

mugen_engine::MEGraphicDevice
<ul style="list-style-type: none"> - Microsoft::WRL::ComPtr< ID3D12Device > m_device - Microsoft::WRL::ComPtr< IDXGIFactory6 > m_dxgiFactory
<ul style="list-style-type: none"> + MEGraphicDevice() + MEGraphicDevice(const MEGraphicDevice &)=delete + void Initialize() + ID3D12Device *const GetDevice() const + IDXGIFactory4 *const GetFactory() const - void _EnableDebugLayer()

mugen_engine::MEGraphicCommandList
<ul style="list-style-type: none"> - Microsoft::WRL::ComPtr< ID3D12Command Allocator > m_cmdAllocator - Microsoft::WRL::ComPtr< ID3D12Graphics CommandList > m_cmdList - Microsoft::WRL::ComPtr< ID3D12Command Queue > m_cmdQueue - Microsoft::WRL::ComPtr< ID3D12Fence > m_fence - UINT64 m_fenceVal
<ul style="list-style-type: none"> + MEGraphicCommandList() + void Initialize(const MEGraphicDevice &device) + void Execute() + ID3D12CommandQueue *const GetCommandQueue() const + ID3D12GraphicsCommandList *const GetCommandList() const

mugen_engine::MEGraphicPipeline
<ul style="list-style-type: none"> - std::vector< char > m_vsBlob - std::vector< char > m_psBlob - Microsoft::WRL::ComPtr< ID3DBlob > m_errorBlob - Microsoft::WRL::ComPtr< ID3D12Pipeline State > m_pipelineState - Microsoft::WRL::ComPtr< ID3D12Root Signature > m_rootSignature
<ul style="list-style-type: none"> + MEGraphicPipeline() + void Initialize(const MEGraphicDevice &device, const D3D12_INPUT_ELEMENT_DESC inputLayout[], const int layoutSize) + void SetPipelineState(const int type, MEGraphicCommandList &cmdList) - void _ProcessBlobError(HRESULT result) - void _CreateRootSignarure(const MEGraphicDevice &device) - void _LoadShader() - void _CreatePipelineState(const MEGraphicDevice &device, const D3D12_INPUT_ELEMENT_DESC inputLayout[], const int layoutSize)

mugen_engine::MEGraphicRenderTarget
<ul style="list-style-type: none"> - const int m_numBackBuffer - Microsoft::WRL::ComPtr< DXGI SwapChain4 > m_swapchain - Microsoft::WRL::ComPtr< ID3D12Descriptor Heap > m_rtvHeaps - std::vector< Microsoft::WRL::ComPtr< ID3D12Resource > > m_backBuffers - D3D12_CPU_DESCRIPTOR_HANDLE m_renderTargetHandle - D3D12_VIEWPORT m_viewport - D3D12_RECT m_scissorRect
<ul style="list-style-type: none"> + MEGraphicRenderTarget() + void Initialize(const MEGraphicDevice &device, const MEGraphicCommandList &cmdList, HWND hwnd, const int window_width, const int window_height) + void Present() + void SetBarrierBeforeRender(MEGraphicDevice &device, MEGraphicCommandList &cmdList) + void SetBarrierBeforePresent(MEGraphicCommandList &cmdList) + void Clear(float clearColor[4], MEGraphicCommandList &cmdList) + void SetRenderArea(MEGraphicCommandList &cmdList, const int topX, const int topY, const int bottomX, const int bottomY) + void SetRenderBaseCommand(MEGraphicCommandList &cmdList)

mugen_engine::MEFontData
<ul style="list-style-type: none"> - HDC m_hdc - HFONT m_oldFont - std::unordered_map< wchar_t, MEGraphicCharacterUnit > m_loadedCharacters
<ul style="list-style-type: none"> + MEFontData() + MEFontData(std::wstring fontName, int fontSize, MEGraphicDevice &device, MEGraphicCommandList &cmdList, MEGraphicPipeline &pipeline, MEGraphicRenderTarget &renderTarget) + MEFontData & operator=(const MEFontData &rhs) + void DrawString(const int x, const int y, const float color[4], float priority, const std::wstring text) + void DrawFormatString(const int x, const int y, const float color[4], float priority, const std::wstring text,...)

