Compétence 12 : FRONT END REACT

|  |  |
| --- | --- |
| Filière | : Full Stack 202 CONTROLE N1 le 30/10/2023 |

On souhaite créer un formulaire en utilisant React JS qui gère la liste des salaries suivantes :

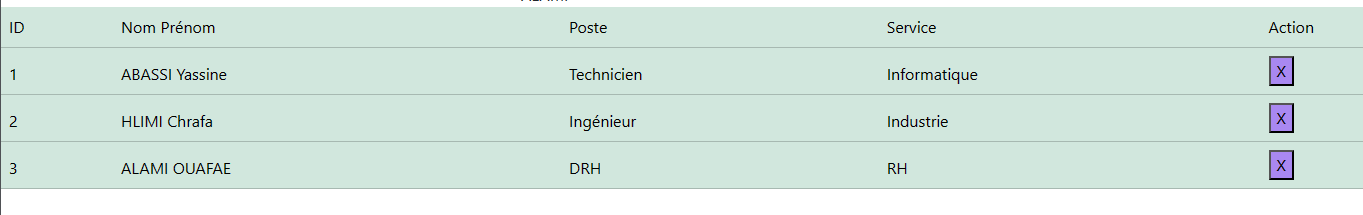
Tsalaries=[

        {id:1, Nom: "ABASSI", Prénom: "Yassine", Poste: "Technicien", Service :"Informatique"},

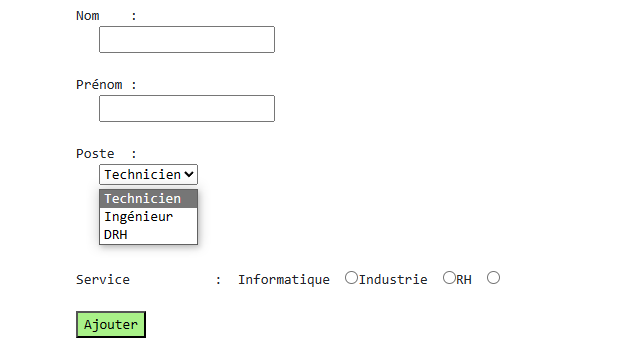
        {id:2, Nom: "HLIMI", Prénom: "Chrafa", Poste: "Ingénieur", Service :"Industrie"},

        {id:3,Nom: "ALAMI", Prénom: " OUAFAE", Poste: "DRH", Service :"RH"}]

1. Créer une composante Salarié1.js qui permet d’afficher les informations des salariées (initialisées la variable de state salaries en appelant le tableau Tsalaries) (5 pts)

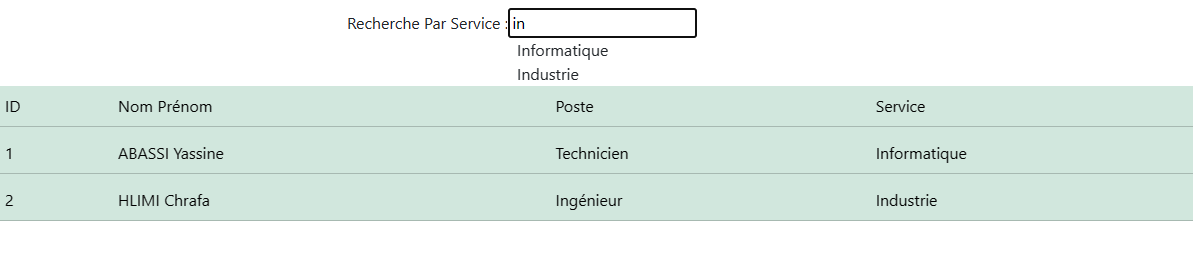


1. Ajouter à la composante précédente la méthode supprimé qui permet de supprimer un salarie par son id. (5 pts)
2. Créer un composant Salarié2.js permettant de saisir les informations d’un salarie dans une variable state (5 pts)

