**1.create package.json**

>npm init

**2. Install expressjs**

>npm i express --save

**3.meanBackEnd/index.js**

var http = require('http');

var express=require('express');

var port = process.env.port || 8080;

var app= express();

var appRoutes=require('./routes/appRoutes');

app.use('/',appRoutes);

http.createServer(app).listen(port);

console.log("Backend Running on Port :",port )

**4.meanBackend/routes/approutes.js**

var express=require('express');

var router=express.Router();

router.post('/create',(req,res,next)=>{

res.status(200).json({msg:"Post Request is working"});

});

router.get('/read',(req,res,next)=>{

res.status(200).json({msg:"Get Request is working"});

});

router.put('/update',(req,res,next)=>{

res.status(200).json({msg:"Update Request is working"});

});

router.delete('/delete/:id',(req,res,next)=>{

res.status(200).json({msg:"Delete Request is working"});

});

module.exports=router

<http://localhost:8080/read>

**5. Install Mongoose**

>npm install mongoose --save

**6. install bodyparser**

npm install body-parser --save

**index.js**

var bodyParser=require('body-parser');

app.use(bodyParser.urlencoded({extended:true}));

app.use(bodyParser.json());

**now :----------------------------------------------------------------**

**Index.js**

var http = require('http');

var express=require('express');

var port = process.env.port || 8080;

var app= express();

var appRoutes=require('./routes/appRoutes');

var mongoose =require('mongoose');

var bodyParser=require('body-parser');

mongoose.connect('mongodb://localhost/meanDb',{useUnifiedTopology:true,useNewUrlParser:true});

app.use(bodyParser.urlencoded({extended:true}));

app.use(bodyParser.json());

app.use('/',appRoutes);

http.createServer(app).listen(port);

console.log("Backend Running on Port :",port );

**appRoutes.js**

var express=require('express');

var router=express.Router();

var Country=require('../models/dataSchema');

router.post('/create',(req,res,next)=>{

var newCountry=new Country({

name:req.body.name,

capital:req.body.capital

});

newCountry.save((err,country)=>{

if(err)

res.status(500).json({errmsg:err});

res.status(200).json({msg:country});

});//res.status(200).json({msg:"Post Request is working"});

});

router.get('/read',(req,res,next)=>{

Country.find({},(err,countries)=>{

if(err)

res.status(500).json({errmsg:err});

res.status(200).json({msg:countries});

}); // res.status(200).json({msg:"Get Request is working"});

});

router.put('/update',(req,res,next)=>{

Country.findById(req.body.\_id,(err,country)=>{

if(err)

res.status(500).json({errmsg:err});

country.name=req.body.name;

country.capital=req.body.capital;

country.save((err,country)=>{

if(err)

res.status(500).json({errmsg:err});

res.status(200).json({msg:country});

})

});

// res.status(200).json({msg:"Update Request is working"});

});

router.delete('/delete/:id',(req,res,next)=>{

Country.findOneAndRemove({\_id:req.params.id},(err,country)=>{

if(err)

res.status(500).json({errmsg:err});

res.status(200).json({msg:countries});

});// res.status(200).json({msg:"Delete Request is working"});

});

module.exports=router;

**dataSchema.js**

var mongoose =require('mongoose');

