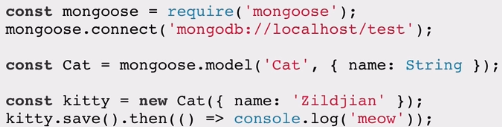
ODM (Object Document Mapper)

* It will allow your Node.js app which speaks the language of JavaScript objects to be able to talk to your MongoDB database which speaks in the language of documents and collections and databases.
* Our ODM is Mongoose
  + <https://mongoosejs.com/docs/api/model.html#model_Model-insertMany>



* line 1: we require the mongoose package
* line 2: we connect to the database or create one if it doesn’t exist
* line 3: we create a new model of how our data will be represented
* line 4: with the new model we create a new document
* line 5: we save our model to the mongodb database

Starting the database connection and Executing database commands

- type mongod on the hyperterminal

- then open a new hyper tab and type mongo

-then you can interact with your database on the tab where you wrote mongo

How to use mogoose

Step 1: Create a directory with app.js file and type npm -y init in terminal

Step 2: - type  in app.js

Step 3: -type ‘npm install mongoose’ in terminal

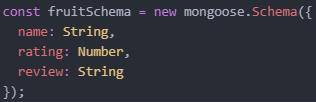
-if mogodb isn’t installed, install it also

Step 4: - add the below in app.js to connect to your database



- it's going to make a connection to our MongoDB server and then it's going to look for a database called fruitsDB, and if it doesn't exist, then it will create this brand new database.

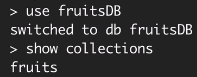
Step 5: - Create a new schema. This is basically a blueprint or the structure of our data that we're going to save into our MongoDB database. To create the schema type the below code in app.js:-



Step 6:-write the below code in app.js



* it specifies the collection name. Mongo will automatically drop the capital letter in “Fruit” and convert the word into plural form. So, our collection name will be “fruits”



Step 7:- Write the below code in app.js to add an element to your fruits collection



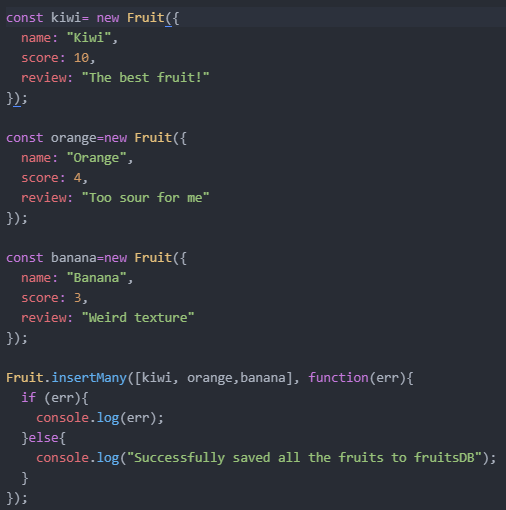
- fruit.save calls the save method in Mongoose to save this fruit document into a fruit collection inside our fruitsDB. And that replaces our insertDocuments method from the Mongo client.

Step 8:- type node app.js in Hyperterminal to save the fruit and add it to the database. If you touch node app.js again without commenting out .save. There will be duplicates.

Step 9: Write the code below in node.js in order to add in bulk

-Additional information:https://mongoosejs.com/docs/api/model.html

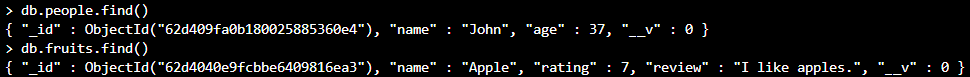
-see code in next page



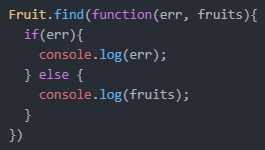
Sample code for connecting to mongoose with corresponding result







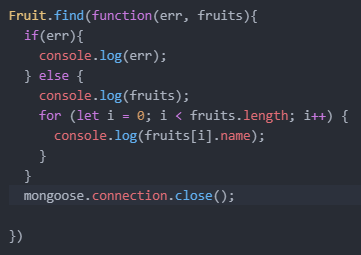
Looking through your collection(table):- Model.find()

