

# TriEngine2D

v0.0.8

Generated by Doxygen 1.8.3.1

Fri Feb 15 2013 21:03:57



# Contents

<b>1</b>	<b>Main Page</b>	<b>1</b>
<b>2</b>	<b>Namespace Index</b>	<b>3</b>
2.1	Namespace List . . . . .	3
<b>3</b>	<b>Hierarchical Index</b>	<b>5</b>
3.1	Class Hierarchy . . . . .	5
<b>4</b>	<b>Class Index</b>	<b>7</b>
4.1	Class List . . . . .	7
<b>5</b>	<b>File Index</b>	<b>9</b>
5.1	File List . . . . .	9
<b>6</b>	<b>Namespace Documentation</b>	<b>11</b>
6.1	Package TriDevs . . . . .	11
6.2	Package TriDevs.TriEngine2D . . . . .	11
6.3	Package TriDevs.TriEngine2D.Audio . . . . .	11
6.3.1	Enumeration Type Documentation . . . . .	12
6.3.1.1	AudioFormat . . . . .	12
6.4	Package TriDevs.TriEngine2D.Extensions . . . . .	12
6.5	Package TriDevs.TriEngine2D.Helpers . . . . .	13
6.6	Package TriDevs.TriEngine2D.Input . . . . .	13
6.7	Package TriDevs.TriEngine2D.Input.Events . . . . .	13
6.7.1	Function Documentation . . . . .	14
6.7.1.1	KeyDownEventHandler . . . . .	14
6.7.1.2	KeyPressEventHandler . . . . .	14
6.7.1.3	KeyUpEventHandler . . . . .	14
6.7.1.4	MouseDownEventHandler . . . . .	14
6.7.1.5	MouseUpEventHandler . . . . .	14
6.7.1.6	MouseWheelChangedEventHandler . . . . .	15
6.7.1.7	MouseWheelDownEventHandler . . . . .	15
6.7.1.8	MouseWheelUpEventHandler . . . . .	15

6.8	Package TriDevs.TriEngine2D.Interfaces	15
6.9	Package TriDevs.TriEngine2D.Logging	15
6.10	Package TriDevs.TriEngine2D.Native	15
6.11	Package TriDevs.TriEngine2D.Serializing	16
6.12	Package TriDevs.TriEngine2D.UI	16
6.13	Package TriDevs.TriEngine2D.UI.Events	16
6.13.1	Function Documentation	16
6.13.1.1	ControlClickedEventHandler	16
<b>7</b>	<b>Class Documentation</b>	<b>17</b>
7.1	TriDevs.TriEngine2D.Audio.AudioManager Class Reference	17
7.1.1	Detailed Description	18
7.1.2	Constructor & Destructor Documentation	19
7.1.2.1	AudioManager	19
7.1.3	Member Function Documentation	19
7.1.3.1	Dispose	19
7.1.3.2	GetSong	19
7.1.3.3	GetSound	19
7.1.3.4	HasSong	20
7.1.3.5	HasSound	20
7.1.3.6	LoadSong	20
7.1.3.7	LoadSound	21
7.1.3.8	StopAll	21
7.1.3.9	StopAllSongs	22
7.1.3.10	StopAllSounds	22
7.2	TriDevs.TriEngine2D.UI.Color Struct Reference	22
7.2.1	Detailed Description	22
7.2.2	Constructor & Destructor Documentation	23
7.2.2.1	Color	23
7.2.2.2	Color	23
7.2.3	Member Function Documentation	23
7.2.3.1	ToVector3	23
7.2.3.2	ToVector4	23
7.2.4	Member Data Documentation	24
7.2.4.1	A	24
7.2.4.2	B	24
7.2.4.3	G	24
7.2.4.4	R	24
7.3	TriDevs.TriEngine2D.UI.Control Class Reference	24
7.3.1	Detailed Description	25

7.3.2	Member Function Documentation	26
7.3.2.1	Disable	26
7.3.2.2	Enable	26
7.3.2.3	Hide	26
7.3.2.4	OnClicked	26
7.3.2.5	Show	27
7.3.3	Property Documentation	27
7.3.3.1	Color	27
7.3.3.2	Enabled	27
7.3.3.3	Position	27
7.3.3.4	Size	27
7.3.3.5	Text	27
7.3.3.6	Visible	27
7.3.4	Event Documentation	27
7.3.4.1	Clicked	27
7.4	TriDevs.TriEngine2D.Extensions.EnumerationExtensions Class Reference	27
7.4.1	Detailed Description	27
7.4.2	Member Function Documentation	28
7.4.2.1	Has< T >	28
7.4.2.2	Include< T >	28
7.4.2.3	Missing< T >	29
7.4.2.4	Remove< T >	29
7.5	TriDevs.TriEngine2D.Native.Helpers Class Reference	30
7.5.1	Detailed Description	30
7.5.2	Member Function Documentation	30
7.5.2.1	GetErrorMessage	30
7.5.2.2	GetLastError	30
7.5.2.3	GetLastErrorInfo	31
7.5.2.4	GetLastErrorMessage	31
7.5.2.5	GetLastWin32Exception	31
7.5.2.6	GetWin32Exception	31
7.6	TriDevs.TriEngine2D.Audio.IAudioManager Interface Reference	32
7.6.1	Detailed Description	33
7.6.2	Member Function Documentation	33
7.6.2.1	GetSong	33
7.6.2.2	GetSound	33
7.6.2.3	HasSong	34
7.6.2.4	HasSound	34
7.6.2.5	LoadSong	34
7.6.2.6	LoadSound	34

7.6.2.7	StopAll	35
7.6.2.8	StopAllSongs	35
7.6.2.9	StopAllSounds	35
7.7	TriDevs.TriEngine2D.UI.IControl Interface Reference	35
7.7.1	Detailed Description	37
7.7.2	Member Function Documentation	37
7.7.2.1	Disable	37
7.7.2.2	Enable	37
7.7.2.3	Hide	37
7.7.2.4	Show	37
7.7.3	Property Documentation	37
7.7.3.1	Color	37
7.7.3.2	Enabled	37
7.7.3.3	Position	37
7.7.3.4	Size	37
7.7.3.5	Text	38
7.7.3.6	Visible	38
7.7.4	Event Documentation	38
7.7.4.1	Clicked	38
7.8	TriDevs.TriEngine2D.Interfaces.IDrawable Interface Reference	38
7.8.1	Detailed Description	38
7.8.2	Member Function Documentation	38
7.8.2.1	Draw	38
7.9	TriDevs.TriEngine2D.Input.IInputManager Interface Reference	38
7.9.1	Detailed Description	40
7.9.2	Member Function Documentation	40
7.9.2.1	IsKeyDown	40
7.9.2.2	IsKeyUp	40
7.9.2.3	IsMouseDown	41
7.9.2.4	IsMouseUp	41
7.9.2.5	IsWheelChanged	41
7.9.2.6	IsWheelDown	41
7.9.2.7	IsWheelUp	42
7.9.2.8	KeyPressed	42
7.9.2.9	KeyReleased	42
7.9.2.10	MousePressed	42
7.9.2.11	MouseReleased	43
7.9.2.12	Update	43
7.9.2.13	WheelChange	43
7.9.3	Property Documentation	43

7.9.3.1	<a href="#">MousePosition</a>	43
7.9.3.2	<a href="#">MouseWheelValue</a>	43
7.9.3.3	<a href="#">MouseX</a>	43
7.9.3.4	<a href="#">MouseY</a>	43
7.9.3.5	<a href="#">this[Key key]</a>	43
7.9.3.6	<a href="#">this[MouseButton button]</a>	44
7.9.4	<a href="#">Event Documentation</a>	44
7.9.4.1	<a href="#">KeyDown</a>	44
7.9.4.2	<a href="#">KeyPress</a>	44
7.9.4.3	<a href="#">KeyUp</a>	44
7.9.4.4	<a href="#">MouseDown</a>	44
7.9.4.5	<a href="#">MouseUp</a>	44
7.9.4.6	<a href="#">WheelChanged</a>	44
7.9.4.7	<a href="#">WheelDown</a>	44
7.9.4.8	<a href="#">WheelUp</a>	45
7.10	<a href="#">TriDevs.TriEngine2D.Input.InputManager Class Reference</a>	45
7.10.1	<a href="#">Detailed Description</a>	47
7.10.2	<a href="#">Constructor &amp; Destructor Documentation</a>	47
7.10.2.1	<a href="#">InputManager</a>	47
7.10.2.2	<a href="#">InputManager</a>	47
7.10.3	<a href="#">Member Function Documentation</a>	47
7.10.3.1	<a href="#">IsKeyDown</a>	47
7.10.3.2	<a href="#">IsKeyUp</a>	48
7.10.3.3	<a href="#">IsMouseDown</a>	48
7.10.3.4	<a href="#">IsMouseUp</a>	48
7.10.3.5	<a href="#">IsWheelChanged</a>	48
7.10.3.6	<a href="#">IsWheelDown</a>	49
7.10.3.7	<a href="#">IsWheelUp</a>	49
7.10.3.8	<a href="#">KeyPressed</a>	49
7.10.3.9	<a href="#">KeyReleased</a>	50
7.10.3.10	<a href="#">MousePressed</a>	50
7.10.3.11	<a href="#">MouseReleased</a>	50
7.10.3.12	<a href="#">Update</a>	51
7.10.3.13	<a href="#">WheelChange</a>	51
7.10.4	<a href="#">Property Documentation</a>	51
7.10.4.1	<a href="#">MousePosition</a>	51
7.10.4.2	<a href="#">MouseWheelValue</a>	51
7.10.4.3	<a href="#">MouseX</a>	51
7.10.4.4	<a href="#">MouseY</a>	51
7.10.4.5	<a href="#">this[Key key]</a>	51

7.10.4.6	<a href="#">this[MouseButton button]</a>	51
7.10.5	<a href="#">Event Documentation</a>	51
7.10.5.1	<a href="#">KeyDown</a>	51
7.10.5.2	<a href="#">KeyPress</a>	51
7.10.5.3	<a href="#">KeyUp</a>	51
7.10.5.4	<a href="#">MouseDown</a>	52
7.10.5.5	<a href="#">MouseUp</a>	52
7.10.5.6	<a href="#">WheelChanged</a>	52
7.10.5.7	<a href="#">WheelDown</a>	52
7.10.5.8	<a href="#">WheelUp</a>	52
7.11	<a href="#">TriDevs.TriEngine2D.Helpers.IO Class Reference</a>	52
7.11.1	<a href="#">Detailed Description</a>	52
7.11.2	<a href="#">Member Function Documentation</a>	52
7.11.2.1	<a href="#">GetAbsolutePath</a>	52
7.12	<a href="#">TriDevs.TriEngine2D.Audio.ISong Interface Reference</a>	53
7.12.1	<a href="#">Detailed Description</a>	54
7.12.2	<a href="#">Member Function Documentation</a>	54
7.12.2.1	<a href="#">Pause</a>	54
7.12.2.2	<a href="#">Play</a>	54
7.12.2.3	<a href="#">Resume</a>	54
7.12.2.4	<a href="#">Stop</a>	54
7.12.3	<a href="#">Property Documentation</a>	55
7.12.3.1	<a href="#">File</a>	55
7.12.3.2	<a href="#">IsLooped</a>	55
7.12.3.3	<a href="#">Name</a>	55
7.12.3.4	<a href="#">Volume</a>	55
7.13	<a href="#">TriDevs.TriEngine2D.Audio.ISound Interface Reference</a>	55
7.13.1	<a href="#">Detailed Description</a>	56
7.13.2	<a href="#">Member Function Documentation</a>	56
7.13.2.1	<a href="#">Play</a>	56
7.13.2.2	<a href="#">Stop</a>	56
7.13.3	<a href="#">Property Documentation</a>	57
7.13.3.1	<a href="#">File</a>	57
7.13.3.2	<a href="#">Name</a>	57
7.14	<a href="#">TriDevs.TriEngine2D.Input.Events.KeyCharEventArgs Class Reference</a>	57
7.14.1	<a href="#">Detailed Description</a>	58
7.14.2	<a href="#">Member Data Documentation</a>	58
7.14.2.1	<a href="#">KeyChar</a>	58
7.15	<a href="#">TriDevs.TriEngine2D.Input.Events.KeyEventArgs Class Reference</a>	58
7.15.1	<a href="#">Detailed Description</a>	59



7.15.2	Member Data Documentation	59
7.15.2.1	Key	59
7.16	TriDevs.TriEngine2D.UI.Label Class Reference	59
7.16.1	Detailed Description	61
7.16.2	Property Documentation	61
7.16.2.1	Text	61
7.17	TriDevs.TriEngine2D.UI.LinkLabel Class Reference	61
7.17.1	Detailed Description	62
7.17.2	Member Function Documentation	62
7.17.2.1	OnClicked	62
7.17.3	Property Documentation	63
7.17.3.1	Url	63
7.18	TriDevs.TriEngine2D.Logging.LogManager Class Reference	63
7.18.1	Detailed Description	63
7.18.2	Member Function Documentation	63
7.18.2.1	ClearOldLogs	63
7.18.2.2	DestroyConsole	64
7.18.2.3	GetLogger	64
7.18.2.4	LoadConfig	65
7.18.2.5	SetupConsole	65
7.19	TriDevs.TriEngine2D.Helpers.Math Class Reference	66
7.19.1	Detailed Description	66
7.19.2	Member Function Documentation	67
7.19.2.1	Clamp	67
7.19.2.2	Clamp	67
7.19.2.3	Clamp	67
7.19.2.4	Clamp	68
7.19.2.5	Clamp	68
7.19.2.6	Clamp	68
7.19.2.7	Clamp	69
7.19.2.8	Clamp	69
7.19.2.9	Clamp	69
7.20	TriDevs.TriEngine2D.Audio.NullAudioManager Class Reference	70
7.20.1	Detailed Description	71
7.20.2	Member Function Documentation	72
7.20.2.1	Dispose	72
7.20.2.2	GetSong	72
7.20.2.3	GetSound	72
7.20.2.4	HasSong	72
7.20.2.5	HasSound	73

7.20.2.6	LoadSong	73
7.20.2.7	LoadSound	73
7.20.2.8	StopAll	74
7.20.2.9	StopAllSongs	74
7.20.2.10	StopAllSounds	74
7.21	TriDevs.TriEngine2D.Input.NullInputManager Class Reference	74
7.21.1	Detailed Description	76
7.21.2	Member Function Documentation	76
7.21.2.1	IsKeyDown	76
7.21.2.2	IsKeyUp	77
7.21.2.3	IsMouseDown	77
7.21.2.4	IsMouseUp	77
7.21.2.5	IsWheelChanged	78
7.21.2.6	IsWheelDown	78
7.21.2.7	IsWheelUp	78
7.21.2.8	KeyPressed	78
7.21.2.9	KeyReleased	79
7.21.2.10	MousePressed	79
7.21.2.11	MouseReleased	79
7.21.2.12	Update	80
7.21.2.13	WheelChange	80
7.21.3	Property Documentation	80
7.21.3.1	MousePosition	80
7.21.3.2	MouseWheelValue	80
7.21.3.3	MouseX	80
7.21.3.4	MouseY	80
7.21.3.5	this[Key key]	80
7.21.3.6	this[MouseButton button]	80
7.21.4	Event Documentation	80
7.21.4.1	KeyDown	80
7.21.4.2	KeyPress	81
7.21.4.3	KeyUp	81
7.21.4.4	MouseDown	81
7.21.4.5	MouseUp	81
7.21.4.6	WheelChanged	81
7.21.4.7	WheelDown	81
7.21.4.8	WheelUp	81
7.22	TriDevs.TriEngine2D.Audio.NullSong Class Reference	81
7.22.1	Detailed Description	83
7.22.2	Member Function Documentation	83

7.22.2.1	Dispose	83
7.22.2.2	Pause	83
7.22.2.3	Play	83
7.22.2.4	Resume	83
7.22.2.5	Stop	83
7.22.3	Property Documentation	84
7.22.3.1	File	84
7.22.3.2	IsLooped	84
7.22.3.3	Name	84
7.22.3.4	Volume	84
7.23	TriDevs.TriEngine2D.Audio.NullSound Class Reference	84
7.23.1	Detailed Description	85
7.23.2	Member Function Documentation	85
7.23.2.1	Dispose	85
7.23.2.2	Play	85
7.23.2.3	Stop	86
7.23.3	Property Documentation	86
7.23.3.1	File	86
7.23.3.2	Name	86
7.24	TriDevs.TriEngine2D.Point< T > Struct Template Reference	86
7.24.1	Detailed Description	86
7.24.2	Constructor & Destructor Documentation	87
7.24.2.1	Point	87
7.24.3	Member Data Documentation	87
7.24.3.1	X	87
7.24.3.2	Y	87
7.25	TriDevs.TriEngine2D.Serializing.Serializer Class Reference	87
7.25.1	Detailed Description	87
7.25.2	Member Function Documentation	87
7.25.2.1	Deserialize< T >	87
7.25.2.2	Serialize< T >	88
7.25.2.3	Serialize< T >	88
7.26	TriDevs.TriEngine2D.Services Class Reference	89
7.26.1	Detailed Description	89
7.26.2	Member Function Documentation	89
7.26.2.1	Provide	89
7.26.2.2	Provide	90
7.26.2.3	Provide	90
7.26.3	Property Documentation	90
7.26.3.1	Audio	90

7.26.3.2	Input	90
7.27	TriDevs.TriEngine2D.Audio.Song Class Reference	90
7.27.1	Detailed Description	92
7.27.2	Member Function Documentation	92
7.27.2.1	Dispose	92
7.27.2.2	Pause	92
7.27.2.3	Play	92
7.27.2.4	Resume	92
7.27.2.5	Stop	93
7.27.3	Property Documentation	93
7.27.3.1	File	93
7.27.3.2	IsLooped	93
7.27.3.3	Name	93
7.27.3.4	Volume	93
7.28	TriDevs.TriEngine2D.Audio.Sound Class Reference	93
7.28.1	Detailed Description	94
7.28.2	Member Function Documentation	94
7.28.2.1	Dispose	94
7.28.2.2	Play	94
7.28.2.3	Stop	95
7.28.3	Property Documentation	95
7.28.3.1	File	95
7.28.3.2	Name	95
7.29	TriDevs.TriEngine2D.Extensions.StringExtensions Class Reference	95
7.29.1	Detailed Description	95
7.29.2	Member Function Documentation	95
7.29.2.1	Replace	95
7.29.2.2	Replace	96
7.29.2.3	ReplaceFirst	96
7.30	TriDevs.TriEngine2D.Helpers.Threading Class Reference	97
7.30.1	Detailed Description	97
7.30.2	Member Function Documentation	97
7.30.2.1	SetCurrentThreadName	97
7.31	TriDevs.TriEngine2D.Version Class Reference	97
7.31.1	Detailed Description	98
7.31.2	Member Data Documentation	98
7.31.2.1	Major	98
7.31.2.2	Minor	98
7.31.2.3	Patch	98
7.31.2.4	Suffix	98

7.31.2.5	VersionStringFormat	98
7.31.2.6	VersionStringFormatWithSuffix	98
7.31.3	Property Documentation	98
7.31.3.1	VersionString	98
7.32	TriDevs.TriEngine2D.Native.WinAPI Class Reference	99
7.32.1	Detailed Description	99
7.32.2	Member Function Documentation	99
7.32.2.1	AllocConsole	99
7.32.2.2	FreeConsole	99
7.32.2.3	GetStdHandle	100
7.32.3	Member Data Documentation	100
7.32.3.1	CODE_PAGE	100
7.32.3.2	STD_OUTPUT_HANDLE	100
<b>8</b>	<b>File Documentation</b>	<b>103</b>
8.1	README.md File Reference	103
8.2	TriDevs.TriEngine2D/Audio/AudioFormat.cs File Reference	103
8.3	TriDevs.TriEngine2D/Audio/AudioManager.cs File Reference	103
8.4	TriDevs.TriEngine2D/Audio/IAudioManager.cs File Reference	103
8.5	TriDevs.TriEngine2D/Audio/ISong.cs File Reference	104
8.6	TriDevs.TriEngine2D/Audio/ISound.cs File Reference	104
8.7	TriDevs.TriEngine2D/Audio/NullAudioManager.cs File Reference	104
8.8	TriDevs.TriEngine2D/Audio/NullSong.cs File Reference	104
8.9	TriDevs.TriEngine2D/Audio/NullSound.cs File Reference	105
8.10	TriDevs.TriEngine2D/Audio/Song.cs File Reference	105
8.11	TriDevs.TriEngine2D/Audio/Sound.cs File Reference	105
8.12	TriDevs.TriEngine2D/Extensions/EnumerationExtensions.cs File Reference	105
8.13	TriDevs.TriEngine2D/Extensions/StringExtensions.cs File Reference	106
8.14	TriDevs.TriEngine2D/Helpers/IO.cs File Reference	106
8.15	TriDevs.TriEngine2D/Helpers/Math.cs File Reference	106
8.16	TriDevs.TriEngine2D/Helpers/Threading.cs File Reference	106
8.17	TriDevs.TriEngine2D/Input/Events/Key.cs File Reference	107
8.18	TriDevs.TriEngine2D/Input/Events/Mouse.cs File Reference	107
8.19	TriDevs.TriEngine2D/Input/IInputManager.cs File Reference	108
8.20	TriDevs.TriEngine2D/Input/InputManager.cs File Reference	108
8.21	TriDevs.TriEngine2D/Input/NullInputManager.cs File Reference	108
8.22	TriDevs.TriEngine2D/Interfaces/IDrawable.cs File Reference	108
8.23	TriDevs.TriEngine2D/Logging/LogManager.cs File Reference	109
8.24	TriDevs.TriEngine2D/Native/Helpers.cs File Reference	109
8.25	TriDevs.TriEngine2D/Native/WinAPI.cs File Reference	109

8.26	TriDevs.TriEngine2D/Point.cs File Reference . . . . .	109
8.27	TriDevs.TriEngine2D/Properties/AssemblyInfo.cs File Reference . . . . .	110
8.28	TriDevs.TriEngine2D/Serializing/Serializer.cs File Reference . . . . .	110
8.29	TriDevs.TriEngine2D/Services.cs File Reference . . . . .	110
8.30	TriDevs.TriEngine2D/UI/Color.cs File Reference . . . . .	110
8.31	TriDevs.TriEngine2D/UI/Control.cs File Reference . . . . .	110
8.32	TriDevs.TriEngine2D/UI/Events/ControlClicked.cs File Reference . . . . .	111
8.33	TriDevs.TriEngine2D/UI/IControl.cs File Reference . . . . .	111
8.34	TriDevs.TriEngine2D/UI/Label.cs File Reference . . . . .	111
8.35	TriDevs.TriEngine2D/UI/LinkLabel.cs File Reference . . . . .	111
8.36	TriDevs.TriEngine2D/Version.cs File Reference . . . . .	112

<b>Index</b>	<b>112</b>
--------------	------------

# Chapter 1

## Main Page

2D general-purpose engine in C#/OpenGL

### IRC

[TriDevs](#) has an IRC channel, feel free to hop in if you have a question about anything: **Server:** irc.kottnet.net **Port:** 6667, 6697 (SSL) **Channel:** [TriDevs](#)

The channel topic contains further info.

### License

Copyright © 2013 by [Adam Hellberg](#), [Sijmen Schoon](#) and [Preston Shumway](#).

TriEngine2D is licensed under the [MIT License](#), more info can be found in the **LICENSE** file.

### Contributing

You are free to fork this project and make your own changes, as long as you follow the MIT License.

If you want to make a pull request, please do so to the [main project](#) and not any of the "official" forks.

For your pull request to be accepted, please follow our coding style:

- Indent with 4 spaces, not tabs.
- Curly braces placed on next line.
- All **public** methods, accessors and members must be properly documented.
- Use sensible variable names that describe what they are for.
- Method declarations written as:

```
“c# public void Hello(string world) “
```

- If your method accepts many parameters, it can be useful to put parameters on separate lines, as per this style:

```
“c# public void Hello(string world, bool print) “
```

- Please write tests for your code (not strictly required, but it's a plus)

By looking through the current source code, you should be able to get a good understanding of the formatting we use.

If you're using Visual Studio, you can change the indent behaviour by going to: **Tools -> Options -> Text Editor -> C# -> Tabs** and make sure "Insert spaces" is checked.

If you write tests for your code, please place these tests in their own project: "<strong>Namespace</strong> >.Tests", create said project if it does not exist (of type Class Library).

We use NUnit as test framework, feel free to use something else if you want to, but make sure you document what framework you are using and that it is freely available for anyone to obtain.

## Dependencies

TriEngine2D depends on [log4net](#), which is included in the **libs/log4net** folder.

TriEngine2D depends on [OpenTK](#), this is not included and you will have to build/install it yourself. OpenTK depends on OpenGL drivers being installed, they are usually in your normal video card drivers.

TriEngine2D depends on [Json.NET](#), this is not included, but is specified in the NuGet package config. If you [properly configure your NuGet settings](#), NuGet will automatically download Json.NET when building any projects that depend on it.

TriEngine2D depends on [NVorbis](#), which is included in the **libs/NVorbis** folder. [NVorbis is licensed under Ms-PL](#).

If you want to run the tests you will need to have [NUnit](#) installed.