var countrySchema=mongoose.Schema({

name:{type:String},

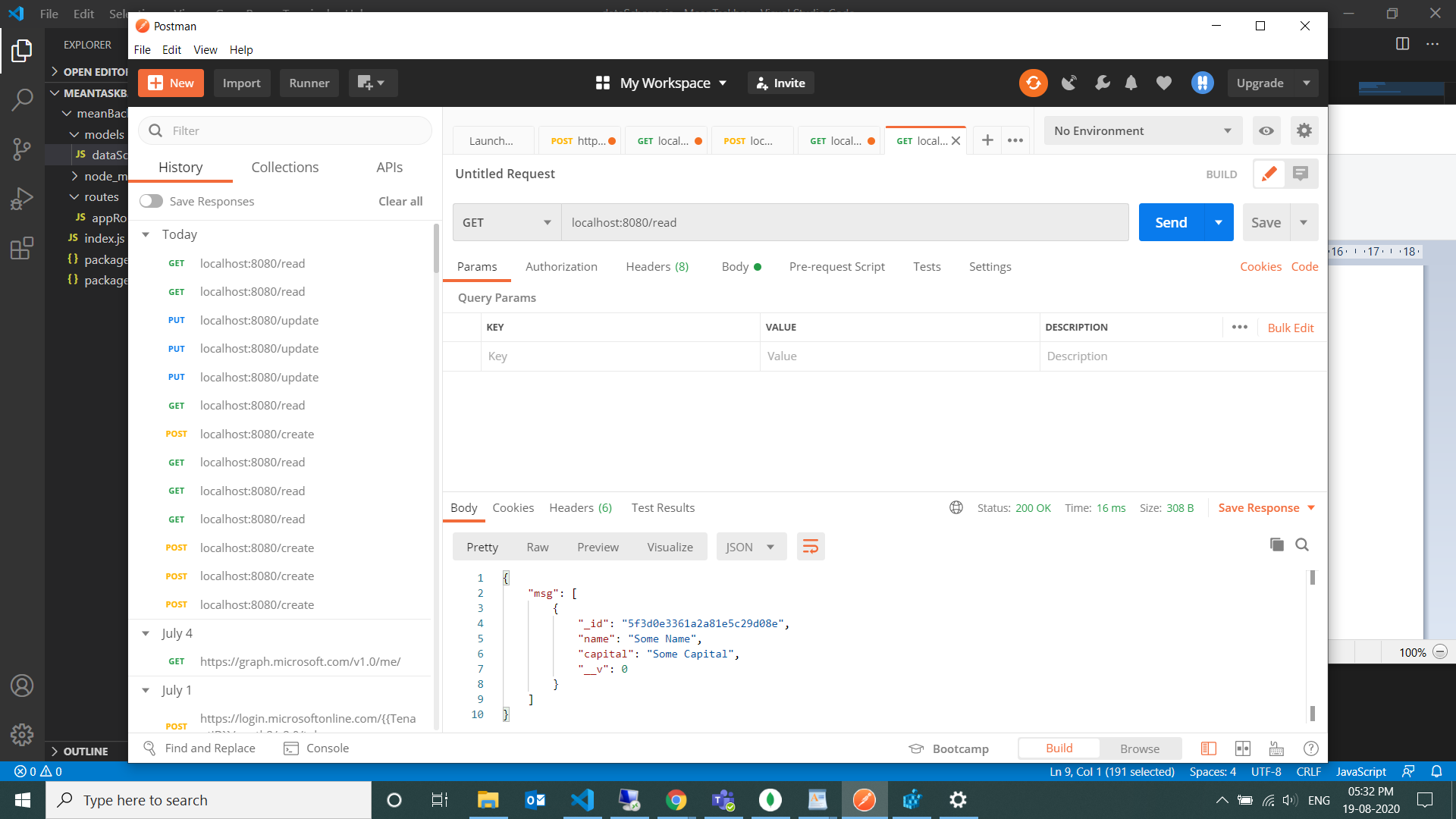
capital:{type:String}

});

