MUMBAI EDUCATIONAL TRUST

MET Institute of Computer Science

THE MET LEAGUE OF COLLEGES
MET
AS SHARP AS YOU CAN GET
Bhujbal Knowledge City

Program No:	1
Roll No :	1545
Title of Program :	MongoDb commands
Objective :	Perform basic operations in MongoDB database

1) Data Definition

Create a database:

Creating a database is dynamic
use command - create new database and switch to that database and if database already exists then existing database is used
cmd: use TeacherDB1

```
> use TeachersDB1
switched to db TeachersDB1
```

```
> show dbs
EmpDb       0.000GB
RetailDb    0.000GB
SalesDb     0.000GB
StudentDb   0.000GB
admin       0.000GB
config      0.000GB
local       0.000GB
mca         0.000GB
>
```

Creating a Collection:

Creating a database itself does not allocate the memory you have to create a collection

```
> db.TeacherColl.insert(
... {
... "tid": 101,
... "tname": "Om"
... }
... );
WriteResult({ "nInserted" : 1 })
```

Inserting one document

```
> db.TeacherColl.insertOne(
... {
... "tid": 102,
... "tname": "om1",
... "salary": 20000
... });
{
        "acknowledged" : true,
        "insertedId" : ObjectId("686b69b328a22fc429746e47")
}
```

Inserting many documents

```
> db.TeacherColl.insertMany( [ { "tid": 103, "tname": "ca", "salary": 25000 }, { "tid": 104, "tname": "fg", "salary": 20
000, "experience": 20 }, { "tid": 105, "tname": "pg" }]);
{
        "acknowledged" : true,
        "insertedIds" : [
                ObjectId("686b6ae128a22fc429746e48"),
                ObjectId("686b6ae128a22fc429746e49"),
                ObjectId("686b6ae128a22fc429746e4a")
        ]
}
```

To see list of collections

```
> show collections
TeacherColl
```

To display the records

```
> db.TeacherColl.find();
{ "_id" : ObjectId("686b686e28a22fc429746e46"), "tid" : 101, "tname" : "Om" }
{ "_id" : ObjectId("686b69b328a22fc429746e47"), "tid" : 102, "tname" : "om1", "salary" : 20000 }
{ "_id" : ObjectId("686b6ae128a22fc429746e48"), "tid" : 103, "tname" : "ca", "salary" : 25000 }
{ "_id" : ObjectId("686b6ae128a22fc429746e49"), "tid" : 104, "tname" : "fg", "salary" : 20000, "experience" : 20 }
{ "_id" : ObjectId("686b6ae128a22fc429746e4a"), "tid" : 105, "tname" : "pg" }
```

To display the records in json format

```
> db.TeacherColl.find().pretty();
{
        "_id" : ObjectId("686b686e28a22fc429746e46"),
        "tid" : 101,
        "tname" : "Om"
}
{
        "_id" : ObjectId("686b69b328a22fc429746e47"),
        "tid" : 102,
        "tname" : "om1",
        "salary" : 20000
}
{
        "_id" : ObjectId("686b6ae128a22fc429746e48"),
        "tid" : 103,
        "tname" : "ca",
        "salary" : 25000
}
{
        "_id" : ObjectId("686b6ae128a22fc429746e49"),
        "tid" : 104,
        "tname" : "fg",
        "salary" : 20000,
        "experience" : 20
}
{
        "_id" : ObjectId("686b6ae128a22fc429746e4a"),
        "tid" : 105,
        "tname" : "pg"
}
> |
```

Creating a Collection statically

MUMBAI EDUCATIONAL TRUST

MET Institute of Computer Science

विद्याधनं सर्वधन प्रधानम्।

THE MET LEAGUE OF COLLEGES

MET

AS SHARP AS YOU CAN GET

Bhujbal Knowledge City

```
> db.createCollection("Students")
{ "ok" : 1 }
> show collections
Students
TeacherColl
```

Gives error when we give same _id to two documents

```
> db.Students.insertOne(
... {
... "_id": 101,
... "sname": "shivam",
... });
{ "acknowledged" : true, "insertedId" : 101 }
> db.Students.insertOne(
... {
... "_id": 101,
... "sname": "ajay",
... });
2025-07-07T12:23:14.602+0530 E QUERY    [js] WriteError: E11000 duplicate key error collection:
 TeachersDB1.Students index: _id_ dup key: { : 101.0 } :
WriteError({
        "index" : 0,
        "code" : 11000,
        "errmsg" : "E11000 duplicate key error collection: TeachersDB1.Students index: _id_ dup
 key: { : 101.0 }",
        "op" : {
                "_id" : 101,
                "sname" : "ajay"
        }
})
```

```
> db.Students.insertOne( { "sid": 101, "sname": "ajay", });
{
        "acknowledged" : true,
        "insertedId" : ObjectId("686b6f2b28a22fc429746e4b")
}
> db.Students.find().pretty();
{ "_id" : 101, "sname" : "shivam" }
{
        "_id" : ObjectId("686b6f2b28a22fc429746e4b"),
        "sid" : 101,
        "sname" : "ajay"
}
```

