

“How can we use the ‘Group By’ function in MongoDB to count the number of documents in a collection?”

Authored by
stats writer

July 2, 2024

RECOMMENDED CITATION

stats writer (2024). “How can we use the ‘Group By’ function in MongoDB to count the number of documents in a collection?”. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165657>

The 'Group By' function in MongoDB allows users to group documents in a collection based on a specified field and perform aggregate functions on the grouped data. This function can be used to count the number of documents in a collection by grouping them together and using the \$sum operator to calculate the total count. This feature is useful for organizing and analyzing large datasets, making it easier to retrieve specific information and gain insights from the data. By utilizing the 'Group By' function, users can efficiently count the number of documents in a collection and gain a better understanding of the data contained within it.

MongoDB: Group By and Count

You can use the following syntax to group by and count in MongoDB:

```
db.collection.aggregate()
```

Note that `field_name` is the field you'd like to group by.

The following examples show how to use this syntax with 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})
```

Example 1: Group By and Count

We can use the following code to group by the 'position' field and count the occurrences of each position.

```
db.teams.aggregate()
```

This returns the following results:

```
{_id: 'Forward', count: 1 }  
{_id: 'Guard', count: 3 }  
{_id: 'Center', count: 1 }
```

This tells us:

The position 'Forward' occurs 1 time. The position 'Guard' occurs 3 times. The position 'Center' occurs 1 time.

Example 2: Group By and Count (Then Sort)

We can use the following code to count the occurrences of each position and automatically sort the results in ascending order:

```
db.teams.aggregate()
```

This returns the following results:

```
{ _id: 'Forward', count: 1 }
```

```
{ _id: 'Center', count: 1 }
```

```
{ _id: 'Guard', count: 3 }
```

db.teams.aggregate()

This returns the following results:

```
{ _id: 'Guard', count: 3 }
```

```
{ _id: 'Forward', count: 1 }
```

```
{ _id: 'Center', count: 1 }
```

Note: You can find the complete documentation for `$group` .

Additional Resources

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