Namespace Coree.NETStandard.Abstractions

Enter here to filter...

Classes

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.Dependen 1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDepender
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBas
- + Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- + Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- + Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- + Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- + Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- + Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- + Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- + Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

DependencySingleton<T> (Coree.NETStandard.Abstractions.DependencySingleton-1.html)

Represents an abstract base class for creating thread-safe singleton instances with optional dependency injection support. This pattern can be utilized in both dependency injection (DI) and non-DI scenarios.

Example usage:

```
//File ISampleSerivce.cs
public interface ISampleService
{
    Task StandardDelayAsync(CancellationToken cancellationToken = default);  void Standardelay();
}

//File DSSampleSerivce.cs
public partial class SampleService : DependencySingleton<SampleService>, ISampleService, IDependencySingleton
{
    public SampleService(ILogger<SampleService> logger, IConfiguration configuration) : base(logger, configuration) { }
}

//File SampleSerivce.cs
public partial class SampleService : DependencySingleton<SampleService>, ISampleService, IDependencySingleton
{
    public void StandardDelay() { StandardDelayAsync(CancellationToken.None).GetAwaiter().GetResult(); }
    public async Task StandardDelayAsync(CancellationToken cancellationToken = default) { await Task.Delay(5000, cancellationToken); }
}

//File Program.cs
static async Task Main(string[] args)
{
    SampleService.Instance.SetMinimumLogLevel(LogLevel.Trace);
    // The ILogger<SampleService> logger and IConfiguration configuration will be initialized with their own service stacks.
    SampleService.Instance.StandardDelay();

    // Normal DI usage
    var host = Host.CreateDefaultBuilder(args)
        .ConfigureLogging((context, logging) => {  logging.AddConsole(); logging.AddDebug(); })
        .ConfigureServices((context, services) => { services.AddSingleton<ISampleService>(); services.AddSingleton<IMyService,MyService>(); });
    await host.Build().RunAsync();
}

//Standard Service
public class MyService
{
    private readonly ISampleService sampleService;

    public MyService(ISampleService sampleService)
    {
        this.sampleService = sampleService;
    }
```

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.DependencySingleton1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBas
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
        public async Task UseSample()
        {
            await sampleService.StandardDelayAsync();
        }
    }
```

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.Dependen 1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependen
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBas
- + Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- + Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- + Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- + Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- + Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- + Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- + Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- + Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

FluentBase (Coree.NETStandard.Abstractions.FluentBase.html)

Acts as an abstract foundation for structured error handling in classes using a fluent interface pattern. Offers mechanisms to either immediately throw or accumulate exceptions, alongside managing continuation after validation errors.

Interfaces

IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)

IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)

Serves as a foundational class for fluent interface pattern error handling, enabling mechanisms for both error accumulation and immediate exception throwing, alongside optional continuation after encountering validation errors. This abstract class standardizes error management for classes utilizing fluent interfaces.

Class DependencySingleton<T>

Represents an abstract base class for creating thread-safe singleton instances with optional dependency injection support. This pattern can be utilized in both dependency injection (DI) and non-DI scenarios. Example usage:

Enter here to filter...

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.Dependen 1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDepender
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBas
- + Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- + Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- + Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- + Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- + Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- + Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- + Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- + Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
//File ISampleSerivce.cs
public interface ISampleService
{
    Task StandardDelayAsync(CancellationToken cancellationToken = default);  void Standar
dDelay();
}

//File DSSampleSerivce.cs
public partial class SampleService : DependencySingleton<SampleService>, ISampleService ,
IDependencySingleton
{
    public SampleService(ILogger<SampleService> logger, IConfiguration configuration) : b
ase(logger, configuration) { }
}
//File SimpleSerivce.cs
public partial class SampleService : DependencySingleton<SampleService>, ISampleService ,
IDependencySingleton
{
    public void StandardDelay() { StandardDelayAsync(CancellationToken.None).GetAwaiter
().GetResult(); }
    public async Task StandardDelayAsync(CancellationToken cancellationToken = default)
{ await Task.Delay(5000, cancellationToken); }
}

//File Program.cs
static async Task Main(string[] args)
{
    SampleService.Instance.SetMinimumLogLevel(LogLevel.Trace);
    // The ILogger<SampleService> logger and IConfiguration configuration will be initiali
zed with their own service stacks.
    SampleService.Instance.StandardDelay();

    // Normal DI usage
    var host = Host.CreateDefaultBuilder(args)
        .ConfigureLogging((context, logging) => {  logging.AddConsole(); logging.AddDebug
(); })
        .ConfigureServices((context, services) => { services.AddSingleton<ISampleService,S
ampleService>(); services.AddSingleton<IMyService,MyService>(); });
    await host.Build().RunAsync();
}

//Standard Sage
public class MyService
{
    private readonly ISampleService sampleService;

    public MyService(ISampleService sampleService)
    {
        this.sampleService = sampleService;
    }
```

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton (Coree.NETStandard.Abstractions.DependencySingleton1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase)
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton)
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBase)
- + Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- + Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- + Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- + Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- + Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- + Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- + Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- + Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
        public async Task UseSample()
        {
            await sampleService.StandardDelayAsync();
        }
    }
```

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ DependencySingleton<T>

↳ RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsights.RuntimeInsightsService.html)

Implements

IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)

IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Abstractions (Coree.NETStandard.Abstractions.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public abstract class DependencySingleton<T> : IDependencySingleton
```

Type Parameters

Name	Description
T	The type of the singleton instance.

Constructors

DependencySingleton(ILogger<T>, IConfiguration)

Initializes a new instance of the DependencySingleton<T> (Coree.NETStandard.Abstractions.DependencySingleton-1.html) class with the specified logger and configuration services.

Declaration

```
protected DependencySingleton(ILogger<T> logger, IConfiguration configuration)
```

Parameters

Type	Name	Description
ILogger<T> (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.ilogger-1)	*logger*	The logger service.

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T>
  - IDependencyBase (Coree.NETStandard.Abstractions.IDependencyBase.html)
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Name	Description
IConfiguration (https://learn.microsoft.com/dotnet/api/microsoft.extensions.configuration.iconfiguration)	*configuration*	The configuration service.

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.DependencySingleton-1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase.html)
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Fields

configuration

Represents the configuration for the application. This encompasses settings from various configuration sources (e.g., appsettings.json, environment variables) and is used to access DependencySingleton settings such as connection strings, feature toggles, and other configurable aspects of the application.

Declaration

```
protected readonly IConfiguration configuration
```

Field Value

Type	Description
IConfiguration (https://learn.microsoft.com/dotnet/api/microsoft.extensions.configuration.iconfiguration)	

logger

Provides logging capabilities for the singleton instance. By default, the logger's minimum logging level is set to Trace, allowing all log messages to be captured. However, the default filter level applied to log messages is set to Information, meaning that only logs at Information level or higher will be emitted unless otherwise adjusted. The logging level filter can be dynamically changed at runtime using the SetMinimumLogLevel(LogLevel) method to control the verbosity of the logging output.

Declaration

```
protected readonly ILogger<T> logger
```

Field Value

Type	Description
ILogger<T> (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.ilogger-1)	

Properties

Instance

Gets the singleton instance of the class, ensuring thread safety.

Declaration

```
public static T Instance { get; }
```

Property Value

Type	Description
DependencySingleton<T> (Coree.NETStandard.Abstractions.DependencySingleton-1.html)	The singleton instance of type `T`, which is the specific implementation of the `DependencySingleton<T>` (Coree.NETStandard.Abstractions.DependencySingleton-1.html). Type `T` represents the derived singleton class that includes implementations for required services such as logging and configuration. The instance is initialized with these services upon the first request, adhering to the singleton pattern to ensure only one instance is created and shared throughout the application.

Methods

SetMinimumLogLevel(LogLevel)

Sets the minimum log level for the application. This level acts as a filter for the logs that are emitted. Logs below this level will not be emitted.

Declaration

```
public void SetMinimumLogLevel(LogLevel logLevel = LogLevel.Trace)
```

Parameters

Type	Name	Description
LogLevel (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.loglevel)	*logLevel*	The log level to set as the minimum threshold for logging.

Implements

IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.DependencySingleton-1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase.html)
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.Dependen 1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDepende
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBas
- + Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- + Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- + Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- + Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- + Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- + Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- + Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- + Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class FluentBase

Acts as an abstract foundation for structured error handling in classes using a fluent interface pattern. Offers mechanisms to either immediately throw or accumulate exceptions, alongside managing continuation after validation errors.

Inheritance

object (https://learn.microsoft.com/dotnet/api/system.object)
↳ FluentBase
↳ Product (Coree.NETStandard.CoreeHttpClient.Product.html)

Implements

IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Abstractions (Coree.NETStandard.Abstractions.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public abstract class FluentBase : IFluentBase
```

Remarks

Implements the IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html) interface to supply default methods and properties, facilitating the easy integration of a uniform error handling strategy across derived classes. This base class is designed to empower your fluent interfaces with sophisticated error handling capabilities, allowing for the graceful accumulation of errors for later analysis or immediate failure upon error detection to prevent further processing under invalid conditions.

Example usage in a simplified Product class inheriting from FluentBase and an extension method to adjust its price.

Enter here to filter...

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.Dependen 1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDepende
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBas
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
// Inheriting FluentBase in a Product class
public class Product : FluentBase
{
    public string Name { get; set; }
    public decimal Price { get; set; }
}
// Extension method for the Product class to adjust its price
public static class ProductExtensions
{
    public static Product AdjustPrice(this Product product, decimal adjustment)
    {
        // Early exit if the product is invalid and continuation on validation error is not enable
        if (!product.IsValid && !product.ContinueOnValidationError) return product;
        try
        {
            // Attempt to adjust the product's price
            var newPrice = product.Price + adjustment;
            if (newPrice < 0)
            {
                throw new ArgumentOutOfRangeException(nameof(adjustment), "Adjustment results
                in a negative price.");
            }
            product.Price = newPrice;
        }
        catch (Exception ex)
        {
            // Add a validation error in case of an exception
            product.AddValidationError($"Error adjusting price by {adjustment}: {ex.Message}",
            ex);
        }
        return product;
    }
}
```

This exemplifies how `Product`, by inheriting `FluentBase`, utilizes built-in error handling for a price adjustment operation, demonstrating the class's utility in creating fluent interfaces with integrated error management.

