



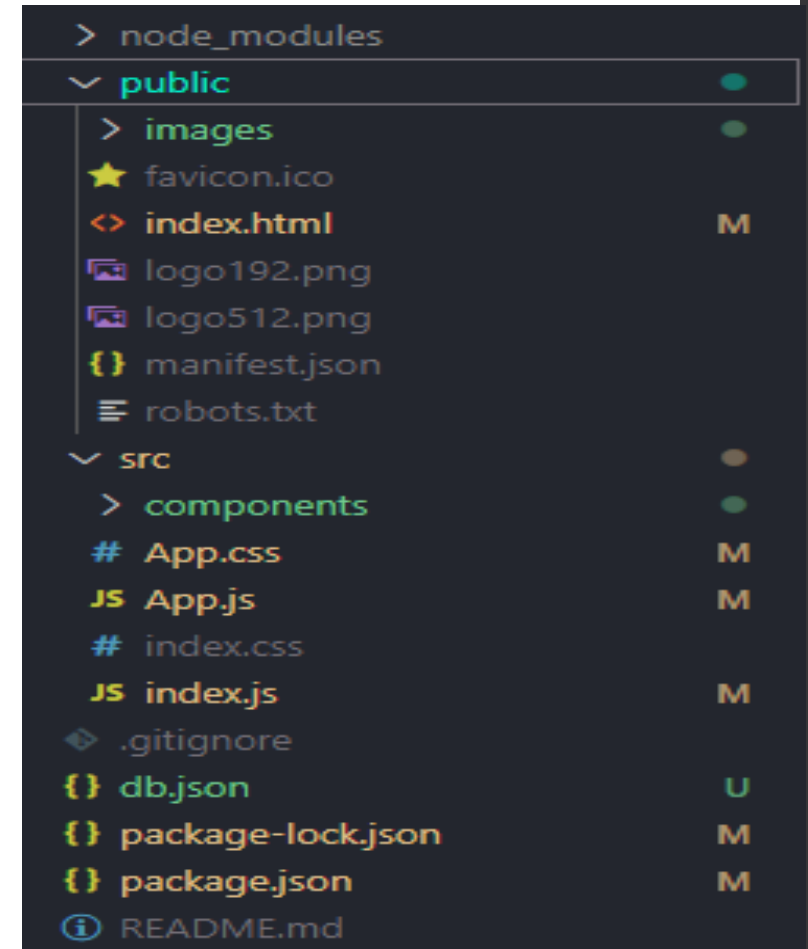
React JS

Développement
Front-end

Creation de Application

- Créer et Démarrer une Application React
 - `npx create-react-app`
 - `cd folder-name`
 - `npm start`
- Pour installer les dépendances :
 - `npm install dependence_name`

- Structure d'un project React :



Class VS Function Component

- Class :

```
1 import React, { Component } from 'react'
2
3 export default class AjouterUser extends Component {
4   render() {
5     return (
6       <div>AjouterUser</div>
7     )
8   }
9 }
10
```

- Function :

```
1 import React from 'react'
2
3 function AjouterUser() {
4   return (
5     <div>AjouterUser</div>
6   )
7 }
8
9 export default AjouterUser
```

```
1 import React from 'react'
2
3 const AjouterUser = () => {
4   return (
5     <div>AjouterUser</div>
6   )
7 }
8
9 export default AjouterUser
```

JSX

HTML	JSX
Class	className
<code><input type="text" ></code> ✓	<code><input type="text" ></code> ✗ <code><input type="text" /></code> ✓ ALL TAGS MUST BE CLOSE
for	htmlFor
onclick,onchange,onfocus...	onClick,onChange,onFocus...

Tags that are meant to be output as HTML (`<p> </p>`) must be all lowercase.

Tags that reference React components (`<SideBar />`) must be UpperCamelCase.

