IRENDERERTEXT SCRIPT INTERFACE

The `IRendererText` interface manages text wrapping, drawing text to a canvas, and then uploading the result to a texture. This makes it easy to display text in Construct's renderer. It is created via the IRenderer method `createWebGLText()` .

IRendererText APIs

release()

Destroy the object and its resources. `IRendererText` must be released when it is no longer needed; do not simply drop references, otherwise not all of its resources will be collected. If you are developing a plugin that creates an IRendererText, it should release any it still uses in its own `_release()` method.

fontFace

Set or get a string of the name of the font face used for drawing text.

sizePt

Set or get the size of the font, in points, used for drawing text.

lineHeight

Set or get the extra line height spacing, in pixels, used for drawing text. Note 0 is the default, indicating no offset to the default line height.

isBold

Set or get a boolean indicating whether the text is bold.

isItalic

Set pr get a boolean indicating whether the text is italic.

setColor(color)

Set the color of the text using an array with three elements for the RGB components in the range [0, 1].

SetColorRgb(r, g, b)

Set the color of the text using separate RGB components in the range [0, 1].

setCssColor(str)

Set the color of the text using a string with a CSS color, e.g. "red", "#00ffee", "rgb(0, 128, 192)" etc.

horizontalAlign

Set or get the horizontal alignment of the text within its bounding box. This can be one of `"left"` , `"center"` or `"right"` .

verticalAlign

Set the vertical alignment of the text within its bounding box. This can be one of `"top"` , `"center"` or `"bottom"` .

wordWrapMode

Set the word wrapping mode. This can be one of `"word"` (for space-delimited word wrapping) or `"character"` (for wrapping on any character).

textDirection

Set or get the text writing direction, which can be one of `"ltr"` or `"rtl"` .

text

Set or get the text string to be drawn.

setSize(width, height, zoomScale)

Set the size of the area that text can be drawn in. The size is specified in CSS pixels. The `zoomScale` can be increased to render the text at a higher resolution, which can be useful when zooming in.

getTexture()

Get an ITexture interface representing the texture with the requested text rendered on to it. **Note:** the texture is generated asynchronously, so can return `null` when first requested. Use `setTextureUpdateCallback()` to get a callback when the texture has updated.

getTexRect()

Return a DOMRect representing the content area of the text on the texture. This is the subset of the texture that ought to be rendered. Note: this is only valid when `getTexture()` returns a non-null result.

---

setTextureUpdateCallback(callback)

Set a function to call when the texture containing the rendered text is updated. Since the texture is generated asynchronously, this is necessary to know when to redraw anything that may be displaying the text, so they can redraw with the updated texture.

---

releaseTexture()

Release the underlying texture. This can be used to save memory. However the texture will be re-created the next time `getTexture()` is called.

---

textWidth
textHeight

Return the size of the text bounding box after processing word wrap. This allows determining the size of the actual visible text, rather than the box used for word wrap bounds.