

NAME :- MANALI Y. BHASKAR.

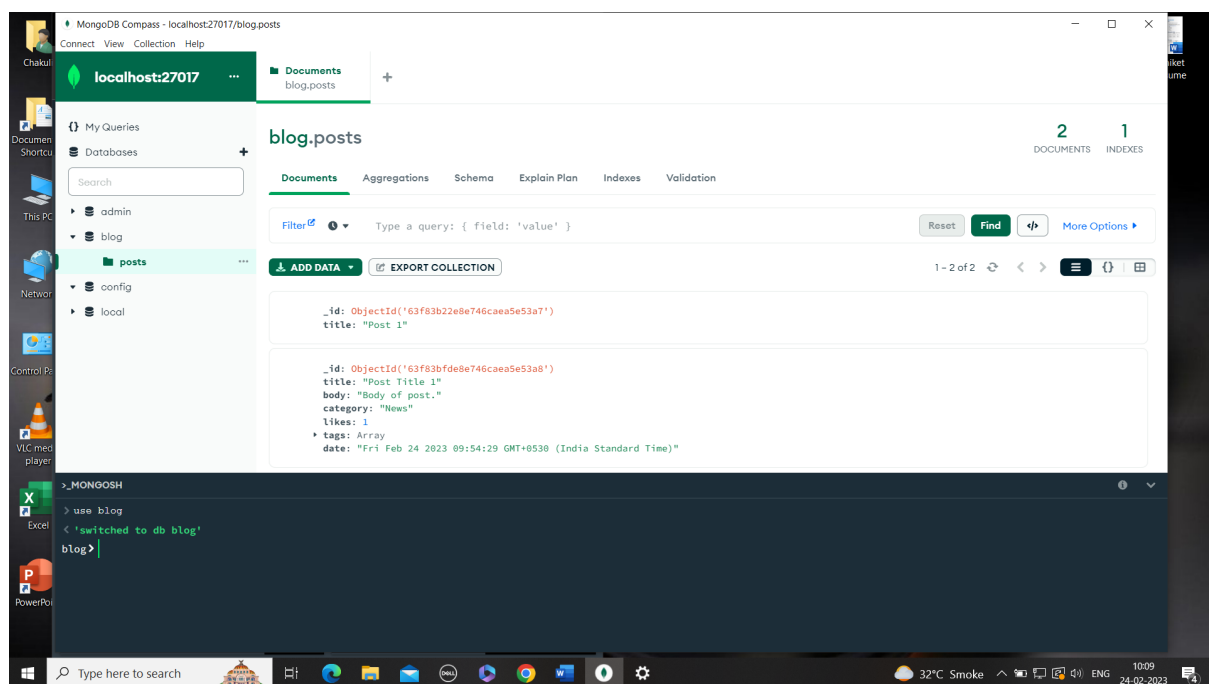
ROLL NO. :- 528

Mongodb-Performing queries

Mongodb:-

MongoDB is an open-source document-oriented database that is designed to store a large scale of data and also allows you to work with that data very efficiently. It is categorized under the NoSQL (Not only SQL) database because the storage and retrieval of data in the MongoDB are not in the form of tables.

STEP1 :- You can change or create a new database by typing **use** then the name of the database.



To insert a single document, use the `insertOne()` method.

