

# Javascript API

## (Reference sheet)

### Logging functions

logDebug ()

A safe way of logging debug information. The function first check if the console exist before sending a debug message to it. (Prevents crashing)

Parameters:

passStr – String to be sent to the console

Returns:

None

---

logError ()

A safe way of logging error information. The function first check if the console exist before sending a error message to it. (Prevents crashing)

Parameters:

passStr – String to be sent to the console

Returns:

None

### Osmosis()

Description: Osmosis is an abstraction layer in-between the plugin and JavaScript. Every display object extends from Osmosis for common functionality.

Public members	Description
.id	The ID number of the node internal to the plugin. This is how the plugin knows what object is being referenced.
.height	The last recorded height of the object. To get the current height of an object run the .update() function.
.width	The last recorded width of the object. To get the current height of an object run the .update() function.
.x	The last recorded x position of the object. To get the current x position of an object run the .update() function.
.y	The last recorded x position of the object. To get the current x position of an object run the .update() function. Note: The y axis is inverted so 0 starts at the bottom of the plugin and 320 is at the top.

### Public methods

getId()

Get the node ID of the object for reference in the plugin.

Parameters:

None

Returns:

A string of the node ID

---

setPosition()

Sets the object to passed coordinates.

Parameters:

x – Number of the new x position of the object.

y – Number of the new y position of the object.

Returns:

None

---

setDimensions()

Sets the object to passed dimensions. Images will scale automatically.

Parameters:

width – Number of the new width of the object.

height – Number of the new height of the object.

Returns:

None

---

setColor()

Sets the background color of the object. Typically used for layers, text labels and textboxes.

Parameters:

r – Number of the amount of red in the object limit (0-1)

g – Number of the amount of green in the object limit (0-1)

b – Number of the amount of blue in the object limit (0-1)

a – Number of the amount of alpha in the object limit (0-1)

Returns:

None

---

toggleVisibility()

Toggles the visibility of this object.

Parameters:

None

Returns:

None

---

setVisibility()

Sets the visibility of this object.

Parameters:

visibility – Boolean flag true means visible and false means invisible.

Returns:

None

---

addChild()

Adds the passed child to the display list. Child objects will move with the parent object.

Parameters:

child – Object to be added to this object  
Returns:  
None

---

removeChild()  
Removes the passed child from the display list.  
Parameters:  
child – Object to be removed from this object  
Returns:  
None

---

addMoveTo()  
Tells the director to linearly move the object to the passed global position in the passed amount of time.  
Parameters:  
x - Number of the new x position globally to move the object to.  
y - Number of the new y position globally to move the object to.  
duration - Number of the amount of time the translation takes.  
Returns:  
None

---

addMoveBy()  
Tells the director to linearly move the object to the passed position based on the objects current position in the passed amount of time.  
Parameters:  
x - Number of the new x offset to move the object to.  
y - Number of the new y offset to move the object to.  
duration - Number of the amount of time the translation takes.  
Returns:  
None

---

getPosition()  
Returns the current position and dimensions of the object  
Parameters:  
None  
Returns:  
An array of 4 numbers [x, y, width, height]

---

update()  
Updates the current position and dimensions of the object and writes them to the .x, .y, .width and .height members.  
Parameters:  
None  
Returns:  
None

---

checkId()

An internal function to test if the .id member has valid information in it.

Parameters:

None

Returns:

True if valid and false otherwise

## BaseSprite()

Description: Sprite objects and buttons object extends from BaseSprite for common functionality.

Public members	Description
.boolSubscribed	If true then the object can receive events otherwise if false then the object ignores events.

## Public methods

subscribe()

Allows the object to receive events from the plugin

Parameters:

None

Returns:

None

---

unsubscribe()

The object ignores events from the plugin

Parameters:

None

Returns:

None

---

moveCallback()

Adds an event listener to this object for mouse move callback. This event is triggered whenever the mouse moves over this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.  
func – Function pointer to the callback function.

Returns:

None

---

downCallback()

Adds an event listener to this object for mouse down callback. This event is triggered whenever the mouse clicks down on this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.  
func – Function pointer to the callback function.

Returns:

None

---

overCallback()

Adds an event listener to this object for mouse over callback. This event is triggered once whenever the mouse moves over this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.  
func – Function pointer to the callback function.

Returns:

None

---

outCallback()

Adds an event listener to this object for mouse over callback. This event is triggered once whenever the mouse leaves this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.  
func – Function pointer to the callback function.

Returns:

None

---

upCallback()

Adds an event listener to this object for mouse up callback. This event is triggered whenever the mouse button is released over this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.  
func – Function pointer to the callback function.

Returns:

None

---

clickCallback()

Adds an event listener to this object for mouse click callback. This event is triggered whenever the mouse is pressed and released over this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.  
func – Function pointer to the callback function.