Properties

ContinueOnValidationError

Indicates whether operations should continue after a validation error has occurred, based on the current configuration.

Declaration

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.DependencySingleton1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase.html)
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
public bool ContinueOnValidationError { get; }
```

Property Value

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - IDependencySingleton<T> (Coree.NETStandard.Abstractions.DependencySingleton1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase.html)
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)
- + Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- + Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.ExtensionsBool (Coree.NETStandard.ExtensionsBool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)
- + Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- + Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
- + Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- + Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- + Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)
- + Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

IsValid

Indicates whether the current object is in a valid state, based on the absence of validation errors.

Declaration

```
public bool IsValid { get; }
```

Property Value

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	

ThrowOnError

Gets a value indicating whether exceptions are thrown immediately upon encountering a validation error. This property is set through EnableThrowOnError() (Coree.NETStandard.Abstractions.FluentBase.html#Coree_NETStandard_Abstractions_FluentBase_EnableThrowOnError) and DisableThrowOnError(bool) (Coree.NETStandard.Abstractions.FluentBase.html#Coree_NETStandard_Abstractions_FluentBase_DisableThrowOnError_System_Boolean_).methods.

Declaration

```
public bool ThrowOnError { get; }
```

Property Value

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	

ValidationErrors

Gets a collection of exceptions that have been accumulated during operation executions.

Declaration

```
public List<Exception> ValidationErrors { get; }
```

Property Value

Type	Description
List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1)< Exception (https://learn.microsoft.com/dotnet/api/system.exception)>	

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.Dependen
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDepender
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBas
- + Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- + Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- + Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- + Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- + Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- + Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- + Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- + Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Methods

AddValidationError(string, Exception)

Adds a specified validation error to the collection, throwing an exception if configured to do so.

Declaration

```
public IFluentBase AddValidationError(string message, Exception innerException)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*message*	The error message.
Exception (https://learn.microsoft.com/dotnet/api/system.exception)	*innerException*	The exception that represents the error.

DisableThrowOnError(bool)

Configures the class to accumulate validation errors without throwing exceptions immediately, with an option to continue operations after encountering a validation error.

Declaration

```
public IFluentBase DisableThrowOnError(bool continueOnValidationError = false)
```

Parameters

Type	Name	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*continueOnValidationError*	Indicates whether to continue operations after a validation error has occurred.

Returns

Type	Description
IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)	The current instance of FluentBase (Coree.NETStandard.Abstractions.FluentBase.html), enabling fluent configuration.

EnableThrowOnError()

Configures the class to throw exceptions immediately upon encountering a validation error.

Declaration

```
public IFluentBase EnableThrowOnError()
```

Returns

Type	Description
IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)	The current instance of FluentBase (Coree.NETStandard.Abstractions.FluentBase.html), enabling fluent configuration.

Implements

IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.DependencySingleton<T>.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase.html)
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Interface IDependencySingleton

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Abstractions (Coree.NETStandard.Abstractions.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public interface IDependencySingleton
```

Methods

SetMinimumLogLevel(LogLevel)

Declaration

```
void SetMinimumLogLevel(LogLevel logLevel = LogLevel.Trace)
```

Parameters

Type	Name	Description
LogLevel (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.loglevel)	*logLevel*	

Enter here to filter...

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.Dependen 1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDepender
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBas
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Interface IFluentBase

Serves as a foundational class for fluent interface pattern error handling, enabling mechanisms for both error accumulation and immediate exception throwing, alongside optional continuation after encountering validation errors. This abstract class standardizes error management for classes utilizing fluent interfaces.

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Abstractions (Coree.NETStandard.Abstractions.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public interface IFluentBase
```

Remarks

Implements IFluentBase, offering:

- Error accumulation during operations.
- Conditional continuation of operations post-errors.
- Flexible toggling between immediate exception throwing or error accumulation, adapting to various error handling strategies.

Examples

Illustrates the use of FluentBase with a concrete Product class and an extension method for price adjustment with comprehensive error handling:

Enter here to filter...

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependency1.html)
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
public abstract class FluentBase : IFluentBase
{
    private bool throwOnError = true;
    public List<Exception> ValidationErrors { get; private set; } = new List<Exception>();
    public bool IsValid => !ValidationErrors.Any();
    public bool ContinueOnValidationError { get; private set; } = false;

    public void AddValidationError(string message, Exception innerException)
    {
        ValidationErrors.Add(new Exception(message, innerException));
        if (throwOnError)
        {
            throw innerException;
        }
    }

    public IFluentBase EnableThrowOnError()
    {
        throwOnError = true;
        ContinueOnValidationError = false;
        return this;
    }

    public IFluentBase DisableThrowOnError(bool continueOnValidationError = false)
    {
        throwOnError = false;
        ContinueOnValidationError = continueOnValidationError;
        return this;
    }
}

// Simplified Product class inheriting FluentBase
public class Product : FluentBase
{
    public string Name { get; set; }
    public decimal Price { get; set; }
}

// Extension methods for Product
public static class ProductExtensions
{
    public static Product AdjustPrice(this Product product, decimal adjustment)
    {
        if (!product.IsValid && !product.ContinueOnValidationError) return product;

        try
        {
            var newPrice = product.Price + adjustment;
            if (newPrice < 0)
            {
                throw new ArgumentOutOfRangeException(nameof(adjustment), "Adjustment resu
lts in a negative price.");
```

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.Dependen 1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase
  - IDependencySingleton (Coree.NETStandard.Abstractions.Dependen
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBas
- + Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- + Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- + Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- + Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- + Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- + Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- + Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- + Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
            }
            product.Price = newPrice;
        }
        catch (Exception ex)
        {
            product.AddValidationError($"Error adjusting price by {adjustment}: {ex.Messag
e}", ex);
        }
        return product;
    }
}
```

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingleton<T> (Coree.NETStandard.Abstractions.Dependen 1.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDepender
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBas
- + Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- + Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- + Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- + Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- + Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- + Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- + Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights
- + Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Properties

ContinueOnValidationError

Indicates whether operations should continue after a validation error has occurred.

Declaration

```
bool ContinueOnValidationError { get; }
```

Property Value

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	

IsValid

Indicates whether the current object is in a valid state.

Declaration

```
bool IsValid { get; }
```

Property Value

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	

ValidationErrors

A collection of exceptions that have been accumulated during operations.

Declaration

```
List<Exception> ValidationErrors { get; }
```

Property Value

Type	Description
List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) < Exception (https://learn.microsoft.com/dotnet/api/system.exception) >	

Methods

AddValidationError(string, Exception)

Adds a validation error to the collection with a custom message and the exception that occurred.

Declaration

```
void AddValidationError(string message, Exception innerException)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*message*	The error message.
Exception (https://learn.microsoft.com/dotnet/api/system.exception)	*innerException*	The exception that occurred during validation.

DisableThrowOnError(bool)

Disables the behavior of throwing exceptions immediately, allowing operations to accumulate validation errors. Optionally allows continuation after validation errors based on the `continueOnValidationError` parameter.

Declaration

```
IFluentBase DisableThrowOnError(bool continueOnValidationError = false)
```

Parameters

Type	Name	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*continueOnValidationError*	Determines whether to continue operations after a validation error has occurred.

Returns

Type	Description
IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)	The current instance of IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html).

EnableThrowOnError()

Enables the behavior to throw exceptions immediately when a validation error occurs. Resets `continueOnValidationError` to `false`.

Declaration

```
IFluentBase EnableThrowOnError()
```

Returns

Type	Description
IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)	The current instance of IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html).

- Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - DependencySingletonAttribute (Coree.NETStandard.Abstractions.DependencySingletonAttribute.html)
  - FluentBase (Coree.NETStandard.Abstractions.FluentBase.html)
  - IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)
  - IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace Coree.NETStandard.Classes

Classes

Enter here to filter...

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  - ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeColl 1.html)
  - ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValu 1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeCollection-1.html)

Represents a thread-safe collection of objects that can be accessed by multiple threads concurrently.

ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValue-1.html)

Provides a thread-safe wrapper for a value of type `T`.

Class ThreadSafeCollection<T>

Represents a thread-safe collection of objects that can be accessed by multiple threads concurrently.

Enter here to filter...

Inheritance

object (https://learn.microsoft.com/dotnet/api/system.object)
↳ ThreadSafeCollection<T>

Implements

IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<T>
IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.ienumerable)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Classes (Coree.NETStandard.Classes.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class ThreadSafeCollection<T> : IEnumerable<T?>, IEnumerable where T : class?
```

Type Parameters

Name	Description
T	The type of elements in the collection. This type can be a class, including nullable reference types.

Remarks

This collection uses locking to ensure that its operations are thread-safe. The collection is implemented as a list where items can be added, removed, or retrieved in a manner that prevents race conditions and data corruption when accessed from multiple threads. Null values can be stored, depending on the type `T`.

Constructors

ThreadSafeCollection()

Initializes a new instance of the ThreadSafeCollection class that is empty.

Declaration

```
public ThreadSafeCollection()
```

Remarks

Creates an empty collection with no items. The collection is ready to be accessed by multiple threads using its thread-safe methods. It supports storing null values, if the type `T` permits.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  - ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeCollection1.html)
  - ThreadSafeValueT (Coree.NETStandard.Classes.ThreadSafeValu1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Properties

Count

Gets the number of items in the collection, ensuring thread safety.