## Chapter 2

# Namespace Index

### 2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

<a href="#">TriDevs</a>	11
<a href="#">TriDevs.TriEngine2D</a>	11
<a href="#">TriDevs.TriEngine2D.Audio</a>	11
<a href="#">TriDevs.TriEngine2D.Extensions</a>	12
<a href="#">TriDevs.TriEngine2D.Helpers</a>	13
<a href="#">TriDevs.TriEngine2D.Input</a>	13
<a href="#">TriDevs.TriEngine2D.Input.Events</a>	13
<a href="#">TriDevs.TriEngine2D.Interfaces</a>	15
<a href="#">TriDevs.TriEngine2D.Logging</a>	15
<a href="#">TriDevs.TriEngine2D.Native</a>	15
<a href="#">TriDevs.TriEngine2D.Serializing</a>	16
<a href="#">TriDevs.TriEngine2D.UI</a>	16
<a href="#">TriDevs.TriEngine2D.UI.Events</a>	16



## Chapter 3

# Hierarchical Index

### 3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

TriDevs.TriEngine2D.UI.Color . . . . .	22
TriDevs.TriEngine2D.Extensions.EnumerationExtensions . . . . .	27
EventArgs	
TriDevs.TriEngine2D.Input.Events.KeyCharEventArgs . . . . .	57
TriDevs.TriEngine2D.Input.Events.KeyEventArgs . . . . .	58
TriDevs.TriEngine2D.Native.Helpers . . . . .	30
TriDevs.TriEngine2D.UI.IControl . . . . .	35
TriDevs.TriEngine2D.UI.Control . . . . .	24
TriDevs.TriEngine2D.UI.Label . . . . .	59
TriDevs.TriEngine2D.UI.LinkLabel . . . . .	61
IDisposable	
TriDevs.TriEngine2D.Audio.IAudioManager . . . . .	32
TriDevs.TriEngine2D.Audio.AudioManager . . . . .	17
TriDevs.TriEngine2D.Audio.NullAudioManager . . . . .	70
TriDevs.TriEngine2D.Audio.ISong . . . . .	53
TriDevs.TriEngine2D.Audio.NullSong . . . . .	81
TriDevs.TriEngine2D.Audio.Song . . . . .	90
TriDevs.TriEngine2D.Audio.ISound . . . . .	55
TriDevs.TriEngine2D.Audio.NullSound . . . . .	84
TriDevs.TriEngine2D.Audio.Sound . . . . .	93
TriDevs.TriEngine2D.Interfaces.IDrawable . . . . .	38
TriDevs.TriEngine2D.Input.IInputManager . . . . .	38
TriDevs.TriEngine2D.Input.InputManager . . . . .	45
TriDevs.TriEngine2D.Input.NullInputManager . . . . .	74
TriDevs.TriEngine2D.Helpers.IO . . . . .	52
TriDevs.TriEngine2D.Logging.LogManager . . . . .	63
TriDevs.TriEngine2D.Helpers.Math . . . . .	66
TriDevs.TriEngine2D.Point< T > . . . . .	86
TriDevs.TriEngine2D.Serializing.Serializer . . . . .	87
TriDevs.TriEngine2D.Services . . . . .	89
TriDevs.TriEngine2D.Extensions.StringExtensions . . . . .	95
TriDevs.TriEngine2D.Helpers.Threading . . . . .	97
TriDevs.TriEngine2D.Version . . . . .	97
TriDevs.TriEngine2D.Native.WinAPI . . . . .	99



## Chapter 4

# Class Index

### 4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">TriDevs.TriEngine2D.Audio.AudioManager</a>	17
Class to manage engine audio. . . . .	
<a href="#">TriDevs.TriEngine2D.UI.Color</a>	22
Represents an RGBA color that can be used with <a href="#">TriEngine2D UI</a> controls. . . . .	
<a href="#">TriDevs.TriEngine2D.UI.Control</a>	24
Base control class that all other controls inherits from. Defines basic <a href="#">UI</a> control behaviour. . . . .	
<a href="#">TriDevs.TriEngine2D.Extensions.EnumerationExtensions</a>	27
<a href="#">Extensions</a> for System.Enum. . . . .	
<a href="#">TriDevs.TriEngine2D.Native.Helpers</a>	30
Helper class with various methods to help native coding and debugging. . . . .	
<a href="#">TriDevs.TriEngine2D.Audio.IAudioManager</a>	32
Provides various methods to manipulate audio. . . . .	
<a href="#">TriDevs.TriEngine2D.UI.IControl</a>	35
A <a href="#">UI</a> control that can be drawn on screen and interacted with. . . . .	
<a href="#">TriDevs.TriEngine2D.Interfaces.IDrawable</a>	38
Indicates that the implementing class is able to be drawn on the screen. . . . .	
<a href="#">TriDevs.TriEngine2D.Input.IInputManager</a>	38
Provides various methods to query input devices like the keyboard. . . . .	
<a href="#">TriDevs.TriEngine2D.Input.InputManager</a>	45
Input manager interfacing with input methods provided by a <a href="#">GameWindow</a> . . . . .	
<a href="#">TriDevs.TriEngine2D.Helpers.IO</a>	52
Provides various helper functions for doing <a href="#">IO</a> operations. . . . .	
<a href="#">TriDevs.TriEngine2D.Audio.ISong</a>	53
A song that will be streamed in the audio player. . . . .	
<a href="#">TriDevs.TriEngine2D.Audio.ISound</a>	55
A sound file for use with the audio manager. . . . .	
<a href="#">TriDevs.TriEngine2D.Input.Events.KeyCharEventArgs</a>	57
EventArgs class used for keychar-related events. Contains information about the character related with the event. . . . .	
<a href="#">TriDevs.TriEngine2D.Input.Events.KeyEventArgs</a>	58
EventArgs class used for key-related events. Contains information about the key related with the event. . . . .	
<a href="#">TriDevs.TriEngine2D.UI.Label</a>	59
A simple label to display text on the screen. . . . .	
<a href="#">TriDevs.TriEngine2D.UI.LinkLabel</a>	61
A label that, when clicked, will open a URL. . . . .	

<a href="#">TriDevs.TriEngine2D.Logging.LogManager</a>	
Class to manage logging. ILog interfaces should be obtained from this class' methods, as opposed to calling default log4net methods. . . . .	63
<a href="#">TriDevs.TriEngine2D.Helpers.Math</a>	
Various helper methods for working with math. . . . .	66
<a href="#">TriDevs.TriEngine2D.Audio.NullAudioManager</a>	
Used as a fallback <a href="#">AudioManager</a> object when the service locator fails to find one. . . . .	70
<a href="#">TriDevs.TriEngine2D.Input.NullInputManager</a>	
Used as a fallback <a href="#">InputManager</a> object when the service locator fails to find one. . . . .	74
<a href="#">TriDevs.TriEngine2D.Audio.NullSong</a>	
Fallback song class used in <a href="#">NullAudioManager</a> . . . . .	81
<a href="#">TriDevs.TriEngine2D.Audio.NullSound</a>	
Fallback sound class used in <a href="#">NullAudioManager</a> . . . . .	84
<a href="#">TriDevs.TriEngine2D.Point&lt; T &gt;</a>	
A struct representing an X/Y coordinate. . . . .	86
<a href="#">TriDevs.TriEngine2D.Serializing.Serializer</a>	
Provides serialization methods. . . . .	87
<a href="#">TriDevs.TriEngine2D.Services</a>	
Provides different game-related service interfaces. . . . .	89
<a href="#">TriDevs.TriEngine2D.Audio.Song</a>	
<a href="#">Song</a> class that can be used with <a href="#">AudioManager</a> . . . . .	90
<a href="#">TriDevs.TriEngine2D.Audio.Sound</a>	
<a href="#">Sound</a> class that can be used with the <a href="#">AudioManager</a> . . . . .	93
<a href="#">TriDevs.TriEngine2D.Extensions.StringExtensions</a>	
<a href="#">Extensions</a> for System.String . . . . .	95
<a href="#">TriDevs.TriEngine2D.Helpers.Threading</a>	
Provides various helper functions for doing threading operations. . . . .	97
<a href="#">TriDevs.TriEngine2D.Version</a>	
<a href="#">Version</a> class specifying the version of this project. . . . .	97
<a href="#">TriDevs.TriEngine2D.Native.WinAPI</a>	
Holds various <a href="#">WinAPI</a> stuff. . . . .	99

## Chapter 5

# File Index

### 5.1 File List

Here is a list of all files with brief descriptions:

TriDevs.TriEngine2D/ <a href="#">Point.cs</a>	109
TriDevs.TriEngine2D/ <a href="#">Services.cs</a>	110
TriDevs.TriEngine2D/ <a href="#">Version.cs</a>	112
TriDevs.TriEngine2D/Audio/ <a href="#">AudioFormat.cs</a>	103
TriDevs.TriEngine2D/Audio/ <a href="#">AudioManager.cs</a>	103
TriDevs.TriEngine2D/Audio/ <a href="#">IAudioManager.cs</a>	103
TriDevs.TriEngine2D/Audio/ <a href="#">ISong.cs</a>	104
TriDevs.TriEngine2D/Audio/ <a href="#">ISound.cs</a>	104
TriDevs.TriEngine2D/Audio/ <a href="#">NullAudioManager.cs</a>	104
TriDevs.TriEngine2D/Audio/ <a href="#">NullSong.cs</a>	104
TriDevs.TriEngine2D/Audio/ <a href="#">NullSound.cs</a>	105
TriDevs.TriEngine2D/Audio/ <a href="#">Song.cs</a>	105
TriDevs.TriEngine2D/Audio/ <a href="#">Sound.cs</a>	105
TriDevs.TriEngine2D/Extensions/ <a href="#">EnumerationExtensions.cs</a>	105
TriDevs.TriEngine2D/Extensions/ <a href="#">StringExtensions.cs</a>	106
TriDevs.TriEngine2D/Helpers/ <a href="#">IO.cs</a>	106
TriDevs.TriEngine2D/Helpers/ <a href="#">Math.cs</a>	106
TriDevs.TriEngine2D/Helpers/ <a href="#">Threading.cs</a>	106
TriDevs.TriEngine2D/Input/ <a href="#">IInputManager.cs</a>	108
TriDevs.TriEngine2D/Input/ <a href="#">InputManager.cs</a>	108
TriDevs.TriEngine2D/Input/ <a href="#">NullInputManager.cs</a>	108
TriDevs.TriEngine2D/Input/Events/ <a href="#">Key.cs</a>	107
TriDevs.TriEngine2D/Input/Events/ <a href="#">Mouse.cs</a>	107
TriDevs.TriEngine2D/Interfaces/ <a href="#">IDrawable.cs</a>	108
TriDevs.TriEngine2D/Logging/ <a href="#">LogManager.cs</a>	109
TriDevs.TriEngine2D/Native/ <a href="#">Helpers.cs</a>	109
TriDevs.TriEngine2D/Native/ <a href="#">WinAPI.cs</a>	109
TriDevs.TriEngine2D/Properties/ <a href="#">AssemblyInfo.cs</a>	110
TriDevs.TriEngine2D/Serializing/ <a href="#">Serializer.cs</a>	110
TriDevs.TriEngine2D/UI/ <a href="#">Color.cs</a>	110
TriDevs.TriEngine2D/UI/ <a href="#">Control.cs</a>	110
TriDevs.TriEngine2D/UI/ <a href="#">IControl.cs</a>	111
TriDevs.TriEngine2D/UI/ <a href="#">Label.cs</a>	111
TriDevs.TriEngine2D/UI/ <a href="#">LinkLabel.cs</a>	111
TriDevs.TriEngine2D/UI/Events/ <a href="#">ControlClicked.cs</a>	111





## Chapter 6

# Namespace Documentation

### 6.1 Package TriDevs

#### Namespaces

- package [TriEngine2D](#)

### 6.2 Package TriDevs.TriEngine2D

#### Namespaces

- package [Audio](#)
- package [Extensions](#)
- package [Helpers](#)
- package [Input](#)
- package [Interfaces](#)
- package [Logging](#)
- package [Native](#)
- package [Serializing](#)
- package [UI](#)

#### Classes

- struct [Point< T >](#)  
*A struct representing an X/Y coordinate.*
- class [Services](#)  
*Provides different game-related service interfaces.*
- class [Version](#)  
*[Version](#) class specifying the version of this project.*

### 6.3 Package TriDevs.TriEngine2D.Audio

#### Classes

- class [AudioManager](#)  
*Class to manage engine audio.*

- interface [IAudioManager](#)  
*Provides various methods to manipulate audio.*
- interface [ISong](#)  
*A song that will be streamed in the audio player.*
- interface [ISound](#)  
*A sound file for use with the audio manager.*
- class [NullAudioManager](#)  
*Used as a fallback [AudioManager](#) object when the service locator fails to find one.*
- class [NullSong](#)  
*Fallback song class used in [NullAudioManager](#).*
- class [NullSound](#)  
*Fallback sound class used in [NullAudioManager](#).*
- class [Song](#)  
*[Song](#) class that can be used with [AudioManager](#).*
- class [Sound](#)  
*[Sound](#) class that can be used with the [AudioManager](#).*

## Enumerations

- enum [AudioFormat](#) { [Ogg](#), [Wav](#) }  
*Different audio formats usable with audio managers.*

### 6.3.1 Enumeration Type Documentation

#### 6.3.1.1 enum TriDevs.TriEngine2D.Audio.AudioFormat

Different audio formats usable with audio managers.

#### Enumerator

**Ogg** The Ogg Vorbis format.

**Wav** The WAVE format

```

30     {
31         Ogg,
32         Wav
33     }
34
35
36
37
38
39
40

```

## 6.4 Package TriDevs.TriEngine2D.Extensions

### Classes

- class [EnumerationExtensions](#)  
*[Extensions](#) for System.Enum.*
- class [StringExtensions](#)  
*[Extensions](#) for System.String*

## 6.5 Package TriDevs.TriEngine2D.Helpers

### Classes

- class [IO](#)  
*Provides various helper functions for doing [IO](#) operations.*
- class [Math](#)  
*Various helper methods for working with math.*
- class [Threading](#)  
*Provides various helper functions for doing threading operations.*

## 6.6 Package TriDevs.TriEngine2D.Input

### Namespaces

- package [Events](#)

### Classes

- interface [IInputManager](#)  
*Provides various methods to query input devices like the keyboard.*
- class [InputManager](#)  
*[Input](#) manager interfacing with input methods provided by a [GameWindow](#).*
- class [NullInputManager](#)  
*Used as a fallback [InputManager](#) object when the service locator fails to find one.*

## 6.7 Package TriDevs.TriEngine2D.Input.Events

### Classes

- class [KeyEventArgs](#)  
*EventArgs class used for key-related events. Contains information about the key related with the event.*
- class [KeyCharEventArgs](#)  
*EventArgs class used for keychar-related events. Contains information about the character related with the event.*

### Functions

- delegate void [KeyDownEventHandler](#) (object sender, [KeyEventArgs](#) e)  
*Event handler delegate for the [KeyDown](#) event.*
- delegate void [KeyUpEventHandler](#) (object sender, [KeyEventArgs](#) e)  
*Event handler delegate for the [KeyUp](#) event.*
- delegate void [KeyPressEventHandler](#) (object sender, [KeyCharEventArgs](#) e)  
*Event handler delegate for the [KeyPress](#) event.*
- delegate void [MouseDownEventHandler](#) (object sender, [MouseButtonEventArgs](#) e)  
*Event handler delegate for the [MouseDown](#) event.*
- delegate void [MouseUpEventHandler](#) (object sender, [MouseButtonEventArgs](#) e)  
*Event handler delegate for the [MouseUp](#) event.*
- delegate void [MouseWheelChangedEventHandler](#) (object sender, [MouseWheelEventArgs](#) e)  
*Event handler delegate for the [MouseWheelChanged](#) event.*

- delegate void [MouseWheelDownEventHandler](#) (object sender, MouseEventArgs e)  
*Event handler delegate for the MouseWheelDown event.*
- delegate void [MouseWheelUpEventHandler](#) (object sender, MouseEventArgs e)  
*Event handler delegate for the MouseWheelUp event.*

### 6.7.1 Function Documentation

#### 6.7.1.1 delegate void TriDevs.TriEngine2D.Input.Events.KeyDownEventHandler ( object sender, EventArgs e )

Event handler delegate for the KeyDown event.

##### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	<a href="#">EventArgs</a> object with information about the event.

#### 6.7.1.2 delegate void TriDevs.TriEngine2D.Input.Events.KeyPressEventHandler ( object sender, KeyCharEventArgs e )

Event handler delegate for the KeyPress event.

##### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	<a href="#">KeyEventArgs</a> object with information about the event.

#### 6.7.1.3 delegate void TriDevs.TriEngine2D.Input.Events.KeyUpEventHandler ( object sender, EventArgs e )

Event handler delegate for the KeyUp event.

##### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	<a href="#">EventArgs</a> object with information about the event.

#### 6.7.1.4 delegate void TriDevs.TriEngine2D.Input.Events.MouseDownEventHandler ( object sender, MouseButtonEventArgs e )

Event handler delegate for the MouseDown event.

##### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	Mouse button information associated with the event.

#### 6.7.1.5 delegate void TriDevs.TriEngine2D.Input.Events.MouseUpEventHandler ( object sender, MouseButtonEventArgs e )

Event handler delegate for the MouseUp event.

##### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	Mouse button information associated with the event.

6.7.1.6 `delegate void TriDevs.TriEngine2D.Input.Events.MouseWheelChangedEventHandler ( object sender, MouseWheelEventArgs e )`

Event handler delegate for the MouseWheelChanged event.

#### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	Mouse wheel information associated with the event.

6.7.1.7 `delegate void TriDevs.TriEngine2D.Input.Events.MouseWheelDownEventHandler ( object sender, MouseWheelEventArgs e )`

Event handler delegate for the MouseWheelDown event.

#### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	Mouse wheel information associated with the event.

6.7.1.8 `delegate void TriDevs.TriEngine2D.Input.Events.MouseWheelUpEventHandler ( object sender, MouseWheelEventArgs e )`

Event handler delegate for the MouseWheelUp event.

#### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	Mouse wheel information associated with the event.

## 6.8 Package TriDevs.TriEngine2D.Interfaces

### Classes

- interface [IDrawable](#)

*Indicates that the implementing class is able to be drawn on the screen.*

## 6.9 Package TriDevs.TriEngine2D.Logging

### Classes

- class [LogManager](#)

*Class to manage logging. ILog interfaces should be obtained from this class' methods, as opposed to calling default log4net methods.*

## 6.10 Package TriDevs.TriEngine2D.Native

### Classes

- class [Helpers](#)

*Helper class with various methods to help native coding and debugging.*

- class [WinAPI](#)

*Holds various [WinAPI](#) stuff.*

## 6.11 Package TriDevs.TriEngine2D.Serializing

### Classes

- class [Serializer](#)

*Provides serialization methods.*

## 6.12 Package TriDevs.TriEngine2D.UI

### Namespaces

- package [Events](#)

### Classes

- struct [Color](#)

*Represents an RGBA color that can be used with [TriEngine2D UI](#) controls.*

- class [Control](#)

*Base control class that all other controls inherits from. Defines basic [UI](#) control behaviour.*

- interface [IControl](#)

*A [UI](#) control that can be drawn on screen and interacted with.*

- class [Label](#)

*A simple label to display text on the screen.*

- class [LinkLabel](#)

*A label that, when clicked, will open a URL.*

## 6.13 Package TriDevs.TriEngine2D.UI.Events

### Functions

- delegate void [ControlClickedEventHandler](#) (object sender, EventArgs e)

*Delegate handler for the control clicked event.*

### 6.13.1 Function Documentation

#### 6.13.1.1 delegate void TriDevs.TriEngine2D.UI.Events.ControlClickedEventHandler ( object sender, EventArgs e )

Delegate handler for the control clicked event.

#### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	

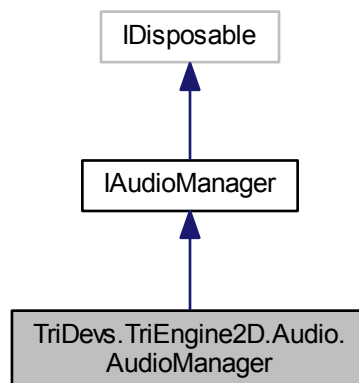
## Chapter 7

# Class Documentation

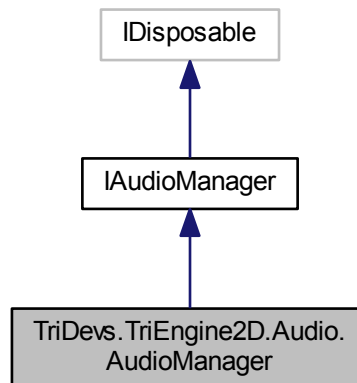
### 7.1 TriDevs.TriEngine2D.Audio.AudioManager Class Reference

Class to manage engine audio.

Inheritance diagram for TriDevs.TriEngine2D.Audio.AudioManager:



Collaboration diagram for TriDevs.TriEngine2D.Audio.AudioManager:



## Public Member Functions

- [AudioManager](#) ()  
*Creates a new instance of [AudioManager](#).*
- void [Dispose](#) ()
- void [StopAll](#) ()  
*Immediately stops all playback of sounds and songs.*
- [ISound LoadSound](#) (string name, string file, [AudioFormat](#) format=AudioFormat.Wav)  
*Loads a sound into the audio manager.*
- bool [HasSound](#) (string name)  
*Returns a value indicating whether a sound with the specified name has been loaded into the audio manager.*
- [ISound GetSound](#) (string name)  
*Gets the sound with the specified name.*
- void [StopAllSounds](#) ()  
*Immediately stops playback of all sounds.*
- [ISong LoadSong](#) (string name, string file, [AudioFormat](#) format=AudioFormat.Ogg)  
*Load a song into the audio manager.*
- bool [HasSong](#) (string name)  
*Returns a value indicating whether a song with the specified name has been loaded into the audio manager.*
- [ISong GetSong](#) (string name)  
*Gets the song with the specified name.*
- void [StopAllSongs](#) ()  
*Immediately stops playback of all songs.*

### 7.1.1 Detailed Description

Class to manage engine audio.



## 7.1.2 Constructor & Destructor Documentation

### 7.1.2.1 TriDevs.TriEngine2D.Audio.AudioManager.AudioManager ( )

Creates a new instance of [AudioManager](#).

```

47     {
48         _context = new AudioContext();
49         _oggStreamer = new OggStreamer();
50         _sounds = new List<ISound>();
51         _songs = new List<ISong>();
52     }

```

## 7.1.3 Member Function Documentation

### 7.1.3.1 void TriDevs.TriEngine2D.Audio.AudioManager.Dispose ( )

```

55     {
56         foreach (var sound in _sounds.Where(sound => sound != null))
57             sound.Dispose();
58
59         foreach (var song in _songs.Where(song => song != null))
60             song.Dispose();
61
62         if (_oggStreamer != null)
63         {
64             _oggStreamer.Dispose();
65             _oggStreamer = null;
66         }
67
68         if (_context != null)
69         {
70             _context.Dispose();
71             _context = null;
72         }
73     }

```

### 7.1.3.2 ISong TriDevs.TriEngine2D.Audio.AudioManager.GetSong ( string name )

Gets the song with the specified name.

#### Parameters

<i>name</i>	Name of the song to get.
-------------	--------------------------

#### Returns

An object implementing the [ISong](#) interface, or null if no song matched the requested name.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

132     {
133         return _songs.FirstOrDefault(s => s.Name == name);
134     }

```

### 7.1.3.3 ISound TriDevs.TriEngine2D.Audio.AudioManager.GetSound ( string name )

Gets the sound with the specified name.

#### Parameters

<i>name</i>	Name of the sound to get.
-------------	---------------------------

**Returns**

An object implementing the [ISound](#) interface, or null if no sound matched the requested name.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

102     {
103         return _sounds.FirstOrDefault(s => s.Name == name);
104     }

```

**7.1.3.4 bool TriDevs.TriEngine2D.Audio.AudioManager.HasSong ( string name )**

Returns a value indicating whether a song with the specified name has been loaded into the audio manager.

**Parameters**

<i>name</i>	Name to check for.
-------------	--------------------

**Returns**

True if the song has been loaded, false otherwise.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

127     {
128         return _songs.Any(s => s.Name == name);
129     }

```

**7.1.3.5 bool TriDevs.TriEngine2D.Audio.AudioManager.HasSound ( string name )**

Returns a value indicating whether a sound with the specified name has been loaded into the audio manager.

**Parameters**

<i>name</i>	Name to check for.
-------------	--------------------

**Returns**

True if the sound has been loaded, false otherwise.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

97     {
98         return _sounds.Any(s => s.Name == name);
99     }

```

**7.1.3.6 ISong TriDevs.TriEngine2D.Audio.AudioManager.LoadSong ( string name, string file, AudioFormat format = AudioFormat.Ogg )**

Load a song into the audio manager.

If a song with the same name has already been loaded, it will be returned instead.

**Parameters**

<i>name</i>	Name to assign to the new song object.
<i>file</i>	Path to the file that will be used.
<i>format</i>	Format of the file.

**Returns**

The resulting object implementing the [ISong](#) interface.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

112     {
113         var existing = _songs.FirstOrDefault(s => s.Name == name);
114         if (existing != null)
115             return existing;
116
117         var loaded = _songs.FirstOrDefault(s => s.File == file) != null;
118         if (loaded)
119             throw new Exception("The song file \"" + file + "\" has already been loaded under a
different name.");
120
121         var song = new Song(name, file, format);
122         _songs.Add(song);
123         return song;
124     }

```

### 7.1.3.7 ISound TriDevs.TriEngine2D.Audio.AudioManager.LoadSound ( string name, string file, AudioFormat format = AudioFormat.Wav )

Loads a sound into the audio manager.

If a sound with the same name has already been loaded, it will be returned instead.

**Parameters**

<i>name</i>	Name to assign to the new sound object.
<i>file</i>	Path to the file that will be used.
<i>format</i>	The format of the file to load.

**Returns**

The resulting object implementing the [ISound](#) interface.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

82     {
83         var existing = _sounds.FirstOrDefault(s => s.Name == name);
84         if (existing != null)
85             return existing;
86
87         var loaded = _sounds.FirstOrDefault(s => s.File == file) != null;
88         if (loaded)
89             throw new Exception("The sound file \"" + file + "\" has already been loaded under a
different name.");
90
91         var sound = new Sound(name, file, format);
92         _sounds.Add(sound);
93         return sound;
94     }

```

### 7.1.3.8 void TriDevs.TriEngine2D.Audio.AudioManager.StopAll ( )

Immediately stops all playback of sounds and songs.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

76     {
77         StopAllSounds();
78         StopAllSongs();
79     }

```

### 7.1.3.9 void TriDevs.TriEngine2D.Audio.AudioManager.StopAllSongs ( )

Immediately stops playback of all songs.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```
137         {
138             _songs.ForEach(s => s.Stop());
139         }
```

### 7.1.3.10 void TriDevs.TriEngine2D.Audio.AudioManager.StopAllSounds ( )

Immediately stops playback of all sounds.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```
107         {
108             _sounds.ForEach(s => s.Stop());
109         }
```

The documentation for this class was generated from the following file:

- [TriDevs.TriEngine2D/Audio/AudioManager.cs](#)

## 7.2 TriDevs.TriEngine2D.UI.Color Struct Reference

Represents an RGBA color that can be used with [TriEngine2D UI](#) controls.

### Public Member Functions

- [Color](#) (byte r, byte g, byte b, byte a=255)  
*Creates a new color with the specified red, green, blue and alpha values.*
- [Color](#) (float r, float g, float b, float a=1.0f)  
*Creates a new color with the specified red, green, blue and alpha values.*
- [Vector4 ToVector4](#) ()  
*Returns a Vector4 representation of this color. This can be used with most OpenTK methods.*
- [Vector3 ToVector3](#) ()  
*Returns a Vector3 representation of this color (ommits alpha value). This can be used with most OpenTK methods.*

### Public Attributes

- readonly float [R](#)  
*The red component of the color.*
- readonly float [G](#)  
*The green component of the color.*
- readonly float [B](#)  
*The blue component of the color.*
- readonly float [A](#)  
*The color's alpha value.*

#### 7.2.1 Detailed Description

Represents an RGBA color that can be used with [TriEngine2D UI](#) controls.

## 7.2.2 Constructor & Destructor Documentation

### 7.2.2.1 TriDevs.TriEngine2D.UI.Color.Color ( byte *r*, byte *g*, byte *b*, byte *a* = 255 )

Creates a new color with the specified red, green, blue and alpha values.

#### Parameters

<i>r</i>	Value of the red component (0-255).
<i>g</i>	Value of the green component (0-255).
<i>b</i>	Value of the blue component (0-255).
<i>a</i>	Alpha value (0-255) where 0 is transparent and 255 is opaque.

```

60                                     : this(r / 255.0f, g / 255.0f, b / 255.0f, a /
61         255.0f)
62         {
63         }

```

### 7.2.2.2 TriDevs.TriEngine2D.UI.Color.Color ( float *r*, float *g*, float *b*, float *a* = 1.0f )

Creates a new color with the specified red, green, blue and alpha values.

#### Parameters

<i>r</i>	Value of the red component (0.0-1.0).
<i>g</i>	Value of the green component (0.0-1.0).
<i>b</i>	Value of the blue component (0.0-1.0).
<i>a</i>	Alpha value (0.0-1.0) where 0.0 is transparent and 1.0 is opaque.

```

73         {
74             R = 0.0f;
75             R = Helpers.Math.Clamp(r, 0.0f, 1.0f);
76             G = Helpers.Math.Clamp(g, 0.0f, 1.0f);
77             B = Helpers.Math.Clamp(b, 0.0f, 1.0f);
78             A = Helpers.Math.Clamp(a, 0.0f, 1.0f);
79         }

```

## 7.2.3 Member Function Documentation

### 7.2.3.1 Vector3 TriDevs.TriEngine2D.UI.Color.ToVector3 ( )

Returns a Vector3 representation of this color (ommits alpha value). This can be used with most OpenTK methods.

#### Returns

```

97         {
98             return new Vector3(R, G, B);
99         }

```

### 7.2.3.2 Vector4 TriDevs.TriEngine2D.UI.Color.ToVector4 ( )

Returns a Vector4 representation of this color. This can be used with most OpenTK methods.

#### Returns

```

87         {
88             return new Vector4(R, G, B, A);
89         }

```

## 7.2.4 Member Data Documentation

### 7.2.4.1 readonly float TriDevs.TriEngine2D.UI.Color.A

The color's alpha value.

### 7.2.4.2 readonly float TriDevs.TriEngine2D.UI.Color.B

The blue component of the color.

### 7.2.4.3 readonly float TriDevs.TriEngine2D.UI.Color.G

The green component of the color.

### 7.2.4.4 readonly float TriDevs.TriEngine2D.UI.Color.R

The red component of the color.

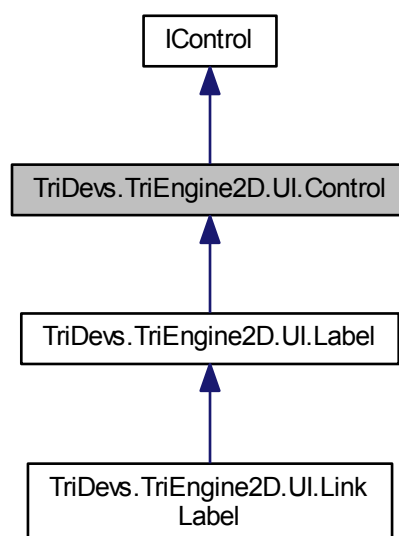
The documentation for this struct was generated from the following file:

- TriDevs.TriEngine2D/UI/[Color.cs](#)

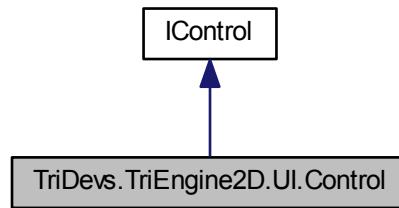
## 7.3 TriDevs.TriEngine2D.UI.Control Class Reference

Base control class that all other controls inherits from. Defines basic [UI](#) control behaviour.

Inheritance diagram for TriDevs.TriEngine2D.UI.Control:



Collaboration diagram for TriDevs.TriEngine2D.UI.Control:



### Public Member Functions

- virtual void [Enable](#) ()  
*Enables the control.*
- virtual void [Disable](#) ()  
*Disables the control.*
- virtual void [Show](#) ()  
*Shows the control.*
- virtual void [Hide](#) ()  
*Hides the control.*

### Protected Member Functions

- virtual void [OnClicked](#) ()

### Properties

- virtual bool [Enabled](#) [get, set]
- virtual bool [Visible](#) [get, set]
- virtual [Color](#) [Color](#) [get, set]
- virtual [Point< int >](#) [Position](#) [get, set]
- virtual [Point< int >](#) [Size](#) [get, set]
- virtual string [Text](#) [get, set]

### Events

- [ControlClickedEventHandler](#) [Clicked](#)  
*Raised when this control is clicked on by the user.*

#### 7.3.1 Detailed Description

Base control class that all other controls inherits from. Defines basic [UI](#) control behaviour.

## 7.3.2 Member Function Documentation

### 7.3.2.1 virtual void TriDevs.TriEngine2D.UI.Control.Disable ( ) [virtual]

Disables the control.

Implements [TriDevs.TriEngine2D.UI.IControl](#).

```

73         {
74             Enabled = false;
75         }

```

### 7.3.2.2 virtual void TriDevs.TriEngine2D.UI.Control.Enable ( ) [virtual]

Enables the control.

Implements [TriDevs.TriEngine2D.UI.IControl](#).

```

68         {
69             Enabled = true;
70         }

```

### 7.3.2.3 virtual void TriDevs.TriEngine2D.UI.Control.Hide ( ) [virtual]

Hides the control.

Implements [TriDevs.TriEngine2D.UI.IControl](#).

```

83         {
84             Visible = false;
85         }

```

### 7.3.2.4 virtual void TriDevs.TriEngine2D.UI.Control.OnClicked ( ) [protected],[virtual]

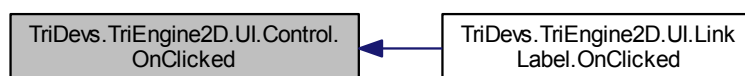
Reimplemented in [TriDevs.TriEngine2D.UI.LinkLabel](#).

```

52         {
53             var func = Clicked;
54             if (func != null)
55                 func(this, null);
56         }

```

Here is the caller graph for this function:





7.3.2.5 virtual void TriDevs.TriEngine2D.UI.Control.Show ( ) [virtual]

Shows the control.

Implements [TriDevs.TriEngine2D.UI.IControl](#).

```

78         {
79             Visible = true;
80         }

```

### 7.3.3 Property Documentation

7.3.3.1 virtual Color TriDevs.TriEngine2D.UI.Control.Color [get], [set]

7.3.3.2 virtual bool TriDevs.TriEngine2D.UI.Control.Enabled [get], [set]

7.3.3.3 virtual Point<int> TriDevs.TriEngine2D.UI.Control.Position [get], [set]

7.3.3.4 virtual Point<int> TriDevs.TriEngine2D.UI.Control.Size [get], [set]

7.3.3.5 virtual string TriDevs.TriEngine2D.UI.Control.Text [get], [set]

7.3.3.6 virtual bool TriDevs.TriEngine2D.UI.Control.Visible [get], [set]

### 7.3.4 Event Documentation

7.3.4.1 ControlClickedEventHandler TriDevs.TriEngine2D.UI.Control.Clicked

Raised when this control is clicked on by the user.

The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/UI/[Control.cs](#)

## 7.4 TriDevs.TriEngine2D.Extensions.EnumerationExtensions Class Reference

[Extensions](#) for System.Enum.

### Static Public Member Functions

- static T [Include< T >](#) (this Enum value, T append)  
*Includes an enumerated type and returns the new value.*
- static T [Remove< T >](#) (this Enum value, T remove)  
*Removes an enumerated type and returns the new value.*
- static bool [Has< T >](#) (this Enum value, T check)  
*Checks if an enumerated type contains a value.*
- static bool [Missing< T >](#) (this Enum obj, T value)  
*Checks if an enumerated type is missing a value.*

### 7.4.1 Detailed Description

[Extensions](#) for System.Enum.

## 7.4.2 Member Function Documentation

### 7.4.2.1 static bool TriDevs.TriEngine2D.Extensions.EnumerationExtensions.Has< T > ( this Enum value, T check ) [static]

Checks if an enumerated type contains a value.

#### Template Parameters

<i>T</i>	The enum type.
----------	----------------

#### Parameters

<i>value</i>	The enum to check.
<i>check</i>	Value to check for.

#### Returns

True if the enum has the value(s), false otherwise.

```

148     {
149         Type type = value.GetType();
150
151         //determine the values
152         var parsed = new _Value(check, type);
153         if (parsed.Signed.HasValue) //if (parsed.Signed is long)
154         {
155             return (Convert.ToInt64(value) & (long)parsed.Signed) == (long)parsed.Signed;
156         }
157         if (parsed.Unsigned.HasValue) //if (parsed.Unsigned is ulong)
158         {
159             return (Convert.ToUInt64(value) & (ulong)parsed.Unsigned) == (ulong)parsed.Unsigned;
160         }
161         return false;
162     }

```

### 7.4.2.2 static T TriDevs.TriEngine2D.Extensions.EnumerationExtensions.Include< T > ( this Enum value, T append ) [static]

Includes an enumerated type and returns the new value.

#### Template Parameters

<i>T</i>	The enum type.
----------	----------------

#### Parameters

<i>value</i>	The enum to append to.
<i>append</i>	Value to append.

#### Returns

New enum T with the new values.

```

94     {
95         var type = value.GetType();
96
97         //determine the values
98         object result = value;
99         var parsed = new _Value(append, type);
100         if (parsed.Signed.HasValue) //if (parsed.Signed is long)
101         {
102             result = Convert.ToInt64(value) | (long)parsed.Signed;
103         }

```

```

104         else if (parsed.Unsigned.HasValue) //else if (parsed.Unsigned is ulong)
105         {
106             result = Convert.ToUInt64(value) | (ulong)parsed.Unsigned;
107         }
108
109         //return the final value
110         return (T)Enum.Parse(type, result.ToString());
111     }

```

**7.4.2.3** static bool TriDevs.TriEngine2D.Extensions.EnumerationExtensions.Missing< T > ( this Enum *obj*, T *value* )  
[static]

Checks if an enumerated type is missing a value.

#### Template Parameters

<i>T</i>	The enum type.
----------	----------------

#### Parameters

<i>obj</i>	The enum to check.
<i>value</i>	Value to check for.

#### Returns

True if the enum is missing the value(s), false otherwise.

```

172     {
173         return !Has(obj, value);
174     }

```

**7.4.2.4** static T TriDevs.TriEngine2D.Extensions.EnumerationExtensions.Remove< T > ( this Enum *value*, T *remove* )  
[static]

Removes an enumerated type and returns the new value.

#### Template Parameters

<i>T</i>	The enum type.
----------	----------------

#### Parameters

<i>value</i>	The enum to remove from.
<i>remove</i>	Value to remove.

#### Returns

New enum T with the value(s) removed.

```

121     {
122         Type type = value.GetType();
123
124         //determine the values
125         object result = value;
126         var parsed = new _Value(remove, type);
127         if (parsed.Signed.HasValue) //if (parsed.Signed is long)
128         {
129             result = Convert.ToInt64(value) & ~(long)parsed.Signed;
130         }
131         else if (parsed.Unsigned.HasValue) //else if (parsed.Unsigned is ulong)
132         {
133             result = Convert.ToUInt64(value) & ~(ulong)parsed.Unsigned;

```

```

134         }
135
136         //return the final value
137         return (T)Enum.Parse(type, result.ToString());
138     }

```

The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Extensions/[EnumerationExtensions.cs](#)

## 7.5 TriDevs.TriEngine2D.Native.Helpers Class Reference

Helper class with various methods to help native coding and debugging.

### Static Public Member Functions

- static int [GetLastError](#) ()  
*Gets the last error that was thrown.*
- static int [GetLastErrorInfo](#) (out string message)  
*Gets information about the last error that was thrown.*
- static string [GetErrorMessage](#) (int err)  
*Gets the error message associated with an error code.*
- static string [GetLastErrorMessage](#) ()  
*Gets the error message associated with the last thrown error.*
- static Win32Exception [GetWin32Exception](#) (int err)  
*Gets the Win32Exception associated with the specified error code.*
- static Win32Exception [GetLastWin32Exception](#) ()  
*Gets the Win32Exception associated with the last thrown error.*

### 7.5.1 Detailed Description

Helper class with various methods to help native coding and debugging.

### 7.5.2 Member Function Documentation

#### 7.5.2.1 static string TriDevs.TriEngine2D.Native.Helpers.GetErrorMessage ( int err ) [static]

Gets the error message associated with an error code.

#### Parameters

<i>err</i>	The error code to get information about.
------------	--

#### Returns

The error message.

```

61     {
62         return GetWin32Exception(err).Message;
63     }

```

#### 7.5.2.2 static int TriDevs.TriEngine2D.Native.Helpers.GetLastError ( ) [static]

Gets the last error that was thrown.

**Returns**

The error code associated with the thrown error.

```

39      {
40          return Marshal.GetLastWin32Error();
41      }

```

**7.5.2.3 static int TriDevs.TriEngine2D.Native.Helpers.GetLastErrorInfo ( out string *message* ) [static]**

Gets information about the last error that was thrown.

**Parameters**

<i>message</i>	Will be set to the error message.
----------------	-----------------------------------

**Returns**

The error code associated with the thrown error.

```

49      {
50          var err = GetLastError();
51          message = GetWin32Exception(err).Message;
52          return err;
53      }

```

**7.5.2.4 static string TriDevs.TriEngine2D.Native.Helpers.GetLastErrorMessage ( ) [static]**

Gets the error message associated with the last thrown error.

**Returns**

The error message.

```

70      {
71          return GetErrorMessage(GetLastError());
72      }

```

**7.5.2.5 static Win32Exception TriDevs.TriEngine2D.Native.Helpers.GetLastWin32Exception ( ) [static]**

Gets the Win32Exception associated with the last thrown error.

**Returns**

The Win32Exception.

```

89      {
90          return GetWin32Exception(GetLastError());
91      }

```

**7.5.2.6 static Win32Exception TriDevs.TriEngine2D.Native.Helpers.GetWin32Exception ( int *err* ) [static]**

Gets the Win32Exception associated with the specified error code.

**Parameters**

<i>err</i>	The error code.
------------	-----------------

### Returns

The Win32Exception for the provided error code.

