

CRUD Operations 1

```
mongos> use test
```

```
mongos> db.c1.insert({city: 'Newcastle'})
```

```
WriteResult({ "nInserted" : 1 })
```

```
mongos> db.c1.insertOne({city: 'London'})
```

```
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60ad3982c19cbcf173c75132")
}
```

```
mongos>
```

```
mongos> db.c1.insertOne({_id:100, city: 'Durham'})
```

```
{ "acknowledged" : true, "insertedId" : 100 }
```

```
mongos> db.c1.insert({_id:100, msg: 'Err by insert'})
```

```
WriteResult({
  "nInserted" : 0,
  "writeError" : {
    "code" : 11000,
    "errmsg" : "E11000 duplicate key error collection: test.c1 index: _id_ dup key: { _id:
100.0 }"
  }
})
```

```
mongos> db.c2.insert([{city: 'London'}, {nation: 'UK'}], {ordered: true})
```

```
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 2,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
```

```
mongos> db.c2.find()
```

```
{ "_id" : ObjectId("60ad3cdcc19cbcf173c75133"), "city" : "London" }
{ "_id" : ObjectId("60ad3cdcc19cbcf173c75134"), "nation" : "UK" }
```

```

mongos> db.c2.insert([{_id:10, msg:'First'},{_id:10, msg:'Second'},{_id:20, msg:'Third'}])
BulkWriteResult({
  "writeErrors" : [
    {
      "index" : 1,
      "code" : 11000,
      "errmsg" : "E11000 duplicate key error collection: test.c2 index: _id_ dup key: {
_id: 10.0 }",
      "op" : {
        "_id" : 10,
        "msg" : "Second"
      }
    }
  ],
  "writeConcernErrors" : [],
  "nInserted" : 1,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : []
})

```

```

mongos> db.c2.find()
{ "_id" : ObjectId("60ad3cdcc19cbcf173c75133"), "city" : "London" }
{ "_id" : ObjectId("60ad3cdcc19cbcf173c75134"), "nation" : "UK" }
{ "_id" : 10, "msg" : "First" }

```

```

mongo> db.c3.insert([{_id:10, msg:'First'},{_id:10, msg:'Second'},{_id:20,
msg:'Third'}],{ordered:false})
BulkWriteResult({
  "writeErrors" : [
    {
      "index" : 1,
      "code" : 11000,
      "errmsg" : "E11000 duplicate key error collection: test.c3 index: _id_ dup key: {
_id: 10.0 }",
      "op" : {
        "_id" : 10,
        "msg" : "Second"
      }
    }
  ],
  "writeConcernErrors" : [],
  "nInserted" : 2,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : []
})

```

```

mongo> db.c3.find()
{ "_id" : 10, "msg" : "First" }
{ "_id" : 20, "msg" : "Third" }

```

```

mongo> db.c4.insertMany([{city: 'New York'},{nation:'US'}])
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("60ad4073c19cbcf173c75135"),
    ObjectId("60ad4073c19cbcf173c75136")
  ]
}

```

```

mongo> db.c1.insertOne({msg:'Hi'})
{
  "acknowledged" : true,
  "insertedId" : ObjectId("60ad40d6c19cbcf173c75137")
}

```

```

mongo> db.c4.insertMany([{_id:10, msg:'First'},{_id:10, msg:'Second'}])
uncaught exception: BulkWriteError({
  "writeErrors" : [
    {
      "index" : 1,
      "code" : 11000,
      "errmsg" : "E11000 duplicate key error collection: test.c4 index: _id_ dup key: {
_id: 10.0 }",
      "op" : {
        "_id" : 10,
        "msg" : "Second"
      }
    }
  ],
  "writeConcernErrors" : [],
  "nInserted" : 1,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : []
}):
BulkWriteError({
  "writeErrors" : [
    {
      "index" : 1,
      "code" : 11000,
      "errmsg" : "E11000 duplicate key error collection: test.c4 index: _id_ dup key: {
_id: 10.0 }",
      "op" : {
        "_id" : 10,
        "msg" : "Second"
      }
    }
  ],
  "writeConcernErrors" : [],
  "nInserted" : 1,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : []
})

```

mongos>

```
db.runCommand({insert:'c5',documents:[{city:'Paris'},{nation:'France'}],ordered:false})
```

```
{
  "n" : 2,
  "ok" : 1,
  "operationTime" : Timestamp(1621969861, 2),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1621969861, 2),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
```

```
mongos> db.runCommand({insert:'c5',documents:[{id:1, msg:'first'}, {_id:1, mdg:'second'},
{msg:'third'}],ordered:false})
```

```
{
  "n" : 3,
  "ok" : 1,
  "operationTime" : Timestamp(1621969985, 3),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1621969985, 3),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
```