Declaration

```
public int Count { get; }
```

Property Value

Type	Description
int (https://learn.microsoft.com/dotnet/api/system.int32)	The number of items currently in the collection. Access is synchronized to prevent race conditions.

HasItems

Gets a value indicating whether the collection contains any items, ensuring thread safety.

Declaration

```
public bool HasItems { get; }
```

Property Value

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	`true` if the collection contains one or more items; otherwise, `false`.

Remarks

This property locks the collection to ensure that the check is performed in a thread-safe manner, preventing race conditions that could arise from concurrent modifications to the collection.

this[int]

Gets or sets the item at the specified index in a thread-safe manner.

Declaration

```
public T? this[int index] { get; set; }
```

Parameters

Type	Name	Description
int (https://learn.microsoft.com/dotnet/api/system.int32)	*index*	The zero-based index of the item to get or set.

Property Value

Type	Description
T	The item at the specified index.

Remarks

Access to the item is synchronized to ensure thread safety. Setting an item acquires a lock to prevent race conditions.

Exceptions

Type	Condition
ArgumentOutOfRangeException (https://learn.microsoft.com/dotnet/api/system.argumentoutofrangeexception)	Thrown when the index is outside the bounds of the collection.

Methods

Add(T?)

Adds an item to the collection in a thread-safe manner.

Declaration

```
public void Add(T? item)
```

Parameters

Type	Name	Description
T	*item*	The item to be added to the collection. Can be null if the type `T` permits.

Remarks

Synchronizes access to the collection to safely add the item, supporting concurrent modifications.

Dequeue()

Removes and returns the item at the beginning of the collection in a thread-safe manner.

Declaration

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  - ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeCollection1.html)
  - ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValue1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
public T? Dequeue()
```

Returns

Type	Description
T	The item at the beginning of the collection if it exists; otherwise, throws an InvalidOperationException.

Remarks

This method locks the collection to ensure thread safety during the remove operation. It is part of the thread-safe queue functionality, providing a way to dequeue items (remove from the beginning). If the collection is empty, it throws an InvalidOperationException to enforce handling of the empty state condition by the caller.

Exceptions

Type	Condition
InvalidOperationException (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception)	Thrown when attempting to perform the Dequeue operation on an empty collection.

Enqueue(T?)

Adds an item to the end of the collection in a thread-safe manner.

Declaration

```
public void Enqueue(T? item)
```

Parameters

Type	Name	Description
T	*item*	The item to add to the collection. Can be null if the type `T` allows it.

Remarks

This method ensures that items can be added to the collection safely from multiple threads by synchronizing access. It functions as part of the thread-safe queue functionality, allowing items to be enqueued (added to the end).

GetCollectionCopyToList()

Creates and returns a deep copy of the entire collection as a List of T.

Declaration

```
public List<T?> GetCollectionCopyToList()
```


+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeCollection1.html)
  ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValue1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Returns

Type	Description
List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) <T>	A new List containing deep copies of the items in the collection. Returns an empty list if the collection is empty.

Remarks

Locks the collection to ensure thread safety during the copy process. This method uses JSON serialization to create deep copies of the items, providing isolation from the original items. Modifications to the returned list or its items will not affect the original collection.

GetEnumerator()

Returns an enumerator that iterates through the collection in a thread-safe manner.

Declaration

```
public IEnumerator<T?> GetEnumerator()
```

Returns

Type	Description
IEnumerator (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerator-1) <T?>	An IEnumerator<T?> that can be used to iterate through the collection.

Remarks

This method creates a snapshot of the current state of the collection and returns an enumerator for this snapshot, allowing for safe iteration over the collection items even when other threads might be modifying the collection concurrently. Note that the snapshot is a shallow copy; thus, the enumeration reflects the collection's state at the moment of the snapshot's creation.

GetItemAt(int)

Retrieves the item at the specified index in a thread-safe manner.

Declaration

```
public T? GetItemAt(int index)
```

Parameters

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  - ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeCollection1.html)
  - ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValue1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Name	Description
int (https://learn.microsoft.com/dotnet/api/system.int32)	*index*	The zero-based index of the item to retrieve.

Returns

Type	Description
T	The item at the specified index. Can be null if the item itself is null or if the type `T` allows null values.

Remarks

Accesses the collection in a synchronized context to ensure safe retrieval of the item.

Exceptions

Type	Condition
ArgumentOutOfRangeException (https://learn.microsoft.com/dotnet/api/system.argumentoutofrangeexception)	Thrown when the specified index is outside the bounds of the collection.

GetItemCopyAt(int)

Returns a deep copy of the item at the specified index, ensuring thread safety and isolation from the original collection.

Declaration

```
T? GetItemCopyAt(int index)
```

Parameters

Type	Name	Description
int (https://learn.microsoft.com/dotnet/api/system.int32)	*index*	The zero-based index of the item to copy.

Returns

Type	Description
T	A deep copy of the item at the specified index. Returns null if the original item is null.

Remarks

This method locks the collection during the operation and uses JSON serialization to create a deep copy of the item, ensuring that modifications to the returned object do not affect the original item in the collection. Throws ArgumentOutOfRangeException (https://learn.microsoft.com/dotnet/api/system.argumentoutofrangeexception) if the index is out of the valid range.

Exceptions

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeCollection1.html)
  ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValue1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Condition
ArgumentOutOfRangeException (https://learn.microsoft.com/dotnet/api/system.argumentoutofrangeexception)	Thrown when the index is out of range.

LockedLinqDeepCopy<TResult>(Func<IEnumerable<T?>, IEnumerable<TResult>>)

Performs a LINQ operation on a snapshot of the collection and returns a deep copy of the results in a thread-safe manner.

Declaration

```
public IEnumerable<TResult> LockedLinqDeepCopy<TResult>(Func<IEnumerable<T?>, IEnumerable<
    TResult>> operation)
```

Parameters

Type	Name	Description
Func (https://learn.microsoft.com/dotnet/api/system.func-2)<IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<T>, IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<TResult>>	*operation*	A function representing the LINQ operation to be performed on the collection.

Returns

Type	Description
IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<TResult>	An IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) that contains a deep copy of the result of applying the LINQ operation on the collection snapshot.

Type Parameters

Name	Description
TResult	The type of the result elements.

Remarks

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  - ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeColl 1.html)
  - ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValu 1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Similar to LockedLinq<TResult>(Func<IEnumerable<T?>, IEnumerable<TResult>>) (Coree.NETStandard.Classes.ThreadSafeCollection-1.html#Coree_NETStandard_Classes_ThreadSafeCollection_1_LockedLinq__1_System_Func_System_Collections_Generic_IEnumerable__0__System_Collections_Generic_IEnumerable___0___), this method locks the collection and operates on a snapshot to ensure thread safety. After performing the specified LINQ operation, it creates a deep copy of the result using JSON serialization. This approach guarantees that the operation's results are completely isolated from the original items in the collection, providing additional safety against mutations.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  - ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeColl 1.html)
  - ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValu 1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

LockedLinq<TResult>(Func<IEnumerable<T?>, IEnumerable<TResult>>)

Performs a LINQ operation on a snapshot of the collection in a thread-safe manner.

Declaration

```
public IEnumerable<TResult> LockedLinq<TResult>(Func<IEnumerable<T?>, IEnumerable<TResult>> operation)
```

Parameters

Type	Name	Description
Func (https://learn.microsoft.com/dotnet/api/system.func-2)<IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<T>, IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<TResult>>	*operation*	A function representing the LINQ operation to be performed on the collection.

Returns

Type	Description
IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<TResult>	An IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) that contains the result of applying the LINQ operation on the collection snapshot.

Type Parameters

Name	Description
TResult	The type of the result elements.

Remarks

This method locks the collection, takes a snapshot, and then applies the specified LINQ operation. It ensures thread safety by preventing other operations from modifying the collection during execution. The operation is performed on a snapshot to avoid locking during the entire enumeration, but it means the operation does not reflect changes made to the collection after the snapshot is taken.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeColl
  1.html)
  ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValu
  1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Peek()

Returns the last item from the collection without removing it, in a thread-safe manner.

Declaration

```
public T? Peek()
```

Returns

Type	Description
T	The last item of the collection if it exists; otherwise, throws an InvalidOperationException.

Remarks

This method locks the collection to ensure thread safety during the retrieval operation. If the collection is empty, it throws an InvalidOperationException to notify the caller of the empty state, aligning with the behavior of Pop.

Exceptions

Type	Condition
InvalidOperationException (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception)	Thrown when the collection is empty.

Pop()

Removes and returns the last item from the collection in a thread-safe manner.

Declaration

```
public T? Pop()
```

Returns

Type	Description
T	The last item of the collection if it exists; otherwise, throws an InvalidOperationException.

Remarks

This method locks the collection to ensure thread safety during the remove operation. If the collection is empty, it throws an InvalidOperationException. This change ensures that the caller is explicitly aware of the empty state.

Exceptions

Type	Condition
InvalidOperationException (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception)	Thrown when the collection is empty.

Push(T?)

Adds an item to the collection in a thread-safe manner.

Declaration

```
public void Push(T? item)
```

Parameters

Type	Name	Description
T	*item*	The item to add to the collection. Can be null for nullable reference types.

Remarks

This method ensures that the collection can be safely modified from multiple threads by synchronizing access. The item is added to the end of the collection. It accepts null values for types that allow it.

Remove(T?)

Removes the first occurrence of a specific item from the collection in a thread-safe manner.

Declaration

```
public void Remove(T? item)
```

Parameters

Type	Name	Description
T	*item*	The item to remove from the collection. Can be null if the type `T` permits.

Remarks

Synchronizes access to the collection to safely remove the item, supporting concurrent modifications. If the item is not found, no action is taken.

Take(Func<T?, bool>)

Finds and removes the first item matching the given predicate from the collection in a thread-safe manner.

Declaration

