

FRONT END DEV

CHOISIR UNE OPTION

Dans le cadre du cours, nous allons nous appuyer sur:

- Langage: Typescript (verbeux et mauvaise inférence, mais nécessaire à maîtriser en 2021)
- Frameworks : React
 - Seule lib front 'mainstream' à avoir une approche expressive !
 - Un composant React = Une fonction qui prend en paramètre un objet props et retourne un objet de type React.Element
 - Un composant monté dans le DOM correspond donc à un objet instancié par un pattern Factory
 - Vous pouvez utiliser le DSL JSX pour décrire les Elements

```
import React from "react";  
  
const Welcome = ({ name } /* destructuring de props */) => <h1>Hello, {name} </h1>;
```

NE CREEZ PAS DE COMPOSANTS AVEC DES CLASS, N'UTILISEZ PAS LES LIFECYCLES

POINT D'ATTENTION SUR REACT

C'EST PAS SIMPLE DE DÉBUTER REACT EN 2021 ... SURTOUT QUE REACTJS.ORG EST DATÉ

2014

2016



2017

2019

```
var Component2014 = React.createClass({
  render(props){
    ...
  }
});
```

```
const StatelessComponent2016 = (props) => <div>...</div>;
class StatefulComponent extends React.Component {
  constructor(props) {
    super(props);
    ...
  }
  render(props) {
    ...
  }
}
```

```
class StatelessComponent2017 extends React.PureComponent {
  render(props) {
    ...
  }
}
class StatefulComponent extends React.Component {
  constructor(props) {
    super(props);
    ...
  }
  render(props) {
    ...
  }
}
```

```
const futureComponent = (props) => {
  ...
  return <div>...</div>
}
```

+ Mixins
+ Flux

+ HOC
+ REDUX

+ API changes

+ Render props
+ Context

+ API changes

+ Hook
+ Context

Les approches >2016 fonctionnent encore

Autant de « visions » que de projets ... c'est normal react est une lib qui laisse beaucoup de liberté dans l'architecture du projet