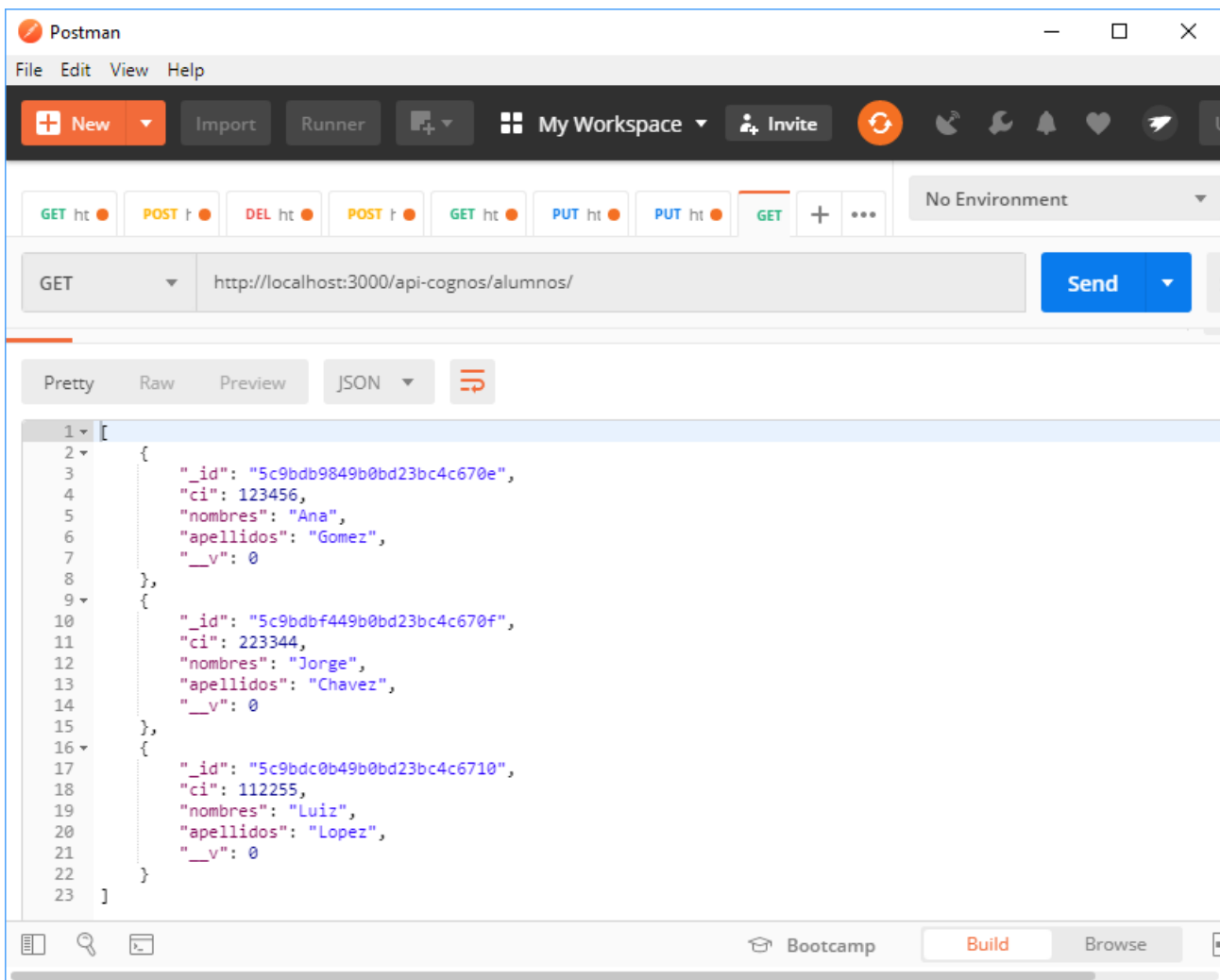# /api-cognos/alumnos

#npm i express mongoose

```
C:\WINDOWS\system32\cmd.exe                                    —    □    ✕

D:\marcelomos\laboratorios>mkdir api-express-mongo-node

D:\marcelomos\laboratorios>cd api-express-mongo-node

D:\marcelomos\laboratorios\api-express-mongo-node>npm init --yes
Wrote to D:\marcelomos\laboratorios\api-express-mongo-node\package.json:

{
  "name": "api-express-mongo-node",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
```
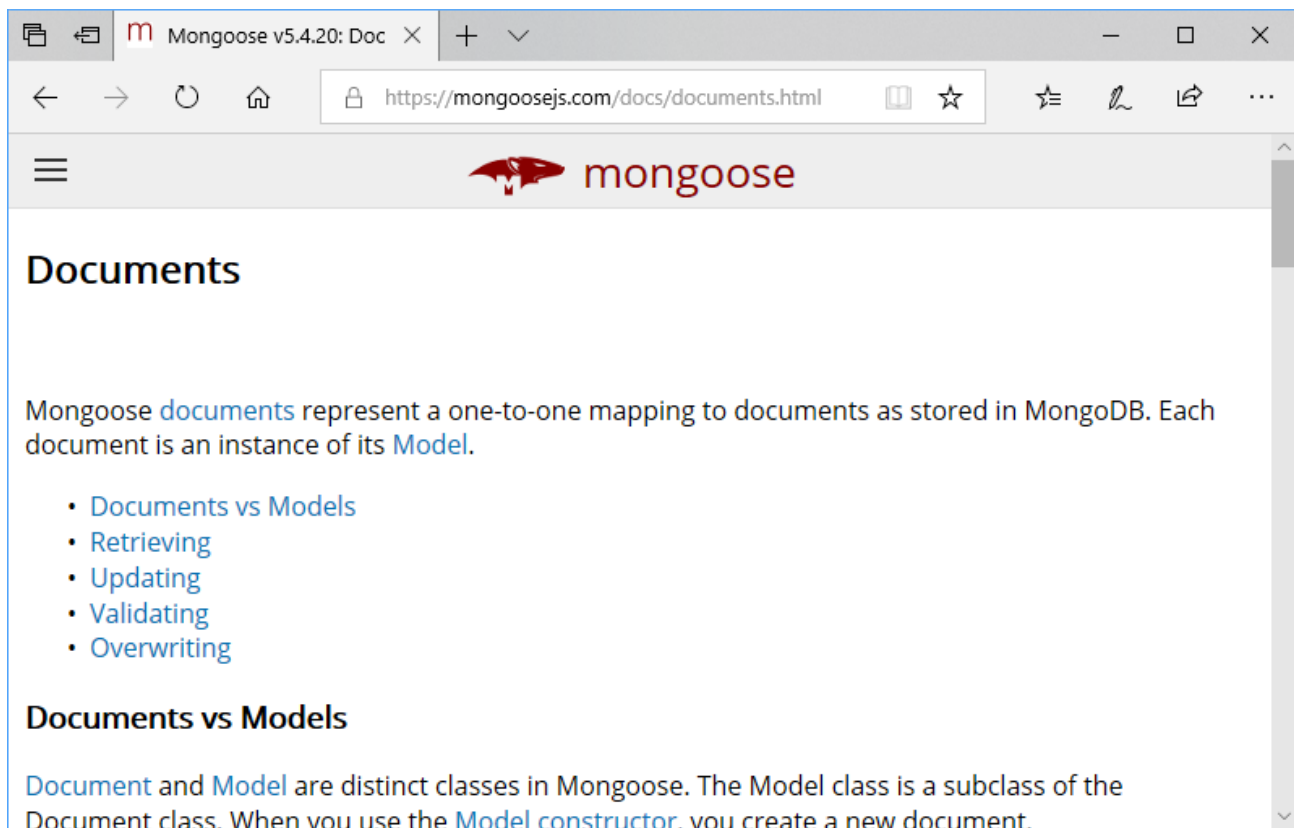
```
C:\WINDOWS\system32\cmd.exe                                    —    □    ✕