Utilisatation de javascript en JSX :

Exemple : `<p> age : { 2022-1999 } </p>`

Props

```
import * as React from 'react';

const App = () => {
  const greeting = 'Welcome to React';

  return (
    <div>
      <Welcome text={greeting} />
    </div>
  );
};

const Welcome = ({ text }) => {
  return <h1>{text}</h1>;
};

export default App;
```

```
import * as React from 'react';

const App = () => {
  const greeting = {
    title: 'React',
    description: 'Your component library for ...',
  };

  return (
    <div>
      <Welcome text={greeting} />
    </div>
  );
};

const Welcome = ({ text }) => {
  return (
    <div>
      <Headline title={`Welcome to ${text.title}`} />
      <Description paragraph={text.description} />
    </div>
  );
};

const Headline = ({ title }) => <h1>{title}</h1>;
const Description = ({ paragraph }) => <p>{paragraph}</p>;

export default App;
```

Hooks : useState & useEffect

```
1 import { useState } from 'react'
2 const Counter = () => {
3   const [count, setCount] = useState(0)
4   return (
5     <div>
6       <p>You clicked {count} times</p>
7       <button onClick={() => setCount(count + 1)}>Cl</button>
8     </div>
9   )
10 }
11 export default Counter;
```

```
1 import { useState, useEffect } from "react";
2 function CounterUseEffect() {
3   const [count, setCount] = useState(0);
4   useEffect(() => {
5     console.log(`You clicked ${count} times`)
6   })
7   return (
8     <div>
9       <p>You clicked {count} times</p>
10      <button onClick={() => setCount(count + 1)}>Cliquer ici</button>
11    </div>
12  )
13 }
14
15 export default CounterUseEffect;
```

```
useEffect(() => {
  console.log(`You clicked ${count} times`)
}, [])
return (
```

Router :

- Installation de react-router-dom
- Importer BrowserRouter , Routes (Switch for version 5 and older) et Route
- Comment utiliser le routage ?

1ère Etape : (BrowserRouter ou HashRouter)

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

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <BrowserRouter>
      <App />
    </BrowserRouter>
  </React.StrictMode>
);
```

2ème Etape : Routes & Route

```
import React from 'react';
import { Routes ,Route} from "react-router-dom";
import Home from './components/Home';
import AjouterUser from './components/AjouterUser';
import Navbar from './components/Navbar';
import ListUsers from './components/ListUsers';
export default function App() {
  return (
    <div>
      <Navbar />
      <Routes>
        <Route path="/" exact element={<Home />} />
        <Route path="/listUsers" exact element={<ListUsers />} />
        <Route path="/ajouterUser" exact element={<AjouterUser />} />
      </Routes>
    </div>
  )
}
```

3ème Etape : Link to

```
import React from 'react'
import { Link } from "react-router-dom";

const Navbar = () => {
  return (
    <div className="container">
      <nav className="navbar navbar-expand-lg navbar-light bg-light">
        <div className="container-fluid">
          <Link to="/"><div className="navbar-brand" >Home</div></Link>
          <Link to="/listUsers"><div className="navbar-brand">Liste des utilisateurs</div></Link>
          <Link to="/ajouterUser"><div className="navbar-brand">Ajouter des utilisateur </div></Link>
        </div>
      </nav>
    </div>
  )
}

export default Navbar
```

consommer un API :

Fetch

```
1 import React, {useState, useEffect} from 'react'
2
3 function ListUsers() {
4   const [resultat, setResultat] = useState([])
5   useEffect(()=>{
6     fetch('http://localhost:3001/user')
7       .then(response =>{
8         return response.json();
9       })
10      .then((data) =>{
11        setResultat(data)
12      })
13    },[])
14    return (
15      <div>
16        {resultat.map((row)=>{
17          return(
18            <h1>{row.name}</h1>
19          )
20        })}
21      </div>
22    )
23  }
24
25 export default ListUsers;
```

Axios