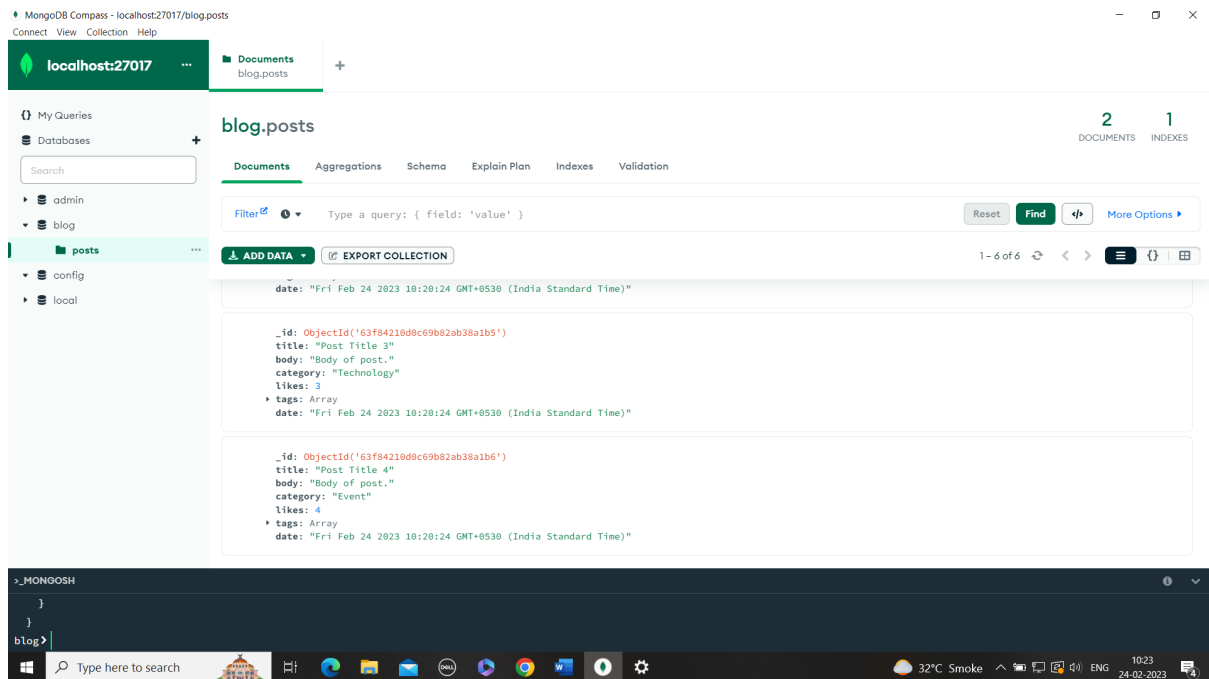
The screenshot shows the MongoDB Compass interface for the `blog.posts` collection. The left sidebar shows the database structure with `posts` selected. The main panel displays the `blog.posts` collection with 2 documents. The first document is expanded, showing its structure: `_id`, `title`, `body`, `category`, `likes`, `tags`, and `date`. Below the interface, a terminal window shows the command used to insert the document:

```
>_MONGOSH
> use blog
< 'switched to db blog'
> db.posts.insertOne({title: "Post Title 1",
  body: "Body of post.",
  category: "News",
  likes: 1,
  tags: ["news", "events"],
  date: Date()})
```

