

```

myDB> db.students.drop();
true
myDB> show dbs
admin      40.00 KiB
config     72.00 KiB
local     120.00 KiB
students   72.00 KiB
myDB> db.createCollection("Student");
{ ok: 1 }
myDB> show dbs
admin      40.00 KiB
config     72.00 KiB
local     120.00 KiB
myDB       8.00 KiB
students   72.00 KiB
myDB> db.Student.insert({_id:1,StudeName:"MichelleJacintha",Grade:"VII",Hobbies:"InternetSurfing",});
DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.
{ acknowledged: true, insertedIds: { '0': 1 } }
myDB> db.Student.insertOne({_id:1,StudeName:"MichelleJacintha",Grade:"VII",Hobbies:"InternetSurfing",});
MongoServerError: E11000 duplicate key error collection: myDB.Student index: _id_ dup key: { _id: 1 }
myDB> db.Student.insertOne({_id:2,StudeName:"MichelleJacintha",Grade:"VII",Hobbies:"InternetSurfing",});
{ acknowledged: true, insertedId: 2 }
myDB> db.Student.find();
[
  {
    _id: 1,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  },
  {
    _id: 2,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  }
]

```

```

]
myDB> show collections;
Student
myDB> db.Student.updateOne({_id:2,StudeName:"AryanDavid",Grade:"VIII"},{$set:{Hobbies:"Chess"}},{upsert:true});
MongoServerError: E11000 duplicate key error collection: myDB.Student index: _id_ dup key: { _id: 2 }
myDB> db.Student.find({StudeName:"AryanDavid"});

myDB> db.Student.find();
[
  {
    _id: 1,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  },
  {
    _id: 2,
    StudeName: 'AryanDavid',
    Grade: 'VIII',
    Hobbies: 'Skating'
  }
]
myDB> db.Student.find({StudeName:"AryanDavid"});
[
  {
    _id: 2,
    StudeName: 'AryanDavid',
    Grade: 'VIII',
    Hobbies: 'Skating'
  }
]
myDB> db.Student.updateOne({_id:2,StudeName:"AryanDavid",Grade:"VIII"},{$set:{Hobbies:"Chess"}},{upsert:true});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
myDB> db.Student.find();
[
  {
    _id: 1,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  },
  { _id: 2, StudeName: 'AryanDavid', Grade: 'VIII', Hobbies: 'Chess' }
]
myDB> db.Student.find({StudeName:"AryanDavid"});
[
  { _id: 2, StudeName: 'AryanDavid', Grade: 'VIII', Hobbies: 'Chess' }
]

```

```

myDB> db.Student.find({Grade:{Seq:'VII'}}).pretty();
[
  {
    _id: 1,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  }
]
myDB> db.Student.find({Hobbies:{$in:['Chess','Skating']}}).pretty();
[
  { _id: 2, StudeName: 'AryanDavid', Grade: 'VIII', Hobbies: 'Chess' }
]
myDB> db.Student.find({StudeName:/^M/}).pretty();
[
  {
    _id: 1,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  }
]
myDB> db.Student.find({StudeName:/e/}).pretty();
[
  {
    _id: 1,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  }
]
myDB> db.Student.find({StudeName:/d/}).pretty();
[
  { _id: 2, StudeName: 'AryanDavid', Grade: 'VIII', Hobbies: 'Chess' }
]
myDB> db.
... db.Student.count();
DeprecationWarning: Collection.count() is deprecated. Use countDocuments or estimatedDocumentCount.
0
myDB> db.Student.count();
2
myDB> db.Student.find().sort({StudName:-1}).pretty();
[
  {
    _id: 1,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  },
  { _id: 2, StudeName: 'AryanDavid', Grade: 'VIII', Hobbies: 'Chess' }
]

```

```

myDB> db.Student.find().sort({StudName:1}).pretty();
[
  {
    _id: 1,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  },
  { _id: 2, StudeName: 'AryanDavid', Grade: 'VIII', Hobbies: 'Chess' }
]

```

```

myDB> db.Student.save({StudeName:"Vamsi",Grade:"VI"});

```

```

TypeError: db.Student.save is not a function

```

```

myDB> db.Student.save({StudeName:"Vamsi",Grade:"VI"});

```

```

TypeError: db.Student.save is not a function

```

```

myDB> db.Student.updateOne({_id:2},{ $set:{Location:"Network"}});

```

```

{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

```

myDB> db.Student.find();

```

```

[
  {
    _id: 1,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing'
  },
  {
    _id: 2,
    StudeName: 'AryanDavid',
    Grade: 'VIII',
    Hobbies: 'Chess',
    Location: 'Network'
  }
]

```

