Suica 2.0 for JavaScript

<script src="**suica.js**"></script>

boytchev.github.io/suica
version 1

BASIC OBJECTS

circle (center, size, color)
cone (center, size, color)
cube (center, size, color)
cylinder (center, size, color)
line (center, size, color)
point (center, size, color)
polygon (count, center, size, color)
prism (count, center, size, color)
pyramid (count, center, size, color)
sphere (center, size, color)
square (center, size, color)

BASIC PROPERTIES

.center = [**.x**, **.y**, **.z**]
.color = "colorname" / 0xFFFFFF / [r,g,b]$^{0..1}$
= **rgb** (r,g,b)^{255} / **hsl** (h^{360},s^{100},l^{100})
.count = count / [count, count]
.image = drawing / "filename" / **image** ("filename")
.images = .count / [count, count]
.size = width / [**.width**, **.height**, **.depth**]
.spin = spinH / [**.spinH**, **.spinV**, **.spinT**]
.wireframe = true / yes / false / no

SUICA

background (color)
oxyz (size, color)
demo (distance, altitude, speed)
orbit (distance, altitude, speed)
lookAt (from, to, up)
perspective (near, far, fov)
orthographic (near, far)
fullWindow ()
fullScreen ()
stereo (distance)
anaglyph (distance)
vr ()
capture (filename, time, fps, format, skipframes)

MISC

its
obj.**clone**
obj.**style** ({name: value, ...})
allObjects ()
findPosition (event)
findObject (event)
findObjects (event)
objectPosition (local)
screenPosition (local, global)
radians (degrees)
degrees (radians)
random (from, to)
random ([value,...])

ADVANCED OBJECTS

construct (expression, size, color)
convex (src^{2}, size, color)
group (object, object, ...)
.**add** (object, object, ...)
model (filename, center, size)
model.save (filename, [object, object, ...])
surface (center, curve^{3}, count, size, color)
text3d (text, font$^{!}$, center, size, color)
tube (center, curve^{2}, radius, count, size, color)
spline (src^{2}, closed, interpolating)
(src^{5}, param, param)
splane (src^{3}, closed$^{!,2}$, interpolating$^{!,2}$)
(src^{4}, param, param)

ADVANCED PROPERTIES

.closed = bool$^{!}$ / [bool, bool]^{2}
.curve^{2} = [point, ...] / spline / f(u)
^{3} = [[point, ...], ...] / splane / f(u,v)
.expression = "string" A+B, A-B, A*B, (...)
.font$^{!}$ = "fontname.json"
.interpolating = bool$^{!}$ / [bool, bool]^{2}
.src = [point, ... point]^{2} / f(u)^{5}
= [[point, ...], ... [point, ...]]^{3} / f(u,v)^{4}
.threejs = THREE.Mesh
.**material** = THREE.Material
.**geometry** = THREE.BufferGeometry

EVENTS

onPointerEnter, **onPointerLeave**, **onPointerMove**, **onPointerDown**, **onPointerUp**, **onClick**, **onTime**

obj.**addEventListener** (eventName, eventHandler)
obj.**removeEventListener** (eventName)
obj.eventName = eventHandler

function pointerEventHandler (event) { ... }
function timeEventHandler (time, dTime) { ... }

proactive ()

DRAWINGS

drawing (width, height, color)
moveTo (x, y, x, y, ...)
lineTo (x, y, x, y, ...)
curveTo (m_x, m_y, x, y)
arc (x, y, radius, from, to, cw)
stroke (color, width, closed)
fill (color)
fillText (x, y, text, color, font)
"bold 20px Courier"
clear (color)

LMS

scorm
.**api**, .**score**, .**studentName**, .**getValue** (value) .**setValue** (name, value), .**derandomize** (seed)

Suica 2.0 for HTML

<script src="**suica.js**"></script>

boytchev.github.io/suica
version 1

BASIC OBJECTS

<**circle** center size color ...>
<**cone** center size color ...>
<**cube** center size color ...>
<**cylinder** center size color ...>
<**line** from to color ...>
<**point** center size color ...>
<**polygon** count center size color ...>
<**prism** count center size color ...>
<**pyramid** count center size color ...>
<**sphere** center size color ...>
<**square** center size color ...>
<... id spin image images wireframe>

BASIC PROPERTIES

center = "**x**, **y**, **z**"
color = "colorname" / "0xFFFFFF", "r,g,b"[1]
= "**rgb** (r,g,b)"[255] / "**hsl** (h[360],s[100],l[100])"
count = "count" / "count, count"
id = "string"
image = "drawing" / "filename"
images = "count" / "count, count"
size = "width" / "**width**, **height**, **depth**"
spin = "spinH" / "**spinH**, **spinV**, **spinT**"
wireframe = "bool"

SUICA

<**suica** width height background orientation proactive
perspective orthographic fullWindow fullScreen
stereo anaglyph vr> ... </**suica**>
<**background** color>
<**oxyz** size color>
<**demo** distance altitude speed>
<**orbit** id distance altitude speed>
<**lookAt** from to up>
<**perspective** near far fov>
<**orthographic** near far>
<**fullWindow**>
<**fullScreen**>
<**stereo** distance>
<**anaglyph** distance>
<**vr**>
<**capture** filename time fps format skipframes>

DRAWINGS

<**drawing** size color>
<**moveTo** point>
<**lineTo** point>
<**curveTo** m point>
<**arc** point radius from to cw>
<**stroke** color width closed>
<**fill** color>
<**fillText** point text color font>
<**clear** color>

point="x, y" or **x**="x" **y**="y"
font="bold 20px Courier"

ADVANCED OBJECTS

<**clone** src[1] center size color ...>
<**construct** expression center size color>
<**convex** src[2] size color ...>
<**group** center size color ...> ... </**group**>
<**model** filename center size ...>
<**surface** center curve[3] count size color ...>
<**text3d** text font[1] center size color ...>
<**tube** center curve[2] radius count size color ...>
<**splane** src[2,5] closed[1] interpolating[1]>
<**splane** src[3,4] closed[1,2] interpolating[1,2]>
<... id spin image images wireframe>

ADVANCED PROPERTIES

closed = "bool"[1] / "bool, bool"[2]
curve[2] = "point; ..." / "spline" / "func(u)"
[3] = "point; ... | ... " / "splane" / "func(u,v)"
expression = "string" A+B, A-B, A*B, (...)
font[1] = "fontname.json"
interpolating = "bool"[1] / "bool, bool"[2]
src = "id"[1] / "point; ..."[2] / "func(u)"[5]
= "point; ... | ... "[3] / "func(u,v)"[4]

interpolating vs **approximating**
closed vs **open**, **cw** vs **ccw**

EVENTS

onPointerEnter, **onPointerLeave**, **onPointerMove**,
onPointerDown, **onPointerUp**, **onClick**, **onTime**

<tag ... eventName="eventHandler">

<**proactive**>