MUMBAI EDUCATIONAL TRUST

MET Institute of Computer Science

विद्याधनं सर्वधन प्रधानम्।

THE MET LEAGUE OF COLLEGES

MET
AS SHARP AS YOU CAN GET

Bhujbal Knowledge City

2) Data Manipulation

$gt	greater than
$lt	less than
$gte	greater than or equal to
$lte	less than or equal to
$eq	equality
$ne	not equal to

Write a query to find all the teacher earning salary greater than 20000

```
> db.TeacherColl.find(
... {
... "salary": {$gt: 20000}
... });
{ "_id" : ObjectId("686b6ae128a22fc429746e48"), "tid" : 103, "tname" : "ca", "salary" : 25000 }
```

Write a query to find all the teacher earning salary less than 30000

```
> db.TeacherColl.find( { "salary": {$lt: 30000} });
{ "_id" : ObjectId("686b69b328a22fc429746e47"), "tid" : 102, "tname" : "om1", "salary" : 20000
}
{ "_id" : ObjectId("686b6ae128a22fc429746e48"), "tid" : 103, "tname" : "ca", "salary" : 25000 }
{ "_id" : ObjectId("686b6ae128a22fc429746e49"), "tid" : 104, "tname" : "fg", "salary" : 20000,
"experience" : 20 }
```

Write a query to find all the teacher earning salary 20000

```
> db.TeacherColl.find( { "salary": {$eq: 20000} });
{ "_id" : ObjectId("686b69b328a22fc429746e47"), "tid" : 102, "tname" : "om1", "salary" : 20000
}
{ "_id" : ObjectId("686b6ae128a22fc429746e49"), "tid" : 104, "tname" : "fg", "salary" : 20000,
"experience" : 20 }
```

MUMBAI EDUCATIONAL TRUST

MET Institute of Computer Science

THE MET LEAGUE OF COLLEGES
MET
AS SHARP AS YOU CAN GET
Bhujbal Knowledge City

विद्याधनं सर्वधनं प्रधानम्।

Write a query to find all the teachers who does not have experience of 20

```
> db.TeacherColl.find( { "experience": {$ne: 20} });
{ "_id" : ObjectId("686b686e28a22fc429746e46"), "tid" : 101, "tname" : "Om" }
{ "_id" : ObjectId("686b69b328a22fc429746e47"), "tid" : 102, "tname" : "om1", "salary" : 20000
}
{ "_id" : ObjectId("686b6ae128a22fc429746e48"), "tid" : 103, "tname" : "ca", "salary" : 25000 }
{ "_id" : ObjectId("686b6ae128a22fc429746e4a"), "tid" : 105, "tname" : "pg" }
```

Find teacher with id 103

```
> db.TeacherColl.find(
... {
... "tid": 103
... });
{ "_id" : ObjectId("686b6ae128a22fc429746e48"), "tid" : 103, "tname" : "ca", "salary" : 25000 }
>
```

Display all teacher info except 105

```
> db.TeacherColl.find( { "tid": {$ne: 105 } });
{ "_id" : ObjectId("686b686e28a22fc429746e46"), "tid" : 101, "tname" : "Om" }
{ "_id" : ObjectId("686b69b328a22fc429746e47"), "tid" : 102, "tname" : "om1", "salary" : 20000
}
{ "_id" : ObjectId("686b6ae128a22fc429746e48"), "tid" : 103, "tname" : "ca", "salary" : 25000 }
{ "_id" : ObjectId("686b6ae128a22fc429746e49"), "tid" : 104, "tname" : "fg", "salary" : 20000,
"experience" : 20 }
```

Find all the teacher whose salary is greater than 30000 or experience greater than 15 years

```
> db.TeacherColl.find(
... {
... $or: [{"salary": {$gt: 30000}}, {"experience": {$gt: 15}}]
... });
{ "_id" : ObjectId("686b6ae128a22fc429746e49"), "tid" : 104, "tname" : "fg", "salary" : 20000,
"experience" : 20 }
>
```

Find all the teacher whose salary is greater than 10000 and experience greater than 15 years

```
> db.TeacherColl.find( { $and: [ {"salary": {$gt: 10000}}, {"experience": {$gt: 15}} ]});
{ "_id" : ObjectId("686b6ae128a22fc429746e49"), "tid" : 104, "tname" : "fg", "salary" : 20000, "experience" : 20 }
```