

# Hello React

## Structure des fichiers de l'application

Le répertoire `/var/www`, où l'application a été créée, contient deux sous-répertoires:

- **public**: contient la page elle-même et les fichiers qui peuvent faire l'objet d'une requête HTTP directe;
- **src**: contient les fichiers de ressources de l'application, tant les composants javascript que les feuilles de style, images ou autres.

### index.html

Le fichier qui contient le code HTML de l'application est `/var/www/public/index.html`. Ce fichier a la structure habituelle d'un fichier HTML (éléments HEAD, BODY, DIV etc.). Remarquez l'élément

ayant l'identifiant `root` : il constitue la racine de l'application; tout ce qu'il contient sera généré à partir de React.

### index.js

Ce fichier est le point d'entrée de l'application. Il contient le code qui génère la page.

La première ligne met dans une variable (nommée `root`) l'élément `<div>` que nous avons vu dans le fichier `index.html`:

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

Dans la deuxième ligne, on appelle la fonction `render()` avec (entre autres) l'élément `<App />` comme paramètre:

```
root.render(  
  <React.StrictMode>  
    <App />  
  </React.StrictMode>  
);
```

Ceci a pour effet d'appeler la fonction `App()` du fichier `App.js`. C'est cette fonction qui contient le code HTML qui sera affiché dans la page.

### App.js

Avec React, les page HTML sont construites grâce à des appels de fonction: les valeurs retournées par les fonctions seront rattachées aux différents éléments du DOM dans la page.

La fonction `App()` retourne les éléments qui seront immédiatement rattachés au DIV racine:

```
function App() {  
  return (  
    <div className="App">  
      <header className="App-header">  
        <img src={logo} className="App-logo" alt="logo" />  
        <p>  
          Edit <code>src/App.js</code> and save to reload.  
        </p>  
        <a  
          className="App-link"  
          href="https://reactjs.org"  
          target="_blank"  
          rel="noopener noreferrer"  
        >  
          Learn React  
        </a>  
      </header>  
    </div>  
  );  
}
```

Quelques remarques au sujet de l'exemple précédent:

- Le code est similaire à du javascript
- Le HTML retourné par la fonction n'est pas entre guillemets
- L'attribut HTML class s'appelle ici className

Ces particularités viennent du fait que le langage utilisé dans cette page n'est pas exactement du javascript, mais une extension de javascript qu'on nomme JSX.

Une introduction à JSX sera faite dans la section suivante.