

# What is the Vulkan API?

A brief summary of the Vulkan API and its history

# What is the Vulkan API?

- **A low-level, cross-platform, graphics and computing API.**
- **Low-level: Vulkan is VERY verbose! Unlike OpenGL, handles almost every individual GPU process manually.**
- **Cross-platform: Unlike Direct3D, it can run on any platform!**
  - However, it requires additional work to operate on macOS and iOS:  
<https://moltengl.com/moltenvk/>
- **Graphics:** Used for processing graphical data on the GPU (both online + offline rendering).
- **Computing:** Can also be used for generic parallel programming tasks, much like OpenCL.

# History of Vulkan and OpenGL

- **What is OpenGL?**
  - OpenGL (Open Graphics Library) is cross-platform graphics API.
  - Cross-platform alternative to Direct3D.
- **Before Vulkan announced: OpenGL “sequel” expected.**
  - Often referred to as “glNext”
- **In 2015, Vulkan API announced as OpenGL successor**
  - Said to be much lower-level than OpenGL and Direct3D 12
- **Supported by Valve through group called LunarG**
- **Vulkan 1.1 released in 2018**

# “Vulkan is very verbose!”

- **Vulkan commonly said to be very “verbose”**
- **This means there is *a lot* of code.**
- **Understandable! GPUs are complex, and Vulkan gives access to all that complexity.**
  - Why? Because more control = more optimisation!
- **Can take up to 1,000 lines of code to draw a triangle!**
- **However, most code is just setting values on GPU and creating objects. Logically very simple code.**

## Important Note About Memory

- This course is not about Memory Management.
- Optimal Vulkan programs require good memory management.
- If you want to get the most out of Vulkan, learn to manage memory efficiently!

## Next Lesson...

- Setup and test code!
- Won't be teaching how anything works.
- Just ensuring Vulkan works on your system.

# Summary

- Vulkan is a low-level, cross-platform, graphics and computing API.
- Successor to OpenGL.
- Very verbose! There will be a lot of code (but mostly very simple code).

**See you next video!**