



Internationalization

Animé par Mazen Gharbi

Principe

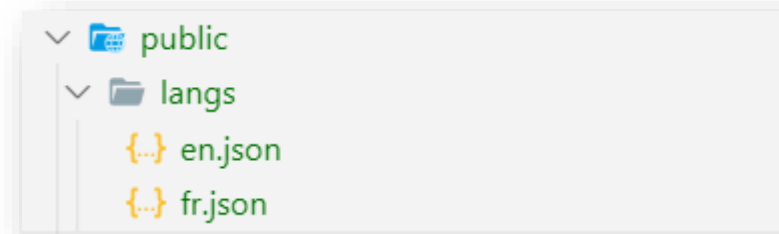
- ▷ Pour optimiser l'expérience utilisateur, il est souvent nécessaire de proposer différentes langues pour votre application WEB
- ▷ Il va être nécessaire de faire cela de façon **fluide**
- ▷ La communauté React propose plusieurs outils

```
npm install react-i18next@legacy i18next --save
```



Fonctionnement

- ▷ Nos traductions vont être stockées dans des fichiers .json différents
 - › Chacun représentant une langue différente



- ▷ On pourra importer ces fichiers par la suite

Fichiers de traduction

▷ Voici ce que contient nos .json :

```
{  
  "home.title": "ReactJS",  
  "home.description": "Bienvenue dans la meilleure formation du siècle"  
}
```

fr.json

```
{  
  "home.title": "ReactJS",  
  "home.description": "Welcome to the best course in the world !"  
}
```

en.json

Configuration

► La prochaine étape va consister à configurer notre multi-langue en implémentant le plugin

```
import i18n from "i18next";
import { reactI18nextModule } from "react-i18next";

import langFR from '../public/langs/fr.json';
import langEN from '../public/langs/en.json';

const resources = {
  fr: {
    translation: langFR
  },
  en: {
    translation: langEN
  }
};
```

config/i18n.js

Configuration

config/i18n.js (suite)

```
i18n
  .use(reactI18nextModule)
  .init({
    resources,
    lng: "fr",

    interpolation: {
      escapeValue: false // On échappe rien, React protège déjà contre le XSS
    }
  });

export default i18n;
```

index.js

```
import './config/i18n';

ReactDOM.render(<App />, document.getElementById('root'));
```

Et enfin

```
import React from 'react';
import './App.css';
import { withNamespaces } from 'react-i18next';

function App(props) {
  const { t } = props;

  return (
    <div className='App'>
      <h1>{t('home.title')}</h1>

      <p>{t('home.description')}</p>
    </div>
  );
}

export default withNamespaces()(App);
```

Changement dynamique

► L'utilisateur peut évidemment changer la langue comme bon lui semble

Français Anglais

ReactJS

Welcome to the best course in the world !

```
<div>
  <button onClick={() => changeLanguage('fr')}>Français</button>
  <button onClick={() => changeLanguage('en')}>Anglais</button>
</div>
```

Français Anglais

ReactJS

Bienvenue dans la meilleure formation du siècle

```
const changeLanguage = (lng) => {
  i18n.changeLanguage(lng);
}
```


Gestion de catégories

▷ On peut définir des groupes de traduction :

```
{  
  "home": {  
    "title": "ReactJS",  
    "description": "Bienvenue dans la meilleure formation du siècle"  
  }  
}
```



```
<h1>{t('home.title')}</h1>  
<p>{t('home.description')}</p>
```


Pluralization

► Parfois, il peut s'avérer utile d'afficher une chaîne ou une autre en fonction d'une variable

› « Il y a 1 utilisateur connecté » / « Il y a 20 utilisateurs connectés »

```
{  
  "home": {  
    "title": "ReactJS",  
    "description": "Bienvenue dans la meilleure formation du siècle",  
    "nbUsers": "Il y a {{nb}} utilisateurs connectés"  
  }  
}
```

Variable attendue !



`<p>{t('home.nbUsers', { nb: 10 })}</p>`  Il y a 10 utilisateurs connectés

Plural

▷ Un petit problème :

Bienvenue dans la meilleure formation du siècle

Il y a 0 utilisateurs connectés

```
"home": {  
  "title": "ReactJS",  
  "description": "Bienvenue dans la meilleure formation du siècle",  
  "nbUsers_plural": "Il y a {{nb}} utilisateurs connectés",  
  "nbUsers": "Aucun utilisateur connecté"  
}
```

Bienvenue dans la meilleure formation du siècle

Aucun utilisateur connecté

Questions ?