

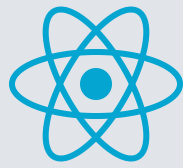
cours **React** – J5

IG2I 2021



CE QU'IL FALLAIT RETENIR

- `<div><MonSousComposant /></div>`
- `<DonEladio drink={gus.getPoison()} /> / this.props.drink`
- `FunctionComponent = (props) => lol`
- `videoPlayer = React.createRef()`



React

1. Rappels
2. Syntaxes ES6+
3. Premiers pas avec React
4. Les composants
- 5. Formulaires et AJAX**
6. Hooks & React Router

5. FORMULAIRES ET AJAX

- Gestion des formulaires
- AJAX

FORMULAIRES

Deux approches possibles:

- composants non contrôlés
- composants contrôlés

COMPOSANTS NON CONTRÔLÉS^{2.4}

1. Une ref par champ de saisie
2. Récupération de la valeur via l'API DOM (input.value)

- 😊 Facile à mettre en place
- 😓 pas de contrôle sur la saisie

COMPOSANTS CONTRÔLÉS ^{2.5}

1. Écoute du changement de valeur
2. Stockage de la valeur saisie dans le state
3. Dans le render : affichage de la valeur dans le champ

- 😊 Contrôle total sur les valeurs des champs
- 😊 Permet de vérifier les valeurs au fur et à mesure
- 😊 Permet de modifier les valeurs avant de les afficher
- 😱😱😱 Complexe à mettre en place

COMPOSANTS NON CONTRÔLÉS ^{2.6}

```
export default class UserForm extends React.Component {
  firstNameInput = React.createRef();
  lastNameInput = React.createRef();

  handleSubmit(event) {
    event.preventDefault();
    api.addUser({
      lastName: this.lastNameInput.current.value,
      firstName: this.firstNameInput.current.value
    });
  }

  render() {
    return (
      <form onSubmit={event => this.handleSubmit(event)}>
        <input
          type="text"
          name="firstName"
          ref={ this.firstNameInput } />
        <input
          type="text"
          name="lastName"
          ref={ this.lastNameInput } />
        <input type="submit" value="Valider" />
      </form>
    );
  }
}
```

Notes :

Cette approche qui est la plus simple à mettre en place convient pour les formulaires simple.

Il est possible de définir des valeurs par défaut via la prop "defaultValue" qui seront utilisées lors du premier rendu du composant. Ensuite il ne sera plus possible de modifier le contenu du formulaire via la méthode "render".

COMPOSANTS CONTRÔLÉS ^{2.7}

```
export default class UserForm extends React.Component {
  state = {firstName: '', lastName: ''};

  handleSubmit(event) {
    event.preventDefault();
    // les données sont déjà dans le state
    api.addUser(this.state);
    // On efface le state, ce qui vide le formulaire :
    this.setState({ firstName: '', lastName: '' });
  }
  handleFirstNameChange(event) {
    this.setState({ firstName: event.target.value });
  }
  handleLastNameChange(event) {
    this.setState({ lastName: event.target.value } );
  }
  render() {
    return (
      <form onSubmit={event => this.handleSubmit(event)}>
        <input
          type="text"
          name="firstName"
          onChange={e => this.handleFirstNameChange(e)}
          value={this.state.firstName}
        />
        <input
          type="text"
          name="lastName"
          onChange={e => this.handleLastNameChange(e)}
          value={this.state.lastName}
        />
        <input type="submit" value="Valider" />
      </form>
    );
  }
}
```

Notes :

Cette méthode présente l'avantage d'avoir le contrôle total du formulaire. En effet, puisque cette fois on utilise l'attribut "value" et non "defaultValue" il nous est possible de mettre à jour la valeur des champs via le render.

Cependant on peut constater que par rapport au formulaire précédent, elle requiert d'écouter le changement sur chacun des champs du formulaire et nécessite donc une quantité de code beaucoup plus importante qui est plus ou moins proportionnelle au nombre de champs que l'on a dans le formulaire.

redux-form vs formik vs react-final-form vs react-hook-form

Enter an npm package...

