

# CAPH.WUI.ANI.BOUNCEANIMATION

BounceAnimation represents a class that provides bounce effect for widgets. The bounce effect is like a basketball that fall to the ground, it would bounce back and forth, and the range of the bounce is less and less, stopping on the ground at last. This is one of the six basic types of animation. Bounce animation could not perform with other animation in the same loader parallel.

## Contents

**Constructor**  
**BounceAnimation**

**Methods**  
**BounceAnimation**  
**add**  
**clone**  
**remove**  
**getList**  
**equals**

## Constructor

BounceAnimation		
Description		
Create a BounceAnimation.		
Parameters		
obj	Object	A widget created by user.
option	Object	- The option include below properties * time : (Number) indicates the times for widgets using bounce animation, widgets bounce up and down, or left and right is regarded as once(one time) * direction : (String) indicates the direction for widget using bounce animation, possible values are below - 'up' : bounce up and back - 'down' : bounce down and back - 'left' : bounce left and back - 'right' : bounce right and back * distance : (Number) indicates the maximum range for widgets using bounce animation, the unit is pixe(px) * ease : (String) describes the motion tween of the batch of animations, it could be set as one of the values below - 'SineCubic.InOut', 'Linear.None', if user don't set ease, this is default. 'Quadratic.In', 'Quadratic.Out', 'Quadratic.InOut', 'Cubic.In', 'Cubic.Out', 'Cubic.InOut', 'Quartic.In', 'Quartic.Out', 'Quartic.InOut', 'Quintic.In', 'Quintic.Out', 'Quintic.InOut', 'Sinusoidal.In', 'Sinusoidal.Out', 'Sinusoidal.InOut', 'Exponential.In', 'Exponential.Out', 'Exponential.InOut', 'Circular.In', 'Circular.Out', 'Circular.InOut', 'Elastic.In', 'Elastic.Out', 'Elastic.InOut', 'Back.In', 'Back.Out', 'Back.InOut', 'Bounce.In', 'Bounce.Out', 'Bounce.InOut' * duration : (Number) describes how long the batch of animations to be performed in the meanwhile would last - [default : 1000, the unit is millisecond(ms)] * delay : (Number) describes the time duration of the batch of animations would defer starting, the unit is millisecond(ms)}
Emulator Support	Y	
SDK Constraint	None	
Example		
<pre>var BounceAnimation = caph.wui.ani.BounceAnimation; var AniLoader = caph.wui.ani.AniLoader; var Box = caph.wui.widget.Box; var UIContext = caph.wui.widget.UIContext;  var bounce = new BounceAnimation(); var loader = new AniLoader(); var uiContext = new UIContext(); var widget = new Box(); widget.render(uiContext);  var bounceOpt = {direction:'up', times: 10, duration:2000, distance: 100}; bounce.add(widget, bounceOpt);  loader.add(bounce); loader.start(uiContext);</pre>		

## Methods

BounceAnimation	
Description	
(Constructor) Create a BounceAnimation.	
Parameters	<div>■obj</div> <div>- Object</div> <div>- A widget created by user.</div> <div>■option (Optional)</div> <div>- Object</div> <div>- The option include below properties</div> <div>* time : (Number) indicates the times for widgets using bounce animation, widgets bounce up and down, or left and right is regarded as once(one time)</div> <div>* direction : (String) indicates the direction for widget using bounce animation, possible values are below</div> <div>- 'up' : bounce up and back</div> <div>- 'down' : bounce down and back</div> <div>- 'left' : bounce left and back</div> <div>- 'right' : bounce right and back</div> <div>* distance : (Number) indicates the maximum range for widgets using bounce animation, the unit is pixe(px)</div> <div>* ease : (String) describes the motion tween of the batch of animations, it could be set as one of the values below</div> <div>- 'SineCubic.InOut', 'Linear.None', if user don't set ease, this is default. 'Quadratic.In', 'Quadratic.Out', 'Quadratic.InOut', 'Cubic.In', 'Cubic.Out', 'Cubic.InOut', 'Quartic.In', 'Quartic.Out', 'Quartic.InOut', 'Quintic.In', 'Quintic.Out', 'Quintic.InOut',</div> <div>* duration : (Number) describes how long the batch of animations to be performed in the meanwhile would last</div> <div>- [default : 1000, the unit is millisecond(ms)]</div> <div>* delay : (Number) describes the time duration of the batch of animations would defer starting, the unit is millisecond(ms)}</div>
Return	■Void
Emulator Support	Y
SDK Constraint	none

