

React

Installation

[NodeJS](#) requis

Standalone project

Création du projet :

```
npx create-react-app my-app
```

Eventuellement avec le support typeScript :

```
npx create-react-app my-app --template typescript
```

Démarrage du serveur

```
cd my-app  
npx start
```

Composants

En React, tout est composant, et les composants peuvent être créés à partir de fonctions (JS), ou de classes (ES6 ou TypeScript).

</WRAP>

==== Fonctions ====

Un composant peut être créé à partir d'une fonction :

```
<sxh javascript;title:hello.js>  
function Hello() {  
  return (  
    <div>  
      <h1>props.message</h1>  
    </div>  
  );  
}  
  
export default Hello;
```

Classe

Un composant peut être créé à partir d'une classe ES6 :

```
import React from "react";

class Hello extends React.Component {
  render() {
    return <h1>{this.props.message}</h1>;
  }
}
export default Hello;
```

Classe TypeScript

Un composant peut être créé à partir d'une classe TypeScript:

```
import React from "react";

export default class Hello extends React.Component<{message: string}> {
  render() {
    return (
      <h1>{this.props.message}</h1>
    );
  }
}
```

La classe Hello déclare explicitement une propriété de nom **message** de type **string**.

Utilisation

Utilisation de **Hello** dans l'**App**, et initialisation en JSX de la prop **message** avec l'attribut **message** :

```
import Hello from './hello';

function App() {
  return (
    <div className="App">
      <Hello message="Hello World!"/>
    </div>
  );
}

export default App;
```

Props et state

Sur un composant React :

- Les propriétés sont accessibles avec l'objet **props** passé en paramètre (du constructeur ou de la fonction)
- Ces propriétés sont initialisées en JSX/HTML via les attributs de l'élément.

- Elles sont immutables (en lecture seule).
- Il est nécessaire d'utiliser l'objet **state** pour modifier l'état d'un composant.

Exemple classe TypeScript

```
import React from "react";

export default class Hello extends React.Component<{message:string}, {msg: string}> {
    constructor(props: {message:string}) {
        super(props);
        this.state = {msg: props.message};
    }
    handleChange = (event: React.ChangeEvent<HTMLInputElement>) => {
        this.setState({msg: event.target.value})
    }
    render() {
        return (
            <div>
                <h1>{this.state.msg}</h1>
                <input value={this.state.msg} onChange={this.handleChange}>
            </div>
        );
    }
}
```

Exemple fonction JavaScript

```
import {useState} from "react";

export default function Olla(props){
    const [message, setMessage] = useState(props.message);
    const handleChange = (e) => {
        setMessage(e.target.value);
    }
    return (
        <>
            <h1>{message}</h1>
            <input type="text" value={message} onChange={handleChange}>
    );
} </sxh>
```

From:

<http://slamwiki2.kobject.net/> - **SlamWiki 2.1**



Permanent link:

<http://slamwiki2.kobject.net/web/framework/react?rev=1679009179>

Last update: **2023/03/17 00:26**