redux-form x

formik x

react-final-form x

react-hook-form x

+ react-redux-form

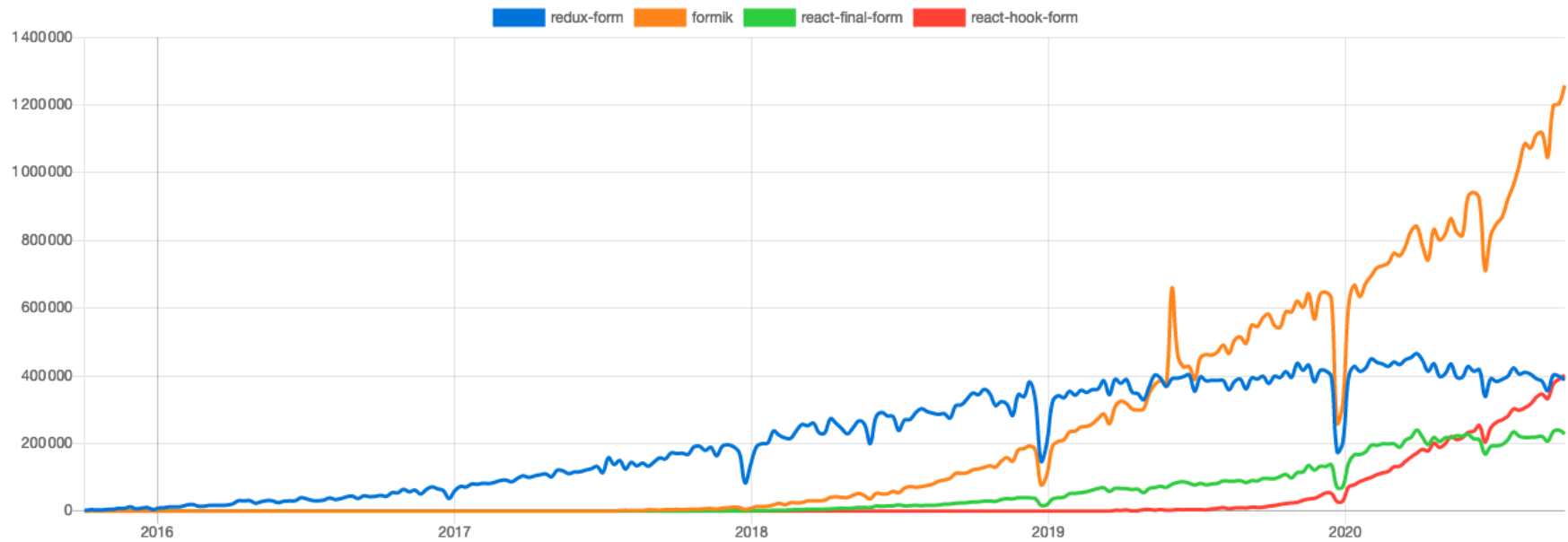
+ react-form

+ final-form

+ informed

+ formsy

Downloads in past 5 Years



Stats

	stars 🌟	issues ⚠️	updated 🔧	created 🍷	size 📦
redux-form					minzipped size 26.4 KB

Notes :

Comparaison des principales librairies de gestion des formulaires

<https://www.npmtrends.com/redux-form-vs-formik-vs-react-final-form-vs-react-hook-form>

4. FORMULAIRES ET AJAX

- Gestion des formulaires
- **AJAX**

AJAX

- = Asynchronous JavaScript And XML
- Appel à un script distant depuis JavaScript sans rechargement de page
- pas limité à XML (plutôt JSON en général)

XMLHttpRequest

```
const request = new XMLHttpRequest();
request.open('GET', 'api/users');
request.onload = event => {
  console.log( request.responseText );
}
request.send();
```

Notes :

XMLHttpRequest est une classe JS de base assez ancienne qui permet de faire des appels ajax.

On peut également faire des requêtes POST avec XMLHttpRequest :

```
const request = new XMLHttpRequest();
request.open('POST', 'api/users');
request.setRequestHeader('Content-type',
  'application/x-www-form-urlencoded');
request.onload = event => {
  console.log( request.responseText );
}
request.send('firstName=Jon&lastName=Targaryen');
```

Exemple : <http://codepen.io/kumquats/pen/QdjYyw?editors=0010>

FETCH

- nouvelle syntaxe plus simple
- support des Promises

```
fetch( 'api/users/todd' )  
  .then( response => response.text() )  
  .then( responseText => console.log(responseText) );
```

Notes :

Pour gérer les erreurs avec fetch, on a plusieurs outils à notre disposition :

- response.ok : true/false
- response.status : code HTTP (200, 400, 404, 500, ...)
- response.statusText : 'OK', 'Bad Request', 'Not found', etc.

```
fetch( 'api/users/todd' )  
  .then( response => {  
    if ( ! response.ok ) {  
      throw new Error( response.statusText );  
    }  
    return response.text();  
  })  
  .then( responseText => console.log(responseText) )  
  .catch( error => alert( error.message ) )
```

Fetch

A modern replacement for XMLHttpRequest.

