

Assignment No. 1

emp> show dbs

Internship 336.00 KiB

admin 40.00 KiB

config 60.00 KiB

inotebook 8.00 KiB

local 88.00 KiB

mydb 8.00 KiB

emp> use mydb

switched to db mydb

mydb> show dbs

Internship 336.00 KiB

admin 40.00 KiB

config 60.00 KiB

inotebook 8.00 KiB

local 88.00 KiB

mydb 8.00 KiB

mydb> db.movie.insert({"name":"tutorials point"})

DeprecationWarning: Collection.insert() is deprecated. Use insertOne, insertMany, or bulkWrite.

```
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("65417ab3f1a2c5c7b3297941") }
}
```

mydb> show dbs

Internship 336.00 KiB

admin 40.00 KiB

config 60.00 KiB

inotebook 8.00 KiB

local 88.00 KiB

mydb 48.00 KiB

```
mydb> db.dropDatabase()
```

```
{ ok: 1, dropped: 'mydb' }
```

```
mydb> show dbs
```

```
Internship 336.00 KiB
```

```
admin      40.00 KiB
```

```
config     72.00 KiB
```

```
inotebook  8.00 KiB
```

```
local      88.00 KiB
```

```
mydb> use test
```

```
switched to db test
```

```
test> db.createCollection("mycollection")
```

```
{ ok: 1 }
```

```
test> show collections
```

```
mycollection
```

```
test> db.mycol.insert({title:"Mongodb Overview",desc:"Is no sql database",by:"Atre Pramod",url:"https://pramod-atre.web.app",tags:['mongodb','database','Nosql']});
```

```
{  
  acknowledged: true,  
  insertedIds: { '0': ObjectId("65417d0ff1a2c5c7b3297942") }  
}
```

```
test> show dbs
```

```
Internship 336.00 KiB
```

```
admin      40.00 KiB
```

```
config     96.00 KiB
```

```
inotebook  8.00 KiB
```

```
local      88.00 KiB
```

```
test       48.00 KiB
```

```
test> use emp
```

```
switched to db emp
```

```
emp> db.createCollection("emp_info")
```

```
{ ok: 1 }
```

```
emp> show dbs
```

Internship 336.00 KiB

admin 40.00 KiB

config 96.00 KiB

emp 8.00 KiB

inotebook 8.00 KiB

local 88.00 KiB

test 48.00 KiB

emp> show collections

emp_info

emp>

db.emp_info.insertOne({emp_id:1,emp_name:"pramod",address:"Sangamner",salary:50000});

{

acknowledged: true,

insertedId: ObjectId("65417f32f1a2c5c7b3297943")

}

emp> db.emp_info.find();

[

{

_id: ObjectId("65417f32f1a2c5c7b3297943"),

emp_id: 1,

emp_name: 'pramod',

address: 'Sangamner',

salary: 50000

}

]

emp> db.emp_info.find().pretty();

[

{

_id: ObjectId("65417f32f1a2c5c7b3297943"),

emp_id: 1,

emp_name: 'pramod',

address: 'Sangamner',

```

    salary: 50000
  }
]
emp> db.emp_info.insertOne({emp_id:2,emp_name:"Atre",address:"Nasik",salary:50000});
{
  acknowledged: true,
  insertedId: ObjectId("65417fbdf1a2c5c7b3297944")
}

```

emp> db.emp_info.find().pretty();

```

[
  {
    _id: ObjectId("65417f32f1a2c5c7b3297943"),
    emp_id: 1,
    emp_name: 'pramod',
    address: 'Sangamner',
    salary: 50000
  },
  {
    _id: ObjectId("65417fbdf1a2c5c7b3297944"),
    emp_id: 2,
    emp_name: 'Atre',
    address: 'Nasik',
    salary: 50000
  }
]

```

emp> db.emp_info.remove({emp_id:1});

