

 무료 전자 책

배우기

underscore.js

Free unaffiliated eBook created from
Stack Overflow contributors.

#underscor

e.js

.....	1
1: underscore.js	2
.....	2
Examples.....	2
.....	2
2:	3
Examples.....	3
.....	3
.....	3
.....	5

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.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

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

1: underscore.js

underscore.js .

underscore.js . underscore.js .

Examples

Underscore JavaScript . Underscore , JavaScript .

Node.js

node npm .

```
npm install underscore
```

node, npm bower

```
bower install underscore
```

NuGet

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

script HTML .

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

[underscore.js](https://riptutorial.com/ko/underscore-js/topic/7688/underscore-js-) : <https://riptutorial.com/ko/underscore-js/topic/7688/underscore-js->

2:

Examples

`_.map iteratee iteratee` .

`iteratee 3` .

1. `item` -
2. `i` -
3. `list` - /

Array trasformed .

:

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

ES6 .

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

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

.

:

```
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']
```

`_.each` , `iteratee` , `context` `iteratee` . `iteratee 3`

```
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)
```

1:

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

"hello", "world" .

2 :

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

3 .

.each (map, pluck, values) iteratee .

: <https://riptutorial.com/ko/underscore-js/topic/7695/>

S. No		Contributors
1	underscore.js	Ashiquzzaman , Community , svarog
2		svarog