TILED BACKGROUND SCRIPT INTERFACE

The `ITiledBackgroundInstance` interface derives from IWorldInstance to add APIs specific to the Tiled Background plugin.

Tiled Background APIs

imageWidth

imageHeight

getImageSize()

The original dimensions of the Tiled Background's current image in pixels. This does not include tiling - it returns the size as shown in Construct's image editor. The method allows retrieving both values at the same time.

imageOffsetX

imageOffsetY

setImageOffset(imageOffsetX, imageOffsetY)

getImageOffset()

The offset of the displayed Tiled Background image in pixels. The methods allow setting or getting both values at the same time.

imageScaleX

imageScaleY

setImageScale(imageScaleX, imageScaleY)

getImageScale()

The scale of the displayed Tiled Background image, defaulting to 1 for original size. The methods allow setting or getting both values at the same time.

imageAngle

The angle of the displayed Tiled Background image in radians. If this is changed, `imageAngleDegrees` updates accordingly.

imageAngleDegrees

The angle of the displayed Tiled Background image in degrees. If this is changed, `imageAngle` updates accordingly.

---

enableTileRandomization

A boolean indicating whether tile randomization is enabled.

---

tileXRandom

tileYRandom

setTileRandom(tileXRandom, tileYRandom)

getTileRandom()

When tile randomization is enabled, the amount of random horizontal and vertical offset to use, as a percentage in the range 0-1. The methods allow setting or getting both values at the same time.

---

tileAngleRandom

When tile randomization is enabled, the amount of random rotation to use, as a percentage in the range 0-1.

---

tileBlendMarginX

tileBlendMarginY

setTileBlendMargin(tileBlendMarginX, tileBlendMarginY)

getTileBlendMargin()

When tile randomization is enabled, the percentage of the tile width or height which will fade in to the adjacent tile, as a percentage in the range 0-1. The methods allow setting or getting both values at the same time.

---

async replaceImage(blob)

Replace the current image with the contents of a Blob representing an image file such as a PNG image. The blob can be locally generated or retrieved from a URL, for example:

```
// Loading an image from a URL
const response = await fetch(url);
const blob = await response.blob();
await tbInst.replaceImage(blob);
```