deno apideno api

目录

[1.1 Broadcast Channel 1](#_Toc114769980)

[1.2 Compression Streams API 1](#_Toc114769981)

[1.3 Console and Debugging 1](#_Toc114769982)

[1.4 DOM APIs 1](#_Toc114769983)

[1.5 DOM Events 2](#_Toc114769984)

[1.6 Encoding API 3](#_Toc114769985)

[1.7 Errors 4](#_Toc114769986)

[1.8 ES Modules 5](#_Toc114769987)

[1.9 Fetch API 5](#_Toc114769988)

[1.10 File System 6](#_Toc114769989)

[1.11 HTTP Server 11](#_Toc114769990)

[1.12 I/O 12](#_Toc114769991)

[1.13 Network 14](#_Toc114769992)

[1.14 Observability 16](#_Toc114769993)

[1.15 Performance API 16](#_Toc114769994)

[1.16 Permissions 17](#_Toc114769995)

[1.17 Runtime Environment 18](#_Toc114769996)

[1.18 Scheduling 19](#_Toc114769997)

[1.19 Streams API 19](#_Toc114769998)

[1.20 Sub Process 22](#_Toc114769999)

[1.21 Testing 22](#_Toc114770000)

[1.22 Timers 23](#_Toc114770001)

[1.23 Typed Arrays 23](#_Toc114770002)

[1.24 Web APIs 23](#_Toc114770003)

[1.25 Web Crypto API 25](#_Toc114770004)

[1.26 Web File API 27](#_Toc114770005)

[1.27 Web Sockets 27](#_Toc114770006)

[1.28 Web Storage API 28](#_Toc114770007)

[1.29 Web Workers 28](#_Toc114770008)

[1.30 WebAssembly 28](#_Toc114770009)

[1.31 WebGPU 31](#_Toc114770010)

[1.32 Uncategorized 37](#_Toc114770011)

There are APIs that are built into the Deno CLI that are beyond those that are built-ins for JavaScript. They are a combination of web platform APIs Deno has implemented and Deno specific APIs.

We try to keep non-standard, Deno specific, APIs in the {@link Deno} namespace. We have grouped the APIs into the following functional categories.

## [Broadcast Channel](https://deno.land/api@v1.25.3#Broadcast_Channel)

|  |  |
| --- | --- |
| I  [**BroadcastChannel**](https://deno.land/api@v1.25.3?s=BroadcastChannel) |  |
| I  [**BroadcastChannelEventMap**](https://deno.land/api@v1.25.3?s=BroadcastChannelEventMap) |  |
| v  [**BroadcastChannel**](https://deno.land/api@v1.25.3?s=BroadcastChannel) |  |

## [Compression Streams API](https://deno.land/api@v1.25.3#Compression_Streams_API)

|  |  |
| --- | --- |
| c  [**CompressionStream**](https://deno.land/api@v1.25.3?s=CompressionStream) | An API for compressing a stream of data. |
| c  [**DecompressionStream**](https://deno.land/api@v1.25.3?s=DecompressionStream) | An API for decompressing a stream of data. |

## [Console and Debugging](https://deno.land/api@v1.25.3#Console_and_Debugging)

|  |  |
| --- | --- |
| I  [**Console**](https://deno.land/api@v1.25.3?s=Console) |  |
| I  [**Deno.InspectOptions**](https://deno.land/api@v1.25.3?s=Deno.InspectOptions) |  |
| v  [**console**](https://deno.land/api@v1.25.3?s=console) |  |
| v  [**Deno.customInspect**](https://deno.land/api@v1.25.3?s=Deno.customInspect)  deprecated | A symbol which can be used as a key for a custom method which will be called when Deno.inspect() is called, or when the object is logged to the console. |
| f  [**Deno.inspect**](https://deno.land/api@v1.25.3?s=Deno.inspect) | Converts the input into a string that has the same format as printed by console.log(). |

## [DOM APIs](https://deno.land/api@v1.25.3#DOM_APIs)

|  |  |
| --- | --- |
| c  [**MessageChannel**](https://deno.land/api@v1.25.3?s=MessageChannel) | The MessageChannel interface of the Channel Messaging API allows us to create a new message channel and send data through it via its two MessagePort properties. |
| c  [**MessagePort**](https://deno.land/api@v1.25.3?s=MessagePort) | The MessagePort interface of the Channel Messaging API represents one of the two ports of a MessageChannel, allowing messages to be sent from one port and listening out for them arriving at the other. |
| I  [**DomIterable**](https://deno.land/api@v1.25.3?s=DomIterable) |  |
| I  [**DOMStringList**](https://deno.land/api@v1.25.3?s=DOMStringList) |  |
| I  [**ErrorConstructor**](https://deno.land/api@v1.25.3?s=ErrorConstructor) |  |
| I  [**MessagePortEventMap**](https://deno.land/api@v1.25.3?s=MessagePortEventMap) |  |
| I  [**StructuredSerializeOptions**](https://deno.land/api@v1.25.3?s=StructuredSerializeOptions) |  |
| T  [**PostMessageOptions**](https://deno.land/api@v1.25.3?s=PostMessageOptions)  deprecated | This type has been renamed to StructuredSerializeOptions. Use that type for new code. |
| T  [**Transferable**](https://deno.land/api@v1.25.3?s=Transferable) |  |
| f  [**structuredClone**](https://deno.land/api@v1.25.3?s=structuredClone) | Creates a deep copy of a given value using the structured clone algorithm. |

## [DOM Events](https://deno.land/api@v1.25.3#DOM_Events)

|  |  |
| --- | --- |
| c  [**CustomEvent**](https://deno.land/api@v1.25.3?s=CustomEvent) |  |
| c  [**DOMException**](https://deno.land/api@v1.25.3?s=DOMException) |  |
| c  [**ErrorEvent**](https://deno.land/api@v1.25.3?s=ErrorEvent) |  |
| c  [**Event**](https://deno.land/api@v1.25.3?s=Event) | An event which takes place in the DOM. |
| c  [**EventTarget**](https://deno.land/api@v1.25.3?s=EventTarget) | EventTarget is a DOM interface implemented by objects that can receive events and may have listeners for them. |
| c  [**ProgressEvent**](https://deno.land/api@v1.25.3?s=ProgressEvent) | Events measuring progress of an underlying process, like an HTTP request (for an XMLHttpRequest, or the loading of the underlying resource of an , , , or ). |
| I  [**AddEventListenerOptions**](https://deno.land/api@v1.25.3?s=AddEventListenerOptions) |  |
| I  [**CustomEventInit**](https://deno.land/api@v1.25.3?s=CustomEventInit) |  |
| I  [**ErrorEventInit**](https://deno.land/api@v1.25.3?s=ErrorEventInit) |  |
| I  [**EventInit**](https://deno.land/api@v1.25.3?s=EventInit) |  |
| I  [**EventListener**](https://deno.land/api@v1.25.3?s=EventListener) |  |
| I  [**EventListenerObject**](https://deno.land/api@v1.25.3?s=EventListenerObject) |  |
| I  [**EventListenerOptions**](https://deno.land/api@v1.25.3?s=EventListenerOptions) |  |
| I  [**ProgressEventInit**](https://deno.land/api@v1.25.3?s=ProgressEventInit) |  |
| T  [**EventListenerOrEventListenerObject**](https://deno.land/api@v1.25.3?s=EventListenerOrEventListenerObject) |  |
| v  [**onerror**](https://deno.land/api@v1.25.3?s=onerror) |  |
| v  [**onload**](https://deno.land/api@v1.25.3?s=onload) |  |
| v  [**onunload**](https://deno.land/api@v1.25.3?s=onunload) |  |
| f  [**addEventListener**](https://deno.land/api@v1.25.3?s=addEventListener) | Registers an event listener in the global scope, which will be called synchronously whenever the event type is dispatched. |
| f  [**dispatchEvent**](https://deno.land/api@v1.25.3?s=dispatchEvent) | Dispatches an event in the global scope, synchronously invoking any registered event listeners for this event in the appropriate order. Returns false if event is cancelable and at least one of the event handlers which handled this event called Event.preventDefault(). Otherwise it returns true. |
| f  [**removeEventListener**](https://deno.land/api@v1.25.3?s=removeEventListener) | Remove a previously registered event listener from the global scope |

## [Encoding API](https://deno.land/api@v1.25.3#Encoding_API)

|  |  |
| --- | --- |
| I  [**TextDecodeOptions**](https://deno.land/api@v1.25.3?s=TextDecodeOptions) |  |
| I  [**TextDecoder**](https://deno.land/api@v1.25.3?s=TextDecoder) |  |
| I  [**TextDecoderOptions**](https://deno.land/api@v1.25.3?s=TextDecoderOptions) |  |
| I  [**TextDecoderStream**](https://deno.land/api@v1.25.3?s=TextDecoderStream) |  |
| I  [**TextEncoder**](https://deno.land/api@v1.25.3?s=TextEncoder) |  |
| I  [**TextEncoderEncodeIntoResult**](https://deno.land/api@v1.25.3?s=TextEncoderEncodeIntoResult) |  |
| I  [**TextEncoderStream**](https://deno.land/api@v1.25.3?s=TextEncoderStream) |  |
| v  [**TextDecoder**](https://deno.land/api@v1.25.3?s=TextDecoder) |  |
| v  [**TextDecoderStream**](https://deno.land/api@v1.25.3?s=TextDecoderStream) |  |
| v  [**TextEncoder**](https://deno.land/api@v1.25.3?s=TextEncoder) |  |
| v  [**TextEncoderStream**](https://deno.land/api@v1.25.3?s=TextEncoderStream) |  |
| f  [**atob**](https://deno.land/api@v1.25.3?s=atob) | Decodes a string of data which has been encoded using base-64 encoding. |
| f  [**btoa**](https://deno.land/api@v1.25.3?s=btoa) | Creates a base-64 ASCII encoded string from the input string. |

## [Errors](https://deno.land/api@v1.25.3#Errors)

|  |  |
| --- | --- |
| N  [**Deno.errors**](https://deno.land/api@v1.25.3?s=Deno.errors) | A set of error constructors that are raised by Deno APIs. |
| c  [**Deno.errors.AddrInUse**](https://deno.land/api@v1.25.3?s=Deno.errors.AddrInUse) |  |
| c  [**Deno.errors.AddrNotAvailable**](https://deno.land/api@v1.25.3?s=Deno.errors.AddrNotAvailable) |  |
| c  [**Deno.errors.AlreadyExists**](https://deno.land/api@v1.25.3?s=Deno.errors.AlreadyExists) |  |
| c  [**Deno.errors.BadResource**](https://deno.land/api@v1.25.3?s=Deno.errors.BadResource) |  |
| c  [**Deno.errors.BrokenPipe**](https://deno.land/api@v1.25.3?s=Deno.errors.BrokenPipe) |  |
| c  [**Deno.errors.Busy**](https://deno.land/api@v1.25.3?s=Deno.errors.Busy) |  |
| c  [**Deno.errors.ConnectionAborted**](https://deno.land/api@v1.25.3?s=Deno.errors.ConnectionAborted) |  |
| c  [**Deno.errors.ConnectionRefused**](https://deno.land/api@v1.25.3?s=Deno.errors.ConnectionRefused) |  |
| c  [**Deno.errors.ConnectionReset**](https://deno.land/api@v1.25.3?s=Deno.errors.ConnectionReset) |  |
| c  [**Deno.errors.Http**](https://deno.land/api@v1.25.3?s=Deno.errors.Http) |  |
| c  [**Deno.errors.Interrupted**](https://deno.land/api@v1.25.3?s=Deno.errors.Interrupted) |  |
| c  [**Deno.errors.InvalidData**](https://deno.land/api@v1.25.3?s=Deno.errors.InvalidData) |  |
| c  [**Deno.errors.NotConnected**](https://deno.land/api@v1.25.3?s=Deno.errors.NotConnected) |  |
| c  [**Deno.errors.NotFound**](https://deno.land/api@v1.25.3?s=Deno.errors.NotFound) |  |
| c  [**Deno.errors.NotSupported**](https://deno.land/api@v1.25.3?s=Deno.errors.NotSupported) |  |
| c  [**Deno.errors.PermissionDenied**](https://deno.land/api@v1.25.3?s=Deno.errors.PermissionDenied) |  |
| c  [**Deno.errors.TimedOut**](https://deno.land/api@v1.25.3?s=Deno.errors.TimedOut) |  |
| c  [**Deno.errors.UnexpectedEof**](https://deno.land/api@v1.25.3?s=Deno.errors.UnexpectedEof) |  |
| c  [**Deno.errors.WriteZero**](https://deno.land/api@v1.25.3?s=Deno.errors.WriteZero) |  |