- 

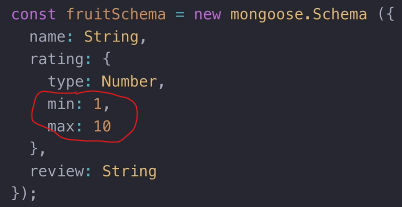
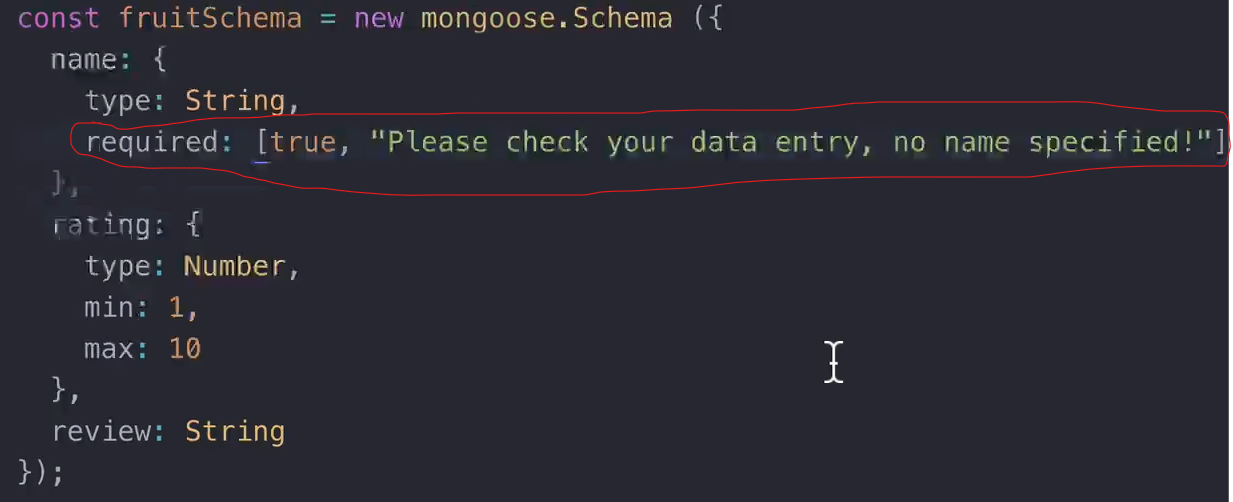
-We tap into our fruits collection through the Fruit model, we call the find function on it and then when that's completed a callback gets triggered and we have the possibility of having an error if anything went wrong with finding our fruits but otherwise we would get some results back which we've called fruit.

-We should comment out the ‘insertMany’ or ‘.save’ Part of our code to avoid repeated insertion.

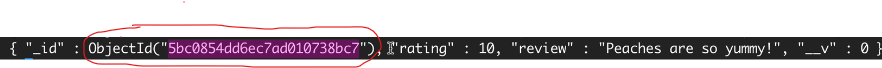
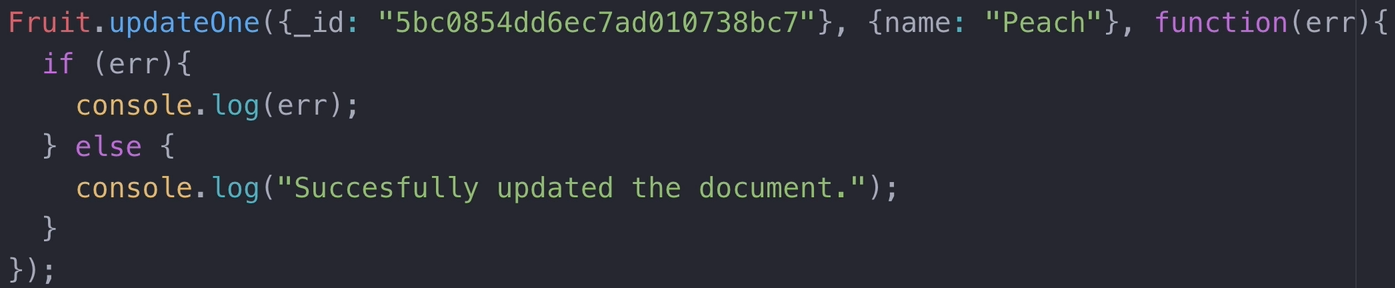
Closing a mongoose connection

