

Referência > Operadores > Operadores de consulta e projeção > Operadores de consulta lógica > \$ not

# \$ não

## Nesta página

- Comportamento

## \$not

**Sintaxe** : { field: { \$not: { <operator-expression> } } }

\$not executa uma NOT operação lógica no especificado <operator-expression> e seleciona os documentos que *não* correspondem <operator-expression>. Isso inclui documentos que não contêm o field.

Considere a seguinte consulta:

```
db . inventário . find ( { price : { $ not : { $ gt : 1.99 } } } )
```

Esta consulta selecionará todos os documentos da `inventory` coleção em que:

- o `price` valor do campo é menor ou igual a 1.99 ou
- o `price` campo não existe

{ \$not: { \$gt: 1.99 } } é diferente do \$lte operador. retorna *apenas* os documentos em que o campo existe e seu valor é menor ou igual a .{ \$lte: 1.99 }price1.99

Lembre-se de que o \$not operador afeta apenas *outros operadores* e não pode verificar campos e documentos independentemente. Portanto, use o \$not operador para disjunções lógicas e o \$ne operador para testar diretamente o conteúdo dos campos.

## Comportamento

### \$note tipos de dados

A operação do \$not operador é consistente com o comportamento do MongoDB. Documentação Documentação

resultados inesperados com alguns tipos de dados, como i

[Search Documentation](#)

## \$not expressões regulares

\$not O operador pode executar NOT operação lógica em:

- objetos de expressão regular (ie /pattern/)

Por exemplo, a consulta a seguir seleciona todos os documentos da inventory coleção em que o item valor do campo *não* começa com a letra p.

```
db . inventário . find ( { item : { $ not : /^p.* / } } )
```

- \$regex expressão do operador (Iniciando no MongoDB 4.0.7)

Por exemplo, a consulta a seguir seleciona todos os documentos da inventory coleção em que o item valor do campo *não* começa com a letra p.


```
db.inventory.find( { item: { $not: { $regex: "^p.*" } } } )
db.inventory.find( { item: { $not: { $regex: /^p.* / } } } )
```

- driver language's regular expression objects

For example, the following PyMongo query uses Python's `re.compile()` method to compile a regular expression:

```
import re
para noMatch em db . inventário . find ( { "item" : { "$ not" : re . compi
    imprima noMatch
```

### VEJA TAMBÉM:

`find()`, `update()`, `$set`, `$gt`, `$regex`, [PyMongo](#) , `motorista` .