mongos>

```
db.runCommand({insert:'c5',documents:[{city:'Paris'},{nation:'France'}],ordered:false})
```

```
{
  "n" : 2,
  "ok" : 1,
  "operationTime" : Timestamp(1621969861, 2),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1621969861, 2),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
```

```

mongos> db.runCommand({insert:'c6',documents:[{_id:1, msg:'first'}, {_id:1, msg:'second'},
{msg:'third'}],ordered:false})
{
  "n" : 1,
  "writeErrors" : [
    {
      "index" : 0,
      "code" : 11000,
      "codeName" : "DuplicateKey",
      "errmsg" : "E11000 duplicate key error collection: test.c6 index: _id_ dup key: {
_id: 1.0 }"
    },
    {
      "index" : 1,
      "code" : 11000,
      "codeName" : "DuplicateKey",
      "errmsg" : "E11000 duplicate key error collection: test.c6 index: _id_ dup key: {
_id: 1.0 }"
    }
  ],
  "ok" : 1,
  "operationTime" : Timestamp(1621970209, 6),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1621970209, 6),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}

```

```

mongos> db.c6.find()
{ "_id" : ObjectId("60ad4c9e9de8e2e63893631c"), "id" : 1, "msg" : "first" }
{ "_id" : ObjectId("60ad4c9e9de8e2e63893631d"), "msg" : "third" }

```

```

mongos> for(var i=1;i<=100;i++){db.c7.insert({msg:'cursor',num:i})}
WriteResult({ "nInserted" : 1 })

```

```

mongos> db.c7.find()

```

```

mongos> db.c7.findOne()
{
  "_id" : ObjectId("60ad4e0bc19cbcf173c7513a"),
  "msg" : "cursor",