DeprecationWarning: Collection.remove() is deprecated. Use deleteOne, deleteMany, findOneAndDelete, or bulkWrite.

```
{ acknowledged: true, deletedCount: 1 }
```

emp> db.emp_info.find().pretty();

```

[
  {

```

```

    _id: ObjectId("65417fbdf1a2c5c7b3297944"),
    emp_id: 2,
    emp_name: 'Atre',
    address: 'Nasik',
    salary: 50000
  }
]

emp>
db.emp_info.insertOne({emp_id:1,emp_name:"pramod",address:"Sangamner",salary:
50000});
{
  acknowledged: true,
  insertedId: ObjectId("65418124f1a2c5c7b3297945")
}

emp> db.emp_info.find().pretty();
[
  {
    _id: ObjectId("65417fbdf1a2c5c7b3297944"),
    emp_id: 2,
    emp_name: 'Atre',
    address: 'Nasik',
    salary: 50000
  },
  {
    _id: ObjectId("65418124f1a2c5c7b3297945"),
    emp_id: 1,
    emp_name: 'pramod',
    address: 'Sangamner',
    salary: 50000
  }
]

emp> db.emp_info.updateOne({emp_id:1},{ $set: {emp_name:"Pramod Atre"}});
{

```

```
    acknowledged: true,
    insertedId: null,
    matchedCount: 1,
    modifiedCount: 1,
    upsertedCount: 0
  }
emp> db.emp_info.find().pretty();
[
  {
    _id: ObjectId("65417fbdf1a2c5c7b3297944"),
    emp_id: 2,
    emp_name: 'Atre',
    address: 'Nasik',
    salary: 50000
  },
  {
    _id: ObjectId("65418124f1a2c5c7b3297945"),
    emp_id: 1,
    emp_name: 'Pramod Atre',
    address: 'Sangamner',
    salary: 50000
  }
]
```

Assignment No. 2

emp> show dbs

```
Internship 336.00 KiB
admin      40.00 KiB
config     96.00 KiB
emp        72.00 KiB
inotebook  8.00 KiB
local      88.00 KiB
test       48.00 KiB
```

emp> use bank

switched to db bank

bank> db.createCollection("account")

{ ok: 1 }

bank> db.createCollection("branch")

{ ok: 1 }

bank> db.createCollection("customer")

{ ok: 1 }

bank> show dbs

```
Internship 336.00 KiB
admin      40.00 KiB
bank       24.00 KiB
config     108.00 KiB
emp        72.00 KiB
inotebook  8.00 KiB
local      88.00 KiB
test       48.00 KiB
```

bank> show collections

account

branch

customer

```
bank>
db.account.insertMany([
  {ac_no:1111001,cus_id:"cus001",branch_id:"br001",balance:30000},
  {ac_no:1111002,cus_id:"cus002",branch_id:"br001",balance:10000},
  {ac_no:1111003,cus_id:"cus003",branch_id:"br002",balance:3000},
  {ac_no:1111004,cus_id:"cus004",branch_id:"br002",balance:5000},
  {ac_no:1111005,cus_id:"cus005",branch_id:"br003",balance:3000},
  {ac_no:1111006,cus_id:"cus006",branch_id:"br003",balance:5000},
  {ac_no:1111007,cus_id:"cus007",branch_id:"br004",balance:400},
  {ac_no:1111008,cus_id:"cus008",branch_id:"br004",balance:2000}
]);
```

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("6541873df1a2c5c7b3297946"),
    '1': ObjectId("6541873df1a2c5c7b3297947"),
    '2': ObjectId("6541873df1a2c5c7b3297948"),
    '3': ObjectId("6541873df1a2c5c7b3297949"),
    '4': ObjectId("6541873df1a2c5c7b329794a"),
    '5': ObjectId("6541873df1a2c5c7b329794b"),
    '6': ObjectId("6541873df1a2c5c7b329794c"),
    '7': ObjectId("6541873df1a2c5c7b329794d")
  }
}
```

```
bank> db.account.find();
```

```
[
  {
    _id: ObjectId("6541873df1a2c5c7b3297946"),
    ac_no: 1111001,
    cus_id: 'cus001',
    branch_id: 'br001',
    balance: 30000
  },
  {
    _id: ObjectId("6541873df1a2c5c7b3297947"),
    ac_no: 1111002,
    cus_id: 'cus002',
```



