

LAB 06

-Jainil Trivedi (CE166)

1. Write queries to write crud operations in MongoDB

INSERT

```
> db.Students.insertMany([ {name:'Harprit',year:4,sem:7,cgpa:9}, {name:'Harshil',year:4,cgpa:8.5} ]);
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("630c9311399ad5138bc9fa96"),
    ObjectId("630c9311399ad5138bc9fa97")
  ]
}
```

FIND/READ

```
> db.Students.find();
{ "_id" : ObjectId("630c9226399ad5138bc9fa95"), "name" : "Jainil", "year" : 4, "sem" : 7, "cgpa" : 8 }
{ "_id" : ObjectId("630c9311399ad5138bc9fa96"), "name" : "Harprit", "year" : 4, "sem" : 7, "cgpa" : 9 }
{ "_id" : ObjectId("630c9311399ad5138bc9fa97"), "name" : "Harshil", "year" : 4, "cgpa" : 8.5 }
> db.Students.find().pretty();
{
  "_id" : ObjectId("630c9226399ad5138bc9fa95"),
  "name" : "Jainil",
  "year" : 4,
  "sem" : 7,
  "cgpa" : 8
}
{
  "_id" : ObjectId("630c9311399ad5138bc9fa96"),
  "name" : "Harprit",
  "year" : 4,
  "sem" : 7,
  "cgpa" : 9
}
{
  "_id" : ObjectId("630c9311399ad5138bc9fa97"),
  "name" : "Harshil",
  "year" : 4,
  "cgpa" : 8.5
}
> db.Students.find({sem:7});
{ "_id" : ObjectId("630c9226399ad5138bc9fa95"), "name" : "Jainil", "year" : 4, "sem" : 7, "cgpa" : 8 }
{ "_id" : ObjectId("630c9311399ad5138bc9fa96"), "name" : "Harprit", "year" : 4, "sem" : 7, "cgpa" : 9 }
>
```

UPDATE

```
> db.Students.update({name:'Harshil'},{$set:{sem:7}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Students.find.pretty();
uncaught exception: TypeError: db.Students.find.pretty is not a function :
@(shell):1:1
> db.Students.find.pretty();
uncaught exception: TypeError: db.Students.find.pretty is not a function :
@(shell):1:1
> db.Students.find().pretty();
{
  "_id" : ObjectId("630c9226399ad5138bc9fa95"),
  "name" : "Jainil",
  "year" : 4,
  "sem" : 7,
  "cgpa" : 8
}
{
  "_id" : ObjectId("630c9311399ad5138bc9fa96"),
  "name" : "Harprit",
  "year" : 4,
  "sem" : 7,
  "cgpa" : 9
}
{
  "_id" : ObjectId("630c9311399ad5138bc9fa97"),
  "name" : "Harshil",
  "year" : 4,
  "cgpa" : 8.5,
  "sem" : 7
}
> □
```

DELETE

```
> db.Students.remove({name:'Raumil'});
WriteResult({ "nRemoved" : 1 })
> db.Students.find().pretty();
{
  "_id" : ObjectId("630c9226399ad5138bc9fa95"),
  "name" : "Jainil",
  "year" : 4,
  "sem" : 7,
  "cgpa" : 8,
  "Hobbies" : "Chess"
}
{
  "_id" : ObjectId("630c9311399ad5138bc9fa96"),
  "name" : "Harprit",
  "year" : 4,
  "sem" : 7,
  "cgpa" : 9,
  "Hobbies" : "Chess"
}
{
  "_id" : ObjectId("630c9311399ad5138bc9fa97"),
  "name" : "Harshil",
  "year" : 4,
  "cgpa" : 8.5,
  "sem" : 7,
  "Hobbies" : "Chess"
}
> □
```

2. To practice MapReduce programming in MongoDB.

```
> db.books.find().pretty();
{
  "_id" : 1,
  "category" : "Machine Learning",
  "name" : "Machine Learning for hackers",
  "qty" : 25,
  "price" : 400,
  "rol" : 30,
  "pages" : 350
}
{
  "_id" : 2,
  "category" : "Business Intelligence",
  "name" : "Fundamentals of Business Analytics",
  "qty" : 55,
  "price" : 500,
  "rol" : 30,
  "pages" : 250
}
{
  "_id" : 3,
  "category" : "Analytics",
  "name" : "Competing on Analytics",
  "qty" : 8,
  "price" : 150,
  "rol" : 20,
  "pages" : 150
}
{ "_id" : 4, "name" : "Visualizing Data", "qty" : 12, "pages" : 450 }
{
  "_id" : 5,
  "name" : "Learning R",
  "price" : 850,
  "qty" : 5,
  "pages" : 120
}
> 
```

```
var map = function(){
.. if(this.pages>350)
..   emit('Big',1);
.. else
..   emit('Small',1);
.. }
var map = function(){ if(this.pages>300)   emit('Big',1); else   emit('Small',1); }
var reduce = function(key,values) {
.. return sum(values);
.. }
db.books.mapReduce(
.. map,
.. reduce,
.. {out: 'outList'}
.. );
```

```

> var reduce = function(key,values) { return Array.sum(values); }
> db.books.mapReduce( map, reduce, {out: 'outList'} );
{ "result" : "outList", "ok" : 1 }
> outlist.find().pretty();
uncaught exception: ReferenceError: outlist is not defined :
@(shell):1:1
> outList.find().pretty();
uncaught exception: ReferenceError: outList is not defined :
@(shell):1:1
> db.books.mapReduce( map, reduce, {out: 'outList',query:{}} );
{ "result" : "outList", "ok" : 1 }
> outList.find().pretty();
uncaught exception: ReferenceError: outList is not defined :
@(shell):1:1
> db.outList.find().pretty();
{ "_id" : "Big", "value" : 2 }
{ "_id" : "Small", "value" : 3 }
> 

```

3. To practice import, export and aggregation in MongoDB.

```

hadoop@celab2-ThinkCentre-neo-50s-Gen-3:~/Desktop/Lab$ mongoimport --db imp --collection=csv --file=House_Rent_Dataset.csv --type=csv --field
File=Dataset_Glossary.txt
2022-08-29T17:26:54.263+0530    connected to: mongod://localhost/
2022-08-29T17:26:54.411+0530    4747 document(s) imported successfully. 0 document(s) failed to import.
hadoop@celab2-ThinkCentre-neo-50s-Gen-3:~/Desktop/Lab$ 

```