use mydb1	to create a database, and switch to the database
db.dropDatabase()	to drop the current database
show dbs	to list all databases
show collections	to list all collections
db.employee.insertOne({empno:1,ename:'Rajan'})	to insert one document in employee collection, if collection doesnot exists it will create a collection and then add the document
db.employee.insertMany([{empno:1,eaname:'Revati',dept:{deptno:11,dname:'Admin'}}, { empno:2,eaname:'Geet',sal:45678},{ empno:3,eaname:'Isha'}])	to add many documents in a collection

In mongodb data and functions are casesensitive

In mongodb there are 2 types of collection

1. capped collection-→while creating collection if you want to keep a limit on how many maximum documents a collection can store, then it is called as capped collection. We cannot use deleteOne, deleteMany function to delete documents from capped collection

db.createcollection(“bolgs”,{capped:true,max:100,size:50000,autoIndexId:true})

capped :true	It represents that there is a limit on number of documents
max:100	It represents that in a collection maximum 100 documents can be stored
size:50000	It is number of maximum bytes assigned to store max number of documents in the collection
autoIndexId:true	It is optional property, by default index will get created on primary key (_id)

2. collection-→ while creating collection if there is no limit on how many maximum documents a collection can store, then it is called as collection.

   db.createcollection(“emp”)-→ it will create a empty collection

3. create a capped collection mytest to store maximum 15 documents maximum bytes should be 20000
   db.createCollection(“mytest”,{capped:true,max:15,size:20000})

   To delete the data

db.emp.deleteOne({})
db.emp.deleteOne({<condition>})
	It will delete first document from the collection if the condition is not given,
	If the condition is given , then it will delete the first matching document

db.emp.deleteMany({})
db.emp.deleteMany({<condition>})
	It will delete all documents from the given collection if the condition is not given,
	If the condition is given , then it will delete the all matching document

db.emp.remove({})
db.emp.remove({condition})
	It will delete all documents from the given collection if the condition is not given,
	If the condition is given , then it will delete the all matching document

   To find the records

db.emp.findOne(<condition>,<projection>)	find first matching document
db.emp.find(<condition>,<projection>)	find all matching document in projection you can use either inclusion or exclusion, and not combination of it.-→ db.emp.find({},{blog:1,_id:0}) it will display only blog key and hide all other keys, but to hide _id, explicitly we need to add _id:0

   pretty()---it is a function which will prettify the data, and print it in proper readable format db.movie.find({},{name:1,rating:1,_id:0})----display only name and ratting for all the documents

Document in JSON format

```
{empno:123,
ename:'Revati',
sal:4567.67,
skill:['python','perl','java']
dept:{deptno:10,dname:"admin",location:"mumbai"},
projects:[{name:"p1",duration:800},{name:"p2",duration:500}],
married:true
}
```

db.movie.find({rating:2},{name:1,rating:1,price:1,_id:0})	select name,rating,price from movie where rating=2
db.movie.find({rating:2})	select * from movie where rating=2