```
1 import React, {useState, useEffect} from 'react'
2 import axios from 'axios'
3 function ListUsers() {
4   const [resultat, setResultat] = useState([])
5   useEffect(()=>{
6     axios.get('http://localhost:3001/user').then((res)=>{
7       setResultat(res.data)
8     })
9   },[])
10  console.log(resultat);
11  return (
12    <div>
13      {resultat.map((row)=>{
14        return(
15          <h1>{row.name}</h1>
16        )
17      })}
18    </div>
19  )
20 }
21
22 export default ListUsers;
```

Axios : axios.post()

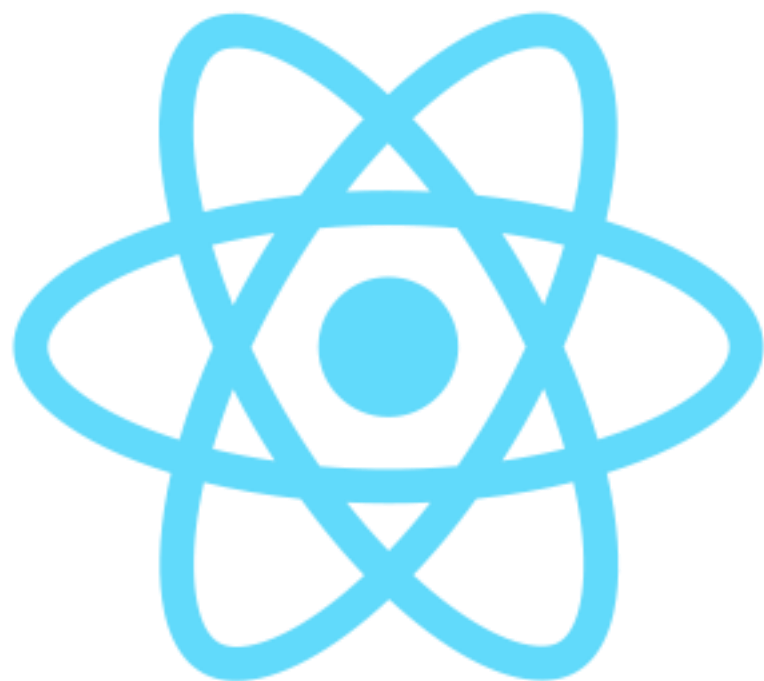
```
1  import React, {useState} from 'react'
2  import axios from 'axios'
3  function AddUsers() {
4      const [name, setName] = useState("")
5      const [address, setAddress] = useState("")
6      const [email, setEmail] = useState("")
7      const [contact, setContact] = useState("")
8      const [age, setAge] = useState("")
9
10     const addToJson = () => {
11         axios.post('http://localhost:3001/user',{
12             name: name,
13             adress: address,
14             email:email,
15             contact: contact,
16             age:age
17         })
18     }
19
20     return (
21         <div>
22             <input type="text" placeholder="Name....." onChange={(e)=>setName(e.target.value)} />
23             <input type="text" placeholder="Adress....." onChange={(e)=>setAddress(e.target.value)} />
24             <input type="email" placeholder="Email....." onChange={(e)=>setEmail(e.target.value)} />
25             <input type="text" placeholder="Contact....." onChange={(e)=>setContact(e.target.value)} />
26             <input type="number" placeholder="Age....." onChange={(e)=>setAge(e.target.value)} />
27
28             <button onClick={ addToJson } > Add User </button>
29
30         </div>
31     )
32 }
33
34 export default AddUsers
```

Axios : axios.delete()

```
1  import React, {useState} from 'react'
2  import axios from 'axios'
3  function DeleteUser() {
4      const [idUser, setIdUser] = useState()
5
6
7      const deleteFromJson = () => {
8          axios.delete(`http://localhost:3001/user/${idUser}`)
9      }
10
11     return (
12         <div>
13             <input type="number" placeholder="Delete....." onChange={(e)=>setIdUser(e.target.value)} />
14
15             <button onClick={ deleteFromJson } > Delete User </button>
16
17         </div>
18     )
19 }
20
21 export default DeleteUser;
```

Axios : axios.put()

```
1  import React, {useState} from 'react'
2  import axios from 'axios'
3
4  function UpdateUser() {
5      const [idUser, setIdUser] = useState()
6
7      const [name, setName] = useState("")
8      const [adress, setAdress] = useState("")
9      const [email, setEmail] = useState("")
10     const [contact, setContact] = useState("")
11     const [age, setAge] = useState("")
12
13     const updateJsonData = () => {
14         axios.put(`http://localhost:3001/user/${idUser}`, {
15             name: name,
16             adress: adress,
17             email:email,
18             contact: contact,
19             age:age
20         })
21     }
22     return (
23         <div>
24             <input type="text" placeholder="Name....." onChange={(e)=>setName(e.target.value)} />
25             <input type="text" placeholder="Adress....." onChange={(e)=>setAdress(e.target.value)} />
26             <input type="email" placeholder="Email....." onChange={(e)=>setEmail(e.target.value)} />
27             <input type="text" placeholder="Contact....." onChange={(e)=>setContact(e.target.value)} />
28             <input type="number" placeholder="Age....." onChange={(e)=>setAge(e.target.value)} />
29             <input type="number" placeholder="Update this user....." onChange={(e)=>setIdUser(e.target.value)} />
30
31             <button onClick={ updateJsonData } > Update User </button>
32
33         </div>
34     )
35 }
36 export default UpdateUser
37
```



React JS

TRAVAUX PRATIQUES

Contrôle continu

Exercice 1 :

On considère l'objet suivant :