D:\marcelomos\laboratorios\api-express-mongo-node>npm i express mongoose
npm notice created a lockfile as package-lock.json. You should commit this
file.
npm WARN api-express-mongo-node@1.0.0 No description
npm WARN api-express-mongo-node@1.0.0 No repository field.

+ express@4.16.4
+ mongoose@5.4.20
added 69 packages from 48 contributors and audited 159 packages in 15.483s
found 0 vulnerabilities


D:\marcelomos\laboratorios\api-express-mongo-node>npm i cors morgan
npm WARN api-express-mongo-node@1.0.0 No description
npm WARN api-express-mongo-node@1.0.0 No repository field.

+ morgan@1.9.1
```
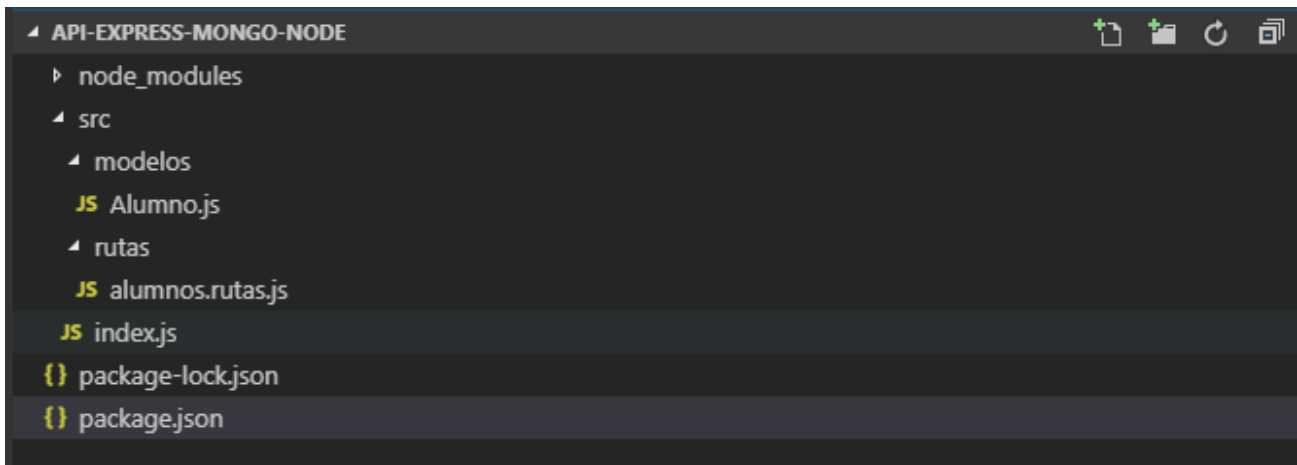
Documents

