DBMSL Assignment No 9
Name:Himanshu Vijay Lonkar
Batch:T3

```
Output:
    #Start:

        test> show dbs
            admin   40.00 KiB
            config  12.00 KiB
            local   40.00 KiB

        test> use Articles
            switched to db Articles
            Articles> db.createCollection('articles')
            { ok: 1 }

        Articles> show dbs
            Articles   8.00 KiB
            admin     40.00 KiB
            config    60.00 KiB
            local     40.00 KiB

    1.Insert 5 Data:

        Articles> db.articles.insertOne({Title:'Entrepreneur',Content:'Businesses',Author:'Bhavik Sarkhedi',Author_age:20,Dop:new Date('2023-10-08'),Category:'FreeLancing',Comments:[{Name:'XYZ',Remarks:'Good'},{Name:'ABC',Remarks:'Can be Better'}]})
            {
              acknowledged: true,
              insertedId: ObjectId("6527c74865659a592e3e2814")
            }

        Articles> db.articles.insertMany([{Title:'Novelist',Content:'English Literature',Author:'Chetan Bhagat',Author_age:25,Dop:new Date('2023-02-10'),Category:'FreeLancing',Comments:[{Name:'LMN',Remarks:'Good'},{Name:'PQR',Remarks:'Can be Better'}]},{Title:'Story Books',Content:'Horror',Author:'A.B.Sharma',Author_age:30,Dop:new Date('2023-3-15'),Category:'FreeLancing',Comments:[{Name:'AAA',Remarks:'Good'},{Name:'BBB',Remarks:'Can be Better'}]}])
            {
              acknowledged: true,
              insertedIds: {
                '0': ObjectId("6527c9dc65659a592e3e2815"),
                '1': ObjectId("6527c9dc65659a592e3e2816")
              }
            }

        Articles> db.articles.insertMany([{Title:'Novelist',Content:'Indian Literature',Author:'Anita Desai',Author_age:16,Dop:new Date('2023-05-22'),Category:'FreeLancing',Comments:[{Name:'QQQ',Remarks:'Good'},{Name:'PPP',Remarks:'Can be Better'}]},{Title:'The God of Small Things',Content:'ScreenPlays',Author:'Arundhati Roy',Author_age:16,Dop:new Date('2022-4-20'),Category:'FreeLancing',Comments:[{Name:'DDD',Remarks:'Good'},{Name:'CCC',Remarks:'Can be Better'}]}])
            {
              acknowledged: true,
              insertedIds: {
                '0': ObjectId("6527cae665659a592e3e2817"),
                '1': ObjectId("6527cae665659a592e3e2818")
              }
            }

    1.2 Display all records:

        Articles> db.articles.find().pretty()
                [
  {
    _id: ObjectId("6527c74865659a592e3e2814"),
    Title: 'Entrepreneur',
    Content: 'Businesses',
    Author: 'Bhavik Sarkhedi',
    Author_age: 20,
    Dop: ISODate("2023-10-08T00:00:00.000Z"),
    Category: 'FreeLancing',
    Comments: [
      { Name: 'XYZ', Remarks: 'Good' },
      { Name: 'ABC', Remarks: 'Can be Better' }
    ]
  },
  {
    _id: ObjectId("6527c9dc65659a592e3e2815"),
    Title: 'Novelist',
    Content: 'English Literature',
    Author: 'Chetan Bhagat',
    Author_age: 25,
    Dop: ISODate("2023-02-10T00:00:00.000Z"),
    Category: 'FreeLancing',
    Comments: [
      { Name: 'LMN', Remarks: 'Good' },
      { Name: 'PQR', Remarks: 'Can be Better' }
    ]
  },
  {
    _id: ObjectId("6527c9dc65659a592e3e2816"),
    Title: 'Story Books',
    Content: 'Horror',
    Author: 'A.B.Sharma',
    Author_age: 30,
    Dop: ISODate("2023-03-14T18:30:00.000Z"),
    Category: 'FreeLancing',
    Comments: [
      { Name: 'AAA', Remarks: 'Good' },
      { Name: 'BBB', Remarks: 'Can be Better' }
    ]
  },
  {
    _id: ObjectId("6527cae665659a592e3e2817"),
    Title: 'Novelist',
    Content: 'Indian Literature',
    Author: 'Anita Desai',
    Author_age: 16,
    Dop: ISODate("2023-05-22T00:00:00.000Z"),
    Category: 'FreeLancing',
    Comments: [
      { Name: 'QQQ', Remarks: 'Good' },
      { Name: 'PPP', Remarks: 'Can be Better' }
    ]
  },
  {
    _id: ObjectId("6527cae665659a592e3e2818"),
    Title: 'The God of Small Things',
    Content: 'ScreenPlays',
    Author: 'Arundhati Roy',
    Author_age: 16,
    Dop: ISODate("2022-04-19T18:30:00.000Z"),
    Category: 'FreeLancing',
    Comments: [
      { Name: 'DDD', Remarks: 'Good' },
      { Name: 'CCC', Remarks: 'Can be Better' }
    ]
  }
]
```

