https://www.mongodb.com/nosql-explained

cap theorem

https://www.ibm.com/in-en/topics/cap-theorem

query

https://www.mongodb.com/developer/products/mongodb/

https://www.jdoodle.com/online-mongodb-terminal/

```
//to create db

use sales

// to create collection and storing a document/data

db.product.insertOne({"name":"pen",cost:20})

//to retrieve data from collection

db.product.findOne()

//nesting docs

db.product.insertOne({"name":"mobile",cost:12000,details:
{brand:"samsung",color:"red"}})

db.product.insertOne({"name":"mobile",cost:12000,details:
{brand:"samsung",color:"green"}})

//prints only one data which is by default first document

db.product.findOne()

//to print all docs available in a collection

db.product.find()

//insert many docs(array of json docs)

db.product.insertMany([{"name":"pencil",cost:10},{"name":"candy",cost:50}])

//CRUD operations, create read update delete

//read already we discussed

read can be done by findone or find

//filter in find method
```

filtering and retrieving one doc/data with a condition satisfied

db.product.findOne({name:"pen"})

// find method with filtering and retrievalmany docs(filtering nested document)

db.product.find({"details.brand":"samsung"})

//updating data

db.product.updateOne({name:"pen"},{$set: {"cost":35}})

use find method to know status of update

//update many also possible

//delete

db.product.deleteOne({name:"pen"})

use find method to know status of delete

// delete many

db.product.deleteMany

//creating collection

db.createCollection('name',{options})

options:size of collection, max

//projection (like we discussed already -> filtering and from the result we project certain attributes)

to retrieve name from all docs

{} first json document is empty our query document or filtering document so empty

{} second json document is projection related

db.product.find({},{name:1})

db.product.find({},{name:1,_id:0})