

6ab6352918 ▾

...

safe-obj / lib / index.js / <> Jump to ▾



tandrewnichols Use constructor instead of typeof

🕒 History

1 contributor

165 lines (145 sloc) | 3.85 KB ...

```
1 ;(function(root) {
2
3   var isObject = function isObject(obj) {
4     // Implementation lifted from underscore
5     return typeof obj === 'function' || typeof obj === 'object' && !!obj;
6   };
7
8   var isArray = function isArray(arr) {
9     return Array.isArray ? Array.isArray(arr) : arr instanceof Array || (arr && arr.constructor && arr.constructor.name === 'Array');
10  };
11
12  var any = function any(list, func) {
13    var i;
14    var l = list.length;
15    for (i = 0; i < l; i++) {
16      if (func(list[i])) {
17        return true;
18      }
19    }
20    return false;
21  };
22
23  var every = function every(list, func) {
24    var i;
25    var l = list.length;
26    for (i = 0; i < l; i++) {
27      if (!func(list[i])) {
28        return false;
29      }
30    }
31    return true;
32  };
33
34  var isEqual = function isEqual(a, b) {
35    if (a === b) {
36      return true;
37    }
38
39    if (a.constructor !== b.constructor) {
40      return false;
41    }
42
43    if (typeof a === 'object') {
44      if (a instanceof Array) {
45        if (a.length !== b.length) {
46          return false;
47        }
48        var i;
49        var l = a.length;
50        for (i = 0; i < l; i++) {
51          if (!isEqual(a[i], b[i])) {
52            return false;
53          }
54        }
55        return true;
56      } else {
57        for (var k in a) {
58          if (!(k in b) || !isEqual(a[k], b[k])) {
59            return false;
60          }
61        }
62        return true;
63      }
64    }
65  } else {
66    return false;
67  }
68 };
69
70 var anyOrAll = function anyOrAll(method, _args) {
71   var paths = [], obj = {};
72
73   // args is of type Arguments
74   var args = [].slice.call(_args);
75
76   // Allow array or list of strings
77   if (isArray(args[1])) {
78     obj = args[0];
```

```

79     paths = args[1];
80 } else {
81     paths = args;
82     obj = paths.shift();
83 }
84
85 if (isObject(obj)) {
86     return method(paths, function(path) {
87         return _safe.safe(obj, path);
88     });
89 } else {
90     return false;
91 }
92 };
93
94 var _safe = {
95     safe: function (obj, path, otherwise) {
96         if (!path) {
97             return otherwise;
98         }
99         obj = isObject(obj) ? obj : {};
100         var props = path.split('.');
101         if (props.length === 1) {
102             if (typeof obj[props[0]] === 'undefined') {
103                 return otherwise;
104             } else if (obj[props[0]] === null) {
105                 return typeof otherwise === 'undefined' ? null : otherwise;
106             } else {
107                 return obj[props.shift()];
108             }
109         } else {
110             var prop = props.shift();
111             return isObject(obj[prop]) ? _safe.safe(obj[prop], props.join('.'), otherwise) : otherwise;
112         }
113     },
114
115     expand: function (obj, path, thing) {
116         if (!path || typeof thing === 'undefined') {
117             return;
118         }
119         obj = isObject(obj) && obj !== null ? obj : {};
120         var props = path.split('.');
121         if (props.length === 1) {
122             obj[props.shift()] = thing;
123         } else {
124             var prop = props.shift();
125             if (!(prop in obj)) {
126                 obj[prop] = {};
127             }
128             _safe.expand(obj[prop], props.join('.'), thing);
129         }
130     },
131
132     ensure: function (obj, path, disallowed, otherwise) {
133         if (arguments.length === 3) {
134             otherwise = disallowed;
135             disallowed = [];
136         }
137         var current = _safe.safe(obj, path);
138         if (typeof current === 'undefined' || current === null || any(disallowed, function(item) { return isEqual(current, item); })) {
139             _safe.expand(obj, path, otherwise);
140         }
141     },
142
143     allOf: function() {
144         return anyOrAll.apply(null, [every].concat(arguments));
145     },
146
147     anyOf: function() {
148         return anyOrAll.apply(null, [any].concat(arguments));
149     },
150
151     noneOf: function() {
152         return !_safe.anyOf.apply(null, arguments);
153     }
154 };
155
156 if (typeof exports !== 'undefined') {
157     if (typeof module !== 'undefined' && module.exports) {
158         module.exports = _safe;
159     }
160     exports._safe = _safe;
161 }
162
163 root._ = root._ || {};
164 root._._safe = _safe;
165 })(this);

```