```
branch_id: 'br001',
balance: 10000
},
{
  _id: ObjectId("6541873df1a2c5c7b3297948"),
  ac_no: 1111003,
  cus_id: 'cus003',
  branch_id: 'br002',
  balance: 3000
},
{
  _id: ObjectId("6541873df1a2c5c7b3297949"),
  ac_no: 1111004,
  cus_id: 'cus004',
  branch_id: 'br002',
  balance: 5000
},
{
  _id: ObjectId("6541873df1a2c5c7b329794a"),
  ac_no: 1111005,
  cus_id: 'cus005',
  branch_id: 'br003',
  balance: 3000
},
{
  _id: ObjectId("6541873df1a2c5c7b329794b"),
  ac_no: 1111006,
  cus_id: 'cus006',
  branch_id: 'br003',
  balance: 5000
},
{
```

```

    _id: ObjectId("6541873df1a2c5c7b329794c"),
    ac_no: 1111007,
    cus_id: 'cus007',
    branch_id: 'br004',
    balance: 400
  },
  {
    _id: ObjectId("6541873df1a2c5c7b329794d"),
    ac_no: 1111008,
    cus_id: 'cus008',
    branch_id: 'br004',
    balance: 2000
  }
]

bank>
db.branch.insertMany([{id:"br001",name:"UBION",city:"Sangamner",assets:20000},{
id:"br002",name:"UBION",city:"mumbai",assets:20000},{id:"br003",name:"UBION"
,city:"pune",assets:2000},{id:"br004",name:"UBION",city:"Nasik",assets:10000}]);
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("654188ebf1a2c5c7b329794e"),
    '1': ObjectId("654188ebf1a2c5c7b329794f"),
    '2': ObjectId("654188ebf1a2c5c7b3297950"),
    '3': ObjectId("654188ebf1a2c5c7b3297951")
  }
}

bank> db.branch.find();
[
  {
    _id: ObjectId("654188ebf1a2c5c7b329794e"),
    id: 'br001',
    name: 'UBION',

```

```

    city: 'Sangamner',
    assets: 20000
  },
  {
    _id: ObjectId("654188ebf1a2c5c7b329794f"),
    id: 'br002',
    name: 'UBION',
    city: 'mumbai',
    assets: 20000
  },
  {
    _id: ObjectId("654188ebf1a2c5c7b3297950"),
    id: 'br003',
    name: 'UBION',
    city: 'pune',
    assets: 2000
  },
  {
    _id: ObjectId("654188ebf1a2c5c7b3297951"),
    id: 'br004',
    name: 'UBION',
    city: 'Nasik',
    assets: 10000
  }
]

```

bank>

```

db.customer.insertMany([
  {id:"cus001",name:"Pramod",street:"sangamner",city:"Sangamner"},
  {id:"cus002",name:"ABC",street:"sangamner",city:"Sangamner"},
  {id:"cus003",name:"PQR",street:"sangamner",city:"mumbai"},
  {id:"cus004",name:"Atre",street:"sangamner",city:"mumbai"},
  {id:"cus005",name:"vwx",street:"sangamner",city:"pune"},
  {id:"cus006",name:"LMN",street:"sangamner",city:"pune"},
  {id:"cus007",name:"STU",street:"sangamner",city:"Nasik"},
  {id:"cus008",name:"XYZ",street:"sangamner",city:"Nasik"}
]);

```

```

{

```

```
acknowledged: true,
insertedIds: {
  '0': ObjectId("65418bdf1a2c5c7b3297952"),
  '1': ObjectId("65418bdf1a2c5c7b3297953"),
  '2': ObjectId("65418bdf1a2c5c7b3297954"),
  '3': ObjectId("65418bdf1a2c5c7b3297955"),
  '4': ObjectId("65418bdf1a2c5c7b3297956"),
  '5': ObjectId("65418bdf1a2c5c7b3297957"),
  '6': ObjectId("65418bdf1a2c5c7b3297958"),
  '7': ObjectId("65418bdf1a2c5c7b3297959")
}
}
```