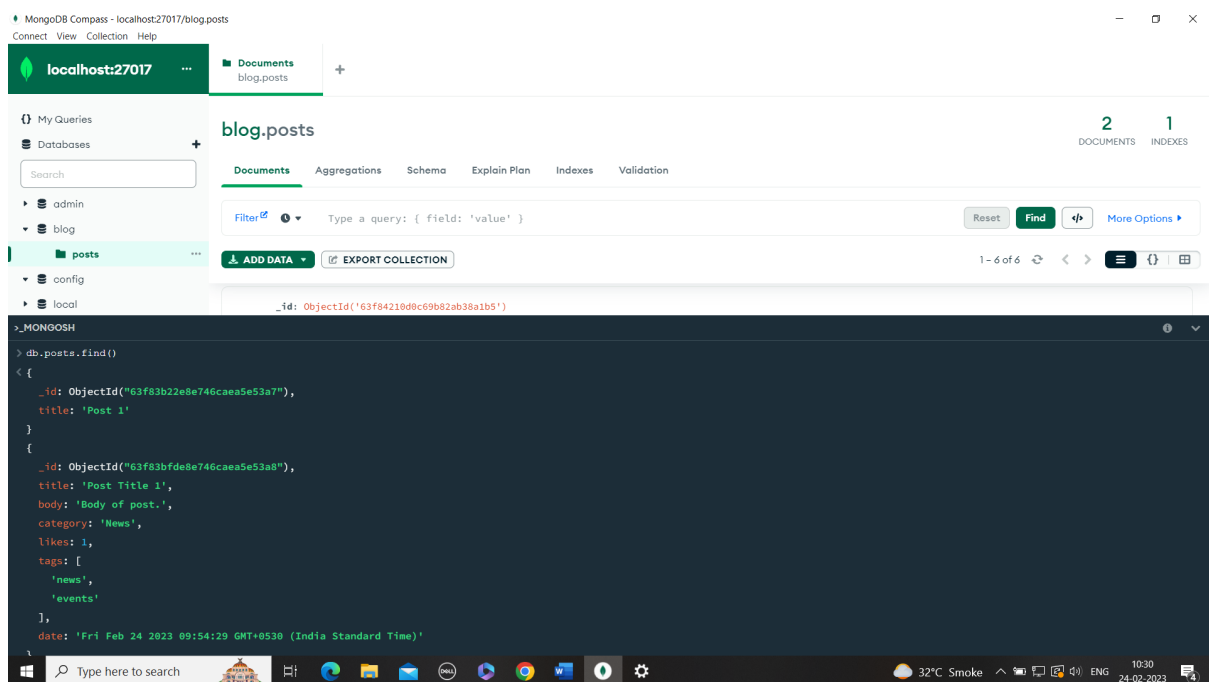
To insert multiple documents at once, use the `insertMany()` method.

The screenshot shows the MongoDB Compass interface for the `blog.posts` collection. The left sidebar shows the database structure with `posts` selected. The main panel displays the `blog.posts` collection with 2 documents. The first document is expanded, showing its structure: `_id`, `title`, `body`, `category`, `likes`, `tags`, and `date`. Below the interface, a terminal window shows the command used to insert multiple documents:

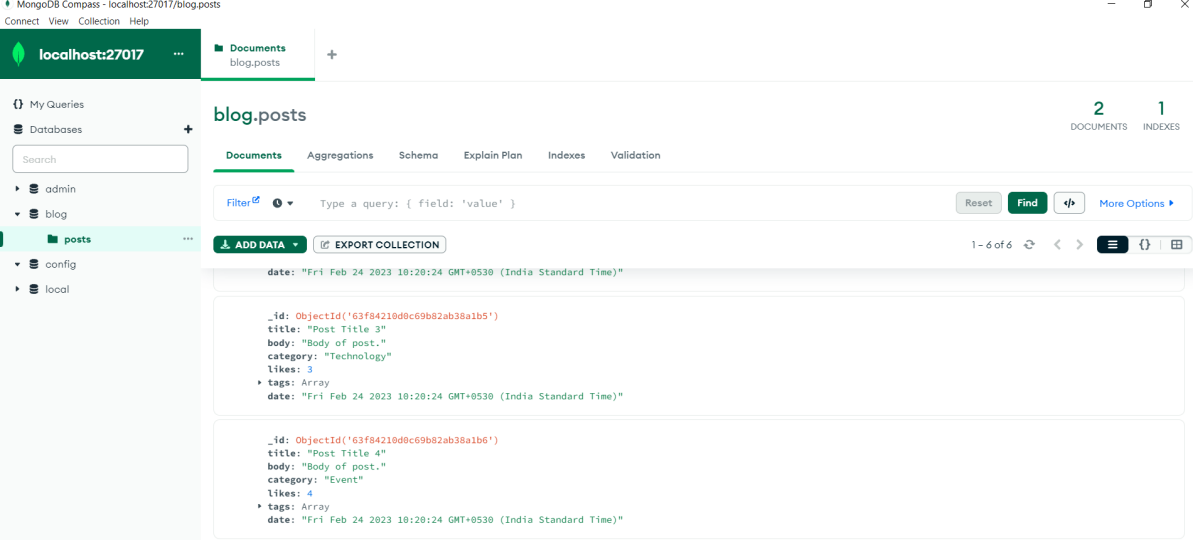
```
>_MONGOSH
> db.posts.insertMany([
  {
    title: "Post Title 2",
    body: "Body of post.",
    category: "Event",
    likes: 2,
    tags: ["news", "events"],
    date: Date()
  },
  {
    title: "Post Title 3",
    body: "Body of post.",
    category: "Technology",
    likes: 3,
    tags: ["news", "events"],
    date: Date()
  },
  {
    title: "Post Title 4",
```



To select data from a collection in MongoDB, we can use the `find()` method.