## [ES Modules](https://deno.land/api@v1.25.3#ES_Modules)

|  |  |
| --- | --- |
| I  [**ImportMeta**](https://deno.land/api@v1.25.3?s=ImportMeta) | Deno provides extra properties on import.meta. These are included here to ensure that these are still available when using the Deno namespace in conjunction with other type libs, like dom. |

## [Fetch API](https://deno.land/api@v1.25.3#Fetch_API)

|  |  |
| --- | --- |
| c  [**Headers**](https://deno.land/api@v1.25.3?s=Headers) |  |
| c  [**Request**](https://deno.land/api@v1.25.3?s=Request) | This Fetch API interface represents a resource request. |
| c  [**Response**](https://deno.land/api@v1.25.3?s=Response) | This Fetch API interface represents the response to a request. |
| I  [**Body**](https://deno.land/api@v1.25.3?s=Body) |  |
| I  [**FormData**](https://deno.land/api@v1.25.3?s=FormData) | Provides a way to easily construct a set of key/value pairs representing form fields and their values, which can then be easily sent using the XMLHttpRequest.send() method. It uses the same format a form would use if the encoding type were set to "multipart/form-data". |
| I  [**Headers**](https://deno.land/api@v1.25.3?s=Headers) | This Fetch API interface allows you to perform various actions on HTTP request and response headers. These actions include retrieving, setting, adding to, and removing. A Headers object has an associated header list, which is initially empty and consists of zero or more name and value pairs. You can add to this using methods like append() (see Examples). In all methods of this interface, header names are matched by case-insensitive byte sequence. |
| I  [**RequestInit**](https://deno.land/api@v1.25.3?s=RequestInit) |  |
| I  [**ResponseInit**](https://deno.land/api@v1.25.3?s=ResponseInit) |  |
| T  [**BodyInit**](https://deno.land/api@v1.25.3?s=BodyInit) |  |
| T  [**FormDataEntryValue**](https://deno.land/api@v1.25.3?s=FormDataEntryValue) |  |
| T  [**HeadersInit**](https://deno.land/api@v1.25.3?s=HeadersInit) |  |
| T  [**ReferrerPolicy**](https://deno.land/api@v1.25.3?s=ReferrerPolicy) |  |
| T  [**RequestCache**](https://deno.land/api@v1.25.3?s=RequestCache) |  |
| T  [**RequestCredentials**](https://deno.land/api@v1.25.3?s=RequestCredentials) |  |
| T  [**RequestDestination**](https://deno.land/api@v1.25.3?s=RequestDestination) |  |
| T  [**RequestInfo**](https://deno.land/api@v1.25.3?s=RequestInfo) |  |
| T  [**RequestMode**](https://deno.land/api@v1.25.3?s=RequestMode) |  |
| T  [**RequestRedirect**](https://deno.land/api@v1.25.3?s=RequestRedirect) |  |
| T  [**ResponseType**](https://deno.land/api@v1.25.3?s=ResponseType) |  |
| v  [**FormData**](https://deno.land/api@v1.25.3?s=FormData) |  |
| f  [**fetch**](https://deno.land/api@v1.25.3?s=fetch) | Fetch a resource from the network. It returns a Promise that resolves to the Response to that Request, whether it is successful or not. |

## [File System](https://deno.land/api@v1.25.3#File_System)

|  |  |
| --- | --- |
| c  [**Deno.File**](https://deno.land/api@v1.25.3?s=Deno.File)  deprecated | The Deno abstraction for reading and writing files. |
| c  [**Deno.FsFile**](https://deno.land/api@v1.25.3?s=Deno.FsFile) | The Deno abstraction for reading and writing files. |
| I  [**Deno.DirEntry**](https://deno.land/api@v1.25.3?s=Deno.DirEntry) |  |
| I  [**Deno.FileInfo**](https://deno.land/api@v1.25.3?s=Deno.FileInfo) | A FileInfo describes a file and is returned by stat, lstat, statSync, lstatSync. |
| I  [**Deno.FsEvent**](https://deno.land/api@v1.25.3?s=Deno.FsEvent) |  |
| I  [**Deno.FsWatcher**](https://deno.land/api@v1.25.3?s=Deno.FsWatcher) | FsWatcher is returned by Deno.watchFs function when you start watching the file system. You can iterate over this interface to get the file system events, and also you can stop watching the file system by calling .close() method. |
| I  [**Deno.MakeTempOptions**](https://deno.land/api@v1.25.3?s=Deno.MakeTempOptions) |  |
| I  [**Deno.MkdirOptions**](https://deno.land/api@v1.25.3?s=Deno.MkdirOptions) |  |
| I  [**Deno.OpenOptions**](https://deno.land/api@v1.25.3?s=Deno.OpenOptions) |  |
| I  [**Deno.ReadFileOptions**](https://deno.land/api@v1.25.3?s=Deno.ReadFileOptions) |  |
| I  [**Deno.RemoveOptions**](https://deno.land/api@v1.25.3?s=Deno.RemoveOptions) |  |
| I  [**Deno.WriteFileOptions**](https://deno.land/api@v1.25.3?s=Deno.WriteFileOptions) | Options for writing to a file. |
| T  [**Deno.FsEventFlag**](https://deno.land/api@v1.25.3?s=Deno.FsEventFlag) | Additional information for FsEvent objects with the "other" kind. |
| T  [**Deno.SymlinkOptions**](https://deno.land/api@v1.25.3?s=Deno.SymlinkOptions) |  |
| f  [**Deno.chmod**](https://deno.land/api@v1.25.3?s=Deno.chmod) | Changes the permission of a specific file/directory of specified path. Ignores the process's umask. |
| f  [**Deno.chmodSync**](https://deno.land/api@v1.25.3?s=Deno.chmodSync) | Synchronously changes the permission of a specific file/directory of specified path. Ignores the process's umask. |
| f  [**Deno.chown**](https://deno.land/api@v1.25.3?s=Deno.chown) | Change owner of a regular file or directory. This functionality is not available on Windows. |
| f  [**Deno.chownSync**](https://deno.land/api@v1.25.3?s=Deno.chownSync) | Synchronously change owner of a regular file or directory. This functionality is not available on Windows. |
| f  [**Deno.copyFile**](https://deno.land/api@v1.25.3?s=Deno.copyFile) | Copies the contents and permissions of one file to another specified path, by default creating a new file if needed, else overwriting. Fails if target path is a directory or is unwritable. |
| f  [**Deno.copyFileSync**](https://deno.land/api@v1.25.3?s=Deno.copyFileSync) | Synchronously copies the contents and permissions of one file to another specified path, by default creating a new file if needed, else overwriting. Fails if target path is a directory or is unwritable. |
| f  [**Deno.create**](https://deno.land/api@v1.25.3?s=Deno.create) | Creates a file if none exists or truncates an existing file and resolves to an instance of Deno.FsFile. |
| f  [**Deno.createSync**](https://deno.land/api@v1.25.3?s=Deno.createSync) | Creates a file if none exists or truncates an existing file and returns an instance of Deno.FsFile. |
| f  [**Deno.fstat**](https://deno.land/api@v1.25.3?s=Deno.fstat) | Returns a Deno.FileInfo for the given file stream. |
| f  [**Deno.fstatSync**](https://deno.land/api@v1.25.3?s=Deno.fstatSync) | Synchronously returns a Deno.FileInfo for the given file stream. |
| f  [**Deno.ftruncate**](https://deno.land/api@v1.25.3?s=Deno.ftruncate) | Truncates or extends the specified file stream, to reach the specified len. |
| f  [**Deno.ftruncateSync**](https://deno.land/api@v1.25.3?s=Deno.ftruncateSync) | Synchronously truncates or extends the specified file stream, to reach the specified len. |
| f  [**Deno.link**](https://deno.land/api@v1.25.3?s=Deno.link) | Creates newpath as a hard link to oldpath. |
| f  [**Deno.linkSync**](https://deno.land/api@v1.25.3?s=Deno.linkSync) | Synchronously creates newpath as a hard link to oldpath. |
| f  [**Deno.lstat**](https://deno.land/api@v1.25.3?s=Deno.lstat) | Resolves to a Deno.FileInfo for the specified path. If path is a symlink, information for the symlink will be returned instead of what it points to. |
| f  [**Deno.lstatSync**](https://deno.land/api@v1.25.3?s=Deno.lstatSync) | Synchronously returns a Deno.FileInfo for the specified path. If path is a symlink, information for the symlink will be returned instead of what it points to.. |
| f  [**Deno.makeTempDir**](https://deno.land/api@v1.25.3?s=Deno.makeTempDir) | Creates a new temporary directory in the default directory for temporary files, unless dir is specified. Other optional options include prefixing and suffixing the directory name with prefix and suffix respectively. |
| f  [**Deno.makeTempDirSync**](https://deno.land/api@v1.25.3?s=Deno.makeTempDirSync) | Synchronously creates a new temporary directory in the default directory for temporary files, unless dir is specified. Other optional options include prefixing and suffixing the directory name with prefix and suffix respectively. |
| f  [**Deno.makeTempFile**](https://deno.land/api@v1.25.3?s=Deno.makeTempFile) | Creates a new temporary file in the default directory for temporary files, unless dir is specified. Other optional options include prefixing and suffixing the directory name with prefix and suffix respectively. |
| f  [**Deno.makeTempFileSync**](https://deno.land/api@v1.25.3?s=Deno.makeTempFileSync) | Synchronously creates a new temporary file in the default directory for temporary files, unless dir is specified. Other optional options include prefixing and suffixing the directory name with prefix and suffix respectively. |
| f  [**Deno.mkdir**](https://deno.land/api@v1.25.3?s=Deno.mkdir) | Creates a new directory with the specified path. |
| f  [**Deno.mkdirSync**](https://deno.land/api@v1.25.3?s=Deno.mkdirSync) | Synchronously creates a new directory with the specified path. |
| f  [**Deno.open**](https://deno.land/api@v1.25.3?s=Deno.open) | Open a file and resolve to an instance of Deno.FsFile. The file does not need to previously exist if using the create or createNew open options. It is the callers responsibility to close the file when finished with it. |
| f  [**Deno.openSync**](https://deno.land/api@v1.25.3?s=Deno.openSync) | Synchronously open a file and return an instance of Deno.FsFile. The file does not need to previously exist if using the create or createNew open options. It is the callers responsibility to close the file when finished with it. |
| f  [**Deno.readDir**](https://deno.land/api@v1.25.3?s=Deno.readDir) | Reads the directory given by path and returns an async iterable of Deno.DirEntry. |
| f  [**Deno.readDirSync**](https://deno.land/api@v1.25.3?s=Deno.readDirSync) | Synchronously reads the directory given by path and returns an iterable of Deno.DirEntry. |
| f  [**Deno.readFile**](https://deno.land/api@v1.25.3?s=Deno.readFile) | Reads and resolves to the entire contents of a file as an array of bytes. TextDecoder can be used to transform the bytes to string if required. Reading a directory returns an empty data array. |
| f  [**Deno.readFileSync**](https://deno.land/api@v1.25.3?s=Deno.readFileSync) | Synchronously reads and returns the entire contents of a file as an array of bytes. TextDecoder can be used to transform the bytes to string if required. Reading a directory returns an empty data array. |
| f  [**Deno.readLink**](https://deno.land/api@v1.25.3?s=Deno.readLink) | Resolves to the full path destination of the named symbolic link. |
| f  [**Deno.readLinkSync**](https://deno.land/api@v1.25.3?s=Deno.readLinkSync) | Returns the full path destination of the named symbolic link. |
| f  [**Deno.readTextFile**](https://deno.land/api@v1.25.3?s=Deno.readTextFile) | Asynchronously reads and returns the entire contents of a file as utf8 encoded string. Reading a directory throws an error. |
| f  [**Deno.readTextFileSync**](https://deno.land/api@v1.25.3?s=Deno.readTextFileSync) | Synchronously reads and returns the entire contents of a file as utf8 encoded string. Reading a directory throws an error. |
| f  [**Deno.realPath**](https://deno.land/api@v1.25.3?s=Deno.realPath) | Resolves to the absolute normalized path, with symbolic links resolved. |
| f  [**Deno.realPathSync**](https://deno.land/api@v1.25.3?s=Deno.realPathSync) | Returns absolute normalized path, with symbolic links resolved. |
| f  [**Deno.remove**](https://deno.land/api@v1.25.3?s=Deno.remove) | Removes the named file or directory. |
| f  [**Deno.removeSync**](https://deno.land/api@v1.25.3?s=Deno.removeSync) | Synchronously removes the named file or directory. |
| f  [**Deno.rename**](https://deno.land/api@v1.25.3?s=Deno.rename) | Renames (moves) oldpath to newpath. Paths may be files or directories. If newpath already exists and is not a directory, rename() replaces it. OS-specific restrictions may apply when oldpath and newpath are in different directories. |
| f  [**Deno.renameSync**](https://deno.land/api@v1.25.3?s=Deno.renameSync) | Synchronously renames (moves) oldpath to newpath. Paths may be files or directories. If newpath already exists and is not a directory, renameSync() replaces it. OS-specific restrictions may apply when oldpath and newpath are in different directories. |
| f  [**Deno.stat**](https://deno.land/api@v1.25.3?s=Deno.stat) | Resolves to a Deno.FileInfo for the specified path. Will always follow symlinks. |
| f  [**Deno.statSync**](https://deno.land/api@v1.25.3?s=Deno.statSync) | Synchronously returns a Deno.FileInfo for the specified path. Will always follow symlinks. |
| f  [**Deno.symlink**](https://deno.land/api@v1.25.3?s=Deno.symlink) | Creates newpath as a symbolic link to oldpath. |
| f  [**Deno.symlinkSync**](https://deno.land/api@v1.25.3?s=Deno.symlinkSync) | Creates newpath as a symbolic link to oldpath. |
| f  [**Deno.truncate**](https://deno.land/api@v1.25.3?s=Deno.truncate) | Truncates or extends the specified file, to reach the specified len. If len is not specified then the entire file contents are truncated. |
| f  [**Deno.truncateSync**](https://deno.land/api@v1.25.3?s=Deno.truncateSync) | Synchronously truncates or extends the specified file, to reach the specified len. If len is not specified then the entire file contents are truncated. |
| f  [**Deno.watchFs**](https://deno.land/api@v1.25.3?s=Deno.watchFs) | Watch for file system events against one or more paths, which can be files or directories. These paths must exist already. One user action (e.g. touch test.file) can generate multiple file system events. Likewise, one user action can result in multiple file paths in one event (e.g. mv old\_name.txt new\_name.txt). Recursive option is true by default and, for directories, will watch the specified directory and all sub directories. Note that the exact ordering of the events can vary between operating systems. |
| f  [**Deno.writeFile**](https://deno.land/api@v1.25.3?s=Deno.writeFile) | Write data to the given path, by default creating a new file if needed, else overwriting. |
| f  [**Deno.writeFileSync**](https://deno.land/api@v1.25.3?s=Deno.writeFileSync) | Synchronously write data to the given path, by default creating a new file if needed, else overwriting. |
| f  [**Deno.writeTextFile**](https://deno.land/api@v1.25.3?s=Deno.writeTextFile) | Asynchronously write string data to the given path, by default creating a new file if needed, else overwriting. |
| f  [**Deno.writeTextFileSync**](https://deno.land/api@v1.25.3?s=Deno.writeTextFileSync) | Synchronously write string data to the given path, by default creating a new file if needed, else overwriting. |