```
80     {  
81         return new Win32Exception(err);  
82     }
```

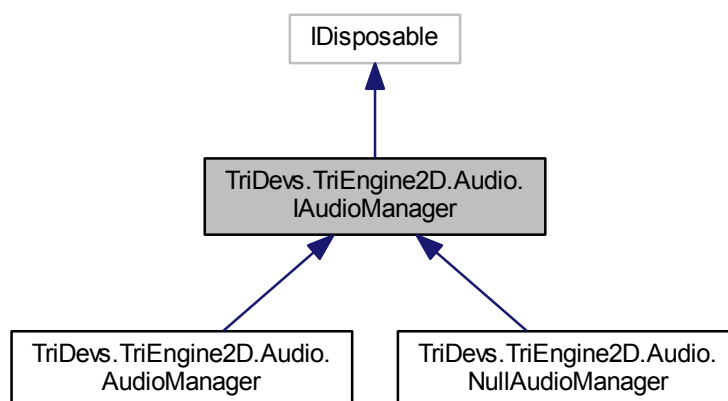
The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Native/[Helpers.cs](#)

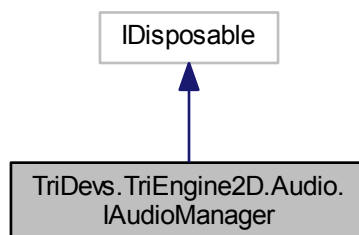
## 7.6 TriDevs.TriEngine2D.Audio.IAudioManager Interface Reference

Provides various methods to manipulate audio.

Inheritance diagram for TriDevs.TriEngine2D.Audio.IAudioManager:



Collaboration diagram for TriDevs.TriEngine2D.Audio.IAudioManager:



## Public Member Functions

- void [StopAll](#) ()  
*Immediately stops all playback of sounds and songs.*
- [ISound LoadSound](#) (string name, string file, [AudioFormat](#) format=AudioFormat.Wav)  
*Loads a sound into the audio manager.*
- bool [HasSound](#) (string name)  
*Returns a value indicating whether a sound with the specified name has been loaded into the audio manager.*
- [ISound GetSound](#) (string name)  
*Gets the sound with the specified name.*
- void [StopAllSounds](#) ()  
*Immediately stops playback of all sounds.*
- [ISong LoadSong](#) (string name, string file, [AudioFormat](#) format=AudioFormat.Ogg)  
*Load a song into the audio manager.*
- bool [HasSong](#) (string name)  
*Returns a value indicating whether a song with the specified name has been loaded into the audio manager.*
- [ISong GetSong](#) (string name)  
*Gets the song with the specified name.*
- void [StopAllSongs](#) ()  
*Immediately stops playback of all songs.*

### 7.6.1 Detailed Description

Provides various methods to manipulate audio.

### 7.6.2 Member Function Documentation

#### 7.6.2.1 [ISong](#) TriDevs.TriEngine2D.Audio.IAudioManager.GetSong ( string name )

Gets the song with the specified name.

##### Parameters

<i>name</i>	Name of the song to get.
-------------	--------------------------

##### Returns

An object implementing the [ISong](#) interface, or null if no song matched the requested name.

Implemented in [TriDevs.TriEngine2D.Audio.AudioManager](#), and [TriDevs.TriEngine2D.Audio.NullAudioManager](#).

#### 7.6.2.2 [ISound](#) TriDevs.TriEngine2D.Audio.IAudioManager.GetSound ( string name )

Gets the sound with the specified name.

##### Parameters

<i>name</i>	Name of the sound to get.
-------------	---------------------------

**Returns**

An object implementing the [ISound](#) interface, or null if no sound matched the requested name.

Implemented in [TriDevs.TriEngine2D.Audio.AudioManager](#), and [TriDevs.TriEngine2D.Audio.NullAudioManager](#).

**7.6.2.3 bool TriDevs.TriEngine2D.Audio.IAudioManager.HasSong ( string name )**

Returns a value indicating whether a song with the specified name has been loaded into the audio manager.

**Parameters**

<i>name</i>	Name to check for.
-------------	--------------------

**Returns**

True if the song has been loaded, false otherwise.

Implemented in [TriDevs.TriEngine2D.Audio.AudioManager](#), and [TriDevs.TriEngine2D.Audio.NullAudioManager](#).

**7.6.2.4 bool TriDevs.TriEngine2D.Audio.IAudioManager.HasSound ( string name )**

Returns a value indicating whether a sound with the specified name has been loaded into the audio manager.

**Parameters**

<i>name</i>	Name to check for.
-------------	--------------------

**Returns**

True if the sound has been loaded, false otherwise.

Implemented in [TriDevs.TriEngine2D.Audio.AudioManager](#), and [TriDevs.TriEngine2D.Audio.NullAudioManager](#).

**7.6.2.5 ISong TriDevs.TriEngine2D.Audio.IAudioManager.LoadSong ( string name, string file, AudioFormat format = AudioFormat.Ogg )**

Load a song into the audio manager.

If a song with the same name has already been loaded, it will be returned instead.

**Parameters**

<i>name</i>	Name to assign to the new song object.
<i>file</i>	Path to the file that will be used.
<i>format</i>	Format of the file.

**Returns**

The resulting object implementing the [ISong](#) interface.

Implemented in [TriDevs.TriEngine2D.Audio.AudioManager](#), and [TriDevs.TriEngine2D.Audio.NullAudioManager](#).

**7.6.2.6 ISound TriDevs.TriEngine2D.Audio.IAudioManager.LoadSound ( string name, string file, AudioFormat format = AudioFormat.Wav )**

Loads a sound into the audio manager.



If a sound with the same name has already been loaded, it will be returned instead.

**Parameters**

<i>name</i>	Name to assign to the new sound object.
<i>file</i>	Path to the file that will be used.
<i>format</i>	The format of the file to load.

**Returns**

The resulting object implementing the [ISound](#) interface.

Implemented in [TriDevs.TriEngine2D.Audio.AudioManager](#), and [TriDevs.TriEngine2D.Audio.NullAudioManager](#).

**7.6.2.7 void TriDevs.TriEngine2D.Audio.IAudioManager.StopAll ( )**

Immediately stops all playback of sounds and songs.

Implemented in [TriDevs.TriEngine2D.Audio.AudioManager](#), and [TriDevs.TriEngine2D.Audio.NullAudioManager](#).

**7.6.2.8 void TriDevs.TriEngine2D.Audio.IAudioManager.StopAllSongs ( )**

Immediately stops playback of all songs.

Implemented in [TriDevs.TriEngine2D.Audio.AudioManager](#), and [TriDevs.TriEngine2D.Audio.NullAudioManager](#).

**7.6.2.9 void TriDevs.TriEngine2D.Audio.IAudioManager.StopAllSounds ( )**

Immediately stops playback of all sounds.

Implemented in [TriDevs.TriEngine2D.Audio.AudioManager](#), and [TriDevs.TriEngine2D.Audio.NullAudioManager](#).

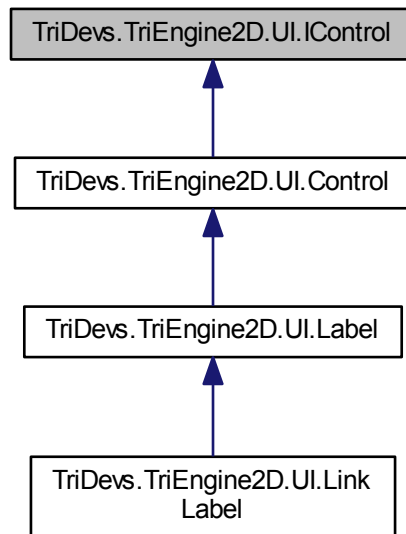
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine2D/Audio/IAudioManager.cs](#)

## 7.7 TriDevs.TriEngine2D.UI.IControl Interface Reference

A [UI](#) control that can be drawn on screen and interacted with.

Inheritance diagram for TriDevs.TriEngine2D.UI.IControl:



## Public Member Functions

- void `Enable` ()  
*Enables the control.*
- void `Disable` ()  
*Disables the control.*
- void `Show` ()  
*Shows the control.*
- void `Hide` ()  
*Hides the control.*

## Properties

- bool `Enabled` [get, set]  
*Gets or sets a value indicating whether this control can be interacted with.*
- bool `Visible` [get, set]  
*Gets or sets a value indicating whether this control should be drawn to the screen.*
- Color `Color` [get, set]  
*The color of this control.*
- Point< int > `Position` [get, set]  
*The position of this control, in pixel coordinates.*
- Point< int > `Size` [get, set]  
*Gets or sets the size of this control, in pixels.*
- string `Text` [get, set]  
*Gets or sets the text associated with this control.*

## Events

- `ControlClickedEventHandler` [Clicked](#)  
*Raised when this control is clicked on by the user.*

### 7.7.1 Detailed Description

A [UI](#) control that can be drawn on screen and interacted with.

### 7.7.2 Member Function Documentation

#### 7.7.2.1 `void TriDevs.TriEngine2D.UI.IControl.Disable ( )`

Disables the control.

Implemented in [TriDevs.TriEngine2D.UI.Control](#).

#### 7.7.2.2 `void TriDevs.TriEngine2D.UI.IControl.Enable ( )`

Enables the control.

Implemented in [TriDevs.TriEngine2D.UI.Control](#).

#### 7.7.2.3 `void TriDevs.TriEngine2D.UI.IControl.Hide ( )`

Hides the control.

Implemented in [TriDevs.TriEngine2D.UI.Control](#).

#### 7.7.2.4 `void TriDevs.TriEngine2D.UI.IControl.Show ( )`

Shows the control.

Implemented in [TriDevs.TriEngine2D.UI.Control](#).

### 7.7.3 Property Documentation

#### 7.7.3.1 `Color TriDevs.TriEngine2D.UI.IControl.Color` `[get]`, `[set]`

The color of this control.

#### 7.7.3.2 `bool TriDevs.TriEngine2D.UI.IControl.Enabled` `[get]`, `[set]`

Gets or sets a value indicating whether this control can be interacted with.

#### 7.7.3.3 `Point<int> TriDevs.TriEngine2D.UI.IControl.Position` `[get]`, `[set]`

The position of this control, in pixel coordinates.

#### 7.7.3.4 `Point<int> TriDevs.TriEngine2D.UI.IControl.Size` `[get]`, `[set]`

Gets or sets the size of this control, in pixels.

**7.7.3.5** `string TriDevs.TriEngine2D.UI.IControl.Text` `[get]`, `[set]`

Gets or sets the text associated with this control.

**7.7.3.6** `bool TriDevs.TriEngine2D.UI.IControl.Visible` `[get]`, `[set]`

Gets or sets a value indicating whether this control should be drawn to the screen.

## 7.7.4 Event Documentation

**7.7.4.1** `ControlClickedEventHandler TriDevs.TriEngine2D.UI.IControl.Clicked`

Raised when this control is clicked on by the user.

The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine2D/UI/IControl.cs](#)

## 7.8 TriDevs.TriEngine2D.Interfaces.IDrawable Interface Reference

Indicates that the implementing class is able to be drawn on the screen.

### Public Member Functions

- `void Draw ()`  
*Draw the object to screen.*

### 7.8.1 Detailed Description

Indicates that the implementing class is able to be drawn on the screen.

### 7.8.2 Member Function Documentation

**7.8.2.1** `void TriDevs.TriEngine2D.Interfaces.IDrawable.Draw ( )`

Draw the object to screen.

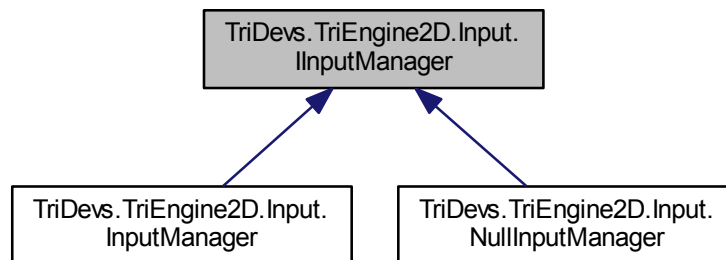
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine2D/Interfaces/IDrawable.cs](#)

## 7.9 TriDevs.TriEngine2D.Input.InputManager Interface Reference

Provides various methods to query input devices like the keyboard.

Inheritance diagram for TriDevs.TriEngine2D.Input.IInputManager:



## Public Member Functions

- void **Update** ()  
*Updates the input manager, refreshing all current and previous states.*
- bool **IsKeyUp** (Key key)  
*Returns whether or not the specified key is currently unpressed.*
- bool **IsKeyDown** (Key key)  
*Returns whether or not the specified key is currently being pressed.*
- bool **KeyPressed** (Key key)  
*Returns whether or not the specified key has been pressed.*
- bool **KeyReleased** (Key key)  
*Returns whether or not the specified key has been released.*
- bool **IsMouseUp** (MouseButton button)  
*Returns whether or not the specified mouse button is currently unpressed.*
- bool **IsMouseDown** (MouseButton button)  
*Returns whether or not the specified mouse button is currently being pressed.*
- bool **MousePressed** (MouseButton button)  
*Returns whether or not the specified mouse button has been pressed.*
- bool **MouseReleased** (MouseButton button)  
*Returns whether or not the specified mouse button has been released.*
- bool **IsWheelUp** ()  
*Returns whether the mouse wheel was scrolled up.*
- bool **IsWheelDown** ()  
*Returns whether the mouse wheel was scrolled down.*
- bool **IsWheelChanged** ()  
*Returns whether the mouse wheel scrolled at all.*
- int **WheelChange** ()  
*Returns the mouse wheel's change in value.*

## Properties

- int **MouseX** [get]  
*Gets the absolute X position of the pointer, in window pixel coordinates.*
- int **MouseY** [get]

- Gets the absolute Y position of the pointer, in window pixel coordinates.*  
 • `Point< int > MousePosition` `[get]`  
*Gets a Point representing the position of the mouse pointer, in window pixel coordinates.*
- `int MouseWheelValue` `[get]`  
*Gets the current value of the mouse wheel.*
- `bool this\[Key key\]` `[get]`  
*Gets a boolean value indicating whether the specified OpenTK.Input.Key is pressed.*
- `bool this\[MouseButton button\]` `[get]`  
*Gets a boolean value indicating whether the specified OpenTK.Input.MouseButton is pressed.*

## Events

- `KeyDownEventHandler KeyDown`  
*Raised when a key is pressed down.*
- `KeyUpEventHandler KeyUp`  
*Raised when a key is released.*
- `KeyPressEventHandler KeyPress`  
*Raised when a character is typed.*
- `MouseDownEventHandler MouseDown`  
*Raised when a mouse button is pressed down.*
- `MouseUpEventHandler MouseUp`  
*Raised when a mouse button is released.*
- `MouseWheelChangedEventHandler WheelChanged`  
*Raised when the mouse wheel value changes.*
- `MouseWheelDownEventHandler WheelDown`  
*Raised when the mouse wheel is scrolled downwards.*
- `MouseWheelUpEventHandler WheelUp`  
*Raised when the mouse wheel is scrolled upwards.*

### 7.9.1 Detailed Description

Provides various methods to query input devices like the keyboard.

### 7.9.2 Member Function Documentation

#### 7.9.2.1 `bool TriDevs.TriEngine2D.Input.IInputManager.IsKeyDown ( Key key )`

Returns whether or not the specified key is currently being pressed.

##### Parameters

<code>key</code>	Key to query for.
------------------	-------------------

##### Returns

True if key is currently being pressed, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

#### 7.9.2.2 `bool TriDevs.TriEngine2D.Input.IInputManager.IsKeyUp ( Key key )`

Returns whether or not the specified key is currently unpressed.

## Parameters

<i>key</i>	Key to query for.
------------	-------------------

## Returns

True if the key is currently up (not pressed), false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

### 7.9.2.3 bool TriDevs.TriEngine2D.Input.IInputManager.IsMouseDown ( MouseButton *button* )

Returns whether or not the specified mouse button is currently being pressed.

## Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

## Returns

True if button is currently being pressed, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

### 7.9.2.4 bool TriDevs.TriEngine2D.Input.IInputManager.IsMouseUp ( MouseButton *button* )

Returns whether or not the specified mouse button is currently unpressed.

## Parameters

<i>button</i>	Button to query for.
---------------	----------------------

## Returns

True if the button is currently up (not pressed), false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

### 7.9.2.5 bool TriDevs.TriEngine2D.Input.IInputManager.IsWheelChanged ( )

Returns whether the mouse wheel scrolled at all.

## Returns

True if the mouse wheel scrolled, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

### 7.9.2.6 bool TriDevs.TriEngine2D.Input.IInputManager.IsWheelDown ( )

Returns whether the mouse wheel was scrolled down.

## Returns

True if mouse wheel was scrolled down, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

#### 7.9.2.7 bool TriDevs.TriEngine2D.Input.IInputManager.IsWheelUp ( )

Returns whether the mouse wheel was scrolled up.

##### Returns

True if mouse wheel was scrolled up, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

#### 7.9.2.8 bool TriDevs.TriEngine2D.Input.IInputManager.KeyPressed ( Key key )

Returns whether or not the specified key has been pressed.

Only returns true if the last state of the key was not pressed.

##### Parameters

<i>key</i>	Key to query for.
------------	-------------------

##### Returns

True if key was pressed, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

#### 7.9.2.9 bool TriDevs.TriEngine2D.Input.IInputManager.KeyReleased ( Key key )

Returns whether or not the specified key has been released.

Only returns true if the last state of the key was pressed.

##### Parameters

<i>key</i>	Key to query for.
------------	-------------------

##### Returns

True if key was released, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

#### 7.9.2.10 bool TriDevs.TriEngine2D.Input.IInputManager.MousePressed ( MouseButton *button* )

Returns whether or not the specified mouse button has been pressed.

Only returns true if the last state of the mouse button was not pressed.

##### Parameters

<i>button</i>	Button to query for.
---------------	----------------------

##### Returns

True if button was pressed, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).



#### 7.9.2.11 `bool TriDevs.TriEngine2D.Input.IInputManager.MouseReleased ( MouseButton button )`

Returns whether or not the specified mouse button has been released.

Only returns true if the last state of the button was pressed.

##### Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

##### Returns

True if the button was released, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

#### 7.9.2.12 `void TriDevs.TriEngine2D.Input.IInputManager.Update ( )`

Updates the input manager, refreshing all current and previous states.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

#### 7.9.2.13 `int TriDevs.TriEngine2D.Input.IInputManager.WheelChange ( )`

Returns the mouse wheel's change in value.

##### Returns

Negative value if wheel scrolled down, positive value if scrolled up, zero if not scrolled.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

### 7.9.3 Property Documentation

#### 7.9.3.1 `Point<int> TriDevs.TriEngine2D.Input.IInputManager.MousePosition` `[get]`

Gets a Point representing the position of the mouse pointer, in window pixel coordinates.

#### 7.9.3.2 `int TriDevs.TriEngine2D.Input.IInputManager.MouseWheelValue` `[get]`

Gets the current value of the mouse wheel.

#### 7.9.3.3 `int TriDevs.TriEngine2D.Input.IInputManager.MouseX` `[get]`

Gets the absolute X position of the pointer, in window pixel coordinates.

#### 7.9.3.4 `int TriDevs.TriEngine2D.Input.IInputManager.MouseY` `[get]`

Gets the absolute Y position of the pointer, in window pixel coordinates.

#### 7.9.3.5 `bool TriDevs.TriEngine2D.Input.IInputManager.this[Key key]` `[get]`

Gets a boolean value indicating whether the specified OpenTK.Input.Key is pressed.

## Parameters

<i>key</i>	The key to query.
------------	-------------------

## Returns

True if pressed, false otherwise.

**7.9.3.6** `bool TriDevs.TriEngine2D.Input.IInputManager.this[MouseButton button] [get]`

Gets a boolean value indicating whether the specified OpenTK.Input.MouseButton is pressed.

## Parameters

<i>button</i>	The button to query.
---------------	----------------------

## Returns

True if pressed, false otherwise.

**7.9.4 Event Documentation****7.9.4.1** `KeyDownEventHandler TriDevs.TriEngine2D.Input.IInputManager.KeyDown`

Raised when a key is pressed down.

**7.9.4.2** `KeyPressEventHandler TriDevs.TriEngine2D.Input.IInputManager.KeyPress`

Raised when a character is typed.

**7.9.4.3** `KeyUpEventHandler TriDevs.TriEngine2D.Input.IInputManager.KeyUp`

Raised when a key is released.

**7.9.4.4** `MouseDownEventHandler TriDevs.TriEngine2D.Input.IInputManager.MouseDown`

Raised when a mouse button is pressed down.

**7.9.4.5** `MouseUpEventHandler TriDevs.TriEngine2D.Input.IInputManager.MouseUp`

Raised when a mouse button is released.

**7.9.4.6** `MouseWheelChangedEventHandler TriDevs.TriEngine2D.Input.IInputManager.WheelChanged`

Raised when the mouse wheel value changes.

**7.9.4.7** `MouseWheelDownEventHandler TriDevs.TriEngine2D.Input.IInputManager.WheelDown`

Raised when the mouse wheel is scrolled downwards.

#### 7.9.4.8 MouseWheelUpEventHandler TriDevs.TriEngine2D.Input.IInputManager.WheelUp

Raised when the mouse wheel is scrolled upwards.

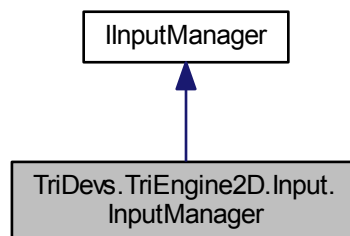
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine2D/Input/IInputManager.cs](#)

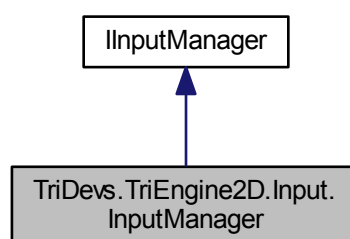
## 7.10 TriDevs.TriEngine2D.Input.InputManager Class Reference

[Input](#) manager interfacing with input methods provided by a [GameWindow](#).

Inheritance diagram for TriDevs.TriEngine2D.Input.InputManager:



Collaboration diagram for TriDevs.TriEngine2D.Input.InputManager:



### Public Member Functions

- [InputManager](#) ()  
*Creates a new [InputManager](#) with only basic low-level input support.*
- [InputManager](#) (GameWindow window)  
*Creates a new [InputManager](#) associated with the specified [GameWindow](#).*
- void [Update](#) ()

- Updates the input manager, refreshing all current and previous states.*

  - bool [IsKeyUp](#) (Key key)  
*Returns whether or not the specified key is currently unpressed.*
  - bool [IsKeyDown](#) (Key key)  
*Returns whether or not the specified key is currently being pressed.*
  - bool [KeyPressed](#) (Key key)  
*Returns whether or not the specified key has been pressed.*
  - bool [KeyReleased](#) (Key key)  
*Returns whether or not the specified key has been released.*
  - bool [IsMouseUp](#) (MouseButton button)  
*Returns whether or not the specified mouse button is currently unpressed.*
  - bool [IsMouseDown](#) (MouseButton button)  
*Returns whether or not the specified mouse button is currently being pressed.*
  - bool [MousePressed](#) (MouseButton button)  
*Returns whether or not the specified mouse button has been pressed.*
  - bool [MouseReleased](#) (MouseButton button)  
*Returns whether or not the specified mouse button has been released.*
  - bool [IsWheelUp](#) ()  
*Returns whether the mouse wheel was scrolled up.*
  - bool [IsWheelDown](#) ()  
*Returns whether the mouse wheel was scrolled down.*
  - bool [IsWheelChanged](#) ()  
*Returns whether the mouse wheel scrolled at all.*
  - int [WheelChange](#) ()  
*Returns the mouse wheel's change in value.*

## Properties

- int [MouseX](#) [get]
- int [MouseY](#) [get]
- Point< int > [MousePosition](#) [get]
- int [MouseWheelValue](#) [get]
- bool [this\[Key key\]](#) [get]
- bool [this\[MouseButton button\]](#) [get]

## Events

- KeyDownEventHandler [KeyDown](#)  
*Raised when a key is pressed down.*
- KeyUpEventHandler [KeyUp](#)  
*Raised when a key is released.*
- KeyPressEventHandler [KeyPress](#)  
*Raised when a character is typed.*
- MouseDownEventHandler [MouseDown](#)  
*Raised when a mouse button is pressed down.*
- MouseUpEventHandler [MouseUp](#)  
*Raised when a mouse button is released.*
- MouseWheelChangedEventHandler [WheelChanged](#)  
*Raised when the mouse wheel value changes.*
- MouseWheelDownEventHandler [WheelDown](#)  
*Raised when the mouse wheel is scrolled downwards.*
- MouseWheelUpEventHandler [WheelUp](#)  
*Raised when the mouse wheel is scrolled upwards.*

## 7.10.1 Detailed Description

[Input](#) manager interfacing with input methods provided by a [GameWindow](#).

## 7.10.2 Constructor & Destructor Documentation

### 7.10.2.1 TriDevs.TriEngine2D.Input.InputManager.InputManager ( )

Creates a new [InputManager](#) with only basic low-level input support.

Creating [InputManager](#) without a driver container will cause the events to be useless and never be raised, only the methods on this class will return any useful info. If you want event support, construct the [InputManager](#) with a [GameWindow](#) or other supported driver providers (NYI).

```

109     {
110         // We're assigning an empty mouse device.
111         // This will make position functions return a constant 0.
112         // Instead of being null and causing exceptions.
113         _mouse = new MouseDevice();
114         // We don't have to assign an empty keyboard device,
115         // since we don't have any code that directly relies on it being present.
116     }
```

### 7.10.2.2 TriDevs.TriEngine2D.Input.InputManager.InputManager ( GameWindow window )

Creates a new [InputManager](#) associated with the specified [GameWindow](#).

#### Parameters

<i>window</i>	The <a href="#">GameWindow</a> this <a href="#">InputManager</a> will interface with.
---------------	---

```

123     {
124         _keyboard = window.Keyboard;
125         _mouse = window.Mouse;
126         _keyboard.KeyDown += OnKeyDown;
127         _keyboard.KeyUp += OnKeyUp;
128         window.KeyPress += OnKeyPress;
129         _mouse.ButtonDown += OnMouseDown;
130         _mouse.ButtonUp += OnMouseUp;
131         _mouse.WheelChanged += OnMouseWheelChanged;
132     }
```

## 7.10.3 Member Function Documentation

### 7.10.3.1 bool TriDevs.TriEngine2D.Input.InputManager.IsKeyDown ( Key key )

Returns whether or not the specified key is currently being pressed.

#### Parameters

<i>key</i>	Key to query for.
------------	-------------------

#### Returns

True if key is currently being pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

201     {
202         return _keyboardState[key];
203     }
```

### 7.10.3.2 bool TriDevs.TriEngine2D.Input.InputManager.IsKeyUp ( Key *key* )

Returns whether or not the specified key is currently unpressed.

#### Parameters

<i>key</i>	Key to query for.
------------	-------------------

#### Returns

True if the key is currently up (not pressed), false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

196         {
197             return !_keyboardState[key];
198         }
```

### 7.10.3.3 bool TriDevs.TriEngine2D.Input.InputManager.IsMouseDown ( MouseButton *button* )

Returns whether or not the specified mouse button is currently being pressed.

#### Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

#### Returns

True if button is currently being pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

221         {
222             return _mouseState[button];
223         }
```

### 7.10.3.4 bool TriDevs.TriEngine2D.Input.InputManager.IsMouseUp ( MouseButton *button* )

Returns whether or not the specified mouse button is currently unpressed.

#### Parameters

<i>button</i>	Button to query for.
---------------	----------------------

#### Returns

True if the button is currently up (not pressed), false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

216         {
217             return !_mouseState[button];
218         }
```

### 7.10.3.5 bool TriDevs.TriEngine2D.Input.InputManager.IsWheelChanged ( )

Returns whether the mouse wheel scrolled at all.

**Returns**

True if the mouse wheel scrolled, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

246         {
247             return _mouseState.Wheel != _lastMouseState.Wheel;
248         }

```

**7.10.3.6 bool TriDevs.TriEngine2D.Input.InputManager.IsWheelDown ( )**

Returns whether the mouse wheel was scrolled down.

**Returns**

True if mouse wheel was scrolled down, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

241         {
242             return _mouseState.Wheel < _lastMouseState.Wheel;
243         }

```

**7.10.3.7 bool TriDevs.TriEngine2D.Input.InputManager.IsWheelUp ( )**

Returns whether the mouse wheel was scrolled up.

**Returns**

True if mouse wheel was scrolled up, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

236         {
237             return _mouseState.Wheel > _lastMouseState.Wheel;
238         }

```

**7.10.3.8 bool TriDevs.TriEngine2D.Input.InputManager.KeyPressed ( Key key )**

Returns whether or not the specified key has been pressed.

Only returns true if the last state of the key was not pressed.

**Parameters**

<i>key</i>	Key to query for.
------------	-------------------

**Returns**

True if key was pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

206         {
207             return _keyboardState[key] && !_lastKeyboardState[key];
208         }

```

#### 7.10.3.9 bool TriDevs.TriEngine2D.Input.InputManager.KeyReleased ( Key *key* )

Returns whether or not the specified key has been released.

Only returns true if the last state of the key was pressed.

##### Parameters

<i>key</i>	Key to query for.
------------	-------------------

##### Returns

True if key was released, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

211     {
212         return !_keyboardState[key] && _lastKeyboardState[key];
213     }
```

#### 7.10.3.10 bool TriDevs.TriEngine2D.Input.InputManager.MousePressed ( MouseButton *button* )

Returns whether or not the specified mouse button has been pressed.

Only returns true if the last state of the mouse button was not pressed.

##### Parameters

<i>button</i>	Button to query for.
---------------	----------------------

##### Returns

True if button was pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

226     {
227         return _mouseState[button] && !_lastMouseState[button];
228     }
```

#### 7.10.3.11 bool TriDevs.TriEngine2D.Input.InputManager.MouseReleased ( MouseButton *button* )

Returns whether or not the specified mouse button has been released.

Only returns true if the last state of the button was pressed.

##### Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

##### Returns

True if the button was released, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

231     {
232         return !_mouseState[button] && _lastMouseState[button];
233     }
```



**7.10.3.12 void TriDevs.TriEngine2D.Input.InputManager.Update ( )**

Updates the input manager, refreshing all current and previous states.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

187     {
188         _lastKeyboardState = _keyboardState;
189         _keyboardState = Keyboard.GetState();
190
191         _lastMouseState = _mouseState;
192         _mouseState = Mouse.GetState();
193     }

```

**7.10.3.13 int TriDevs.TriEngine2D.Input.InputManager.WheelChange ( )**

Returns the mouse wheel's change in value.

**Returns**

Negative value if wheel scrolled down, positive value if scrolled up, zero if not scrolled.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

251     {
252         return _mouseState.Wheel - _lastMouseState.Wheel;
253     }

```

**7.10.4 Property Documentation**

**7.10.4.1 Point<int> TriDevs.TriEngine2D.Input.InputManager.MousePosition** [get]

**7.10.4.2 int TriDevs.TriEngine2D.Input.InputManager.MouseWheelValue** [get]

**7.10.4.3 int TriDevs.TriEngine2D.Input.InputManager.MouseX** [get]

**7.10.4.4 int TriDevs.TriEngine2D.Input.InputManager.MouseY** [get]

**7.10.4.5 bool TriDevs.TriEngine2D.Input.InputManager.this[Key key]** [get]

**7.10.4.6 bool TriDevs.TriEngine2D.Input.InputManager.this[MouseButton button]** [get]

**7.10.5 Event Documentation**

**7.10.5.1 KeyDownEventHandler TriDevs.TriEngine2D.Input.InputManager.KeyDown**

Raised when a key is pressed down.

**7.10.5.2 KeyPressEventHandler TriDevs.TriEngine2D.Input.InputManager.KeyPress**

Raised when a character is typed.

**7.10.5.3 KeyUpEventHandler TriDevs.TriEngine2D.Input.InputManager.KeyUp**

Raised when a key is released.

#### 7.10.5.4 `MouseDownEventHandler` `TriDevs.TriEngine2D.Input.InputManager.MouseDown`

Raised when a mouse button is pressed down.

#### 7.10.5.5 `MouseUpEventHandler` `TriDevs.TriEngine2D.Input.InputManager.MouseUp`

Raised when a mouse button is released.

#### 7.10.5.6 `MouseWheelChangedEventHandler` `TriDevs.TriEngine2D.Input.InputManager.WheelChanged`

Raised when the mouse wheel value changes.

#### 7.10.5.7 `MouseWheelDownEventHandler` `TriDevs.TriEngine2D.Input.InputManager.WheelDown`

Raised when the mouse wheel is scrolled downwards.

#### 7.10.5.8 `MouseWheelUpEventHandler` `TriDevs.TriEngine2D.Input.InputManager.WheelUp`

Raised when the mouse wheel is scrolled upwards.

The documentation for this class was generated from the following file:

- `TriDevs.TriEngine2D/Input/InputManager.cs`

## 7.11 `TriDevs.TriEngine2D.Helpers.IO` Class Reference

Provides various helper functions for doing [IO](#) operations.

### Static Public Member Functions

- static string [GetAbsolutePath](#) (string path)  
*Resolves the absolute path from a relative path.*

#### 7.11.1 Detailed Description

Provides various helper functions for doing [IO](#) operations.

#### 7.11.2 Member Function Documentation

##### 7.11.2.1 static string `TriDevs.TriEngine2D.Helpers.IO.GetAbsolutePath ( string path )` `[static]`

Resolves the absolute path from a relative path.

#### Parameters

<i>path</i>	The relative path to resolve.
-------------	-------------------------------

### Returns

The absolute path to the item.

```
39         {  
40             return Path.Combine(Directory.GetCurrentDirectory(), path);  
41         }
```

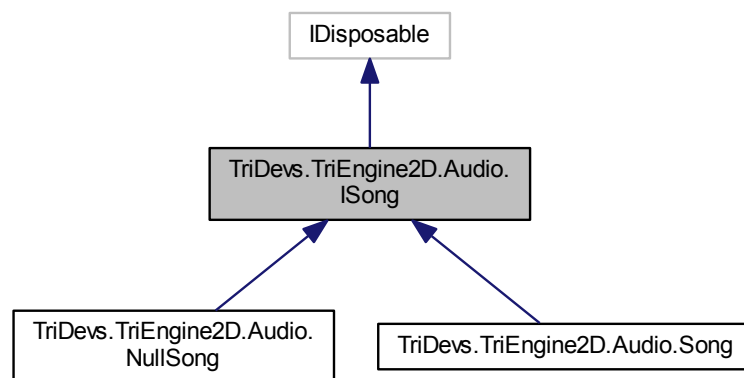
The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Helpers/[IO.cs](#)

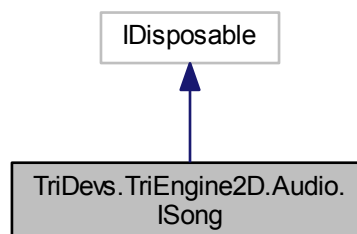
## 7.12 TriDevs.TriEngine2D.Audio.ISong Interface Reference

A song that will be streamed in the audio player.

Inheritance diagram for TriDevs.TriEngine2D.Audio.ISong:



Collaboration diagram for TriDevs.TriEngine2D.Audio.ISong:



### Public Member Functions

- void [Play](#) ()

- Starts playback of the song.*
  - void [Stop](#) ()
- Stops playback of the song.*
  - void [Pause](#) ()
- Pauses playback of the song.*
  - void [Resume](#) ()
- Resumes playback of a paused song.*

## Properties

- string [Name](#) [get]
  - Unique name of this song object.*
- string [File](#) [get]
  - Path to the file associated with this song object.*
- float [Volume](#) [get, set]
  - Gets or sets the song volume.*
- bool [IsLooped](#) [get, set]
  - Gets or sets a value indicating that the song should be looped once it reaches the end.*

### 7.12.1 Detailed Description

A song that will be streamed in the audio player.

### 7.12.2 Member Function Documentation

#### 7.12.2.1 void [TriDevs.TriEngine2D.Audio.ISong.Pause](#) ( )

Pauses playback of the song.

Call the [Resume](#) or [Play](#) method to resume playback.

Implemented in [TriDevs.TriEngine2D.Audio.Song](#), and [TriDevs.TriEngine2D.Audio.NullSong](#).

#### 7.12.2.2 void [TriDevs.TriEngine2D.Audio.ISong.Play](#) ( )

Starts playback of the song.

Implemented in [TriDevs.TriEngine2D.Audio.Song](#), and [TriDevs.TriEngine2D.Audio.NullSong](#).

#### 7.12.2.3 void [TriDevs.TriEngine2D.Audio.ISong.Resume](#) ( )

Resumes playback of a paused song.

Implemented in [TriDevs.TriEngine2D.Audio.Song](#), and [TriDevs.TriEngine2D.Audio.NullSong](#).

#### 7.12.2.4 void [TriDevs.TriEngine2D.Audio.ISong.Stop](#) ( )

Stops playback of the song.

Implemented in [TriDevs.TriEngine2D.Audio.Song](#), and [TriDevs.TriEngine2D.Audio.NullSong](#).

### 7.12.3 Property Documentation

#### 7.12.3.1 `string TriDevs.TriEngine2D.Audio.ISong.File` `[get]`

Path to the file associated with this song object.

#### 7.12.3.2 `bool TriDevs.TriEngine2D.Audio.ISong.IsLooped` `[get]`, `[set]`

Gets or sets a value indicating that the song should be looped once it reaches the end.

#### 7.12.3.3 `string TriDevs.TriEngine2D.Audio.ISong.Name` `[get]`

Unique name of this song object.

#### 7.12.3.4 `float TriDevs.TriEngine2D.Audio.ISong.Volume` `[get]`, `[set]`

Gets or sets the song volume.

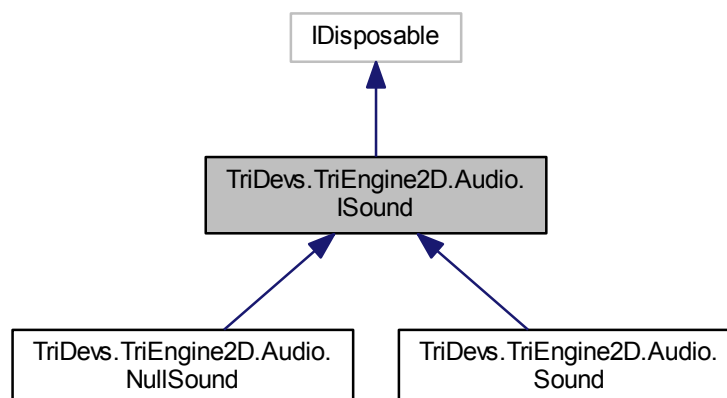
The documentation for this interface was generated from the following file:

- `TriDevs.TriEngine2D/Audio/ISong.cs`

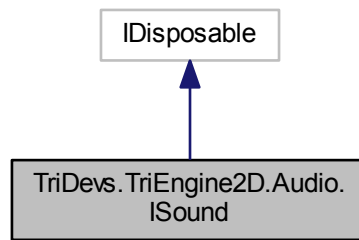
## 7.13 TriDevs.TriEngine2D.Audio.ISound Interface Reference

A sound file for use with the audio manager.