```
public T? Take(Func<T?, bool> predicate)
```

Parameters

Type	Name	Description
Func (https://learn.microsoft.com/dotnet/api/system.func-2)<T, bool (https://learn.microsoft.com/dotnet/api/system.boolean)>	*predicate*	The predicate used to find the item to remove.

Returns

Type	Description
T	The item that was removed from the collection, or `null` if no item matched the predicate.

Remarks

Synchronizes access to the collection to safely remove the item, supporting concurrent modifications. If no item matches the predicate, `null` is returned.

TakeAll(Func<T?, bool>)

Finds and removes all items matching the given predicate from the collection in a thread-safe manner.

Declaration

```
public IEnumerable<T?> TakeAll(Func<T?, bool> predicate)
```

Parameters

Type	Name	Description
Func (https://learn.microsoft.com/dotnet/api/system.func-2)<T, bool (https://learn.microsoft.com/dotnet/api/system.boolean)>	*predicate*	The predicate used to find the items to remove.

Returns

Type	Description
IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<T>	An enumerable of all items that were removed from the collection.

Remarks

Synchronizes access to the collection to safely remove the items, supporting concurrent modifications. Returns an empty enumerable if no items match the predicate.

TakeAt(int)

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  - ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeColl 1.html)
  - ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValu 1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Gets and removes the item at the specified index from the collection in a thread-safe manner.

Declaration

```
public T? TakeAt(int index)
```

Parameters

Type	Name	Description
int (https://learn.microsoft.com/dotnet/api/system.int32)	*index*	The zero-based index of the item to fetch and remove.

Returns

Type	Description
T	The item that was removed from the collection.

Remarks

Synchronizes access to the collection to safely remove the item, supporting concurrent modifications. If the index is out of range, an ArgumentOutOfRangeException (https://learn.microsoft.com/dotnet/api/system.argumentoutofrangeexception) is thrown.

Exceptions

Type	Condition
ArgumentOutOfRangeException (https://learn.microsoft.com/dotnet/api/system.argumentoutofrangeexception)	Thrown when the index is out of range.

Implements

IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)
IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.ienumerable)

Extension Methods

EnumerableExtension.RemoveNullEntries<T>(IEnumerable<T?>)
(Coree.NETStandard.Extensions.Enumerable.EnumerableExtension.html#Coree_NETStandard_Extensions_Enumerable_EnumerableExtension_RemoveNullEntries__1_System_Collections_Generic_IEnumerable___0__)

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  - ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeCollection1.html)
  - ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValue1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class ThreadSafeValue<T>

Provides a thread-safe wrapper for a value of type `T`.

Enter here to filter...

Inheritance

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - object (https://learn.microsoft.com/dotnet/api/system.object)
    - ThreadSafeValue<T>
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  - ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeColl 1.html)
  - ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValu 1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Classes (Coree.NETStandard.Classes.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class ThreadSafeValue<T>
```

Type Parameters

Name	Description
T	The type of the value to be stored. This can be any type, including nullable and non-nullable types.

Remarks

This class is designed to allow safe concurrent access to a value by multiple threads. It uses a locking mechanism to ensure that only one thread can read or modify the value at any time, preventing race conditions and ensuring consistency of the value across threads. The type `T` can be either a value type or a reference type. If `T` is a reference type.

Constructors

ThreadSafeValue(T?)

Initializes a new instance of the ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValue-1.html) class, optionally with an initial value.

Declaration

```
public ThreadSafeValue(T? initialValue = default)
```

Parameters

Type	Name	Description
T	*initialValue*	The initial value to store. Default is the default value of type `T`, which is null for reference types and zero or the default constructor value for value types.

Properties

Value

Gets or sets the value in a thread-safe manner.

Declaration

```
public T? Value { get; set; }
```

Property Value

Type	Description
T	The current value stored within the ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValue-1.html) instance.

Remarks

Access to the value is synchronized, ensuring that read and write operations are safe to use from multiple threads concurrently. The property getter returns the current value, and the property setter updates the value. Both operations are performed with thread safety in mind, using a private lock to synchronize access.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
  - ThreadSafeCollection<T> (Coree.NETStandard.Classes.ThreadSafeColl 1.html)
  - ThreadSafeValue<T> (Coree.NETStandard.Classes.ThreadSafeValu 1.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace Coree.NETStandard.CoreeHttpClient

Enter here to filter...

Classes

CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeHttpClient.html)

CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeHttpClient2.html)

HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpHeadersExtensions.html)

HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRequestHeadersExtensions.html)

HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpResponseResult.html)

HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpResponseResult2.html)

Product (Coree.NETStandard.CoreeHttpClient.Product.html)

ProductExtensions (Coree.NETStandard.CoreeHttpClient.ProductExtensions.html)

RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.RequestContentBuilder.html)

RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.RequestParamBuilder.html)

ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.ServiceCollectionExtensions.html)

UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExtensions.html)

Interfaces

ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoreeHttpClient.html)

Enums

HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpResponseResult.OperationStatus.html)

HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpResponseResult2.OperationStatus.html)

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)

Class CoreeHttpClient

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ CoreeHttpClient (Coree.NETStandard.Abstractions.html)

Implements

ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoreeHttpClient.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class CoreeHttpClient : ICoreeHttpClient
```

Constructors

CoreeHttpClient(ILogger<CoreeHttpClient>, IHttpClientFactory, IMemoryCache)

Declaration

```
public CoreeHttpClient(ILogger<CoreeHttpClient> logger, IHttpClientFactory httpClientFacto
ry, IMemoryCache memoryCache)
```

Parameters

Type	Name	Description
ILogger (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.ilogger-1)<CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeHttpClient.html)>	*logger*	
IHttpClientFactory (https://learn.microsoft.com/dotnet/api/system.net.http.ihttpclientfactory)	*httpClientFactory*	
IMemoryCache (https://learn.microsoft.com/dotnet/api/microsoft.extensions.caching.memory.imemorycache)	*memoryCache*	

Enter here to filter...

+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeI
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeI
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpR
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpR
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.ServiceC
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte

+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)

+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)

+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht

Methods

GetAsync(string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
public Task<HttpResponseResult> GetAsync(string url, Dictionary<string, string>? headers =
    null, TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryDelay = null, Cance
    llationToken cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
Dictionary<string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpResponseResult.html)>	

GetJsonDocumentAsync(string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
public Task<JsonDocument?> GetJsonDocumentAsync(string url, Dictionary<string, string>? he
    aders = null, TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryDelay = nul
    l, CancellationToken cancellationToken = default)
```

Parameters

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeHttpClient2)
  - Dictionary<string, string> (Coree.NETStandard.CoreeHttpClient.CoreeHttpClient.Dictionary-2)
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpHeadersExtensions)
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRequestHeadersExtensions)
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpResponseResult)
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpResponseResult.OperationStatus)
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpResponseResult2)
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpResponseResult2.OperationStatus)
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoreeHttpClient)
  - Product (Coree.NETStandard.CoreeHttpClient.Product)
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.ProductExtensions)
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.RequestContentBuilder)
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.RequestParamBuilder)
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.ServiceCollectionExtensions)
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExtensions)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) <string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)< JsonDocument (https://learn.microsoft.com/dotnet/api/system.text.json.jsondocument)>	

GetJsonNodeAsync(string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
public Task<JsonNode?> GetJsonNodeAsync(string url, Dictionary<string, string>? headers =
    null, TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryDelay = null, Cance
    llationToken cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) <string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	

Type	Name	Description
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<JsonNode (https://learn.microsoft.com/dotnet/api/system.text.json.nodes.jsonnode)>	

GetJsonPathResultAsync(string, string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
public Task<PathResult?> GetJsonPathResultAsync(string url, string jsonPath, Dictionary<st
ring, string>? headers = null, TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan?
retryDelay = null, CancellationToken cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*jsonPath*	
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)<string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<PathResult>	

(Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)
- Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.CoreeH
CoreeHttpClient2
(Coree.NETStandard.CoreeHttpClient.CoreeH
HttpHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpHe
HttpRequestHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
ICoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.ICoree
Product
(Coree.NETStandard.CoreeHttpClient.Produc
ProductExtensions
(Coree.NETStandard.CoreeHttpClient.Produc
RequestContentBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
RequestParamBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
ServiceCollectionExtensions
(Coree.NETStandard.CoreeHttpClient.Service
UriExtensions
(Coree.NETStandard.CoreeHttpClient.UriExte
+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht

GetJsonPathResultAsync<T>(string, string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
public Task<List<T>?> GetJsonPathResultAsync<T>(string url, string jsonPath, Dictionary<string, string>? headers = null, TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryDelay = null, CancellationToken cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*jsonPath*	
Dictionary<string, string> (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1)<T>>	

Type Parameters

Name	Description
T	

GetStringAsync(string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

Coree.NETStandard.Abstractions.html)

+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)

CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.CoreeH

CoreeHttpClient2
(Coree.NETStandard.CoreeHttpClient.CoreeH

HttpHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpH

HttpRequestHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpRe

HttpResponseResult
(Coree.NETStandard.CoreeHttpClient.HttpRe

HttpResponseResult.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe

HttpResponseResult2
(Coree.NETStandard.CoreeHttpClient.HttpRe

HttpResponseResult2.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe

ICoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.ICoree

Product
(Coree.NETStandard.CoreeHttpClient.Produ

ProductExtensions
(Coree.NETStandard.CoreeHttpClient.Produ

RequestContentBuilder
(Coree.NETStandard.CoreeHttpClient.Reque

RequestParamBuilder
(Coree.NETStandard.CoreeHttpClient.Reque

ServiceCollectionExtensions
(Coree.NETStandard.CoreeHttpClient.Service

UriExtensions
(Coree.NETStandard.CoreeHttpClient.UriExte

+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)

