

## Assignment 10 : mongoDB CRUD

```
show dbs
use Asg10
```

1. Create a collection named books.

```
db.createCollection("Books")
```

2. Insert 5 records with field TITLE,DESCRIPTION, BY, URL, TAGS AND LIKES

```
db.Books.insertOne(
... {
... title : "mongodb",
... description : "this is book 1",
... by : "john",
... url : "mongodb.com",
... tags : ["mongodb", "mongo", "nosql"],
... likes : 55
... }
... )
```

```
db.Books.insertOne( { title: "nosql", description: "this is book 2",
by: "adam", url: "nosql.com", tags: ["db", "dbms", "nosql"], likes :
23})
```

```
db.Books.insertMany(
[
    { title : "sql", description : "this is book 3", by : "john",
url : "sql.com", tags:["sql", "rdbms"],likes : 9 },
    { title : "plsql", description : "this is book 4", by :
"hardy", url : "plsql.com", tags:["plsql", "rdbms"],likes : 99},
    {title:"acid", description:"acid properties", by : "john",
url : "acid.com", tags : ["acid", "properties"], likes : 3}
]
)
db.Books.find()
```

## Assignment 10 : mongoDB CRUD

3. Insert 1 more document in collection with additional field of user name and comments.

```
db.Books.insertOne(
... {
... title : "connectivity",
... description : "this is book 6",
... by : "ben",
... url : "connect.com",
... tags : ["dbconnect", "connectivity"],
... likes : 34,
... user : "sohamkulkarni",
... comments : ["Amazing Book", "Easy to understand"]
... }
... )
```

4. Display all the documents whose title is 'mongodb',  
db.Books.find({title : "mongodb"})

5. Display all the documents written by 'john' or whose title is 'mongodb'.

```
db.Books.find({$or : [ {title : "mongodb"}, {by : "john"}] })
```

6. Display all the documents written by 'john' and whose title is 'mongodb'.

```
db.Books.find({$and : [ {title : "mongodb"}, {by : "john"}] })
```

7. Display all the documents whose like is greater than 10.

```
db.Books.find({ likes : {$gt:10} })
```

8. Display all the documents whose like is greater than 100 and whose title is either 'mongodb' or written by 'john'.

```
db.Books.find(
{
  $and:[ {likes:{$gt:25}},
        { $or: [ {title : "mongodb"},
                  {by:'john'}
                ]
        }
  ]
}
)
```

## Assignment 10 : mongoDB CRUD

9. Update the title of 'mongodb' document to 'mongodb overview'

```
db.Books.updateOne(  
  {title:"mongodb"},  
  {$set:  
    {title : "mongodb overview"}  
  }  
)
```

10. Delete the document titled 'nosql overview.'

```
db.Books.deleteOne({title : "nosql"})
```

11. Display exactly two documents written by 'john'.

```
db.Books.find({by:"john"}).limit(2)
```

12. Display the second document published by 'john'. 13. Display all the books in the sorted fashion.

```
db.Books.find({by:"john"}).limit(1).skip(1)
```

13. Display all the books in sorted fashion (ascending by title):

```
db.Books.find().sort({ title: 1 })
```

14. Insert a document using save method

```
var docToSave = {title : "last", description : "the last book ", by  
: "lastAuthor", url : "last.com", tags:["last"], likes : 25}
```

```
db.Books.save(docToSave);
```

TypeError: db.Books.save is not a function

```
db.Books.insertOne(docToSave);
```