

# MongoDB - Query Operators

Topics : [MongoDB](#)

Written on [December 30, 2023](#)

In MongoDB, query operators are used to specify conditions in queries. They allow you to filter documents based on various criteria. Here are some common MongoDB query operators:

## Comparison Operators:

### 1. Equality:

- \$eq: Matches values that are equal to a specified value.

```
db.collection_name.find({ field: { $eq: value } });
```

### 2. Inequality:

- \$ne: Matches values that are not equal to a specified value.

```
db.collection_name.find({ field: { $ne: value } });
```

### 3. Greater Than:

- \$gt: Matches values that are greater than a specified value.

```
db.collection_name.find({ field: { $gt: value } });
```

### 4. Greater Than or Equal To:

- \$gte: Matches values that are greater than or equal to a specified value.

```
db.collection_name.find({ field: { $gte: value } });
```

### 5. Less Than:

- \$lt: Matches values that are less than a specified value.

```
db.collection_name.find({ field: { $lt: value } });
```

### 6. Less Than or Equal To:

- \$lte: Matches values that are less than or equal to a specified value.

```
db.collection_name.find({ field: { $lte: value } });
```

## Logical Operators:

### 1. Logical AND:

- \$and: Joins query clauses with a logical AND.

```
db.collection_name.find({ $and: [ { field1: value1 }, { field2: value2 } ] });
```

### 2. Logical OR:

- \$or: Joins query clauses with a logical OR.

```
db.collection_name.find({ $or: [ { field1: value1 }, { field2: value2 } ] });
```

### 3. Logical NOT:

- \$not: Inverts the effect of a query expression and returns documents that do not match the query expression.

```
db.collection_name.find({ field: { $not: { $eq: value } } });
```

## Element Operators:

### 1. Existence:

- \$exists: Matches documents that have the specified field.

```
db.collection_name.find({ field: { $exists: true } });
```

### 2. Type:

- \$type: Matches documents where the value of a field is of the specified BSON data type.

```
db.collection_name.find({ field: { $type: "string" } });
```

## Array Operators:

### 1. Element in Array:

- \$in: Matches any of the values specified in an array.

```
db.collection_name.find({ field: { $in: [value1, value2] } });
```

## 2. All Elements in Array:

- \$all: Matches arrays that contain all elements specified in the query.

```
db.collection_name.find({ field: { $all: [value1, value2] } });
```

## 3. Size of Array:

- \$size: Matches arrays with a specific number of elements.

```
db.collection_name.find({ field: { $size: 3 } });
```