# MongoDb Software Installation

## Reference URL:

<https://www.youtube.com/watch?v=YYnf5vJRuUY&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal>

* <https://www.mongodb.com/download-center/community>
* After installing navigate to following path C:\Program Files\MongoDB\Server\4.2\bin
* And open cmd and run mongod.exe to ensure server is running properly
* By default you will get an error as shown in below
* <https://stackoverflow.com/questions/41420466/mongodb-shuts-down-with-code-100>
* And follow above link by creating a folder named C:\data\db where mongo uses a space in storage
* After creating the above folder again run the command mongod.exe in cmd this time it runs properly
* Open another terminal command in same path and enter mongo in cmd you will see server information connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
* Enter db then it will display test

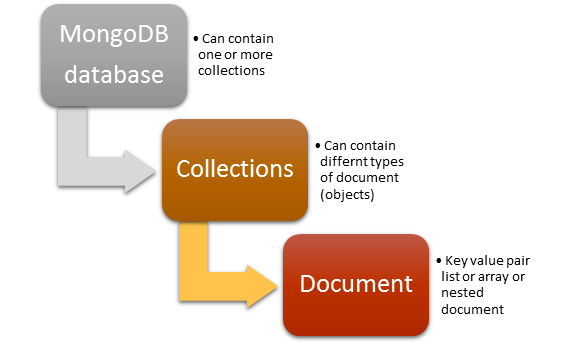
# Path and MongoChef

## ReferenceURL:

<https://www.youtube.com/watch?v=4YZPfX5sKKI&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=2>

* We have to specify path in environment variables C:\Program Files\MongoDB\Server\4.2\bin
* Download <https://studio3t.com/download-thank-you/?OS=win64>

# Structure of MongoDb



# 

# Create/Drop Database

## Reference URL :

<https://www.youtube.com/watch?v=mENTYOgtifc&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=3>

* After installing studio 3t then we can open it and we can get a new connection by providing a connection name
* My connection name is mongodbplay with port 27017
* By default it comes up with 3 Databases admin,config,local select local db which have some collections in it
* Now navigate to IntelliShell by selecting database
* **use testdb**  use will creates a db if not available
* To check current database enter **db**
* **show dbs** used to list all databases
* In mongo to see a database we need to insert a document
* Command to insert a document in mongo
* **db.userCollection.insert({"name":"ratna","age":27,"worklocation":"hyd"})** db refers current db and insert a document into collection here userCollection is collection
* **db.dropDatabase()** will delete current database

# Create/Drop Collections

## Reference URL

<https://www.youtube.com/watch?v=k4UopPslnxE&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=4>

* We can create a collection in two approaches

1. db.createCollection(“Test”)
2. db.testCollection.insert({"test":"testCollection"}) if testCollection doesn’t exist it creates and inserts into the testcollection.

* Check available Collections ->show collections
* To Drop a collection->db.testCollection.drop();

# Insert Documents into Database

## Reference URL

<https://www.youtube.com/watch?v=Ik4lVkapEpg&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=5>

We can insert into a collection either single way or list of documents

Single way

db.employees.insert({

"EmpNo":"1",

"FirstName":"Andrew",

"LastName":"Neil",

"Age":"30",

"Gender":"Male",

"Skill":"MongoDB",

"Phone":"408-1234567",

"Email":"Andrew.Neil@gmail.com",

"Salary":"80000"

});

Multiple records can be inserted in the form of array db.employees.insert([{},{}])

db.employees.insert([{

"EmpNo":"2",

"FirstName":"Brian",

"LastName":"Hall",

"Age":"27",

"Gender":"Male",

"Skill":"Javascript",

"Phone":"408-1298367",

"Email":"Brian.Hall@gmail.com",

"Salary":"60000"

},

{

"EmpNo":"3",

"FirstName":"Chris",

"LastName":"White",

"Age":"40",

"Gender":"Male",

"Skill":"Python",

"Phone":"408-4444567",

"Email":"Chris.White@gmail.com",

"Salary":"100000"