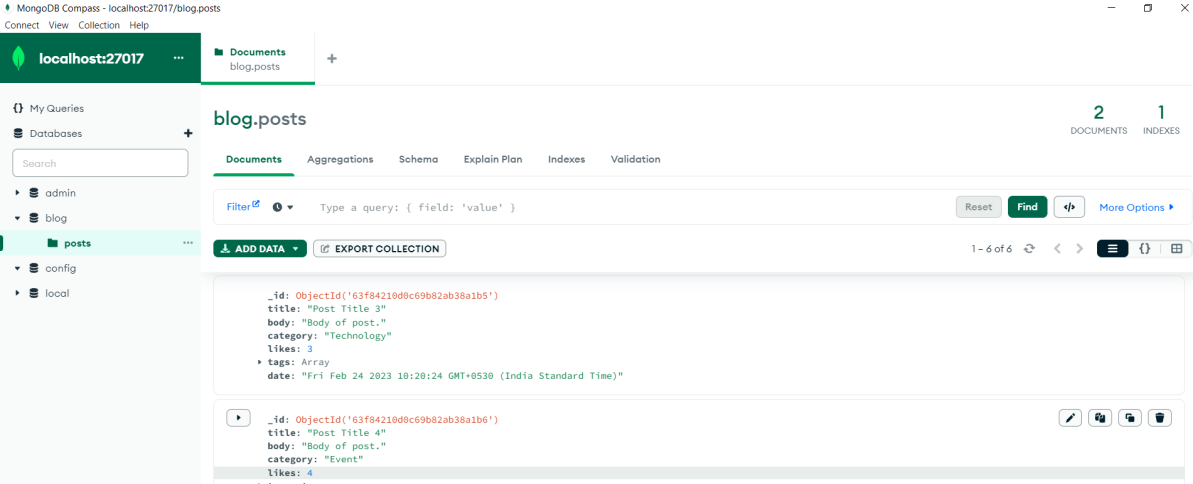


To select only one document, we can use the `findOne()` method.



The screenshot shows the MongoDB Compass interface for the 'blog.posts' collection. The left sidebar shows the database structure with 'posts' selected. The main area displays two documents in a list view. The first document has a title 'Post Title 3' and 3 likes. The second document has a title 'Post Title 4' and 4 likes. The top right corner shows 2 documents and 1 index. The bottom status bar shows the system time as 10:32 on 24-02-2023.

```
>_MONGOSH
{
}
> db.posts.find()
< {
  _id: ObjectId('63f84210d8c69b82ab38a1b5'),
  title: 'Post Title 3',
  body: 'Body of post.',
  category: 'Technology',
  likes: 3,
  tags: Array
    date: 'Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)'
  },
  _id: ObjectId('63f84210d8c69b82ab38a1b6'),
  title: 'Post Title 4',
  body: 'Body of post.',
  category: 'Event',
  likes: 4,
  tags: Array
    date: 'Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)'
  }
```



The screenshot shows the MongoDB Compass interface for the 'blog.posts' collection. The left sidebar shows the database structure with 'posts' selected. The main area displays two documents in a list view. The first document is selected, showing its details: title 'Post Title 3', body 'Body of post.', category 'Technology', and 3 likes. The second document is visible below it. The top right corner shows 2 documents and 1 index. The bottom status bar shows the system time as 10:34 on 24-02-2023.

```
>_MONGOSH
{
}
> db.posts.findOne()
< {
  _id: ObjectId('63f83b22e8e746caea5e53a7'),
  title: 'Post 1'
}
blog>
```

To query, or filter, data we can include a query in our `find()` or `findOne()` methods.