Returns:

None

---

setShape()

Sets the overall shape of the image. By default the image is a "circle".

Parameters:

shape – String of either "circle" or a "square".

Returns:

None

## Layer()

Description: A rectangle layer that can be filled with a solid color or used to add other display objects to it. Layers are extended from Osmosis.

Public members	Description
.plugin	The internal reference to the plugin.
.id	The ID number of the node internal to the plugin. This is how the plugin knows what object is being referenced.

## Public methods

None

## Scene()

Description: Each question in a demo has it's own scene. A scene contains it's own callback functions and information about the current question.

Public members	Description
.plugin	The internal reference to the plugin.
.boolStep	<b>Obsolete</b> An old method of splitting up large questions.
.strInstruction	A string of the instruction text for this question.
.initQuiz	Function pointer which is called once when the developer calls thmDemo.begin().
.loadQuiz	Function pointer which is called when a question begun.
.checkAnswer	Function pointer which is called when the user presses the "Submit" button.
.resetQuiz	Function pointer which is called when the user presses the "Reset" button.
.showCorrectAnswer	Function pointer which is called when the user presses the "Show Answer" button or answers in correctly.
.cleanUp	Function pointer which is called when the user moves on to another question.
.tries	Number of times the user can attempt to answer a question.
.correct	Boolean true if the user is correct and false otherwise.
.completed	Boolean true if the user has complete all three attempt to answer and false otherwise.
.serverStatus	Boolean true if the server has received the users answer and false otherwise.
.id	The ID number of the node internal to the plugin. This is how the plugin knows what object is being referenced.
.bgLayer	Layer which is above the background image but still below both panels. When adding a child to a scene use the .bgLayer to ensure it's placed in the right z-order.

## Public methods

addScene()

Adds a scene to the plugin so it can be displayed.

Parameters:

None

Returns:

None

---

nextScene()

Changes the display to the next scene in the display list.

Parameters:

None

Returns:

None

---

prevScene()

Changes the display to the previous scene in the display list.

Parameters:

None

Returns:

None

---

setScene()

Changes the display to the this scene in the display list.

Parameters:

None

Returns:

None

---

setTries()

Changes the number of tries left in this question.

Parameters:

tries – Number of tries to set the question to.

Returns:

None

---

decrementTries ()

Decrement the number of tries by 1.

Parameters:

None

Returns:

None

---

getTries()

Returns the number of tries left for this question.

Parameters:

None

Returns:

Number of tries left for this question.

---

setCorrect()

Sets the Boolean flag .correct to the passed value.

Parameters:

correct – Boolean value to set .correct to.

Returns:

None

---

getCorrect()

Gets the Boolean flag from .correct and returns it.

Parameters:

None

Returns:

Returns the Boolean flag from .correct.

---

setCompleted()

Sets the Boolean flag .completed to the passed value.

Parameters:

completed – Boolean value to set .correct to.

Returns:

None

---

getCompleted()

Gets the Boolean flag from .completed and returns it.

Parameters:

None

Returns:

Returns the Boolean flag from .completed.

---

setServerStatus()

Sets the Boolean flag .serverStatus to the passed value.

Parameters:

serverStatus – Boolean value to set .serverStatus to.

Returns:

None

---

getServerStatus()

Gets the Boolean flag from .serverStatus and returns it.

Parameters:

None

Returns:

Returns the Boolean flag from .serverStatus.



## BaseLabel()

Description: Label and textbox objects extends from BaseSprite for common functionality.

Public members	Description
.boolSubscribed	If true then the object can receive events otherwise if false then the object ignores events.

## Public methods

setText()

Change the text of this label

Parameters:

text – String of the what to change the label to

Returns:

None

---

subscribe()

Allows the object to receive events from the plugin

Parameters:

None

Returns:

None

---

unsubscribe()

The object ignores events from the plugin

Parameters:

None

Returns:

None

---

moveCallback()

Adds an event listener to this object for mouse move callback. This event is triggered whenever the mouse moves over this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.

func – Function pointer to the callback function.

Returns:

None

---

downCallback()

Adds an event listener to this object for mouse down callback. This event is triggered whenever the mouse clicks down on this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.

func – Function pointer to the callback function.

Returns:

None

---

overCallback()

Adds an event listener to this object for mouse over callback. This event is triggered once whenever the mouse moves over this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.  
func – Function pointer to the callback function.

Returns:

None

---

outCallback()

Adds an event listener to this object for mouse over callback. This event is triggered once whenever the mouse leaves this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.  
func – Function pointer to the callback function.

Returns:

None

---

upCallback()

Adds an event listener to this object for mouse up callback. This event is triggered whenever the mouse button is released over this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.  
func – Function pointer to the callback function.

Returns:

None

---

clickCallback()

Adds an event listener to this object for mouse click callback. This event is triggered whenever the mouse is pressed and released over this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.  
func – Function pointer to the callback function.