## [HTTP Server](https://deno.land/api@v1.25.3#HTTP_Server)

|  |  |
| --- | --- |
| I  [**Deno.HttpConn**](https://deno.land/api@v1.25.3?s=Deno.HttpConn) |  |
| I  [**Deno.RequestEvent**](https://deno.land/api@v1.25.3?s=Deno.RequestEvent) |  |
| f  [**Deno.serveHttp**](https://deno.land/api@v1.25.3?s=Deno.serveHttp) | Services HTTP requests given a TCP or TLS socket. |

## [I/O](https://deno.land/api@v1.25.3#I/O)

|  |  |
| --- | --- |
| c  [**Deno.Buffer**](https://deno.land/api@v1.25.3?s=Deno.Buffer)  deprecated | A variable-sized buffer of bytes with read() and write() methods. |
| I  [**Deno.Closer**](https://deno.land/api@v1.25.3?s=Deno.Closer) |  |
| I  [**Deno.Reader**](https://deno.land/api@v1.25.3?s=Deno.Reader) |  |
| I  [**Deno.ReaderSync**](https://deno.land/api@v1.25.3?s=Deno.ReaderSync) |  |
| I  [**Deno.Seeker**](https://deno.land/api@v1.25.3?s=Deno.Seeker) |  |
| I  [**Deno.SeekerSync**](https://deno.land/api@v1.25.3?s=Deno.SeekerSync) |  |
| I  [**Deno.Writer**](https://deno.land/api@v1.25.3?s=Deno.Writer) |  |
| I  [**Deno.WriterSync**](https://deno.land/api@v1.25.3?s=Deno.WriterSync) |  |
| v  [**Deno.stderr**](https://deno.land/api@v1.25.3?s=Deno.stderr) | A handle for stderr. |
| v  [**Deno.stdin**](https://deno.land/api@v1.25.3?s=Deno.stdin) | A handle for stdin. |
| v  [**Deno.stdout**](https://deno.land/api@v1.25.3?s=Deno.stdout) | A handle for stdout. |
| f  [**Deno.close**](https://deno.land/api@v1.25.3?s=Deno.close) | Close the given resource ID (rid) which has been previously opened, such as via opening or creating a file. Closing a file when you are finished with it is important to avoid leaking resources. |
| f  [**Deno.copy**](https://deno.land/api@v1.25.3?s=Deno.copy)  deprecated | Copies from src to dst until either EOF (null) is read from src or an error occurs. It resolves to the number of bytes copied or rejects with the first error encountered while copying. |
| f  [**Deno.fdatasync**](https://deno.land/api@v1.25.3?s=Deno.fdatasync) | Flushes any pending data operations of the given file stream to disk.  const file = await Deno.open("my\_file.txt", { read: true, write: true, create: true });  await Deno.write(file.rid, new TextEncoder().encode("Hello World"));  await Deno.fdatasync(file.rid);  console.log(new TextDecoder().decode(await Deno.readFile("my\_file.txt"))); // Hello World |
| f  [**Deno.fdatasyncSync**](https://deno.land/api@v1.25.3?s=Deno.fdatasyncSync) | Synchronously flushes any pending data operations of the given file stream to disk. |
| f  [**Deno.fsync**](https://deno.land/api@v1.25.3?s=Deno.fsync) | Flushes any pending data and metadata operations of the given file stream to disk. |
| f  [**Deno.fsyncSync**](https://deno.land/api@v1.25.3?s=Deno.fsyncSync) | Synchronously flushes any pending data and metadata operations of the given file stream to disk. |
| f  [**Deno.isatty**](https://deno.land/api@v1.25.3?s=Deno.isatty) | Check if a given resource id (rid) is a TTY. |
| f  [**Deno.iter**](https://deno.land/api@v1.25.3?s=Deno.iter)  deprecated | Turns a Reader, r, into an async iterator. |
| f  [**Deno.iterSync**](https://deno.land/api@v1.25.3?s=Deno.iterSync)  deprecated | Turns a ReaderSync, r, into an iterator. |
| f  [**Deno.read**](https://deno.land/api@v1.25.3?s=Deno.read) | Read from a resource ID (rid) into an array buffer (buffer). |
| f  [**Deno.readAll**](https://deno.land/api@v1.25.3?s=Deno.readAll)  deprecated | Read Reader r until EOF (null) and resolve to the content as Uint8Array`. |
| f  [**Deno.readAllSync**](https://deno.land/api@v1.25.3?s=Deno.readAllSync)  deprecated | Synchronously reads Reader r until EOF (null) and returns the content as Uint8Array. |
| f  [**Deno.readSync**](https://deno.land/api@v1.25.3?s=Deno.readSync) | Synchronously read from a resource ID (rid) into an array buffer (buffer). |
| f  [**Deno.seek**](https://deno.land/api@v1.25.3?s=Deno.seek) | Seek a resource ID (rid) to the given offset under mode given by whence. The call resolves to the new position within the resource (bytes from the start). |
| f  [**Deno.seekSync**](https://deno.land/api@v1.25.3?s=Deno.seekSync) | Synchronously seek a resource ID (rid) to the given offset under mode given by whence. The new position within the resource (bytes from the start) is returned. |
| f  [**Deno.write**](https://deno.land/api@v1.25.3?s=Deno.write) | Write to the resource ID (rid) the contents of the array buffer (data). |
| f  [**Deno.writeAll**](https://deno.land/api@v1.25.3?s=Deno.writeAll)  deprecated | Write all the content of the array buffer (arr) to the writer (w). |
| f  [**Deno.writeAllSync**](https://deno.land/api@v1.25.3?s=Deno.writeAllSync)  deprecated | Synchronously write all the content of the array buffer (arr) to the writer (w). |
| f  [**Deno.writeSync**](https://deno.land/api@v1.25.3?s=Deno.writeSync) | Synchronously write to the resource ID (rid) the contents of the array buffer (data). |
| E  [**Deno.SeekMode**](https://deno.land/api@v1.25.3?s=Deno.SeekMode) |  |

