|  |
| --- |
| Server.js  const express = require('express');  const mongoose = require('mongoose');  const cors = require('cors');  const app = express();  app.use(cors());  app.use(express.json());  const port = 8000;  const dishStaticRouter = require("./routes/dishes.router.js");  const dishDBRouter = require("./routes/dishesDB.router.js");  var mongoURL = "mongodb+srv://gauravraj:Mongo%40raj11@cluster0.cddlpyx.mongodb.net/FoodApp?retryWrites=true&w=majority" ;  mongoose.connect(mongoURL, {useNewUrlParser: true,useUnifiedTopology:true, useNewUrlParser:true})  const con = mongoose.connection  con.on('open', () => {  console.log('connected...')  })  app.get('/', (req, res) => res.send('Server is working !' + port));  app.use('/api/dishes' , dishStaticRouter); //localhost:8000/api/dishes  app.use('/getdishes', dishDBRouter); //localhost:8000/getdishes  app.listen(port, () => console.log(`Food app is listening on port ${port}!`)) |
| dishModel.js : in models folder  const mongoose = require("mongoose");  const dishSchema = mongoose.Schema({  id : {type: Number , require} ,  name : {type: String , require} ,  image : {type: String , require} ,  category: {type: String , require} ,  label: {type: String , require} ,  price: {type: Number , require} ,  description:{type: String , require},  comments: []  },  {  timestamps: true,  })  const dishModel = mongoose.model("dishes",dishSchema);    module.exports = dishModel; |
| dishesDB.router.js : in routes folder  const express = require('express')  const router = express.Router();  const Dishes = require("../models/dishModel.js");  router.route('/').get( (req, res) => {  Dishes.find().then((docs)=> {  if(!docs){  res.status(400).send({message: "something went wrong"});  }else{  res.send(docs);  }  }).catch((err)=>{  res.status(500).send({message: "server not available"});  })    });  module.exports = router; |
|  |
|  |