+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)

+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht

```
public Task<string?> GetStringAsync(string url, Dictionary<string, string>? headers = nul
l, TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryDelay = null, Cancella
tionToken cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
Dictionary<string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)> (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<string (https://learn.microsoft.com/dotnet/api/system.string)>	

Implements

ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoreeHttpClient.html)

Coree.NETStandard.Abstractions.html)

+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.CoreeH
CoreeHttpClient2
(Coree.NETStandard.CoreeHttpClient.CoreeH
HttpHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpH
HttpRequestHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
ICoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.ICoree
Product
(Coree.NETStandard.CoreeHttpClient.Produ
ProductExtensions
(Coree.NETStandard.CoreeHttpClient.Produ
RequestContentBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
RequestParamBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
ServiceCollectionExtensions
(Coree.NETStandard.CoreeHttpClient.Servic
UriExtensions
(Coree.NETStandard.CoreeHttpClient.UriExte

+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)

+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)

+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht

Class CoreeHttpClient2

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ CoreeHttpClient2

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class CoreeHttpClient2
```

Constructors

CoreeHttpClient2(ILogger<CoreeHttpClient2>, IHttpClientFactory, IMemoryCache)

Declaration

```
public CoreeHttpClient2(ILogger<CoreeHttpClient2> logger, IHttpClientFactory httpClientFac
tory, IMemoryCache memoryCache)
```

Parameters

Type	Name	Description
ILogger (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.ilogger-1) <CoreeHttpClient2>	*logger*	
IHttpClientFactory (https://learn.microsoft.com/dotnet/api/system.net.http.ihttpclientfactory)	*httpClientFactory*	
IMemoryCache (https://learn.microsoft.com/dotnet/api/microsoft.extensions.caching.memory.imemorycache)	*memoryCache*	

Methods

GetAsync(HttpMethod, Uri, RequestParamBuilder?, RequestContentBuilder?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
public Task<HttpResponseResult2> GetAsync(HttpMethod httpMethod, Uri baseUrl, RequestParam
Builder? requestParamBuilder = null, RequestContentBuilder? httpContentBuilder = null, Tim
eSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryDelay = null, CancellationTo
ken cancellationToken = default)
```

Parameters

Type	Name	Description
HttpMethod (https://learn.microsoft.com/dotnet/api/system.net.http.httpmethod)	*httpMethod*	
Uri (https://learn.microsoft.com/dotnet/api/system.uri)	*baseUrl*	
RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.RequestParamBuilder.html)	*requestParamBuilder*	
RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.RequestContentBuilder.html)	*httpContentBuilder*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpResponseResult2.html)>	

+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpR
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte

+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)

+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)

+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht

Class HttpHeadersExtensions

Inheritance

object (https://learn.microsoft.com/dotnet/api/system.object)

HttpHeadersExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.CoreeHttpClient.dll

Syntax

```
public static class HttpHeadersExtensions
```

Methods

GetContentEncoding(HttpHeaders)

Declaration

```
public static Encoding GetContentEncoding(this HttpHeaders headers)
```

Parameters

Type	Name	Description
HttpHeaders (https://learn.microsoft.com/dotnet/api/system.net.http.headers.httpheaders)	*headers*	

Returns

Type	Description
Encoding (https://learn.microsoft.com/dotnet/api/system.text.encoding)	

Enter here to filter...

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.Classes.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.Coreel
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.Coreel
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- Coree.NETStandard.Extensions.Stringbuilder

Class HttpRequestHeadersExtensions

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ HttpRequestHeadersExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class HttpRequestHeadersExtensions
```

Methods

AddRequestHeaderCookies(HttpRequestHeaders, Dictionary<string, string>?)

Declaration

```
public static void AddRequestHeaderCookies(this HttpRequestHeaders headers, Dictionary<str
ing, string>? cookies)
```

Parameters

Type	Name	Description
HttpRequestHeaders (https://learn.microsoft.com/dotnet/api/system.net.http.headers.httprequestheaders)	*headers*	
Dictionary<string, string> (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*cookies*	

Enter here to filter...

- Coree.NETStandard (Coree.NETStandard.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
    - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.Coreel
    - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.Coreel
    - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
    - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
    - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
    - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
    - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
    - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
    - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
    - Product (Coree.NETStandard.CoreeHttpClient.Produc
    - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produc
    - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
    - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
    - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
    - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Class HttpResponseResult

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ HttpResponseResult (Coree.NETStandard.Classes.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class HttpResponseResult
```

Constructors

HttpResponseResult(byte[]?, HttpResponseHeaders?, HttpRequestHeaders?, bool, HttpStatusCode)

Declaration

```
public HttpResponseResult(byte[]? contentBytes, HttpResponseHeaders? responseHeaders, HttpRequestHeaders? requestHeaders, bool isFromCache, HttpStatusCode statusCode)
```

Parameters

Type	Name	Description
byte (https://learn.microsoft.com/dotnet/api/system.byte)[]	*contentBytes*	
HttpResponseHeaders (https://learn.microsoft.com/dotnet/api/system.net.http.headers.httpresponseheaders)	*responseHeaders*	
HttpRequestHeaders (https://learn.microsoft.com/dotnet/api/system.net.http.headers.httprequestheaders)	*requestHeaders*	
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*isFromCache*	
HttpStatusCode (https://learn.microsoft.com/dotnet/api/system.net.httpstatuscode)	*statusCode*	

HttpResponseResult(Exception?, HttpStatusCode)

Declaration

```
public HttpResponseResult(Exception? exception, HttpStatusCode statusCode)
```

Parameters

Type	Name	Description
Exception (https://learn.microsoft.com/dotnet/api/system.exception)	*exception*	
HttpStatusCode (https://learn.microsoft.com/dotnet/api/system.net.httpstatuscode)	*statusCode*	

Properties

ContentBytes

Declaration

```
public byte[]? ContentBytes { get; set; }
```

Property Value

Type	Description
byte (https://learn.microsoft.com/dotnet/api/system.byte)[]	

ContentString

Declaration

```
public string? ContentString { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

Exception

Declaration

```
public Exception? Exception { get; set; }
```

Property Value

Type	Description
Exception (https://learn.microsoft.com/dotnet/api/system.exception)	

IsFromCache

Declaration

```
public bool IsFromCache { get; set; }
```

Property Value

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	

RequestHeaders

Declaration

```
public HttpRequestHeaders? RequestHeaders { get; set; }
```

Property Value

Type	Description
HttpRequestHeaders (https://learn.microsoft.com/dotnet/api/system.net.http.headers.httprequestheaders)	

ResponseHeaders

Declaration

```
public HttpResponseHeaders? ResponseHeaders { get; set; }
```

Property Value

Type	Description
HttpResponseHeaders (https://learn.microsoft.com/dotnet/api/system.net.http.headers.httpresponseheaders)	

Status

Declaration

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

```
public HttpResponseResult.OperationStatus Status { get; set; }
```

Property Value

Type	Description
HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpResponseResult.html).OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpResponseResult.OperationStatus.html)	

StatusCode

Declaration

```
public HttpStatusCode StatusCode { get; set; }
```

Property Value

Type	Description
HttpStatusCode (https://learn.microsoft.com/dotnet/api/system.net.httpstatuscode)	

(Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Enum HttpResponseResult.OperationStatus

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public enum HttpResponseResult.OperationStatus
```

Fields

Name	Description
Failure	
Success	

Enter here to filter...

(Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Class HttpResponseResult2

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ HttpResponseResult2 (Coree.NETStandard.Classes.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class HttpResponseResult2
```

Constructors

HttpResponseResult2(byte[]?, HttpResponseHeaders?, HttpRequestHeaders?, bool, HttpStatusCode)

Declaration

```
public HttpResponseResult2(byte[]? contentBytes, HttpResponseHeaders? responseHeaders, HttpRequestHeaders? requestHeaders, bool isFromCache, HttpStatusCode statusCode)
```

Parameters

Type	Name	Description
byte (https://learn.microsoft.com/dotnet/api/system.byte)[]	*contentBytes*	
HttpResponseHeaders (https://learn.microsoft.com/dotnet/api/system.net.http.headers.httpresponseheaders)	*responseHeaders*	
HttpRequestHeaders (https://learn.microsoft.com/dotnet/api/system.net.http.headers.httprequestheaders)	*requestHeaders*	
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*isFromCache*	
HttpStatusCode (https://learn.microsoft.com/dotnet/api/system.net.httpstatuscode)	*statusCode*	

HttpResponseResult2(Exception?, HttpStatusCode)

Declaration

Enter here to filter...
- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
Product (Coree.NETStandard.CoreeHttpClient.Produc
ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produc
RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

```
public HttpResponseResult2(Exception? exception, HttpStatusCode statusCode)
```

Parameters

Type	Name	Description
Exception (https://learn.microsoft.com/dotnet/api/system.exception)	*exception*	
HttpStatusCode (https://learn.microsoft.com/dotnet/api/system.net.httpstatuscode)	*statusCode*	

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produc
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produc
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Properties

ContentBytes

Declaration

```
public byte[]? ContentBytes { get; set; }
```

Property Value

Type	Description
byte (https://learn.microsoft.com/dotnet/api/system.byte)[]	

ContentLength

Declaration

```
public long? ContentLength { get; set; }
```

Property Value

Type	Description
long (https://learn.microsoft.com/dotnet/api/system.int64)?	