```
bank> db.customer.find();
```

```
[
  {
    _id: ObjectId("65418bdf1a2c5c7b3297952"),
    id: 'cus001',
    name: 'Pramod',
    street: 'sangamner',
    city: 'Sangamner'
  },
  {
    _id: ObjectId("65418bdf1a2c5c7b3297953"),
    id: 'cus002',
    name: 'ABC',
    street: 'sangamner',
    city: 'Sangamner'
  },
  {
    _id: ObjectId("65418bdf1a2c5c7b3297954"),
    id: 'cus003',
    name: 'PQR',
```

```
    street: 'sangamner',
    city: 'mumbai'
  },
  {
    _id: ObjectId("65418bdf1a2c5c7b3297955"),
    id: 'cus004',
    name: 'Atre',
    street: 'sangamner',
    city: 'mumbai'
  },
  {
    _id: ObjectId("65418bdf1a2c5c7b3297956"),
    id: 'cus005',
    name: 'vwx',
    street: 'sangamner',
    city: 'pune'
  },
  {
    _id: ObjectId("65418bdf1a2c5c7b3297957"),
    id: 'cus006',
    name: 'LMN',
    street: 'sangamner',
    city: 'pune'
  },
  {
    _id: ObjectId("65418bdf1a2c5c7b3297958"),
    id: 'cus007',
    name: 'STU',
    street: 'sangamner',
    city: 'Nasik'
  },
  {
```

```
  _id: ObjectId("65418bdf1a2c5c7b3297959"),
  id: 'cus008',
  name: 'XYZ',
  street: 'sangamner',
  city: 'Nasik'
}
]
```

```
bank> db.customer.find({city:"Nasik"});
```

```
[
  {
    _id: ObjectId("65418bdf1a2c5c7b3297958"),
    id: 'cus007',
    name: 'STU',
    street: 'sangamner',
    city: 'Nasik'
  },
  {
    _id: ObjectId("65418bdf1a2c5c7b3297959"),
    id: 'cus008',
    name: 'XYZ',
    street: 'sangamner',
    city: 'Nasik'
  }
]
```

```
bank> db.branch.find({$or:[{city:"pune"},{asset:{Sgt:3000}}]});
```

```
[
  {
    _id: ObjectId("654188ebf1a2c5c7b3297950"),
    id: 'br003',
    name: 'UBION',
    city: 'pune',
    assets: 2000
  }
]
```

```

    }
  ]
bank> db.branch.find({assets:{$gt:3000}},{city:"Nasik"});
[
  { _id: ObjectId("654188ebf1a2c5c7b329794e"), city: 'Nasik' },
  { _id: ObjectId("654188ebf1a2c5c7b329794f"), city: 'Nasik' },
  { _id: ObjectId("654188ebf1a2c5c7b3297951"), city: 'Nasik' }
]

```

```

bank> db.account.find().sort({balance:-1});
[
  {
    _id: ObjectId("6541873df1a2c5c7b3297946"),
    ac_no: 1111001,
    cus_id: 'cus001',
    branch_id: 'br001',
    balance: 30000
  },
  {
    _id: ObjectId("6541873df1a2c5c7b3297947"),
    ac_no: 1111002,
    cus_id: 'cus002',
    branch_id: 'br001',
    balance: 10000
  },
  {
    _id: ObjectId("6541873df1a2c5c7b3297949"),
    ac_no: 1111004,
    cus_id: 'cus004',
    branch_id: 'br002',
    balance: 5000
  },
  {

```

```
_id: ObjectId("6541873df1a2c5c7b329794b"),
ac_no: 1111006,
cus_id: 'cus006',
branch_id: 'br003',
balance: 5000
},
{
_id: ObjectId("6541873df1a2c5c7b3297948"),
ac_no: 1111003,
cus_id: 'cus003',
branch_id: 'br002',
balance: 3000
},
{
_id: ObjectId("6541873df1a2c5c7b329794a"),
ac_no: 1111005,
cus_id: 'cus005',
branch_id: 'br003',
balance: 3000
},
{
_id: ObjectId("6541873df1a2c5c7b329794d"),
ac_no: 1111008,
cus_id: 'cus008',
branch_id: 'br004',
balance: 2000
},
{
_id: ObjectId("6541873df1a2c5c7b329794c"),
ac_no: 1111007,
cus_id: 'cus007',
branch_id: 'br004',
```