},

{

"EmpNo":"4",

"FirstName":"Debbie",

"LastName":"Long",

"Age":"32",

"Gender":"Female",

"Skill":"Project Management",

"Phone":"408-1299963",

"Email":"Debbie.Long@gmail.com",

"Salary":"105000"

},

{

"EmpNo":"5",

"FirstName":"Ethan",

"LastName":"Murphy",

"Age":"45",

"Gender":"Male",

"Skill":"C#",

"Phone":"408-3314567",

"Email":"Ethan.Murphy@gmail.com",

"Salary":"120000"

},

{

"EmpNo":"6",

"FirstName":"Felicia",

"LastName":"Lee",

"Age":"33",

"Gender":"Female",

"Skill":"MongoDB",

"Phone":"408-8832567",

"Email":"Felicia.Lee@gmail.com",

"Salary":"85000"

},

{

"EmpNo":"7",

"FirstName":"George",

"LastName":"Cyrus",

"Age":"36",

"Gender":"Male",

"Skill":"MongoDB",

"Phone":"408-9984567",

"Email":"George.Cyrus@gmail.com",

"Salary":"88000"

},

{

"EmpNo":"8",

"FirstName":"Hannah",

"LastName":"Johnson",

"Age":"26",

"Gender":"Female",

"Skill":"AngularJS",

"Phone":"408-7654321",

"Email":"Hannah.Johnson@gmail.com",

"Salary":"72000"

}

])

# Querying MongoDB

## Reference URL

<https://www.youtube.com/watch?v=w2BoKwUB75I&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=6>

* Command to retrieve all the documents

db.employees.find()

* Here employees is the collection
* To get the data in structured way use db.employees.find().pretty()
* To retrieve the first document of the collection use following command

db.employees.findOne()

* Retrieve documents on equality condition

db.employees.find({"EmpNo":"2"})

* Retrieve documents whose age less than 30

db.employees.find({"Age":{$lt:"30"}})

* Retrieve documents whose age less than or equal to 30

db.employees.find({"Age":{$lte:"30"}})

* Retrieve documents whose age greater than 30

db.employees.find({"Age":{$gt:"30"}})

* Retrieve documents whose age greater than or equal to 30

db.employees.find({"Age":{$gte:"30"}})

* Retrieve documents whose age not equal to 30

db.employees.find({"Age":{$ne:"30"}})

# 

# And/Or Conditions MongoDB

## Reference URL

<https://www.youtube.com/watch?v=tkVoqzCbUP4&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=7>

* To retrieve employees with skill MongoDB

db.employees.find({"Skill":"MongoDB"}).pretty()

* To retrieve employees with skill MongoDB and Salary is 80000

db.employees.find({"Skill":"MongoDB","Salary":"80000"}).pretty()

* To retrieve employees with skill MongoDB or Salary is 80000

db.employees.find({$or:[{"Skill":"MongoDB"},{"Salary":"80000"}]}).pretty()

* Combining And Or conditions

db.employees.find({"Skill":"MongoDB",$or:[{"Salary":"80000"},{"Salary":"85000"}]}).pretty()

# Update Documents

## Reference URL

<https://www.youtube.com/watch?v=Xl-Q4HjmDoo&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=8>

* Update Salary from 80000 to 90000 of Andrew Neil

db.employees.update(

{"\_id" : ObjectId("5ec8e24106a96146dc5bd7ad")},

{$set:{"Salary":"90000"}}

)

* Update Salary of all employees having skill in MongoDB
* By Default mongoDb updates only single document to make it update all the documents we have to specify a parameter multi=true

db.employees.update(

{"Skill" : "MongoDB"},

{$set:{"Salary":"150000"}},

{multi:true}

)

# Remove Documents

## Reference URL

<https://www.youtube.com/watch?v=vJf7xZ2PSkY&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=9>

* Deleting any employee with EmpNo 2

db.employees.remove({ "\_id" : ObjectId("5ec8e2d106a96146dc5bd7ae")})

