

MONGODB

## **ADD , UPDATE AND DELETE DATA**

First we need to give use db(database)

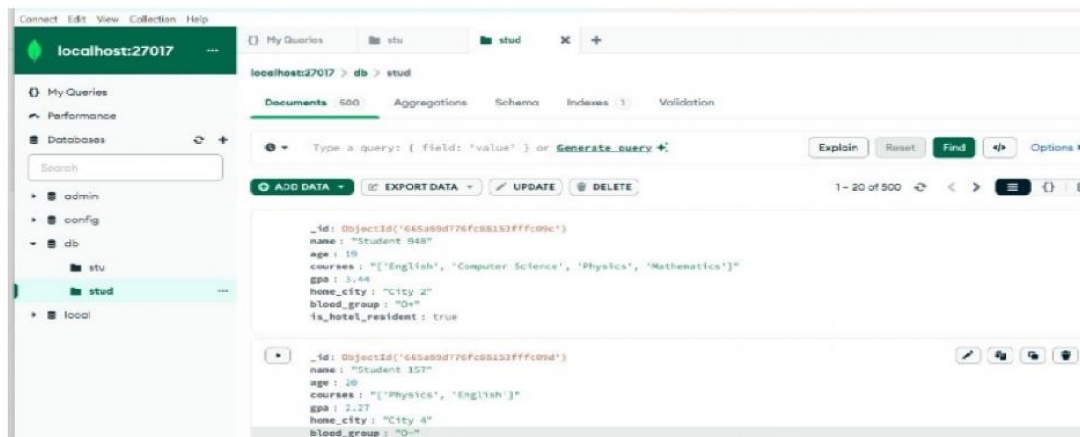
```
test> use db  
switched to db db
```

Now ,as we can observe the database is switched to db

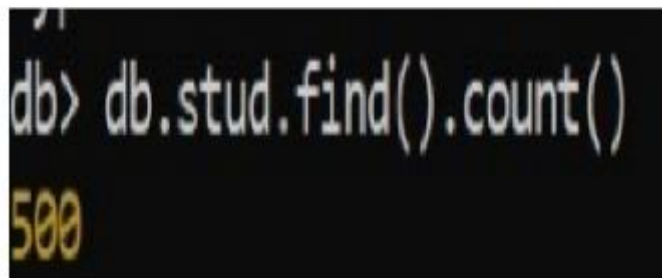
To find whether the data present in the given collection, here collection name is about the information of students.

```
test> use db  
switched to db db  
db> show collections  
stu  
stud
```

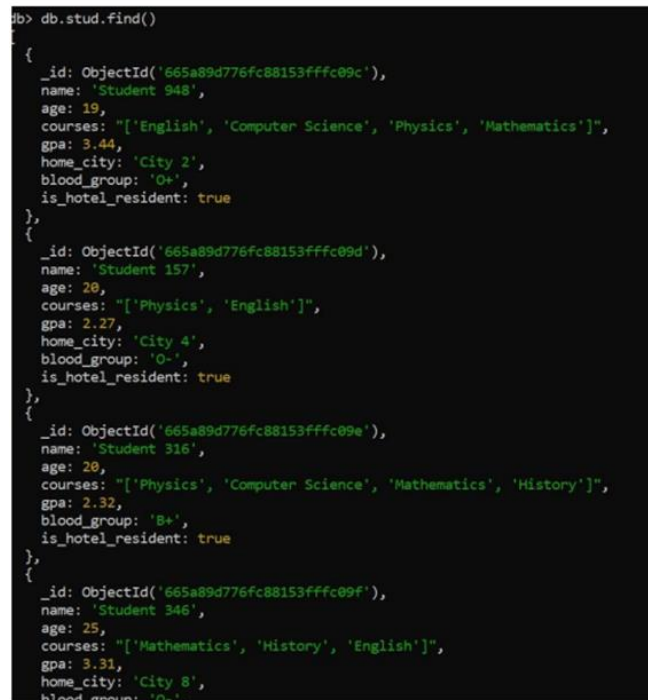
To find the total number of collection of the database used in command. “db.stud.find().count()”.



The below example the collection name is stud or stu.



To find the collection of the database use the command.  
“db.stud.find()”



## WHERE, AND, OR & CRUD:

### WHERE:

Given a collection you want to filter a subset based on a condition. That is the place WHERE is used. To find all students with GPA greater than 3.5, we use command-

**"db.stud.find({gpa:{\$gt:3.5}});"**

```
db> db.stud.find({gpa:{$gt:3.5}});
[
  {
    _id: ObjectId('665a89d776fc88153fffc0a0'),
    name: 'Student 930',
    age: 25,
    courses: "['English', 'Computer Science', 'Mathematics', 'History']",
    gpa: 3.63,
    home_city: 'City 3',
    blood_group: 'A-',
    is_hotel_resident: true
  },
  {
    _id: ObjectId('665a89d776fc88153fffc0a2'),
    name: 'Student 268',
    age: 21,
    courses: "['Mathematics', 'History', 'Physics']",
    gpa: 3.98,
  },
]
db> db.stud.find({gpa:{$gt:3.5}});
[
  {
    is_hotel_resident: false
  },
  {
    _id: ObjectId('665a89d776fc88153fffc0a0'),
    name: 'Student 930',
    age: 25,
  },
  {
    _id: ObjectId('665a89d776fc88153fffc0ac'),
    name: 'Student 368',
    age: 25,
  },
]
```

Here \$gt represent the greater than, and it gives the information about students that are belongs to greater than 3.5 gpa.