```
    balance: 400
  }
]
```

```
bank> db.account.update({balance:{$lt:1000}},{$inc:{balance:3000}});
```

DeprecationWarning: Collection.update() is deprecated. Use updateOne, updateMany, or bulkWrite.

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

```
bank> db.account.find().sort({balance:-1});
```

```
[
  {
    _id: ObjectId("6541873df1a2c5c7b3297946"),
    ac_no: 1111001,
    cus_id: 'cus001',
    branch_id: 'br001',
    balance: 30000
  },
  {
    _id: ObjectId("6541873df1a2c5c7b3297947"),
    ac_no: 1111002,
    cus_id: 'cus002',
    branch_id: 'br001',
    balance: 10000
  },
  {
    _id: ObjectId("6541873df1a2c5c7b3297949"),
    ac_no: 1111004,
```

```
cus_id: 'cus004',
branch_id: 'br002',
balance: 5000
},
{
  _id: ObjectId("6541873df1a2c5c7b329794b"),
  ac_no: 1111006,
  cus_id: 'cus006',
  branch_id: 'br003',
  balance: 5000
},
{
  _id: ObjectId("6541873df1a2c5c7b329794c"),
  ac_no: 1111007,
  cus_id: 'cus007',
  branch_id: 'br004',
  balance: 3400
},
{
  _id: ObjectId("6541873df1a2c5c7b3297948"),
  ac_no: 1111003,
  cus_id: 'cus003',
  branch_id: 'br002',
  balance: 3000
},
{
  _id: ObjectId("6541873df1a2c5c7b329794a"),
  ac_no: 1111005,
  cus_id: 'cus005',
  branch_id: 'br003',
  balance: 3000
},
```

```

{
  _id: ObjectId("6541873df1a2c5c7b329794d"),
  ac_no: 1111008,
  cus_id: 'cus008',
  branch_id: 'br004',
  balance: 2000
}
]
bank> db.account.find({assets:{$gt:3000,$lt:5000}});
bank> db.account.find().sort({balance:-1});
[
  {
    _id: ObjectId("6541873df1a2c5c7b3297946"),
    ac_no: 1111001,
    cus_id: 'cus001',
    branch_id: 'br001',
    balance: 30000
  },
  {
    _id: ObjectId("6541873df1a2c5c7b3297947"),
    ac_no: 1111002,
    cus_id: 'cus002',
    branch_id: 'br001',
    balance: 10000
  },
  {
    _id: ObjectId("6541873df1a2c5c7b3297949"),
    ac_no: 1111004,
    cus_id: 'cus004',
    branch_id: 'br002',
    balance: 5000
  },

```

```

{
  _id: ObjectId("6541873df1a2c5c7b329794b"),
  ac_no: 1111006,
  cus_id: 'cus006',
  branch_id: 'br003',
  balance: 5000
},
{
  _id: ObjectId("6541873df1a2c5c7b3297948"),
  ac_no: 1111003,
  cus_id: 'cus003',
  branch_id: 'br002',
  balance: 3000
},
{
  _id: ObjectId("6541873df1a2c5c7b329794a"),
  ac_no: 1111005,
  cus_id: 'cus005',
  branch_id: 'br003',
  balance: 3000
},
{
  _id: ObjectId("6541873df1a2c5c7b329794d"),
  ac_no: 1111008,
  cus_id: 'cus008',
  branch_id: 'br004',
  balance: 2000
}
]
bank> db.customer.update({name:"XYZ"},{$set:{city:"Mumbai"}});
{
  acknowledged: true,

```

