MongoDB\_Lab1

1 – open mongo shell and view the help

mongo

help;

2 – identify your current working database and show list of available databases

> show dbs;

> db;

test

3 – create a new database called Iti and create a collection named “students”. Insert whatever data you want about yourself (include name and age in your details).

> use iti;

> db.students.insert({name:"reham",age:24});

4– show a list of available databases. What did you notice?

> show dbs;

admin 0.000GB

config 0.000GB

iti 0.000GB

local 0.000GB

5 – Insert un-structured or semi-structured data for 10 of your friends (include name and age in your details. The documents should have different types of data i.e., arrays, strings, documents, integers).

> var students=[{"name":"rana","age":23},{"name":"nor","age":25},

... {"name":"mariam","age":22},

... {"name":"dina","age":23},

... {"name":"rowan","age":23},

... {"name":"zahia","age":22},

... {"name":"aya","age":25}]

> ;

> db.students.insert(students);

> db.students.insertMany([{"name":"shaimaa","age":20},{"name":"hoda","age":21}, {"name":"maram","age":20}]);

6 – Search for your object by name.

> db.students.find({"name":"reham"});

{ "\_id" : ObjectId("6239cd6202d04055de436f29"), "name" : "reham", "age" : 24 }

7– Search for yourf riend(s) byage.

> db.students.find({"age":23});

{ "\_id" : ObjectId("6239d27802d04055de436f31"), "name" : "rana", "age" : 23 }

{ "\_id" : ObjectId("6239d27802d04055de436f34"), "name" : "dina", "age" : 23 }

{ "\_id" : ObjectId("6239d27802d04055de436f35"), "name" : "rowan", "age" : 23 }

8 – Search for all of your friends whose age is older than yours.

> db.students.find({"age":{$gt:24}});

{ "\_id" : ObjectId("6239d27802d04055de436f32"), "name" : "nor", "age" : 25 }

{ "\_id" : ObjectId("6239d27802d04055de436f37"), "name" : "aya", "age" : 25 }

**9** – delete any of your friends by id.

>db.students.deleteOne({"\_id":ObjectId("6239d27802d04055de436f37")});

10 – view all documents in students' collection in a prettified format.

> db.students.find().pretty();

{

"\_id" : ObjectId("6239cd6202d04055de436f29"),

"name" : "reham",

"age" : 24

}

{

"\_id" : ObjectId("6239d27802d04055de436f31"),

"name" : "rana",

"age" : 23

}

{ "\_id" : ObjectId("6239d27802d04055de436f32"), "name" : "nor", "age" : 25 }

{

"\_id" : ObjectId("6239d27802d04055de436f33"),

"name" : "mariam",

"age" : 22

}

{

"\_id" : ObjectId("6239d27802d04055de436f34"),

"name" : "dina",

"age" : 23

}

{

"\_id" : ObjectId("6239d27802d04055de436f35"),

"name" : "rowan",

"age" : 23

}

{

"\_id" : ObjectId("6239d27802d04055de436f36"),

"name" : "zahia",

"age" : 22

}

{

"\_id" : ObjectId("6239d7b03d629285d40f3114"),

"name" : "shaimaa",

"age" : 20

}

{

"\_id" : ObjectId("6239d7b03d629285d40f3115"),

"name" : "hoda",

"age" : 21

}

{

"\_id" : ObjectId("6239d7b03d629285d40f3116"),

"name" : "maram",

"age" : 20

}

11 – count all documents in students' collection. (self-learning)

> db.students.count();

**---------------------------------------------------------**

**part 2**

1- Create database with name ems

> use ems;

2- Insert the following data into "faculty" collection

> db.createCollection("faculty");

> db.faculty.insertMany([{ "name":"Krish", "age":35,"gender":"M","exp":10,subjects:["DS","C","OS"],"type":"Full Time","qualification":"M.Tech" },

... { "name":"Manoj", "age":38,"gender":"M","exp":12,subjects:["JAVA","DBMS"],"type":"Full Time", "qualification":"Ph.D"},

... { "name":"Anush", "age":32,"gender":"F","exp":8,subjects:["C","CPP"],"type":"Part Time","qualification":"M.Tech" },

... { "name":"Suresh", "age":40,"gender":"M","exp":9,subjects:["JAVA","DBMS","NETWORKING"],"type":"Full Time", "qualification":"Ph.D"},

... { "name":"Rajesh", "age":35,"gender":"M","exp":7,subjects:["DS","C","OS"],"type":"Full Time","qualification":"M.Tech" },

... { "name":"Mani", "age":38,"gender":"F","exp":10,subjects:["JAVA","DBMS","OS"],"type":"Part Time", "qualification":"Ph.D"},

