

#mongoose

We use mongoose embeded with node.js & with it we can use mongoDB with ease

mongoose help us to make us ease to connect with mongoDB database with a small code itself
(test is database)

Code

```
const mongoose = require('mongoose');
mongoose.connect('mongodb://localhost/test');
// test is get created if not exists
```

```
const cat = mongoose.model('cat', {name:string})
```

```
const kitty = new cat({name: "manu"});
```

```
kitty.save().then(() => console.log('meow!'));
```

for deleting dB in mongo

use dbName

db.dropDatabase() it will delete that database

after connecting to specified link then came new challenge & that is create a schema for your database

so far that

```
const fruitschema = new mongoose.Schema({[JSObject]})
```

& such how you can create a new schema

the object id
watermelon

Ex:- cat schema

attribute of cat , name so create by
colour = "Single" / "multiple"

age =

type =

```
const catSchema = new mongoose.Schema({})
```

name : string,

age : number,

or colour : string ,

type : string ,

so here we created the schema .

so lets create a attribute for database
cat on top of catSchema

```
const cat = mongoose.model("cat", catSchema)
```

model will define which data we
are going to create & for that which schema
we are going to implicate

```
const cat1 = new cat({})
```

= model declared a 1

name : "Mani",

colour : "white-orange",

age : 2,

type : "normal",

order
not man
this is

after that

`(cat1.save())`

& it will save that object in cat collection

It works like insert function

How to insert in bulk

You have to create instant related to every structure and at end

`const cat2 = new cat({});`

`const cat3`

`const catn =`

so our model name is `cat`

`cat.insertMany([cat1, cat2, cat3, ..., catn],
function(err) {`

`console.log(err err);`

`}`

as in `send` app `sent` in express we get response

same here in function get errors as

acknowledgment

) In shell we use same like

`db.cat.insertMany([array of obj])`

function invoked
on error

object id
watermelon prop

If your input breach this we will get a validation error

Same can be applied for the database like having type string

#If you want to make a field mandatory

name: {

 type: string

 required: true

}

If you don't give this field it will throw an error

Object
watermelon

Update & delete data Mongoose
 model.updateOne()
 model.updateMany()

Parameters \Rightarrow (query, what to update, callback)
 Id or something else

```
model.updateOne({_id: "abc"}, {name: "xyz"},  

  function(error){  

    if (err) { console.log(error); }  

    else { console.log("success"); }  

  })
```

\Rightarrow Order of properties inside a object will does not matter at all

Establish Relationships

so we can able to insert one schema into another schema

Say you already created student schema
 say your going to use persons
 that in colled schema

```
const person = mongoose.Schema({
```

name: '\$tn'

xyz: studentSchema

}

Thus how you now able to make connection / Relationship betn college schema & relationship schema

④ You can also add variables of another model in declaration of variables of another model then that can be helpful for you

model fruit

```
④ const watermelon = new fruit {  
    name: "watermelon",  
    taste: "sweet",  
    price: .50,  
}
```

```
const ghansham = new person {
```

```
name: "ghansham",  
FavoriteFruit: "Watermelons",  
}
```

the object id of watermelon variable & watermelon properties will remain same.

```
f) watermelon.name = "orange" (200)  
watermelon.taste = "sweet"  
watermelon.price = .50
```