MongoDB Create Collection

MongoDB Create Collection

MongoDB Create Collection – In this MongoDB Tutorial, we shall learn ways to create a MongoDB Collection There are both implicit and explicit ways to create a collection.

Implicitly Create Collection

To create a new collection in MongoDB implicitly, follow below steps :

1. Select a MongoDB database you like to Create a Collection within, using USE command. Following is the syntax of USE command :use <database_name>
2. Insert a record to Collection, with Collection Name mentioned in the command as shown belowdb.<collection_name>.insert(<document>) The collection should be created.
3. View the existing collections using following commandshow collections

Example to Create a MongoDB Collection

Following is an example where we shall try creating a collection namedcustomers intutorialkart database.

Open Mongo Shell and follow the commands in sequence.

Mongo Shell - Commands to Create Collection Implicitly

```
> use tutorialkart
switched to db
```

```
> use tutorialkart
switched to db tutorialkart
> show collections
> db.customers.insert({ name: "Honey", age: 25, cars: [ "Audi R8" ] })
WriteResult({ "nInserted" : 1 })
> show collections
customers
>
```

Following is the explanation for each mongodb command we executed above

1. use tutorialkart switched to tutorialkart database.
2. show collections lists the collections in the selected database. There are no collections at this moment.
3. db.customers.insert() makes a check if the collection is present. As collection is not present yet, it creates one with the namecustomers specified in the command.
4. show collections now listscustomers collection.

Explicitly Create Collection

To create a new collection in MongoDB explicitly, follow below steps :

1. Select a MongoDB database you like to Create a Collection within, using USE command. Following is the syntax of USE command :use <database_name>
2. Following is the syntax of createCollection() command that creates a new collection.

```
db.createCollection(name, options)
```

name	[mandatory] name of collection
options	[optional] mongodb document specifying information about collection

Options

Field	Type	Description
capped	Boolean	If **true** then collection is limited in size. i.e., Number of documents that could be stored in the collection is limited.
autoIndexId	Boolean	[depreciated]
size	Number	Size of Collection in Bytes
max	Number	Number of MongoDB Documents that could be stored in the collection

Note#1 : All fields in Options are optional.Note#2 : If you specify capped as true, then you should specify size.Note#3 : When a new Document arrives Collection for insertion, size and max are the prerequisites that are verified. If the collection comes to the threshold of any, old documents are overwritten in a round robin fashion.

3. View the existing collections using following commandshow collections

Example to Create a MongoDB Collection

Following is an example where we shall try creating a collection namedmyNewCollection intutorialkart database.

Open Mongo Shell and follow the commands in sequence.

Mongo Shell - Commands to Create a Collection Explicitly

```
> use tutorialkart
switched to db
```

```
> use tutorialkart
switched to db tutorialkart
> show collections
customers
> db.createCollection("myNewCollection", { capped : true, size : 280000, max : 1000 } )
{ "ok" : 1 }
> show collections
customers
myNewCollection
>
```

Following is the explanation for each mongodb command we executed above

1. `use tutorialkart` switched to tutorialkart database.
2. `show collections` listed the collections in the selected database. There is only a collection named `customers` .
3. `db.createCollection()` created new collection named `myNewCollection` .
4. `show collections` now lists `customers` collection.

Conclusion :

In this MongoDB Tutorial –**MongoDB Create Database**, we have learnt to create a database using an example scenario.

MongoDB Tutorial

- MongoDB Tutorial
- Install MongoDB on Ubuntu
- Start MongoDB Server
- MongoDB Shell
- Check MongoDB Version
- MongoDB Server Port Number - Default Value and How to Change it
- MongoDB Script

Database

- MongoDB Database
- MongoDB Create Database
- MongoDB Delete Database

Collections

- MongoDB Collection
- MongoDB Create Collection
- MongoDB Delete Collection

MongoDB Delete Collection

Documents

- MongoDB Document
- MongoDB Insert Document
- MongoDB Query Documents
- MongoDB Project Fields in Result
- MongoDB Update Document
- MongoDB Delete Document
- MongoDB Limit Documents
- MongoDB Skip Documents
- MongoDB Sort Documents
- MongoDB Setup Replica Set
- MongoDB Locks

MongoDB Concepts

- MongoDB Text Search
- MongoDB MapReduce
- MongoDB Backup - mongodump

MongoDB Queries

- MongoDB Date

MongoDB Queries

- MongoDB Date()

MongoDB Integration

MongoDB Java

- Connect to MongoDB from Java

MongoDB Python

- Connect to MongoDB from Python

MongoDB Kotlin

- Connect to MongoDB from Kotlin

MongoDB Node.js

- Node.js MongoDB
- Node.js MongoDB Connection
- Node.js MongoDB Create Database
- Node.js MongoDB Drop Database
- Node.js MongoDB Create Collection

- Node.js MongoDB Create Collection
- Node.js MongoDB Delete Collection
- Node.js MongoDB Insert Documents
- MongoError: failed to connect to server

MongoDB Others

- MongoDB Interview Questions
- Uninstall MongoDB from Ubuntu

Useful Resources

- How to Learn Programming