Example
<pre>var BounceAnimation = caph.wui.ani.BounceAnimation; var AniLoader = caph.wui.ani.AniLoader; var Box = caph.wui.widget.Box; var UIContext = caph.wui.widget.UIContext;  var bounce = new BounceAnimation(); var loader = new AniLoader(); var uiContext = new UIContext(); var widget = new Box(); widget.render(uiContext);  var bounceOpt = {direction:'up', times: 10, duration:2000, distance: 100}; bounce.add(widget, bounceOpt);  loader.add(bounce); loader.start(uiContext);</pre>

## add

Description	
Bind the options needed for bounce animation with user's widget.	
Parameters	<div> <div>■obj</div> <div>- Object</div> <div>- A widget created by user.</div> <div>■option (Optional)</div> <div>- Object</div> <div>- The option include below properties</div> <div>* time : (Number) indicates the times for widgets using bounce animation, widgets bounce up and down, or left and right is regarded as once(one time)</div> <div>* direction : (String) indicates the direction for widget using bounce animation, possible values are below</div> <div>- 'up' : bounce up and back</div> <div>- 'down' : bounce down and back</div> <div>- 'left' : bounce left and back</div> <div>- 'right' : bounce right and back.</div> <div>* distance : (Number) indicates the maximum range for widgets using bounce animation, the unit is pixel(px)</div> <div>* ease : (String) describes the motion tween of the batch of animations, it could be set as one of the values below</div> <div>- 'SineCubic.InOut', 'Linear.None', if user don't set ease, this is default. 'Quadratic.In', 'Quadratic.Out', 'Quadratic.InOut', 'Cubic.In', 'Cubic.Out', 'Cubic.InOut', 'Quartic.In', 'Quartic.Out', 'Quartic.InOut', 'Quintic.In', 'Quintic.Out', 'Quintic.InOut',</div> <div>* duration : (Number) describes how long the batch of animations to be performed in the meanwhile would last</div> <div>- [default : 1000, the unit is millisecond(ms)]</div> <div>* delay : (Number) describes the time duration of the batch of animations would defer starting, the unit is millisecond(ms))</div> </div>
Return	■Void
Emulator Support	Y
SDK Constraint	none

Example
<pre>var BounceAnimation = caph.wui.ani.BounceAnimation; var AniLoader = caph.wui.ani.AniLoader; var Box = caph.wui.widget.Box; var UIContext = caph.wui.widget.UIContext;  var bounce = new BounceAnimation(); var loader = new AniLoader(); var uiContext = new UIContext(); var widget = new Box(); widget.render(uiContext);  var bounceOpt = {direction:'up', times: 10, duration:2000, distance: 100}; bounce.add(widget, bounceOpt);  loader.add(bounce); loader.start(uiContext);</pre>

## clone

Description	
Creates and returns clone object from current object, the cloned object will have the same properties and same methods with the current object.	
Parameters	■Void
Return	■Object - The cloned object.
Emulator Support	Y
SDK Constraint	none

Example
<pre>var BounceAnimation = caph.wui.ani.BounceAnimation; var bounce = new BounceAnimation(); var obj = bounce.clone(); // obj is now a cloned Object.</pre>

## remove

Description	
Removes all the arguments that needed for animation on the widget.	
Parameters	■obj - Object - An instance of widget created by user.
Return	■Void
Emulator Support	Y
SDK Constraint	none

Example
<pre>var BounceAnimation = caph.wui.ani.BounceAnimation; var bounce = new BounceAnimation();  // Let's say that obj and options are given. bounce.add(obj, options); bounce.remove(obj);</pre>

# getList

Description

Returns animation list.

Parameters

■Void

Return

■Array

- The array of a object pairs, including widget and options of its' animation.

Emulator Support

Y

SDK Constraint

none

Example

```
var BounceAnimation = caph.wui.ani.BounceAnimation;
var bounce = new BounceAnimation();

// Use Animation Sub-classes....

var arr = bounce.getList();
for(var i = 0; i < arr.length; i++){
    console.log(arr);
}
```

# equals

Description

Compares the contents of two objects using strict equality, objects are considered equal if they both have the same set of properties and the values of those properties are equal.

Parameters

■Object

- Object

- The object which wants to compare with current object.

Return

■Boolean

- Indicates whether the two objects are equal,

- true : if they are equal , return true.

- false : if they aren't equal, return false.

Emulator Support

Y

SDK Constraint

none

Example

```
var BounceAnimation = caph.wui.ani.BounceAnimation;
var bounce = new BounceAnimation();
var obj = bounce.clone();
var isequal = obj.equals( bounce );
```