```
    insertedId: null,  
    matchedCount: 1,  
    modifiedCount: 1,  
    upsertedCount: 0  
  }  
bank> db.customer.find()  
[  
  {  
    ...  
    ...  
    {  
      _id: ObjectId("65418bdf1a2c5c7b3297959"),  
      id: 'cus008',  
      name: 'XYZ',  
      street: 'sangamner',  
      city: 'Mumbai'  
    }  
  ]
```

```
bank> db.branch.drop()  
true  
bank> db.branch.find()
```

```
test> use student  
switched to db student  
student>  
db.stud.insertOne({id:1,name:"Pramod",address:"sangamner",mono:9112904855});  
{  
  acknowledged: true,  
  insertedId: ObjectId("6543a87599c6af16de25c4b7")  
}
```

```
student>
db.stud.insertMany([{id:2,name:"Atre",address:"Nasik",mono:9112900000},{id:3,name:"Haral",address:"Nasik",mono:91129000},{id:4,name:"Shubham",address:"Nasik",mono:9112900},{id:5,name:"Sidhi",address:"Sangamner",mono:911290},{id:6,name:"shub",address:"Pune",mono:9112}]);
```

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("6543aa5f99c6af16de25c4b8"),
    '1': ObjectId("6543aa5f99c6af16de25c4b9"),
    '2': ObjectId("6543aa5f99c6af16de25c4ba"),
    '3': ObjectId("6543aa5f99c6af16de25c4bb"),
    '4': ObjectId("6543aa5f99c6af16de25c4bc")
  }
}
```

```
student> db.stud.find().explain();
{
  explainVersion: '1',
  queryPlanner: {
    namespace: 'student.stud',
    indexFilterSet: false,
    parsedQuery: {},
    queryHash: '17830885',
    planCacheKey: '17830885',
    maxIndexedOrSolutionsReached: false,
    maxIndexedAndSolutionsReached: false,
    maxScansToExplodeReached: false,
    winningPlan: { stage: 'COLLSCAN', direction: 'forward' },
    rejectedPlans: []
  },
  command: { find: 'stud', filter: {}, '$db': 'student' },
  serverInfo: {
    host: 'pramod',
```

```

    port: 27017,
    version: '6.0.5',
    gitVersion: 'c9a99c120371d4d4c52cbb15dac34a36ce8d3b1d'
  },
  serverParameters: {
    internalQueryFacetBufferSizeBytes: 104857600,
    internalQueryFacetMaxOutputDocSizeBytes: 104857600,
    internalLookupStageIntermediateDocumentMaxSizeBytes: 104857600,
    internalDocumentSourceGroupMaxMemoryBytes: 104857600,
    internalQueryMaxBlockingSortMemoryUsageBytes: 104857600,
    internalQueryProhibitBlockingMergeOnMongoS: 0,
    internalQueryMaxAddToSetBytes: 104857600,
    internalDocumentSourceSetWindowFieldsMaxMemoryBytes: 104857600
  },
  ok: 1
}

student> db.stud.ensureIndex({id:1,address:1});
[ 'id_1_address_1' ]

student> db.stud.find().explain("executionStats");
{
  explainVersion: '1',
  queryPlanner: {
    namespace: 'student.stud',
    indexFilterSet: false,
    parsedQuery: {},
    queryHash: '17830885',
    planCacheKey: '17830885',
    maxIndexedOrSolutionsReached: false,
    maxIndexedAndSolutionsReached: false,
    maxScansToExplodeReached: false,
    winningPlan: { stage: 'COLLSCAN', direction: 'forward' },
    rejectedPlans: []
  }
}