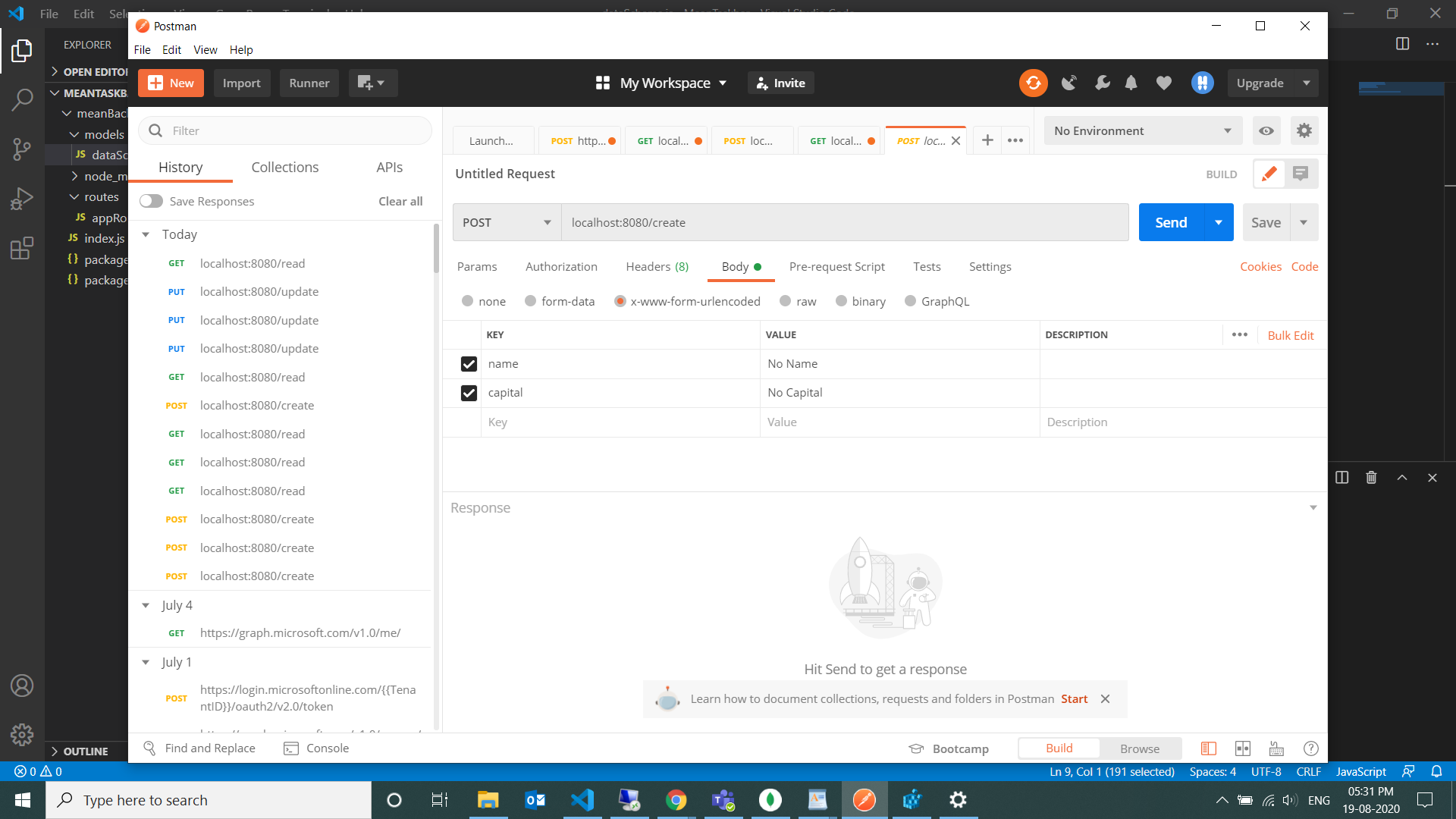
module.exports=mongoose.model('country',countrySchema);