```
produits = [ {id :11, nom : "clé USB" , prix : 70 , famille: "INFO"},  
             {id :22, nom : "Table" , prix : 250 , famille: "Mobilier"},  
             {id :33, nom : "Souris" , prix : 40 , famille: "INFO"},  
             {id :44, nom : "Chaise" , prix : 170 , famille: "Mobilier"},  
             {id :55, nom : "Pince" , prix : 35 , famille: "INFO"},  
             {id :66, nom : "Routeur" , prix : 300 , famille: "INFO"},,  
             .....  
             ]
```

- 1- Créer un fichier JSON nommé "db.json" dans le dossier src et remplie le avec les données de liste produits.
- 2-Ecrire le code JS qui retourne les produits de la famille INFO qui ont le prix supérieure à 80
- 3-Ecrire le code JS qui calcule la Moyenne des prix des produits
- 4-Ecrire le code JS qui détermine le nom du produit le plus cher
- 5-Ecrire le code JS qui crée un autre tableau produits2 contenant les éléments du tableau produits en supprimant le produit qui a le ID = 33

☺ ☺ Pour Q1 & Q2 & Q3 Créer un component qui gère les états et affiche le résultat

Dans package.json
Ajouter aux "scripts" :

```
"json-serve": "json-server --watch src/db.json --port 3001"
```

```
"scripts": {  
  "start": "react-scripts start",  
  "build": "react-scripts build",  
  "test": "react-scripts test",  
  "eject": "react-scripts eject",  
  "json-serve": "json-server --watch db.json --port 3001"  
},
```

```
C:\Users\younes\Desktop\react-0.14.3\redux>npm run json-serve  
  
> my-app@0.1.0 json-serve  
> json-server --watch db.json --port 3001
```

```
Loading db.json  
Done  
  
Resources  
http://localhost:3001/user  
http://localhost:3001/produits  
  
Home  
http://localhost:3001
```

localhost:3001/produits x React A
localhost:3001/produits
Gmail YouTube View site information F

```
[  
  {  
    "id": 11,  
    "nom": "clé USB",  
    "prix": 70,  
    "famille": "INFO"  
  },  
  {  
    "id": 22,  
    "nom": "table",  
    "prix": 250,  
    "famille": "Mobilier"  
  },  
  {  
    "id": 33,  
    "nom": "Souris",  
    "prix": 40,  
    "famille": "INFO"  
  },  
  {  
    "id": 44,  
    "nom": "Chaise",  
    "prix": 170,  
    "famille": "Mobilier"  
  },  
  {  
    "id": 55,  
    "nom": "Pince",  
    "prix": 35,  
    "famille": "INFO"  
  },  
  {  
    "id": 66,  
    "nom": "Routeur",  
    "prix": 300,  
    "famille": "INFO"  
  }  
]
```

```
"produits": [  
  {  
    "id": 11,  
    "nom": "clé USB",  
    "prix": 70,  
    "famille": "INFO"  
  },  
  {  
    "id": 22,  
    "nom": "table",  
    "prix": 250,  
    "famille": "Mobilier"  
  },  
  {  
    "id": 33,  
    "nom": "Souris",  
    "prix": 40,  
    "famille": "INFO"  
  },  
  {  
    "id": 44,  
    "nom": "Chaise",  
    "prix": 170,  
    "famille": "Mobilier"  
  },  
  {  
    "id": 55,  
    "nom": "Pince",  
    "prix": 35,  
    "famille": "INFO"  
  },  
  {  
    "id": 66,  
    "nom": "Routeur",  
    "prix": 300,  
    "famille": "INFO"  
  }  
]
```

1) Ecrire le code JS qui retourne les produits de la famille INFO qui ont le prix supérieure à 80