```

```
    },
    executionStats: {
      executionSuccess: true,
      nReturned: 6,
      executionTimeMillis: 0,
      totalKeysExamined: 0,
      totalDocsExamined: 6,
      executionStages: {
        stage: 'COLLSCAN',
        nReturned: 6,
        executionTimeMillisEstimate: 0,
        works: 8,
        advanced: 6,
        needTime: 1,
        needYield: 0,
        saveState: 0,
        restoreState: 0,
        isEOF: 1,
        direction: 'forward',
        docsExamined: 6
      }
    },
    command: { find: 'stud', filter: {}, '$db': 'student' },
    serverInfo: {
      host: 'pramod',
      port: 27017,
      version: '6.0.5',
      gitVersion: 'c9a99c120371d4d4c52cbb15dac34a36ce8d3b1d'
    },
    serverParameters: {
      internalQueryFacetBufferSizeBytes: 104857600,
      internalQueryFacetMaxOutputDocSizeBytes: 104857600,
```



```

    internalLookupStageIntermediateDocumentMaxSizeBytes: 104857600,
    internalDocumentSourceGroupMaxMemoryBytes: 104857600,
    internalQueryMaxBlockingSortMemoryUsageBytes: 104857600,
    internalQueryProhibitBlockingMergeOnMongoS: 0,
    internalQueryMaxAddToSetBytes: 104857600,
    internalDocumentSourceSetWindowFieldsMaxMemoryBytes: 104857600
  },
  ok: 1
}

```

student> db.stud.getIndexes()

```

[
  { v: 2, key: { _id: 1 }, name: '_id_' },
  { v: 2, key: { id: 1, address: 1 }, name: 'id_1_address_1' }
]

```

student> db.stud1.createIndex({id:1,address:1});

id_1_address_1

student> db.stud1.getIndexes()

```

[
  { v: 2, key: { _id: 1 }, name: '_id_' },
  { v: 2, key: { id: 1, address: 1 }, name: 'id_1_address_1' }
]

```

student> db.stud.find({address:"Nasik"}).explain("executionStats");

```

{
  explainVersion: '1',
  queryPlanner: {
    namespace: 'student.stud',
    indexFilterSet: false,
    parsedQuery: { address: { '$eq': 'Nasik' } },
    queryHash: '67A566EC',
    planCacheKey: 'BD9CCA81',
    maxIndexedOrSolutionsReached: false,
    maxIndexedAndSolutionsReached: false,

```

```
maxScansToExplodeReached: false,
winningPlan: {
  stage: 'COLLSCAN',
  filter: { address: { '$eq': 'Nasik' } },
  direction: 'forward'
},
rejectedPlans: []
},
executionStats: {
  executionSuccess: true,
  nReturned: 3,
  executionTimeMillis: 0,
  totalKeysExamined: 0,
  totalDocsExamined: 6,
  executionStages: {
    stage: 'COLLSCAN',
    filter: { address: { '$eq': 'Nasik' } },
    nReturned: 3,
    executionTimeMillisEstimate: 0,
    works: 8,
    advanced: 3,
    needTime: 4,
    needYield: 0,
    saveState: 0,
    restoreState: 0,
    isEOF: 1,
    direction: 'forward',
    docsExamined: 6
  }
},
command: { find: 'stud', filter: { address: 'Nasik' }, '$db': 'student' },
serverInfo: {
```

```

    host: 'pramod',
    port: 27017,
    version: '6.0.5',
    gitVersion: 'c9a99c120371d4d4c52cbb15dac34a36ce8d3b1d'
  },
  serverParameters: {
    internalQueryFacetBufferSizeBytes: 104857600,
    internalQueryFacetMaxOutputDocSizeBytes: 104857600,
    internalLookupStageIntermediateDocumentMaxSizeBytes: 104857600,
    internalDocumentSourceGroupMaxMemoryBytes: 104857600,
    internalQueryMaxBlockingSortMemoryUsageBytes: 104857600,
    internalQueryProhibitBlockingMergeOnMongoS: 0,
    internalQueryMaxAddToSetBytes: 104857600,
    internalDocumentSourceSetWindowFieldsMaxMemoryBytes: 104857600
  },
  ok: 1
}
student> db.stud1.find();
[
  {
    _id: ObjectId("6543ada499c6af16de25c4bd"),
    id: 2,
    name: 'Atre',
    address: 'Nasik',
    mono: 9112900000
  },
  {
    _id: ObjectId("6543ada499c6af16de25c4be"),
    id: 3,
    name: 'Haral',
    address: 'Nasik',
    mono: 91129000
  }
]