**Postman**

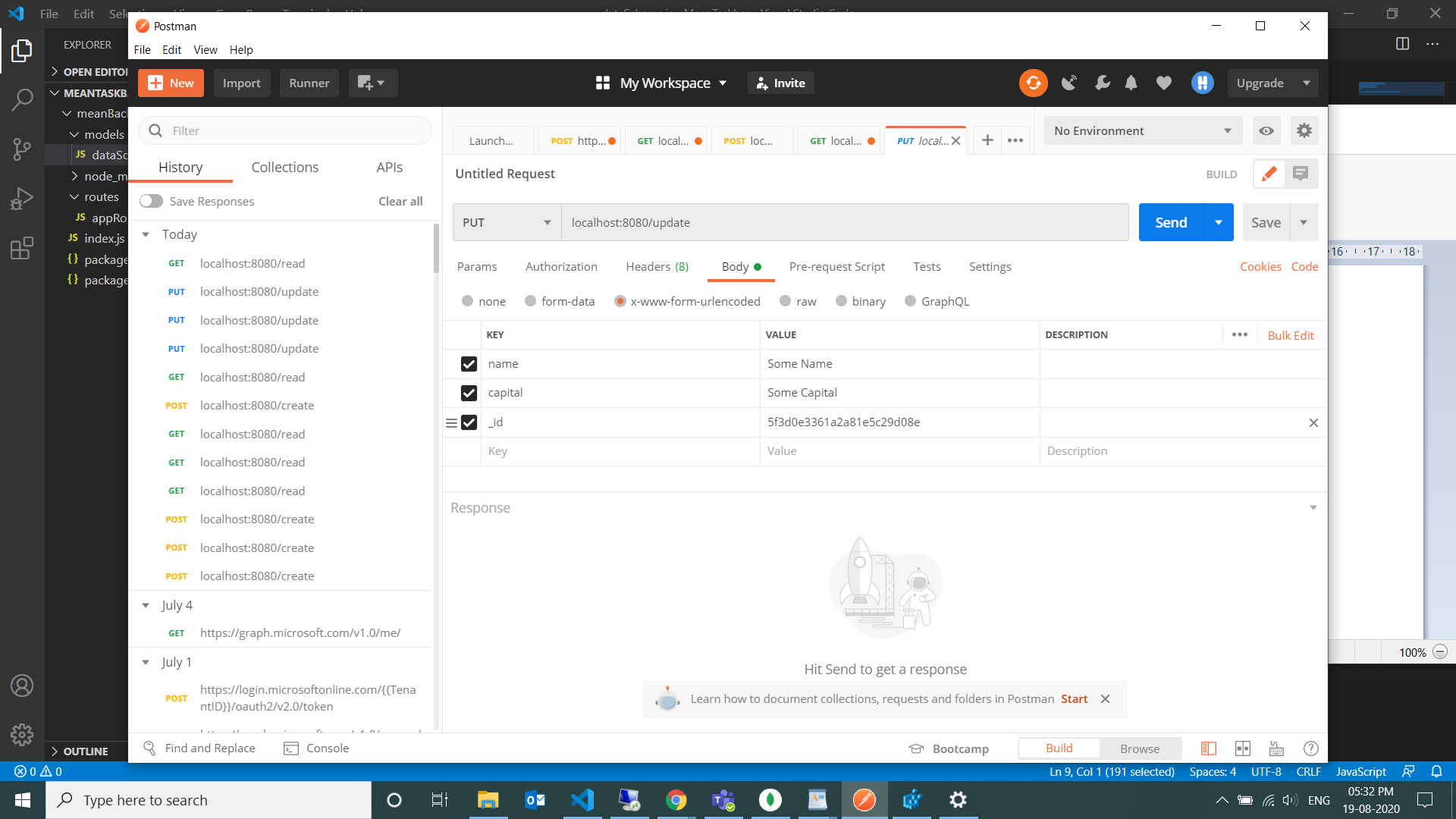
**localhost:8080/read**

****

**localhost:8080/create**

****

**localhost:8080/update**

****

**Angular app**

>ng new MeanAngular

>ng g c Components/navbar

>ng g c Components/createUpdate

>ng g c Components/list

>ng g class country

**Bootstrap:**

Go to Bootstrap.io

Index.html

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z" crossorigin="anonymous">

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script>

Index.html

<!doctype html>

<html lang="en">

<head>

  <meta charset="utf-8">

  <title>MeanAngular</title>

  <base href="/">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="icon" type="image/x-icon" href="favicon.ico">

