

Bogate Interfejsy Użytkownika

Underscore.js

Łukasz Rybka · Gdańsk 2015

UNDERSCORE.JS



Underscore.js

Underscore.js

**Biblioteka JavaScript dostarczająca dużą
ilość funkcyjnych metod wspomagających
development**

Underscore.js

Biblioteka JavaScript dostarczająca dużą
ilość funkcyjnych metod wspomagających
development

NIE rozszerza obiektów Host i Native

Underscore.js

Biblioteka JavaScript dostarczająca dużą
ilość funkcyjnych metod wspomagających
development

NIE rozszerza obiektów Host i Native

Alternatywne (i lepsze) rozwiązanie od
Prototype.js

Underscore.js

**Nie wymaga globalnego obiektu window -
może być wykorzystywane zarówno client-
jak i server-side**

Underscore.js

Nie wymaga globalnego obiektu window -
może być wykorzystywane zarówno client-
jak i server-side

Dokumentacja przez przykłady - każda funkcja
(a jest ich ponad 100!) jest dobrze opisana i
zawiera przykład wykorzystania

Komponenty

Collections

Komponenty

Komponenty

Collections
Arrays

Komponenty

Collections

Arrays

Function

Komponenty

Collections

Arrays

Function

Object

Komponenty

Collections

Arrays

Function

Object

Utility

Underscore-contrib

Underscore-contrib

Rozszerzenie biblioteki Underscore.js

Rozszerzenie biblioteki Underscore.js

Zbiór funkcji typu utility, które z jakichś powodów nie zostały dodane do wersji core biblioteki

Instalacja

Node.js npm install underscore

Instalacja

Node.js npm install underscore

Meteor.js meteor add underscore

Instalacja

Node.js `npm install underscore`

Meteor.js `meteor add underscore`

Require.js `require(["underscore"], ...`

Instalacja

Node.js `npm install underscore`

Meteor.js `meteor add underscore`

Require.js `require(["underscore"], ...`

Bower `bower install underscore`

Instalacja

Node.js npm install underscore

Meteor.js meteor add underscore

Require.js require(["underscore"], ...

Bower bower install underscore

Component component install

jashkenas/underscore

Instalacja

```
<!DOCTYPE html>
<html>
<body>
  <script src="underscore.js"></script>
  <script src="underscore-contrib.js"></script>
  <script type="text/javascript">
    _.each([1, 2, 3], function (element) {
      console.log(element);
    });
  </script>
</body>
</html>
```

Tworzenie biblioteki

```
(function() {  
    // Establish the root object, `window` in the browser, or `require`  
    it on the server.  
    if (typeof exports === 'object') {  
        _ = module.exports = require('underscore');  
    }  
  
    _.mixin({  
        someMethod: function () {  
            // ...  
        }  
    });  
}).call(this);
```

Object functions - properties

```
(function() {  
  _.keys({one: 1, two: 2, three: 3});  
  _.values({one: 1, two: 2, three: 3});  
  _.invert({Moe: "Moses", Larry: "Louis", Curly: "Jerome"});  
  _.functions(_);  
  _.extend({name: 'moe'}, {age: 50});  
  var moe = _.create(Stooge.prototype, {name: "Moe"});  
  _.clone({name: 'moe'});  
  _.extend({name: 'moe'}, {age: 50});  
})();
```

Object functions - type comparison

```
(function() {  
    var stooge = {name: 'moe', luckyNumbers: [13, 27, 34]};  
    var clone  = {name: 'moe', luckyNumbers: [13, 27, 34]};  
  
    _.isEqual(stooge, clone);  
  
    _.isMatch({name: 'moe', age: 32}, {age: 32});  
  
    _.isEmpty({});  
  
    _.isElement(jQuery('body')[0]);  
  
    _.isArray([1,2,3]);  
  
    _.isObject({});  
  
    _.isFunction(alert);  
  
    _.isString("moe");  
})();
```

Object functions - type comparison

```
(function() {  
    _.isNumber(8.4 * 5);  
  
    _.isFinite(-101);  
    _.isFinite(-Infinity);  
  
    _.isBoolean(null);  
    _.isDate(new Date());  
    _.isRegExp(/moe/);  
    _.isError(new TypeError("Example"));  
  
    _.isNaN(NaN);  
    _.isNaN(undefined);  
  
    _.isNull(null);  
    _.isNull(undefined);  
  
    _.isUndefined(window.missingVariable);  
    _.isUndefined(NaN);  
    _.isUndefined(null);  
})();
```

