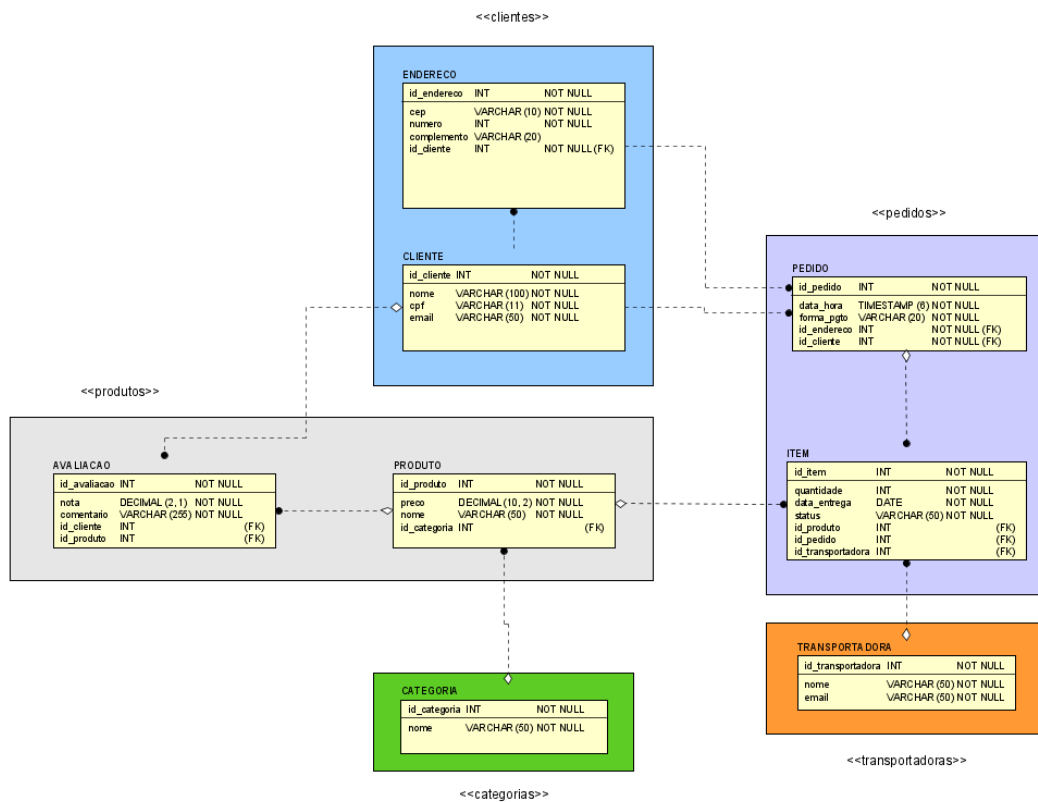


A4 – Consultas no Oracle e MongoDB

Realizado por Lorenzo Panato e Gabriel Dienstmann

1. Modelagem não-relacional:

A modelagem não-relacional foi desenvolvida baseada no modelo abaixo, onde optamos por agrupar as seguintes entidades:



2. Consultas:

- Definindo a variável para conexão com o banco (para o MongoDB):

```
bd = conexao["ecommerce"]
```

a) Consulta que retorna produtos com preço maior que 50 e pertencem à categoria 'Livros' ou 'Roupas':

- SQL:

SELECT *

FROM PRODUTO

WHERE preco > 50 AND (id_categoria = 2 OR id_categoria = 3);

Resultado:

	id_produto	preco	nome	id_categoria
	6	99.99	Hábitos Atômicos	2
	7	59.99	Pai Rico, Pai Pobre	2
	9	79.99	Camiseta	3
	10	239.99	Calça de sarja	3
	11	59.99	Boné	3
	12	299.99	Casaco	3
▶*	NULL	NULL		NULL

- MongoDB:

```
bd.produtos.find({
  '$and': [
    {'preco': {'$gt': 50}},
    {'$or': [{'id_categoria': 2}, {'id_categoria': 3}]}
  ]
})
```