2.Display first document found in db:

```
  Articles> db.articles.find().limit(1)
      [
        {
          _id: ObjectId("6527c74865659a592e3e2814"),
          Title: 'Entrepreneur',
          Content: 'Businesses',
          Author: 'Bhavik Sarkhedi',
          Author_age: 20,
          Dop: ISODate("2023-10-08T00:00:00.000Z"),
          Category: 'FreeLancing',
          Comments: [
            { Name: 'XYZ', Remarks: 'Good' },
            { Name: 'ABC', Remarks: 'Can be Better' }
          ]
        }
      ]
```

3.Display first document belonging to a certain author says "Sharma":

```
  Articles> db.articles.find({Author:/Sharma/})
      [
        {
          _id: ObjectId("6527c9dc65659a592e3e2816"),
          Title: 'Story Books',
          Content: 'Horror',
          Author: 'A.B.Sharma',
          Author_age: 30,
          Dop: ISODate("2023-03-14T18:30:00.000Z"),
          Category: 'FreeLancing',
          Comments: [
            { Name: 'AAA', Remarks: 'Good' },
            { Name: 'BBB', Remarks: 'Can be Better' }
          ]
        }
      ]
```

4.Modify the comment by certain person on a certain article:

```
  Articles>
  db.articles.update({Title:'Entrepreneur','Comments.Name':'ABC'},{$set:{'Comments.
  $.Remarks':'Excellent'}})
      {
        acknowledged: true,
        insertedId: null,
        matchedCount: 1,
        modifiedCount: 1,
        upsertedCount: 0
      }
      Articles> db.articles.find().limit(1)
      [
        {
          _id: ObjectId("6527c74865659a592e3e2814"),
          Title: 'Entrepreneur',
          Content: 'Businesses',
          Author: 'Bhavik Sarkhedi',
          Author_age: 20,
          Dop: ISODate("2023-10-08T00:00:00.000Z"),
          Category: 'FreeLancing',
          Comments: [
            { Name: 'XYZ', Remarks: 'Good' },
            { Name: 'ABC', Remarks: 'Excellent' }
          ]
        }
      ]
```

5.1 Insert records with save method with object _id:

```
  Articles>
  db.articles.save({_id:10,Title:'DBMS',Content:'MongoDB',Author:'Prof.M.S.Agarwal'
        ,Author_age:30,Dop:new
        Date('2023-10-10'),Category:'Database',Comments:[{Name:'HHH',Remarks:'Better'},{N
        ame:'JJJ',Remarks:'Excellent'}]});

              > db.articles.find({Title:"DBMS"}).pretty()

              {
        "_id" : 10,
        "Title" : "DBMS",
        "Content" : "MongoDB",
        "Author" : "Prof.M.S.Agarwal",
        "Author_age" : 30,
        "Dop" : ISODate("2023-10-10T00:00:00Z"),
        "Category" : "Database",
        "Comments" : [
            {
                "Name" : "HHH",
                "Remarks" : "Better"
            },
            {
                "Name" : "JJJ",
                "Remarks" : "Excellent"
            },
            {
                "Name" : "FFF",
                "Remarks" : "Nice"
            }
        ]
    }
```

5.2 Insert records with save method without object _id:

```
    >
    db.articles.save({Title:'TOC',Content:'DFA',Author:'Prof.R.R.Bathe',Author_age:30
    ,Dop:new
    Date('2023-10-20'),Category:'Automata',Comments:[{Name:'EEE',Remarks:'Better'},
                {Name:'SSS',Remarks:'Excellent'}]});
    WriteResult({ "nInserted" : 1 })

    > db.articles.find({Title:"TOC"}).pretty()
    {
        "_id" : ObjectId("652cd16e3110241d64b6456d"),
        "Title" : "TOC",
        "Content" : "DFA",
        "Author" : "Prof.R.R.Bathe",
        "Author_age" : 30,
        "Dop" : ISODate("2023-10-20T00:00:00Z"),
        "Category" : "Automata",
        "Comments" : [
            {
                "Name" : "EEE",
                "Remarks" : "Better"
            },
            {
                "Name" : "SSS",
                "Remarks" : "Excellent"
            }
        ]
    }
```

6.Update a Collection with save method:

```
    >
    db.articles.save({_id:10,Title:'DBMS',Content:'MySql',Author:'Prof.M.S.Agarwal',A
    uthor_age:25,Dop:new
    Date("2023-10-16"),Category:'Database',Comments:[{Name:'HHH',Remarks:'Good'},
                {Name:'JJJ',Remarks:'Nice'}]})
    WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

