da coleção onde status "A" e os iguais e menores que (\$lt) ou começa com o caractere: qty 30 item p

```
db.inventory.find( {  
  status: "A",  
  $or: [ { qty: { $lt: 30 } }, { item: /^p/ } ]  
} )
```



## Disciplina BANCO DE DADOS II

### Agregação e Agrupamentos

**Primeira Etapa:** O estágio `$match` filtra os documentos pelo campo e passa para a próxima etapa os documentos que têm igual a status "A".

**Segunda Etapa:** O `$group` agrupa os documentos pelo campo para calcular a soma do valor para cada `cust_id`.

```
db.orders.aggregate([
  { $match: { status: "A" } },
  { $group: { _id: "$cust_id", total: { $sum: "$amount" } } }
])
```

### 5.4.3.5. Operadores de Consulta

#### Comparação

Nome	Descrição
<code>\$eq</code>	Corresponde a valores iguais a um valor especificado.
<code>\$gt</code>	Corresponde a valores maiores do que um valor especificado.
<code>\$gte</code>	Corresponde a valores maiores ou iguais a um valor especificado.
<code>\$in</code>	Corresponde a qualquer um dos valores especificados em uma matriz.
<code>\$lt</code>	Corresponde a valores menores a um valor especificado.
<code>\$lte</code>	Corresponde a valores menores ou iguais a um valor especificado.
<code>\$ne</code>	Corresponde a todos os valores que não são iguais a um valor especificado.
<code>\$nin</code>	Corresponde a nenhum dos valores especificados em uma matriz.

#### Lógico

Nome	Descrição
<code>\$and</code>	Junta-se a cláusulas de consulta com retornos lógicos todos os documentos que correspondem às condições de ambas as cláusulas. AND
<code>\$not</code>	Inverte o efeito de uma expressão de consulta e devolve documentos que <i>não</i> correspondem à expressão de consulta.
<code>\$nor</code>	Junta-se a cláusulas de consulta com retornos lógicos todos os documentos que não correspondem a ambas as cláusulas. NOR
<code>\$or</code>	Junta-se a cláusulas de consulta com retornos lógicos todos os documentos que correspondem às condições de qualquer cláusula. OR

#### Elemento

Nome	Descrição
<code>\$exists</code>	Corresponde aos documentos que possuem o campo especificado.
<code>\$type</code>	Seleciona documentos se um campo for do tipo especificado.



## Disciplina BANCO DE DADOS II

### 5.4.3.6. Operadores de Atualização

Nome	Descrição
<code>\$currentDate</code>	Define o valor de um campo para a data atual, seja como uma data ou um Timestamp.
<code>\$inc</code>	Aumenta o valor do campo pelo valor especificado.
<code>\$min</code>	Só atualiza o campo se o valor especificado for menor que o valor de campo existente.
<code>\$max</code>	Só atualiza o campo se o valor especificado for maior do que o valor de campo existente.
<code>\$mul</code>	Multiplica o valor do campo pelo valor especificado.
<code>\$rename</code>	Renomeia um campo.
<code>\$set</code>	Define o valor de um campo em um documento.
<code>\$setOnInsert</code>	Define o valor de um campo se uma atualização resultar em uma inserção de um documento. Não tem efeito nas operações de atualização que modificam documentos existentes.
<code>\$unset</code>	Remove o campo especificado de um documento.

### 5.4.3.7. Operadores de Agrupamento

Nome	Descrição
<code>\$avg</code>	Retorna uma média de valores numéricos. Ignora valores não numéricos.
<code>\$first</code>	Retorna um valor do primeiro documento para cada grupo. A ordem só é definida se os documentos estiverem em uma ordem definida. Diferente do operador de matriz <code>\$first</code> .
<code>\$last</code>	Retorna um valor do último documento para cada grupo. A ordem só é definida se os documentos estiverem em uma ordem definida. Diferente do operador de matriz <code>\$last</code> .
<code>\$max</code>	Retorna o maior valor de expressão para cada grupo.
<code>\$mergeObjects</code>	Retorna um documento criado combinando os documentos de entrada de cada grupo.
<code>\$min</code>	Retorna o menor valor de expressão para cada grupo.
<code>\$stdDevPop</code>	Retorna o desvio padrão populacional dos valores de entrada.
<code>\$stdDevSamp</code>	Retorna o desvio padrão da amostra dos valores de entrada.
<code>\$sum</code>	Retorna uma soma de valores numéricos. Ignora valores não numéricos.





## Disciplina BANCO DE DADOS II

### 5.4.4. Procedimentos para fazer o Download e Instalar o MongoDB

**a)** Fazer o Download do SGBD MongoDB

<https://www.mongodb.com/downloads>

ou tudo pode ser encontrado em:

<https://docs.mongodb.com/manual/>

**b)** Realizar a instalação (localhost) e iniciar o serviço

Tem opções para a instalação local ([localhost](#)), bem como o uso em [Nuvem](#).

No caso da localhost, tem instalações que deixam automático para iniciar o serviço, configurando durante a instalação.

Tem casos em que isso pode ser feito via linha de comando, conforme especificado genericamente a seguir:

Abrir prompt como Administrador e executar a linha na pasta para iniciar o serviço:

```
C:\Windows\system32>
```

```
sc.exe create MongoDB binPath="\"C:\MongoDB\bin\mongod.exe\" --  
service --config=\"C:\MongoDB\mongod.cfg\" DisplayName=\"MongoDB\"  
start=\"auto\"
```

Abrir prompt como Administrator e executar o MongoDB:

```
C:\MongoDB\bin>mongod
```

**c)** Abrir ferramenta de acesso ao MongoDB

No meu caso estou usando o Robo 3T (Robomongo)

Baixar em <https://robomongo.org/>