```

```
    "num" : 1
  }
```

```
mongos> load('cars.js')
```

```
true
```

```
mongos> cars
```

```
[
  {
    "brand" : "Toyota",
    "model" : "Supra",
    "engine" : "Petrol",
    "gearbox" : "Manual"
  },
  {
    "brand" : "Toyota",
    "model" : "Camry",
    "engine" : "Petrol",
    "gearbox" : "Auto",
    "mileage" : 120000,
    "year" : 2010
  },
  {
    "brand" : "Honda",
    "model" : "CR-V",
    "engine" : "Hybrid",
    "gearbox" : "CVT"
  },
  {
    "brand" : "Honda",
    "model" : "Civic",
    "engine" : "Petrol",
    "gearbox" : "Auto",
    "mileage" : NumberInt(5000),
    "year" : 2019
  },
  {
    "brand" : "Toyota",
    "model" : "Corolla",
    "engine" : "Petrol",
    "gearbox" : "Manual",
    "mileage" : NumberLong(200000),
    "year" : 2000
  }
]
```

```
    },
    {
        "brand" : "Honda",
        "model" : "Accord",
        "engine" : "Diesel",
        "gearbox" : "CVT",
        "mileage" : NumberLong(65000),
        "year" : 2010
    },
    {
        "brand" : "Lexus",
        "model" : "RX350H",
        "engine" : "Hybrid",
        "gearbox" : "CVT",
        "mileage" : 7000,
        "year" : 2018
    },
    {
        "brand" : "Nissan",
        "model" : "Leaf",
        "engine" : "Electric",
        "gearbox" : "CVT",
        "mileage" : 15000,
        "year" : 2010
    },
    {
        "brand" : "Ford",
        "model" : "F-150",
        "engine" : "Petrol",
        "gearbox" : "Auto",
        "mileage" : 27000,
        "year" : 2015
    },
    {
        "brand" : "Honda",
        "model" : "Civic",
        "engine" : "Diesel",
        "gearbox" : "Manual",
        "mileage" : 35000,
        "year" : 2016
    }
]
```



```
mongos> use cars
switched to db cars
```

```
mongos> db.list.insert(cars)
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 10,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
```

```
mongos> db.list.find()
{ "_id" : ObjectId("60ad5cfec19cbcf173c7519e"), "brand" : "Toyota", "model" : "Supra",
  "engine" : "Petrol", "gearbox" : "Manual" }
{ "_id" : ObjectId("60ad5cfec19cbcf173c7519f"), "brand" : "Toyota", "model" : "Camry",
  "engine" : "Petrol", "gearbox" : "Auto", "mileage" : 120000, "year" : 2010 }
{ "_id" : ObjectId("60ad5cfec19cbcf173c751a0"), "brand" : "Honda", "model" : "CR-V", "engine"
  : "Hybrid", "gearbox" : "CVT" }
{ "_id" : ObjectId("60ad5cfec19cbcf173c751a1"), "brand" : "Honda", "model" : "Civic", "engine"
  : "Petrol", "gearbox" : "Auto", "mileage" : 5000, "year" : 2019 }
{ "_id" : ObjectId("60ad5cfec19cbcf173c751a2"), "brand" : "Toyota", "model" : "Corolla",
  "engine" : "Petrol", "gearbox" : "Manual", "mileage" : NumberLong(200000), "year" : 2000 }
{ "_id" : ObjectId("60ad5cfec19cbcf173c751a3"), "brand" : "Honda", "model" : "Accord",
  "engine" : "Diesel", "gearbox" : "CVT", "mileage" : NumberLong(65000), "year" : 2010 }
{ "_id" : ObjectId("60ad5cfec19cbcf173c751a4"), "brand" : "Lexus", "model" : "RX350H",
  "engine" : "Hybrid", "gearbox" : "CVT", "mileage" : 7000, "year" : 2018 }
{ "_id" : ObjectId("60ad5cfec19cbcf173c751a5"), "brand" : "Nissan", "model" : "Leaf", "engine" :
  "Electric", "gearbox" : "CVT", "mileage" : 15000, "year" : 2010 }
{ "_id" : ObjectId("60ad5cfec19cbcf173c751a6"), "brand" : "Ford", "model" : "F-150", "engine" :
  "Petrol", "gearbox" : "Auto", "mileage" : 27000, "year" : 2015 }
{ "_id" : ObjectId("60ad5cfec19cbcf173c751a7"), "brand" : "Honda", "model" : "Civic", "engine"
  : "Diesel", "gearbox" : "Manual", "mileage" : 35000, "year" : 2016 }
```

```
mongos> db.list.find({}, {_id:0})
{"brand": "Toyota", "model": "Supra", "engine": "Petrol", "gearbox": "Manual" }
{"brand": "Toyota", "model": "Camry", "engine": "Petrol", "gearbox": "Auto", "mileage" :
120000, "year" : 2010 }
{"brand": "Honda", "model": "CR-V", "engine": "Hybrid", "gearbox": "CVT" }
{"brand": "Honda", "model": "Civic", "engine": "Petrol", "gearbox": "Auto", "mileage" : 5000,
"year" : 2019 }
{"brand": "Toyota", "model": "Corolla", "engine": "Petrol", "gearbox": "Manual", "mileage" :
NumberLong(200000), "year" : 2000 }
{"brand": "Honda", "model": "Accord", "engine": "Diesel", "gearbox": "CVT", "mileage" :
NumberLong(65000), "year" : 2010 }
{"brand": "Lexus", "model": "RX350H", "engine": "Hybrid", "gearbox": "CVT", "mileage" :
7000, "year" : 2018 }
{"brand": "Nissan", "model": "Leaf", "engine": "Electric", "gearbox": "CVT", "mileage" :
15000, "year" : 2010 }
{"brand": "Ford", "model": "F-150", "engine": "Petrol", "gearbox": "Auto", "mileage" : 27000,
"year" : 2015 }
{"brand": "Honda", "model": "Civic", "engine": "Diesel", "gearbox": "Manual", "mileage" :
35000, "year" : 2016 }
```

```
mongos> db.list.find({brand:{$eq: 'Honda'}}, {_id:false})
{"brand": "Honda", "model": "CR-V", "engine": "Hybrid", "gearbox": "CVT" }
{"brand": "Honda", "model": "Civic", "engine": "Petrol", "gearbox": "Auto", "mileage" : 5000,
"year" : 2019 }
{"brand": "Honda", "model": "Accord", "engine": "Diesel", "gearbox": "CVT", "mileage" :
NumberLong(65000), "year" : 2010 }
{"brand": "Honda", "model": "Civic", "engine": "Diesel", "gearbox": "Manual", "mileage" :
35000, "year" : 2016 }
```

```
mongos> db.list.find({gearbox:{$in: ['Auto','CVT']}}, {_id:0})
{"brand": "Toyota", "model": "Camry", "engine": "Petrol", "gearbox": "Auto", "mileage" :
120000, "year" : 2010 }
{"brand": "Honda", "model": "CR-V", "engine": "Hybrid", "gearbox": "CVT" }
{"brand": "Honda", "model": "Civic", "engine": "Petrol", "gearbox": "Auto", "mileage" : 5000,
"year" : 2019 }
{"brand": "Honda", "model": "Accord", "engine": "Diesel", "gearbox": "CVT", "mileage" :
NumberLong(65000), "year" : 2010 }
{"brand": "Lexus", "model": "RX350H", "engine": "Hybrid", "gearbox": "CVT", "mileage" :
7000, "year" : 2018 }
{"brand": "Nissan", "model": "Leaf", "engine": "Electric", "gearbox": "CVT", "mileage" :
15000, "year" : 2010 }
```

```
{ "brand" : "Ford", "model" : "F-150", "engine" : "Petrol", "gearbox" : "Auto", "mileage" : 27000, "year" : 2015 }
```

```
mongos> db.list.find({$and: [{year: {$gte:2005}}, {year: {$lte:2015}}]}, {_id:0})
{ "brand" : "Toyota", "model" : "Camry", "engine" : "Petrol", "gearbox" : "Auto", "mileage" : 120000, "year" : 2010 }
{ "brand" : "Honda", "model" : "Accord", "engine" : "Diesel", "gearbox" : "CVT", "mileage" : NumberLong(65000), "year" : 2010 }
{ "brand" : "Nissan", "model" : "Leaf", "engine" : "Electric", "gearbox" : "CVT", "mileage" : 15000, "year" : 2010 }
{ "brand" : "Ford", "model" : "F-150", "engine" : "Petrol", "gearbox" : "Auto", "mileage" : 27000, "year" : 2015 }
```