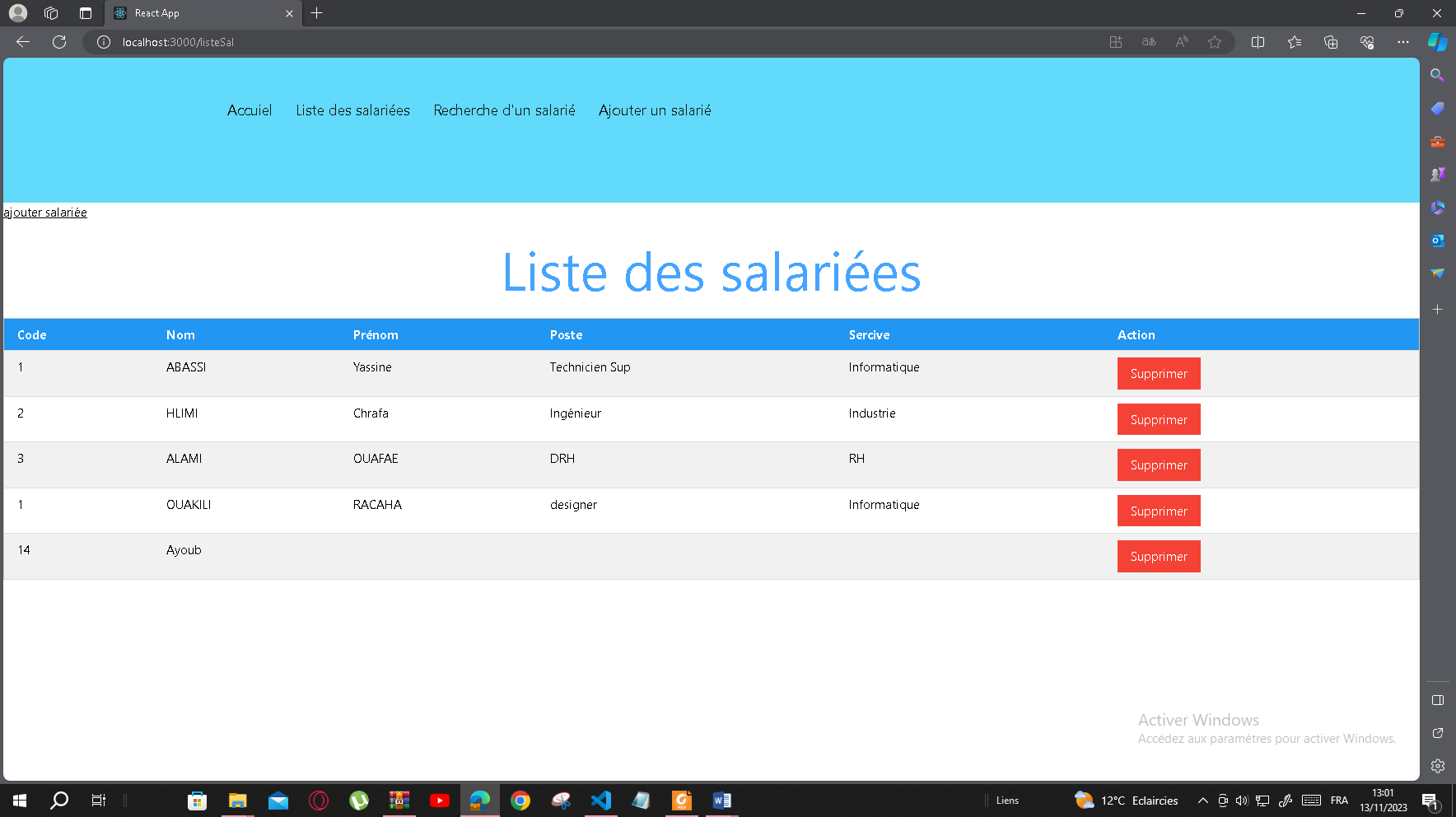


1. Créer la composante qui permettant d’effectuer la recherche par service sur les données stockés dans l’objet state. (5 pts)

Exemple :



Correction avec utilisation des router



AppRouter.js

import React, { useState } from 'react'

import { Routes,Link, Route, BrowserRouter } from 'react-router-dom'

import Accuiel from './Accuiel'

import Salarié from './Salarié'

import Rechercher from './Rechercher'

import Ajouter from './Ajouter'

const Tsalaries=[

    {id:1, Nom: "ABASSI", Prénom: "Yassine", Poste: "Technicien Sup", Service :"Informatique"},

    {id:2, Nom: "HLIMI", Prénom: "Chrafa", Poste: "Ingénieur", Service :"Industrie"},

    {id:3,Nom: "ALAMI", Prénom: " OUAFAE", Poste: "DRH", Service :"RH"},

    {id:1, Nom: "OUAKILI", Prénom: "RACAHA", Poste: "designer", Service :"Informatique"},

]

export default function AppRouter() {

    const [listsal,setListsal]=useState(Tsalaries)

  return (

<>

<div class="scrollmenu">

        <div class="div1">

<Link to='/'>Accuiel</Link>

<Link to='/listeSal'>Liste des salariées</Link>

<Link to='/rechercher'>Recherche d'un salarié </Link>

<Link to='/ajout'>Ajouter un salarié</Link>

</div></div>

<div class="scrollmenu2">

<Routes>

<Route path='/' element={<Accuiel/>}></Route>

<Route path='/listeSal' element={<Salarié listsal={listsal}/>}></Route>

<Route path='/rechercher' element={<Rechercher listsal={listsal}></Rechercher>}></Route>

<Route path='/ajout' element={<Ajouter listsal={listsal} setListsal={setListsal}></Ajouter>}></Route>

<Route path='/ajouts' element={<Ajouter listsal={listsal} setListsal={setListsal}></Ajouter>}></Route>

</Routes>

</div>

</>

  )

}

Index.js

import React from 'react';

import ReactDOM from 'react-dom/client';

import './index.css';

import App from './App';

import reportWebVitals from './reportWebVitals';

import AppRouter from './AppRouter';

import { BrowserRouter } from 'react-router-dom';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(