```

myDB> db.Student.updateOne({_id:2},{ $unset:{Location:"Network"}});

```

```

{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

```

myDB> db.Student.find();

```



```

1
myDB> db.Student.find().sort({StudeName:1}).pretty();
[
  {
    _id: 2,
    StudeName: 'AryanDavid',
    Grade: 'VIII',
    Hobbies: 'Chess',
    Location: 'Network'
  },
  {
    _id: 1,
    StudeName: 'MichelleJacintha',
    Grade: 'VII',
    Hobbies: 'InternetSurfing',
    Location: 'Sales'
  }
]
myDB> db.Student.find().skip(1).pretty();
[
  {
    _id: 2,
    StudeName: 'AryanDavid',
    Grade: 'VIII',
    Hobbies: 'Chess',
    Location: 'Network'
  }
]
myDB> db.createCollection("Food");
{ ok: 1 }
myDB> db.food.inertOne({_id:1,fruits:['grapes','mango','apple']});
TypeError: db.food.inertOne is not a function
myDB> db.food.insertOne({_id:1,fruits:['grapes','mango','apple']});
{ acknowledged: true, insertedId: 1 }
myDB> db.food.insertOne({_id:2,fruits:['grapes','mango','cherry']});
{ acknowledged: true, insertedId: 2 }

```

```
[
myDB> db.food.find();
[
  { _id: 1, fruits: [ 'grapes', 'mango', 'apple' ] },
  { _id: 2, fruits: [ 'grapes', 'mango', 'cherry' ] },
  { _id: 3, fruits: [ 'banana', 'mango' ] }
]
myDB> db.food.find({fruits:['grapes','mango','apple']});
[ { _id: 1, fruits: [ 'grapes', 'mango', 'apple' ] } ]
myDB> db.food.find({});
[
  { _id: 1, fruits: [ 'grapes', 'mango', 'apple' ] },
  { _id: 2, fruits: [ 'grapes', 'mango', 'cherry' ] },
  { _id: 3, fruits: [ 'banana', 'mango' ] }
]
myDB> db.food.find({'fruits':{'$size:2'}});
[ { _id: 3, fruits: [ 'banana', 'mango' ] } ]
myDB>
```

```
myDB>
bmscsecse@bmscsecse-HP-Elite-Tower-800-G9-Desktop-PC: $ mongoexport --url="mongodb://localhost:27017" --db=myDB --collection=Student --type=csv --out=/home/bmscsecse/Desktop/studentnew.csv --fields="StudentName,Grade,Hobbies,Location"
2025-03-11T14:53:00.104+0530 connected to: mongodb://localhost:27017
2025-03-11T14:53:00.112+0530 exported 2 records
bmscsecse@bmscsecse-HP-Elite-Tower-800-G9-Desktop-PC: $ mongoexport --url="mongodb://localhost:27017" --db=myDB --collection=Student --type=csv --out=/home/bmscsecse/Desktop/studentnew.csv --fields="StudentName,Grade,Hobbies,Location"
2025-03-11T14:56:22.648+0530 connected to: mongodb://localhost:27017
2025-03-11T14:56:22.656+0530 exported 2 records
bmscsecse@bmscsecse-HP-Elite-Tower-800-G9-Desktop-PC: $ mongoimport --db myDB --collection Student --type csv --headerline --file /home/bmscsecse/Desktop/studentnew.csv
2025-03-11T14:57:55.803+0530 error parsing command line options: unknown option "file/home/bmscsecse/Desktop/studentnew.csv"
2025-03-11T14:57:55.803+0530 try 'mongoimport --help' for more information
bmscsecse@bmscsecse-HP-Elite-Tower-800-G9-Desktop-PC: $ mongoimport --db myDB --collection Student --type csv --headerline --file /home/bmscsecse/Desktop/studentnew.csv
2025-03-11T14:58:20.148+0530 error validating settings: must specify --fields, --fieldFile or --headerline to import this file type
bmscsecse@bmscsecse-HP-Elite-Tower-800-G9-Desktop-PC: $ mongoimport --db myDB --collection Student --type csv --headerline --file /home/bmscsecse/Desktop/studentnew.csv
2025-03-11T14:58:35.628+0530 connected to: mongodb://localhost/
2025-03-11T14:58:35.763+0530 2 document(s) imported successfully. 0 document(s) failed to import.
bmscsecse@bmscsecse-HP-Elite-Tower-800-G9-Desktop-PC: $
```