

# \$ eq

## Nesta página

- Comportamento
- Exemplos

## \$eq

*Novo na versão 3.0.*

Especifica a condição de igualdade. O \$eq operador corresponde aos documentos em que o valor de um campo é igual ao valor especificado.

```
{ < campo >: { $ eq : < valor > } }
```

A \$eq expressão é equivalente a `{ field: <value> }`

## Comportamento

### Ordem de comparação

Para comparação de diferentes valores do tipo BSON, consulte a ordem de comparação do BSON especificada .

### Corresponder a um valor do documento

Se o especificado <value> for um documento, a ordem dos campos no documento é importante.

### Corresponder a um valor de matriz

Se o especificado <value> for uma matriz, o MongoDB corresponderá aos documentos em que <field> corresponde exatamente à matriz ou <field> contém um elemento que corresponde exatamente à matriz. A ordem dos elementos é importante. Para um exemplo, consulte Igual a um valor de matriz .

# Exemplos

mongoDB. Documentation

Os exemplos a seguir consultam a `inventory` coleção com os seguintes documentos:

```
{ _id: 1, item: { name: "ab", code: "123" }, qty: 15, tags: [ "A", "B", "C" ] }
{ _id: 2, item: { name: "cd", code: "123" }, qty: 20, tags: [ "B" ] }
{ _id: 3, item: { name: "ij", code: "456" }, qty: 25, tags: [ "A", "B" ] }
{ _id: 4, item: { name: "xy", code: "456" }, qty: 30, tags: [ "B", "A" ] }
{ _id: 5, item: { name: "mn", code: "000" }, qty: 20, tags: [ [ "A", "B" ], "C" ] }
```

## Equals a Specified Value

The following example queries the `inventory` collection to select all documents where the value of the `qty` field equals 20:

```
db.inventory.find( { qty: { $eq: 20 } } )
```

The query is equivalent to:

```
db.inventory.find( { qty: 20 } )
```

Both queries match the following documents:

```
{ _id: 2, item: { name: "cd", code: "123" }, qty: 20, tags: [ "B" ] }
{ _id: 5, item: { name: "mn", code: "000" }, qty: 20, tags: [ [ "A", "B" ], "C" ] }
```

## Field in Embedded Document Equals a Value

The following example queries the `inventory` collection to select all documents where the value of the `name` field in the `item` document equals "ab". To specify a condition on a field in an embedded document, use the dot notation.

```
db.inventory.find( { "item.name": { $eq: "ab" } } )
```

The query is equivalent to the following query:

```
db.inventory.find( { "item.name": "ab" } )
```

Both queries match the following document:

```
{ _id: 1, item: { name: "ab", code: "123" }, qty: 15, tags: [ "A", "B", "C" ] }
```

SEE ALSO:

Query Embedded Documents

## Array Element Equals a Value

The following example queries the `inventory` collection to select all documents where the `tags` array contains an element with the value `"B"` [1]:

```
db.inventory.find( { tags: { $eq: "B" } } )
```

The query is equivalent to:

```
db.inventory.find( { tags: "B" } )
```

Both queries match the following documents:

```
{ _id: 1, item: { name: "ab", code: "123" }, qty: 15, tags: [ "A", "B", "C" ] }  
{ _id: 2, item: { name: "cd", code: "123" }, qty: 20, tags: [ "B" ] }  
{ _id: 3, item: { name: "ij", code: "456" }, qty: 25, tags: [ "A", "B" ] }  
{ _id: 4, item: { name: "xy", code: "456" }, qty: 30, tags: [ "B", "A" ] }
```

SEE ALSO:

\$elemMatch, Query Arrays

[1] The query DSL also has documentation on the MongoDB Search Documentation page.

## Equals an Array Value

The following example queries the `inventory` collection to select all documents where the `tags` array equals exactly the specified array or the `tags` array contains an element that equals the array [ "A", "B" ].

```
db.inventory.find( { tags: { $eq: [ "A", "B" ] } } )
```

The query is equivalent to:

```
db.inventory.find( { tags: [ "A", "B" ] } )
```

Both queries match the following documents:

```
{ _id: 3, item: { name: "ij", code: "456" }, qty: 25, tags: [ "A", "B" ] }  
{ _id: 5, item: { name: "mn", code: "000" }, qty: 20, tags: [ [ "A", "B" ], "C" ] }
```