Returns:

None

---

setWrap()

Sets the label to use word wrapping for text longer then one line.

Parameters:

wrap – Boolean if true then word wrap is on otherwise if false no word wrap.

Returns:

None

---

addKeyListener()

Adds a key listener to this plugin.

Parameters:

None

Returns:

None

## TextBox()

Description: Textbox objects are extended from BaseLabel. They allow users to input a string text into the demo

Public members	Description
.plugin	The internal reference to the plugin.
.id	The ID number of the node internal to the plugin. This is how the plugin knows what object is being referenced.
.text	String of text displayed.

## Public methods

addEnterCallback()

Adds an event listener to this object for enter callback. This event is triggered once whenever the user pressed enter on this object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.

func – Function pointer to the callback function.

Returns:

None

---

addFocusChangedCallback()

Adds an event listener to this object for loose focus callback. This event is triggered once whenever the user changes the focus from this object to another object.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.

func – Function pointer to the callback function.

Returns:

None

---

setInteraction()

Allows the developer to disable or enable the text box. This stops the user selecting focus or typing into the textbox.

Parameters:

interaction – Boolean if true then textbox is enable else textbox is disabled

Returns:

None

## Button()

Description: Button objects are extended from BaseSprite. Buttons are special sprites with mouse events already added to them.

Public members	Description
.plugin	The internal reference to the plugin.
.id	The ID number of the node internal to the plugin. This is how the plugin knows what object is being referenced.
.type	String of the type of button to create

## Public methods

setActive()

Set the button active so it can be clicked or removes callbacks and turns it greyscale.

Parameters:

active – Boolean if true then enables callbacks else remove callbacks.

Returns:

None

## Primitive()

Description: Primitive objects are extended from Osmosis. Primitives are basic openGL shapes that can change the color and transparency.

Public members	Description
.plugin	The internal reference to the plugin.
.id	The ID number of the node internal to the plugin. This is how the plugin knows what object is being referenced.

## Public methods

setPoints ()

Set the position and dimensions of the primitive.

Parameters:

x1 – A number at the start X position of the primitive.

y1 – A number at the start Y position of the primitive.

x2 – A number at the overall width of the primitive.

y2 – A number at the overall height of the primitive.

Returns:

None

---

setCornerRadius()

Puts rounded corners on rectangles.

Parameters:

radius– A number of pixels to round a corner.

Returns:

None

## ScrollBar()

Description: A GUI object with a vertical scroll bar pre-programmed in to it.

Public members	Description
.plugin	The internal reference to the plugin.
.id	The ID number of the node internal to the plugin. This is how the plugin knows what object is being referenced.

## Public methods

None

## DropDown()

Description: A GUI object with a drop down menu pre-programmed in to it.

Public members	Description
.plugin	The internal reference to the plugin.
.id	The ID number of the node internal to the plugin. This is how the plugin knows what object is being referenced.

## Public methods

addOption()

Adds another string to the drop down menu

Parameters:

option – String of the option to add to the drop down menu.

Returns:

None

---

addChangedCallback()

Adds an event listener to this object for a changed option callback. This event is triggered once whenever the user changes from the current option to a new one.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.

func – Function pointer to the callback function.

Returns:

None

---

removeChangedCallback()

Removes an event listener from this object for a changed option callback.

Parameters:

obj – Reference value of the that will be assign to the callback's function this.

func – Function pointer to the callback function.

Returns:

None

---

removeAllCallbacks()

Removes all event listeners from this object.

Parameters:

None

Returns:

None

---

setDefaultOption()

Set the passed text as the default option for this object.

Parameters:

option – String of the option to set as the default option.

Returns:

None

---

getText()

Get the text of the currently selected option and return it as a string.

Parameters:

None

Returns:

String of the text currently selected in the drop menu.

---

setText()

Set the text of the currently selected option.

Parameters:

text - String of the text currently to set in the drop menu.

Returns:

None.

---

setColor()

Set the background color of the drop down menu.

Parameters:

r – Number of the amount of red in the object limit (0-1)

g – Number of the amount of green in the object limit (0-1)

b – Number of the amount of blue in the object limit (0-1)

a – Number of the amount of alpha in the object limit (0-1)

Returns:

None

---

**Bezier()**

Description: Draw's a bezier curve based on the pass parameters.

Public members	Description
.plugin	The internal reference to the plugin.
.id	The ID number of the node internal to the plugin. This is how the plugin knows what object is being referenced.

---

**Public methods**

setPoints()

Set the position and dimensions of the Bezier curve.

Parameters:

points – An array of points (x1, y1, x2, y2, x3, y3, ...);

Returns:

None

---

setThickness()

Changes the thickness of this Bezier curve

Parameters:

thickness – The number of pixels this Bezier curve is thick.

Returns:

None