

[Metal](#) / MTL4ArgumentTableDescriptor

Class


MTL4ArgumentTableDescriptor

Groups parameters for the creation of a Metal argument table.

iOS 26.0+ | iPadOS 26.0+ | Mac Catalyst 26.0+ | macOS 26.0+ | tvOS 26.0+ | visionOS 26.0+

```
class MTL4ArgumentTableDescriptor
```

Mentioned in

 Understanding the Metal 4 core API

Overview

Argument tables provide resource bindings to your Metal pipeline states.

Topics

Instance Properties

`var initializeBindings: Bool`

Configures whether Metal initializes the bindings to nil values upon creation of argument table.

`var label: String?`

Assigns an optional label with the argument table for debug purposes.

`var maxBufferBindCount: Int`

Determines the number of buffer-binding slots for the argument table.

`var maxSamplerStateBindCount: Int`

Determines the number of sampler state-binding slots for the argument table.

`var maxTextureBindCount: Int`

Determines the number of texture-binding slots for the argument table.

`var supportAttributeStrides: Bool`

Controls whether Metal should reserve memory for attribute strides in the argument table.

Relationships

Inherits From

`NSObject`

Conforms To

`CVarArg`

`CustomDebugStringConvertible`

`CustomStringConvertible`

`Equatable`

`Hashable`

`NSCopying`

`NSObjectProtocol`

See Also

Submitting work to a GPU with Metal 4

`protocol MTL4CommandQueue`

An abstraction representing a command queue that you use commit and synchronize command buffers and to perform other GPU operations.

`class MTL4CommandQueueDescriptor`

Groups together parameters for the creation of a new command queue.

`struct MTL4CommandQueueError`

`enum Code`

Enumeration of kinds of errors that committing an array of command buffers instances can produce.

`let MTL4CommandQueueErrorDomain: String`

`protocol MTL4CommandBuffer`

Records a sequence of GPU commands.

`class MTL4CommandBufferOptions`

Options to configure a command buffer before encoding work into it.

`protocol MTL4CommandEncoder`

An encoder that writes GPU commands into a command buffer.

`struct MTL4RenderEncoderOptions`

Custom render pass options you specify at encoder creation time.

`protocol MTL4ArgumentTable`

Provides a mechanism to manage and provide resource bindings for buffers, textures, sampler states and other Metal resources.

`protocol MTL4CommandAllocator`

Manages the memory backing the encoding of GPU commands into command buffers.

`class MTL4CommandAllocatorDescriptor`

Groups together parameters for creating a command allocator.

`class MTL4CommitOptions`

Represents options to configure a commit operation on a command queue.

`protocol MTL4CommitFeedback`

Describes an object containing debug information from Metal to your app after completing a workload.

`typealias MTL4CommitFeedbackHandler`

Defines the block signature for a callback Metal invokes to provide your app feedback after completing a workload.