Inheritance diagram for `TriDevs.TriEngine2D.Audio.ISound`:



Collaboration diagram for TriDevs.TriEngine2D.Audio.ISound:



## Public Member Functions

- void [Play](#) ()  
*Plays the sound.*
- void [Stop](#) ()  
*Stops the sound.*

## Properties

- string [Name](#) [get]  
*Unique name of this sound object.*
- string [File](#) [get]  
*Path to the file this sound object is associated with.*

### 7.13.1 Detailed Description

A sound file for use with the audio manager.

### 7.13.2 Member Function Documentation

#### 7.13.2.1 void TriDevs.TriEngine2D.Audio.ISound.Play ( )

Plays the sound.

Implemented in [TriDevs.TriEngine2D.Audio.Sound](#), and [TriDevs.TriEngine2D.Audio.NullSound](#).

#### 7.13.2.2 void TriDevs.TriEngine2D.Audio.ISound.Stop ( )

Stops the sound.

Implemented in [TriDevs.TriEngine2D.Audio.Sound](#), and [TriDevs.TriEngine2D.Audio.NullSound](#).

### 7.13.3 Property Documentation

#### 7.13.3.1 `string TriDevs.TriEngine2D.Audio.ISound.File` [get]

Path to the file this sound object is associated with.

#### 7.13.3.2 `string TriDevs.TriEngine2D.Audio.ISound.Name` [get]

Unique name of this sound object.

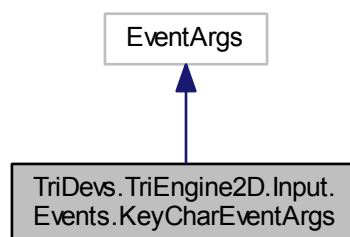
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine2D/Audio/ISound.cs](#)

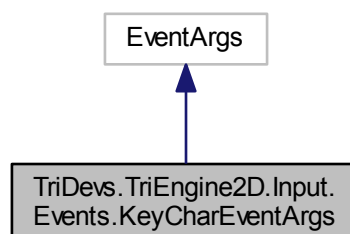
## 7.14 TriDevs.TriEngine2D.Input.Events.KeyCharEventArgs Class Reference

EventArgs class used for keychar-related events. Contains information about the character related with the event.

Inheritance diagram for TriDevs.TriEngine2D.Input.Events.KeyCharEventArgs:



Collaboration diagram for TriDevs.TriEngine2D.Input.Events.KeyCharEventArgs:



## Public Attributes

- readonly char [KeyChar](#)

*The char that was involved.*

### 7.14.1 Detailed Description

EventArgs class used for keychar-related events. Contains information about the character related with the event.

### 7.14.2 Member Data Documentation

#### 7.14.2.1 readonly char TriDevs.TriEngine2D.Input.Events.KeyCharEventArgs.KeyChar

The char that was involved.

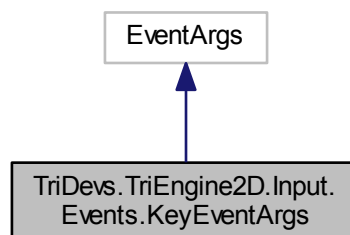
The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Input/Events/[Key.cs](#)

## 7.15 TriDevs.TriEngine2D.Input.Events.KeyEventArgs Class Reference

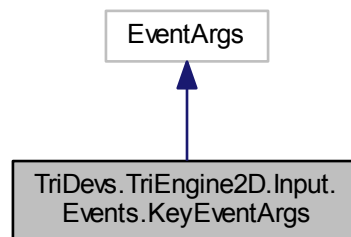
EventArgs class used for key-related events. Contains information about the key related with the event.

Inheritance diagram for TriDevs.TriEngine2D.Input.Events.KeyEventArgs:





Collaboration diagram for TriDevs.TriEngine2D.Input.Events.KeyEventArgs:



## Public Attributes

- readonly Key [Key](#)

*The [Key](#) that was involved.*

## 7.15.1 Detailed Description

EventArgs class used for key-related events. Contains information about the key related with the event.

## 7.15.2 Member Data Documentation

### 7.15.2.1 readonly Key TriDevs.TriEngine2D.Input.Events.KeyEventArgs.Key

The [Key](#) that was involved.

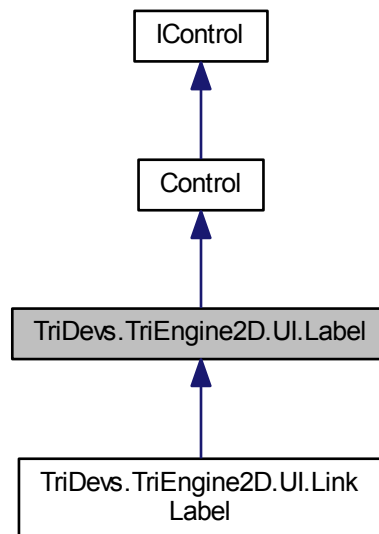
The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Input/Events/[Key.cs](#)

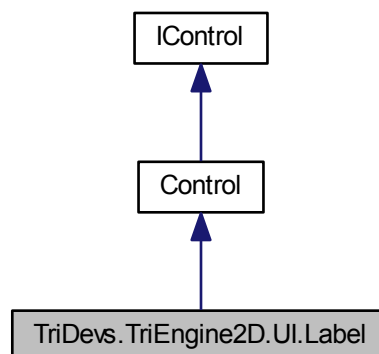
## 7.16 TriDevs.TriEngine2D.UI.Label Class Reference

A simple label to display text on the screen.

Inheritance diagram for TriDevs.TriEngine2D.UI.Label:



Collaboration diagram for TriDevs.TriEngine2D.UI.Label:



## Properties

- override string [Text](#) [get, set]

## Additional Inherited Members

### 7.16.1 Detailed Description

A simple label to display text on the screen.

### 7.16.2 Property Documentation

#### 7.16.2.1 `override string TriDevs.TriEngine2D.UI.Label.Text` `[get]`, `[set]`

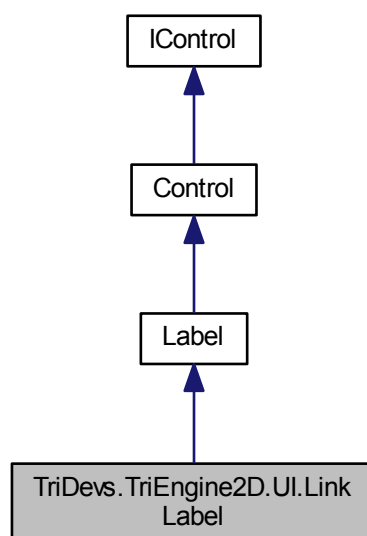
The documentation for this class was generated from the following file:

- `TriDevs.TriEngine2D/UI/Label.cs`

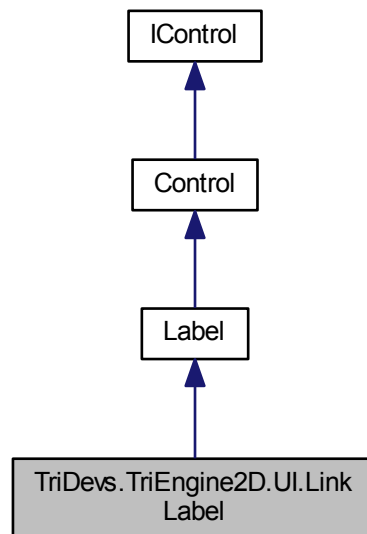
## 7.17 TriDevs.TriEngine2D.UI.LinkLabel Class Reference

A label that, when clicked, will open a URL.

Inheritance diagram for `TriDevs.TriEngine2D.UI.LinkLabel`:



Collaboration diagram for TriDevs.TriEngine2D.UI.LinkLabel:



### Protected Member Functions

- override void [OnClicked](#) ()

### Properties

- string [Url](#) [get, set]  
*Gets or sets the URL that will open when this label is clicked.*

### Additional Inherited Members

#### 7.17.1 Detailed Description

A label that, when clicked, will open a URL.

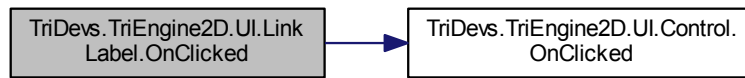
#### 7.17.2 Member Function Documentation

##### 7.17.2.1 override void `TriDevs.TriEngine2D.UI.LinkLabel.OnClicked ( )` [protected], [virtual]

Reimplemented from [TriDevs.TriEngine2D.UI.Control](#).

```
47     {  
48         System.Diagnostics.Process.Start(_url);  
49         base.OnClicked();  
50     }
```

Here is the call graph for this function:



### 7.17.3 Property Documentation

#### 7.17.3.1 string TriDevs.TriEngine2D.UI.LinkLabel.Url [get], [set]

Gets or sets the URL that will open when this label is clicked.

The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/UI/[LinkLabel.cs](#)

## 7.18 TriDevs.TriEngine2D.Logging.LogManager Class Reference

Class to manage logging. ILog interfaces should be obtained from this class' methods, as opposed to calling default log4net methods.

### Static Public Member Functions

- static void [LoadConfig](#) (string file=null)  
*Load a config to use with log4net.*
- static ILog [GetLogger](#) (object sender)  
*Gets an ILog object for the specified object.*
- static void [SetupConsole](#) ()  
*Set up a new console for this process. Will not set up a console if a debugger is attached. This method does nothing if DEBUG is not #defined.*
- static void [DestroyConsole](#) ()  
*Destroys the console associated with the process, if loaded. This method does nothing if DEBUG is not #defined.*
- static void [ClearOldLogs](#) (int daysOld=7, string logsDir="logs")  
*Clear logs that are older than the specified amount of days.*

### 7.18.1 Detailed Description

Class to manage logging. ILog interfaces should be obtained from this class' methods, as opposed to calling default log4net methods.

### 7.18.2 Member Function Documentation

#### 7.18.2.1 static void TriDevs.TriEngine2D.Logging.LogManager.ClearOldLogs ( int daysOld = 7, string logsDir = "logs" ) [static]

Clear logs that are older than the specified amount of days.

## Parameters

<i>daysOld</i>	Logs older than this amount of days will be deleted.
<i>logsDir</i>	The directory to clear.

```

136     {
137         var log = GetLogger(typeof(LogManager));
138
139         log.InfoFormat(">> ClearOldLogs({0}, \"{1}\")", daysOld, logsDir);
140
141         if (!Directory.Exists(logsDir))
142         {
143             log.InfoFormat("Directory {0} not found, no logs to clear", logsDir);
144             log.Info("<< ClearOldLogs()");
145             return;
146         }
147
148         var now = DateTime.Now;
149         var max = new TimeSpan(daysOld, 0, 0, 0);
150         var count = 0;
151         foreach (var file in from file in Directory.GetFiles(logsDir)
152                             let modTime = File.GetLastAccessTime(file)
153                             let age = now.Subtract(modTime)
154                             where age > max
155                             select file)
156         {
157             try
158             {
159                 File.Delete(file);
160                 log.InfoFormat("Deleted old log file: {0}", file);
161                 count++;
162             }
163             catch (IOException ex)
164             {
165                 log.WarnFormat("Failed to delete log file: {0} ({1})", file, ex.Message);
166             }
167         }
168
169         log.InfoFormat("Done! Cleared {0} log files.", count);
170         log.Info("<< ClearOldLogs()");
171     }

```

### 7.18.2.2 static void TriDevs.TriEngine2D.Logging.LogManager.DestroyConsole ( ) [static]

Destroys the console associated with the process, if loaded. This method does nothing if DEBUG is not #defined.

```

123     {
124     #if DEBUG
125         if (_consoleLoaded)
126             WinAPI.FreeConsole();
127     #endif
128     }

```

Here is the call graph for this function:



### 7.18.2.3 static ILog TriDevs.TriEngine2D.Logging.LogManager.GetLogger ( object sender ) [static]

Gets an ILog object for the specified object.

To get the logger object for a static class, or from static context, call GetLogger(typeof(YourClass)).

## Parameters

<i>sender</i>	The object or Type to get an ILog object for.
---------------	---

## Returns

The ILog object.

```

89         {
90             if (!_loaded)
91                 LoadConfig();
92
93             return log4net.LogManager.GetLogger(sender.GetType().ToString() == "System.RuntimeType" ? (Type
)sender : sender.GetType());
94         }

```

#### 7.18.2.4 static void TriDevs.TriEngine2D.Logging.LogManager.LoadConfig ( string file = null ) [static]

Load a config to use with log4net.

LoadConfig will first try to load the specified file, if not null. If it is unable to find the specified file, it will call itself again with file set to null. If no file is specified, it will attempt to load a config file following the pattern: "(Assembly-Name).config" If it is unable to load the config, it will default to BasicConfigurator.

## Parameters

<i>file</i>	The config file to load, null if automatic loading is preferred.
-------------	--

```

57         {
58             if (file == null)
59             {
60                 if (File.Exists(AppDomain.CurrentDomain.FriendlyName + ".config"))
61                     XmlConfigurator.Configure();
62                 else
63                     BasicConfigurator.Configure();
64             }
65             else
66             {
67                 if (File.Exists(file))
68                     XmlConfigurator.Configure(new FileInfo(file));
69                 else
70                 {
71                     LoadConfig();
72                     return;
73                 }
74             }
75
76             _loaded = true;
77         }

```

#### 7.18.2.5 static void TriDevs.TriEngine2D.Logging.LogManager.SetupConsole ( ) [static]

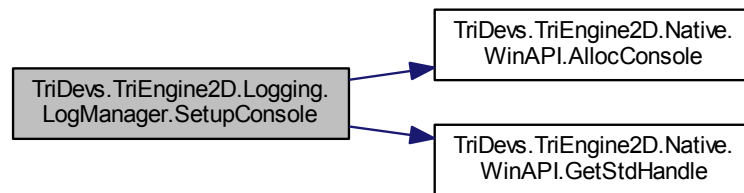
Set up a new console for this process. Will not set up a console if a debugger is attached. This method does nothing if DEBUG is not #defined.

```

102         {
103             #if DEBUG
104                 if (System.Diagnostics.Debugger.IsAttached)
105                     return;
106
107                 WinAPI.AllocConsole();
108                 var stdHandle = WinAPI.GetStdHandle(WinAPI.
STD_OUTPUT_HANDLE);
109                 var safeFileHandle = new SafeFileHandle(stdHandle, true);
110                 var fileStream = new FileStream(safeFileHandle, FileAccess.Write);
111                 var encoding = Encoding.GetEncoding(WinAPI.CODE_PAGE);
112                 var stdout = new StreamWriter(fileStream, encoding) { AutoFlush = true };
113                 Console.SetOut(stdout);
114                 _consoleLoaded = true;
115             #endif
116         }

```

Here is the call graph for this function:



The documentation for this class was generated from the following file:

- `TriDevs.TriEngine2D/Logging/LogManager.cs`

## 7.19 TriDevs.TriEngine2D.Helpers.Math Class Reference

Various helper methods for working with math.

### Static Public Member Functions

- static byte [Clamp](#) (byte value, byte min, byte max)  
*Clamps the specified byte between a minimum and maximum value.*
- static short [Clamp](#) (short value, short min, short max)  
*Clamps the specified short between a minimum and maximum value.*
- static ushort [Clamp](#) (ushort value, ushort min, ushort max)  
*Clamps the specified unsigned short between a minimum and maximum value.*
- static int [Clamp](#) (int value, int min, int max)  
*Clamps the specified integer between a minimum and maximum value.*
- static uint [Clamp](#) (uint value, uint min, uint max)  
*Clamps the specified unsigned integer between a minimum and maximum value.*
- static long [Clamp](#) (long value, long min, long max)  
*Clamps the specified 64-bit integer between a minimum and maximum value.*
- static ulong [Clamp](#) (ulong value, ulong min, ulong max)  
*Clamps the specified 64-bit unsigned integer between a minimum and maximum value.*
- static float [Clamp](#) (float value, float min, float max)  
*Clamps the specified float between a minimum and maximum value.*
- static double [Clamp](#) (double value, double min, double max)  
*Clamps the specified double between a minimum and maximum value.*

### 7.19.1 Detailed Description

Various helper methods for working with math.



## 7.19.2 Member Function Documentation

### 7.19.2.1 static byte TriDevs.TriEngine2D.Helpers.Math.Clamp ( byte *value*, byte *min*, byte *max* ) [static]

Clamps the specified byte between a minimum and maximum value.

#### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified byte is below this value, then this will be returned.
<i>max</i>	If the specified byte is above this value, then this will be returned.

#### Returns

The clamped value of the byte.

```
41         {  
42             if (min > max)  
43                 throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");  
44  
45             return value < min ? min : (value > max ? max : value);  
46         }
```

### 7.19.2.2 static short TriDevs.TriEngine2D.Helpers.Math.Clamp ( short *value*, short *min*, short *max* ) [static]

Clamps the specified short between a minimum and maximum value.

#### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified short is below this value, then this will be returned.
<i>max</i>	If the specified short is above this value, then this will be returned.

#### Returns

The clamped value of the short.

```
56         {  
57             if (min > max)  
58                 throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");  
59  
60             return value < min ? min : (value > max ? max : value);  
61         }
```

### 7.19.2.3 static ushort TriDevs.TriEngine2D.Helpers.Math.Clamp ( ushort *value*, ushort *min*, ushort *max* ) [static]

Clamps the specified unsigned short between a minimum and maximum value.

#### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified unsigned short is below this value, then this will be returned.
<i>max</i>	If the specified unsigned short is above this value, then this will be returned.

#### Returns

The clamped value of the unsigned short.

```
71         {
```

```

72         if (min > max)
73             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
74
75         return value < min ? min : (value > max ? max : value);
76     }

```

#### 7.19.2.4 static int TriDevs.TriEngine2D.Helpers.Math.Clamp ( int value, int min, int max ) [static]

Clamps the specified integer between a minimum and maximum value.

##### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified integer is below this value, then this will be returned.
<i>max</i>	If the specified integer is above this value, then this will be returned.

##### Returns

The clamped value of the integer.

```

86     {
87         if (min > max)
88             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
89
90         return value < min ? min : (value > max ? max : value);
91     }

```

#### 7.19.2.5 static uint TriDevs.TriEngine2D.Helpers.Math.Clamp ( uint value, uint min, uint max ) [static]

Clamps the specified unsigned integer between a minimum and maximum value.

##### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified unsigned integer is below this value, then this will be returned.
<i>max</i>	If the specified unsigned integer is above this value, then this will be returned.

##### Returns

The clamped value of the unsigned integer.

```

101     {
102         if (min > max)
103             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
104
105         return value < min ? min : (value > max ? max : value);
106     }

```

#### 7.19.2.6 static long TriDevs.TriEngine2D.Helpers.Math.Clamp ( long value, long min, long max ) [static]

Clamps the specified 64-bit integer between a minimum and maximum value.

##### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified 64-bit integer is below this value, then this will be returned.
<i>max</i>	If the specified 64-bit integer is above this value, then this will be returned.

**Returns**

The clamped value of the 64-bit integer.

```

116     {
117         if (min > max)
118             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
119
120         return value < min ? min : (value > max ? max : value);
121     }

```

**7.19.2.7 static ulong TriDevs.TriEngine2D.Helpers.Math.Clamp ( ulong *value*, ulong *min*, ulong *max* ) [static]**

Clamps the specified 64-bit unsigned integer between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified 64-bit unsigned integer is below this value, then this will be returned.
<i>max</i>	If the specified 64-bit unsigned integer is above this value, then this will be returned.

**Returns**

The clamped value of the 64-bit unsigned integer.

```

131     {
132         if (min > max)
133             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
134
135         return value < min ? min : (value > max ? max : value);
136     }

```

**7.19.2.8 static float TriDevs.TriEngine2D.Helpers.Math.Clamp ( float *value*, float *min*, float *max* ) [static]**

Clamps the specified float between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified float is below this value, then this will be returned.
<i>max</i>	If the specified float is above this value, then this will be returned.

**Returns**

The clamped value of the float.

```

146     {
147         if (min > max)
148             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
149
150         return value < min ? min : (value > max ? max : value);
151     }

```

**7.19.2.9 static double TriDevs.TriEngine2D.Helpers.Math.Clamp ( double *value*, double *min*, double *max* ) [static]**

Clamps the specified double between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified double is below this value, then this will be returned.
<i>max</i>	If the specified double is above this value, then this will be returned.

### Returns

The clamped value of the double.

```
161     {
162         if (min > max)
163             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
164         return value < min ? min : (value > max ? max : value);
165     }
166 }
```

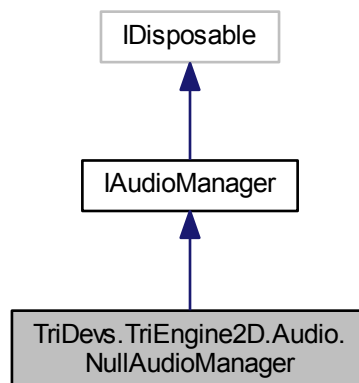
The documentation for this class was generated from the following file:

- [TriDevs.TriEngine2D/Helpers/Math.cs](#)

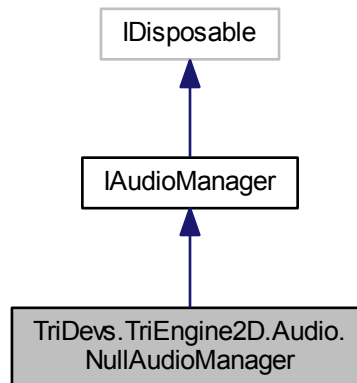
## 7.20 TriDevs.TriEngine2D.Audio.NullAudioManager Class Reference

Used as a fallback [AudioManager](#) object when the service locator fails to find one.

Inheritance diagram for TriDevs.TriEngine2D.Audio.NullAudioManager:



Collaboration diagram for TriDevs.TriEngine2D.Audio.NullAudioManager:



## Public Member Functions

- void [Dispose](#) ()
- void [StopAll](#) ()  
*Immediately stops all playback of sounds and songs.*
- [ISound LoadSound](#) (string name, string file, [AudioFormat](#) format=AudioFormat.Wav)  
*Loads a sound into the audio manager.*
- bool [HasSound](#) (string name)  
*Returns a value indicating whether a sound with the specified name has been loaded into the audio manager.*
- [ISound GetSound](#) (string name)  
*Gets the sound with the specified name.*
- void [StopAllSounds](#) ()  
*Immediately stops playback of all sounds.*
- [ISong LoadSong](#) (string name, string file, [AudioFormat](#) format=AudioFormat.Ogg)  
*Load a song into the audio manager.*
- bool [HasSong](#) (string name)  
*Returns a value indicating whether a song with the specified name has been loaded into the audio manager.*
- [ISong GetSong](#) (string name)  
*Gets the song with the specified name.*
- void [StopAllSongs](#) ()  
*Immediately stops playback of all songs.*

### 7.20.1 Detailed Description

Used as a fallback [AudioManager](#) object when the service locator fails to find one.

## 7.20.2 Member Function Documentation

### 7.20.2.1 void TriDevs.TriEngine2D.Audio.NullAudioManager.Dispose ( )

```
35         {  
36  
37         }
```

### 7.20.2.2 ISong TriDevs.TriEngine2D.Audio.NullAudioManager.GetSong ( string *name* )

Gets the song with the specified name.

#### Parameters

<i>name</i>	Name of the song to get.
-------------	--------------------------

#### Returns

An object implementing the [ISong](#) interface, or null if no song matched the requested name.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```
75         {  
76             return Song;  
77         }
```

### 7.20.2.3 ISound TriDevs.TriEngine2D.Audio.NullAudioManager.GetSound ( string *name* )

Gets the sound with the specified name.

#### Parameters

<i>name</i>	Name of the sound to get.
-------------	---------------------------

#### Returns

An object implementing the [ISound](#) interface, or null if no sound matched the requested name.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```
55         {  
56             return Sound;  
57         }
```

### 7.20.2.4 bool TriDevs.TriEngine2D.Audio.NullAudioManager.HasSong ( string *name* )

Returns a value indicating whether a song with the specified name has been loaded into the audio manager.

#### Parameters

<i>name</i>	Name to check for.
-------------	--------------------

**Returns**

True if the song has been loaded, false otherwise.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```
70     {
71         return false;
72     }
```

**7.20.2.5 bool TriDevs.TriEngine2D.Audio.NullAudioManager.HasSound ( string name )**

Returns a value indicating whether a sound with the specified name has been loaded into the audio manager.

**Parameters**

<i>name</i>	Name to check for.
-------------	--------------------

**Returns**

True if the sound has been loaded, false otherwise.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```
50     {
51         return false;
52     }
```

**7.20.2.6 ISong TriDevs.TriEngine2D.Audio.NullAudioManager.LoadSong ( string name, string file, AudioFormat format = AudioFormat.Ogg )**

Load a song into the audio manager.

If a song with the same name has already been loaded, it will be returned instead.

**Parameters**

<i>name</i>	Name to assign to the new song object.
<i>file</i>	Path to the file that will be used.
<i>format</i>	Format of the file.

**Returns**

The resulting object implementing the [ISong](#) interface.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```
65     {
66         return Song;
67     }
```

**7.20.2.7 ISound TriDevs.TriEngine2D.Audio.NullAudioManager.LoadSound ( string name, string file, AudioFormat format = AudioFormat.Wav )**

Loads a sound into the audio manager.

If a sound with the same name has already been loaded, it will be returned instead.

## Parameters

<i>name</i>	Name to assign to the new sound object.
<i>file</i>	Path to the file that will be used.
<i>format</i>	The format of the file to load.

## Returns

The resulting object implementing the [ISound](#) interface.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

45         {
46             return Sound;
47         }
```

#### 7.20.2.8 void TriDevs.TriEngine2D.Audio.NullAudioManager.StopAll ( )

Immediately stops all playback of sounds and songs.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

40         {
41
42         }
```

#### 7.20.2.9 void TriDevs.TriEngine2D.Audio.NullAudioManager.StopAllSongs ( )

Immediately stops playback of all songs.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

80         {
81
82         }
```

#### 7.20.2.10 void TriDevs.TriEngine2D.Audio.NullAudioManager.StopAllSounds ( )

Immediately stops playback of all sounds.