```
1 import React, {useState, useEffect} from 'react'
2 import axios from 'axios'
3
4 function Products() {
5   const [products, setProducts] = useState([])
6   useEffect(()=>{
7     axios.get('http://localhost:3001/produits').then(res=>{
8       setProducts(res.data)
9     })
10  },[])
11  const infoSup80 = products.filter((p)=>{
12    return p.famille == "INFO" && p.prix > 80
13  })
14  return (
15    <div>
16      {infoSup80.map(row => {
17        return(
18          <>
19            <p>{row.nom}</p>
20            <p>{row.prix}</p>
21            <p>{row.famille}</p>
22          </>
23        )
24      })}
25    </div>
26  )
27 }
28
29 export default Products
```

Routeur

300

INFO

2) Ecrire le code JS qui calcule la moyenne des prix des produits

```
const [moyenne , setMoyenne] = useState()

useEffect(()=>{
  let total = 0;
  products.map((product)=>{
    total += parseInt(product.prix);
  })
  let avg = total / products.length;
  setMoyenne(avg);
},[])
```

```
<h1>Moyenne des produits : {moyenne}</h1>
```

Moyenne des produits : 144.16666666666666

3) Ecrire le code JS qui détermine le nom du produit le plus cher

```
const [mostExpensiveProduct , setMostExpensiveProduct] = useState("")

useEffect(()=>{
  let found = products.find((elem) =>{
    return elem.prix == Math.max(...products.map(obj => obj.prix))
  })
  setMostExpensiveProduct(found.nom)
},[products])
```

```
<h1>Le nom de Produit le plus Cher est : {mostExpensiveProduct} </h1>
```

Le nom de Produit le plus Cher est : Routeur

- 4) Ecrire le code JS qui crée un autre tableau *produits2* contenant les éléments du tableau *produits* en supprimant le produit qui a le ID égal à 33

```
const idToRemove = 33;

const filteredProducts = products.filter((item) =>
  item.id !== idToRemove
);
console.log(filteredProducts);
```

```
▼ (5) [{...}, {...}, {...}, {...}, {...}] ⓘ
  ▶ 0: {id: 11, nom: 'clé USB', prix: 70, famille: 'INFO'}
  ▶ 1: {id: 22, nom: 'table', prix: 250, famille: 'Mobilier'}
  ▶ 2: {id: 44, nom: 'Chaise', prix: 170, famille: 'Mobilier'}
  ▶ 3: {id: 55, nom: 'Pince', prix: 35, famille: 'INFO'}
  ▶ 4: {id: 66, nom: 'Routeur', prix: 300, famille: 'INFO'}
  length: 5
```

Exercice II :

5) Créer un composant de type classe appelé *Inscription* qui contient une zone de texte pour saisir le mot de passe et une autre zone de texte pour confirmer le mot de passe et un bouton « Valider ».

Au clic sur le bouton Valider, on doit vérifier la saisie :

- Si les mots de passe saisis ne sont pas identiques, on affiche en rouge le message “Mot de passe saisi ne coïncide pas avec le mot de passe confirmé”
- Si les 2 mots de passe saisis sont identiques, on affiche en vert le message “Bienvenu”

Mot de passe :

Confirmer :

Mot de passe saisi ne coïncide pas avec le mot de passe confirmé

Mot de passe

Confirmez votre Mot de passe

Mot de passe saisie ne coincide pas avec le mot de passe confirmé

Mot de passe

Confirmez votre Mot de passe

Bienvenue

