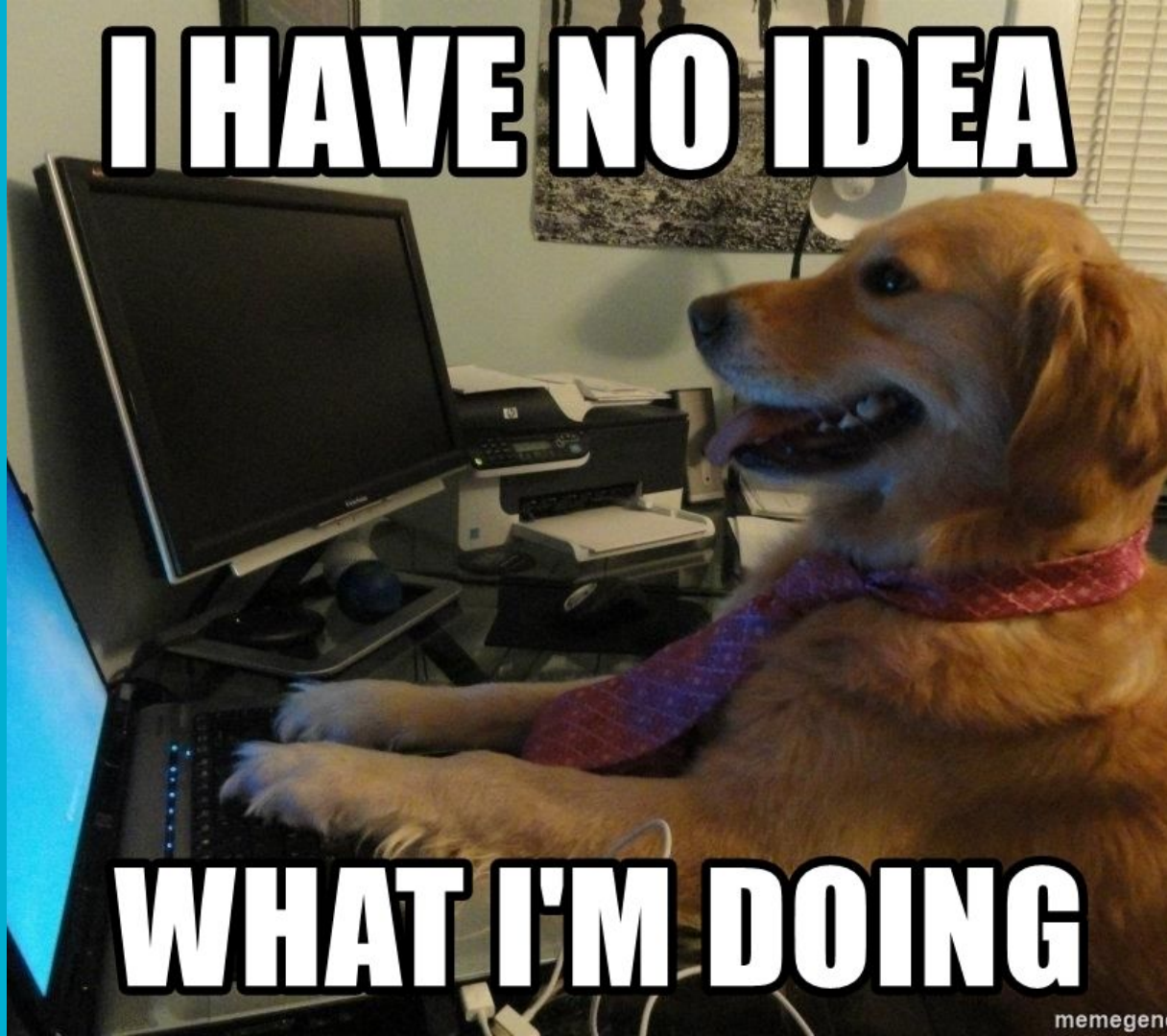
Mongoose

What is Mongoose?



Mongoose

- Mongoose is a popular npm package.
- Known as the ODM (Object Document Mapper. A code library that converts the transfer of data stored in database tables into objects.

Why is mongoose important?

- Mongoose gives us the power to organize our database by using schemas.
- Allows you to define objects with a schema that is mapped to a MongoDB document.
- Once a schema is defined, then you can create a model based on a specific schema.
- Once you have defined your schemas and models, Mongoose contains several functions that allow you to validate, save, delete, and query your data using similar MongoDb functions.

```
const mongoose =require("mongoose");
const mySchema =mongoose.mySchema;

const musicSchema = new mySchema({
    // 'string' is the shorthand version of {type: string}
    title: string,
    artist: string,
    genre: string,
    commecnts: [{ body: String, date: Date}]
});

// In order to use the schema, we'll convert the schema into a model that can be worked with

const music = mongoose.model("music", musicSchema);
module.exports = music;
```

MongoDB aka No-SQL

- MongoDb is a database that stores your data as documents. (JSON Structure).
- The documents are then saved to collections.
- Scalable and consistent
- Uses the Mongo Shell
- MongoDB has a flexible data model, making it easier to change data within your application

```
const dbName = 'characters';
const db = client.db(dbName);

async function main() {
  // Use connect method to connect to the server
  await client.connect();
  console.log('Connected successfully to server');




  db.createCollection("looneytunes");
  db.createCollection("starwars");
  db.createCollection("simpsons");
  db.createCollection("looneytunez");


  const looneytunes = db.collection('looneytunes');
  const starwars = db.collection("starwars");
  const simpsons = db.collection("simpsons")


  try {
   await  starwars.insertMany([
        {"name":"Luke Skywalker","role":"Jedi","best_character":false,"first_appearance":"Star Wars:A New Hope"},
        {"name":"Darth Vader","role":"Sith Lord","best_character":false,"first_appearance":"Star Wars:A New Hope"},
        {"name":"Jean Luke Picard","role":"Captain","best_character":false,"first_appearance":""},
        {"name":"Han Solo","role":"Smuggler","best_character":false, "first_appearance":"Star Wars:A New Hope"},
        {"name":"C-3PO","role":"","best_character":true,"first_appearance":"Star Wars:A New Hope"},
        {"name":"Spock","role":"Lieutenant","best_character":false,"first_appearance":""},
        {"name":"Yoda","role":"Jedi Master","best_character":false,"first_appearance":"Star Wars: The Empire Strikes Back"},
        {"name":"Boba Fett","role":"Bounty Hunter","best_character":false,"first_appearance":"The Star Wars Holiday Special"},
        {"name":"Luke Skywalker","role":"Jedi","best_character":false}])
```

Mongoose Terminology

- Collections: Tables in relational databases that hold multiple JSON documents.
- Fields: Columns.
- Schema: defines the structure of a type of data or document (shape of the document)/(Array, Boolean, Number, String, etc.)
  - A way to express expected properties and values as well as constraints.
  - Schemas can be reused
  - Can contain several child-schemas
- Models: Defines the programming interface for interacting with the database (create, fetch, update, delete, ect.)

How do we create Schemas and models?

- First we…. 'npm install mongoose' since mongoose is a package.
  - The our server.js file we'll want to require mongoose
  - **Const mongoose = require('mongoose');**
- Then we connect our server.js file to a models.js file
- In our model.js file we will create our schemas and models

- We use 'npm server.js' to view everything in the terminal