Implements [TriDevs.TriEngine2D.Audio.IAudioManager](#).

```

60         {
61
62         }
```

The documentation for this class was generated from the following file:

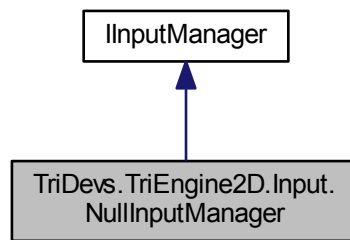
- TriDevs.TriEngine2D/Audio/[NullAudioManager.cs](#)

## 7.21 TriDevs.TriEngine2D.Input.NullInputManager Class Reference

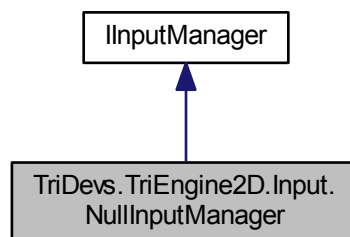
Used as a fallback [InputManager](#) object when the service locator fails to find one.



Inheritance diagram for TriDevs.TriEngine2D.Input.NullInputManager:



Collaboration diagram for TriDevs.TriEngine2D.Input.NullInputManager:



## Public Member Functions

- void `Update` ()  
*Updates the input manager, refreshing all current and previous states.*
- bool `IsKeyUp` (Key key)  
*Returns whether or not the specified key is currently unpressed.*
- bool `IsKeyDown` (Key key)  
*Returns whether or not the specified key is currently being pressed.*
- bool `KeyPressed` (Key key)  
*Returns whether or not the specified key has been pressed.*
- bool `KeyReleased` (Key key)  
*Returns whether or not the specified key has been released.*
- bool `IsMouseUp` (MouseButton button)  
*Returns whether or not the specified mouse button is currently unpressed.*
- bool `IsMouseDown` (MouseButton button)  
*Returns whether or not the specified mouse button is currently being pressed.*
- bool `MousePressed` (MouseButton button)  
*Returns whether or not the specified mouse button has been pressed.*

- bool [MouseReleased](#) (MouseButton button)  
*Returns whether or not the specified mouse button has been released.*
- bool [IsWheelUp](#) ()  
*Returns whether the mouse wheel was scrolled up.*
- bool [IsWheelDown](#) ()  
*Returns whether the mouse wheel was scrolled down.*
- bool [IsWheelChanged](#) ()  
*Returns whether the mouse wheel scrolled at all.*
- int [WheelChange](#) ()  
*Returns the mouse wheel's change in value.*

## Properties

- int [MouseX](#) [get]
- int [MouseY](#) [get]
- Point< int > [MousePosition](#) [get]
- int [MouseWheelValue](#) [get]
- bool [this\[Key key\]](#) [get]
- bool [this\[MouseButton button\]](#) [get]

## Events

- KeyDownEventHandler [KeyDown](#)  
*Raised when a key is pressed down.*
- KeyUpEventHandler [KeyUp](#)  
*Raised when a key is released.*
- KeyPressEventHandler [KeyPress](#)  
*Raised when a character is typed.*
- MouseDownEventHandler [MouseDown](#)  
*Raised when a mouse button is pressed down.*
- MouseUpEventHandler [MouseUp](#)  
*Raised when a mouse button is released.*
- MouseWheelChangedEventHandler [WheelChanged](#)  
*Raised when the mouse wheel value changes.*
- MouseWheelDownEventHandler [WheelDown](#)  
*Raised when the mouse wheel is scrolled downwards.*
- MouseWheelUpEventHandler [WheelUp](#)  
*Raised when the mouse wheel is scrolled upwards.*

### 7.21.1 Detailed Description

Used as a fallback [InputManager](#) object when the service locator fails to find one.

### 7.21.2 Member Function Documentation

#### 7.21.2.1 bool TriDevs.TriEngine2D.Input.NullInputManager.IsKeyDown ( Key key )

Returns whether or not the specified key is currently being pressed.

#### Parameters

<i>key</i>	Key to query for.
------------	-------------------

#### Returns

True if key is currently being pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
132     {  
133         return false;  
134     }
```

#### 7.21.2.2 bool TriDevs.TriEngine2D.Input.NullInputManager.IsKeyUp ( Key *key* )

Returns whether or not the specified key is currently unpressed.

#### Parameters

<i>key</i>	Key to query for.
------------	-------------------

#### Returns

True if the key is currently up (not pressed), false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
127     {  
128         return true;  
129     }
```

#### 7.21.2.3 bool TriDevs.TriEngine2D.Input.NullInputManager.IsMouseDown ( MouseButton *button* )

Returns whether or not the specified mouse button is currently being pressed.

#### Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

#### Returns

True if button is currently being pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
152     {  
153         return false;  
154     }
```

#### 7.21.2.4 bool TriDevs.TriEngine2D.Input.NullInputManager.IsMouseUp ( MouseButton *button* )

Returns whether or not the specified mouse button is currently unpressed.

#### Parameters

<i>button</i>	Button to query for.
---------------	----------------------

**Returns**

True if the button is currently up (not pressed), false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
147         {
148             return true;
149         }
```

**7.21.2.5 bool TriDevs.TriEngine2D.Input.NullInputManager.IsWheelChanged ( )**

Returns whether the mouse wheel scrolled at all.

**Returns**

True if the mouse wheel scrolled, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
177         {
178             return false;
179         }
```

**7.21.2.6 bool TriDevs.TriEngine2D.Input.NullInputManager.IsWheelDown ( )**

Returns whether the mouse wheel was scrolled down.

**Returns**

True if mouse wheel was scrolled down, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
172         {
173             return false;
174         }
```

**7.21.2.7 bool TriDevs.TriEngine2D.Input.NullInputManager.IsWheelUp ( )**

Returns whether the mouse wheel was scrolled up.

**Returns**

True if mouse wheel was scrolled up, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
167         {
168             return false;
169         }
```

**7.21.2.8 bool TriDevs.TriEngine2D.Input.NullInputManager.KeyPressed ( Key key )**

Returns whether or not the specified key has been pressed.

Only returns true if the last state of the key was not pressed.

**Parameters**

<i>key</i>	Key to query for.
------------	-------------------

#### Returns

True if key was pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
137     {  
138         return false;  
139     }
```

#### 7.21.2.9 bool TriDevs.TriEngine2D.Input.NullInputManager.KeyReleased ( Key *key* )

Returns whether or not the specified key has been released.

Only returns true if the last state of the key was pressed.

#### Parameters

<i>key</i>	Key to query for.
------------	-------------------

#### Returns

True if key was released, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
142     {  
143         return false;  
144     }
```

#### 7.21.2.10 bool TriDevs.TriEngine2D.Input.NullInputManager.MousePressed ( MouseButton *button* )

Returns whether or not the specified mouse button has been pressed.

Only returns true if the last state of the mouse button was not pressed.

#### Parameters

<i>button</i>	Button to query for.
---------------	----------------------

#### Returns

True if button was pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
157     {  
158         return false;  
159     }
```

#### 7.21.2.11 bool TriDevs.TriEngine2D.Input.NullInputManager.MouseReleased ( MouseButton *button* )

Returns whether or not the specified mouse button has been released.

Only returns true if the last state of the button was pressed.

## Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

## Returns

True if the button was released, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

162     {
163         return false;
164     }
```

#### 7.21.2.12 void TriDevs.TriEngine2D.Input.NullInputManager.Update ( )

Updates the input manager, refreshing all current and previous states.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

122     {
123         // Do nothing
124     }
```

#### 7.21.2.13 int TriDevs.TriEngine2D.Input.NullInputManager.WheelChange ( )

Returns the mouse wheel's change in value.

## Returns

Negative value if wheel scrolled down, positive value if scrolled up, zero if not scrolled.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

182     {
183         return 0;
184     }
```

### 7.21.3 Property Documentation

7.21.3.1 `Point<int> TriDevs.TriEngine2D.Input.NullInputManager.MousePosition` [get]

7.21.3.2 `int TriDevs.TriEngine2D.Input.NullInputManager.MouseWheelValue` [get]

7.21.3.3 `int TriDevs.TriEngine2D.Input.NullInputManager.MouseX` [get]

7.21.3.4 `int TriDevs.TriEngine2D.Input.NullInputManager.MouseY` [get]

7.21.3.5 `bool TriDevs.TriEngine2D.Input.NullInputManager.this[Key key]` [get]

7.21.3.6 `bool TriDevs.TriEngine2D.Input.NullInputManager.this[MouseButton button]` [get]

### 7.21.4 Event Documentation

7.21.4.1 `KeyDownEventHandler TriDevs.TriEngine2D.Input.NullInputManager.KeyDown`

Raised when a key is pressed down.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.21.4.2 KeyPressEventHandler TriDevs.TriEngine2D.Input.NullInputManager.KeyPress

Raised when a character is typed.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.21.4.3 KeyUpEventHandler TriDevs.TriEngine2D.Input.NullInputManager.KeyUp

Raised when a key is released.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.21.4.4 MouseDownEventHandler TriDevs.TriEngine2D.Input.NullInputManager.MouseDown

Raised when a mouse button is pressed down.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.21.4.5 MouseUpEventHandler TriDevs.TriEngine2D.Input.NullInputManager.MouseUp

Raised when a mouse button is released.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.21.4.6 MouseWheelChangedEventHandler TriDevs.TriEngine2D.Input.NullInputManager.WheelChanged

Raised when the mouse wheel value changes.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.21.4.7 MouseWheelDownEventHandler TriDevs.TriEngine2D.Input.NullInputManager.WheelDown

Raised when the mouse wheel is scrolled downwards.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.21.4.8 MouseWheelUpEventHandler TriDevs.TriEngine2D.Input.NullInputManager.WheelUp

Raised when the mouse wheel is scrolled upwards.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

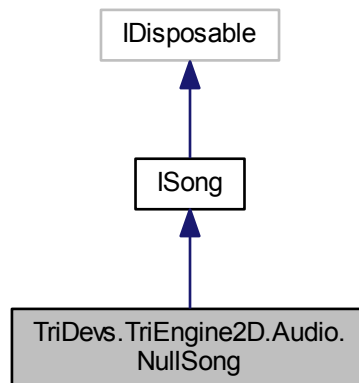
The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Input/[NullInputManager.cs](#)

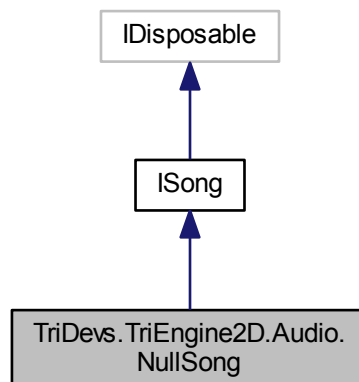
## 7.22 TriDevs.TriEngine2D.Audio.NullSong Class Reference

Fallback song class used in [NullAudioManager](#).

Inheritance diagram for TriDevs.TriEngine2D.Audio.NullSong:



Collaboration diagram for TriDevs.TriEngine2D.Audio.NullSong:



## Public Member Functions

- void `Dispose` ()
- void `Play` ()  
*Starts playback of the song.*
- void `Stop` ()  
*Stops playback of the song.*
- void `Pause` ()  
*Pauses playback of the song.*
- void `Resume` ()  
*Resumes playback of a paused song.*



## Properties

- string [Name](#) [get]
- string [File](#) [get]
- float [Volume](#) [get, set]
- bool [IsLooped](#) [get, set]

### 7.22.1 Detailed Description

Fallback song class used in [NullAudioManager](#).

### 7.22.2 Member Function Documentation

#### 7.22.2.1 void TriDevs.TriEngine2D.Audio.NullSong.Dispose ( )

```
37         {  
38  
39     }
```

#### 7.22.2.2 void TriDevs.TriEngine2D.Audio.NullSong.Pause ( )

Pauses playback of the song.

Call the [Resume](#) or [Play](#) method to resume playback.

Implements [TriDevs.TriEngine2D.Audio.ISong](#).

```
52         {  
53  
54     }
```

#### 7.22.2.3 void TriDevs.TriEngine2D.Audio.NullSong.Play ( )

Starts playback of the song.

Implements [TriDevs.TriEngine2D.Audio.ISong](#).

```
42         {  
43  
44     }
```

#### 7.22.2.4 void TriDevs.TriEngine2D.Audio.NullSong.Resume ( )

Resumes playback of a paused song.

Implements [TriDevs.TriEngine2D.Audio.ISong](#).

```
57         {  
58  
59     }
```

#### 7.22.2.5 void TriDevs.TriEngine2D.Audio.NullSong.Stop ( )

Stops playback of the song.

Implements [TriDevs.TriEngine2D.Audio.ISong](#).

```
47         {  
48  
49     }
```

### 7.22.3 Property Documentation

7.22.3.1 `string TriDevs.TriEngine2D.Audio.NullSong.File` `[get]`

7.22.3.2 `bool TriDevs.TriEngine2D.Audio.NullSong.IsLooped` `[get]`, `[set]`

7.22.3.3 `string TriDevs.TriEngine2D.Audio.NullSong.Name` `[get]`

7.22.3.4 `float TriDevs.TriEngine2D.Audio.NullSong.Volume` `[get]`, `[set]`

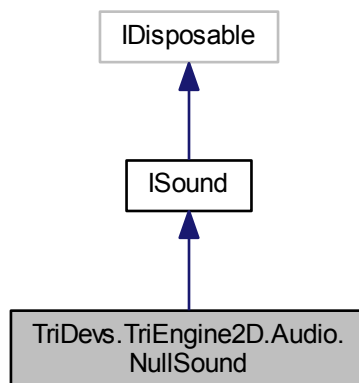
The documentation for this class was generated from the following file:

- `TriDevs.TriEngine2D/Audio/NullSong.cs`

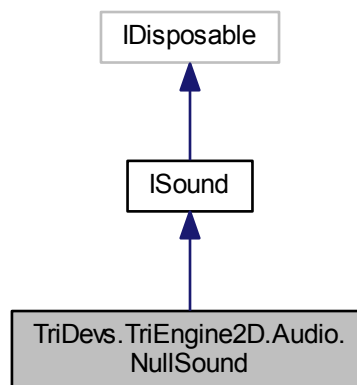
## 7.23 TriDevs.TriEngine2D.Audio.NullSound Class Reference

Fallback sound class used in [NullAudioManager](#).

Inheritance diagram for `TriDevs.TriEngine2D.Audio.NullSound`:



Collaboration diagram for TriDevs.TriEngine2D.Audio.NullSound:



## Public Member Functions

- void [Play](#) ()  
*Plays the sound.*
- void [Stop](#) ()  
*Stops the sound.*
- void [Dispose](#) ()

## Properties

- string [Name](#) [get]
- string [File](#) [get]

### 7.23.1 Detailed Description

Fallback sound class used in [NullAudioManager](#).

### 7.23.2 Member Function Documentation

#### 7.23.2.1 void TriDevs.TriEngine2D.Audio.NullSound.Dispose ( )

```
45     {  
46  
47     }
```

#### 7.23.2.2 void TriDevs.TriEngine2D.Audio.NullSound.Play ( )

Plays the sound.

Implements [TriDevs.TriEngine2D.Audio.ISound](#).

```

35         {
36
37     }

```

### 7.23.2.3 void TriDevs.TriEngine2D.Audio.NullSound.Stop ( )

Stops the sound.

Implements [TriDevs.TriEngine2D.Audio.ISound](#).

```

40         {
41
42     }

```

## 7.23.3 Property Documentation

7.23.3.1 string TriDevs.TriEngine2D.Audio.NullSound.File [get]

7.23.3.2 string TriDevs.TriEngine2D.Audio.NullSound.Name [get]

The documentation for this class was generated from the following file:

- [TriDevs.TriEngine2D/Audio/NullSound.cs](#)

## 7.24 TriDevs.TriEngine2D.Point< T > Struct Template Reference

A struct representing an X/Y coordinate.

### Public Member Functions

- [Point](#) (T x, T y)  
*Creates a new Point<T> with the specified X and Y values.*

### Public Attributes

- T [X](#)  
*The X value of the coordinate.*
- T [Y](#)  
*The Y value of the coordinate.*

### 7.24.1 Detailed Description

A struct representing an X/Y coordinate.

#### Template Parameters

<i>T</i>	The type used for the X and Y members.
----------	--

#### Type Constraints

***T* : struct**

## 7.24.2 Constructor & Destructor Documentation

### 7.24.2.1 TriDevs.TriEngine2D.Point< T >.Point ( T x, T y )

Creates a new Point<T> with the specified X and Y values.

#### Parameters

x	The X value.
y	The Y value.

```

48     {
49         x = x;
50         y = y;
51     }
```

## 7.24.3 Member Data Documentation

### 7.24.3.1 T TriDevs.TriEngine2D.Point< T >.X

The X value of the coordinate.

### 7.24.3.2 T TriDevs.TriEngine2D.Point< T >.Y

The Y value of the coordinate.

The documentation for this struct was generated from the following file:

- TriDevs.TriEngine2D/[Point.cs](#)

## 7.25 TriDevs.TriEngine2D.Serializing.Serializer Class Reference

Provides serialization methods.

### Static Public Member Functions

- static string [Serialize< T >](#) (T data)  
*Serialize an object to string.*
- static void [Serialize< T >](#) (T data, string file, Formatting formatting=Formatting.Indented)  
*Serializes an object to file.*
- static T [Deserialize< T >](#) (string file)  
*Deserialize a serialized object from file.*

### 7.25.1 Detailed Description

Provides serialization methods.

### 7.25.2 Member Function Documentation

#### 7.25.2.1 static T TriDevs.TriEngine2D.Serializing.Serializer.Deserialize< T > ( string file ) [static]

Deserialize a serialized object from file.

## Template Parameters

<i>T</i>	Type of the object being deserialized.
----------	--

## Parameters

<i>file</i>	File to read from.
-------------	--------------------

## Returns

The deserialized object.

```

84         {
85             T data;
86             // Create the StreamReader
87             using (var reader = new StreamReader(file))
88             {
89                 // And the json reader
90                 using (var jsonReader = new JsonTextReader(reader))
91                 {
92                     // Now deserialize the file to the requested object...
93                     data = JsonSerializer.Value.Deserialize<T>(jsonReader);
94                     // ... and close the json reader.
95                     jsonReader.Close();
96                 }
97                 // Finally, close the file reader
98                 reader.Close();
99             }
100             return data;
101         }

```

### 7.25.2.2 static string TriDevs.TriEngine2D.Serializing.Serializer.Serialize< T > ( T data ) [static]

Serialize an object to string.

## Template Parameters

<i>T</i>	Type of data.
----------	---------------

## Parameters

<i>data</i>	Data to serialize.
-------------	--------------------

## Returns

The serialized object in string format.

```

44         {
45             // Create a StringWriter to hold the serialized object
46             var writer = new StringWriter();
47             // Serialize the object into the writer
48             JsonSerializer.Value.Serialize(writer, data);
49             return writer.ToString(); // Return the data as string
50         }

```

### 7.25.2.3 static void TriDevs.TriEngine2D.Serializing.Serializer.Serialize< T > ( T data, string file, Formatting formatting = Formatting.Indented ) [static]

Serializes an object to file.

## Template Parameters

<i>T</i>	Type of the data.
----------	-------------------

## Parameters

<i>data</i>	Data to serialize.
<i>file</i>	File to serialize to.
<i>formatting</i>	The formatting to use for the JSON output.

```

60     {
61         // Create the StreamWriter
62         using (var writer = new StreamWriter(file, false))
63         {
64             // Create the json writer
65             using (var jsonWriter = new JsonTextWriter(writer){Formatting = formatting})
66             {
67                 // Now serialize the object to the file...
68                 JsonSerializer.Value.Serialize(jsonWriter, data);
69                 // ... and close the json writer
70                 jsonWriter.Close();
71             }
72             // Finally, close the file writer
73             writer.Close();
74         }
75     }

```

The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Serializing/[Serializer.cs](#)

## 7.26 TriDevs.TriEngine2D.Services Class Reference

Provides different game-related service interfaces.

### Static Public Member Functions

- static void [Provide](#) ([IInputManager](#) input)  
*Specifies an input manager service to provide.*
- static void [Provide](#) ([IAudioManager](#) audio)  
*Specifies an audio manager service to provide.*
- static void [Provide](#) ([IInputManager](#) input, [IAudioManager](#) audio)  
*Specifies what services to provide.*

### Properties

- static [IInputManager](#) [Input](#) [get]  
*The input manager service.*
- static [IAudioManager](#) [Audio](#) [get]  
*The audio manager service.*

### 7.26.1 Detailed Description

Provides different game-related service interfaces.

Actual service providers must be supplied from external code. All Service properties are initialized with Null-type services that provide no real functionality.

### 7.26.2 Member Function Documentation

#### 7.26.2.1 static void TriDevs.TriEngine2D.Services.Provide ( IInputManager input ) [static]

Specifies an input manager service to provide.

**Parameters**

<i>input</i>	An object implementing the <code>IInputManager</code> interface.
--------------	--

```

57         {
58             _input = input;
59         }

```

**7.26.2.2 static void TriDevs.TriEngine2D.Services.Provide ( `IAudioManager audio` ) [static]**

Specifies an audio manager service to provide.

**Parameters**

<i>audio</i>	An object implementing the <code>IAudioManager</code> interface.
--------------	--

```

66         {
67             _audio = audio;
68         }

```

**7.26.2.3 static void TriDevs.TriEngine2D.Services.Provide ( `IInputManager input`, `IAudioManager audio` ) [static]**

Specifies what services to provide.

**Parameters**

<i>input</i>	The input service to provide.
<i>audio</i>	The audio service to provide.

```

76         {
77             Provide(input);
78             Provide(audio);
79         }

```

**7.26.3 Property Documentation****7.26.3.1 `IAudioManager TriDevs.TriEngine2D.Services.Audio` [static], [get]**

The audio manager service.

**7.26.3.2 `IInputManager TriDevs.TriEngine2D.Services.Input` [static], [get]**

The input manager service.

The documentation for this class was generated from the following file:

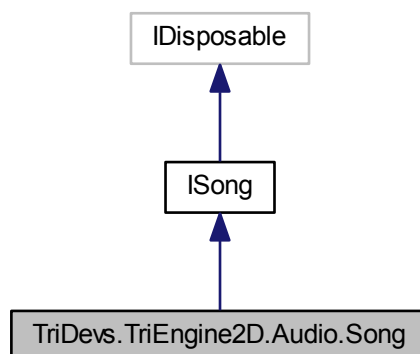
- `TriDevs.TriEngine2D/Services.cs`

**7.27 TriDevs.TriEngine2D.Audio.Song Class Reference**

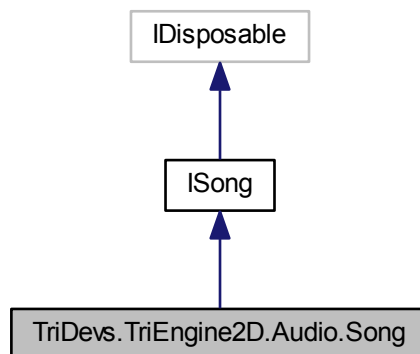
[Song](#) class that can be used with [AudioManager](#).



Inheritance diagram for TriDevs.TriEngine2D.Audio.Song:



Collaboration diagram for TriDevs.TriEngine2D.Audio.Song:



## Public Member Functions

- void [Dispose](#) ()
- void [Play](#) ()  
*Starts playback of the song.*
- void [Stop](#) ()  
*Stops playback of the song.*
- void [Pause](#) ()  
*Pauses playback of the song.*
- void [Resume](#) ()  
*Resumes playback of a paused song.*

## Properties

- string [Name](#) [get]
- string [File](#) [get]
- float [Volume](#) [get, set]
- bool [IsLooped](#) [get, set]

### 7.27.1 Detailed Description

[Song](#) class that can be used with [AudioManager](#).

The song class will stream from a specified file, instead of loading it into memory.

### 7.27.2 Member Function Documentation

#### 7.27.2.1 void TriDevs.TriEngine2D.Audio.Song.Dispose ( )

```

80         {
81             if (_stream == null)
82                 return;
83
84             Stop();
85             _stream.Dispose();
86             _stream = null;
87         }

```

#### 7.27.2.2 void TriDevs.TriEngine2D.Audio.Song.Pause ( )

Pauses playback of the song.

Call the [Resume](#) or [Play](#) method to resume playback.

Implements [TriDevs.TriEngine2D.Audio.ISong](#).

```

108         {
109             _stream.Pause();
110         }

```

#### 7.27.2.3 void TriDevs.TriEngine2D.Audio.Song.Play ( )

Starts playback of the song.

Implements [TriDevs.TriEngine2D.Audio.ISong](#).

```

90         {
91             try
92             {
93                 _stream.Play();
94             }
95             catch (InvalidDataException)
96             {
97                 Stop();
98                 Play();
99             }
100        }

```

#### 7.27.2.4 void TriDevs.TriEngine2D.Audio.Song.Resume ( )

Resumes playback of a paused song.

Implements [TriDevs.TriEngine2D.Audio.ISong](#).

```
113     {  
114         _stream.Resume();  
115     }
```

#### 7.27.2.5 void TriDevs.TriEngine2D.Audio.Song.Stop ( )

Stops playback of the song.

Implements [TriDevs.TriEngine2D.Audio.ISong](#).

```
103     {  
104         _stream.Stop();  
105     }
```

### 7.27.3 Property Documentation

7.27.3.1 string TriDevs.TriEngine2D.Audio.Song.File [get]

7.27.3.2 bool TriDevs.TriEngine2D.Audio.Song.IsLooped [get], [set]

7.27.3.3 string TriDevs.TriEngine2D.Audio.Song.Name [get]

7.27.3.4 float TriDevs.TriEngine2D.Audio.Song.Volume [get], [set]

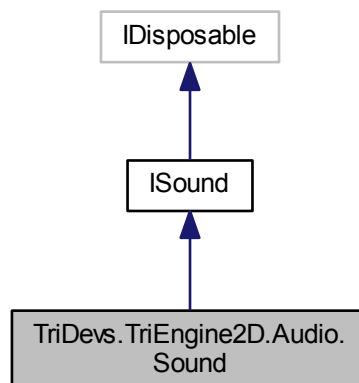
The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Audio/[Song.cs](#)

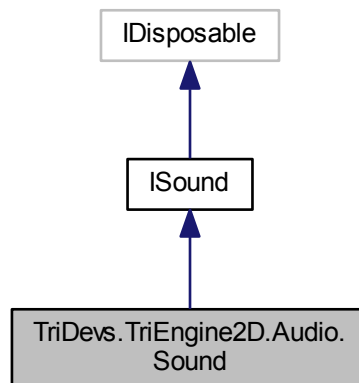
## 7.28 TriDevs.TriEngine2D.Audio.Sound Class Reference

[Sound](#) class that can be used with the [AudioManager](#).

Inheritance diagram for TriDevs.TriEngine2D.Audio.Sound:



Collaboration diagram for TriDevs.TriEngine2D.Audio.Sound:



## Public Member Functions

- void [Play](#) ()  
*Plays the sound.*
- void [Stop](#) ()  
*Stops the sound.*
- void [Dispose](#) ()

## Properties

- string [Name](#) [get]
- string [File](#) [get]

### 7.28.1 Detailed Description

[Sound](#) class that can be used with the [AudioManager](#).

### 7.28.2 Member Function Documentation

#### 7.28.2.1 void TriDevs.TriEngine2D.Audio.Sound.Dispose ( )

```
187         {  
188             Stop();  
189             AL.DeleteSources(_sources);  
190             AL.DeleteBuffer(_buffer);  
191         }
```

#### 7.28.2.2 void TriDevs.TriEngine2D.Audio.Sound.Play ( )

Plays the sound.

Implements [TriDevs.TriEngine2D.Audio.ISound](#).

