

TP React 2

JSX et composants

Objectifs pédagogiques

À la fin de ce TP, vous serez capable de :

- comprendre la notion de composant React
- créer plusieurs composants
- importer et utiliser des composants
- structurer une interface React simple

Rappel

Dans le TP précédent, vous avez :

- créé une application React
- modifié le composant principal App
- découvert la syntaxe JSX

Dans ce TP, nous allons apprendre à **découper une interface en composants**.

1 Principe des composants

Un composant React est une fonction JavaScript qui retourne du JSX.

Une application React est composée d'un ensemble de composants assemblés entre eux.

Chaque composant a un rôle précis :

- affichage d'un en-tête
- affichage d'un pied de page
- affichage d'un contenu spécifique

2 Creation d'un premier composant

Dans le dossier `src`, crezez un dossier nomme `components`.

Dans ce dossier, crezez le fichier `Header.jsx` :

```
function Header() {  
  return (  
    <header>  
      <h1>Mon site React</h1>  
    </header>  
  );  
}  
  
export default Header;
```

3 Creation d'un second composant

Toujours dans le dossier `components`, crezez le fichier `Footer.jsx` :

```
function Footer() {  
  return (  
    <footer>  
      <p>&copy; 2026 - Mon site React</p>  
    </footer>  
  );  
}  
  
export default Footer;
```

4 Utilisation des composants dans App

Ouvrez le fichier `src/App.jsx` et modifiez-le :

```
import Header from "./components/Header";  
import Footer from "./components/Footer";  
  
function App() {  
  return (  
    <div>  
      <Header />  
      <main>  
        <p>Bienvenue sur mon application React.</p>  
      </main>  
      <Footer />  
    </div>  
  );  
}  
  
export default App;
```

Les composants sont importes puis utilises comme des balises HTML personnalisees.

5 Structure du projet

Votre projet doit maintenant être organisé ainsi :

```
src/
  components/
    Header.jsx
    Footer.jsx
  App.jsx
  main.jsx
```

6 Règles importantes sur les composants

- Le nom d'un composant commence par une majuscule
- Un composant retourne un seul élément parent
- Un composant a une responsabilité précise

Un composant = une responsabilité.

7 JSX et expressions JavaScript

Dans JSX, il est possible d'utiliser des expressions JavaScript à l'aide des accolades.

Exemple dans `App.jsx` :

```
function App() {
  const nom = "Étudiant";

  return (
    <div>
      <p>Bonjour {nom}</p>
    </div>
  );
}

export default App;
```

Conclusion

Dans ce TP, vous avez appris à :

- créer et organiser des composants
- importer et utiliser des composants
- structurer une interface React
- utiliser JSX avec des expressions JavaScript

Vous savez maintenant découper une interface React en composants.