Resultado:

```
[45] result = bd.produtos.find({
    '$and': [
        {'preco': {'$gt': 50}},
        {'$or': [{'id_categoria': 2}, {'id_categoria': 3}]}
    ]
})

# Exibe os resultados
for document in result:
    print(document)
```

```
{'_id': 6, 'preco': 99.99, 'nome': 'Hábitos Atômicos', 'id_categoria': 2, 'avaliacoes': []}
{'_id': 7, 'preco': 59.99, 'nome': 'Pai Rico, Pai Pobre', 'id_categoria': 2, 'avaliacoes': []}
{'_id': 9, 'preco': 79.99, 'nome': 'Camiseta', 'id_categoria': 3, 'avaliacoes': [{'id_aval': 1, 'preco': 10, 'nome': 'Camiseta', 'id_categoria': 3, 'avaliacoes': []}]}
{'_id': 10, 'preco': 239.99, 'nome': 'Calça de sarja', 'id_categoria': 3, 'avaliacoes': []}
{'_id': 11, 'preco': 59.99, 'nome': 'Boné', 'id_categoria': 3, 'avaliacoes': []}
{'_id': 12, 'preco': 299.99, 'nome': 'Casaco', 'id_categoria': 3, 'avaliacoes': []}
```

b) Consulta que calcula o preço médio dos produtos:

- SQL:

```
SELECT AVG(preco) AS preco_medio
FROM PRODUTO;
```

Resultado:

	preco_medio
▶	1039.156667

- MongoDB:

```
bd.produtos.aggregate([
  { '$group': { '_id': None, 'preco_medio': { '$avg': '$preco' } } }
]);
```

Resultado:

```
✓ 0s ▶ result = bd.produtos.aggregate([
    { '$group': { '_id': None, 'preco_medio': { '$avg': '$preco' } } }
  ]);

# Exibe os resultados
for document in result:
    print(document)

↵ {'_id': None, 'preco_medio': 1039.1566666666665}
```

c) Consulta que apresenta dados de pedidos e itens relacionados:

- SQL:

```
SELECT P.id_pedido, I.id_item, I.quantidade, I.status
FROM PEDIDO P
JOIN ITEM I ON P.id_pedido = I.id_pedido;
```

Resultado:

	id_pedido	id_item	quantidade	status
▶	1	1	1	Entregue
	1	2	2	Em Trânsito
	2	3	1	Entregue
	2	4	3	Entregue
	3	5	1	Em Trânsito
	3	6	2	Entregue

- MongoDB:

```
bd.pedidos.aggregate( [
```

```

{ "$unwind": "$itens" },
{ "$project": {
    "_id": 0,
    "id_pedido": "$_id",
    "id_item": "$itens.id_item",
    "quantidade": "$itens.quantidade",
    "status": "$itens.status"
}}
])

```

Resultado:

```

[60] result = bd.pedidos.aggregate([
    ...{"$unwind": "$itens"},
    ...{"$project": {
        ..."_id": 0,
        ..."id_pedido": "$_id",
        ..."id_item": "$itens.id_item",
        ..."quantidade": "$itens.quantidade",
        ..."status": "$itens.status"
    }}
])

# Exibe os resultados
for document in result:
    print(document)

```

```

➡ {'id_pedido': 1, 'id_item': 1, 'quantidade': 1, 'status': 'Entregue'}
{'id_pedido': 1, 'id_item': 2, 'quantidade': 2, 'status': 'Em Trânsito'}
{'id_pedido': 2, 'id_item': 3, 'quantidade': 1, 'status': 'Entregue'}
{'id_pedido': 2, 'id_item': 4, 'quantidade': 3, 'status': 'Entregue'}
{'id_pedido': 3, 'id_item': 5, 'quantidade': 1, 'status': 'Em Trânsito'}
{'id_pedido': 3, 'id_item': 6, 'quantidade': 2, 'status': 'Entregue'}

```

d) Consultas avançadas:

1. Consulta que retorna o nome dos produtos e a média de notas das avaliações desses produtos:

- **SQL:**

```

SELECT P.nome, AVG(A.nota) AS media_nota
FROM PRODUTO P
JOIN AVALIACAO A ON P.id_produto = A.id_produto
GROUP BY P.nome;

```

Resultado:

	nome	media_nota
▶	Iphone 13	4.60000
	Ntebook	3.20000
	A revolução dos bichos	5.00000
	Camiseta	4.80000

- MongoDB:

```

bd.produtos.aggregate( [
  { "$unwind": "$avaliacoes" },

  {
    "$group": {
      "_id": "$_id",
      "nome_produto": { "$first": "$nome" },
      "media_notas": { "$avg": "$avaliacoes.nota" }
    }
  },

  {
    "$project": {
      "_id": 0,
      "nome_produto": 1,
      "media_notas": 1
    }
  }
] )

```

Resultado:

```
result = bd.produtos.aggregate( [
    { "$unwind": "$avaliacoes" },

    {
        "$group": {
            "_id": "$_id",
            "nome_produto": { "$first": "$nome" },
            "media_notas": { "$avg": "$avaliacoes.nota" }
        }
    },

    {
        "$project": {
            "_id": 0,
            "nome_produto": 1,
            "media_notas": 1
        }
    }
])

# Exibe os resultados
for document in result:
    print(document)
```

```
{'nome_produto': 'Iphone 13', 'media_notas': 4.6}
{'nome_produto': 'Notebook', 'media_notas': 3.2}
{'nome_produto': 'Camiseta', 'media_notas': 4.8}
{'nome_produto': 'A revolução dos bichos', 'media_notas': 5.0}
```

2. Consulta que retorna os clientes que fizeram mais de uma avaliação:

- SQL:

```
SELECT C.nome, COUNT(A.id_avaliacao) AS numero_avaliacoes
FROM CLIENTE C
JOIN AVALIACAO A ON C.id_cliente = A.id_cliente
GROUP BY C.nome
```

Resultado:

	nome	numero_avaliacoes
▶	João da Silva	1
	Maria Oliveira	1
	Carlos Souza	1
	Ana Pereira	1
	Marcela Lima	1

- MongoDB:

```
bd.produtos.aggregate( [
    { "$unwind": "$avaliacoes" },
```

```

{
  "$group": {
    "_id": "$avaliacoes.id_cliente",
    "total_avaliacoes": { "$sum": 1 }
  },
  {
    "$lookup": {
      "from": "clientes",
      "localField": "_id",
      "foreignField": "_id",
      "as": "cliente_info"
    }
  },
  { "$unwind": "$cliente_info" },
  {
    "$project": {
      "_id": 0,
      "nome_cliente": "$cliente_info.nome",
      "total_avaliacoes": 1
    }
  }
}
])

```

Resultado:

0s



```
result = bd.produtos.aggregate( [
  { "$unwind": "$avaliacoes" },
  {
    "$group": {
      "_id": "$avaliacoes.id_cliente",
      "total_avaliacoes": { "$sum": 1 }
    }
  },
  {
    "$lookup": {
      "from": "clientes",
      "localField": "_id",
      "foreignField": "_id",
      "as": "cliente_info"
    }
  },
  { "$unwind": "$cliente_info" },
  {
    "$project": {
      "_id": 0,
      "nome_cliente": "$cliente_info.nome",
      "total_avaliacoes": 1
    }
  }
])

# Exibe os resultados
for document in result:
    print(document)
```



```
{'total_avaliacoes': 1, 'nome_cliente': 'Ana Pereira'}
{'total_avaliacoes': 1, 'nome_cliente': 'Marcela Lima'}
{'total_avaliacoes': 1, 'nome_cliente': 'João da Silva'}
{'total_avaliacoes': 1, 'nome_cliente': 'Maria Oliveira'}
{'total_avaliacoes': 1, 'nome_cliente': 'Carlos Souza'}
```