ContentString

Declaration

```
public string? ContentString { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

Exception

Declaration

```
public Exception? Exception { get; set; }
```

Property Value

Type	Description
Exception (https://learn.microsoft.com/dotnet/api/system.exception)	

IsFromCache

Declaration

```
public bool IsFromCache { get; set; }
```

Property Value

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	

RequestHeaders

Declaration

```
public HttpRequestHeaders? RequestHeaders { get; set; }
```

Property Value

Type	Description
HttpRequestHeaders (https://learn.microsoft.com/dotnet/api/system.net.http.headers.httprequestheaders)	

ResponseHeaders

Declaration

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

```
public HttpResponseHeaders? ResponseHeaders { get; set; }
```

Property Value

Type	Description
HttpResponseHeaders (https://learn.microsoft.com/dotnet/api/system.net.http.headers.httpresponseheaders)	

(Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)

CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.CoreeH
CoreeHttpClient2
(Coree.NETStandard.CoreeHttpClient.CoreeH
HttpHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpH
HttpRequestHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
ICoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.ICoree
Product
(Coree.NETStandard.CoreeHttpClient.Produc
ProductExtensions
(Coree.NETStandard.CoreeHttpClient.Produc
RequestContentBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
RequestParamBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
ServiceCollectionExtensions
(Coree.NETStandard.CoreeHttpClient.Service
UriExtensions
(Coree.NETStandard.CoreeHttpClient.UriExte

+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)

+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)

+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht

+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h

Status

Declaration

```
public HttpResponseResult2.OperationStatus Status { get; set; }
```

Property Value

Type	Description
HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpResponseResult2.html).OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpResponseResult2.OperationStatus.html)	

StatusCode

Declaration

```
public HttpStatusCode StatusCode { get; set; }
```

Property Value

Type	Description
HttpStatusCode (https://learn.microsoft.com/dotnet/api/system.net.httpstatuscode)	

Enum HttpResponseResult2.OperationStatus

Enter here to filter...

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public enum HttpResponseResult2.OperationStatus
```

Fields

Name	Description
Failure	
Success	

(Coree.NETStandard.Classes.html)
- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.Coree
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.Coree
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Interface ICoreeHttpClient

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public interface ICoreeHttpClient
```

Methods

GetAsync(string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
Task<HttpResponseResult> GetAsync(string url, Dictionary<string, string>? headers = null,
TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryDelay = null, Cancellatio
nToken cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
Dictionary<string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpResponseResult.html)>	

GetJsonDocumentAsync(string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
Task<JsonDocument?> GetJsonDocumentAsync(string url, Dictionary<string, string>? headers =
null, TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryDelay = null, Cance
llationToken cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
Dictionary<string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)> (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<JsonDocument (https://learn.microsoft.com/dotnet/api/system.text.json.jsondocument)>	

GetJsonNodeAsync(string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
Task<JsonNode?> GetJsonNodeAsync(string url, Dictionary<string, string>? headers = null, T
imeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryDelay = null, Cancellation
Token cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	

Type	Name	Description
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) <string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<JsonNode (https://learn.microsoft.com/dotnet/api/system.text.json.nodes.jsonnode)>	

GetJsonPathResultAsync(string, string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
Task<PathResult?> GetJsonPathResultAsync(string url, string jsonPath, Dictionary<string, s
    tring>? headers = null, TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryD
    elay = null, CancellationToken cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*jsonPath*	
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) <string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeHttpClient-2)
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpHeadersExtensions)
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRequestHeadersExtensions)
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpResponseResult)
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpResponseResult2)
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpResponseResult2)
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpResponseResult2.OperationStatus)
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoreeHttpClient)
  - Product (Coree.NETStandard.CoreeHttpClient.Product)
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.ProductExtensions)
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.RequestContentBuilder)
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.RequestParamBuilder)
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.ServiceCollectionExtensions)
  - UriExt (Coree.NETStandard.CoreeHttpClient.UriExt)
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Type	Name	Description
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<PathResult>	

GetJsonPathResultAsync<T>(string, string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
Task<List<T>?> GetJsonPathResultAsync<T>(string url, string jsonPath, Dictionary<string, s
tring>? headers = null, TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryD
elay = null, CancellationToken cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*jsonPath*	
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) <string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1)<T>>	

Type Parameters

- Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Servic
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Name	Description
T	

GetStringAsync(string, Dictionary<string, string>?, TimeSpan?, int, TimeSpan?, CancellationToken)

Declaration

```
Task<string?> GetStringAsync(string url, Dictionary<string, string>? headers = null, TimeSpan? cacheDuration = null, int maxTries = 3, TimeSpan? retryDelay = null, CancellationToken cancellationToken = default)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*url*	
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)<string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*headers*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*cacheDuration*	
int (https://learn.microsoft.com/dotnet/api/system.int32)	*maxTries*	
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan)?	*retryDelay*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<string (https://learn.microsoft.com/dotnet/api/system.string)>	

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Class Product

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ FluentBase (Coree.NETStandard.Abstractions.FluentBase.html)

↳ Product

Implements

IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)

Inherited Members

Enter here to filter...

(Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.CoreeH
CoreeHttpClient2
(Coree.NETStandard.CoreeHttpClient.CoreeH
HttpHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpH
HttpRequestHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
ICoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.ICoree
Product
(Coree.NETStandard.CoreeHttpClient.Produ
ProductExtensions
(Coree.NETStandard.CoreeHttpClient.Produc
RequestContentBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
RequestParamBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
ServiceCollectionExtensions
(Coree.NETStandard.CoreeHttpClient.Service
UriExtensions
(Coree.NETStandard.CoreeHttpClient.UriExte
+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class Product : FluentBase, IFluentBase
```

Properties

Name

Declaration

```
public string? Name { get; set; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

Price

Declaration

```
public decimal? Price { get; set; }
```

Property Value

Type	Description
decimal (https://learn.microsoft.com/dotnet/api/system.decimal)?	

Implements

IFluentBase (Coree.NETStandard.Abstractions.IFluentBase.html)

Extension Methods

ProductExtensions.AdjustPrice(Product, decimal)
(Coree.NETStandard.CoreeHttpClient.ProductExtensions.html#Coree_NETStandard_CoreeHttpClient_ProductExtensions_AdjustPrice_Coree_NETStandard_CoreeHttpClient_Product_System_Decimal_)

ProductExtensions.AdjustPriceAsync(Product, decimal)
(Coree.NETStandard.CoreeHttpClient.ProductExtensions.html#Coree_NETStandard_CoreeHttpClient_ProductExtensions_AdjustPriceAsync_Coree_NETStandard_CoreeHttpClient_Product_System_Decimal_)

(Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.CoreeH
CoreeHttpClient2
(Coree.NETStandard.CoreeHttpClient.CoreeH
HttpHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpHe
HttpRequestHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
ICoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.ICoree
Product
(Coree.NETStandard.CoreeHttpClient.Produc
ProductExtensions
(Coree.NETStandard.CoreeHttpClient.Produc
RequestContentBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
RequestParamBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
ServiceCollectionExtensions
(Coree.NETStandard.CoreeHttpClient.Service
UriExtensions
(Coree.NETStandard.CoreeHttpClient.UriExte

+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)

+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)

+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht

+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h

Class ProductExtensions

Inheritance

object (https://learn.microsoft.com/dotnet/api/system.object)

ProductExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class ProductExtensions
```

Methods

AdjustPrice(Product, decimal)

Declaration

```
public static Product AdjustPrice(this Product product, decimal adjustment)
```

Parameters

Type	Name	Description
Product (Coree.NETStandard.CoreeHttpClient.Product.html)	*product*	
decimal (https://learn.microsoft.com/dotnet/api/system.decimal)	*adjustment*	

Returns

Type	Description
Product (Coree.NETStandard.CoreeHttpClient.Product.html)	

AdjustPriceAsync(Product, decimal)

Declaration

```
public static Task<Product> AdjustPriceAsync(this Product product, decimal adjustment)
```

Parameters

Type	Name	Description
Product (Coree.NETStandard.CoreeHttpClient.Product.html)	*product*	
decimal (https://learn.microsoft.com/dotnet/api/system.decimal)	*adjustment*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<Product (Coree.NETStandard.CoreeHttpClient.Product.html)>	

AdjustPriceAsync(Task<Product>, decimal)

Declaration

```
public static Task<Product> AdjustPriceAsync(this Task<Product> productTask, decimal adjus
tment)
```

Parameters

Type	Name	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<Product (Coree.NETStandard.CoreeHttpClient.Product.html)>	*productTask*	
decimal (https://learn.microsoft.com/dotnet/api/system.decimal)	*adjustment*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<Product (Coree.NETStandard.CoreeHttpClient.Product.html)>	

(Coree.NETStandard.Classes.html)
- Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.CoreeH
CoreeHttpClient2
(Coree.NETStandard.CoreeHttpClient.CoreeH
HttpHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpH
HttpRequestHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
ICoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.ICoree
Product
(Coree.NETStandard.CoreeHttpClient.Produc
ProductExtensions
(Coree.NETStandard.CoreeHttpClient.Produc
RequestContentBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
RequestParamBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
ServiceCollectionExtensions
(Coree.NETStandard.CoreeHttpClient.Service
UriExtensions
(Coree.NETStandard.CoreeHttpClient.UriExte
+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h

Class RequestContentBuilder

Inheritance

object (https://learn.microsoft.com/dotnet/api/system.object)

RequestContentBuilder

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class RequestContentBuilder
```

Constructors

RequestContentBuilder()

Declaration

```
public RequestContentBuilder()
```

RequestContentBuilder(byte[], string?)

Declaration

```
public RequestContentBuilder(byte[] bytes, string? name = null)
```

Parameters

Type	Name	Description
byte (https://learn.microsoft.com/dotnet/api/system.byte)[]	*bytes*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*name*	

RequestContentBuilder(Dictionary<string, string>, string?)