```
1 import React, { Component } from 'react'
2
3 export default class Inscription extends Component {
4
5   constructor(props){
6     super(props)
7     this.state = {password : "" , confirmPassword : "",message: ""}
8   }
9
10  validate = () => {
11
12    if(this.state.password === this.state.confirmPassword ){
13      document.getElementById("message").style.color = "green"
14      this.setState({message:"Bienvenue"})
15    }else{
16      document.getElementById("message").style.color = "red"
17      this.setState({message:"Mot de passe saisie ne coincide pas avec le mot de passe confirmé"})
18    }
19  }
20
21  render() {
22    return (
23      <div className="w-25 mx-auto mt-5">
24        <div className="form-floating mb-3">
25          <input type="text" className="form-control" onChange={(e)=>{this.setState({password:e.target.value})}} />
26          <label>Mot de passe</label>
27        </div>
28        <div className="form-floating mb-3">
29          <input type="text" className="form-control" onChange={(e)=>{this.setState({confirmPassword:e.target.value})}} />
30          <label>Confiermez votre Mot de passe</label>
31        </div>
32        <button className='btn btn-primary btn-outline-dark btn-lg d-grid gap-2 col-6 mx-auto' onClick={this.validate}>Valider</button>
33        <p id='message'>{this.state.message}</p>
34      </div>
35    )
36  }
37 }
```

6) Réécrire le même composant *Inscription* , mais de type fonction avec en plus la vérification si le mot de passe est « faible », « moyen » ou « fort » :

Après clic sur le bouton « Valider » , si les 2 mots de passe saisis sont identiques, on affiche le message :

- “Bienvenu, mot de passe faible ” si la taille du mot de passe est inférieure à 8
- “Bienvenu, mot de passe moyen ” si la taille du mot de passe est supérieure à 8
- “Bienvenu, mot de passe fort ” si la taille du mot de passe est supérieure à 8 et contient un mélange de lettres majuscules et minuscules.

Mot de passe
ofppt12

Confirmez votre Mot de passe
ofppt12

Valider

Bienvenue ,Mot de passe faible

Mot de passe
ofppt1234

Confirmez votre Mot de passe
ofppt1234

Valider

Bienvenue ,Mot de passe moyen

Mot de passe
ofPPt1234

Confirmez votre Mot de passe
ofPPt1234

Valider

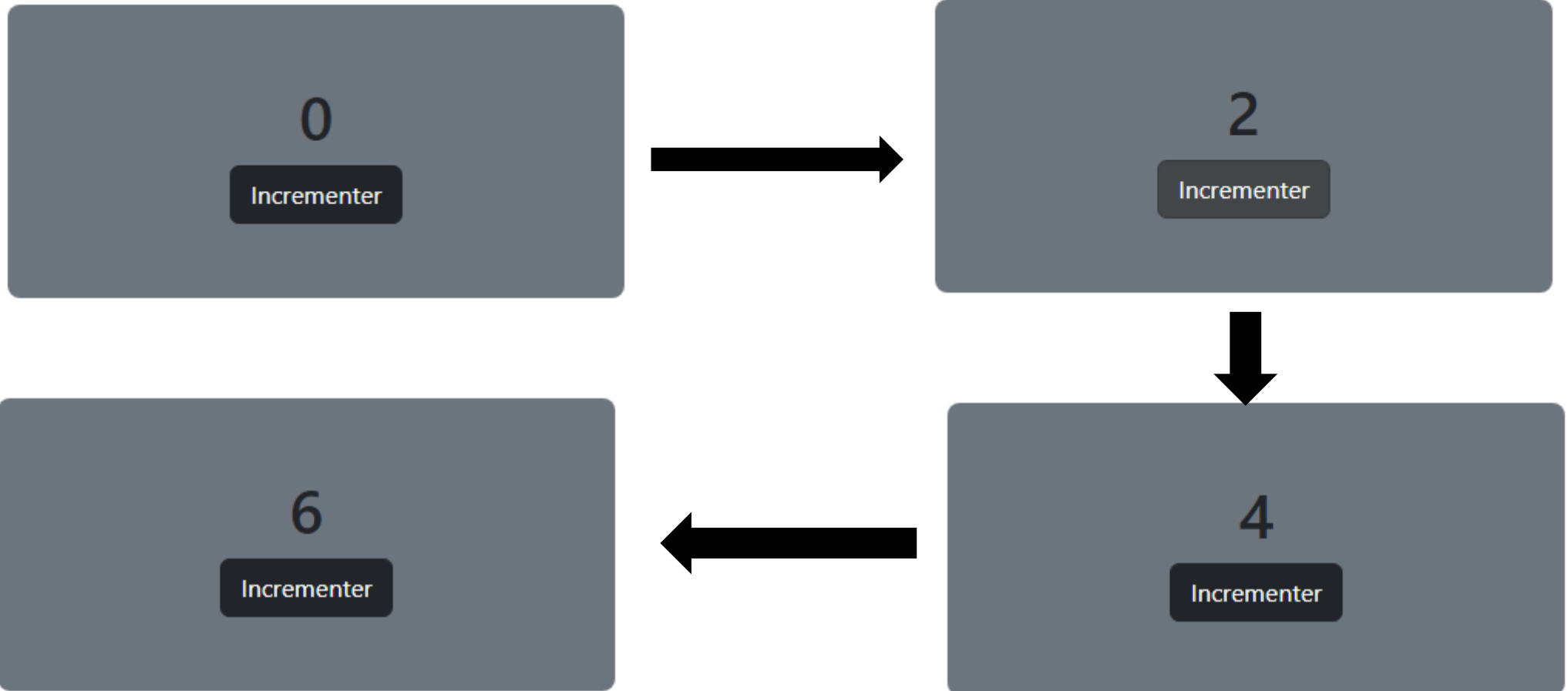
Bienvenue ,Mot de passe fort

