**MongoDB\_Lab1**

1 – open mongo shell and view the help

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

admin 40.00 KiB

config 72.00 KiB

local 72.00 KiB

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\_Database

<iti\_Database>db.Students.insert({name:"Farah"age:23})

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

ITI 572.00 KiB

ITIDB 348.00 KiB

admin 40.00 KiB

config 72.00 KiB

local 72.00 KiB

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).

db.Students.insert({name: "Hadeer", age: 23,courses:["javascript","mongodb","C#"]})

db.Students.insert({name: "Ahmed", age: 23,courses:["javascript","mongodb","C#"]},{name: "Mohamed", age: 23,courses:["javascript","mongodb","C#"]})

db.Students.insert({name: "Sohila", age: 23,courses:["javascript","mongodb","C#"]},{name: "Momen", age: 23,courses:["javascript","mongodb","C#"]},{name: "Mariam", age: 23,courses:["javascript","mongodb","C#"]},{name: "Shahenda", age: 24,courses:["javascript","mongodb","C#"]},{name: "Hadeer", age: 23,courses:["javascript","mongodb","C#"]},{name: "Areej", age: 23,courses:["javascript","mongodb","C#"]},{name: "Aya", age: 23,courses:["javascript","mongodb","C#"]})

{

acknowledged: true,

insertedIds: { '0': ObjectId("6438024427b1f16d729ada77") }

}

6 – Search for your object by name.

db.Students.find({name:"Ahmed"})

[

{

\_id: ObjectId("6438001627b1f16d729ada76"),

name: 'Ahmed',

age: 23,

courses: [ 'javascript', 'mongodb', 'C#' ]

}

]

7– Search for your friend(s) by age.

db.Students.find({age:23})

{

\_id: ObjectId("6438024427b1f16d729ada77"),

name: 'Sohila',

age: 23,

courses: [ 'javascript', 'mongodb', 'C#' ]

}

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

db.Students.find({age:{$gt:23}})

**Output:**

[

{

\_id: ObjectId("6438067b27b1f16d729ada78"),

name: 'Shahenda',

age: 25,

courses: [ 'javascript', 'mongodb', 'C#' ]

},

{

\_id: ObjectId("6438077b27b1f16d729ada7a"),

name: 'Mohamed',

age: 30,

courses: [ 'javascript', 'mongodb', 'c#' ]

},

{

\_id: ObjectId("643807c827b1f16d729ada7c"),

name: 'Mostafa',

age: 30,

courses: [ 'javascript', 'mongodb', 'c#' ]

}

]

9 – delete any of your friends by id.

db.Students.deleteOne({\_id:ObjectId("6437fa8927b1f16d729ada74")})

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

db.Students.find().pretty()

11 – count all documents in students’ collection.

var cursor =db.Students.find()

cursor.count()

Output: 8

---------------------------------------------------------

part 2

1. Create database with name ems

use ems

2- Insert the following data into "faculty" collection

{ "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"}

let facultyarray=[{ "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"}]

db.faculty.insert(facultyarray)

1. Get the details of all the faculty.

db.faculty.find({})

1. Get the count of all faculty members.

db.faculty.find({}).count()

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

db.Faculty.find({qualification:"Ph.D"})

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

db.Faculty.find( { exp: { $gt: 8, $lt: 12 } } )

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

db.Faculty.find( { subjects: { $in: [ "MATHS", "NETWORKING" ]} } )

1. 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( { $and:[{subjects: "MATHS" },{age:{$gt:30}},{qualification:"Ph.D" }]})

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

db.faculty.find({$or:[{subjects: "JAVA"},{type: "Full Time"}]})

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.updateOne({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: {$in:["MATHS"]} },{$set:{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,qualification:1})

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,qualification:1,exp:1}).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)