```

```
},
{
  _id: ObjectId("6543ada499c6af16de25c4bf"),
  id: 4,
  name: 'Shubham',
  address: 'Nasik',
  mono: 9112900
},
{
  _id: ObjectId("6543ada499c6af16de25c4c0"),
  id: 5,
  name: 'Sidhi',
  address: 'Sangamner',
  mono: 911290
},
{
  _id: ObjectId("6543ada499c6af16de25c4c1"),
  id: 6,
  name: 'shub',
  address: 'Pune',
  mono: 9112
}
]
```

Assignment no 03

```
>use student
> db.student.insert({"std_name":"Atre","sub_name":"toc","sub_marks":86});
WriteResult({ "nInserted" : 1 })
> db.student.insert({"std_name":"Atre","sub_name":"os","sub_marks":80});
WriteResult({ "nInserted" : 1 })
> db.student.insert({"std_name":"Atre","sub_name":"Adbms","sub_marks":81});
WriteResult({ "nInserted" : 1 })
> db.student.insert({"std_name":"Sanket","sub_name":"Adbms","sub_marks":80});
WriteResult({ "nInserted" : 1 })
> db.student.insert({"std_name":"Sanket","sub_name":"os","sub_marks":88});
WriteResult({ "nInserted" : 1 })
> db.student.insert({"std_name":"Sanket","sub_name":"toc","sub_marks":98});
WriteResult({ "nInserted" : 1 })
> db.student.insert({"std_name":"Pramod","sub_name":"toc","sub_marks":98});
WriteResult({ "nInserted" : 1 })
> db.student.insert({"std_name":"Pramod","sub_name":"os","sub_marks":88});
WriteResult({ "nInserted" : 1 })
> db.student.insert({"std_name":"Pramod","sub_name":"Adbms","sub_marks":88});
WriteResult({ "nInserted" : 1 })
> var map =function(){emit(this.std_name,this.sub_marks);};
> var reduce =function(std_name,sub_marks){return Array.sum(sub_marks);};
> db.student.mapReduce(map,reduce,{out:'total'});
{
  "result" : "total",
  "timeMillis" : 663,
  "counts" : {
    "input" : 9,
    "emit" : 9,
    "reduce" : 3,
    "output" : 3
  },
  "ok" : 1
}
> db.total.find();
{ "_id" : "Atre", "value" : 247 }
{ "_id" : "Pramod", "value" : 274 }
{ "_id" : "Sanket", "value" : 266 }
> db.student.find();
{ "_id" : ObjectId("654359252281a926512802fc"), "std_name" : "Atre", "sub_name" : "toc", "sub_marks" : 86 }
{ "_id" : ObjectId("6543593a2281a926512802fd"), "std_name" : "Atre", "sub_name" : "os", "sub_marks" : 80 }
{ "_id" : ObjectId("6543594d2281a926512802fe"), "std_name" : "Atre", "sub_name" : "Adbms", "sub_marks" : 81 }
{ "_id" : ObjectId("6543595c2281a926512802ff"), "std_name" : "Sanket", "sub_name" : "Adbms", "sub_marks" : 80 }
{ "_id" : ObjectId("654359672281a92651280300"), "std_name" : "Sanket", "sub_name" : "os", "sub_marks" : 88 }
```

```
{ "_id" : ObjectId("654359712281a92651280301"), "std_name" : "Sanket", "sub_name" :  
"toc", "sub_marks" : 98 }  
{ "_id" : ObjectId("654359822281a92651280302"), "std_name" : "Pramod", "sub_name" :  
"toc", "sub_marks" : 98 }  
{ "_id" : ObjectId("6543598e2281a92651280303"), "std_name" : "Pramod", "sub_name" :  
"os", "sub_marks" : 88 }  
{ "_id" : ObjectId("6543599b2281a92651280304"), "std_name" : "Pramod", "sub_name" :  
"Adbms", "sub_marks" : 88 }  
> db.total.find();  
{ "_id" : "Atre", "value" : 247 }  
{ "_id" : "Pramod", "value" : 274 }  
{ "_id" : "Sanket", "value" : 266 }
```