## [Network](https://deno.land/api@v1.25.3#Network)

|  |  |
| --- | --- |
| I  [**Deno.CAARecord**](https://deno.land/api@v1.25.3?s=Deno.CAARecord) | If resolveDns is called with "CAA" record type specified, it will return an array of this interface. |
| I  [**Deno.Conn**](https://deno.land/api@v1.25.3?s=Deno.Conn) |  |
| I  [**Deno.ConnectOptions**](https://deno.land/api@v1.25.3?s=Deno.ConnectOptions) |  |
| I  [**Deno.ConnectTlsOptions**](https://deno.land/api@v1.25.3?s=Deno.ConnectTlsOptions) |  |
| I  [**Deno.Listener**](https://deno.land/api@v1.25.3?s=Deno.Listener) | A generic network listener for stream-oriented protocols. |
| I  [**Deno.ListenOptions**](https://deno.land/api@v1.25.3?s=Deno.ListenOptions) |  |
| I  [**Deno.ListenTlsOptions**](https://deno.land/api@v1.25.3?s=Deno.ListenTlsOptions) |  |
| I  [**Deno.MXRecord**](https://deno.land/api@v1.25.3?s=Deno.MXRecord) | If resolveDns is called with "MX" record type specified, it will return an array of this interface. |
| I  [**Deno.NAPTRRecord**](https://deno.land/api@v1.25.3?s=Deno.NAPTRRecord) | If resolveDns is called with "NAPTR" record type specified, it will return an array of this interface. |
| I  [**Deno.NetAddr**](https://deno.land/api@v1.25.3?s=Deno.NetAddr) |  |
| I  [**Deno.ResolveDnsOptions**](https://deno.land/api@v1.25.3?s=Deno.ResolveDnsOptions) |  |
| I  [**Deno.SOARecord**](https://deno.land/api@v1.25.3?s=Deno.SOARecord) | If resolveDns is called with "SOA" record type specified, it will return an array of this interface. |
| I  [**Deno.SRVRecord**](https://deno.land/api@v1.25.3?s=Deno.SRVRecord) | If resolveDns is called with "SRV" record type specified, it will return an array of this interface. |
| I  [**Deno.StartTlsOptions**](https://deno.land/api@v1.25.3?s=Deno.StartTlsOptions) |  |
| I  [**Deno.TcpConn**](https://deno.land/api@v1.25.3?s=Deno.TcpConn) |  |
| I  [**Deno.TlsConn**](https://deno.land/api@v1.25.3?s=Deno.TlsConn) |  |
| I  [**Deno.TlsHandshakeInfo**](https://deno.land/api@v1.25.3?s=Deno.TlsHandshakeInfo) |  |
| I  [**Deno.TlsListener**](https://deno.land/api@v1.25.3?s=Deno.TlsListener) | Specialized listener that accepts TLS connections. |
| I  [**Deno.UnixAddr**](https://deno.land/api@v1.25.3?s=Deno.UnixAddr) |  |
| I  [**Deno.UnixConn**](https://deno.land/api@v1.25.3?s=Deno.UnixConn) |  |
| T  [**Deno.Addr**](https://deno.land/api@v1.25.3?s=Deno.Addr) |  |
| T  [**Deno.RecordType**](https://deno.land/api@v1.25.3?s=Deno.RecordType) | The type of the resource record. Only the listed types are supported currently. |
| f  [**Deno.connect**](https://deno.land/api@v1.25.3?s=Deno.connect) | Connects to the hostname (default is "127.0.0.1") and port on the named transport (default is "tcp"), and resolves to the connection (Conn). |
| f  [**Deno.connectTls**](https://deno.land/api@v1.25.3?s=Deno.connectTls) | Establishes a secure connection over TLS (transport layer security) using an optional cert file, hostname (default is "127.0.0.1") and port. The cert file is optional and if not included Mozilla's root certificates will be used (see also <https://github.com/ctz/webpki-roots> for specifics) |
| f  [**Deno.listen**](https://deno.land/api@v1.25.3?s=Deno.listen) | Listen announces on the local transport address. |
| f  [**Deno.listenTls**](https://deno.land/api@v1.25.3?s=Deno.listenTls) | Listen announces on the local transport address over TLS (transport layer security). |
| f  [**Deno.resolveDns**](https://deno.land/api@v1.25.3?s=Deno.resolveDns) |  |
| f  [**Deno.shutdown**](https://deno.land/api@v1.25.3?s=Deno.shutdown) | Shutdown socket send operations. |
| f  [**Deno.startTls**](https://deno.land/api@v1.25.3?s=Deno.startTls) | Start TLS handshake from an existing connection using an optional list of CA certificates, and hostname (default is "127.0.0.1"). Specifying CA certs is optional. By default the configured root certificates are used. Using this function requires that the other end of the connection is prepared for a TLS handshake. |

## [Observability](https://deno.land/api@v1.25.3#Observability)

|  |  |
| --- | --- |
| c  [**PromiseRejectionEvent**](https://deno.land/api@v1.25.3?s=PromiseRejectionEvent) |  |
| I  [**Deno.Metrics**](https://deno.land/api@v1.25.3?s=Deno.Metrics) |  |
| I  [**Deno.OpMetrics**](https://deno.land/api@v1.25.3?s=Deno.OpMetrics) |  |
| I  [**Deno.ResourceMap**](https://deno.land/api@v1.25.3?s=Deno.ResourceMap) |  |
| I  [**PromiseRejectionEventInit**](https://deno.land/api@v1.25.3?s=PromiseRejectionEventInit) |  |
| v  [**onunhandledrejection**](https://deno.land/api@v1.25.3?s=onunhandledrejection) |  |
| f  [**Deno.metrics**](https://deno.land/api@v1.25.3?s=Deno.metrics) | Receive metrics from the privileged side of Deno. This is primarily used in the development of Deno. 'Ops', also called 'bindings', are the go-between between Deno JavaScript and Deno Rust. |
| f  [**Deno.resources**](https://deno.land/api@v1.25.3?s=Deno.resources) | Returns a map of open resource ids (rid) along with their string representations. This is an internal API and as such resource representation has any type; that means it can change any time. |

## [Performance API](https://deno.land/api@v1.25.3#Performance_API)

|  |  |
| --- | --- |
| c  [**Performance**](https://deno.land/api@v1.25.3?s=Performance) |  |
| c  [**PerformanceEntry**](https://deno.land/api@v1.25.3?s=PerformanceEntry) | Encapsulates a single performance metric that is part of the performance timeline. A performance entry can be directly created by making a performance mark or measure (for example by calling the .mark() method) at an explicit point in an application. |
| c  [**PerformanceMark**](https://deno.land/api@v1.25.3?s=PerformanceMark) | PerformanceMark is an abstract interface for PerformanceEntry objects with an entryType of "mark". Entries of this type are created by calling performance.mark() to add a named DOMHighResTimeStamp (the mark) to the performance timeline. |
| c  [**PerformanceMeasure**](https://deno.land/api@v1.25.3?s=PerformanceMeasure) | PerformanceMeasure is an abstract interface for PerformanceEntry objects with an entryType of "measure". Entries of this type are created by calling performance.measure() to add a named DOMHighResTimeStamp (the measure) between two marks to the performance timeline. |
| I  [**Performance**](https://deno.land/api@v1.25.3?s=Performance) | Deno supports user timing Level 3 (see: <https://w3c.github.io/user-timing>) which is not widely supported yet in other runtimes. These types are here so that these features are still available when using the Deno namespace in conjunction with other type libs, like dom. |
| I  [**PerformanceMarkOptions**](https://deno.land/api@v1.25.3?s=PerformanceMarkOptions) |  |
| I  [**PerformanceMeasureOptions**](https://deno.land/api@v1.25.3?s=PerformanceMeasureOptions) |  |
| T  [**PerformanceEntryList**](https://deno.land/api@v1.25.3?s=PerformanceEntryList) |  |
| v  [**performance**](https://deno.land/api@v1.25.3?s=performance) |  |

## [Permissions](https://deno.land/api@v1.25.3#Permissions)

|  |  |
| --- | --- |
| c  [**Deno.Permissions**](https://deno.land/api@v1.25.3?s=Deno.Permissions) |  |
| c  [**Deno.PermissionStatus**](https://deno.land/api@v1.25.3?s=Deno.PermissionStatus) |  |
| I  [**Deno.EnvPermissionDescriptor**](https://deno.land/api@v1.25.3?s=Deno.EnvPermissionDescriptor) |  |
| I  [**Deno.FfiPermissionDescriptor**](https://deno.land/api@v1.25.3?s=Deno.FfiPermissionDescriptor) |  |
| I  [**Deno.HrtimePermissionDescriptor**](https://deno.land/api@v1.25.3?s=Deno.HrtimePermissionDescriptor) |  |
| I  [**Deno.NetPermissionDescriptor**](https://deno.land/api@v1.25.3?s=Deno.NetPermissionDescriptor) |  |
| I  [**Deno.PermissionOptionsObject**](https://deno.land/api@v1.25.3?s=Deno.PermissionOptionsObject) |  |
| I  [**Deno.PermissionStatusEventMap**](https://deno.land/api@v1.25.3?s=Deno.PermissionStatusEventMap) |  |
| I  [**Deno.ReadPermissionDescriptor**](https://deno.land/api@v1.25.3?s=Deno.ReadPermissionDescriptor) |  |
| I  [**Deno.RunPermissionDescriptor**](https://deno.land/api@v1.25.3?s=Deno.RunPermissionDescriptor) |  |
| I  [**Deno.WritePermissionDescriptor**](https://deno.land/api@v1.25.3?s=Deno.WritePermissionDescriptor) |  |
| T  [**Deno.PermissionDescriptor**](https://deno.land/api@v1.25.3?s=Deno.PermissionDescriptor) | Permission descriptors which define a permission and can be queried, requested, or revoked. |
| T  [**Deno.PermissionName**](https://deno.land/api@v1.25.3?s=Deno.PermissionName) | The name of a "powerful feature" which needs permission. |
| T  [**Deno.PermissionOptions**](https://deno.land/api@v1.25.3?s=Deno.PermissionOptions) |  |
| T  [**Deno.PermissionState**](https://deno.land/api@v1.25.3?s=Deno.PermissionState) | The current status of the permission. |
| v  [**Deno.permissions**](https://deno.land/api@v1.25.3?s=Deno.permissions) | Deno's permission management API. |

## [Runtime Environment](https://deno.land/api@v1.25.3#Runtime_Environment)

|  |  |
| --- | --- |
| I  [**Deno.MemoryUsage**](https://deno.land/api@v1.25.3?s=Deno.MemoryUsage) |  |
| T  [**Deno.Signal**](https://deno.land/api@v1.25.3?s=Deno.Signal) |  |
| v  [**Deno.args**](https://deno.land/api@v1.25.3?s=Deno.args) | Returns the script arguments to the program. If for example we run a program: |
| v  [**Deno.build**](https://deno.land/api@v1.25.3?s=Deno.build) | Build related information. |
| v  [**Deno.env**](https://deno.land/api@v1.25.3?s=Deno.env) |  |
| v  [**Deno.mainModule**](https://deno.land/api@v1.25.3?s=Deno.mainModule) | The URL of the entrypoint module entered from the command-line. |
| v  [**Deno.noColor**](https://deno.land/api@v1.25.3?s=Deno.noColor) | Reflects the NO\_COLOR environment variable at program start. |
| v  [**Deno.pid**](https://deno.land/api@v1.25.3?s=Deno.pid) | The current process id of the runtime. |
| v  [**Deno.ppid**](https://deno.land/api@v1.25.3?s=Deno.ppid) | The pid of the current process's parent. |
| v  [**Deno.version**](https://deno.land/api@v1.25.3?s=Deno.version) | Version related information. |
| f  [**Deno.addSignalListener**](https://deno.land/api@v1.25.3?s=Deno.addSignalListener) | Registers the given function as a listener of the given signal event. |
| f  [**Deno.chdir**](https://deno.land/api@v1.25.3?s=Deno.chdir) | Change the current working directory to the specified path. |
| f  [**Deno.cwd**](https://deno.land/api@v1.25.3?s=Deno.cwd) | Return a string representing the current working directory. |
| f  [**Deno.execPath**](https://deno.land/api@v1.25.3?s=Deno.execPath) | Returns the path to the current deno executable. |
| f  [**Deno.exit**](https://deno.land/api@v1.25.3?s=Deno.exit) | Exit the Deno process with optional exit code. If no exit code is supplied then Deno will exit with return code of 0. |
| f  [**Deno.memoryUsage**](https://deno.land/api@v1.25.3?s=Deno.memoryUsage) | Returns an object describing the memory usage of the Deno process measured in bytes. |
| f  [**Deno.removeSignalListener**](https://deno.land/api@v1.25.3?s=Deno.removeSignalListener) | Removes the given signal listener that has been registered with Deno.addSignalListener. |