```
1 import React, { useState } from 'react'
2
3 export default function InscriptionFunctionComponent() {
4
5     const [password, setPassword] = useState("");
6     const [confirmPassword, setConfirmPassword] = useState("");
7     const [message, setMessage] = useState('')
8     const [passLength, setPassLength] = useState("");
9
10    const validate = () => {
11        if(password === confirmPassword ){
12            document.getElementById("message").style.color = "green"
13            setMessage("Bienvenue")
14            if(password.length < 8) setPassLength(",Mot de passe faible");
15            else if (password.length >= 8 && password !== password.toLowerCase()) setPassLength(",Mot de passe fort")
16            else setPassLength(",Mot de passe moyen")
17
18        }else{
19            document.getElementById("message").style.color = "red"
20            setMessage("Mot de passe saisie ne coincide pas avec le mot de passe confirmé")
21        }
22    }
23    console.log(password !== password.toLowerCase());
24    return (
25        <div className="w-25 mx-auto mt-5">
26            <div className="form-floating mb-3">
27                <input type="text" className="form-control" onChange={(e)=>{setPassword(e.target.value)}} />
28                <label>Mot de passe</label>
29            </div>
30            <div className="form-floating mb-3">
31                <input type="text" className="form-control" onChange={(e)=>{setConfirmPassword(e.target.value)}} />
32                <label>Confiermez votre Mot de passe</label>
33            </div>
34            <button className='btn btn-primary btn-outline-dark btn-lg d-grid gap-2 col-6 mx-auto' onClick={ validate }>Valider</button>
35            <p id='message'>{message} {passLength}</p>
36        </div>
37    )
38
39 }
```

Exercice II :

On veut réaliser un composant reactJs appelé *Compteur*. Ce composant contient un élément `h2` qui affiche la valeur du compteur et un bouton « *Incrementer* » qui permet, après clic, d'augmenter la valeur du compteur de 2.

7) Ecrire le code du composant *Compteur* de type classe



```
1 import React, { Component } from 'react'
2
3 export default class Compteur extends Component {
4   constructor(props){
5     super(props)
6     this.state = {
7       compteur : 0
8     }
9   }
10
11   incerement = () => {
12     this.setState({
13       compteur: this.state.compteur+2
14     })
15   }
16   render() {
17     return (
18       <div className='p-5 bg-secondary mt-5 w-25 mx-auto text-center rounded-3'>
19         <h1>{this.state.compteur}</h1>
20         <button className='btn btn-dark text-light' onClick={this.incerement}>Incrementer</button>
21       </div>
22     )
23   }
24 }
```

8) Réécrire le composant *Compteur*, mais de type fonction avec en plus :

- La valeur de l'incrémentation du compteur est aléatoire (à la place de 2)
- La valeur du compteur est affichée avec une couleur qui est passée en props à ce composant
- Quand la valeur du compteur devient égale à 50, la couleur devient rouge

9

Incrémenter

20

Incrémenter

29

Incrémenter

53

Incrémenter

46

Incrémenter

39

Incrémenter

```
1 import { useEffect,useState } from 'react'
2
3 function CompteurFunction(props) {
4   const [compteur, setCompteur] = useState(0)
5
6   const increment = () => {
7     setCompteur(compteur + Math.floor(Math.random() * 21)) // returns a random integer from 0 to 20
8   }
9   useEffect(()=> {
10    if(compteur < 50) document.getElementsByTagName('h1')[0].style.color = props.couleur;
11    else document.getElementsByTagName('h1')[0].style.color = "red";
12  },[compteur])
13  return (
14    <div className='p-5 bg-secondary mt-5 w-25 mx-auto text-center rounded-3'>
15      <h1>{compteur}</h1>
16      <button className='btn btn-dark text-light' onClick={increment}>Incrementer</button>
17    </div>
18  )
19 }
20
21 export default CompteurFunction
```