CAT FORM

import { useState } from "react";

function CatForm (props) {

  const {onAdd} = props;

  const [name,setName]=useState(" ");

  const [type,setType]=useState(" ");

  const [mass,setMass]=useState(" ");

  const addCat=(event)=>{

    event.preventDefault();

    onAdd({

        name,

        type,

        mass

  })

  setName('');

  setType('');

  setMass('');

  }

    return (

      <form>

    <p>Cat Form</p>

    <label>Name</label>

    <input type="text" name="name" id="name" onChange={(event)=>setName(event.target.value)} value= {name}></input>

    <br/>

    <label>Type</label>

    <input type="text" name="type" id="type" onChange={(event)=>setType(event.target.value)} value={type}></input>

    <br/>

    <label>Mass</label>

    <input type="text" name="mass" id="mass" onChange={(event)=>setMass(event.target.value)} value={mass}></input>

    <br/>

    <input type="submit" value='add' id='add-button'onClick={addCat}></input>

      </form>

    )

}

  export default CatForm

REST

app.get('/food-items-ex', async (req, res) => {

    // TOPIC: REST

    try{

        if(req.query.shelfNo && req.query.shelfSize){

            let shelfSize,shelfNo

            if(req.query.shelfSize>10){

                res.status(400).json({message: 'shelf size too big'})

            }

            else{

                shelfNo=req.query.shelfNo;

                shelfSize=req.query.shelfSize;

                let off=shelfSize\*shelfNo

                let foodItems= await FoodItem.findAll({offset:off,limit:shelfSize});

                res.status(200).json(foodItems)

            }

        }

        else{

            res.status(500).json({message: 'missing params'})

        }

    }catch(err){

        res.status(500).json({message: 'server error'})

    }

SERVER CLIENT

<body>

    List of cars

    <div id="list">

    </div>

    <button id="getData">Get Data</button>

    <script type="text/javascript">

        const btn=document.getElementById("getData")

        const carsList=document.getElementById("list")

        function getData(){

            fetch("http://localhost:8080/cars").then(res=>{return res.json()})

        .then(data=>{

            let cars=data;

            carsList.innerHTML=""

            for(let car of cars){

                if(car.color=="red"){

                    const carName=document.createElement("p")

                    carName.appendChild(document.createTextNode(car.name));

                    carsList.appendChild(carName)

                }

            }

            // cars=cars.filter(x=>x.color=="red")

            // console.log(cars)

        })

        }

        btn.addEventListener("click",e=>{

            getData();

        })

    </script>

JAVASCRIPT

class Vehicle {

    constructor(name){

        this.name = name

    }

    drive(distance){

        return `${this.name} drives ${distance}`

    }

    // TODO: implement method

    makeStop(){

        return `${this.name} makes a stop`

    }

}

// TODO: implement Car

class Car extends Vehicle {

    constructor(name){

        if(typeof name=="string"){

            super(name);

        }else{

            throw new Error("CreationError")

        }

    }

    drive(distance){

        return `${this.name} is a Car and cannot drive`

    }

    fuels(liters){

        return `${this.name} fuels ${liters}`

    }

}

module.exports.Vehicle = Vehicle

module.exports.Car = Car