## [Scheduling](https://deno.land/api@v1.25.3#Scheduling)

|  |  |
| --- | --- |
| I  [**VoidFunction**](https://deno.land/api@v1.25.3?s=VoidFunction) |  |
| f  [**queueMicrotask**](https://deno.land/api@v1.25.3?s=queueMicrotask) | A microtask is a short function which is executed after the function or module which created it exits and only if the JavaScript execution stack is empty, but before returning control to the event loop being used to drive the script's execution environment. This event loop may be either the main event loop or the event loop driving a web worker. |

## [Streams API](https://deno.land/api@v1.25.3#Streams_API)

|  |  |
| --- | --- |
| c  [**MessageEvent**](https://deno.land/api@v1.25.3?s=MessageEvent) |  |
| I  [**ByteLengthQueuingStrategy**](https://deno.land/api@v1.25.3?s=ByteLengthQueuingStrategy) |  |
| I  [**CountQueuingStrategy**](https://deno.land/api@v1.25.3?s=CountQueuingStrategy) | This Streams API interface provides a built-in byte length queuing strategy that can be used when constructing streams. |
| I  [**MessageEventInit**](https://deno.land/api@v1.25.3?s=MessageEventInit) |  |
| I  [**PipeOptions**](https://deno.land/api@v1.25.3?s=PipeOptions) |  |
| I  [**QueuingStrategy**](https://deno.land/api@v1.25.3?s=QueuingStrategy) |  |
| I  [**QueuingStrategySizeCallback**](https://deno.land/api@v1.25.3?s=QueuingStrategySizeCallback) |  |
| I  [**ReadableByteStreamController**](https://deno.land/api@v1.25.3?s=ReadableByteStreamController) |  |
| I  [**ReadableByteStreamControllerCallback**](https://deno.land/api@v1.25.3?s=ReadableByteStreamControllerCallback) |  |
| I  [**ReadableStream**](https://deno.land/api@v1.25.3?s=ReadableStream) | This Streams API interface represents a readable stream of byte data. The Fetch API offers a concrete instance of a ReadableStream through the body property of a Response object. |
| I  [**ReadableStreamBYOBReadDoneResult**](https://deno.land/api@v1.25.3?s=ReadableStreamBYOBReadDoneResult) |  |
| I  [**ReadableStreamBYOBReader**](https://deno.land/api@v1.25.3?s=ReadableStreamBYOBReader) |  |
| I  [**ReadableStreamBYOBReadValueResult**](https://deno.land/api@v1.25.3?s=ReadableStreamBYOBReadValueResult) |  |
| I  [**ReadableStreamBYOBRequest**](https://deno.land/api@v1.25.3?s=ReadableStreamBYOBRequest) |  |
| I  [**ReadableStreamDefaultController**](https://deno.land/api@v1.25.3?s=ReadableStreamDefaultController) |  |
| I  [**ReadableStreamDefaultControllerCallback**](https://deno.land/api@v1.25.3?s=ReadableStreamDefaultControllerCallback) |  |
| I  [**ReadableStreamDefaultReader**](https://deno.land/api@v1.25.3?s=ReadableStreamDefaultReader) |  |
| I  [**ReadableStreamErrorCallback**](https://deno.land/api@v1.25.3?s=ReadableStreamErrorCallback) |  |
| I  [**ReadableStreamReadDoneResult**](https://deno.land/api@v1.25.3?s=ReadableStreamReadDoneResult) |  |
| I  [**ReadableStreamReader**](https://deno.land/api@v1.25.3?s=ReadableStreamReader) |  |
| I  [**ReadableStreamReadValueResult**](https://deno.land/api@v1.25.3?s=ReadableStreamReadValueResult) |  |
| I  [**Transformer**](https://deno.land/api@v1.25.3?s=Transformer) |  |
| I  [**TransformStream**](https://deno.land/api@v1.25.3?s=TransformStream) |  |
| I  [**TransformStreamDefaultController**](https://deno.land/api@v1.25.3?s=TransformStreamDefaultController) |  |
| I  [**TransformStreamDefaultControllerCallback**](https://deno.land/api@v1.25.3?s=TransformStreamDefaultControllerCallback) |  |
| I  [**TransformStreamDefaultControllerTransformCallback**](https://deno.land/api@v1.25.3?s=TransformStreamDefaultControllerTransformCallback) |  |
| I  [**UnderlyingByteSource**](https://deno.land/api@v1.25.3?s=UnderlyingByteSource) |  |
| I  [**UnderlyingSink**](https://deno.land/api@v1.25.3?s=UnderlyingSink) |  |
| I  [**UnderlyingSource**](https://deno.land/api@v1.25.3?s=UnderlyingSource) |  |
| I  [**WritableStream**](https://deno.land/api@v1.25.3?s=WritableStream) | This Streams API interface provides a standard abstraction for writing streaming data to a destination, known as a sink. This object comes with built-in backpressure and queuing. |
| I  [**WritableStreamDefaultController**](https://deno.land/api@v1.25.3?s=WritableStreamDefaultController) | This Streams API interface represents a controller allowing control of a WritableStream's state. When constructing a WritableStream, the underlying sink is given a corresponding WritableStreamDefaultController instance to manipulate. |
| I  [**WritableStreamDefaultControllerCloseCallback**](https://deno.land/api@v1.25.3?s=WritableStreamDefaultControllerCloseCallback) |  |
| I  [**WritableStreamDefaultControllerStartCallback**](https://deno.land/api@v1.25.3?s=WritableStreamDefaultControllerStartCallback) |  |
| I  [**WritableStreamDefaultControllerWriteCallback**](https://deno.land/api@v1.25.3?s=WritableStreamDefaultControllerWriteCallback) |  |
| I  [**WritableStreamDefaultWriter**](https://deno.land/api@v1.25.3?s=WritableStreamDefaultWriter) | This Streams API interface is the object returned by WritableStream.getWriter() and once created locks the < writer to the WritableStream ensuring that no other streams can write to the underlying sink. |
| I  [**WritableStreamErrorCallback**](https://deno.land/api@v1.25.3?s=WritableStreamErrorCallback) |  |
| T  [**ReadableStreamBYOBReadResult**](https://deno.land/api@v1.25.3?s=ReadableStreamBYOBReadResult) |  |
| T  [**ReadableStreamReadResult**](https://deno.land/api@v1.25.3?s=ReadableStreamReadResult) |  |
| v  [**ByteLengthQueuingStrategy**](https://deno.land/api@v1.25.3?s=ByteLengthQueuingStrategy) |  |
| v  [**CountQueuingStrategy**](https://deno.land/api@v1.25.3?s=CountQueuingStrategy) |  |
| v  [**ReadableByteStreamController**](https://deno.land/api@v1.25.3?s=ReadableByteStreamController) |  |
| v  [**ReadableStream**](https://deno.land/api@v1.25.3?s=ReadableStream) |  |
| v  [**ReadableStreamDefaultController**](https://deno.land/api@v1.25.3?s=ReadableStreamDefaultController) |  |
| v  [**ReadableStreamDefaultReader**](https://deno.land/api@v1.25.3?s=ReadableStreamDefaultReader) |  |
| v  [**ReadableStreamReader**](https://deno.land/api@v1.25.3?s=ReadableStreamReader) |  |
| v  [**TransformStream**](https://deno.land/api@v1.25.3?s=TransformStream) |  |
| v  [**TransformStreamDefaultController**](https://deno.land/api@v1.25.3?s=TransformStreamDefaultController) |  |
| v  [**WritableStream**](https://deno.land/api@v1.25.3?s=WritableStream) |  |
| v  [**WritableStreamDefaultController**](https://deno.land/api@v1.25.3?s=WritableStreamDefaultController) |  |
| v  [**WritableStreamDefaultWriter**](https://deno.land/api@v1.25.3?s=WritableStreamDefaultWriter) |  |

## [Sub Process](https://deno.land/api@v1.25.3#Sub_Process)

|  |  |
| --- | --- |
| c  [**Deno.Process**](https://deno.land/api@v1.25.3?s=Deno.Process) |  |
| I  [**Deno.RunOptions**](https://deno.land/api@v1.25.3?s=Deno.RunOptions) |  |
| T  [**Deno.ProcessStatus**](https://deno.land/api@v1.25.3?s=Deno.ProcessStatus) |  |
| f  [**Deno.kill**](https://deno.land/api@v1.25.3?s=Deno.kill) | Send a signal to process under given pid. |
| f  [**Deno.run**](https://deno.land/api@v1.25.3?s=Deno.run) | Spawns new subprocess. RunOptions must contain at a minimum the opt.cmd, an array of program arguments, the first of which is the binary. |

## [Testing](https://deno.land/api@v1.25.3#Testing)

|  |  |
| --- | --- |
| I  [**Deno.TestContext**](https://deno.land/api@v1.25.3?s=Deno.TestContext) |  |
| I  [**Deno.TestDefinition**](https://deno.land/api@v1.25.3?s=Deno.TestDefinition) |  |
| I  [**Deno.TestStepDefinition**](https://deno.land/api@v1.25.3?s=Deno.TestStepDefinition) |  |
| f  [**Deno.test**](https://deno.land/api@v1.25.3?s=Deno.test) | Register a test which will be run when deno test is used on the command line and the containing module looks like a test module. fn can be async if required.  import {assert, fail, assertEquals} from "https://deno.land/std/testing/asserts.ts"; |

## [Timers](https://deno.land/api@v1.25.3#Timers)

|  |  |
| --- | --- |
| f  [**clearInterval**](https://deno.land/api@v1.25.3?s=clearInterval) | Cancels a timed, repeating action which was previously started by a call to setInterval() |
| f  [**clearTimeout**](https://deno.land/api@v1.25.3?s=clearTimeout) | Cancels a scheduled action initiated by setTimeout() |
| f  [**setInterval**](https://deno.land/api@v1.25.3?s=setInterval) | Repeatedly calls a function , with a fixed time delay between each call. |
| f  [**setTimeout**](https://deno.land/api@v1.25.3?s=setTimeout) | Sets a timer which executes a function once after the timer expires. Returns an id which may be used to cancel the timeout. |

## [Typed Arrays](https://deno.land/api@v1.25.3#Typed_Arrays)

|  |  |
| --- | --- |
| T  [**BufferSource**](https://deno.land/api@v1.25.3?s=BufferSource) |  |