MongoDB Compass - localhost:27017/blog.posts

Connect View Collection Help

localhost:27017 Documents blog.posts

My Queries Databases Search

posts config local

blog.posts 2 DOCUMENTS 1 INDEXES

Documents Aggregations Schema Explain Plan Indexes Validation

Filter Type a query: { field: 'value' } Reset Find More Options

ADD DATA EXPORT COLLECTION

1 - 6 of 6

```
{
  "body": "Body of post.",
  "category": "Technology",
  "likes": 3,
  "tags": Array,
  "date": "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)"
}
```

```
{
  "_id": ObjectId('63f84210d8c69b82ab38a1b6'),
  "title": "Post Title 4",
  "body": "Body of post.",
  "category": "Event",
  "likes": 4,
  "tags": Array,
  "date": "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)"
}
```

>_MONGOSH

```
}
> db.posts.find( {category: "News"} )
< [
  {
    "_id": ObjectId("63f83bfde8e746caea5e53a8"),
    "title": "Post Title 1",
    "body": "Body of post.",
    "category": "News",
  }
]
```

Type here to search 32°C Smoke ENG 10:38 24-02-2023

This example will only display the **title** and **date** fields in the results.

MongoDB Compass - localhost:27017/blog.posts

Connect View Collection Help

localhost:27017 Documents blog.posts

My Queries Databases Search

posts config local

blog.posts 2 DOCUMENTS 1 INDEXES

Documents Aggregations Schema Explain Plan Indexes Validation

Filter Type a query: { field: 'value' } Reset Find More Options

ADD DATA EXPORT COLLECTION

1 - 6 of 6

```
{
  "body": "Body of post.",
  "category": "Technology",
  "likes": 3,
  "tags": Array,
  "date": "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)"
}
```

```
{
  "_id": ObjectId('63f84210d8c69b82ab38a1b6'),
  "title": "Post Title 4",
  "body": "Body of post.",
  "category": "Event",
  "likes": 4,
  "tags": Array,
  "date": "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)"
}
```

>_MONGOSH

```
}
> db.posts.find({}, {title: 1, date: 1})
< [
  {
    "_id": ObjectId("63f83b22e8e746caea5e53a7"),
    "title": "Post 1",
  }
]
```

Type here to search 32°C Smoke ENG 10:40 24-02-2023

We will get an error if we try to specify both 0 and 1 in the same object.

MongoDB Compass - localhost:27017/blog.posts

Connect View Collection Help

localhost:27017 Documents blog.posts

My Queries Databases

Search

admin blog posts config local

blog.posts 2 DOCUMENTS 1 INDEXES

Documents Aggregations Schema Explain Plan Indexes Validation

Filter Type a query: { field: 'value' } Reset Find More Options

ADD DATA EXPORT COLLECTION

1 - 6 of 6

body: "Body of post."
category: "Technology"
likes: 3
tags: Array
date: "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)"

_id: ObjectId('63f84210d9c69b82ab38a1b6')
title: "Post Title 4"
body: "Body of post."
category: "Event"
likes: 4
tags: Array
date: "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)"

>_MONGOSH

```
}  
db.posts.find( { title: "Post Title 1" } )  
{  
  _id: ObjectId("63f83bfde8e746caea5e53a8"),  
  title: 'Post Title 1',  
  body: 'Body of post.',  
  category: 'News',  
}
```

Type here to search 32°C Smoke 1044 24-02-2023

MongoDB Compass - localhost:27017/blog.posts

Connect View Collection Help

localhost:27017 Documents blog.posts

My Queries Databases

Search

admin blog posts config local

blog.posts 2 DOCUMENTS 1 INDEXES

Documents Aggregations Schema Explain Plan Indexes Validation

Filter Type a query: { field: 'value' } Reset Find More Options

ADD DATA EXPORT COLLECTION

1 - 6 of 6

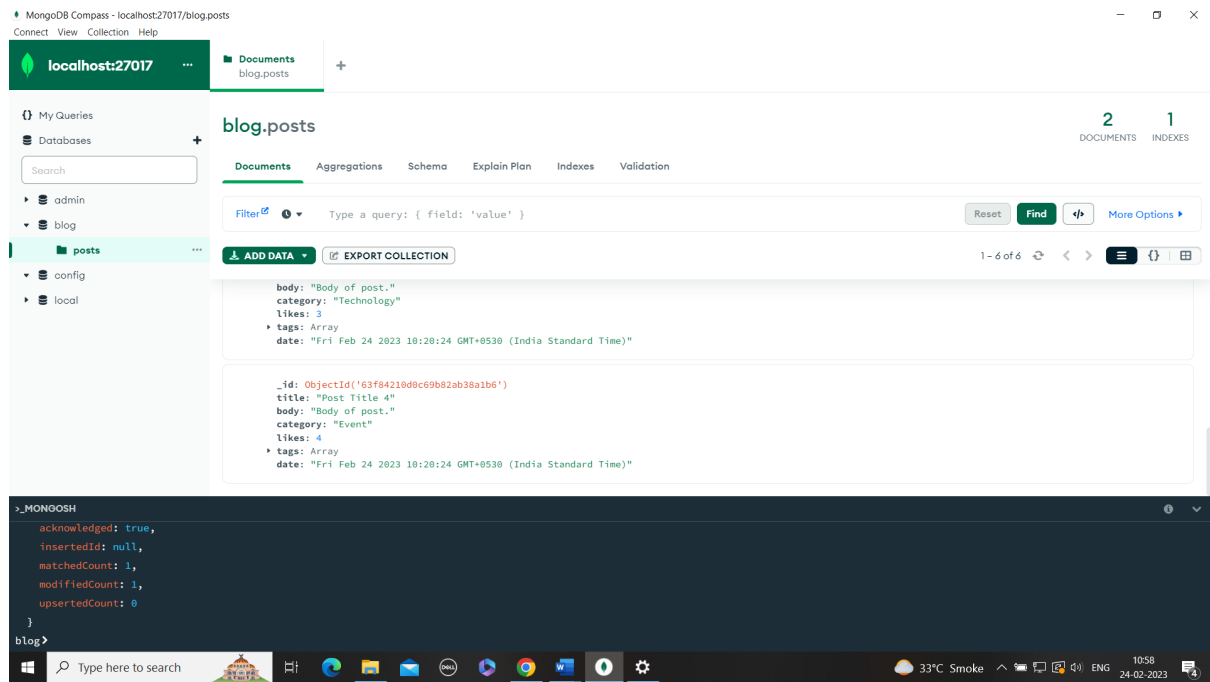
body: "Body of post."
category: "Technology"
likes: 3
tags: Array
date: "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)"

_id: ObjectId('63f84210d9c69b82ab38a1b6')
title: "Post Title 4"
body: "Body of post."
category: "Event"
likes: 4
tags: Array
date: "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)"

>_MONGOSH

```
}  
db.posts.find( { title: "Post Title 1" } )  
{  
  _id: ObjectId("63f83bfde8e746caea5e53a8"),  
  title: 'Post Title 1',  
  body: 'Body of post.',  
  category: 'News',  
}
```

Type here to search 33°C Smoke 1056 24-02-2023



If you would like to insert the document if it is not found, you can use the **upsert** option.

MongoDB Compass - localhost:27017/blog.posts

Connect View Collection Help

localhost:27017 Documents blog.posts

My Queries Databases Search

admin blog posts config local

blog.posts 2 DOCUMENTS 1 INDEXES

Documents Aggregations Schema Explain Plan Indexes Validation

Filter Type a query: { field: 'value' } Reset Find More Options

ADD DATA EXPORT COLLECTION

1 - 6 of 6

```
{
  category: "Event",
  likes: 4,
  tags: Array,
  date: "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)"
}
```

```
>_MONGOOSH
> db.posts.updateOne(
  { title: "Post Title 5" },
  {
    $set: {
      title: "Post Title 5",
      body: "Body of post.",
      category: "Event",
      likes: 5,
      tags: ["news", "events"],
      date: Date()
    }
  },
  { upsert: true }
)
```

Type here to search 33°C Smoke ENG 11:03 24-02-2023

MongoDB Compass - localhost:27017/blog.posts

Connect View Collection Help

localhost:27017 Documents blog.posts

My Queries Databases Search

admin blog posts config local

blog.posts 2 DOCUMENTS 1 INDEXES

Documents Aggregations Schema Explain Plan Indexes Validation

Filter Type a query: { field: 'value' } Reset Find More Options

ADD DATA EXPORT COLLECTION

1 - 7 of 7

```
{
  tags: Array,
  date: "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)"
}
```

```
{
  _id: ObjectId('63f84b8d3938ec14d902418'),
  title: "Post Title 5",
  body: "Body of post.",
  category: "Event",
  date: "Fri Feb 24 2023 11:00:41 GMT+0530 (India Standard Time)",
  likes: 5,
  tags: Array
}
```

```
>_MONGOOSH
{
  acknowledged: true,
  insertedId: ObjectId("63f84b8d3938ec14d902418"),
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 1
}
blog>
```

Type here to search 33°C Smoke ENG 11:07 24-02-2023

The `updateMany()` method will update all documents that match the provided query.

The screenshot shows the MongoDB Compass interface for the `localhost:27017/blog.posts` database. The left sidebar shows the database structure with `posts` selected. The main panel displays the `blog.posts` collection with 2 documents and 1 index. The `Documents` tab is active, showing a list of documents. The first document is expanded, showing its JSON structure: `{ "_id": ObjectId("63f84b81d3938ec14d982418"), "title": "Post Title 5", "body": "Body of post.", "category": "Event", "date": "Fri Feb 24 2023 10:20:24 GMT+0530 (India Standard Time)", "likes": 6, "tags": Array }`. Below the document list, the `_MONGOSH` terminal shows the command `db.posts.updateMany({}, { $inc: { likes: 1 } })` and its output: `{ acknowledged: true, insertedId: null, matchedCount: 7, modifiedCount: 7, upsertedCount: 0 }`. The bottom status bar shows the system clock as 11:11 on 24-02-2023.

The `deleteOne()` method will delete the first document that matches the query provided.

The screenshot shows the MongoDB Compass interface for the `localhost:27017/blog.posts` database. The left sidebar shows the database structure with `posts` selected. The main panel displays the `blog.posts` collection with 2 documents and 1 index. The `Documents` tab is active, showing a list of documents. The first document is expanded, showing its JSON structure: `{ "_id": ObjectId("63f84210d9c69b82ab38a1b6"), "title": "Post Title 4", "body": "Body of post.", "category": "Event", "likes": 5, "tags": Array }`. Below the document list, the `_MONGOSH` terminal shows the command `db.posts.deleteOne({ title: "Post Title 5" })` and its output: `{ acknowledged: true, deletedCount: 1 }`. The bottom status bar shows the system clock as 11:13 on 24-02-2023.

STEP :- The `deleteMany()` method will delete all documents that match the query provided.

The screenshot shows the MongoDB Compass interface for a local database at localhost:27017. The 'blog.posts' collection is selected, and the 'Documents' tab is active. The collection contains 2 documents. A filter query is entered: `{ field: 'value' }`. Below the filter, there are buttons for 'ADD DATA' and 'EXPORT COLLECTION'. The document list shows two documents, each with a 'tags' array and a 'date' field. The bottom terminal window shows the execution of the `deleteMany()` command in the MongoDB shell:

```
> _MONGOSH
acknowledged: true,
deletedCount: 1
}
> db.posts.deleteMany({ category: "Technology" })
< {
  acknowledged: true,
  deletedCount: 0
}
blog>
```

The Windows taskbar at the bottom shows the date as 24-02-2023 and the time as 11:14.