```
> db.articles.find({Title:'DBMS'}).pretty()
{
    "_id" : 10,
    "Title" : "DBMS",
    "Content" : "MySql",
    "Author" : "Prof.M.S.Agarwal",
    "Author_age" : 25,
    "Dop" : ISODate("2023-10-16T00:00:00Z"),
    "Category" : "Database",
    "Comments" : [
        {
            "Name" : "HHH",
            "Remarks" : "Good"
        },
        {
            "Name" : "JJJ",
            "Remarks" : "Nice"
        }
    ]
}
```

7.Add one more comment for a particular Title:

```
> db.articles.update({Title:'TOC'},{$push:{Comments:{Name:'FFF',Remarks:'Nice'}}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })

> db.articles.find({Title:'TOC'}).pretty()
{
    "_id" : ObjectId("652cd16e3110241d64b6456d"),
    "Title" : "TOC",
    "Content" : "DFA",
    "Author" : "Prof.R.R.Bathe",
    "Author_age" : 30,
    "Dop" : ISODate("2023-10-20T00:00:00Z"),
    "Category" : "Automata",
    "Comments" : [
        {
            "Name" : "EEE",
            "Remarks" : "Better"
        },
        {
            "Name" : "SSS",
            "Remarks" : "Excellent"
        },
        {
            "Name" : "FFF",
            "Remarks" : "Nice"
        }
    ]
}
```

8.Delete the documents whose author age<18:

```
> db.articles.deleteMany({Author_age:{$lt:18}})
{ "acknowledged" : true, "deletedCount" : 2 }

> db.articles.find().pretty()
{
    "_id" : ObjectId("6528cd46a0571873b9d71186"),
    "Title" : "Entrepreneur",
    "Content" : "Businesses",
    "Author" : "Bhavik Sarkhedi",
    "Author_age" : 20,
    "Dop" : ISODate("2023-10-08T00:00:00Z"),
    "Category" : "FreeLancing",
    "Comments" : [
        {
            "Name" : "XYZ",
            "Remarks" : "Good"
        },
        {
            "Name" : "ABC",
            "Remarks" : "Excellent"
        }
    ]
}
{
    "_id" : ObjectId("6528cee6a0571873b9d71187"),
    "Title" : "Novelist",
    "Content" : "English Literature",
    "Author" : "Chetan Bhagat",
    "Author_age" : 25,
    "Dop" : ISODate("2023-02-10T00:00:00Z"),
    "Category" : "FreeLancing",
    "Comments" : [
        {
            "Name" : "LMN",
            "Remarks" : "Good"
        },
        {
            "Name" : "PQR",
            "Remarks" : "Can be Better"
        }
    ]
}
{
    "_id" : ObjectId("6528cee6a0571873b9d71188"),
    "Title" : "Story Books",
    "Content" : "Horror",
    "Author" : "A.B.Sharma",
    "Author_age" : 30,
    "Dop" : ISODate("2023-03-15T00:00:00Z"),
    "Category" : "FreeLancing",
    "Comments" : [
        {
            "Name" : "AAA",
            "Remarks" : "Good"
        },
        {
            "Name" : "BBB",
            "Remarks" : "Can be Better"
        }
    ]
}
{
    "_id" : 10,
    "Title" : "DBMS",
    "Content" : "MySql",
    "Author" : "Prof.M.S.Agarwal",
    "Author_age" : 25,
    "Dop" : ISODate("2023-10-16T00:00:00Z"),
    "Category" : "Database",
    "Comments" : [
        {
            "Name" : "HHH",
            "Remarks" : "Good"
        },
        {
            "Name" : "JJJ",
            "Remarks" : "Nice"
        }
    ]
}
{
```

```
    "_id" : ObjectId("652cd16e3110241d64b6456d"),
    "Title" : "TOC",
    "Content" : "DFA",
    "Author" : "Prof.R.R.Bathe",
    "Author_age" : 30,
    "Dop" : ISODate("2023-10-20T00:00:00Z"),
    "Category" : "Automata",
    "Comments" : [
        {
            "Name" : "EEE",
            "Remarks" : "Better"
        },
        {
            "Name" : "SSS",
            "Remarks" : "Excellent"
        },
        {
            "Name" : "FFF",
            "Remarks" : "Nice"
        }
    ]
}
```

9.Remove all the documents from the collection:

```
> db.articles.deleteMany({})
{ "acknowledged" : true, "deletedCount" : 5 }

> db.articles.find().pretty()
```

10.Delete article collection:

```
> db.articles.drop()
true

> db.getCollectionNames()
[ ]
```