BINARY DATA SCRIPT INTERFACE

View online: https://www.construct.net/en/make-games/manuals/construct-3/scripting/scripting-reference/plugin-interfaces/binary-data

The `IBinaryDataInstance` interface derives from IInstance to add APIs specific to the Binary Data plugin. This makes it possible to read and alter the binary data from scripts, which is often more convenient than trying to do so from events.

Binary Data stores its data as an ArrayBuffer, which cannot be directly modified. In JavaScript, data can be read and written to an ArrayBuffer using typed arrays or DataView.

Binary Data APIs

setArrayBufferCopy(viewOrBuffer)

Set the contents of the Binary Data object by an ArrayBuffer or typed array which is copied. This means it is safe to continue using the passed data after this call.

setArrayBufferTransfer(arrayBuffer)

Set the contents of the Binary Data object by an ArrayBuffer which the Binary Data object takes ownership of. You must not use the passed ArrayBuffer after this call, since it is now managed by the Binary Data object. Since this method does not copy the passed ArrayBuffer, it is more efficient if the ArrayBuffer won't be used again in your code. Note this method does not accept a typed array.

getArrayBufferCopy()

Return the contents of the Binary Data object as an ArrayBuffer which is a copy of the internal ArrayBuffer. This means it is safe to modify the returned ArrayBuffer without affecting the state of the Binary Data object.

getArrayBufferReadOnly()

Return the contents of the Binary Data object as a read-only reference to the internal ArrayBuffer. This ArrayBuffer must not be modified since it is managed by the Binary Data object. However it is more efficient to use this method if the data is only read from, e.g. to send over the network, since it does not copy the ArrayBuffer.