

# How can we use a “NOT IN” query in MongoDB?

Authored by  
**stats writer**

July 2, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can we use a “NOT IN” query in MongoDB?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165507>

A "NOT IN" query in MongoDB is a type of query that allows users to retrieve documents from a collection that do not match a specified set of values. This query is useful for filtering out unwanted data and can be used in various scenarios, such as finding documents that do not contain certain keywords or exclude specific categories. To use a "NOT IN" query, the user must specify the field to be checked and the set of values to be excluded. This query can help improve data accuracy and efficiency in MongoDB databases.

## MongoDB: Use a "NOT IN" Query

You can use the following syntax to query for all documents where the value for a particular field is not in a certain list of values:

```
db.collection.find({field1: {$nin: }})
```

This particular query finds all documents where the value in field1 is not equal to value1, value2, or value3.

The following examples show how to use this syntax in practice.

Example 1: Query for "NOT IN" with One Value

Suppose we have a collection teams with the following documents:

```
db.teams.insertOne({team: "Mavs", position: "Guard",  
points: 31})db.teams.insertOne({team: "Spurs",
```

```
position: "Guard", points: 22})db.teams.insertOne({team: "Rockets", position: "Center", points: 19})db.teams.insertOne({team: "Warriors", position: "Forward", points: 26})db.teams.insertOne({team: "Cavs", position: "Guard", points: 33})
```

We can use the following code to find all documents where the "team" field is not equal to the value "Rockets":

```
db.teams.find({team: {$nin: }})
```

This query returns the following documents:

```
{_id: ObjectId("619527e467d6742f66749b72"), team: 'Cavs', position: 'Guard', points: 33 }
```

```
{_id: ObjectId("619527e467d6742f66749b6e"), team: 'Mavs', position: 'Guard', points: 31 }
```

```
{_id: ObjectId("619527e467d6742f66749b6f"),  
team: 'Mavs',  
position: 'Guard',  
points: 22 }
```

Notice that the only documents returned are the ones where the "team" field is not equal to "Rockets."

Example 2: Query for "NOT IN" with List of Values

Suppose we have a collection teams with the following documents:

```
db.teams.insertOne({team: "Mavs", position: "Guard",  
points: 31})db.teams.insertOne({team: "Spurs",  
position: "Guard", points:  
22})db.teams.insertOne({team: "Rockets", position:  
"Center", points: 19})db.teams.insertOne({team:  
"Warriors", position: "Forward", points:  
26})db.teams.insertOne({team: "Cavs", position:  
"Guard", points: 33})
```

We can use the following code to find all documents where the "team" field is not equal to the value "Rockets" or "Cavs":

```
db.teams.find({team: {$nin: }})
```

```
{ _id: ObjectId("619527e467d6742f66749b6e"),  
team: 'Mavs',  
position: 'Guard',  
points: 31 }
```

```
{ _id: ObjectId("619527e467d6742f66749b6f"),  
team: 'Mavs',  
position: 'Guard',  
points: 22 }
```

Notice that the only documents returned are the ones where the "team" field is not equal to "Rockets" or "Cavs."

**Note:** You can find the complete documentation for the \$nin function .

**Additional Resources**

The following tutorials explain how to perform other common operations in MongoDB: