MongoDB Shell Commands & Outputs

Create a Database

```
> use Blog
switched to db Blog
```

Create a Collection

```
> db.createCollection("posts")
{ ok: 1 }
```

Insert First Document

```
> db.posts.insertOne({"title":"my 1st post"})
{
  acknowledged: true,
  insertedId: ObjectId("6655fa3c4a3c6f1f2a7e2b10")
}
```

Insert a Full Document with Fields

```
> db.posts.insertOne({
...   title: "Post Title 1",
...   body: "Body of post.",
...   category: "Sports",
...   likes: 1,
...   tags: ["news", "Sports"],
...   date: Date()
... })
{
  acknowledged: true,
  insertedId: ObjectId("6655fa6f4a3c6f1f2a7e2b11")
}
```

Find All Documents

```
> db.posts.find()
[
  { _id: ObjectId("6655fa3c4a3c6f1f2a7e2b10"), title: "my 1st post" },
  {
    _id: ObjectId("6655fa6f4a3c6f1f2a7e2b11"),
    title: "Post Title 1",
    body: "Body of post.",
    category: "Sports",
    likes: 1,
    tags: [ "news", "Sports" ],
    date: "Tue May 28 2025 14:10:39 GMT+0530 (India Standard Time)"
  }
]
```

Invalid findOne() Usage

```
> db.posts.findOne("my 1st post")
MongoInvalidArgumentError: Query filter must be a plain object or ObjectId
```

Correct Find by Category

```
> db.posts.find({category:"Sports"})
[
  {
    _id: ObjectId("6655fa6f4a3c6f1f2a7e2b11"),
    title: "Post Title 1",
    body: "Body of post.",
    category: "Sports",
    likes: 1,
    tags: [ "news", "Sports" ],
    date: "Tue May 28 2025 14:10:39 GMT+0530 (India Standard Time)"
  }
]
```

Projection - Include Only Title and Date

```
> db.posts.find({}, {title: 1, date: 1})
[
  { _id: ObjectId("6655fa3c4a3c6f1f2a7e2b10"), title: "my 1st post" },
  {
    _id: ObjectId("6655fa6f4a3c6f1f2a7e2b11"),
    title: "Post Title 1",
    date: "Tue May 28 2025 14:10:39 GMT+0530 (India Standard Time)"
  }
]
```

Projection - Exclude _id

```
> db.posts.find({}, {_id: 0, title: 1, date: 1})
[
  { title: "my 1st post" },
```

MongoDB Shell Commands & Outputs

```
  {
    title: "Post Title 1",
    date: "Tue May 28 2025 14:10:39 GMT+0530 (India Standard Time)"
  }
]
```

Update One Document

```
> db.posts.updateOne( { title: "Post Title 1" }, { $set: { likes: 20 } } )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

Update Many Documents

```
> db.posts.updateMany( { category: "Sports" }, { $set: { likes: 99 } } )
{
  acknowledged: true,
  matchedCount: 1,
  modifiedCount: 1
}
```

Upsert - Update or Insert if Not Exists

```
> db.posts.updateOne(
...   { title: "Post Title 3" },
...   {
...     $set: {
...       title: "Post Title 3",
...       body: "Body of post.",
...       category: "Sports",
...       likes: 50,
...       tags: ["news", "Sports"],
...       date: Date()
...     }
...   },
...   { upsert: true }
... )
{
```

MongoDB Shell Commands & Outputs

```
  acknowledged: true,
  insertedId: ObjectId("6655fb1c4a3c6f1f2a7e2b12"),
  matchedCount: 0,
  modifiedCount: 0,
  upsertedCount: 1
}
```

Delete One Document

```
> db.posts.deleteOne({ title: "Post Title 5" })
{ acknowledged: true, deletedCount: 0 }

(Assuming "Post Title 5" doesn't exist)
```

Delete Many Documents

```
> db.posts.deleteMany({ category: "Sports" })
{ acknowledged: true, deletedCount: 2 }

(Assuming 2 posts with category "Sports" were present)
```