... { "name":"Sivani", "age":32,"gender":"F","exp":8,subjects:["C","CPP","MATHS"],"type":"Part Time","qualification":"M.Tech" },

... { "name":"Nagesh", "age":39,"gender":"M","exp":11,subjects:["JAVA","DBMS","NETWORKING"],"type":"Full Time", "qualification":"Ph.D"},

... { "name":"Nagesh", "age":35,"gender":"M","exp":9,subjects:["JAVA",".Net","NETWORKING"],"type":"Full Time", "qualification":"Ph.D"},

... { "name":"Latha", "age":40,"gender":"F","exp":13,subjects:["MATHS"],"type":"Full Time", "qualification":"Ph.D"}]);

{ "name":"Krish", "age":35,"gender":"M","exp":10,subjects:["DS","C","OS"],"type":"Full Time","qualification":"M.Tech" },

{ "name":"Manoj", "age":38,"gender":"M","exp":12,subjects:["JAVA","DBMS"],"type":"Full Time", "qualification":"Ph.D"},

{ "name":"Anush", "age":32,"gender":"F","exp":8,subjects:["C","CPP"],"type":"Part Time","qualification":"M.Tech" },

{ "name":"Suresh", "age":40,"gender":"M","exp":9,subjects:["JAVA","DBMS","NETWORKING"],"type":"Full Time", "qualification":"Ph.D"},

{ "name":"Rajesh", "age":35,"gender":"M","exp":7,subjects:["DS","C","OS"],"type":"Full Time","qualification":"M.Tech" },

{ "name":"Mani", "age":38,"gender":"F","exp":10,subjects:["JAVA","DBMS","OS"],"type":"Part Time", "qualification":"Ph.D"},

{ "name":"Sivani", "age":32,"gender":"F","exp":8,subjects:["C","CPP","MATHS"],"type":"Part Time","qualification":"M.Tech" },

{ "name":"Nagesh", "age":39,"gender":"M","exp":11,subjects:["JAVA","DBMS","NETWORKING"],"type":"Full Time", "qualification":"Ph.D"},

{ "name":"Nagesh", "age":35,"gender":"M","exp":9,subjects:["JAVA",".Net","NETWORKING"],"type":"Full Time", "qualification":"Ph.D"},

{ "name":"Latha", "age":40,"gender":"F","exp":13,subjects:["MATHS"],"type":"Full Time", "qualification":"Ph.D"}

1. Get the details of all the faculty.

db.faculty.find();

2. Get the count of all faculty members.

> db.faculty.count();

10

3. Get all the faculty members whose qualification is “Ph.D”.

> db.faculty.find({"qualification":"Ph.D"});

{ "\_id" : ObjectId("6239dab63d629285d40f3118"), "name" : "Manoj", "age" : 38, "gender" : "M", "exp" : 12, "subjects" : [ "JAVA", "DBMS" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311a"), "name" : "Suresh", "age" : 40, "gender" : "M", "exp" : 9, "subjects" : [ "JAVA", "DBMS", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311c"), "name" : "Mani", "age" : 38, "gender" : "F", "exp" : 10, "subjects" : [ "JAVA", "DBMS", "OS" ], "type" : "Part Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311e"), "name" : "Nagesh", "age" : 39, "gender" : "M", "exp" : 11, "subjects" : [ "JAVA", "DBMS", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311f"), "name" : "Nagesh", "age" : 35, "gender" : "M", "exp" : 9, "subjects" : [ "JAVA", ".Net", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f3120"), "name" : "Latha", "age" : 40, "gender" : "F", "exp" : 13, "subjects" : [ "MATHS" ], "type" : "Full Time", "qualification" : "Ph.D" }

4. Get all the faculty members whose experience is between 8 to 12 years.

> db.faculty.find({"exp":{$gt:8,$lt:12}});

{ "\_id" : ObjectId("6239dab63d629285d40f3117"), "name" : "Krish", "age" : 35, "gender" : "M", "exp" : 10, "subjects" : [ "DS", "C", "OS" ], "type" : "Full Time", "qualification" : "M.Tech" }

{ "\_id" : ObjectId("6239dab63d629285d40f311a"), "name" : "Suresh", "age" : 40, "gender" : "M", "exp" : 9, "subjects" : [ "JAVA", "DBMS", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311c"), "name" : "Mani", "age" : 38, "gender" : "F", "exp" : 10, "subjects" : [ "JAVA", "DBMS", "OS" ], "type" : "Part Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311e"), "name" : "Nagesh", "age" : 39, "gender" : "M", "exp" : 11, "subjects" : [ "JAVA", "DBMS", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311f"), "name" : "Nagesh", "age" : 35, "gender" : "M", "exp" : 9, "subjects" : [ "JAVA", ".Net", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