  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z" crossorigin="anonymous">

</head>

<body>

  <app-root></app-root>

  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script>

    <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script>

    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script>

</body>

</html>

**Services:**

>ng g service shared/country

App.module.ts

import {CountryService}from './shared/country.service';

 providers: [CountryService],

import {HttpClientModule} from '@angular/common/http';

  imports: [

    BrowserModule,

    AppRoutingModule,

    HttpClientModule

  ],

Country.Service.ts

import { Injectable } from '@angular/core';

import {HttpClient,HttpHeaders} from  '@angular/common/http';

import {Country} from '../country';

@Injectable({

  providedIn: 'root'

})

export class CountryService {

private baseUri:string='http://localhost:8080';

private headers=new HttpHeaders().set('Content-Type','application/json')

constructor(private http:HttpClient) { }

createCountry(country:Country){

 return this.http.post(this.baseUri+'/create',country,{headers:this.headers});

}

readCountries(){

  return this.http.get(this.baseUri+'/read',{headers:this.headers});

  }

updateCountry(country:Country){

  return this.http.put(this.baseUri+'/update',country,{headers:this.headers});

    }

deleteCountry(id:string){

  return this.http.delete(this.baseUri+'/delete'+id,{headers:this.headers});

  }

}

list.component.ts

import { Component, OnInit } from '@angular/core';

import {CountryService} from '../../shared/country.service';

import {Country} from '../../country';

@Component({

  selector: 'app-list',

  templateUrl: './list.component.html',

  styleUrls: ['./list.component.css']

})

export class ListComponent implements OnInit {

private countries:Country;

  constructor(private \_countryService:CountryService) { }

  ngOnInit(): void {

    this.readCountries();

  }

  readCountries(){

    this.\_countryService.readCountries().subscribe(

      data=>{

             console.log(data);

       },

      error=>{console.log(error);

       }

    );

  }

}

List.component.html

<div class="card">

    <div class="card-body">

        <h4 class="card-title">Countries and Capitals</h4>

        <table class="table table-hover">

            <thead>

                <tr>

                    <th>ID</th>

                    <th>Name</th>

                    <th>Capital</th>

                </tr>

            </thead>

            <tbody>

                <tr \*ngFor="let country of countries">

                <td>{{country.\_id}}</td>

                <td>{{country.name}}</td>

                <td>{{country.Capital}}</td>

                <td><button class="btn btn-danger">Delete</button></td>

                <td><button class="btn btn-primary">Update</button></td>

            </tr>

            </tbody>

        </table>

    </div>

</div>

**MeanBackend**

>npm install cors –save

Indes.js

var http = require('http');

var express=require('express');

var port = process.env.port || 8080;

var app= express();

var appRoutes=require('./routes/appRoutes');

var mongoose =require('mongoose');

var bodyParser=require('body-parser');

var cors=require('cors');

mongoose.connect('mongodb://localhost/meanDb',{useUnifiedTopology:true,useNewUrlParser:true});

app.use(cors());

