package com.sat.cce;

import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.PreparedStatement;  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.sql.Statement;  
import java.util.ArrayList;  
import java.util.List;

import org.springframework.web.bind.annotation.GetMapping;  
import org.springframework.web.bind.annotation.PathVariable;  
import org.springframework.web.bind.annotation.RestController;  
import org.springframework.web.client.RestTemplate;

@RestController  
public class MyService {

@GetMapping("/first")  
public String retY() {  
return "Radha Krishna";  
}  
  
@GetMapping("/allrec")  
public List<Register> retLog(){  
List<Register> lr=new ArrayList<Register>();  
int i=0;  
try {  
Class.forName("org.postgresql.Driver");  
Connection conn=DriverManager.getConnection("jdbc:postgresql://localhost:5432/EmpDB", "postgres", "admin");;  
Statement st=conn.createStatement();  
ResultSet rs=st.executeQuery("select \* from \"regist\"");  
while(rs.next()) {  
Register reg=new Register();  
reg.setName(rs.getString(1));  
reg.setEmail(rs.getString(2));  
reg.setMobile(rs.getString(3));  
lr.add(reg);  
i++;  
}  
} catch (ClassNotFoundException e) {  
e.printStackTrace();  
} catch (SQLException e) {  
e.printStackTrace();  
}  
  
Register[] arr=new Register[i];  
lr.toArray(arr);  
return lr;  
}  
  
@GetMapping("/insrec/{name}/{email}/{mobile}")  
public Register insLog(@PathVariable String name,@PathVariable String email,@PathVariable String mobile){  
Register lr=new Register();  
lr.setName(name);  
lr.setEmail(email);  
lr.setMobile(mobile);  
  
try {  
Class.forName("org.postgresql.Driver");  
Connection conn=DriverManager.getConnection("jdbc:postgresql://localhost:5432/EmpDB", "postgres", "admin");;  
String query="insert into \"regist\" values('"+name+"','"+email+"','"+mobile+"')";  
PreparedStatement ps=conn.prepareStatement(query);  
ps.execute();  
} catch (ClassNotFoundException e) {  
e.printStackTrace();  
} catch (SQLException e) {  
e.printStackTrace();  
}  
return lr;  
}  
  
@GetMapping("/uprec/{name}/{email}/{mobile}")  
public Register upLog(@PathVariable String name,@PathVariable String email,@PathVariable String mobile){  
Register lr=new Register();  
lr.setName(name);  
lr.setEmail(email);  
lr.setMobile(mobile);  
  
try {  
Class.forName("org.postgresql.Driver");  
Connection conn=DriverManager.getConnection("jdbc:postgresql://localhost:5432/EmpDB", "postgres", "admin");;  
String query="update \"regist\" set name='"+name+"',mobile='"+mobile+"' where email='"+email+"'";  
PreparedStatement ps=conn.prepareStatement(query);  
ps.executeUpdate();  
} catch (ClassNotFoundException e) {  
e.printStackTrace();  
} catch (SQLException e) {  
e.printStackTrace();  
}  
return lr;  
}  
  
@GetMapping("/delrec/{email}")  
public String delLog(@PathVariable String email){  
String sta="InComplete";  
try {  
Class.forName("org.postgresql.Driver");  
Connection conn=DriverManager.getConnection("jdbc:postgresql://localhost:5432/EmpDB", "postgres", "admin");;  
String query="delete from \"regist\" where email='"+email+"'";  
PreparedStatement ps=conn.prepareStatement(query);  
ps.executeUpdate();  
sta="complete";  
} catch (ClassNotFoundException e) {  
e.printStackTrace();  
} catch (SQLException e) {  
e.printStackTrace();  
}  
return sta;  
}  
  
}

[3:57 PM] satish (Guest)

importReactfrom'react';

classDelFormextendsReact.Component {​​​​​​​​

constructor(props) {​​​​​​​​

super(props);

this.state= {​​​​​​​​

a: ''

        }​​​​​​​​

this.onS=this.onS.bind(this);

    }​​​​​​​​

onS= (e) => {​​​​​​​​

        e.preventDefault();

if (this.state.a==='') {​​​​​​​​

alert('Invalid entry');

        }​​​​​​​​ else {​​​​​​​​

leturl="http://localhost:8080/delrec/"+this.state.a;

fetch(url).then(j=> j.json()).then(j=>console.log(j));

        }​​​​​​​​

    }​​​​​​​​

render() {​​​​​​​​

return (

<div>

                <formonSubmit={​​​​​​​​this.onS}​​​​​​​​>

                    <tableclassName="table table-responsive">

                        <tr><td>Email</td><td><inputtype="email"required="true"onChange={​​​​​​​​(e) =>this.setState({​​​​​​​​ a: e.target.value }​​​​​​​​)}​​​​​​​​name="email"/></td></tr>

                        <tr><td><inputtype="submit"value="Send"/></td>

                            <td><inputtype="reset"value="Cancel"/></td></tr>

                        <tr><td><inputtype="number"min="0"max="100"name="t2"/></td></tr>

</table>

</form>

</div>

        );

    }​​​​​​​​

}​​​​​​​​

exportdefaultDelForm;

​

[3:57 PM] satish (Guest)

importReactfrom"react";

classMyFormextendsReact.Component {​​​​​​​​

constructor(props) {​​​​​​​​

super(props);

this.state= {​​​​​​​​

name: '',

email: '',

mobile: ''

        }​​​​​​​​

this.onC=this.onC.bind(this);

this.onS=this.onS.bind(this);

    }​​​​​​​​

onS(e) {​​​​​​​​

        e.preventDefault();

leturl="http://localhost:8080/uprec/"+this.state.name+"/"+this.state.email+"/"+this.state.mobile;

fetch(url).then(j=> j.json()).then(j=>console.log(j));

    }​​​​​​​​

onC(e) {​​​​​​​​

leta= e.target.name;

letb= e.target.value;

this.setState({​​​​​​​​ [a]: b }​​​​​​​​);

    }​​​​​​​​

render() {​​​​​​​​

return (

<div>

                <formonSubmit={​​​​​​​​this.onS}​​​​​​​​>

                    <tableclassName="table table-responsive">

                        <tr><td>Name</td><td><inputtype="text"name="name"onChange={​​​​​​​​this.onC}​​​​​​​​/></td></tr>

                        <tr><td>Email</td><td><inputtype="email"name="email"onChange={​​​​​​​​this.onC}​​​​​​​​/></td></tr>

                        <tr><td>Mobile</td><td><inputtype="text"name="mobile"onChange={​​​​​​​​this.onC}​​​​​​​​/></td></tr>

                        <tr><td><inputtype="submit"value="Send"/></td>

                            <td><inputtype="reset"value="Cancel"/></td></tr>

</table>

</form>

                <divclassName="alert alert-warning">{​​​​​​​​this.state.name+''+this.state.email+''+this.state.mobile}​​​​​​​​</div>

</div>

        );

    }​​​​​​​​

}​​​​​​​​

exportdefaultMyForm;