5. Get all the faculty members who teach “MATHS” or “NETWORKING”.

>db.faculty.find({$or:[{"subjects":"MATHS"},{"subjects":"NETWORKING"}]});

{ "\_id" : ObjectId("6239dab63d629285d40f311a"), "name" : "Suresh", "age" : 40, "gender" : "M", "exp" : 9, "subjects" : [ "JAVA", "DBMS", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311d"), "name" : "Sivani", "age" : 32, "gender" : "F", "exp" : 8, "subjects" : [ "C", "CPP", "MATHS" ], "type" : "Part Time", "qualification" : "M.Tech" }

{ "\_id" : ObjectId("6239dab63d629285d40f311e"), "name" : "Nagesh", "age" : 39, "gender" : "M", "exp" : 11, "subjects" : [ "JAVA", "DBMS", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311f"), "name" : "Nagesh", "age" : 35, "gender" : "M", "exp" : 9, "subjects" : [ "JAVA", ".Net", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f3120"), "name" : "Latha", "age" : 40, "gender" : "F", "exp" : 13, "subjects" : [ "MATHS" ], "type" : "Full Time", "qualification" : "Ph.D" }

6. Get all the faculty members who teach “MATHS” and whose age is more than 30 years and qualification must be “Ph.D”.

> db.faculty.find({$and:[{"subjects":"MATHS"},{"qualification":"Ph.D"},{"age":{$gt:30}}]});

{ "\_id" : ObjectId("6239dab63d629285d40f3120"), "name" : "Latha", "age" : 40, "gender" : "F", "exp" : 13, "subjects" : [ "MATHS" ], "type" : "Full Time", "qualification" : "Ph.D" }

7. Get all the faculty members who are working part-time or who teach “JAVA”.

> db.faculty.find({$or:[{"subjects":"JAVA"},{"type":"part-time"}]});

{ "\_id" : ObjectId("6239dab63d629285d40f3118"), "name" : "Manoj", "age" : 38, "gender" : "M", "exp" : 12, "subjects" : [ "JAVA", "DBMS" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311a"), "name" : "Suresh", "age" : 40, "gender" : "M", "exp" : 9, "subjects" : [ "JAVA", "DBMS", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311c"), "name" : "Mani", "age" : 38, "gender" : "F", "exp" : 10, "subjects" : [ "JAVA", "DBMS", "OS" ], "type" : "Part Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311e"), "name" : "Nagesh", "age" : 39, "gender" : "M", "exp" : 11, "subjects" : [ "JAVA", "DBMS", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

{ "\_id" : ObjectId("6239dab63d629285d40f311f"), "name" : "Nagesh", "age" : 35, "gender" : "M", "exp" : 9, "subjects" : [ "JAVA", ".Net", "NETWORKING" ], "type" : "Full Time", "qualification" : "Ph.D" }

>

8. Add the following new faculty members:

{ "name":"Suresh Babu", "age":55, "gender":"M", "exp":25, subjects: ["MATHS","DE"], "type":"Full Time", "qualification":"Ph.D"}

> db.faculty.insert({"name":"Suresh Babu", "age":55, "gender":"M", "exp":25, subjects: ["MATHS","DE"], "type":"Full Time", "qualification":"Ph.D"});

9. Update the data of all faculty members by incrementing their age and exp by one year.

db.faculty.updateMany({},{$inc:{"age":1 ,"exp":1}});

10. Update the faculty “Sivani” with the following data: update qualification to “Ph.D” and type to “Full Time”.

> db.faculty.updateMany({"name":"Sivani"},{$set:{"qualification":"Ph.D" ,"type":"Full Time"}});

11. Update all faculty members who are teaching “MATHS” such that they should now also teach “PSK”.

db.faculty.updateMany({"subjects":"MATHS"},{$push: {subjects: 'PSK'}});

12. Delete all faculty members whose age is more than 55 years.

> db.faculty.deleteMany({"age":{$gt:55}});

13. Get only the name and qualification of all faculty members.

> db.faculty.find({},{"name":1,"qualifications":1,"\_id":0});

14. Get the name, qualification and exp of all faculty members and display the same in ascending order of exp.

> db.faculty.find({},{"name":1,"qualifications":1,"exp":1,"\_id":0}).sort({"exp":1});

15. Sort the faculty details by their age (descending order) and get the details of the first five faculty members only.

db.faculty.find().sort({age:-1}).limit(5);