* 
  + Don’t forget to add this at then end of your code, it is good practice
  + Add it within the Model.find() command. If you add it outside, there will be an error
    - 

Specifying Data validators

* we do it when we specify our model. Specifying a format
* specifying a value range
  + 
    - if we add a fruit that has a rating above 10, we will encounter an error
* making a file required and specifying what the error message will be
  + 

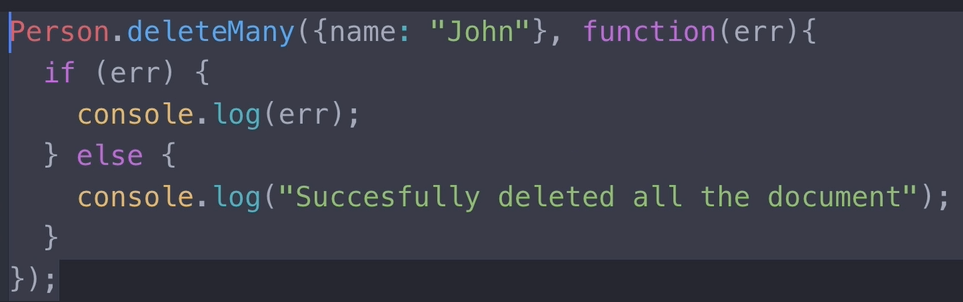
Updating a value

* 
  + Every item added to the database has an id
* Model.updateOne(condition, what\_we\_want to update about it, callback function)
  + 

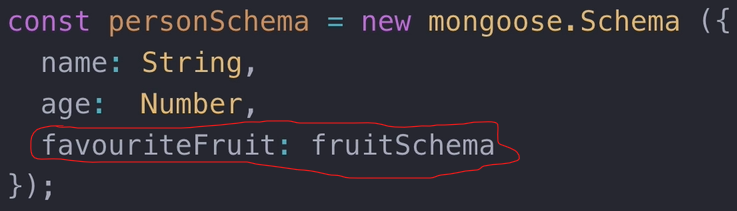
Deleting a single entry

* https://mongoosejs.com/docs/api/query.html#query\_Query-deleteOne
* 
* 

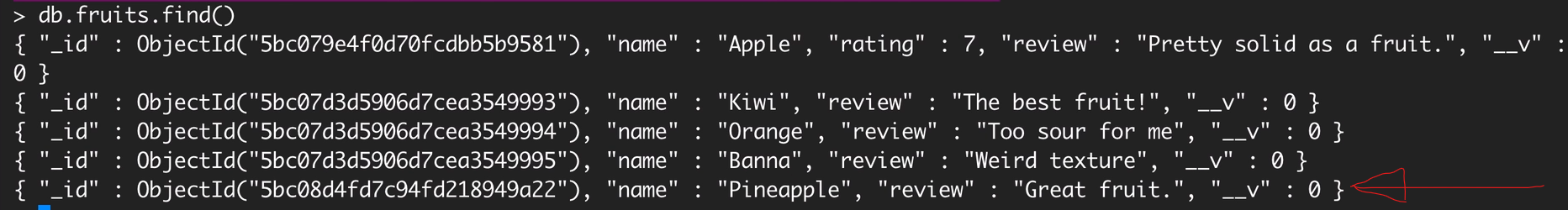
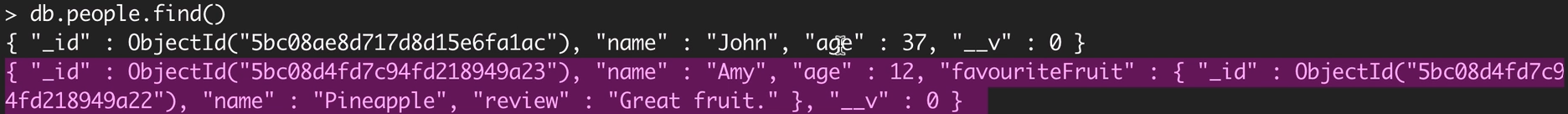
Deleting many entries

* 

Embedding another collection inside the other

* 



* + the fruit will be saved both in the fruit collection and inside the person Amy. It will have the same id in both positions
    - 
    - 

\*\*Don’t forget to comment out collection.save() after the first save

\*\*Go to the below site for queries:

-https://mongoosejs.com/docs/api/query.html