eBook Gratuit

APPRENEZ underscore.js

eBook gratuit non affilié créé à partir des **contributeurs de Stack Overflow.**

#underscor

e.js

Table des matières

À propos

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: underscore-js

It is an unofficial and free underscore.js ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official underscore.js.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapitre 1: Démarrer avec underscore.js

Remarques

Cette section fournit une vue d'ensemble de ce que underscore.js est et pourquoi un développeur peut vouloir l'utiliser.

Il devrait également mentionner tous les grands sujets dans underscore.js, et établir un lien avec les sujets connexes. La documentation de underscore.js étant nouvelle, vous devrez peut-être créer des versions initiales de ces rubriques connexes.

Examples

Installation ou configuration

Underscore est une bibliothèque d'utilitaires de programmation fonctionnelle open source pour JavaScript. Underscore fournit de nombreuses fonctions utiles pour travailler avec des tableaux ou des collections d'objets JavaScript, y compris le filtrage, le tri et les requêtes.

Node.js

Assurez-vous d'avoir installé node et npm, puis tapez la commande suivante

```
npm install underscore
```

Tonnelle

Assurez-vous que node, npm et bower sont installés, puis tapez la commande suivante

```
bower install underscore
```

NuGet

```
Install-Package underscore.js
```

Navigateur

Téléchargez la version de développement ou la version de production et liez le script dans votre fichier HTML à l'aide de l'élément de `script` .

```
<script src="js/underscore/underscore.js"></script>
```

Lire Démarrer avec underscore.js en ligne: https://riptutorial.com/fr/underscore-js/topic/7688/demarrer-avec-underscore-js

Chapitre 2: Collections

Examples

carte

La fonction `.map` accepte un tableau et une fonction iteratee, l'iteratee produit une copie transformée de chaque objet tableau.

La fonction iteratee fournit 3 arguments

1. `item` - L'objet itéré actuel
2. `i` - L'index de l'objet itéré
3. `list` - Une référence au tableau / liste d'origine

Le nouveau tableau aura la même longueur que l'ancien tableau mais contiendra les objets transformés

Exemple:

```
_.map([1, 2, 3, 4], function(item, i, list) {
    return (item*item);
});
// [1, 4, 9, 16]
```

Une manière plus concise d'écrire l'exemple ci-dessus en utilisant ES6 serait

```
_.map([1, 2, 3, 4], (item, i, list) => {
    return (item*item);
});
```

ou en utilisant une expression lambda en ligne

```
_.map([1, 2, 3, 4], (item, i, list) => (item*item));
```

Map est également utile lorsque vous souhaitez extraire des propriétés d'objets et en créer un tableau

Exemple:

```
let people = [{name: 'he-man', age: 22}, {name: 'man-at-arms', age: 44}];

_.map(people, function(item) {
    return item.name;
});
// ['he-man', 'man-at-arms']
```

chaque

La fonction `_.each` accepte un tableau ou un objet, une fonction iteratee et un objet `context` optionnel, la fonction iteratee est invoquée une fois pour chaque élément du tableau La fonction iteratee fournit 3 arguments

```
item - The current iterated object (or value if an object was passed)
i - The index of the iterated object (or key if an object was passed)
list - A reference to the original array/list (the object that was passed)
```

Exemple 1:

```
_.each(["hello", "underscore"], function(item, i, list) {
    alert(item)
});
```

L'exemple ci-dessus montrera 2 alertes, la première avec les mots "hello" et la seconde pour "world".

Exemple 2:

```
_.each({one: 1, two: 2, three: 3}, (value, key, object) =>
    console.log(JSON.stringify(object));
);
```

Cet exemple enregistre trois fois une version de l'objet

`.each` est une opération de terminal, et contrairement aux autres fonctions intermédiaires (map, pluck, valeurs, etc.), vous n'avez pas besoin de retourner dans le corps de la fonction iteratee.

Lire Collections en ligne: https://riptutorial.com/fr/underscore-js/topic/7695/collections

Crédits

S. No	Chapitres	Contributeurs
1	Démarrer avec underscore.js	Ashiquzzaman, Community, svarog
2	Collections	svarog