



# The MongoDB CheatSheet

Created by JS Mastery

Visit [jsmastery.pro](https://jsmastery.pro) for more



mongoDB

# Connect MongoDB Shell

Connects to

`mongodb://127.0.0.1:27017` by default

```
mongo
```

Connect to local network

```
mongo "mongodb://192.168.1.1:27017"
```

Connect to MongoDB Atlas

```
mongo "mongodbURI" --username <username>
```

# Brought to you by JSM

This guide will provide you with useful information and actionable steps, but if you truly want to dominate the competition and secure a high-paying job as a full-stack software developer, [jsmastery.pro](https://jsmastery.pro) is the answer.

Read until the end for more information and **special discounts!** 🙌



# Database Operation

## Show All Databases

```
show dbs
```

## Show Current Database

```
db
```

## Create / Switch to new Database

```
use jsmasterypro
```

# Database Operation

## Delete Database

```
db.dropDatabase()
```

# Collections

Show all collections of current DB

```
show collections
```

Create new Collection

```
db.createCollection('courses')
```

# Create Documents

Insert one document

```
db.courses.insertOne({  
  name: "Ultimate React",  
  price: 50  
})
```

```
db.courses.insert({  
  name: "Ultimate React",  
  price: 50  
})
```

# Create Documents

Insert multiple document

```
db.courses.insert(  
  {  
    name: "Ultimate React",  
    price: 50  
  },  
  {  
    name: "Complete path to JSM",  
    price: 80  
  }  
)
```



# Read Documents

Find one document

```
db.courses.findOne()
```

Find multiple documents

```
db.courses.find()
```

Find Multiple documents with json

```
db.courses.find().pretty()
```

# Read Documents

Find documents by field value

```
db.courses.find({  
  "name": "Ultimate React"  
})
```

Find By Element in Array

```
db.courses.find({  
  comments: {  
    $elemMatch: {  
      "user": "John"  
    }  
  }  
})
```

# Update Documents

## Update one

```
db.courses.updateOne(  
  { "_id": 1 },  
  { $set:  
    { "name": "Ultimate React.js" }  
  }  
)
```

## Update Multiple

```
db.courses.update(  
  { "price": 50 },  
  { $set: { "price": 49 } }  
)
```

# Update Documents

## Increment Field Value

```
db.courses.update(  
  { "_id": 2 },  
  { $inc: { "views": 109 } }  
)
```

## Rename Field

```
db.courses.updateOne(  
  { _id: 3 },  
  { $rename:  
    { 'views': 'likes' }  
  })
```

# Update Documents

## Update Sub-Documents

```
db.courses.update(  
  { "name": "Ultimate React.js" },  
  { $set: {  
    comments: [  
      {  
        "user": "Michael",  
        "body": "Comment One",  
      },  
      {  
        "user": "Angelia",  
        "body": "Comment Two",  
      }  
    ]  
  }  
})
```

# Delete Documents

Delete a document

```
db.courses.remove(  
  { name: 'Ultimate React.js' }  
)
```

# Count

Returns the count of documents

```
db.courses.find().count()
```

Returns the count of documents that would match a find() query

```
db.courses.find({  
  views: 110  
}).count()
```

# Sorting

## Ascending order

```
db.courses.find.sort(  
  { name: 1 }  
)
```

## Descending order

```
db.courses.find.sort(  
  { name: -1 }  
)
```



# Results by Pagination

Skip 7 results

```
db.courses.find({}).skip(7)
```

Fetch only 4 results

```
db.courses.find({}).limit(4)
```

# Results by Pagination

Sort by name , Skip first 7 results,  
fetch only next 4 documents

```
db.courses.find({}).sort({  
  "name": 1  
}).skip(7).limit(4)
```

# Add Index

Create Index on single field

```
db.courses.createIndex(  
  { "name": 1 }  
)
```

Create compound Index

```
db.courses.createIndex(  
  { "name": 1, "date": 1 }  
)
```

# Drop Index

## Drop Index

```
db.courses.dropIndex("name_1")
```

# Text Search

Create Text Index on field

```
db.courses.createIndex(  
  {content: "text"}  
)
```

Search by Text

```
db.courses.find({  
  $content: {  
    $search: "JS"  
  }  
})
```

# Find by range query

find courses where views are greater than 70

```
db.courses.find({  
  'views' : { '$gt' : 70 }  
})
```

find courses where views are greater than or equal to 70

```
db.courses.find({  
  'views' : { '$gte' : 70 }  
})
```

# Find by range query

find courses where views are less than 70

```
db.courses.find({  
  'views' : { '$lt' : 70 }  
})
```

find courses where views are less than or equal to 70

```
db.courses.find({  
  'views' : { '$lte' : 70 }  
})
```

# JS Mastery Pro

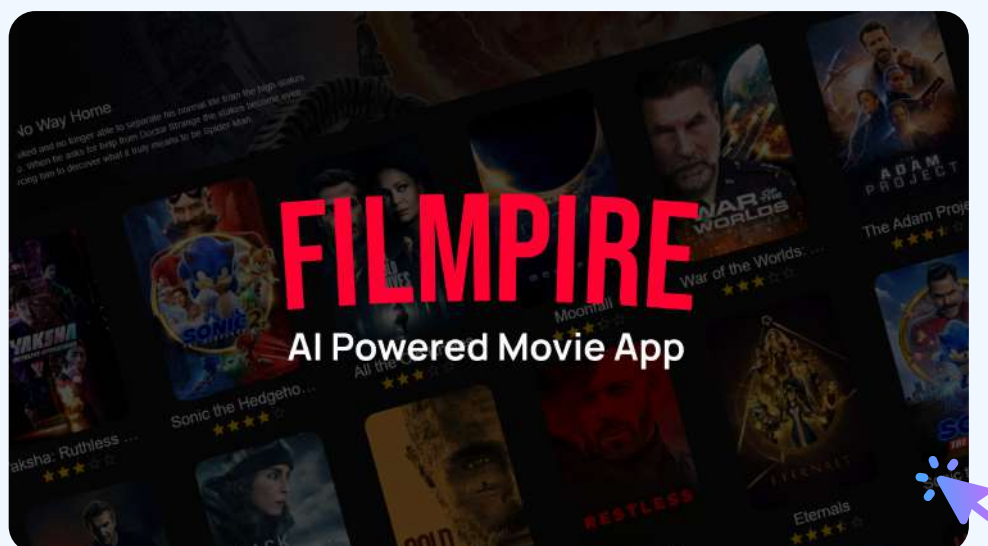
Looking to advance your career and understand the concepts & technologies that top-shelf employers are looking for?

JS Mastery Pro offers two courses that will help you master libraries, tools, and technologies such as React.js, Next.js, Material UI, Solidity, Redux, and many more.

If your goal is to earn a high income while working on projects you love, JS Mastery Pro can help you develop your skills to become a top candidate for lucrative employment and freelance positions.







Become a React.js master as you create a stunning Netflix clone streaming app to showcase movies, actor bios, and more with advanced AI voice functionality.



Leverage Web 3.0 and blockchain technology to build a comprehensive NFT platform where users can discover, create, purchase, & sell non-fungible tokens.

Plus, if you really want to make a splash and add multiple group projects to your portfolio, join the JSM Masterclass Experience to set yourself above the rest and impress hiring managers.



**Collaborate with other developers on exciting monthly group projects, have your code reviewed by industry experts, and participate in mock interviews and live Q&As. With two masterclass options available, this is the best way to truly launch your programming career and secure the job of your dreams!**

Visit [jsmastery.pro](https://jsmastery.pro) today to get started!