## [Web APIs](https://deno.land/api@v1.25.3#Web_APIs)

|  |  |
| --- | --- |
| c  [**AbortController**](https://deno.land/api@v1.25.3?s=AbortController) | A controller object that allows you to abort one or more DOM requests as and when desired. |
| c  [**Location**](https://deno.land/api@v1.25.3?s=Location) | The location (URL) of the object it is linked to. Changes done on it are reflected on the object it relates to. Accessible via globalThis.location. |
| c  [**Navigator**](https://deno.land/api@v1.25.3?s=Navigator) |  |
| c  [**URL**](https://deno.land/api@v1.25.3?s=URL) | The URL interface represents an object providing static methods used for creating object URLs. |
| c  [**URLPattern**](https://deno.land/api@v1.25.3?s=URLPattern) | The URLPattern API provides a web platform primitive for matching URLs based on a convenient pattern syntax. |
| c  [**URLSearchParams**](https://deno.land/api@v1.25.3?s=URLSearchParams) |  |
| c  [**Window**](https://deno.land/api@v1.25.3?s=Window) |  |
| I  [**AbortSignal**](https://deno.land/api@v1.25.3?s=AbortSignal) | A signal object that allows you to communicate with a DOM request (such as a Fetch) and abort it if required via an AbortController object. |
| I  [**AbortSignalEventMap**](https://deno.land/api@v1.25.3?s=AbortSignalEventMap) |  |
| I  [**URLPatternComponentResult**](https://deno.land/api@v1.25.3?s=URLPatternComponentResult) |  |
| I  [**URLPatternInit**](https://deno.land/api@v1.25.3?s=URLPatternInit) |  |
| I  [**URLPatternResult**](https://deno.land/api@v1.25.3?s=URLPatternResult) | URLPatternResult is the object returned from URLPattern.exec. |
| I  [**WindowEventMap**](https://deno.land/api@v1.25.3?s=WindowEventMap) |  |
| T  [**URLPatternInput**](https://deno.land/api@v1.25.3?s=URLPatternInput) |  |
| v  [**AbortSignal**](https://deno.land/api@v1.25.3?s=AbortSignal) |  |
| v  [**location**](https://deno.land/api@v1.25.3?s=location) |  |
| v  [**navigator**](https://deno.land/api@v1.25.3?s=navigator) |  |
| v  [**self**](https://deno.land/api@v1.25.3?s=self) |  |
| v  [**window**](https://deno.land/api@v1.25.3?s=window) |  |
| f  [**alert**](https://deno.land/api@v1.25.3?s=alert) | Shows the given message and waits for the enter key pressed. |
| f  [**confirm**](https://deno.land/api@v1.25.3?s=confirm) | Shows the given message and waits for the answer. Returns the user's answer as boolean. |
| f  [**prompt**](https://deno.land/api@v1.25.3?s=prompt) | Shows the given message and waits for the user's input. Returns the user's input as string. |
| f  [**reportError**](https://deno.land/api@v1.25.3?s=reportError) | Dispatch an uncaught exception. Similar to a synchronous version of:  setTimeout(() => { throw error; }, 0);  The error can not be caught with a try/catch block. An error event will be dispatched to the global scope. You can prevent the error from being reported to the console with Event.prototype.preventDefault():  addEventListener("error", (event) => {  event.preventDefault();  });  reportError(new Error("foo")); // Will not be reported.  In Deno, this error will terminate the process if not intercepted like above. |

## [Web Crypto API](https://deno.land/api@v1.25.3#Web_Crypto_API)

|  |  |
| --- | --- |
| I  [**AesCbcParams**](https://deno.land/api@v1.25.3?s=AesCbcParams) |  |
| I  [**AesCtrParams**](https://deno.land/api@v1.25.3?s=AesCtrParams) |  |
| I  [**AesDerivedKeyParams**](https://deno.land/api@v1.25.3?s=AesDerivedKeyParams) |  |
| I  [**AesGcmParams**](https://deno.land/api@v1.25.3?s=AesGcmParams) |  |
| I  [**AesKeyAlgorithm**](https://deno.land/api@v1.25.3?s=AesKeyAlgorithm) |  |
| I  [**AesKeyGenParams**](https://deno.land/api@v1.25.3?s=AesKeyGenParams) |  |
| I  [**Algorithm**](https://deno.land/api@v1.25.3?s=Algorithm) |  |
| I  [**Crypto**](https://deno.land/api@v1.25.3?s=Crypto) |  |
| I  [**CryptoKey**](https://deno.land/api@v1.25.3?s=CryptoKey) | The CryptoKey dictionary of the Web Crypto API represents a cryptographic key. |
| I  [**CryptoKeyPair**](https://deno.land/api@v1.25.3?s=CryptoKeyPair) | The CryptoKeyPair dictionary of the Web Crypto API represents a key pair for an asymmetric cryptography algorithm, also known as a public-key algorithm. |
| I  [**EcdhKeyDeriveParams**](https://deno.land/api@v1.25.3?s=EcdhKeyDeriveParams) |  |
| I  [**EcdsaParams**](https://deno.land/api@v1.25.3?s=EcdsaParams) |  |
| I  [**EcKeyAlgorithm**](https://deno.land/api@v1.25.3?s=EcKeyAlgorithm) |  |
| I  [**EcKeyGenParams**](https://deno.land/api@v1.25.3?s=EcKeyGenParams) |  |
| I  [**EcKeyImportParams**](https://deno.land/api@v1.25.3?s=EcKeyImportParams) |  |
| I  [**HkdfParams**](https://deno.land/api@v1.25.3?s=HkdfParams) |  |
| I  [**HmacImportParams**](https://deno.land/api@v1.25.3?s=HmacImportParams) |  |
| I  [**HmacKeyAlgorithm**](https://deno.land/api@v1.25.3?s=HmacKeyAlgorithm) |  |
| I  [**HmacKeyGenParams**](https://deno.land/api@v1.25.3?s=HmacKeyGenParams) |  |
| I  [**JsonWebKey**](https://deno.land/api@v1.25.3?s=JsonWebKey) |  |
| I  [**KeyAlgorithm**](https://deno.land/api@v1.25.3?s=KeyAlgorithm) |  |
| I  [**Pbkdf2Params**](https://deno.land/api@v1.25.3?s=Pbkdf2Params) |  |
| I  [**RsaHashedImportParams**](https://deno.land/api@v1.25.3?s=RsaHashedImportParams) |  |
| I  [**RsaHashedKeyAlgorithm**](https://deno.land/api@v1.25.3?s=RsaHashedKeyAlgorithm) |  |
| I  [**RsaHashedKeyGenParams**](https://deno.land/api@v1.25.3?s=RsaHashedKeyGenParams) |  |
| I  [**RsaKeyAlgorithm**](https://deno.land/api@v1.25.3?s=RsaKeyAlgorithm) |  |
| I  [**RsaKeyGenParams**](https://deno.land/api@v1.25.3?s=RsaKeyGenParams) |  |
| I  [**RsaOaepParams**](https://deno.land/api@v1.25.3?s=RsaOaepParams) |  |
| I  [**RsaOtherPrimesInfo**](https://deno.land/api@v1.25.3?s=RsaOtherPrimesInfo) |  |
| I  [**RsaPssParams**](https://deno.land/api@v1.25.3?s=RsaPssParams) |  |
| I  [**SubtleCrypto**](https://deno.land/api@v1.25.3?s=SubtleCrypto) | This Web Crypto API interface provides a number of low-level cryptographic functions. It is accessed via the Crypto.subtle properties available in a window context (via Window.crypto). |
| T  [**AlgorithmIdentifier**](https://deno.land/api@v1.25.3?s=AlgorithmIdentifier) |  |
| T  [**HashAlgorithmIdentifier**](https://deno.land/api@v1.25.3?s=HashAlgorithmIdentifier) |  |
| T  [**KeyFormat**](https://deno.land/api@v1.25.3?s=KeyFormat) |  |
| T  [**KeyType**](https://deno.land/api@v1.25.3?s=KeyType) |  |
| T  [**KeyUsage**](https://deno.land/api@v1.25.3?s=KeyUsage) |  |
| T  [**NamedCurve**](https://deno.land/api@v1.25.3?s=NamedCurve) |  |
| v  [**crypto**](https://deno.land/api@v1.25.3?s=crypto) |  |
| v  [**CryptoKey**](https://deno.land/api@v1.25.3?s=CryptoKey) |  |
| v  [**CryptoKeyPair**](https://deno.land/api@v1.25.3?s=CryptoKeyPair) |  |
| v  [**SubtleCrypto**](https://deno.land/api@v1.25.3?s=SubtleCrypto) |  |

## [Web File API](https://deno.land/api@v1.25.3#Web_File_API)

|  |  |
| --- | --- |
| c  [**Blob**](https://deno.land/api@v1.25.3?s=Blob) | A file-like object of immutable, raw data. Blobs represent data that isn't necessarily in a JavaScript-native format. The File interface is based on Blob, inheriting blob functionality and expanding it to support files on the user's system. |
| c  [**File**](https://deno.land/api@v1.25.3?s=File) | Provides information about files and allows JavaScript in a web page to access their content. |
| I  [**BlobPropertyBag**](https://deno.land/api@v1.25.3?s=BlobPropertyBag) |  |
| I  [**FilePropertyBag**](https://deno.land/api@v1.25.3?s=FilePropertyBag) |  |
| I  [**FileReader**](https://deno.land/api@v1.25.3?s=FileReader) | Lets web applications asynchronously read the contents of files (or raw data buffers) stored on the user's computer, using File or Blob objects to specify the file or data to read. |
| I  [**FileReaderEventMap**](https://deno.land/api@v1.25.3?s=FileReaderEventMap) |  |
| T  [**BlobPart**](https://deno.land/api@v1.25.3?s=BlobPart) |  |
| v  [**FileReader**](https://deno.land/api@v1.25.3?s=FileReader) |  |

## [Web Sockets](https://deno.land/api@v1.25.3#Web_Sockets)

|  |  |
| --- | --- |
| c  [**CloseEvent**](https://deno.land/api@v1.25.3?s=CloseEvent) |  |
| c  [**WebSocket**](https://deno.land/api@v1.25.3?s=WebSocket) | Provides the API for creating and managing a WebSocket connection to a server, as well as for sending and receiving data on the connection. |
| I  [**CloseEventInit**](https://deno.land/api@v1.25.3?s=CloseEventInit) |  |
| I  [**Deno.UpgradeWebSocketOptions**](https://deno.land/api@v1.25.3?s=Deno.UpgradeWebSocketOptions) |  |
| I  [**Deno.WebSocketUpgrade**](https://deno.land/api@v1.25.3?s=Deno.WebSocketUpgrade) |  |
| I  [**WebSocketEventMap**](https://deno.land/api@v1.25.3?s=WebSocketEventMap) |  |
| T  [**BinaryType**](https://deno.land/api@v1.25.3?s=BinaryType) |  |
| f  [**Deno.upgradeWebSocket**](https://deno.land/api@v1.25.3?s=Deno.upgradeWebSocket) | Used to upgrade an incoming HTTP request to a WebSocket. |

## [Web Storage API](https://deno.land/api@v1.25.3#Web_Storage_API)

|  |  |
| --- | --- |
| I  [**Storage**](https://deno.land/api@v1.25.3?s=Storage) | This Web Storage API interface provides access to a particular domain's session or local storage. It allows, for example, the addition, modification, or deletion of stored data items. |
| v  [**localStorage**](https://deno.land/api@v1.25.3?s=localStorage) |  |
| v  [**sessionStorage**](https://deno.land/api@v1.25.3?s=sessionStorage) |  |
| v  [**Storage**](https://deno.land/api@v1.25.3?s=Storage) |  |