## AND:

Given a collection you want to filter a subset based on multiple conditions. To find all students who live in “City 5” AND have a based group of “A+” Here we use the command:

`Db.stud.find({ $and: [ {home_city : “City 5”}, {blood_group: “A+”} ] });`

```
db> db.stud.find({
... $and:[
... {home_city:"City 5"},
... {blood_group:"A+"}
... ]
... });
[
  {
    _id: ObjectId('665a89d776fc88153fffc0d3'),
    name: 'Student 142',
    age: 24,
    courses: "['History', 'English', 'Physics', 'Computer Science']",
    gpa: 3.41,
    home_city: 'City 5',
    blood_group: 'A+',
    is_hotel_resident: false
  },
  {
    _id: ObjectId('665a89d776fc88153fffc1f3'),
    name: 'Student 947',
    age: 20,
    courses: "['Physics', 'History', 'English', 'Computer Science']",
    home_city: 'City 5',
    blood_group: 'A+',
    is_hotel_resident: true
  },
  {
    _id: ObjectId('665a89d776fc88153fffc265'),
    name: 'Student 567',
    age: 22,
    courses: "['Computer Science', 'History', 'English', 'Mathematics']",
    gpa: 2.01,
    home_city: 'City 5',
    blood_group: 'A+',
    is_hotel_resident: true
  }
]
```

Above shown example is filtered based on some particular conditions like:

‘home\_city: City5’ and ‘blood\_group : A+’

## OR:

In the given collection that is student information , we want to filter a subset based on multiple conditions .

```
db.stud.find({ $or: [ { blood_group: "A+" }, { gpa: { $gt: 3.5 } } ] })

{
  _id: ObjectId('665a89d776fc88153fffc0a0'),
  name: 'Student 930',
  age: 25,
  courses: "['English', 'Computer Science', 'Mathematics', 'History']",
  gpa: 3.63,
  home_city: 'City 3',
  blood_group: 'A-',
  is_hotel_resident: true
},
{
  _id: ObjectId('665a89d776fc88153fffc0a2'),
  name: 'Student 268',
  age: 21,
  courses: "['Mathematics', 'History', 'Physics']",
  gpa: 3.98,
  blood_group: 'A+',
  is_hotel_resident: false
},
{
  _id: ObjectId('665a89d776fc88153fffc0a7'),
  name: 'Student 177',
  age: 23,
  courses: "['Mathematics', 'Computer Science', 'Physics']",
  gpa: 2.52,
  home_city: 'City 10',
  blood_group: 'A+',
  is_hotel_resident: true
},
{
  _id: ObjectId('665a89d776fc88153fffc0ac'),
  name: 'Student 368',
  age: 20,
  courses: "['English', 'History', 'Physics', 'Computer Science']",
  gpa: 3.91,
```

In the above example , the student database is filtered based on either 'blood\_group : A+' or 'gpa great.

## Selectors:

Comparison **gt** and **lt** AND operator OR operator.

### Comparison gt lt:

To find all students with age **greater than 20**.

```

db> db.stud.find({age:{$gt:20}});
[
  {
    _id: ObjectId('665a89d776fc88153fffc09f'),
    name: 'Student 346',
    age: 25,
    courses: "['Mathematics', 'History', 'English']",
    gpa: 3.31,
    home_city: 'City 8',
    blood_group: 'O-',
    is_hotel_resident: true
  },
  {
    _id: ObjectId('665a89d776fc88153fffc0a0'),
    name: 'Student 930',
    age: 25,
    courses: "['English', 'Computer Science', 'Mathematics', 'History']",
    gpa: 3.63,
    home_city: 'City 3',
    blood_group: 'A-',
    is_hotel_resident: true
  },
  {
    _id: ObjectId('665a89d776fc88153fffc0a1'),
    name: 'Student 305',
    age: 24,
    courses: "['History', 'Physics', 'Computer Science', 'Mathematics']",
    gpa: 3.4,
    home_city: 'City 6',
    blood_group: 'O+',
    is_hotel_resident: true
  }
]

```

## AND operator:

To find students from “city 2” with blood group “B+”.

```

db.stud.find({
  $and:[
    {home_city: "City 2"},
    {blood_group: "B+" }
  ]
});

_id: ObjectId('665a89d776fc88153fffc0b4'),
name: 'Student 504',
age: 21,
courses: "['Physics', 'Computer Science', 'English', 'Mathematics']",
gpa: 2.42,
home_city: 'City 2',
blood_group: 'B+',
is_hotel_resident: true
,

_id: ObjectId('665a89d776fc88153fffc0eb'),
name: 'Student 367',
age: 19,
courses: "['English', 'Physics', 'History', 'Mathematics']",
gpa: 2.81,
home_city: 'City 2',
blood_group: 'B+',
is_hotel_resident: false

```

## OR Operator:

MongoDB provides different types of logical query operators and **\$or** operator is one of them.

### Syntax:

**{ \$or: [ { Expression1 }, { Expression2 }, ..., { Expression3 } }**

```
db> db.student.find({ $or: [ { is_hotel_resident: true }, { gpa: { $lt: 3.0 } } ] });
[
  {
    _id: ObjectId('665de5dd6e26f71bef17d067'),
    name: 'Student 157',
    age: 20,
    courses: "['Physics', 'English']",
    gpa: 2.27,
    home_city: 'City 4',
    blood_group: 'O-',
    is_hotel_resident: true
  },
  ...
]
```