

How do I replace strings in MongoDB with an example?

Authored by
stats writer

June 30, 2024

RECOMMENDED CITATION

stats writer (2024). *How do I replace strings in MongoDB with an example?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=162509>

Replacing strings in MongoDB refers to the process of updating or modifying specific strings within a collection or document in the database. This can be achieved by using the update or replace methods in MongoDB, which allow for the replacement of old strings with new ones. For example, if we have a collection of documents with a field called "name" and we want to replace all occurrences of "John" with "Jack", we can use the update method to search for "John" and replace it with "Jack". This can be useful for making changes to data or correcting errors in the database.

Replace Strings in MongoDB (With Example)

You can use the following syntax to replace a specific string in a field in MongoDB:

```
db.myCollection.updateMany(  
{ fieldName: { $regex: /old/ } },  
)
```

This particular example replaces the string "old" with "new" in the field titled "fieldName" within the collection titled myCollection.

The following example shows how to use this syntax in practice with a collection teams with the following documents:

```
db.teams.insertOne({team: "Mavs", conference:  
"Western", points: 31})
```

```
db.teams.insertOne({team: "Spurs", conference:
```

```
"Western", points: 22})
```

```
db.teams.insertOne({team: "Rockets", conference:  
"Western", points: 19})
```

```
db.teams.insertOne({team: "Celtics", conference:  
"Eastern", points: 26})
```

```
db.teams.insertOne({team: "Cavs", conference:  
"Eastern", points: 33})
```

```
db.teams.insertOne({team: "Nets", conference:  
"Eastern", points: 38})
```

Example: Replace String in MongoDB

We can use the following code to replace the string "Western" with "West" in the conference field:

```
db.teams.updateMany(  
{ conference: { $regex: /Western/ } },  
  
)
```

Here's what the updated collection now looks like:

```
{ _id: ObjectId("620139494cb04b772fd7a8fa"),  
team: 'Mavs',  
conference: 'West',
```

```
points: 31 }
{ _id: ObjectId("620139494cb04b772fd7a8fb"),
team: 'Spurs',
conference: 'West',
points: 22 }
{ _id: ObjectId("620139494cb04b772fd7a8fc"),
team: 'Rockets',
conference: 'West',
points: 19 }
{ _id: ObjectId("620139494cb04b772fd7a8fd"),
team: 'Celtics',
conference: 'Eastern',
points: 26 }
{ _id: ObjectId("620139494cb04b772fd7a8fe"),
team: 'Cavs',
conference: 'Eastern',
points: 33 }
{ _id: ObjectId("620139494cb04b772fd7a8ff"),
team: 'Nets',
conference: 'Eastern',
points: 38 }
```

Notice that every document that had the string "Western" in the conference field now has "West" in the

conference field.

Any document that did not have the string "Western" in the conference field simply kept their original string.

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

Additional Resources

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