Collection functions - each

```
(function() {  
  _.each([1, 2, 3, 4, 5], function (value, key, list) {  
    console.log(value * value);  
  });  
  
  _.each({one: 1, two: 2, three: 3}, function (value, key, list) {  
    console.log("Key: " + key + " -> " + value);  
  });  
  
  _.each(arguments, function (value, key, list) {  
    console.log("Index: " + key + " -> " + value);  
  });  
})();
```

Collection functions - map

```
(function() {  
    var mapped = _.map([1, 2, 3], function(value, key, list) {  
        return value * 3;  
    });  
  
    console.log(mapped);  
  
    mapped = _.map({one: 1, two: 2}, function(value, key, list) {  
        return value * 3;  
    });  
  
    console.log(mapped);  
})();
```

Collection functions - reduce

```
(function() {  
    var sum = _.reduce([1, 2, 3], function(memo, value, key, list){  
        return memo + value;  
    }, 0);  
  
    console.log(sum);  
  
    sum = _.reduce([1, 2, 3], function(memo, value, key, list){  
        return memo + value;  
    }, 13);  
  
    console.log(sum);  
})();
```


Collection functions - find / filter

```
(function() {  
    var firstEvent = _.find([1, 2, 3, 4, 5, 6], function(value) {  
        return value % 2 == 0;  
    });  
  
    console.log(firstEvent);  
  
    var allEvent = _.filter([1, 2, 3, 4, 5, 6], function(value) {  
        return value % 2 == 0;  
    });  
  
    console.log(allEvent);  
})();
```

Collection functions - where

```
(function() {  
  var listOfPlays = [  
    {title: "King John", year: 1596},  
    {title: "The Merry Wives of Windsor", year: 1597},  
    {title: "Henry V", year: 1599},  
    {title: "Macbeth", year: 1606},  
    {title: "Antony and Cleopatra", year: 1606},  
    {title: "Cymbeline", year: 1610},  
    {title: "The Tempest", author: "Shakespeare", year: 1611}  
  ];  
  
  var plays = _.where(listOfPlays, {year: 1606});  
  
  console.log(plays);  
  
  var firstPlay = _.findWhere(listOfPlays, {year: 1606});  
  
  console.log(firstPlay);  
})();
```

Collection functions - every/all and some/any

```
(function() {  
    var numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9];  
  
    var allEven = _.every(numbers, function (num) {  
        return num % 2 == 0;  
    });  
  
    console.log(allEven);  
  
    var someEven = _.some(numbers, function (num) {  
        return num % 2 == 0;  
    });  
  
    console.log(someEven);  
})();
```

Collection functions - grupowanie

```
(function() {  
    var groupedNumbers = _.groupBy([1.3, 2.1, 2.4], function(num) {  
        return Math.floor(num);  
    });  
  
    console.log(groupedNumbers);  
  
    var groupedStrings = _.groupBy(['one', 'two', 'three'], 'length');  
  
    console.log(groupedStrings);  
})();
```

Array functions

```
(function() {  
  console.log(_.first([5, 4, 3, 2, 1]));  
  console.log(_.first([5, 4, 3, 2, 1], 2));  
  
  console.log(_.initial([5, 4, 3, 2, 1]));  
  console.log(_.initial([5, 4, 3, 2, 1], 2));  
  
  console.log(_.last([5, 4, 3, 2, 1]));  
  console.log(_.last([5, 4, 3, 2, 1], 2));  
  
  console.log(_.flatten([1, [2], [3, [[4]]]]));  
  
  console.log(_.union([1, 2, 3], [101, 2, 1, 10], [2, 1]));  
  
  console.log(_.difference([1, 2, 3, 4, 5], [5, 2, 10]));  
  
  console.log(_.range(10));  
  console.log(_.range(1, 11));  
  console.log(_.range(0, 30, 5));  
})();
```

Chaining

```
(function() {  
  var lyrics = [  
    {line: 1, words: "I'm a lumberjack and I'm okay"},  
    {line: 2, words: "I sleep all night and I work all day"},  
    {line: 3, words: "He's a lumberjack and he's okay"},  
    {line: 4, words: "He sleeps all night and he works all day"}  
  ];  
  
  _.  
    chain(lyrics)  
    .map(function(line) { return line.words.split(' '); })  
    .flatten()  
    .reduce(function(counts, word) {  
      counts[word] = (counts[word] || 0) + 1;  
      return counts;  
    }, {})  
    .value();  
})();
```

ANY
QUESTIONS
?

Pytania?