## [Web Workers](https://deno.land/api@v1.25.3#Web_Workers)

|  |  |
| --- | --- |
| c  [**Worker**](https://deno.land/api@v1.25.3?s=Worker) |  |
| I  [**AbstractWorkerEventMap**](https://deno.land/api@v1.25.3?s=AbstractWorkerEventMap) |  |
| I  [**WorkerEventMap**](https://deno.land/api@v1.25.3?s=WorkerEventMap) |  |
| I  [**WorkerOptions**](https://deno.land/api@v1.25.3?s=WorkerOptions) |  |

## [WebAssembly](https://deno.land/api@v1.25.3#WebAssembly)

|  |  |
| --- | --- |
| N  [**WebAssembly**](https://deno.land/api@v1.25.3?s=WebAssembly) |  |
| c  [**WebAssembly.CompileError**](https://deno.land/api@v1.25.3?s=WebAssembly.CompileError) | The WebAssembly.CompileError object indicates an error during WebAssembly decoding or validation. |
| c  [**WebAssembly.Global**](https://deno.land/api@v1.25.3?s=WebAssembly.Global) | A WebAssembly.Global object represents a global variable instance, accessible from both JavaScript and importable/exportable across one or more WebAssembly.Module instances. This allows dynamic linking of multiple modules. |
| c  [**WebAssembly.Instance**](https://deno.land/api@v1.25.3?s=WebAssembly.Instance) | A WebAssembly.Instance object is a stateful, executable instance of a WebAssembly.Module. Instance objects contain all the Exported WebAssembly functions that allow calling into WebAssembly code from JavaScript. |
| c  [**WebAssembly.LinkError**](https://deno.land/api@v1.25.3?s=WebAssembly.LinkError) | The WebAssembly.LinkError object indicates an error during module instantiation (besides traps from the start function). |
| c  [**WebAssembly.Memory**](https://deno.land/api@v1.25.3?s=WebAssembly.Memory) | The WebAssembly.Memory object is a resizable ArrayBuffer or SharedArrayBuffer that holds the raw bytes of memory accessed by a WebAssembly Instance. |
| c  [**WebAssembly.Module**](https://deno.land/api@v1.25.3?s=WebAssembly.Module) | A WebAssembly.Module object contains stateless WebAssembly code that has already been compiled by the browser — this can be efficiently shared with Workers, and instantiated multiple times. |
| c  [**WebAssembly.RuntimeError**](https://deno.land/api@v1.25.3?s=WebAssembly.RuntimeError) | The WebAssembly.RuntimeError object is the error type that is thrown whenever WebAssembly specifies a trap. |
| c  [**WebAssembly.Table**](https://deno.land/api@v1.25.3?s=WebAssembly.Table) | The WebAssembly.Table() object is a JavaScript wrapper object — an array-like structure representing a WebAssembly Table, which stores function references. A table created by JavaScript or in WebAssembly code will be accessible and mutable from both JavaScript and WebAssembly. |
| I  [**WebAssembly.GlobalDescriptor**](https://deno.land/api@v1.25.3?s=WebAssembly.GlobalDescriptor) | The GlobalDescriptor describes the options you can pass to new WebAssembly.Global(). |
| I  [**WebAssembly.MemoryDescriptor**](https://deno.land/api@v1.25.3?s=WebAssembly.MemoryDescriptor) | The MemoryDescriptor describes the options you can pass to new WebAssembly.Memory(). |
| I  [**WebAssembly.ModuleExportDescriptor**](https://deno.land/api@v1.25.3?s=WebAssembly.ModuleExportDescriptor) | A ModuleExportDescriptor is the description of a declared export in a WebAssembly.Module. |
| I  [**WebAssembly.ModuleImportDescriptor**](https://deno.land/api@v1.25.3?s=WebAssembly.ModuleImportDescriptor) | A ModuleImportDescriptor is the description of a declared import in a WebAssembly.Module. |
| I  [**WebAssembly.TableDescriptor**](https://deno.land/api@v1.25.3?s=WebAssembly.TableDescriptor) | The TableDescriptor describes the options you can pass to new WebAssembly.Table(). |
| I  [**WebAssembly.WebAssemblyInstantiatedSource**](https://deno.land/api@v1.25.3?s=WebAssembly.WebAssemblyInstantiatedSource) | The value returned from WebAssembly.instantiate. |
| T  [**WebAssembly.Exports**](https://deno.land/api@v1.25.3?s=WebAssembly.Exports) |  |
| T  [**WebAssembly.ExportValue**](https://deno.land/api@v1.25.3?s=WebAssembly.ExportValue) |  |
| T  [**WebAssembly.ImportExportKind**](https://deno.land/api@v1.25.3?s=WebAssembly.ImportExportKind) |  |
| T  [**WebAssembly.Imports**](https://deno.land/api@v1.25.3?s=WebAssembly.Imports) |  |
| T  [**WebAssembly.ImportValue**](https://deno.land/api@v1.25.3?s=WebAssembly.ImportValue) |  |
| T  [**WebAssembly.ModuleImports**](https://deno.land/api@v1.25.3?s=WebAssembly.ModuleImports) |  |
| T  [**WebAssembly.TableKind**](https://deno.land/api@v1.25.3?s=WebAssembly.TableKind) |  |
| T  [**WebAssembly.ValueType**](https://deno.land/api@v1.25.3?s=WebAssembly.ValueType) |  |
| f  [**WebAssembly.compile**](https://deno.land/api@v1.25.3?s=WebAssembly.compile) | The WebAssembly.compile() function compiles WebAssembly binary code into a WebAssembly.Module object. This function is useful if it is necessary to compile a module before it can be instantiated (otherwise, the WebAssembly.instantiate() function should be used). |
| f  [**WebAssembly.compileStreaming**](https://deno.land/api@v1.25.3?s=WebAssembly.compileStreaming) | The WebAssembly.compileStreaming() function compiles a WebAssembly.Module directly from a streamed underlying source. This function is useful if it is necessary to a compile a module before it can be instantiated (otherwise, the WebAssembly.instantiateStreaming() function should be used). |
| f  [**WebAssembly.instantiate**](https://deno.land/api@v1.25.3?s=WebAssembly.instantiate) | The WebAssembly.instantiate() function allows you to compile and instantiate WebAssembly code. |
| f  [**WebAssembly.instantiateStreaming**](https://deno.land/api@v1.25.3?s=WebAssembly.instantiateStreaming) | The WebAssembly.instantiateStreaming() function compiles and instantiates a WebAssembly module directly from a streamed underlying source. This is the most efficient, optimized way to load wasm code. |
| f  [**WebAssembly.validate**](https://deno.land/api@v1.25.3?s=WebAssembly.validate) | The WebAssembly.validate() function validates a given typed array of WebAssembly binary code, returning whether the bytes form a valid wasm module (true) or not (false). |

