

How can I select distinct values from multiple fields in MongoDB?

Authored by
stats writer

July 2, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I select distinct values from multiple fields in MongoDB?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165503>

MongoDB is a popular NoSQL database management system that allows for the storage and retrieval of large amounts of data. One common task when working with MongoDB is selecting distinct values from multiple fields. This can be achieved by using the MongoDB aggregation framework, which enables the manipulation and processing of data in a variety of ways. By using the distinct function in conjunction with the aggregation framework, users can specify which fields they want to retrieve unique values from, allowing for efficient and accurate data selection. This method is useful for data analysis and reporting purposes, as it allows for the identification of unique values across multiple fields in a MongoDB database.

MongoDB: Select Distinct Values from Multiple Fields

You can use the following syntax to select distinct values from multiple fields in MongoDB:

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

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

```
db.teams.insertOne({team: "Rockets", position:
```

```
"Forward", points: 29})db.teams.insertOne({team:
"Cavs", position: "Guard", points: 33})
```

Example: Select Distinct Values from Multiple Fields

We can use the following query to find all of the distinct values from the team and position fields:

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

This query returns the following documents:

```
{_id: { team: 'Mavs', position: 'Guard' }}
{_id: { team: 'Rockets', position: 'Forward' }}
{_id: { team: 'Rockets', position: 'Center' }}
{_id: { team: 'Cavs', position: 'Guard' }}
```

And we can use the following query to find all of the distinct values from the team, position, and points fields:

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

This query returns the following documents:

```
{ _id: { team: 'Cavs', position: 'Guard', points: 33 } }  
{ _id: { team: 'Rockets', position: 'Forward', points: 29 } }  
{ _id: { team: 'Mavs', position: 'Guard', points: 22 } }  
{ _id: { team: 'Rockets', position: 'Forward', points: 26 } }  
{ _id: { team: 'Mavs', position: 'Guard', points: 31 } }  
{ _id: { team: 'Rockets', position: 'Center', points: 19 } }
```

Notice that all of the documents are returned since there are no two documents that have identical values for the team, position, *and* points fields.

Additional Resources

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