Mongoose documents represent a one-to-one mapping to documents as stored in MongoDB. Each document is an instance of its Model.

- Documents vs Models
- Retrieving
- Updating
- Validating
- Overwriting

Documents vs Models

Document and Model are distinct classes in Mongoose. The Model class is a subclass of the Document class. When you use the Model constructor, you create a new document.

/src/package.json

```json
{
  "name": "api-express-mongo-node",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "node src/."
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "cors": "^2.8.5",
    "express": "^4.16.4",
    "mongoose": "^5.4.20",
    "morgan": "^1.9.1"
  }
}
```

```
▲ API-EXPRESS-MONGO-NODE
  ▷ node_modules
  ▲ src
    ▲ modelos
    JS Alumno.js
    ▲ rutas
    JS alumnos.rutas.js
  JS index.js
  {} package-lock.json
  {} package.json
```

/src/index.js

```javascript
const express = require('express');
const mongoose = require('mongoose');
const path = require('path');
const cors = require('cors');
const morgan = require('morgan');

// initializations
const app = express();
mongoose.connect('mongodb://localhost/cognosdb');

// settings
app.set('port', process.env.PORT || 3000);

// middlewares
app.use(express.urlencoded({extended: false}));
app.use(cors());
app.use(morgan('dev'));
app.use(express.json());

// routes
app.use('/api-cognos/alumnos',require('./rutas/alumnos.rutas'));

// starting the server
app.listen(app.get('port'), () => {
    console.log(`server on port ${app.get('port')}`);
});
```

/src/rutas/alumnos.rutas.js

```javascript
const express = require('express');
const router = express.Router();

const Alumnos = require('../modelos/Alumno');

router.get('/', async (req, res) => {
  const alumnos = await Alumnos.find({});
  res.json(alumnos);
});

router.get('/:ci', async (req, res) => {
    const ci = req.params.ci;
    const alumno = await Alumnos.find({ci});
    res.json(alumno);
});

router.post('/', async (req, res) => {
  const {ci,nombres,apellidos} = req.body;
  const alumno = new Alumnos({ci,nombres,apellidos});
  await alumno.save();
  res.json({mensaje: 'Alumno registrado'});
});

router.put('/:ci', async (req, res) => {
  const ci = req.params.ci;
  const {nombres,apellidos} = req.body;
  await Alumnos.findOneAndUpdate({ci},{nombres,apellidos});
  res.json({mensaje: 'Alumno actualizado'});
});

router.delete('/:ci', async (req, res) => {
  const ci = req.params.ci;
  await Alumnos.findOneAndRemove({ci});
  res.json({mensaje: 'Alumno eliminado'});
});

module.exports = router;
```

/src/modelos/Alumno.js

```javascript
const mongoose = require('mongoose');
const { Schema } = mongoose;

const Alumno = new Schema({
  ci: Number,
  nombres: String,
  apellidos: String
});

module.exports = mongoose.model('Alumnos', Alumno);
```

```
C:\Program Files\MongoDB\Server\4.0\bin\mongo.exe                            —  □  ✕

> db.alumnos.find().pretty()
{
        "_id" : ObjectId("5c9bdb9849b0bd23bc4c670e"),
        "ci" : 123456,
        "nombres" : "Ana",
        "apellidos" : "Gomez",
        "__v" : 0
}
{
        "_id" : ObjectId("5c9bdbf449b0bd23bc4c670f"),
        "ci" : 223344,
        "nombres" : "Jorge",
        "apellidos" : "Chavez",
        "__v" : 0
}
{
        "_id" : ObjectId("5c9bdc0b49b0bd23bc4c6710"),
        "ci" : 112255,
        "nombres" : "Luisa",
        "apellidos" : "Lopez",
        "__v" : 0
}
{
        "_id" : ObjectId("5c9bea248290eb28d8a6aee5"),
        "ci" : 224466,
        "nombres" : "Carlos",
        "apellidos" : "Mercado",
        "__v" : 0
}
> db.alumnos.find({nombres:/Car/})
{ "_id" : ObjectId("5c9bea248290eb28d8a6aee5"), "ci" : 224466, "nombres" : "Carlos", "apellidos"
 : "Mercado", "__v" : 0 }
>
```