**MONGO SHELL SHORTCUTS**

**cls**

Clear the screen

**mongodb --port 27018**

Explicitly giving a port number

**mongo --port 27018**

Connect to that specific port

**show dbs**

shows all the db databases you have

**use <dbName>**

Creates a new database or enters and existing database

**show collections**

Shows the collections of that database

**db.dropDatabase()**

Remove existing database

**Data Types**

**String:** “some text”

**Boolean:** true or false

**Number(integer, NumberLong, NumberDecimal):**

33, 123456789123456789, 33.34

**ObjectId:** Automatically created

**ISODate:** new Date()

**TimeStamp:** new Timestamp()

**Embedded** Document: {}

**List:** [ ]

**MONGO CRUD OPERATIONS**

**C**reate

insertOne(data, options)

insertMany([data], options)

**R**ead

find(filter, options)

findOne(filter, options)

**U**pdate

updateOne(filter, data, options)

updateMany(filter, data, options)

replaceOne(filter, data, options)

**D**elete

deleteOne(filter, options)

deleteMany(filtcursor methods mnogodb

er, options)

Crud Docs

<https://docs.mongodb.com/manual/crud/>

Operators

<https://docs.mongodb.com/manual/reference/operator/>

**Examples:**

**Create**

db.teachers.insertOne({name:"Pinchas",subjects:[],students:[],capacityLeft:30})

db.teachers.insertMany([{name:"George",subjects:[],students:[],capacityLeft:10},{name:"Harry",subjects:[],students:[],capacityLeft:30}])

**Read**

db.teachers.find().pretty()

db.teachers.findOne({\_id:ObjectId("5ff6fecaf91717875a68fc06")}).pretty()

db.teachers.find({name:”Pinchas})

db.teachers.find( { capacityLeft:{$gt:11}}).pretty()

db.teachers.find( { capacityLeft:{$lt:11}}).pretty()

**Cursor Methods**

db.users.find().toArray()

db.users.find().forEach(user => printjson(user))

db.users.find().forEach(user => printjson(`users name is ${user}`))

db.users.find().count()

db.users.find().limit(2)

**Update**

db.teachers.updateOne({\_id:ObjectId("5ff6ffa6f91717875a68fc09")},{$set:{name:"Jennifer"}})

db.teachers.updateOne({\_id:ObjectId("5ff6ffa6f91717875a68fc09")},{$set:{age:22}})

db.teachers.updateMany({age:22},{$set:{age:23}})

db.teachers.updateMany({age:{$gt:20}},{$set:{young:true}})

db.teachers.replaceOne({\_id:ObjectId("5ff6ffa6f91717875a68fc08")},{name:"harry"})

**Delete**

db.teachers.deleteOne({\_id:ObjectId("5ff6ffa6f91717875a68fc09")})

db.teachers.deleteMany({name:{$exists:true}})

**Dealing with Arrays**

**pushing to an array:**

db.teachers.updateOne({\_id:ObjectId("5ff70df4f91717875a68fc0b")},{$push:{subjects:"history"}})

**removing from an array:**

db.teachers.updateOne({\_id:ObjectId("5ff70df4f91717875a68fc0b")},{$pull:{subjects:"history"}})

db.teachers.updateOne({\_id:ObjectId("5ff70df4f91717875a68fc0b")},{$pop:{subjects:1}})

**get the values from an array:**

db.teachers.findOne({\_id:ObjectId("5ff70df4f91717875a68fc0b")}).subjects