Baffects.js Cheatsheet

0.1Beta

by Roberto Cabezas H

PROPERTIES

```
.property.value // gives the value of property
.property.keyValue(key) // gives value of property at a keyframe index
.property.valueAtTime(time) // gives value of proprty at time in seconds

var shape = b.rect(100,100,100,100); // all creation functions or effects return a Properties object
shape.props(); // print all properties available for animation
```

COLOR

```
b.color(r,g,b,[a]); // creates a color Array from parameters r, g, b, a
b.fill(r,g,b,[a]); // set the color of fill
b.stroke(r,g,b,[a]); // set the color of stroke
b.toRgba(r,g,b,[a]); // Convert from HSBA to RGBA color mode
```

Note: Color can be manipulated in a range of [0 - 255] or [0 - 1] for RGBA, and [0 - 360], [0 - 1] for Hue, [0 - 100], [0 - 1] for Saturation, Brightness and Alpha. Color manipulation of [0 - 1] range is preffered

DRAWING

```
b.solid(); // create a solid
b.rect(x,y,width,height); // draws a rectangle
b.ellipse(x,y,width,height); // draws an ellipse
b.polygon(x,y,radius,n); // draws a polygon of n sides
b.line(x1,y1,x2,y2); // draws a line from one point to another
b.nullLayer(); // creates a null layer
b.strokeWeight(weight); // set the stroke current unit
b.anchorMode(mode); // bounding box position of anchor point mode: 1-9 number
b.set3DMode(mode); //b.ENABLE, b.DISSABLE
–
b.beginShape([mode]); // start drawing of shape
b.vertex(x, y); // add vertex to shape
b.endShape([close]); // build shape
–
b.shapeTrimPaths(layer); // apply effect to shape layer (check different types of effects in the documentation)
```

TEXT

```
b.text("text",x,y); // draws string text
b.textAlign(mode); // set text align mode
b.textSize(size); // set text size
b.textFont("Helvetica-Regular"); // set font with type
b.textAnchorPoint(layer); // apply effect to text layer (check different types of effects in the documentation)
```

KEYFRAMES

```
b.key(prop,value); // set property value
b.key(prop,time,value); // set property value at time
b.interpolationMode(mode); // b.BEZIER, b.LINEAR, b.HOLD set keyframe interpolation mode
b.spatialMode(mode); // b.LINEAR, b.AUTO_BEZIER set keyframe spatial mode
b.speed(prop,index,speed,influence); // manipulates keyframe speed
b.speedAll(prop,speed,influence); // apply speed manipulation to all keyframes of a property
b.expression(prop,expression); // apply expression to property
```

LAYERS

```
b.getLayer(name); // get layer by name as Properties object
b.getAllLayers(); // get Array of all layers in the composition as Properties object
b.addFX(layer,name); // add and effect to layer
b.addPreset(layer,path); // add preset to layer
b.getAllFXProperties(layer); // get all animatable properties from all effects in a layer
b.trackMatte(layer,mode); // set track matte for layer
b.getIndex(layer); // get layer index
b.selectedLayers(); // get manually selected layer from a composition
b.makeParent(parent,children); // set parent children relationship
b.anchor(layer,mode); // change anchor point bounding box position for a layer
b.audioToKeyframes(layer); // create keyframes from amplitude audio analysis
b.createShapes(layers); // create ShapeLayer from illustrator layer
b.blendingMode(mode); // set global blending mode for layers

//Useful attributes from AE current API
shape.layer.name // read-only layer's name
shape.layer.startTime // set and get layer's start time
shape.layer.inPoint // set layer's in point
shape.layer.outPoint // set layer's out point
shape.layer.width // read-only layer's width
shape.layer.height // read-only layer's height
```

COMPOSITIONS

```
b.getComp(name); // get composition reference by name
b.setComp(name); // set current composition to work with
b.cleanComp(); // clean composition from previous created items
b.background(color); // set background colora
b.getTotalFrames(); // get total frames of current composition
b.getDuration(); // get total duration of curret composition
b.setDuration(); // set total duration of new composition
b.numLayers(); // get total number of layers current composition
b.getAllComps(); // get Array of all compositions in project
b.getCenter(); // get x and y center coordinates of current composition
```

MASKS

```
b.maskMode(prop,value); // set global mask mode
_
b.beginMask([mode]); //  start drawing of mask with mode
b.vertex(x,y); // set global mask mode
b.endMask(layer,[close]); // set global mask mode
_
```

DATA

```
b.loadItem(path); // loads an item from data folder: audio, images, illustrator files, video
b.loadSequence(path); // loads a sequence from data folder
b.loadAI(path); // loads AI Illustrator file and creates shapes from layers on file
```

RENDER

```
b.addToRender(comp,template,path); // add compositin to render queue in AE and apply output module template
b.render(); // render AE quere
b.addToRenderAME(comp,template,path); // add compositin to render queue in Adobe Media Encoder
b.renderAME(); // render Adobe Media Encoder queue
```

ENVIRONMENT

/* go() function is to setup the environment, if none parameter is given it creates a 1920x1080 "untitled" composition with a duration of 10 seconds and 24 frame rate.

Default values are:
name: "untitled"
width: 1920
height: 1080
durations: 10 s
frameRate: 24;

Syntax:
b.go( );
b.go(name);
b.go(name, duration);
b.go(name, width, height);
b.go(name, width, height, duration);
b.go(name width, height, duration, frameRate);

```
b.go(name,width,height,duration,frameRate);
```