VRE::VRE_Window
- int mWidth
- int mHeight
- bool mFrameBufferResized
- const std::string mWindowName
- GLFWwindow * mWindow
+ VRE_Window(int width, int height, std::string name)
+ ~VRE_Window()
+ VRE_Window(const VRE _Window &)=delete
+ VRE_Window & operator =(const VRE_Window &) =delete
+ bool ShouldClose()
+ void CreateWindowSurface (VkInstance instance, VkSurface KHR *surface)
+ VkExtent2D GetExtent()
+ bool HasWindowResized()
+ void ResetWindowResizedFlag()
+ GLFWwindow * GetGLFWwindow () const
- void Init()
- static void FrameBufferResized Callback(GLFWwindow *window, int width, int height)
-mWindow
-mWindow
VRE::VRE_Device
+ const bool mEnableValidation Layers
+ VkPhysicalDeviceProperties mProperties
- VkInstance mInstance
- VkDebugUtilsMessengerEXT mDebugMessenger
- VkPhysicalDevice mPhysical Device
- VkCommandPool mCommandPool
- VkDevice mVkDevice
- VkSurfaceKHR mVkSurface
- VkQueue mGraphicsQueue
- VkQueue mPresentQueue
- const std::vector< const char * > mValidation Layers
- const std::vector< const char * > mDeviceExtensions
+ VRE_Device(VRE_Window &window)
+ ~VRE_Device()
+ VRE_Device(const VRE _Device &)=delete
+ VRE_Device & operator =(const VRE_Device &) =delete
+ VRE_Device(VRE_Device &&)=delete
+ VRE_Device & operator =(VRE_Device &&)=delete
+ VkCommandPool GetCommand Pool()
+ VkDevice GetVkDevice()
+ VkPhysicalDevice GetPhysical Device()
+ VkSurfaceKHR Surface()
and 12 more...
- void CreateInstance()
- void SetupDebugMessenger()
- void CreateSurface()
- void PickPhysicalDevice()
- void CreateLogicalDevice()
- void CreateCommandPool()
- bool IsDeviceSuitable (VkPhysicalDevice device)
- std::vector< const char * > GetRequiredExtensions()
- bool CheckValidationLayer Support()
- QueueFamilyIndices FindQueueFamilies(VkPhysical Device device)
- void PopulateDebugMessenger CreateInfo(VkDebugUtilsMessenger CreateInfoEXT &createInfo)
- void HasGflwRequiredInstance Extensions()
- bool CheckDeviceExtension Support(VkPhysicalDevice device)
- SwapChainSupportDetails QuerySwapChainSupport (VkPhysicalDevice device)
-mDevice
VRE::VRE_Renderer
- std::unique_ptr< VRE _SwapChain > mSwapChain
- std::vector< VkCommandBuffer > mCommandBuffers
- uint32_t mCurrentImageIndex
- int mCurrentFrameIndex
- bool mDrawStarted
+ VRE_Renderer(VRE_Window &window, VRE_Device &device)
+ ~VRE_Renderer()
+ VRE_Renderer(const VRE_Renderer &)=delete
+ VRE_Renderer & operator =(const VRE_Renderer &) =delete
+ VkCommandBuffer BeginDraw()
+ void EndDraw()
+ void BeginSwapChainRender Pass(VkCommandBuffer commandBuffer)
+ void EndSwapChainRenderPass (VkCommandBuffer commandBuffer)
+ bool IsDrawInProgress () const
+ VkRenderPass GetSwapChain RenderPass() const
+ float GetAspectRatio () const
+ VkCommandBuffer GetCurrent CommandBuffer() const
+ int GetFrameIndex() const
- void CreateCommandBuffers()
- void RecreateSwapChain()
- void FreeCommandBuffers()
+mDevice
+mRenderer
VRE::VRE_SharedContext
+ std::shared_ptr< VRE _DescriptorSetLayout > mGlobalDescSetLayout
+ std::vector< std::unique _ptr< VRE_DescriptorPool > > mObjectDescPools
+ std::vector< VkDescriptor Set > mSceneDescriptorSets
+ std::vector< VRE_PointLight > mPointLights
+ VRE_GameObject::GameObjects Map * mGameObjMap
+ VRE_SharedContext()