Declaration

```
public RequestContentBuilder(Dictionary<string, string> formData, string? name = null)
```

Parameters

Type	Name	Description
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) <string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*formData*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*name*	

(Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.CoreeH
CoreeHttpClient2
(Coree.NETStandard.CoreeHttpClient.CoreeH
HttpHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpHe
HttpRequestHeadersExtensions
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2
(Coree.NETStandard.CoreeHttpClient.HttpRe
HttpResponseResult2.OperationStatus
(Coree.NETStandard.CoreeHttpClient.HttpRe
ICoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.ICoree
Product
(Coree.NETStandard.CoreeHttpClient.Produc
ProductExtensions
(Coree.NETStandard.CoreeHttpClient.Produc
RequestContentBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
RequestParamBuilder
(Coree.NETStandard.CoreeHttpClient.Reque
ServiceCollectionExtensions
(Coree.NETStandard.CoreeHttpClient.Service
UriExtensions
(Coree.NETStandard.CoreeHttpClient.UriExte
+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h

RequestContentBuilder(Stream, string?, string?)

Declaration

```
public RequestContentBuilder(Stream stream, string? name = null, string? filename = null)
```

Parameters

Type	Name	Description
Stream (https://learn.microsoft.com/dotnet/api/system.io.stream)	*stream*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*name*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*filename*	

RequestContentBuilder(string, string, Encoding?, string?)

Declaration

```
public RequestContentBuilder(string data, string mediaType, Encoding? encoding = null, str
    ing? name = null)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*data*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*mediaType*	
Encoding (https://learn.microsoft.com/dotnet/api/system.text.encoding)	*encoding*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*name*	

RequestContentBuilder(string, Encoding?, string?)

Declaration

```
public RequestContentBuilder(string data, Encoding? encoding = null, string? name = null)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*data*	
Encoding (https://learn.microsoft.com/dotnet/api/system.text.encoding)	*encoding*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*name*	

Methods

AddContent(byte[], string?)

Declaration

```
public void AddContent(byte[] bytes, string? name = null)
```

Parameters

Type	Name	Description
byte (https://learn.microsoft.com/dotnet/api/system.byte)[]	*bytes*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*name*	

AddContent(Dictionary<string, string>, string?)

Declaration

```
public void AddContent(Dictionary<string, string> formData, string? name = null)
```

Parameters

Type	Name	Description
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) <string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*formData*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*name*	

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpR
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

AddContent(Stream, string?, string?)

Declaration

```
public void AddContent(Stream stream, string? name = null, string? filename = null)
```

Parameters

Type	Name	Description
Stream (https://learn.microsoft.com/dotnet/api/system.io.stream)	*stream*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*name*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*filename*	

AddContent(string, string, Encoding?, string?)

Declaration

```
public void AddContent(string data, string mediaType, Encoding? encoding = null, string? n
ame = null)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*data*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*mediaType*	
Encoding (https://learn.microsoft.com/dotnet/api/system.text.encoding)	*encoding*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*name*	

AddContent(string, Encoding?, string?)

Declaration

```
public void AddContent(string data, Encoding? encoding = null, string? name = null)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*data*	
Encoding (https://learn.microsoft.com/dotnet/api/system.text.encoding)	*encoding*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*name*	

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpHe
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Servic
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Build()

Declaration

```
public HttpContent? Build()
```

Returns

Type	Description
HttpContent (https://learn.microsoft.com/dotnet/api/system.net.http.httpcontent)	

(Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.Coree
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.Coree
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Class RequestParamBuilder

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ RequestParamBuilder

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class RequestParamBuilder
```

Constructors

RequestParamBuilder(Dictionary<string, string>?, Dictionary<string, string>?, Dictionary<string, string>?)

Declaration

```
public RequestParamBuilder(Dictionary<string, string>? queryParams = null, Dictionary<string, string>? headers = null, Dictionary<string, string>? cookies = null)
```

Parameters

Type	Name	Description
Dictionary<string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)> (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)	*queryParams*	
Dictionary<string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)> (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)	*headers*	
Dictionary<string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)> (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)	*cookies*	

Properties

Enter here to filter...

- Coree.NETStandard (Coree.NETStandard.html)
  - Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
    - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.Coreel
    - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.Coreel
    - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
    - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
    - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
    - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
    - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
    - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
    - Product (Coree.NETStandard.CoreeHttpClient.Produ
    - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
    - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
    - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
    - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
    - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
  - + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
  - + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
  - + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
  - + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Cookies

Declaration

```
public Dictionary<string, string>? Cookies { get; }
```

Property Value

Type	Description
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) < string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string) >	

Headers

Declaration

```
public Dictionary<string, string>? Headers { get; }
```

Property Value

Type	Description
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) < string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string) >	

QueryParams

Declaration

```
public Dictionary<string, string>? QueryParams { get; }
```

Property Value

Type	Description
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) < string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string) >	

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExt
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Class ServiceCollectionExtensions

Inheritance

object (https://learn.microsoft.com/dotnet/api/system.object)

ServiceCollectionExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class ServiceCollectionExtensions
```

Methods

AddCoreeHttpClient(IServiceCollection)

Declaration

```
public static IServiceCollection AddCoreeHttpClient(this IServiceCollection services)
```

Parameters

Type	Name	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	*services*	

Returns

Type	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	

AddCoreeHttpClient2(IServiceCollection)

Declaration

```
public static IServiceCollection AddCoreeHttpClient2(this IServiceCollection services)
```

Parameters

Type	Name	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	*services*	

Returns

Type	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	

(Coree.NETStandard.Classes.html)

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.CoreeH
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.CoreeH
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produ
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produ
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Class UriExtensions

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ UriExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class UriExtensions
```

Methods

AddOrUpdateQueryParameters(Uri, Dictionary<string, string>?)

Declaration

```
public static Uri AddOrUpdateQueryParameters(this Uri uri, Dictionary<string, string>? par
ametersToAddOrUpdate)
```

Parameters

Type	Name	Description
Uri (https://learn.microsoft.com/dotnet/api/system.uri)	*uri*	
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)<string (https://learn.microsoft.com/dotnet/api/system.string), string (https://learn.microsoft.com/dotnet/api/system.string)>	*parametersToAddOrUpdate*	

Returns

Type	Description
Uri (https://learn.microsoft.com/dotnet/api/system.uri)	

Enter here to filter...

- Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
  - CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.Coreel
  - CoreeHttpClient2 (Coree.NETStandard.CoreeHttpClient.Coreel
  - HttpHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpH
  - HttpRequestHeadersExtensions (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResultOperationStatus (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - HttpResponseResult2.OperationStatus2 (Coree.NETStandard.CoreeHttpClient.HttpRe
  - ICoreeHttpClient (Coree.NETStandard.CoreeHttpClient.ICoree
  - Product (Coree.NETStandard.CoreeHttpClient.Produc
  - ProductExtensions (Coree.NETStandard.CoreeHttpClient.Produc
  - RequestContentBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - RequestParamBuilder (Coree.NETStandard.CoreeHttpClient.Reque
  - ServiceCollectionExtensions (Coree.NETStandard.CoreeHttpClient.Service
  - UriExtensions (Coree.NETStandard.CoreeHttpClient.UriExte
- + Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- + Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- + Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- + Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h

Namespace Coree.NETStandard.Extensions

Classes

Enter here to filter...

+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
- Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)
  CancellationTokenExtensions
  (Coree.NETStandard.Extensions.Cancellation
  HostApplicationBuilderExtensions
  (Coree.NETStandard.Extensions.HostApplica
  ServiceProviderExtensions
  (Coree.NETStandard.Extensions.ServiceProvi
+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings
(Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection
(Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog
(Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services
(Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights
(Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole
(Coree.NETStandard.SpectreConsole.html)

CancellationTokenExtensions (Coree.NETStandard.Extensions.CancellationTokenExtensions.html)

Provides extension methods for CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken) to enhance asynchronous programming patterns.

HostApplicationBuilderExtensions (Coree.NETStandard.Extensions.HostApplicationBuilderExtensions.html)

ServiceProviderExtensions (Coree.NETStandard.Extensions.ServiceProviderExtensions.html)

Class CancellationTokenExtensions

Provides extension methods for CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken) to enhance asynchronous programming patterns.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ CancellationTokenExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Extensions (Coree.NETStandard.Extensions.html)
Assembly: Coree.NETStandard.dll

Syntax

```
public static class CancellationTokenExtensions
```

Methods

WaitForCancellationAsync(CancellationToken)

Creates a Task that completes when the specified CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken) is cancelled. This method facilitates integrating cancellation tokens with asynchronous operations, enabling easy response to cancellation requests in a Task-based pattern.

Example usage (awaiting application start in a background service):

Enter here to filter...
+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
  CancellationTokenExtensions (Coree.NETStandard.Extensions.Cancellation
  HostApplicationBuilderExtensions (Coree.NETStandard.Extensions.HostApplica
  ServiceProviderExtensions (Coree.NETStandard.Extensions.ServiceProvi
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.StringBuilder (Coree.NETStandard.Extensions.StringBuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
public class MyBackgroundService : BackgroundService
{
    private readonly IHostApplicationLifetime _appLifetime;

    public MyBackgroundService(IHostApplicationLifetime appLifetime)
    {
        _appLifetime = appLifetime;
    }

    protected override async Task ExecuteAsync(CancellationToken stoppingToken)
    {
        // Wait for the application to fully start
        await _appLifetime.ApplicationStarted.WaitForCancellationAsync();

        Console.WriteLine("Application has started. Background service is proceeding with its tasks.");

        // Now that the application has started, proceed with the background service's main logic
        while (!stoppingToken.IsCancellationRequested)
        {
            // Example operation: perform some work and wait
            Console.WriteLine("Background service is doing its work...");
            await Task.Delay(10000, stoppingToken); // Simulate some work by delaying
        }
    }
}
```

Note: This extension is particularly useful for synchronizing startup sequences in applications, such as ensuring background services wait for the entire application to start before proceeding with their tasks.

