EBook Gratis

APRENDIZAJE underscore.js

Free unaffiliated eBook created from
Stack Overflow contributors.

#underscor

e.js

Tabla de contenido

Acerca de

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

Capítulo 1: Comenzando con underscore.js

Observaciones

Esta sección proporciona una descripción general de qué es underscore.js y por qué un desarrollador puede querer usarlo.

También debe mencionar cualquier tema grande en underscore.js, y vincular a los temas relacionados. Dado que la Documentación para underscore.js es nueva, es posible que deba crear versiones iniciales de esos temas relacionados.

Examples

Instalación o configuración

Underscore es una biblioteca de utilidad de programación funcional de código abierto para JavaScript. Underscore proporciona muchas funciones útiles para trabajar con matrices o colecciones de objetos de JavaScript, incluido el filtrado, la clasificación y la consulta.

Node.js

Asegúrese de tener nodos y npm instalados, luego escriba el siguiente comando

```
npm install underscore
```

Cenador

Asegúrese de tener nodos, npm y bower instalados, luego escriba el siguiente comando

```
bower install underscore
```

NuGet

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

Navegador

Descargue la versión de desarrollo o la versión de producción y vincule el script dentro de su archivo HTML utilizando el elemento `script` .

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

Lea Comenzando con underscore.js en línea: https://riptutorial.com/es/underscore-js/topic/7688/comenzando-con-underscore-js

Capítulo 2: Colecciones

Examples

mapa

La función `.map` acepta una matriz y una función iterada, la iterada produce una copia transformada de cada objeto de matriz.

La función iteratee proporciona 3 argumentos.

1. `item` - El objeto iterado actual
2. `i` - El índice del objeto iterado
3. `list` - Una referencia a la matriz / lista original

La nueva matriz será de la misma longitud que la matriz anterior pero contendrá los objetos transformados

Ejemplo:

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

Una forma más concisa de escribir el ejemplo anterior utilizando ES6 sería

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

o usando una expresión lambda en línea

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

El mapa también es útil cuando quieres arrancar las propiedades de los objetos y hacer una matriz de ellos

Ejemplo:

```
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']
```

cada

La función `_.each` acepta una matriz o un objeto, una función iterada y un objeto de `context` opcional, la función iteratee se invoca una vez y en orden para cada elemento de la matriz. La función iteratee proporciona 3 argumentos

```
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)
```

Ejemplo 1:

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

El ejemplo anterior mostrará 2 alertas, la primera con las palabras "hola" y la segunda para "mundo".

Ejemplo 2:

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

Este ejemplo registrará una versión stringificada del objeto 3 veces.

`.each` es una operación de terminal, y a diferencia de otras funciones intermedias (mapa, desplume, valores, etc.) no es necesario que regrese dentro del cuerpo de la función iterada.

Lea Colecciones en línea: https://riptutorial.com/es/underscore-js/topic/7695/colecciones

Creditos

S. No	Capítulos	Contributors
1	Comenzando con underscore.js	Ashiquzzaman, Community, svarog
2	Colecciones	svarog