## [WebGPU](https://deno.land/api@v1.25.3#WebGPU)

|  |  |
| --- | --- |
| c  [**GPU**](https://deno.land/api@v1.25.3?s=GPU) |  |
| c  [**GPUAdapter**](https://deno.land/api@v1.25.3?s=GPUAdapter) |  |
| c  [**GPUAdapterInfo**](https://deno.land/api@v1.25.3?s=GPUAdapterInfo) |  |
| c  [**GPUBindGroup**](https://deno.land/api@v1.25.3?s=GPUBindGroup) |  |
| c  [**GPUBindGroupLayout**](https://deno.land/api@v1.25.3?s=GPUBindGroupLayout) |  |
| c  [**GPUBuffer**](https://deno.land/api@v1.25.3?s=GPUBuffer) |  |
| c  [**GPUBufferUsage**](https://deno.land/api@v1.25.3?s=GPUBufferUsage) |  |
| c  [**GPUColorWrite**](https://deno.land/api@v1.25.3?s=GPUColorWrite) |  |
| c  [**GPUCommandBuffer**](https://deno.land/api@v1.25.3?s=GPUCommandBuffer) |  |
| c  [**GPUCommandEncoder**](https://deno.land/api@v1.25.3?s=GPUCommandEncoder) |  |
| c  [**GPUComputePassEncoder**](https://deno.land/api@v1.25.3?s=GPUComputePassEncoder) |  |
| c  [**GPUComputePipeline**](https://deno.land/api@v1.25.3?s=GPUComputePipeline) |  |
| c  [**GPUDevice**](https://deno.land/api@v1.25.3?s=GPUDevice) |  |
| c  [**GPUError**](https://deno.land/api@v1.25.3?s=GPUError) |  |
| c  [**GPUMapMode**](https://deno.land/api@v1.25.3?s=GPUMapMode) |  |
| c  [**GPUOutOfMemoryError**](https://deno.land/api@v1.25.3?s=GPUOutOfMemoryError) |  |
| c  [**GPUPipelineLayout**](https://deno.land/api@v1.25.3?s=GPUPipelineLayout) |  |
| c  [**GPUQuerySet**](https://deno.land/api@v1.25.3?s=GPUQuerySet) |  |
| c  [**GPUQueue**](https://deno.land/api@v1.25.3?s=GPUQueue) |  |
| c  [**GPURenderBundle**](https://deno.land/api@v1.25.3?s=GPURenderBundle) |  |
| c  [**GPURenderBundleEncoder**](https://deno.land/api@v1.25.3?s=GPURenderBundleEncoder) |  |
| c  [**GPURenderPassEncoder**](https://deno.land/api@v1.25.3?s=GPURenderPassEncoder) |  |
| c  [**GPURenderPipeline**](https://deno.land/api@v1.25.3?s=GPURenderPipeline) |  |
| c  [**GPUSampler**](https://deno.land/api@v1.25.3?s=GPUSampler) |  |
| c  [**GPUShaderModule**](https://deno.land/api@v1.25.3?s=GPUShaderModule) |  |
| c  [**GPUShaderStage**](https://deno.land/api@v1.25.3?s=GPUShaderStage) |  |
| c  [**GPUSupportedFeatures**](https://deno.land/api@v1.25.3?s=GPUSupportedFeatures) |  |
| c  [**GPUSupportedLimits**](https://deno.land/api@v1.25.3?s=GPUSupportedLimits) |  |
| c  [**GPUTexture**](https://deno.land/api@v1.25.3?s=GPUTexture) |  |
| c  [**GPUTextureUsage**](https://deno.land/api@v1.25.3?s=GPUTextureUsage) |  |
| c  [**GPUTextureView**](https://deno.land/api@v1.25.3?s=GPUTextureView) |  |
| c  [**GPUUncapturedErrorEvent**](https://deno.land/api@v1.25.3?s=GPUUncapturedErrorEvent) |  |
| c  [**GPUValidationError**](https://deno.land/api@v1.25.3?s=GPUValidationError) |  |
| I  [**GPUBindGroupDescriptor**](https://deno.land/api@v1.25.3?s=GPUBindGroupDescriptor) |  |
| I  [**GPUBindGroupEntry**](https://deno.land/api@v1.25.3?s=GPUBindGroupEntry) |  |
| I  [**GPUBindGroupLayoutDescriptor**](https://deno.land/api@v1.25.3?s=GPUBindGroupLayoutDescriptor) |  |
| I  [**GPUBindGroupLayoutEntry**](https://deno.land/api@v1.25.3?s=GPUBindGroupLayoutEntry) |  |
| I  [**GPUBlendComponent**](https://deno.land/api@v1.25.3?s=GPUBlendComponent) |  |
| I  [**GPUBlendState**](https://deno.land/api@v1.25.3?s=GPUBlendState) |  |
| I  [**GPUBufferBinding**](https://deno.land/api@v1.25.3?s=GPUBufferBinding) |  |
| I  [**GPUBufferBindingLayout**](https://deno.land/api@v1.25.3?s=GPUBufferBindingLayout) |  |
| I  [**GPUBufferDescriptor**](https://deno.land/api@v1.25.3?s=GPUBufferDescriptor) |  |
| I  [**GPUColorDict**](https://deno.land/api@v1.25.3?s=GPUColorDict) |  |
| I  [**GPUColorTargetState**](https://deno.land/api@v1.25.3?s=GPUColorTargetState) |  |
| I  [**GPUCommandBufferDescriptor**](https://deno.land/api@v1.25.3?s=GPUCommandBufferDescriptor) |  |
| I  [**GPUCommandEncoderDescriptor**](https://deno.land/api@v1.25.3?s=GPUCommandEncoderDescriptor) |  |
| I  [**GPUCompilationInfo**](https://deno.land/api@v1.25.3?s=GPUCompilationInfo) |  |
| I  [**GPUCompilationMessage**](https://deno.land/api@v1.25.3?s=GPUCompilationMessage) |  |
| I  [**GPUComputePassDescriptor**](https://deno.land/api@v1.25.3?s=GPUComputePassDescriptor) |  |
| I  [**GPUComputePipelineDescriptor**](https://deno.land/api@v1.25.3?s=GPUComputePipelineDescriptor) |  |
| I  [**GPUDepthStencilState**](https://deno.land/api@v1.25.3?s=GPUDepthStencilState) |  |
| I  [**GPUDeviceDescriptor**](https://deno.land/api@v1.25.3?s=GPUDeviceDescriptor) |  |
| I  [**GPUDeviceLostInfo**](https://deno.land/api@v1.25.3?s=GPUDeviceLostInfo) |  |
| I  [**GPUExtent3DDict**](https://deno.land/api@v1.25.3?s=GPUExtent3DDict) |  |
| I  [**GPUFragmentState**](https://deno.land/api@v1.25.3?s=GPUFragmentState) |  |
| I  [**GPUImageCopyBuffer**](https://deno.land/api@v1.25.3?s=GPUImageCopyBuffer) |  |
| I  [**GPUImageCopyTexture**](https://deno.land/api@v1.25.3?s=GPUImageCopyTexture) |  |
| I  [**GPUImageDataLayout**](https://deno.land/api@v1.25.3?s=GPUImageDataLayout) |  |
| I  [**GPUMultisampleState**](https://deno.land/api@v1.25.3?s=GPUMultisampleState) |  |
| I  [**GPUObjectBase**](https://deno.land/api@v1.25.3?s=GPUObjectBase) |  |
| I  [**GPUObjectDescriptorBase**](https://deno.land/api@v1.25.3?s=GPUObjectDescriptorBase) |  |
| I  [**GPUOrigin3DDict**](https://deno.land/api@v1.25.3?s=GPUOrigin3DDict) |  |
| I  [**GPUPipelineBase**](https://deno.land/api@v1.25.3?s=GPUPipelineBase) |  |
| I  [**GPUPipelineDescriptorBase**](https://deno.land/api@v1.25.3?s=GPUPipelineDescriptorBase) |  |
| I  [**GPUPipelineLayoutDescriptor**](https://deno.land/api@v1.25.3?s=GPUPipelineLayoutDescriptor) |  |
| I  [**GPUPrimitiveState**](https://deno.land/api@v1.25.3?s=GPUPrimitiveState) |  |
| I  [**GPUProgrammablePassEncoder**](https://deno.land/api@v1.25.3?s=GPUProgrammablePassEncoder) |  |
| I  [**GPUProgrammableStage**](https://deno.land/api@v1.25.3?s=GPUProgrammableStage) |  |
| I  [**GPUQuerySetDescriptor**](https://deno.land/api@v1.25.3?s=GPUQuerySetDescriptor) |  |
| I  [**GPURenderBundleDescriptor**](https://deno.land/api@v1.25.3?s=GPURenderBundleDescriptor) |  |
| I  [**GPURenderBundleEncoderDescriptor**](https://deno.land/api@v1.25.3?s=GPURenderBundleEncoderDescriptor) |  |
| I  [**GPURenderEncoderBase**](https://deno.land/api@v1.25.3?s=GPURenderEncoderBase) |  |
| I  [**GPURenderPassColorAttachment**](https://deno.land/api@v1.25.3?s=GPURenderPassColorAttachment) |  |
| I  [**GPURenderPassDepthStencilAttachment**](https://deno.land/api@v1.25.3?s=GPURenderPassDepthStencilAttachment) |  |
| I  [**GPURenderPassDescriptor**](https://deno.land/api@v1.25.3?s=GPURenderPassDescriptor) |  |
| I  [**GPURenderPassLayout**](https://deno.land/api@v1.25.3?s=GPURenderPassLayout) |  |
| I  [**GPURenderPipelineDescriptor**](https://deno.land/api@v1.25.3?s=GPURenderPipelineDescriptor) |  |
| I  [**GPURequestAdapterOptions**](https://deno.land/api@v1.25.3?s=GPURequestAdapterOptions) |  |
| I  [**GPUSamplerBindingLayout**](https://deno.land/api@v1.25.3?s=GPUSamplerBindingLayout) |  |
| I  [**GPUSamplerDescriptor**](https://deno.land/api@v1.25.3?s=GPUSamplerDescriptor) |  |
| I  [**GPUShaderModuleDescriptor**](https://deno.land/api@v1.25.3?s=GPUShaderModuleDescriptor) |  |
| I  [**GPUStencilFaceState**](https://deno.land/api@v1.25.3?s=GPUStencilFaceState) |  |
| I  [**GPUStorageTextureBindingLayout**](https://deno.land/api@v1.25.3?s=GPUStorageTextureBindingLayout) |  |
| I  [**GPUTextureBindingLayout**](https://deno.land/api@v1.25.3?s=GPUTextureBindingLayout) |  |
| I  [**GPUTextureDescriptor**](https://deno.land/api@v1.25.3?s=GPUTextureDescriptor) |  |
| I  [**GPUTextureViewDescriptor**](https://deno.land/api@v1.25.3?s=GPUTextureViewDescriptor) |  |
| I  [**GPUUncapturedErrorEventInit**](https://deno.land/api@v1.25.3?s=GPUUncapturedErrorEventInit) |  |
| I  [**GPUVertexAttribute**](https://deno.land/api@v1.25.3?s=GPUVertexAttribute) |  |
| I  [**GPUVertexBufferLayout**](https://deno.land/api@v1.25.3?s=GPUVertexBufferLayout) |  |
| I  [**GPUVertexState**](https://deno.land/api@v1.25.3?s=GPUVertexState) |  |
| T  [**GPUAddressMode**](https://deno.land/api@v1.25.3?s=GPUAddressMode) |  |
| T  [**GPUAutoLayoutMode**](https://deno.land/api@v1.25.3?s=GPUAutoLayoutMode) |  |
| T  [**GPUBindingResource**](https://deno.land/api@v1.25.3?s=GPUBindingResource) |  |
| T  [**GPUBlendFactor**](https://deno.land/api@v1.25.3?s=GPUBlendFactor) |  |
| T  [**GPUBlendOperation**](https://deno.land/api@v1.25.3?s=GPUBlendOperation) |  |
| T  [**GPUBufferBindingType**](https://deno.land/api@v1.25.3?s=GPUBufferBindingType) |  |
| T  [**GPUBufferUsageFlags**](https://deno.land/api@v1.25.3?s=GPUBufferUsageFlags) |  |
| T  [**GPUColor**](https://deno.land/api@v1.25.3?s=GPUColor) |  |
| T  [**GPUColorWriteFlags**](https://deno.land/api@v1.25.3?s=GPUColorWriteFlags) |  |
| T  [**GPUCompareFunction**](https://deno.land/api@v1.25.3?s=GPUCompareFunction) |  |
| T  [**GPUCompilationMessageType**](https://deno.land/api@v1.25.3?s=GPUCompilationMessageType) |  |
| T  [**GPUCullMode**](https://deno.land/api@v1.25.3?s=GPUCullMode) |  |
| T  [**GPUDeviceLostReason**](https://deno.land/api@v1.25.3?s=GPUDeviceLostReason) |  |
| T  [**GPUErrorFilter**](https://deno.land/api@v1.25.3?s=GPUErrorFilter) |  |
| T  [**GPUExtent3D**](https://deno.land/api@v1.25.3?s=GPUExtent3D) |  |
| T  [**GPUFeatureName**](https://deno.land/api@v1.25.3?s=GPUFeatureName) |  |
| T  [**GPUFilterMode**](https://deno.land/api@v1.25.3?s=GPUFilterMode) |  |
| T  [**GPUFrontFace**](https://deno.land/api@v1.25.3?s=GPUFrontFace) |  |
| T  [**GPUIndexFormat**](https://deno.land/api@v1.25.3?s=GPUIndexFormat) |  |
| T  [**GPULoadOp**](https://deno.land/api@v1.25.3?s=GPULoadOp) |  |
| T  [**GPUMapModeFlags**](https://deno.land/api@v1.25.3?s=GPUMapModeFlags) |  |
| T  [**GPUMipmapFilterMode**](https://deno.land/api@v1.25.3?s=GPUMipmapFilterMode) |  |
| T  [**GPUOrigin3D**](https://deno.land/api@v1.25.3?s=GPUOrigin3D) |  |
| T  [**GPUPipelineStatisticName**](https://deno.land/api@v1.25.3?s=GPUPipelineStatisticName) |  |
| T  [**GPUPowerPreference**](https://deno.land/api@v1.25.3?s=GPUPowerPreference) |  |
| T  [**GPUPrimitiveTopology**](https://deno.land/api@v1.25.3?s=GPUPrimitiveTopology) |  |
| T  [**GPUQueryType**](https://deno.land/api@v1.25.3?s=GPUQueryType) |  |
| T  [**GPUSamplerBindingType**](https://deno.land/api@v1.25.3?s=GPUSamplerBindingType) |  |
| T  [**GPUShaderStageFlags**](https://deno.land/api@v1.25.3?s=GPUShaderStageFlags) |  |
| T  [**GPUStencilOperation**](https://deno.land/api@v1.25.3?s=GPUStencilOperation) |  |
| T  [**GPUStorageTextureAccess**](https://deno.land/api@v1.25.3?s=GPUStorageTextureAccess) |  |
| T  [**GPUStoreOp**](https://deno.land/api@v1.25.3?s=GPUStoreOp) |  |
| T  [**GPUTextureAspect**](https://deno.land/api@v1.25.3?s=GPUTextureAspect) |  |
| T  [**GPUTextureDimension**](https://deno.land/api@v1.25.3?s=GPUTextureDimension) |  |
| T  [**GPUTextureFormat**](https://deno.land/api@v1.25.3?s=GPUTextureFormat) |  |
| T  [**GPUTextureSampleType**](https://deno.land/api@v1.25.3?s=GPUTextureSampleType) |  |
| T  [**GPUTextureUsageFlags**](https://deno.land/api@v1.25.3?s=GPUTextureUsageFlags) |  |
| T  [**GPUTextureViewDimension**](https://deno.land/api@v1.25.3?s=GPUTextureViewDimension) |  |
| T  [**GPUVertexFormat**](https://deno.land/api@v1.25.3?s=GPUVertexFormat) |  |
| T  [**GPUVertexStepMode**](https://deno.land/api@v1.25.3?s=GPUVertexStepMode) |  |

## [Uncategorized](https://deno.land/api@v1.25.3#Uncategorized)

|  |
| --- |
| N  [**Deno**](https://deno.land/api@v1.25.3?s=Deno) |