|  |
| --- |
| db |
| Show dbs |
| db.dropDatabase() |
| Use NEWDB |
| db.createCollection("newcol") |
| show collections |
| db.newcol.insert({"name" : "John"}) |
| db.newcol.drop() |
| db.newcol.insert(  [  {  Name: "John",  Department: "L&D",  by: "IIML",  url: "mongodb.com",  tags: ["IT", "HR", "Academic"],  Salary: 1,00,000  }  ]  ) |
| db.newcol.insertMany(  [  {  First\_Name: "Rahul",  Last\_Name: "Srivastava",  Date\_Of\_Birth: "1995-09-26",  e\_mail: "rahul@company.com",  phone: "9xxxxxxxxxx"  },  {  First\_Name: "Jacob",  Last\_Name: "John",  Date\_Of\_Birth: "1990-02-16",  e\_mail: "jacob@company.com",  phone: "9xxxxxxxxx"  },  {  First\_Name: "Fathima",  Last\_Name: "Sheik",  Date\_Of\_Birth: "1990-02-16",  e\_mail: " fathima@company.com ",  phone: "9xxxxxxxx"  }  ]  ) |
| db.newcol.insertMany(  [  {  First\_Name: "Rahul",  Last\_Name: "Srivastava",  Date\_Of\_Birth: "1995-09-26",  e\_mail: "rahul@company.com",  phone: "9xxxxxxxxxx",  Salary:60000  },  {  First\_Name: "Jacob",  Last\_Name: "John",  Date\_Of\_Birth: "1990-02-16",  e\_mail: "jacob@company.com",  phone: "9xxxxxxxxx",  Salary:50000  },  {  First\_Name: "Jacob",  Last\_Name: "Joseph",  Date\_Of\_Birth: "1990-02-15",  e\_mail: "jacobac@company.com",  phone: "9xxxxxxxxx",  Salary:40000  },  {  First\_Name: "Fathima",  Last\_Name: "Sheik",  Date\_Of\_Birth: "1990-02-16",  e\_mail: " fathima@company.com ",  phone: "9xxxxxxxx",  Salary:30000  }  ]  ) |
| db.newcol.find() |
| db.newcol.find({ }).pretty() |
| db.newcol.findOne({First\_Name: "Jacob"}) |
| db.newcol.findOne({phone: "9xxxxxxxx"}) |
| db.newcol.find({First\_Name: "Fathima"}).pretty() |
| db.newcol.find({Salary:{ $lt:55000} }).pretty() |
| db.newcol.find({Salary:{ $lt:40000} }).pretty() |
| db.newcol.find({Salary:{ $lt:50000} }).pretty() |
| db.newcol.find({Salary:{ $lte:50000} }).pretty() |
| db.newcol.find({Salary:{ $gte:50000} }).pretty() |
| db.newcol.find({Salary:{ $gt:50000} }).pretty() |
| db.newcol.find({Salary:{ $in:[50000,60000]} }).pretty() |
| db.newcol.find({Salary:{ $in:[50000,80000]} }).pretty() |
| db.newcol.find({Salary:{ $nin:[50000,80000]} }).pretty() |
| db.newcol.find({$or:[{ First\_Name: "Jacob"},{Salary:30000}]}).pretty() |
| db.newcol.find({$and:[{},{}]}).pretty() |
| db.newcol.find({$and:[{ First\_Name: "Jacob"},{Salary:50000}]}).pretty() |
| db.newcol.find({$and:[{ First\_Name: "Jacob"},{Salary:30000}]}).pretty() |
| db.newcol.find({ Date\_Of\_Birth: "1990-02-15", $and:[{ First\_Name: "Jacob"},{ e\_mail: "jacobac@company.com"}]}).pretty() |
| db.newcol.find({phone: "9xxxxxxxxx", $and:[{ First\_Name: "Jacob"},{ Salary:{ $lt:50000} }]}).pretty() |
| db.newcol.find({phone: "9xxxxxxxxx", $and:[{ First\_Name: "Jacob"},{ Salary:{ $lte:50000} }]}).pretty() |
| db.newcol.update({Date\_Of\_Birth: "1990-02-15"},{ $set:  { Salary: 1000000 }}) |
| db.newcol.remove({Date\_Of\_Birth: "1990-02-15"}) |
| db.newcol.remove({}) |
| db.newcol.find().limit(2).skip(1) |
| db.newcol.find({Salary:{ $gt:50000} }).limit(2) |
| db.newcol.find({Salary:{ $gt:50000} }).limit(2).skip(1) |
| db.newcol.find().sort({First\_Name:1}) |