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	89	87	89	13.1	13.7			4.4	12.0
10	90	88	90	14	14.4			4.4.4	13.0
11	91	89	91	14.1	14.6	all	91	91	14.0
		90	92	15					


✓ ✗ Partial Support

Global: 95.51% + 0.05% = 95.56%

Notes :

<https://caniuse.com/#feat=fetch>

2 . 15

 Why GitHub? ▾ Enterprise Explore ▾ Marketplace Pricing ▾

Search

Sign in

Sign up

github / fetch

Watch469

Star21,451

Fork1,954

<> Code

Issues8

Pull requests3

Insights

A window.fetch JavaScript polyfill. <http://github.github.io/fetch/>

polyfill

fetch

javascript

promise

587 commits

4 branches

28 releases

51 contributors


MIT

Branch: master ▾

New pull request

Find file

Clone or download ▾

 mislav Merge pull request #684 from justjavac/master ...


Latest commit 3674c98 on 2 Jan

test	field name can not by empty	2 months ago
.eslintrc.json	Compatibility with newer eslint-plugin-github	3 months ago
.gitignore	Build UMD dist file with Rollup	10 months ago
.travis.yml	Update npm API key	6 months ago
CODE_OF_CONDUCT.md	Fix code of conduct contact email	6 months ago
CONTRIBUTING.md	Clarify what parts of the standard we don't want to implement	5 months ago
LICENSE	2016	3 years ago
Makefile	Add flow definitions	6 months ago
README.md	Add note about UMD dist file	5 months ago
bower.json	Remove Bower dependency in development	2 years ago

Notes :

Ce polyfill permet d'utiliser l'api fetch sur Internet Explorer :

<https://github.com/github/fetch>

 Search

LOG INSIGN UP219

84How to pronounce "JSON"?CLOSE

↑84↓

Posted by u/rodrigo-silveira 5 years ago

How to pronounce "JSON"?

I've heard people pronounce it so many ways, it's sometimes almost difficult to know what they're talking about. According to Douglas Crockford, however, this is how it's done:
<https://www.youtube.com/watch?v=zhVdWQWKRqM>

(PS: Crockford is the guy behind JSON, so whatever he says is definite and absolute truth ⇒)

188 CommentsShareSaveHideReport81% Upvoted

This thread is archived

New comments cannot be posted and votes cannot be cast

SORT BY

Best

↑ChaseMoskal 186 points · 5 years ago

↓
"Jay-sawn"

↑atomic1fire 9 points · 5 years ago

↓
derulooooo

Continue this thread →

↑papassinqueso 3 points · 5 years ago

↓
Yai-son or jotason, more locally

↑kylev 10 points · 5 years ago

↓
Yup. The end is like the O in "on". I have yet to annoy anyone with the name Jason since adopting this pronunciation.

Continue this thread →

ⓘ [deleted] 21 points · 5 years ago More than 1 child

↑strHungarianNotation 9 points · 5 years ago

↓
Jay-san like a mr. Jay from Japan :)

↑hugith 2 points · 5 years ago · edited 5 years ago

↓
[As an Icelander, thank you!](#)

↑churro89 4 points · 5 years ago

JS

r/javascript

All about the JavaScript programming language!

1.2mMembers815Online

Created 25 janv. 2008

JOIN

Help

Reddit App

Reddit Coins

Reddit Premium

Reddit Gifts

About

Careers

Press

Advertise

Blog

Terms

Content Policy

Privacy Policy

Mod Policy

Reddit Inc © 2020. All rights reserved

BACK TO TOP

Notes :

Comment on prononce "JSON" au fait ?

La réponse sur reddit (en fait sur youtube) par le créateur de la syntaxe :

https://www.reddit.com/r/javascript/comments/2mapkc/how_to_pronounce_json/

<https://www.youtube.com/watch?v=zhVdWQWKRqM>

TLDR : on prononce comme on veut, mais lui il dit "DJAYSSONNE" :)

FETCH & JSON

response.text() + JSON.parse() **VS** response.json()

```
fetch('api/users/todd')
  .then( response => response.text() )
  .then( responseText => JSON.parse(responseText) )
  .then( todd => console.log(todd.treasure) )
```

```
fetch('api/users/todd')
  .then( response => response.json() )
  .then( todd => console.log(todd.treasure) )
```