```

169         {
170             for (var i = 0; i < SourceCount; i++)
171             {
172                 if (GetSourceState(_sources[i]) != ALSourceState.Playing)
173                 {
174                     AL.SourcePlay(_sources[i]);
175                     return;
176                 }
177             }
178         }

```

### 7.28.2.3 void TriDevs.TriEngine2D.Audio.Sound.Stop ( )

Stops the sound.

Implements [TriDevs.TriEngine2D.Audio.ISound](#).

```

181         {
182             for (var i = 0; i < SourceCount; i++)
183                 AL.SourceStop(_sources[i]);
184         }

```

## 7.28.3 Property Documentation

7.28.3.1 string TriDevs.TriEngine2D.Audio.Sound.File [get]

7.28.3.2 string TriDevs.TriEngine2D.Audio.Sound.Name [get]

The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Audio/[Sound.cs](#)

## 7.29 TriDevs.TriEngine2D.Extensions.StringExtensions Class Reference

[Extensions](#) for System.String

### Static Public Member Functions

- static string [ReplaceFirst](#) (this string s, string search, string replace, bool caseInsensitive=false)  
*Returns a string in which the first occurrence of a specified string is replaced with another string.*
- static string [Replace](#) (this string s, string search, string replace, int count, bool caseInsensitive=false)  
*Returns a string in which the N first occurrences of a specified string are replaced with another string.*
- static string [Replace](#) (this string s, string search, string replace, bool caseInsensitive=false)  
*Returns a string in which all occurrences of a specified string are replaced with another string.*

### 7.29.1 Detailed Description

[Extensions](#) for System.String

### 7.29.2 Member Function Documentation

7.29.2.1 static string TriDevs.TriEngine2D.Extensions.StringExtensions.Replace ( this string s, string search, string replace, int count, bool caseInsensitive = false ) [static]

Returns a string in which the N first occurrences of a specified string are replaced with another string.

## Parameters

<i>s</i>	String to modify.
<i>search</i>	String to search for.
<i>replace</i>	String to replace the match(es) with.
<i>count</i>	Number of occurrences to replace.
<i>caseInsensitive</i>	True for case insensitive search, false for case sensitive.

## Returns

The supplied string with the N first occurrences of the specified string replaced with the other.

```

56     {
57         var re = caseInsensitive ? new Regex(search, RegexOptions.IgnoreCase) : new Regex(search);
58         return re.Replace(s, replace, count);
59     }

```

**7.29.2.2** `static string TriDevs.TriEngine2D.Extensions.StringExtensions.Replace ( this string s, string search, string replace, bool caseInsensitive = false ) [static]`

Returns a string in which all occurrences of a specified string are replaced with another string.

This extension method supports case insensitive searches.

## Parameters

<i>s</i>	String to modify.
<i>search</i>	String to search for.
<i>replace</i>	String to replace the match(es) with.
<i>caseInsensitive</i>	True for case insensitive search, false for case sensitive.

## Returns

The supplied string with all occurrences of the specified string replaced with the other.

```

73     {
74         var re = caseInsensitive ? new Regex(search, RegexOptions.IgnoreCase) : new Regex(search);
75         return re.Replace(s, replace);
76     }

```

**7.29.2.3** `static string TriDevs.TriEngine2D.Extensions.StringExtensions.ReplaceFirst ( this string s, string search, string replace, bool caseInsensitive = false ) [static]`

Returns a string in which the first occurrence of a specified string is replaced with another string.

## Parameters

<i>s</i>	String to modify.
<i>search</i>	String to search for.
<i>replace</i>	String to replace the match with.
<i>caseInsensitive</i>	True for case insensitive search, false for case sensitive.

## Returns

The supplied string with the first occurrence of the specified string replaced with the other.

```

42     {
43         return Replace(s, search, replace, 1, caseInsensitive);
44     }

```

The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Extensions/[StringExtensions.cs](#)

## 7.30 TriDevs.TriEngine2D.Helpers.Threading Class Reference

Provides various helper functions for doing threading operations.

### Static Public Member Functions

- static void [SetCurrentThreadName](#) (string name)  
*Sets the name of the current thread, does nothing if the thread already has a name.*

#### 7.30.1 Detailed Description

Provides various helper functions for doing threading operations.

#### 7.30.2 Member Function Documentation

##### 7.30.2.1 static void TriDevs.TriEngine2D.Helpers.Threading.SetCurrentThreadName ( string name ) [static]

Sets the name of the current thread, does nothing if the thread already has a name.

#### Parameters

<i>name</i>	The new name for the current thread
-------------	-------------------------------------