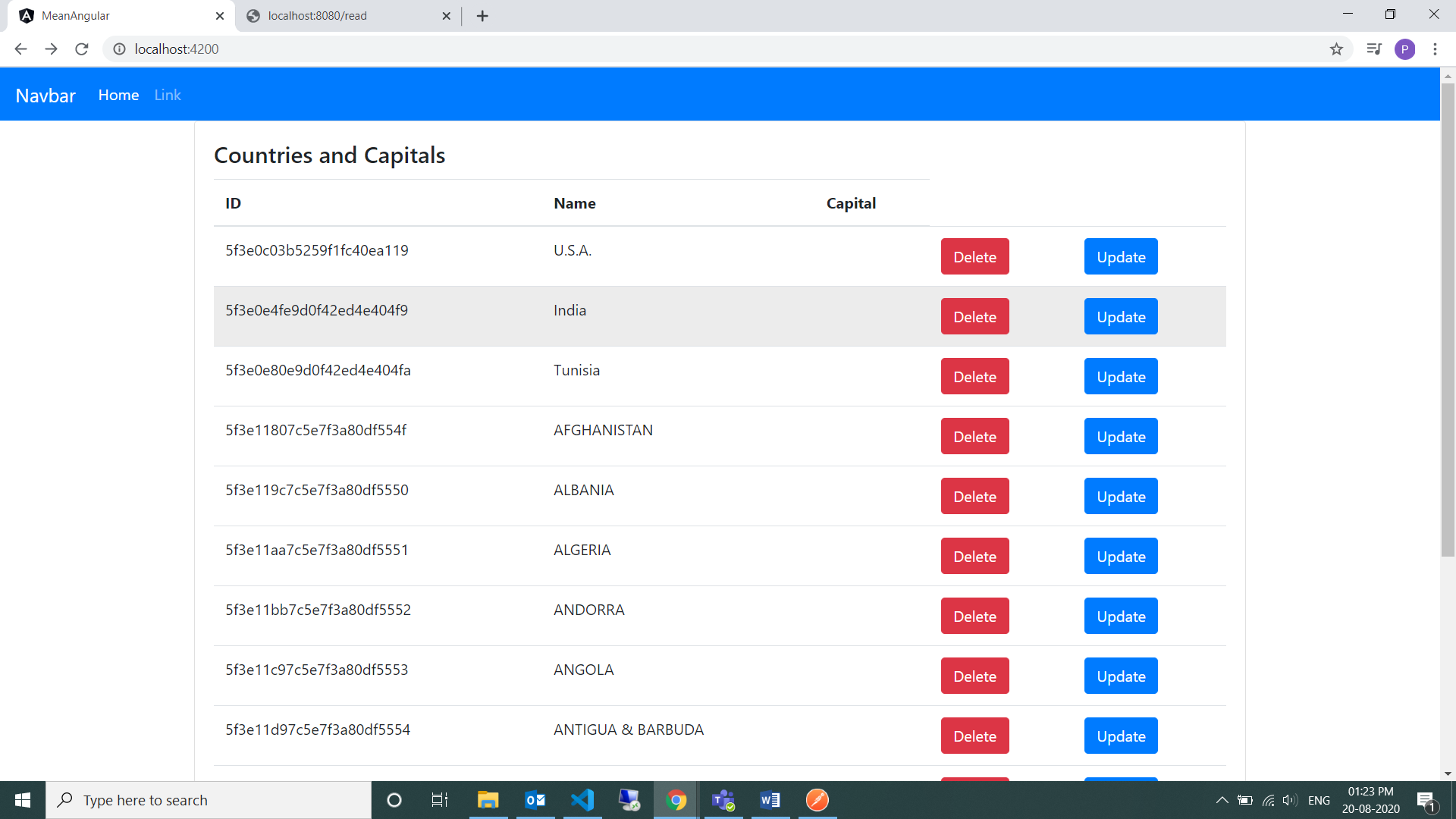
app.use(bodyParser.urlencoded({extended:true}));

app.use(bodyParser.json());

app.use('/',appRoutes);

http.createServer(app).listen(port);

console.log("Backend Running on Port :",port );



New

Navbar.html

 <a class="nav-link" href="#" (click)="newCountry($event)">New</a>

Navbar.ts

  newCountry(event:any){

    event.preventDefault();

    this.countryService.setter(new Country());

    this.router.navigate(['/createUpdate']);

  }

Service.ts

setter(country:Country){

  this.country=country;

}

getter(){

  return this.getter;

}