| db.newcol.aggregate(  [  {  $group:  {  "\_id":" First\_Name ", AG:{$avg: "$Salary" }  }  }  ]  ) |
| db.newcol.aggregate(  [  {  $group:  {  "\_id":" First\_Name ", AG:{$sum: "$Salary" }  }  }  ]  ) |
| db.teams.insertOne({team: "Mavs", points: 30, rebounds: 8})  db.teams.insertOne({team: "Mavs", points: 30, rebounds: 12})  db.teams.insertOne({team: "Spurs", points: 20, rebounds: 7})  db.teams.insertOne({team: "Spurs", points: 25, rebounds: 5})  db.teams.insertOne({team: "Spurs", points: 25, rebounds: 9})  db.teams.aggregate([{$group: {\_id:null, avg\_val:{$avg:"$points"}}}])  db.teams.aggregate([{$group: {\_id:"$team", avg\_val:{$avg:"$points"}}}]) |
| db.teams.insertOne({team: "Mavs", position: "Guard", points: 31})  db.teams.insertOne({team: "Spurs", position: "Guard", points: 22})  db.teams.insertOne({team: "Rockets", position: "Center", points: 19})  db.teams.insertOne({team: "Warriors", position: "Forward", points: 26})  db.teams.insertOne({team: "Cavs", position: "Guard", points: 33})  db.teams.aggregate([  {$group : {\_id:"$position", count:{$sum:1}}}  ]) |
| db.teams.aggregate([  {$group : {\_id:"$position", count:{$sum:1}}},  {$sort: {count:-1}}  ]) |
| db.teams.insertOne({team: "Mavs", position: "Guard", points: 31})  db.teams.insertOne({team: "Mavs", position: "Guard", points: 22})  db.teams.insertOne({team: "Rockets", position: "Center", points: 19})  db.teams.insertOne({team: "Rockets", position: "Forward", points: 26})  db.teams.insertOne({team: "Cavs", position: "Guard", points: 33})  db.teams.distinct('team') |
| db.newsales.insertMany(  [  { "\_id" : 1, "item" : "abc1", description: "product 1", qty: 300 },  { "\_id" : 2, "item" : "abc2", description: "product 2", qty: 200 },  { "\_id" : 3, "item" : "xyz1", description: "product 3", qty: 250 },  { "\_id" : 4, "item" : "VWZ1", description: "product 4", qty: 300 },  { "\_id" : 5, "item" : "VWZ2", description: "product 5", qty: 180 }  ]  )  db.newsales.aggregate(  [  {  $project:  {  item: 1,  qty: 1,  cmpTo250: { $cmp: [ "$qty", 250 ] },  \_id: 0  }  }  ]  ) |
| db.products.insertMany([  { "\_id" : 1, "item" : "apple", "type" : "honey crisp", "price" : NumberDecimal("1.99") },  { "\_id" : 2, "item" : "apple", "type" : "fuji", "price" : NumberDecimal("1.99") },  { "\_id" : 3, "item" : "apple", "type" : "jonagold", "price" : NumberDecimal("1.29") },  { "\_id" : 4, "item" : "apple", "type" : "jonathan", "price" : NumberDecimal("1.29") },  { "\_id" : 5, "item" : "apple", "type" : "mcintosh", "price" : NumberDecimal("1.29") },  { "\_id" : 6, "item" : "apple", "type" : "cortland", "price" : NumberDecimal("1.29") },  { "\_id" : 7, "item" : "orange", "type" : "cara cara", "price" : NumberDecimal("2.99") },  { "\_id" : 9, "item" : "orange", "type" : "satsuma", "price" : NumberDecimal("1.99") },  { "\_id" : 8, "item" : "orange", "type" : "valencia", "price" : NumberDecimal("0.99") },  { "\_id" : 10, "item" : "orange", "type" : "navel", "price" : NumberDecimal("1.39") }  ])  db.products.find().min( { price: NumberDecimal("1.39") } ).max( { price: NumberDecimal("1.99") } ).hint( { price: 1 } ) |