

```
# MongoDB Cheat Sheet
```

```
## Show All Databases
```

```
---
```

```
show dbs
```

```
---
```

```
## Show Current Database
```

```
---
```

```
db
```

```
---
```

```
## Create Or Switch Database
```

```
---
```

```
use acme
```

```
---
```

```
## Drop
```

```
---
```

```
db.dropDatabase()
```

```
---
```

```
## Create Collection
```

```
---
```

```
db.createCollection('posts')
```

```
---
```

```
## Show Collections
```

```
---
```

```
show collections
```

```
---
```

```
## Insert Row
```

```
---
```

```
db.posts.insert({  
  title: 'Post One',  
  body: 'Body of post one',  
  category: 'News',  
  tags: ['news', 'events'],  
  user: {  
    name: 'John Doe',  
    status: 'author'  
  },  
})
```

```
    date: Date()
  })
  ...
```

```
## Insert Multiple Rows
```

```
...
db.posts.insertMany([
  {
    title: 'Post Two',
    body: 'Body of post two',
    category: 'Technology',
    date: Date()
  },
  {
    title: 'Post Three',
    body: 'Body of post three',
    category: 'News',
    date: Date()
  },
  {
    title: 'Post Four',
    body: 'Body of post three',
    category: 'Entertainment',
    date: Date()
  }
])
...
```

```
## Get All Rows
```

```
...
db.posts.find()
...
```

```
## Get All Rows Formatted
```

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

```
## Find Rows
```

```
...
db.posts.find({ category: 'News' })
...
```

```
## Sort Rows
```

```
...
```

```

# asc
db.posts.find().sort({ title: 1 }).pretty()
# desc
db.posts.find().sort({ title: -1 }).pretty()
'''

## Count Rows

'''
db.posts.find().count()
db.posts.find({ category: 'news' }).count()
'''

## Limit Rows

'''
db.posts.find().limit(2).pretty()
'''

## Chaining

'''
db.posts.find().limit(2).sort({ title: 1 }).pretty()
'''

## Foreach

'''
db.posts.find().forEach(function(doc) {
  print("Blog Post: " + doc.title)
})
'''

## Find One Row

'''
db.posts.findOne({ category: 'News' })
'''

## Find Specific Fields

'''
db.posts.find({ title: 'Post One' }, {
  title: 1,
  author: 1
})
'''

## Update Row

```

```
...
db.posts.update({ title: 'Post Two' },
{
  title: 'Post Two',
  body: 'New body for post 2',
  date: Date()
},
{
  upsert: true
})
...
```

Update Specific Field

```
...
db.posts.update({ title: 'Post Two' },
{
  $set: {
    body: 'Body for post 2',
    category: 'Technology'
  }
})
...
```

Increment Field (\$inc)

```
...
db.posts.update({ title: 'Post Two' },
{
  $inc: {
    likes: 5
  }
})
...
```

Rename Field

```
...
db.posts.update({ title: 'Post Two' },
{
  $rename: {
    likes: 'views'
  }
})
...
```

Delete Row

```
...
db.posts.remove({ title: 'Post Four' })
```

```
...
```

Sub-Documents

```
...
```

```
db.posts.update({ title: 'Post One' },
{
  $set: {
    comments: [
      {
        body: 'Comment One',
        user: 'Mary Williams',
        date: Date()
      },
      {
        body: 'Comment Two',
        user: 'Harry White',
        date: Date()
      }
    ]
  }
})
...
```

Find By Element in Array (\$elemMatch)

```
...
```

```
db.posts.find({
  comments: {
    $elemMatch: {
      user: 'Mary Williams'
    }
  }
})
...
```

Add Index

```
...
```

```
db.posts.createIndex({ title: 'text' })
...
```

Text Search

```
...
```

```
db.posts.find({
  $text: {
    $search: "\"Post O\""
  }
})
```

```
)  
```
```

## Greater & Less Than

```
```
```

```
db.posts.find({ views: { $gt: 2 } })  
db.posts.find({ views: { $gte: 7 } })  
db.posts.find({ views: { $lt: 7 } })  
db.posts.find({ views: { $lte: 7 } })  
```
```