* Deleting employees with skill mongodb but applying selection criteria

db.employees.remove({ "Skill":"MongoDB"},1)

* To remove all employees with skill mongodb remove selection criteria

db.employees.remove({ "Skill":"MongoDB"})

# Selecting Fields

## Reference URL

<https://www.youtube.com/watch?v=FLv40w6u9ps&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=10>

* In MongoDB for selecting the command we will use find only but it will take two parameters the first one is for conditions and second is for select

db.employees.find({},{"FirstName":1}).pretty()

db.employees.find({},{"FirstName":1,"LastName":1}).pretty()

* By Default id field will be displayed in MongoDB to hide it use the following

db.employees.find({},{"FirstName":1,"LastName":1,"\_id":0}).pretty()

# Limit, Skip and Sort

## Reference URL

<https://www.youtube.com/watch?v=sEbJCGo02RY&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=11>

* Limit the documents to 5

db.employees.find({},{"FirstName":1,"LastName":1,"\_id":0}).pretty().limit(5)

* Skip first 3 documents and display remaining

db.employees.find({},{"FirstName":1,"EmpNo":1,"\_id":0}).pretty().skip(3)

* Skip first 3 documents and limit to 3 documents

db.employees.find({},{"FirstName":1,"EmpNo":1,"\_id":0}).pretty().skip(3).limit(3)

* Sort employees
* Ascending order use 1
* Descending order use -1

db.employees.find({},{"FirstName":1,"EmpNo":1,"\_id":0}).pretty().sort({"FirstName":-1})

# Indexing

## Reference URL

<https://www.youtube.com/watch?v=JJyOFoMHkBo&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=12>

* Creating an Index on email

db.employees.ensureIndex({"Email":1})

{

"createdCollectionAutomatically" : false,

"numIndexesBefore" : 1.0,

"numIndexesAfter" : 2.0,

"ok" : 1.0

}

* To check what are the indexes available already

db.employees.getIndexes()

* To drop an Index

db.employees.dropIndex({"Email":1})

* Retrieving of data will be fast with indexing

# Aggregation

## Reference URL

<https://www.youtube.com/watch?v=zcN-rM3hgrw&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=13>

* Getting the total genders

db.employees.aggregate([{$group: { \_id: "$Gender", Total: { $sum: 1} } }])

* To get the maximum and minimum age of gender

db.employees.aggregate([{$group: { \_id: "$Gender", MaxAge: { $max: "$Age"} } }])

* To get max salary based on skill

db.employees.aggregate([{$group: { \_id: "$Skill", MaxSal: { $max: "$Salary"} } }])

* To get min salary based on skill

db.employees.aggregate([{$group: { \_id: "$Skill", MinSal: { $min: "$Salary"} } }])

* To get avg salary based on skill

db.employees.aggregate([{$group: { \_id: "$Skill", AvgSal: { $avg: "$Salary"} } }])

# Backup & Restore

## Reference URL

<https://www.youtube.com/watch?v=CHNB38MAvKY&list=PLC3y8-rFHvwh11bWtwm3_qKvo46uDmaal&index=14>

## BackUp The Database

* Open cmd as administrator
* Navigate to the bin directory cd C:\Program Files\MongoDB\Server\4.2\bin
* Enter following command

mongodump

* It will create the dump folders of the databases
* C:\Program Files\MongoDB\Server\4.2\bin\dump
* To take the backup of only required databases
* mongodump --db company

## Restore The Database

* Delete the databases current one which is company

db.dropDatabase()

* show databases

You will see the company database is dropped

* Go back to admin window cd C:\Program Files\MongoDB\Server\4.2\bin

Enter the command

mongorestore

* We got our database restore
* To restore single database

mongorestore --db company dump/company

Mongo restore checks the dump and restores if there is no database available

## Backup the collection

mongodump --db company --collection employees

## Restore the Collection

mongorestore --db company --collection employees