

Questão 1

Inserir 5 novas estações meteorológicas com diferentes sensores

```
db.estacoes_meteorologicas.insertMany([
  { nome: "Estação Sul", localizacao: "Porto Alegre", status: "ativo", sensores:
    ["temperatura", "umidade", "vento"], medicoes: 12 },
  { nome: "Estação Leste", localizacao: "Recife", status: "ativo", sensores: ["chuva",
    "temperatura"], medicoes: 7 },
  { nome: "Estação Oeste", localizacao: "Manaus", status: "inativo", sensores:
    ["umidade", "pressão"], medicoes: 3 },
  { nome: "Estação Central", localizacao: "Brasília", status: "ativo", sensores:
    ["temperatura", "vento", "pressão"], medicoes: 9 },
  { nome: "Estação Serrana", localizacao: "Campos do Jordão", status: "ativo",
    sensores: ["chuva", "temperatura", "umidade", "pressão"], medicoes: 15 }
]);
```

>_ mongosh: localhost:27017



>_MONGOSH



> use clima_alerta

< switched to db clima_alerta

> db.estacoes_meteorologicas.insertMany([

{ nome: "Estação Sul", localizacao: "Porto Alegre", status: "ativo", sensores: ["temper

{ nome: "Estação Leste", localizacao: "Recife", status: "ativo", sensores: ["chuva", "t

{ nome: "Estação Oeste", localizacao: "Manaus", status: "inativo", sensores: ["umidade'

{ nome: "Estação Central", localizacao: "Brasília", status: "ativo", sensores: ["temper

{ nome: "Estação Serrana", localizacao: "Campos do Jordão", status: "ativo", sensores:

]);

< {

acknowledged: true,

insertedIds: {

'0': ObjectId('68f939bbcf2b43b7929b0a12'),

'1': ObjectId('68f939bbcf2b43b7929b0a13'),

'2': ObjectId('68f939bbcf2b43b7929b0a14'),

'3': ObjectId('68f939bbcf2b43b7929b0a15'),

'4': ObjectId('68f939bbcf2b43b7929b0a16')

}

}

Questão 2

Adicionar um novo campo "manutenção" com valor pendente a todas as estações

```
db.estacoes_meteorologicas.updateMany({}, { $set: { manutencao: "pendente" } });
```

```
> db.estacoes_meteorologicas.updateMany({}, { $set: { manutencao: "pendente" } });
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 6,
  modifiedCount: 6,
  upsertedCount: 0
}
> db.estacoes_meteorologicas.find()
< {
  _id: ObjectId('68f93519b9c13166aa6ebdfe'),
  nome: 'Estação Norte',
  localizacao: 'Rio de Janeiro',
  status: 'ativo',
  'medições': 5,
  manutencao: 'pendente'
}
{
  _id: ObjectId('68f939bbcf2b43b7929b0a12'),
  nome: 'Estação Sul',
  localizacao: 'Porto Alegre',
  status: 'ativo',
  sensores: [
    'temperatura',
    'umidade',
    'vento'
  ],
  medicoes: 12,
  manutencao: 'pendente'
}
{
  _id: ObjectId('68f939bbcf2b43b7929b0a13'),
  nome: 'Estação Leste',
  localizacao: 'Recife',
  status: 'ativo',
  sensores: [
    'chuva',
    'temperatura'
  ],
  medicoes: 7,
  manutencao: 'pendente'
}
{
  _id: ObjectId('68f939bbcf2b43b7929b0a14'),
  nome: 'Estação Oeste',
  localizacao: 'Manaus',
  status: 'inativo',
  sensores: [
    'umidade',
    'pressão'
  ],
  medicoes: 3,
  manutencao: 'pendente'
}
```

Questão 3

Atualizar a estação "Estação Sul" para adicionar um novo sensor "pressão".

```
db.estacoes_meteorologicas.updateOne(  
  { nome: "Estação Sul" },  
  { $set: { sensores: "pressão" } }  
);
```

```
> db.estacoes_meteorologicas.updateOne(  
  { nome: "Estação Sul" },  
  { $set: { sensores: "pressão" } }  
);  
< {  
  acknowledged: true,  
  insertedId: null,  
  matchedCount: 1,  
  modifiedCount: 1,  
  upsertedCount: 0  
}
```

Questão 4

Atualizar a "Estação Norte" para remover o sensor "vento"

```
db.estacoes_meteorologicas.updateOne(  
  { nome: "Estação Norte" },  
  { $pull: { sensores: "vento" } }  
);
```

```

> db.estacoes_meteorologicas.updateOne(
  { nome: "Estação Norte" },
  { $pull: { sensores: "vento" } }
);
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 0,
  upsertedCount: 0
}

```

Questão 5

Remover todas as estações que não possuem sensores cadastrados.

```

db.estacoes_meteorologicas.deleteMany({
  $or: [
    { sensores: { $exists: false } },
    { sensores: { $size: 0 } }
  ]
});

```

```

> db.estacoes_meteorologicas.deleteMany({
  $or: [
    { sensores: { $exists: false } },
    { sensores: { $size: 0 } }
  ]
});
< {
  acknowledged: true,
  deletedCount: 1
}

```

Questão 6

Remover todas as estações que possuem menos de 3 sensores

```

db.estacoes_meteorologicas.deleteMany({

```

```
$expr: { $lt: [ { $size: "$sensores" }, 3 ] }
});
```

```
> db.estacoes_meteorologicas.deleteMany({
  $expr: { $lt: [ { $size: "$sensores" }, 3 ] }
});
< {
  acknowledged: true,
  deletedCount: 5
}
```

Questão 7

Listar todas as estações que possuem o sensor "temperatura"

```
db.estacoes_meteorologicas.find({
  sensores: "temperatura"
});
```

Questão 8

Encontrar todas as estações que possuem o sensor "temperatura" E "pressão".

```
db.estacoes_meteorologicas.find({
  sensores: { $all: ["temperatura", "pressão"] }
});
```

Questão 9

Identificar todas as estações cujo campo localizacao seja do tipo string

```
db.estacoes_meteorologicas.find({
  localizacao: { $type: "string" }
});
```

Questão 10

Aplicar paginação para listar 4 estações por vez, ignorando as 2 primeiras

```
db.estacoes_meteorologicas.find().skip(2).limit(4);
```