1)

Npm install express

Npm install nodemon

Npm install body-parser

Npm install mongoos

Npm install dotenv

Npm install cors

2) connection create

const mongoose = require("mongoose");

mongoose.set('strictQuery', false);

const DB = "mongodb+srv://mernuser:supperpassword@cluster0.cqczybv.mongodb.net/mydatabase";

mongoose.connect(DB,{

    useNewUrlParser: true,

    useUnifiedTopology: true

}).then(()=>console.log("connection start")).catch((error)=> console.log(error.message));

3) create schema

const mongoose = require("mongoose");

const userSchema =new mongoose.Schema({

    fullname:{

        type:String

    },

    phone:{

        type:String

    },

    email:{

        type:String

    },

    pass:{

        type:String

    }

});

const userdata = new mongoose.model("studentlist",userSchema);

module.exports = userdata;

4) create route

const express = require("express");

const users = require("../modals/Schema");

const router = express.Router();

/\*create or post api \*/

router.post("/reg", async(req,res)=>{

    const {fullname,email,phone,pass} = req.body;

    const adduser = new users({

        fullname,email,phone,pass

    });

        await adduser.save();

        return res.status(400).json(adduser);

        // console.log(adduser);

});

module.exports = router;

5) get api

a) create get api

router.get("/getdata",async(req,res)=>{

    const adduser = await userdata.find();

    res.json(adduser);

    console.log(adduser);

});

b) frontend

const [datavariable,datafunction] =useState([]);

    const mygetdata = async(e)=>{

      const res = await fetch("http://localhost:8000/getdata",{

        method:"GET",

        headers:{"Content-Type":"application/json"}

      })

      const data = await res.json();

      console.log(data);

      datafunction(data)

    };

    useEffect(()=>{

      mygetdata();

    },[]);

c) delete records

1) create api

/\* delete api\*/

router.delete("/deletecurrentrecord/:id", async(req,res)=>{

    const {id} = req.params;

    const a = await userdata.findByIdAndDelete({\_id:id})

    console.log(a);

    res.status(201).json(a);

});

2) frontend (function)

const mydeleterecord = async(id)=>{

      const delrec = await fetch(`http://localhost:8000/deletecurrentrecord/${id}`,{

        method:"DELETE",

        headers:{"Content-Type":"application/json"}

      })

      const deleterow = await delrec.json();

      console.log(deleterow);

      mygetdata();

    };

3) send id

<button type='button' className='btn btn-danger btn-sm me-2' onClick={()=>mydeleterecord(a.\_id)}>Del</button>