M4Q Animation

Use m4q animation to animate HTMLElements.

About

The `m4q` contains functions for animate HTML elements. Animate function used `requestAnimationFrame` method to executing animation rules.

Animation

To animate element, you can use `$.animate` function. Global function receive next arguments:

```
$.animate(element, draw, duration, timing, callback);
```

- **element** - HTMLElement
- **draw** - function or object of properties to animate
- **duration** - animation duration
- **timing** - easing function name to calculate timing
- **callback** - callback function for executing after animation is done

Draw function

Using the function gives you more control over what happens during the animation. Function receive two arguments: `normalized timing` and `timing with apply easing`.

Normalized timing has value from 0 to 1. Timing with easing has value from 0 to 1 multiplied by the value of the easing function.

Go

```
var el = $("#dc-1, #dc-2");
var startPos = parseInt(el.style("left"));
var maxLeft = startPos === 0 ? el.parent().width() - 62 : 0;
var delta = maxLeft - startPos;

btn.disabled = true;

$.each(el, function(){
    var element = this;
    $.animate(element, function(t, p){

        var m = $(element).id() === 'dc-1' ? t : p;
        $(element).css("left", startPos + (delta * m) + "px");

    }, 3000, "easeInOutBounce", function(){
        btn.disabled = false;
    });
});
```

Draw object

You can use a plane object for setup animation rules. If you use an object, you must define end value for each property, which is going to animate.

Go

```
var el = $("#demo-cube-2");
var startPos = parseInt(el.style("left"));
var maxLeft = startPos === 0 ? el.parent().width() - 62 : 0;

btn.disabled = true;

$.animate(el, {
    left: maxLeft
}, 3000, function(){
    btn.disabled = false;
});
```

Animate elements set

You can animate elements set. To animate elements set, you must use set animate function to animate elements set - `$(selector).animate(...)`

Go

```
var el = $(".c1, .c2, .c3");
var startPos = parseInt(el.style("left"));
var maxLeft = startPos === 0 ? el.parent().width() - 62 : 0;

btn.disabled = true;

el.animate({
    left: maxLeft
}, 3000, function(){
    btn.disabled = false;
});
```

Stopping animation

You can stop animation with special method `$(selector).stop([complete])` . The `complete` optional argument must be boolean and guarantee the end of the animation when `true` .

Go   Stop   Stop and complete

Easing function for timing

Thanks to **Andrew Ray** for easing utils

Easing can really bring life to an effect. Easing controls how an animation progresses over time by manipulating its acceleration. Easing can be applied when using the `m4q.animate()` method.
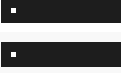
The default easing function is `linear` .

Func name	Demo
linear	
easeInQuad	
easeOutQuad	
easeInOutQuad	
easeInCubic	
easeOutCubic	
easeInOutCubic	
easeInQuart	
easeOutQuart	
easeInOutQuart	
easeInQuint	
easeOutQuint	
easeInOutQuint	
easeInSine	
easeOutSine	
easeInOutSine	
easeInExpo	
easeOutExpo	
easeInOutExpo	
easeInCirc	
easeOutCirc	
easeInOutCirc	
easeInElastic	
easeOutElastic	
easeInOutElastic	
easeInBack	
easeOutBack	
easeInOutBack	
easeInBounce	

Func name	Demo
easeOutBounce	
easeInOutBounce	

Func name	Demo
easeOutBounce	
easeInOutBounce	