```

39         {
40             // We can't set the name on a thread if it's already set, it would throw an exception
41             // So we have to check if the current name is null before trying to set a new one
42             if (string.IsNullOrEmpty(Thread.CurrentThread.Name))
43                 Thread.CurrentThread.Name = name;
44         }

```

The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Helpers/[Threading.cs](#)

## 7.31 TriDevs.TriEngine2D.Version Class Reference

[Version](#) class specifying the version of this project.

### Public Attributes

- const int [Major](#) = 0  
*Major version of the project.*
- const int [Minor](#) = 0  
*Minor version of the project.*
- const int [Patch](#) = 9  
*Patch version of the project.*
- const string [Suffix](#) = ""  
*Optional suffix, empty if no suffix for this version.*

- const string [VersionStringFormat](#) = "{0}.{1}.{2}"

*The format string used when formatting major, minor and patch version to their string representation.*

- const string [VersionStringFormatWithSuffix](#) = [VersionStringFormat](#) + "-{3}"

*The format string used when formatting major, minor and patch version to their string representation (with suffix).*

## Properties

- static string [VersionString](#) [get]

*String representation of the current project version.*

### 7.31.1 Detailed Description

[Version](#) class specifying the version of this project.

### 7.31.2 Member Data Documentation

#### 7.31.2.1 const int TriDevs.TriEngine2D.Version.Major = 0

Major version of the project.

#### 7.31.2.2 const int TriDevs.TriEngine2D.Version.Minor = 0

Minor version of the project.

#### 7.31.2.3 const int TriDevs.TriEngine2D.Version.Patch = 9

Patch version of the project.

#### 7.31.2.4 const string TriDevs.TriEngine2D.Version.Suffix = ""

Optional suffix, empty if no suffix for this version.

Example values could be "beta" and "alpha".

#### 7.31.2.5 const string TriDevs.TriEngine2D.Version.VersionStringFormat = "{0}.{1}.{2}"

The format string used when formatting major, minor and patch version to their string representation.

#### 7.31.2.6 const string TriDevs.TriEngine2D.Version.VersionStringFormatWithSuffix = VersionStringFormat + "-{3}"

The format string used when formatting major, minor and patch version to their string representation (with suffix).

### 7.31.3 Property Documentation

#### 7.31.3.1 string TriDevs.TriEngine2D.Version.VersionString [static], [get]

String representation of the current project version.

The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/[Version.cs](#)



## 7.32 TriDevs.TriEngine2D.Native.WinAPI Class Reference

Holds various [WinAPI](#) stuff.

### Public Member Functions

- static IntPtr [GetStdHandle](#) (int nStdHandle)  
*Retrieves a handle to the specified standard device (standard input, standard output, or standard error).*
- static bool [AllocConsole](#) ()  
*Allocates a new console for the calling process.*
- static int [FreeConsole](#) ()  
*Detaches the calling process from its console.*

### Public Attributes

- const int [STD\\_OUTPUT\\_HANDLE](#) = -11  
*The standard output device. Initially, this is the active console screen buffer, CONOUT\$.*
- const int [CODE\\_PAGE](#) = 437  
*The code page to use for the console.*

### 7.32.1 Detailed Description

Holds various [WinAPI](#) stuff.

### 7.32.2 Member Function Documentation

#### 7.32.2.1 static bool TriDevs.TriEngine2D.Native.WinAPI.AllocConsole ( )

Allocates a new console for the calling process.

#### Returns

If the function succeeds, the return value is nonzero. If the function fails, the return value is zero. To get extended error information, call GetLastError.

Here is the caller graph for this function:



#### 7.32.2.2 static int TriDevs.TriEngine2D.Native.WinAPI.FreeConsole ( )

Detaches the calling process from its console.

**Returns**

If the function succeeds, the return value is nonzero. If the function fails, the return value is zero. To get extended error information, call GetLastError.

Here is the caller graph for this function:

**7.32.2.3 static IntPtr TriDevs.TriEngine2D.Native.WinAPI.GetStdHandle ( int nStdHandle )**

Retrieves a handle to the specified standard device (standard input, standard output, or standard error).

**Parameters**

<i>nStdHandle</i>	The standard device.
-------------------	----------------------

**Returns**

If the function succeeds, the return value is a handle to the specified device, or a redirected handle set by a previous call to SetStdHandle. The handle has GENERIC\_READ and GENERIC\_WRITE access rights, unless the application has used SetStdHandle to set a standard handle with lesser access. If the function fails, the return value is INVALID\_HANDLE\_VALUE. To get extended error information, call GetLastError. If an application does not have associated standard handles, such as a service running on an interactive desktop, and has not redirected them, the return value is NULL.

Here is the caller graph for this function:

**7.32.3 Member Data Documentation****7.32.3.1 const int TriDevs.TriEngine2D.Native.WinAPI.CODE\_PAGE = 437**

The code page to use for the console.

**7.32.3.2 const int TriDevs.TriEngine2D.Native.WinAPI.STD\_OUTPUT\_HANDLE = -11**

The standard output device. Initially, this is the active console screen buffer, CONOUT\$.

The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Native/[WinAPI.cs](#)



## Chapter 8

# File Documentation

### 8.1 README.md File Reference

### 8.2 TriDevs.TriEngine2D/Audio/AudioFormat.cs File Reference

#### Namespaces

- package [TriDevs.TriEngine2D.Audio](#)

#### Enumerations

- enum [TriDevs.TriEngine2D.Audio.AudioFormat](#) { [TriDevs.TriEngine2D.Audio.Ogg](#), [TriDevs.TriEngine2D.Audio.Wav](#) }

*Different audio formats usable with audio managers.*

### 8.3 TriDevs.TriEngine2D/Audio/AudioManager.cs File Reference

#### Classes

- class [TriDevs.TriEngine2D.Audio.AudioManager](#)

*Class to manage engine audio.*

#### Namespaces

- package [TriDevs.TriEngine2D.Audio](#)

### 8.4 TriDevs.TriEngine2D/Audio/IAudioManager.cs File Reference

#### Classes

- interface [TriDevs.TriEngine2D.Audio.IAudioManager](#)

*Provides various methods to manipulate audio.*

## Namespaces

- package [TriDevs.TriEngine2D.Audio](#)

## 8.5 TriDevs.TriEngine2D/Audio/ISong.cs File Reference

### Classes

- interface [TriDevs.TriEngine2D.Audio.ISong](#)  
*A song that will be streamed in the audio player.*

## Namespaces

- package [TriDevs.TriEngine2D.Audio](#)

## 8.6 TriDevs.TriEngine2D/Audio/ISound.cs File Reference

### Classes

- interface [TriDevs.TriEngine2D.Audio.ISound](#)  
*A sound file for use with the audio manager.*

## Namespaces

- package [TriDevs.TriEngine2D.Audio](#)

## 8.7 TriDevs.TriEngine2D/Audio/NullAudioManager.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Audio.NullAudioManager](#)  
*Used as a fallback [AudioManager](#) object when the service locator fails to find one.*

## Namespaces

- package [TriDevs.TriEngine2D.Audio](#)

## 8.8 TriDevs.TriEngine2D/Audio/NullSong.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Audio.NullSong](#)  
*Fallback song class used in [NullAudioManager](#).*

## Namespaces

- package [TriDevs.TriEngine2D.Audio](#)

## 8.9 TriDevs.TriEngine2D/Audio/NullSound.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Audio.NullSound](#)  
*Fallback sound class used in [NullAudioManager](#).*

### Namespaces

- package [TriDevs.TriEngine2D.Audio](#)

## 8.10 TriDevs.TriEngine2D/Audio/Song.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Audio.Song](#)  
*[Song](#) class that can be used with [AudioManager](#).*

### Namespaces

- package [TriDevs.TriEngine2D.Audio](#)

## 8.11 TriDevs.TriEngine2D/Audio/Sound.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Audio.Sound](#)  
*[Sound](#) class that can be used with the [AudioManager](#).*

### Namespaces

- package [TriDevs.TriEngine2D.Audio](#)

## 8.12 TriDevs.TriEngine2D/Extensions/EnumerationExtensions.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Extensions.EnumerationExtensions](#)  
*[Extensions](#) for [System.Enum](#).*

### Namespaces

- package [TriDevs.TriEngine2D.Extensions](#)

## 8.13 TriDevs.TriEngine2D/Extensions/StringExtensions.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Extensions.StringExtensions](#)  
*Extensions for System.String*

### Namespaces

- package [TriDevs.TriEngine2D.Extensions](#)

## 8.14 TriDevs.TriEngine2D/Helpers/IO.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Helpers.IO](#)  
*Provides various helper functions for doing IO operations.*

### Namespaces

- package [TriDevs.TriEngine2D.Helpers](#)

## 8.15 TriDevs.TriEngine2D/Helpers/Math.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Helpers.Math](#)  
*Various helper methods for working with math.*

### Namespaces

- package [TriDevs.TriEngine2D.Helpers](#)

## 8.16 TriDevs.TriEngine2D/Helpers/Threading.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Helpers.Threading](#)  
*Provides various helper functions for doing threading operations.*

### Namespaces

- package [TriDevs.TriEngine2D.Helpers](#)



## 8.17 TriDevs.TriEngine2D/Input/Events/Key.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Input.Events.KeyEventArgs](#)  
*EventArgs class used for key-related events. Contains information about the key related with the event.*
- class [TriDevs.TriEngine2D.Input.Events.KeyCharEventArgs](#)  
*EventArgs class used for keychar-related events. Contains information about the character related with the event.*

### Namespaces

- package [TriDevs.TriEngine2D.Input.Events](#)

### Functions

- delegate void [TriDevs.TriEngine2D.Input.Events.KeyDownEventHandler](#) (object sender, KeyEventArgs e)  
*Event handler delegate for the KeyDown event.*
- delegate void [TriDevs.TriEngine2D.Input.Events.KeyUpEventHandler](#) (object sender, KeyEventArgs e)  
*Event handler delegate for the KeyUp event.*
- delegate void [TriDevs.TriEngine2D.Input.Events.KeyPressEventHandler](#) (object sender, KeyCharEventArgs e)  
*Event handler delegate for the KeyPress event.*

## 8.18 TriDevs.TriEngine2D/Input/Events/Mouse.cs File Reference

### Namespaces

- package [TriDevs.TriEngine2D.Input.Events](#)

### Functions

- delegate void [TriDevs.TriEngine2D.Input.Events.MouseDownEventHandler](#) (object sender, MouseButtonEventArgs e)  
*Event handler delegate for the MouseDown event.*
- delegate void [TriDevs.TriEngine2D.Input.Events.MouseUpEventHandler](#) (object sender, MouseButtonEventArgs e)  
*Event handler delegate for the MouseUp event.*
- delegate void [TriDevs.TriEngine2D.Input.Events.MouseWheelChangedEventHandler](#) (object sender, MouseWheelEventArgs e)  
*Event handler delegate for the MouseWheelChanged event.*
- delegate void [TriDevs.TriEngine2D.Input.Events.MouseWheelDownEventHandler](#) (object sender, MouseWheelEventArgs e)  
*Event handler delegate for the MouseWheelDown event.*
- delegate void [TriDevs.TriEngine2D.Input.Events.MouseWheelUpEventHandler](#) (object sender, MouseWheelEventArgs e)  
*Event handler delegate for the MouseWheelUp event.*

## 8.19 TriDevs.TriEngine2D/Input/IInputManager.cs File Reference

### Classes

- interface [TriDevs.TriEngine2D.Input.IInputManager](#)  
*Provides various methods to query input devices like the keyboard.*

### Namespaces

- package [TriDevs.TriEngine2D.Input](#)

## 8.20 TriDevs.TriEngine2D/Input/InputManager.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Input.InputManager](#)  
*Input manager interfacing with input methods provided by a GameWindow.*

### Namespaces

- package [TriDevs.TriEngine2D.Input](#)

## 8.21 TriDevs.TriEngine2D/Input/NullInputManager.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Input.NullInputManager](#)  
*Used as a fallback [InputManager](#) object when the service locator fails to find one.*

### Namespaces

- package [TriDevs.TriEngine2D.Input](#)

## 8.22 TriDevs.TriEngine2D/Interfaces/IDrawable.cs File Reference

### Classes

- interface [TriDevs.TriEngine2D.Interfaces.IDrawable](#)  
*Indicates that the implementing class is able to be drawn on the screen.*

### Namespaces

- package [TriDevs.TriEngine2D.Interfaces](#)

## 8.23 TriDevs.TriEngine2D/Logging/LogManager.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Logging.LogManager](#)

*Class to manage logging. ILog interfaces should be obtained from this class' methods, as opposed to calling default log4net methods.*

### Namespaces

- package [TriDevs.TriEngine2D.Logging](#)

## 8.24 TriDevs.TriEngine2D/Native/Helpers.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Native.Helpers](#)

*Helper class with various methods to help native coding and debugging.*

### Namespaces

- package [TriDevs.TriEngine2D.Native](#)

## 8.25 TriDevs.TriEngine2D/Native/WinAPI.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Native.WinAPI](#)

*Holds various [WinAPI](#) stuff.*

### Namespaces

- package [TriDevs.TriEngine2D.Native](#)

## 8.26 TriDevs.TriEngine2D/Point.cs File Reference

### Classes

- struct [TriDevs.TriEngine2D.Point< T >](#)

*A struct representing an X/Y coordinate.*

### Namespaces

- package [TriDevs.TriEngine2D](#)

## 8.27 TriDevs.TriEngine2D/Properties/AssemblyInfo.cs File Reference

## 8.28 TriDevs.TriEngine2D/Serializing/Serializer.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Serializing.Serializer](#)  
*Provides serialization methods.*

### Namespaces

- package [TriDevs.TriEngine2D.Serializing](#)

## 8.29 TriDevs.TriEngine2D/Services.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Services](#)  
*Provides different game-related service interfaces.*

### Namespaces

- package [TriDevs.TriEngine2D](#)

## 8.30 TriDevs.TriEngine2D/UI/Color.cs File Reference

### Classes

- struct [TriDevs.TriEngine2D.UI.Color](#)  
*Represents an RGBA color that can be used with [TriEngine2D UI](#) controls.*

### Namespaces

- package [TriDevs.TriEngine2D.UI](#)

## 8.31 TriDevs.TriEngine2D/UI/Control.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.UI.Control](#)  
*Base control class that all other controls inherits from. Defines basic [UI](#) control behaviour.*

### Namespaces

- package [TriDevs.TriEngine2D.UI](#)

## 8.32 TriDevs.TriEngine2D/UI/Events/ControlClicked.cs File Reference

### Namespaces

- package [TriDevs.TriEngine2D.UI.Events](#)

### Functions

- delegate void [TriDevs.TriEngine2D.UI.Events.ControlClickedEventHandler](#) (object sender, EventArgs e)  
*Delegate handler for the control clicked event.*

## 8.33 TriDevs.TriEngine2D/UI/IControl.cs File Reference

### Classes

- interface [TriDevs.TriEngine2D.UI.IControl](#)  
*A [UI](#) control that can be drawn on screen and interacted with.*

### Namespaces

- package [TriDevs.TriEngine2D.UI](#)

## 8.34 TriDevs.TriEngine2D/UI/Label.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.UI.Label](#)  
*A simple label to display text on the screen.*

### Namespaces

- package [TriDevs.TriEngine2D.UI](#)

## 8.35 TriDevs.TriEngine2D/UI/LinkLabel.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.UI.LinkLabel](#)  
*A label that, when clicked, will open a URL.*

### Namespaces

- package [TriDevs.TriEngine2D.UI](#)

## 8.36 TriDevs.TriEngine2D/Version.cs File Reference

### Classes

- class [TriDevs.TriEngine2D.Version](#)  
*[Version](#) class specifying the version of this project.*

### Namespaces

- package [TriDevs.TriEngine2D](#)

# Index

## A

TriDevs::TriEngine2D::UI::Color, [24](#)

## AllocConsole

TriDevs::TriEngine2D::Native::WinAPI, [99](#)

## Audio

TriDevs::TriEngine2D::Services, [90](#)

## AudioFormat

TriDevs::TriEngine2D::Audio, [12](#)

## AudioManager

TriDevs::TriEngine2D::Audio::AudioManager, [19](#)

## B

TriDevs::TriEngine2D::UI::Color, [24](#)

## CODE\_PAGE

TriDevs::TriEngine2D::Native::WinAPI, [100](#)

## Clamp

TriDevs::TriEngine2D::Helpers::Math, [67–69](#)

## ClearOldLogs

TriDevs::TriEngine2D::Logging::LogManager, [63](#)

## Clicked

TriDevs::TriEngine2D::UI::Control, [27](#)

TriDevs::TriEngine2D::UI::IControl, [38](#)

## Color

TriDevs::TriEngine2D::UI::Color, [23](#)

TriDevs::TriEngine2D::UI::Control, [27](#)

TriDevs::TriEngine2D::UI::IControl, [37](#)

## ControlClickedEventHandler

TriDevs::TriEngine2D::UI::Events, [16](#)

## Deserialize< T >

TriDevs::TriEngine2D::Serializing::Serializer, [87](#)

## DestroyConsole

TriDevs::TriEngine2D::Logging::LogManager, [64](#)

## Disable

TriDevs::TriEngine2D::UI::Control, [26](#)

TriDevs::TriEngine2D::UI::IControl, [37](#)

## Dispose

TriDevs::TriEngine2D::Audio::AudioManager, [19](#)

TriDevs::TriEngine2D::Audio::NullAudioManager, [72](#)

TriDevs::TriEngine2D::Audio::NullSong, [83](#)

TriDevs::TriEngine2D::Audio::NullSound, [85](#)

TriDevs::TriEngine2D::Audio::Song, [92](#)

TriDevs::TriEngine2D::Audio::Sound, [94](#)

## Draw

TriDevs::TriEngine2D::Interfaces::IDrawable, [38](#)

## Enable

TriDevs::TriEngine2D::UI::Control, [26](#)

TriDevs::TriEngine2D::UI::IControl, [37](#)

## Enabled

TriDevs::TriEngine2D::UI::Control, [27](#)

TriDevs::TriEngine2D::UI::IControl, [37](#)

## File

TriDevs::TriEngine2D::Audio::ISong, [55](#)

TriDevs::TriEngine2D::Audio::ISound, [57](#)

TriDevs::TriEngine2D::Audio::NullSong, [84](#)

TriDevs::TriEngine2D::Audio::NullSound, [86](#)

TriDevs::TriEngine2D::Audio::Song, [93](#)

TriDevs::TriEngine2D::Audio::Sound, [95](#)

## FreeConsole

TriDevs::TriEngine2D::Native::WinAPI, [99](#)

## G

TriDevs::TriEngine2D::UI::Color, [24](#)

## GetAbsolutePath

TriDevs::TriEngine2D::Helpers::IO, [52](#)

## GetErrorMessage

TriDevs::TriEngine2D::Native::Helpers, [30](#)

## GetLastError

TriDevs::TriEngine2D::Native::Helpers, [30](#)

## GetLastErrorInfo

TriDevs::TriEngine2D::Native::Helpers, [31](#)

## GetLastErrorMessage

TriDevs::TriEngine2D::Native::Helpers, [31](#)

## GetLastWin32Exception

TriDevs::TriEngine2D::Native::Helpers, [31](#)

## GetLogger

TriDevs::TriEngine2D::Logging::LogManager, [64](#)

## GetSong

TriDevs::TriEngine2D::Audio::AudioManager, [19](#)

TriDevs::TriEngine2D::Audio::IAudioManager, [33](#)

TriDevs::TriEngine2D::Audio::NullAudioManager, [72](#)

## GetSound

TriDevs::TriEngine2D::Audio::AudioManager, [19](#)

TriDevs::TriEngine2D::Audio::IAudioManager, [33](#)

TriDevs::TriEngine2D::Audio::NullAudioManager, [72](#)

## GetStdHandle

TriDevs::TriEngine2D::Native::WinAPI, [100](#)

## GetWin32Exception

TriDevs::TriEngine2D::Native::Helpers, [31](#)

## Has< T >

TriDevs::TriEngine2D::Extensions::Enumeration-  
Extensions, [28](#)

## HasSong

- TriDevs::TriEngine2D::Audio::AudioManager, 20
- TriDevs::TriEngine2D::Audio::IAudioManager, 34
- TriDevs::TriEngine2D::Audio::NullAudioManager, 72
- HasSound
  - TriDevs::TriEngine2D::Audio::AudioManager, 20
  - TriDevs::TriEngine2D::Audio::IAudioManager, 34
  - TriDevs::TriEngine2D::Audio::NullAudioManager, 73
- Hide
  - TriDevs::TriEngine2D::UI::Control, 26
  - TriDevs::TriEngine2D::UI::IControl, 37
- Include< T >
  - TriDevs::TriEngine2D::Extensions::Enumeration-Extensions, 28
- Input
  - TriDevs::TriEngine2D::Services, 90
- InputManager
  - TriDevs::TriEngine2D::Input::InputManager, 47
- IsKeyDown
  - TriDevs::TriEngine2D::Input::IInputManager, 40
  - TriDevs::TriEngine2D::Input::InputManager, 47
  - TriDevs::TriEngine2D::Input::NullInputManager, 76
- IsKeyUp
  - TriDevs::TriEngine2D::Input::IInputManager, 40
  - TriDevs::TriEngine2D::Input::InputManager, 47
  - TriDevs::TriEngine2D::Input::NullInputManager, 77
- IsLooped
  - TriDevs::TriEngine2D::Audio::ISong, 55
  - TriDevs::TriEngine2D::Audio::NullSong, 84
  - TriDevs::TriEngine2D::Audio::Song, 93
- IsMouseDown
  - TriDevs::TriEngine2D::Input::IInputManager, 41
  - TriDevs::TriEngine2D::Input::InputManager, 48
  - TriDevs::TriEngine2D::Input::NullInputManager, 77
- IsMouseUp
  - TriDevs::TriEngine2D::Input::IInputManager, 41
  - TriDevs::TriEngine2D::Input::InputManager, 48
  - TriDevs::TriEngine2D::Input::NullInputManager, 77
- IsWheelChanged
  - TriDevs::TriEngine2D::Input::IInputManager, 41
  - TriDevs::TriEngine2D::Input::InputManager, 48
  - TriDevs::TriEngine2D::Input::NullInputManager, 78
- IsWheelDown
  - TriDevs::TriEngine2D::Input::IInputManager, 41
  - TriDevs::TriEngine2D::Input::InputManager, 49
  - TriDevs::TriEngine2D::Input::NullInputManager, 78
- IsWheelUp
  - TriDevs::TriEngine2D::Input::IInputManager, 41
  - TriDevs::TriEngine2D::Input::InputManager, 49
  - TriDevs::TriEngine2D::Input::NullInputManager, 78
- Key
  - TriDevs::TriEngine2D::Input::Events::KeyEvent-Args, 59
- KeyChar
  - TriDevs::TriEngine2D::Input::Events::KeyChar-EventArgs, 58
- KeyDown
  - TriDevs::TriEngine2D::Input::IInputManager, 44
  - TriDevs::TriEngine2D::Input::InputManager, 51
  - TriDevs::TriEngine2D::Input::NullInputManager, 80
- KeyDownEventHandler
  - TriDevs::TriEngine2D::Input::Events, 14
- KeyPress
  - TriDevs::TriEngine2D::Input::IInputManager, 44
  - TriDevs::TriEngine2D::Input::InputManager, 51
  - TriDevs::TriEngine2D::Input::NullInputManager, 80
- KeyPressEventHandler
  - TriDevs::TriEngine2D::Input::Events, 14
- KeyPressed
  - TriDevs::TriEngine2D::Input::IInputManager, 42
  - TriDevs::TriEngine2D::Input::InputManager, 49
  - TriDevs::TriEngine2D::Input::NullInputManager, 78
- KeyReleased
  - TriDevs::TriEngine2D::Input::IInputManager, 42
  - TriDevs::TriEngine2D::Input::InputManager, 49
  - TriDevs::TriEngine2D::Input::NullInputManager, 79
- KeyUp
  - TriDevs::TriEngine2D::Input::IInputManager, 44
  - TriDevs::TriEngine2D::Input::InputManager, 51
  - TriDevs::TriEngine2D::Input::NullInputManager, 81
- KeyUpEventHandler
  - TriDevs::TriEngine2D::Input::Events, 14
- LoadConfig
  - TriDevs::TriEngine2D::Logging::LogManager, 65
- LoadSong
  - TriDevs::TriEngine2D::Audio::AudioManager, 20
  - TriDevs::TriEngine2D::Audio::IAudioManager, 34
  - TriDevs::TriEngine2D::Audio::NullAudioManager, 73
- LoadSound
  - TriDevs::TriEngine2D::Audio::AudioManager, 21
  - TriDevs::TriEngine2D::Audio::IAudioManager, 34
  - TriDevs::TriEngine2D::Audio::NullAudioManager, 73
- Major
  - TriDevs::TriEngine2D::Version, 98
- Minor
  - TriDevs::TriEngine2D::Version, 98
- Missing< T >
  - TriDevs::TriEngine2D::Extensions::Enumeration-Extensions, 29
- MouseDown
  - TriDevs::TriEngine2D::Input::IInputManager, 44
  - TriDevs::TriEngine2D::Input::InputManager, 51
  - TriDevs::TriEngine2D::Input::NullInputManager, 81
- MouseDownEventHandler
  - TriDevs::TriEngine2D::Input::Events, 14
- MousePosition
  - TriDevs::TriEngine2D::Input::IInputManager, 43
  - TriDevs::TriEngine2D::Input::InputManager, 51
  - TriDevs::TriEngine2D::Input::NullInputManager, 80
- MousePressed
  - TriDevs::TriEngine2D::Input::IInputManager, 42



- TriDevs::TriEngine2D::Input::InputManager, 50
- TriDevs::TriEngine2D::Input::NullInputManager, 79
- MouseReleased
  - TriDevs::TriEngine2D::Input::InputManager, 42
  - TriDevs::TriEngine2D::Input::InputManager, 50
  - TriDevs::TriEngine2D::Input::NullInputManager, 79
- MouseUp
  - TriDevs::TriEngine2D::Input::InputManager, 44
  - TriDevs::TriEngine2D::Input::InputManager, 52
  - TriDevs::TriEngine2D::Input::NullInputManager, 81
- MouseUpEventHandler
  - TriDevs::TriEngine2D::Input::Events, 14
- MouseWheelChangedEventHandler
  - TriDevs::TriEngine2D::Input::Events, 14
- MouseWheelDownEventHandler
  - TriDevs::TriEngine2D::Input::Events, 15
- MouseWheelUpEventHandler
  - TriDevs::TriEngine2D::Input::Events, 15
- MouseWheelValue
  - TriDevs::TriEngine2D::Input::InputManager, 43
  - TriDevs::TriEngine2D::Input::InputManager, 51
  - TriDevs::TriEngine2D::Input::NullInputManager, 80
- MouseX
  - TriDevs::TriEngine2D::Input::InputManager, 43
  - TriDevs::TriEngine2D::Input::InputManager, 51
  - TriDevs::TriEngine2D::Input::NullInputManager, 80
- MouseY
  - TriDevs::TriEngine2D::Input::InputManager, 43
  - TriDevs::TriEngine2D::Input::InputManager, 51
  - TriDevs::TriEngine2D::Input::NullInputManager, 80
- Name
  - TriDevs::TriEngine2D::Audio::ISong, 55
  - TriDevs::TriEngine2D::Audio::ISound, 57
  - TriDevs::TriEngine2D::Audio::NullSong, 84
  - TriDevs::TriEngine2D::Audio::NullSound, 86
  - TriDevs::TriEngine2D::Audio::Song, 93
  - TriDevs::TriEngine2D::Audio::Sound, 95
- Ogg
  - TriDevs::TriEngine2D::Audio, 12
- OnClicked
  - TriDevs::TriEngine2D::UI::Control, 26
  - TriDevs::TriEngine2D::UI::LinkLabel, 62
- Patch
  - TriDevs::TriEngine2D::Version, 98
- Pause
  - TriDevs::TriEngine2D::Audio::ISong, 54
  - TriDevs::TriEngine2D::Audio::NullSong, 83
  - TriDevs::TriEngine2D::Audio::Song, 92
- Play
  - TriDevs::TriEngine2D::Audio::ISong, 54
  - TriDevs::TriEngine2D::Audio::ISound, 56
  - TriDevs::TriEngine2D::Audio::NullSong, 83
  - TriDevs::TriEngine2D::Audio::NullSound, 85
  - TriDevs::TriEngine2D::Audio::Song, 92
  - TriDevs::TriEngine2D::Audio::Sound, 94
- Point
  - TriDevs::TriEngine2D::Point< T >, 87
- Position
  - TriDevs::TriEngine2D::UI::Control, 27
  - TriDevs::TriEngine2D::UI::IControl, 37
- Provide
  - TriDevs::TriEngine2D::Services, 89, 90
- R
  - TriDevs::TriEngine2D::UI::Color, 24
- README.md, 103
- Remove< T >
  - TriDevs::TriEngine2D::Extensions::Enumeration-Extensions, 29
- Replace
  - TriDevs::TriEngine2D::Extensions::StringExtensions, 95, 96
- ReplaceFirst
  - TriDevs::TriEngine2D::Extensions::StringExtensions, 96
- Resume
  - TriDevs::TriEngine2D::Audio::ISong, 54
  - TriDevs::TriEngine2D::Audio::NullSong, 83
  - TriDevs::TriEngine2D::Audio::Song, 92
- Serialize< T >
  - TriDevs::TriEngine2D::Serializing::Serializer, 88
- SetCurrentThreadName
  - TriDevs::TriEngine2D::Helpers::Threading, 97
- SetupConsole
  - TriDevs::TriEngine2D::Logging::LogManager, 65
- Show
  - TriDevs::TriEngine2D::UI::Control, 26
  - TriDevs::TriEngine2D::UI::IControl, 37
- Size
  - TriDevs::TriEngine2D::UI::Control, 27
  - TriDevs::TriEngine2D::UI::IControl, 37
- Stop
  - TriDevs::TriEngine2D::Audio::ISong, 54
  - TriDevs::TriEngine2D::Audio::ISound, 56
  - TriDevs::TriEngine2D::Audio::NullSong, 83
  - TriDevs::TriEngine2D::Audio::NullSound, 86
  - TriDevs::TriEngine2D::Audio::Song, 93
  - TriDevs::TriEngine2D::Audio::Sound, 95
- StopAll
  - TriDevs::TriEngine2D::Audio::AudioManager, 21
  - TriDevs::TriEngine2D::Audio::IAudioManager, 35
  - TriDevs::TriEngine2D::Audio::NullAudioManager, 74
- StopAllSongs
  - TriDevs::TriEngine2D::Audio::AudioManager, 21
  - TriDevs::TriEngine2D::Audio::IAudioManager, 35
  - TriDevs::TriEngine2D::Audio::NullAudioManager, 74
- StopAllSounds
  - TriDevs::TriEngine2D::Audio::AudioManager, 22
  - TriDevs::TriEngine2D::Audio::IAudioManager, 35
  - TriDevs::TriEngine2D::Audio::NullAudioManager, 74
- Suffix

- TriDevs::TriEngine2D::Version, 98
- Text
  - TriDevs::TriEngine2D::UI::Control, 27
  - TriDevs::TriEngine2D::UI::IControl, 37
  - TriDevs::TriEngine2D::UI::Label, 61
- ToVector3
  - TriDevs::TriEngine2D::UI::Color, 23
- ToVector4
  - TriDevs::TriEngine2D::UI::Color, 23
- TriDevs::TriEngine2D::Audio
  - Ogg, 12
  - Wav, 12
- TriDevs, 11
- TriDevs.TriEngine2D, 11
- TriDevs.TriEngine2D.Audio, 11
- TriDevs.TriEngine2D.Audio.AudioManager, 17
- TriDevs.TriEngine2D.Audio.IAudioManager, 32
- TriDevs.TriEngine2D.Audio.ISong, 53
- TriDevs.TriEngine2D.Audio.ISound, 55
- TriDevs.TriEngine2D.Audio.NullAudioManager, 70
- TriDevs.TriEngine2D.Audio.NullSong, 81
- TriDevs.TriEngine2D.Audio.NullSound, 84
- TriDevs.TriEngine2D.Audio.Song, 90
- TriDevs.TriEngine2D.Audio.Sound, 93
- TriDevs.TriEngine2D.Extensions, 12
- TriDevs.TriEngine2D.Extensions.EnumerationExtensions, 27
- TriDevs.TriEngine2D.Extensions.StringExtensions, 95
- TriDevs.TriEngine2D.Helpers, 13
- TriDevs.TriEngine2D.Helpers.IO, 52
- TriDevs.TriEngine2D.Helpers.Math, 66
- TriDevs.TriEngine2D.Helpers.Threading, 97
- TriDevs.TriEngine2D.Input, 13
- TriDevs.TriEngine2D.Input.Events, 13
- TriDevs.TriEngine2D.Input.Events.KeyCharEventArgs, 57
- TriDevs.TriEngine2D.Input.Events.KeyEventArgs, 58
- TriDevs.TriEngine2D.Input.IInputManager, 38
- TriDevs.TriEngine2D.Input.InputManager, 45
- TriDevs.TriEngine2D.Input.NullInputManager, 74
- TriDevs.TriEngine2D.Interfaces, 15
- TriDevs.TriEngine2D.Interfaces.IDrawable, 38
- TriDevs.TriEngine2D.Logging, 15
- TriDevs.TriEngine2D.Logging.LogManager, 63
- TriDevs.TriEngine2D.Native, 15
- TriDevs.TriEngine2D.Native.Helpers, 30
- TriDevs.TriEngine2D.Native.WinAPI, 99
- TriDevs.TriEngine2D.Point< T >, 86
- TriDevs.TriEngine2D.Serializing, 16
- TriDevs.TriEngine2D.Serializing.Serializer, 87
- TriDevs.TriEngine2D.Services, 89
- TriDevs.TriEngine2D.UI, 16
- TriDevs.TriEngine2D.UI.Color, 22
- TriDevs.TriEngine2D.UI.Control, 24
- TriDevs.TriEngine2D.UI.Events, 16
- TriDevs.TriEngine2D.UI.IControl, 35
- TriDevs.TriEngine2D.UI.Label, 59
- TriDevs.TriEngine2D.UI.LinkLabel, 61
- TriDevs.TriEngine2D.Version, 97
- TriDevs.TriEngine2D.Audio.AudioFormat.cs, 103
- TriDevs.TriEngine2D.Audio.AudioManager.cs, 103
- TriDevs.TriEngine2D.Audio.IAudioManager.cs, 103
- TriDevs.TriEngine2D.Audio.ISong.cs, 104
- TriDevs.TriEngine2D.Audio.ISound.cs, 104
- TriDevs.TriEngine2D.Audio.NullAudioManager.cs, 104
- TriDevs.TriEngine2D.Audio.NullSong.cs, 104
- TriDevs.TriEngine2D.Audio.NullSound.cs, 105
- TriDevs.TriEngine2D.Audio.Song.cs, 105
- TriDevs.TriEngine2D.Audio.Sound.cs, 105
- TriDevs.TriEngine2D/Extensions/EnumerationExtensions.-cs, 105
- TriDevs.TriEngine2D/Extensions/StringExtensions.cs, 106
- TriDevs.TriEngine2D/Helpers/IO.cs, 106
- TriDevs.TriEngine2D/Helpers/Math.cs, 106
- TriDevs.TriEngine2D/Helpers/Threading.cs, 106
- TriDevs.TriEngine2D/Input/Events/Key.cs, 107
- TriDevs.TriEngine2D/Input/Events/Mouse.cs, 107
- TriDevs.TriEngine2D/Input/IInputManager.cs, 108
- TriDevs.TriEngine2D/Input/InputManager.cs, 108
- TriDevs.TriEngine2D/Input/NullInputManager.cs, 108
- TriDevs.TriEngine2D/Interfaces/IDrawable.cs, 108
- TriDevs.TriEngine2D/Logging/LogManager.cs, 109
- TriDevs.TriEngine2D/Native/Helpers.cs, 109
- TriDevs.TriEngine2D/Native/WinAPI.cs, 109
- TriDevs.TriEngine2D/Point.cs, 109
- TriDevs.TriEngine2D/Properties/AssemblyInfo.cs, 110
- TriDevs.TriEngine2D/Serializing/Serializer.cs, 110
- TriDevs.TriEngine2D/Services.cs, 110
- TriDevs.TriEngine2D/UI/Color.cs, 110
- TriDevs.TriEngine2D/UI/Control.cs, 110
- TriDevs.TriEngine2D/UI/Events/ControlClicked.cs, 111
- TriDevs.TriEngine2D/UI/IControl.cs, 111
- TriDevs.TriEngine2D/UI/Label.cs, 111
- TriDevs.TriEngine2D/UI/LinkLabel.cs, 111
- TriDevs.TriEngine2D/Version.cs, 112
- TriDevs::TriEngine2D::Audio
  - AudioFormat, 12
- TriDevs::TriEngine2D::Audio::AudioManager
  - AudioManager, 19
  - Dispose, 19
  - GetSong, 19
  - GetSound, 19
  - HasSong, 20
  - HasSound, 20
  - LoadSong, 20
  - LoadSound, 21
  - StopAll, 21
  - StopAllSongs, 21
  - StopAllSounds, 22
- TriDevs::TriEngine2D::Audio::IAudioManager
  - GetSong, 33
  - GetSound, 33
  - HasSong, 34
  - HasSound, 34
  - LoadSong, 34

- LoadSound, [34](#)
- StopAll, [35](#)
- StopAllSongs, [35](#)
- StopAllSounds, [35](#)
- TriDevs::TriEngine2D::Audio::ISong
  - File, [55](#)
  - IsLooped, [55](#)
  - Name, [55](#)
  - Pause, [54](#)
  - Play, [54](#)
  - Resume, [54](#)
  - Stop, [54](#)
  - Volume, [55](#)
- TriDevs::TriEngine2D::Audio::ISound
  - File, [57](#)
  - Name, [57](#)
  - Play, [56](#)
  - Stop, [56](#)
- TriDevs::TriEngine2D::Audio::NullAudioManager
  - Dispose, [72](#)
  - GetSong, [72](#)
  - GetSound, [72](#)
  - HasSong, [72](#)
  - HasSound, [73](#)
  - LoadSong, [73](#)
  - LoadSound, [73](#)
  - StopAll, [74](#)
  - StopAllSongs, [74](#)
  - StopAllSounds, [74](#)
- TriDevs::TriEngine2D::Audio::NullSong
  - Dispose, [83](#)
  - File, [84](#)
  - IsLooped, [84](#)
  - Name, [84](#)
  - Pause, [83](#)
  - Play, [83](#)
  - Resume, [83](#)
  - Stop, [83](#)
  - Volume, [84](#)
- TriDevs::TriEngine2D::Audio::NullSound
  - Dispose, [85](#)
  - File, [86](#)
  - Name, [86](#)
  - Play, [85](#)
  - Stop, [86](#)
- TriDevs::TriEngine2D::Audio::Song
  - Dispose, [92](#)
  - File, [93](#)
  - IsLooped, [93](#)
  - Name, [93](#)
  - Pause, [92](#)
  - Play, [92](#)
  - Resume, [92](#)
  - Stop, [93](#)
  - Volume, [93](#)
- TriDevs::TriEngine2D::Audio::Sound
  - Dispose, [94](#)
  - File, [95](#)
  - Name, [95](#)
  - Play, [94](#)
  - Stop, [95](#)
- TriDevs::TriEngine2D::Extensions::EnumerationExtensions
  - Has< T >, [28](#)
  - Include< T >, [28](#)
  - Missing< T >, [29](#)
  - Remove< T >, [29](#)
- TriDevs::TriEngine2D::Extensions::StringExtensions
  - Replace, [95](#), [96](#)
  - ReplaceFirst, [96](#)
- TriDevs::TriEngine2D::Helpers::IO
  - GetAbsolutePath, [52](#)
- TriDevs::TriEngine2D::Helpers::Math
  - Clamp, [67](#)–[69](#)
- TriDevs::TriEngine2D::Helpers::Threading
  - SetCurrentThreadName, [97](#)
- TriDevs::TriEngine2D::Input::Events
  - KeyDownEventHandler, [14](#)
  - KeyPressEventHandler, [14](#)
  - KeyUpEventHandler, [14](#)
  - MouseDownEventHandler, [14](#)
  - MouseUpEventHandler, [14](#)
  - MouseWheelChangedEventHandler, [14](#)
  - MouseWheelDownEventHandler, [15](#)
  - MouseWheelUpEventHandler, [15](#)
- TriDevs::TriEngine2D::Input::Events::KeyCharEventArgs
  - KeyChar, [58](#)
- TriDevs::TriEngine2D::Input::Events::KeyEventArgs
  - Key, [59](#)
- TriDevs::TriEngine2D::Input::InputManager
  - IsKeyDown, [40](#)
  - IsKeyUp, [40](#)
  - IsMouseDown, [41](#)
  - IsMouseUp, [41](#)
  - IsWheelChanged, [41](#)
  - IsWheelDown, [41](#)
  - IsWheelUp, [41](#)
  - KeyDown, [44](#)
  - KeyPress, [44](#)
  - KeyPressed, [42](#)
  - KeyReleased, [42](#)
  - KeyUp, [44](#)
  - MouseDown, [44](#)
  - MousePosition, [43](#)
  - MousePressed, [42](#)
  - MouseReleased, [42](#)
  - MouseUp, [44](#)
  - MouseWheelValue, [43](#)
  - MouseX, [43](#)
  - MouseY, [43](#)
  - Update, [43](#)
  - WheelChange, [43](#)
  - WheelChanged, [44](#)
  - WheelDown, [44](#)
  - WheelUp, [44](#)
- TriDevs::TriEngine2D::Input::InputManager
  - InputManager, [47](#)

- IsKeyDown, [47](#)
- IsKeyUp, [47](#)
- IsMouseDown, [48](#)
- IsMouseUp, [48](#)
- IsWheelChanged, [48](#)
- IsWheelDown, [49](#)
- IsWheelUp, [49](#)
- KeyDown, [51](#)
- KeyPress, [51](#)
- KeyPressed, [49](#)
- KeyReleased, [49](#)
- KeyUp, [51](#)
- MouseDown, [51](#)
- MousePosition, [51](#)
- MousePressed, [50](#)
- MouseReleased, [50](#)
- MouseUp, [52](#)
- MouseWheelValue, [51](#)
- MouseX, [51](#)
- MouseY, [51](#)
- Update, [50](#)
- WheelChange, [51](#)
- WheelChanged, [52](#)
- WheelDown, [52](#)
- WheelUp, [52](#)
- TriDevs::TriEngine2D::Input::NullInputManager
  - IsKeyDown, [76](#)
  - IsKeyUp, [77](#)
  - IsMouseDown, [77](#)
  - IsMouseUp, [77](#)
  - IsWheelChanged, [78](#)
  - IsWheelDown, [78](#)
  - IsWheelUp, [78](#)
  - KeyDown, [80](#)
  - KeyPress, [80](#)
  - KeyPressed, [78](#)
  - KeyReleased, [79](#)
  - KeyUp, [81](#)
  - MouseDown, [81](#)
  - MousePosition, [80](#)
  - MousePressed, [79](#)
  - MouseReleased, [79](#)
  - MouseUp, [81](#)
  - MouseWheelValue, [80](#)
  - MouseX, [80](#)
  - MouseY, [80](#)
  - Update, [80](#)
  - WheelChange, [80](#)
  - WheelChanged, [81](#)
  - WheelDown, [81](#)
  - WheelUp, [81](#)
- TriDevs::TriEngine2D::Interfaces::IDrawable
  - Draw, [38](#)
- TriDevs::TriEngine2D::Logging::LogManager
  - ClearOldLogs, [63](#)
  - DestroyConsole, [64](#)
  - GetLogger, [64](#)
  - LoadConfig, [65](#)
  - SetupConsole, [65](#)
- TriDevs::TriEngine2D::Native::Helpers
  - GetErrorMessage, [30](#)
  - GetLastError, [30](#)
  - GetLastErrorInfo, [31](#)
  - GetLastErrorMessage, [31](#)
  - GetLastWin32Exception, [31](#)
  - GetWin32Exception, [31](#)
- TriDevs::TriEngine2D::Native::WinAPI
  - AllocConsole, [99](#)
  - FreeConsole, [99](#)
  - GetStdHandle, [100](#)
- TriDevs::TriEngine2D::Point< T >
  - Point, [87](#)
  - X, [87](#)
  - Y, [87](#)
- TriDevs::TriEngine2D::Serializing::Serializer
  - Deserialize< T >, [87](#)
  - Serialize< T >, [88](#)
- TriDevs::TriEngine2D::Services
  - Audio, [90](#)
  - Input, [90](#)
  - Provide, [89, 90](#)
- TriDevs::TriEngine2D::UI::Color
  - A, [24](#)
  - B, [24](#)
  - Color, [23](#)
  - G, [24](#)
  - R, [24](#)
  - ToVector3, [23](#)
  - ToVector4, [23](#)
- TriDevs::TriEngine2D::UI::Control
  - Clicked, [27](#)
  - Color, [27](#)
  - Disable, [26](#)
  - Enable, [26](#)
  - Enabled, [27](#)
  - Hide, [26](#)
  - OnClicked, [26](#)
  - Position, [27](#)
  - Show, [26](#)
  - Size, [27](#)
  - Text, [27](#)
  - Visible, [27](#)
- TriDevs::TriEngine2D::UI::Events
  - ControlClickedEventHandler, [16](#)
- TriDevs::TriEngine2D::UI::IControl
  - Clicked, [38](#)
  - Color, [37](#)
  - Disable, [37](#)
  - Enable, [37](#)
  - Enabled, [37](#)
  - Hide, [37](#)
  - Position, [37](#)
  - Show, [37](#)
  - Size, [37](#)
  - Text, [37](#)
  - Visible, [38](#)

- TriDevs::TriEngine2D::UI::Label
  - Text, [61](#)
- TriDevs::TriEngine2D::UI::LinkLabel
  - OnClicked, [62](#)
  - Url, [63](#)
- TriDevs::TriEngine2D::Version
  - Major, [98](#)
  - Minor, [98](#)
  - Patch, [98](#)
  - Suffix, [98](#)
  - VersionString, [98](#)
  - VersionStringFormat, [98](#)
  - VersionStringFormatWithSuffix, [98](#)
- Update
  - TriDevs::TriEngine2D::Input::IInputManager, [43](#)
  - TriDevs::TriEngine2D::Input::InputManager, [50](#)
  - TriDevs::TriEngine2D::Input::NullInputManager, [80](#)
- Url
  - TriDevs::TriEngine2D::UI::LinkLabel, [63](#)
- VersionString
  - TriDevs::TriEngine2D::Version, [98](#)
- VersionStringFormat
  - TriDevs::TriEngine2D::Version, [98](#)
- VersionStringFormatWithSuffix
  - TriDevs::TriEngine2D::Version, [98](#)
- Visible
  - TriDevs::TriEngine2D::UI::Control, [27](#)
  - TriDevs::TriEngine2D::UI::IControl, [38](#)
- Volume
  - TriDevs::TriEngine2D::Audio::ISong, [55](#)
  - TriDevs::TriEngine2D::Audio::NullSong, [84](#)
  - TriDevs::TriEngine2D::Audio::Song, [93](#)
- Wav
  - TriDevs::TriEngine2D::Audio, [12](#)
- WheelChange
  - TriDevs::TriEngine2D::Input::IInputManager, [43](#)
  - TriDevs::TriEngine2D::Input::InputManager, [51](#)
  - TriDevs::TriEngine2D::Input::NullInputManager, [80](#)
- WheelChanged
  - TriDevs::TriEngine2D::Input::IInputManager, [44](#)
  - TriDevs::TriEngine2D::Input::InputManager, [52](#)
  - TriDevs::TriEngine2D::Input::NullInputManager, [81](#)
- WheelDown
  - TriDevs::TriEngine2D::Input::IInputManager, [44](#)
  - TriDevs::TriEngine2D::Input::InputManager, [52](#)
  - TriDevs::TriEngine2D::Input::NullInputManager, [81](#)
- WheelUp
  - TriDevs::TriEngine2D::Input::IInputManager, [44](#)
  - TriDevs::TriEngine2D::Input::InputManager, [52](#)
  - TriDevs::TriEngine2D::Input::NullInputManager, [81](#)
- X
  - TriDevs::TriEngine2D::Point< T >, [87](#)
- Y
  - TriDevs::TriEngine2D::Point< T >, [87](#)