This is a private alpha demonstration of the Lightning Game Engine SDK version 2.0. This version is not intended for development use and is intended to demonstrate the various new functionalities and higher speed of the Lightning Game Engine version 2.0. Lightning is a powerful and easy to use 2D game engine.

**Warning: Anything present in this version may be subject to change before the final release! Do not use the API documentation present in this release, as it is extremely outdated and will be rewritten before the final release of SDK Version 2.0**

**Incompatibility Warning for Lightning Version 1.x Projects**

Lightning projects targeted at version 1.1.x are not compatible in **any way** with Lightning version 2.0. **100% of API calls have changed**, even the **basic initialisation call**, and some of the basic facets of the engine’s design have been changed from version 1.x. Even basic classes have been renamed. There is a possibility more will diverge before the final release of version 2.0.

**There is no point porting your projects. They are incompatible.** If you are currently developing a project using Lightning 1.1.x, the **1.1.4 patch** has been **cancelled** as **pointless** due to the low number of users. If you are using Lightning 1.1.3, continue to use Lightning 1.1.3. **No support will be provided as of December 4, 2022**.

**Lightning 1.0.5** is the **final** 1.0.x version, and **no** support will be provided for version 1.0.x in **any capacity** after **October 29, 2022**.

**Highlights**

* SDL\_gfx is dead
* SDL\_ttf backend replaced with Freetype2, far more reasonable text api
* Object Hierarchy
* Massively improved debugging functionality (console and new debug displays)
* Far more debug settings
* Support for multiple renderers
* Updated to the latest SDL
* A far more consistent API (no more render/draw confusion)
* Load defaults on failure (for example: a missing texture texture)
* New events
* Standard renderable API
* All renderables have names
* Better culling
* Configurable audio settings (to some extent)
* Renderer-independent Renderables
* Everything is a Renderable
* Client/server separation
* Much more!

**Features not yet implemented**

* Network Services
* Rotating textures
* Multitexturing
* Automatic globalsettings saving
* Controller Support
* Fade Animation
* Many, many bugfixes
* Anything else I forgot for v2.0! v2.5 is when I pass my exams