  <BrowserRouter>

  <AppRouter/>

  </BrowserRouter>

);

*// If you want to start measuring performance in your app, pass a function*

*// to log results (for example: reportWebVitals(console.log))*

*// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals*

reportWebVitals();

index.html

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="utf-8" />

    <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />

    <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">

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

    <meta name="theme-color" content="#000000" />

    <meta

      name="description"

      content="Web site created using create-react-app"

    />

    <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />

*<!--*

*manifest.json provides metadata used when your web app is installed on a*

*user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/*

*-->*

    <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />

*<!--*

*Notice the use of %PUBLIC\_URL% in the tags above.*

*It will be replaced with the URL of the `public` folder during the build.*

*Only files inside the `public` folder can be referenced from the HTML.*

*Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC\_URL%/favicon.ico" will*

*work correctly both with client-side routing and a non-root public URL.*

*Learn how to configure a non-root public URL by running `npm run build`.*

*-->*

    <title>React App</title>

  </head>

  <body>

    <noscript>You need to enable JavaScript to run this app.</noscript>

    <div id="root"></div>

*<!--*

*This HTML file is a template.*

*If you open it directly in the browser, you will see an empty page.*

*You can add webfonts, meta tags, or analytics to this file.*

*The build step will place the bundled scripts into the <body> tag.*

*To begin the development, run `npm start` or `yarn start`.*

*To create a production bundle, use `npm run build` or `yarn build`.*

*-->*

  </body>

</html>

Salarié.js

import React from 'react'

import { Link,Routes,Route } from 'react-router-dom'

import Ajouter from './Ajouter'

export default function Salarié({listsal}) {

  return (

    <>

    <Link to='/ajouts'>ajouter salariée</Link>

        <h1 style={{textAlign:"center", color:"#45a4ff", fontSize:"65px"}}>Liste des salariées</h1>

    <table class="w3-table-all">

   <thead>

      <tr class="w3-blue">

<th>Code</th><th>Nom </th><th>Prénom</th><th>Poste</th><th>Sercive</th><th>Action</th>

</tr></thead>

{listsal.map((sal)=><tr>

    <td>{sal.id}</td><td>{sal.Nom}</td><td>{sal.Prénom}</td><td>{sal.Poste}</td><td>{sal.Service}</td><td><button class="w3-button w3-red">Supprimer</button></td>

</tr> ) }

   </table>

</>

  )

}

Recherche.js

import React,{useState} from 'react'

export default function Rechercher(props) {

    const [textrech,setTextrech]=useState("")

    const [listrech,setListrech]=useState([]);

    const  chercher=()=>{

        setListrech(props.listsal.filter((sal)=> sal.Service.toLowerCase()===textrech.toLowerCase()))

    }

  return (

    <div style={{margin :"35px"}}>

<h1 style={{textAlign:"center", color:"#45a4ff", fontSize:"65px"}}>Liste des salariées</h1>

    Recherche par Service:<input type="text"  onChange={(ev)=>setTextrech(ev.target.value)}></input><button  class="w3-btn w3-green" onClick={chercher}>Rechercher</button>

    <br></br>

    <table style={{margin :"55px"}} class="w3-table-all">

   <thead>

      <tr class="w3-blue">

<th>Code</th><th>Nom </th><th>Prénom</th><th>Poste</th><th>Sercive</th></tr></thead>

        {listrech.map((sal)=><tr><td> {sal.id}  </td><td> {sal.Nom}</td><td>{sal.Prénom} </td><td> {sal.Poste} </td><td>{sal.Service}</td>

        </tr> ) }

</table>

    </div>

  )

}

Ajouter.js

import React, { useState } from 'react'

export default function Ajouter(props) {

const [idAj,setidAJ]=useState();

const [nomAJ,setNomAJ]=useState();

   const enregistrer=()=>{

*//const posid=props.listsal.findIndex((sal)=>{ return sal.id===props.listsal.id})*

        const T=[...props.listsal];

        T.push({id:idAj,Nom:nomAJ})

    props.setListsal(T)

     }

  return (

    <div style={{marginLeft:"450px" ,marginTop:"100px"}}>

    <pre>

       ID             :     <input type='=text'  onChange={(ev)=>setidAJ(ev.target.value)} ></input><br/><br/>

       Nom            :     <input type='text' onChange={(ev)=>setNomAJ(ev.target.value) }></input><br/><br/>

       Prénom         :     <input type='text'  ></input><br/><br/>

       Poste          :     <select >

     <option>Technicien Sup</option>

     <option>Ingénieur</option>

     <option>DRH</option>

     <option>designer</option>

        </select><br/><br/>

       Service         :    Informatique <input type="radio"  id="idI" name='s1' value="Informatique"  ></input>

                                RH  <input type="radio" id="idR" name='s1' value="RH"  ></input><br/>

                                <button onClick={enregistrer} style={{backgroundColor:"#A7A711",height:"30px",color:"white"}}> Enregistrer</button>

                         </pre> <br/>

</div>

  )

}