Guides API Community Blog Builds Search ...

**PROJECTS** 

Ember

**Ember Data** 

TAG: V2.11.0

**MODULES** 

ember

ember-application

ember-debug

ember-extension-support

ember-glimmer

ember-htmlbars

ember-metal

ember-routing

ember-runtime

ember-testing

ember-views

NAMESPACES

Ember

Ember. AutoLocation

**Ember.FEATURES** 

Ember.Instrumentation

Ember.Location

Ember.String

Ember.computed

Ember.inject

Ember.run

CLASSES

Backburner

BucketCache

Container

ContainerProxyMixin

Descriptor

Ember. Action Handler

Ember. Action Support

Ember. Application

Ember. Application Instance

 ${\bf Ember. Application Instance. Boot O...}$ 

Ember. Array

Ember. Array Proxy

## **EMBER.COMPUTED NAMESPACE**

**PUBLIC** 

**DEFINED IN:** packages/ember-metal/lib/computed.js:450

**MODULE:** ember-metal

This helper returns a new property descriptor that wraps the passed computed property function. You can use this helper to define properties with mixins or via <code>Ember.defineProperty()</code>.

If you pass a function as an argument, it will be used as a getter. A computed property defined in this way might look like this:

```
1
      let Person = Ember.Object.extend({
2
       init() {
3
          this._super(...arguments);
4
5
          this.firstName = 'Betty';
6
          this.lastName = 'Jones';
7
       }.
8
        fullName: Ember.computed('firstName', 'lastName', function() {
9
10
          return `${this.get('firstName')} ${this.get('lastName')}`;
11
       })
12
     });
13
14
     let client = Person.create();
15
16
      client.get('fullName'); // 'Betty Jones'
17
18
      client.set('lastName', 'Fuller');
19
      client.get('fullName'); // 'Betty Fuller'
```

You can pass a hash with two functions, <code>get</code> and <code>set</code>, as an argument to provide both a getter and setter:

```
1
      let Person = Ember.Object.extend({
2
        init() {
3
          this._super(...arguments);
4
          this.firstName = 'Betty';
          this.lastName = 'Jones';
6
7
8
9
        fullName: Ember.computed('firstName', 'lastName', {
10
          get(key) {
            return `${this.get('firstName')} ${this.get('lastName')}`;
11
12
13
          set(key, value) {
            let [firstName, lastName] = value.split(/\s+/);
14
15
            this.setProperties({ firstName, lastName });
16
            return value;
17
          }
18
        });
19
      })
20
21
      let client = Person.create();
22
      client.get('firstName'); // 'Betty'
23
      client.set('fullName', 'Carroll Fuller');
24
25
      client.get('firstName'); // 'Carroll'
```

Ember. Binding

Ember.Checkbox

Ember.ClassNamesSupport

Ember.Comparable

Ember.Component

Ember.ComputedProperty

Ember.ContainerDebugAdapter

Ember.Controller

Ember.ControllerMixin

Ember.Copyable

Ember.CoreObject

Ember.CoreView

Ember. Data Adapter

Ember. Debug

Ember.DefaultResolver

Ember. Each Proxy

Ember. Engine

Ember. EngineInstance

Ember. Enumerable

Ember. Error

Ember. Event Dispatcher

Ember. Evented

Ember. Freezable

Ember.HTMLBars

Ember. HashLocation

Ember. Helper

Ember. History Location

Ember.InjectedProperty

Ember.LinkComponent

Ember. Map

Ember. MapWithDefault

Ember.Mixin

Ember. Mutable Array

Ember. Mutable Enumerable

Ember. Namespace

Ember. Native Array

Ember. NoneLocation

Ember.Object

Ember.ObjectProxy

Ember.Observable

Ember.OrderedSet

 ${\color{red}Ember.PromiseProxyMixin}$ 

Ember.ProxyMixin

Ember.Route

Ember.Router

Ember. Service

Ember.TargetActionSupport

Ember.Templates.helpers

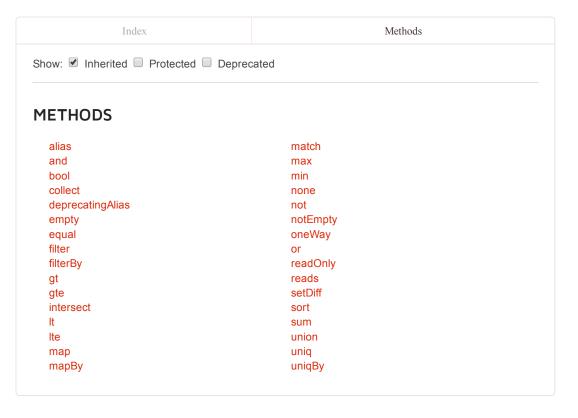
Ember. Test

The set function should accept two parameters, key and value. The value returned from set will be the new value of the property.

Note: This is the preferred way to define computed properties when writing third-party libraries that depend on or use Ember, since there is no guarantee that the user will have prototype Extensions enabled.

The alternative syntax, with prototype extensions, might look like:

```
fullName: function() {
   return this.get('firstName') + ' ' + this.get('lastName');
}.property('firstName', 'lastName')
```



Ember.Test.Adapter

Ember.Test.QUnitAdapter

Ember.TextArea

Ember.TextField

Ember.TextSupport

Ember. View

Ember.ViewMixin

Function

Libraries

Registry

Registry Proxy Mixin

RoutingService

String

COPYRIGHT 2017 TILDE INC.
TEAM | SPONSORS
SECURITY | LEGAL | LOGOS
COMMUNITY GUIDELINES

Ember.js is free, open source and always will be.