Declaration

```
public static Task WaitForCancellationAsync(this CancellationToken cancellationToken)
```

Parameters

Type	Name	Description
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	The CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken) to monitor for cancellation.

Returns

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
  - CancellationTokenExtensions (Coree.NETStandard.Extensions.CancellationTo
  - HostApplicationBuilderExtensions (Coree.NETStandard.Extensions.HostApplica
  - ServiceProviderExtensions (Coree.NETStandard.Extensions.ServiceProvi
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.StringBuilder (Coree.NETStandard.Extensions.StringBuilder
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task)	A Task that completes when the token is cancelled.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
  - CancellationTokenExtensions (Coree.NETStandard.Extensions.Cancellation
  - HostApplicationBuilderExtensions (Coree.NETStandard.Extensions.HostApplica
  - ServiceProviderExtensions (Coree.NETStandard.Extensions.ServiceProvi
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class HostApplicationBuilderExtensions

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ HostApplicationBuilderExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Extensions (Coree.NETStandard.Extensions.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class HostApplicationBuilderExtensions
```

Methods

ConfigureAppConfiguration(IHostApplicationBuilder, Action<IConfigurationBuilder>)

Declaration

```
public static IHostApplicationBuilder ConfigureAppConfiguration(this IHostApplicationBuild
er hostBuilder, Action<IConfigurationBuilder> configureDelegate)
```

Parameters

Type	Name	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	*hostBuilder*	
Action (https://learn.microsoft.com/dotnet/api/system.action-1)< IConfigurationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.configuration.iconfigurationbuilder) >	*configureDelegate*	

Returns

Type	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	

Enter here to filter...

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
  - CancellationTokenExtensions (Coree.NETStandard.Extensions.Cancellation
  - HostApplicationBuilderExtensions (Coree.NETStandard.Extensions.HostApplica
  - ServiceProviderExtensions (Coree.NETStandard.Extensions.ServiceProvi
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

ConfigureAppConfiguration(IHostApplicationBuilder, Action<IHostApplicationBuilder, IConfigurationBuilder>)

Declaration

```
public static IHostApplicationBuilder ConfigureAppConfiguration(this IHostApplicationBuild
er hostBuilder, Action<IHostApplicationBuilder, IConfigurationBuilder> configureDelegate)
```

Parameters

Type	Name	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	*hostBuilder*	
Action (https://learn.microsoft.com/dotnet/api/system.action-2)< IHostApplicationBuilder, IConfigurationBuilder> (https://learn.microsoft.com/dotnet/api/microsoft.extensions.configuration.iconfigurationbuilder)	*configureDelegate*	

Returns

Type	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	

ConfigureHostOptions(IHostApplicationBuilder, Action<HostOptions>)

Declaration

```
public static IHostApplicationBuilder ConfigureHostOptions(this IHostApplicationBuilder ho
stBuilder, Action<HostOptions> configureOptions)
```

Parameters

Type	Name	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	*hostBuilder*	

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
  CancellationTokenExtensions (Coree.NETStandard.Extensions.CancellationTokenExtensions.html)
  HostApplicationBuilderExtensions (Coree.NETStandard.Extensions.HostApplicationBuilderExtensions.html)
  ServiceProviderExtensions (Coree.NETStandard.Extensions.ServiceProviderExtensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Name	Description
Action (https://learn.microsoft.com/dotnet/api/system.action-1)<HostOptions (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.hostoptions)>	*configureOptions*	

Returns

Type	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	

ConfigureHostOptions(IHostApplicationBuilder, Action<IHostApplicationBuilder, HostOptions>)

Declaration

```
public static IHostApplicationBuilder ConfigureHostOptions(this IHostApplicationBuilder ho
stBuilder, Action<IHostApplicationBuilder, HostOptions> configureOptions)
```

Parameters

Type	Name	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	*hostBuilder*	
Action (https://learn.microsoft.com/dotnet/api/system.action-2)<IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder), HostOptions (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.hostoptions)>	*configureOptions*	

Returns

Type	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	

ConfigureLogging(IHostApplicationBuilder, Action<IHostApplicationBuilder, ILoggingBuilder>)

Declaration

```
public static IHostApplicationBuilder ConfigureLogging(this IHostApplicationBuilder hostBu
ilder, Action<IHostApplicationBuilder, ILoggingBuilder> configureDelegate)
```

Parameters

Type	Name	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	*hostBuilder*	
Action (https://learn.microsoft.com/dotnet/api/system.action-2)< IHostApplicationBuilder, ILoggingBuilder> (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.iloggingbuilder)	*configureDelegate*	

Returns

Type	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	

ConfigureLogging(IHostApplicationBuilder, Action<ILoggingBuilder>)

Declaration

```
public static IHostApplicationBuilder ConfigureLogging(this IHostApplicationBuilder hostBu
ilder, Action<ILoggingBuilder> configureDelegate)
```

Parameters

Type	Name	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	*hostBuilder*	

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
  - CancellationTokenExtensions (Coree.NETStandard.Extensions.CancellationTo
  - HostApplicationBuilderExtensions (Coree.NETStandard.Extensions.HostApplica
  - ServiceProviderExtensions (Coree.NETStandard.Extensions.ServiceProvi
  - >
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Name	Description
Action (https://learn.microsoft.com/dotnet/api/system.action-1)< ILoggingBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.iloggingbuilder) >	*configureDelegate*	

Returns

Type	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	

ConfigureServices(IHostApplicationBuilder, Action<IServiceCollection>)

Declaration

```
public static IHostApplicationBuilder ConfigureServices(this IHostApplicationBuilder hostBuilder, Action<IServiceCollection> configureDelegate)
```

Parameters

Type	Name	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	*hostBuilder*	
Action (https://learn.microsoft.com/dotnet/api/system.action-1)< IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) >	*configureDelegate*	

Returns

Type	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	

ConfigureServices(IHostApplicationBuilder, Action<IHostApplicationBuilder, IServiceCollection>)

Declaration

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
  - CancellationTokenExtensions (Coree.NETStandard.Extensions.Cancellation
  - HostApplicationBuilderExtensions (Coree.NETStandard.Extensions.HostApplica
  - ServiceProviderExtensions (Coree.NETStandard.Extensions.ServiceProvi
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
public static IHostApplicationBuilder ConfigureServices(this IHostApplicationBuilder hostB
uilder, Action<IHostApplicationBuilder, IServiceCollection> configureDelegate)
```

Parameters

Type	Name	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	*hostBuilder*	
Action<IHostApplicationBuilder, IServiceCollection> (https://learn.microsoft.com/dotnet/api/system.action-2)	*configureDelegate*	

Returns

Type	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	

UseConsoleLifetime(IHostApplicationBuilder)

Declaration

```
public static IHostApplicationBuilder UseConsoleLifetime(this IHostApplicationBuilder host
Builder)
```

Parameters

Type	Name	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	*hostBuilder*	

Returns

Type	Description
IHostApplicationBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationbuilder)	

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
  - CancellationTokenExtensions (Coree.NETStandard.Extensions.CancellationTokenExtensions.html)
  - HostApplicationBuilderExtensions (Coree.NETStandard.Extensions.HostApplicationBuilderExtensions.html)
  - ServiceProviderExtensions (Coree.NETStandard.Extensions.ServiceProviderExtensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

+ Coree.NETStandard.Abstractions
(Coree.NETStandard.Abstractions.html)

+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)

+ Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)

- Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)

  CancellationTokenExtensions
  (Coree.NETStandard.Extensions.Cancellation
  HostApplicationBuilderExtensions
  (Coree.NETStandard.Extensions.HostApplica
  ServiceProviderExtensions
  (Coree.NETStandard.Extensions.ServiceProvi

+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)

+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht

+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h

+ Coree.NETStandard.Extensions.Strings
(Coree.NETStandard.Extensions.Strings.html)

+ Coree.NETStandard.HostedServicesCollection
(Coree.NETStandard.HostedServicesCollection

+ Coree.NETStandard.Serilog
(Coree.NETStandard.Serilog.html)

+ Coree.NETStandard.Services
(Coree.NETStandard.Services.html)

+ Coree.NETStandard.Services.RuntimeInsights
(Coree.NETStandard.Services.RuntimeInsights.

+ Coree.NETStandard.SpectreConsole
(Coree.NETStandard.SpectreConsole.html)

Class ServiceProviderExtensions

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ ServiceProviderExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Extensions (Coree.NETStandard.Extensions.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class ServiceProviderExtensions
```

Methods

GetRegisteredServices(IServiceProvider)

Declaration

```
public static List<ServiceDescriptor> GetRegisteredServices(this IServiceProvider serviceP
rovider)
```

Parameters

Type	Name	Description
IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider)	*serviceProvider*	

Returns

Type	Description
List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) < ServiceDescriptor (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.servicedescriptor) >	

Enter here to filter...

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
- Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
  - CancellationTokenExtensions (Coree.NETStandard.Extensions.Cancellation
  - HostApplicationBuilderExtensions (Coree.NETStandard.Extensions.HostApplica
  - ServiceProviderExtensions (Coree.NETStandard.Extensions.ServiceProvi
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace Coree.NETStandard.Extensions.Bool

Enter here to filter...

Classes

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
  - BoolExtensions (Coree.NETStandard.Extensions.Bool.BoolEx
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

BoolExtensions (Coree.NETStandard.Extensions.Bool.BoolExtensions.html)

Provides extension methods for working with nullable boolean values.

Class BoolExtensions

Provides extension methods for working with nullable boolean values.

Enter here to filter...

Inheritance

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - object (https://learn.microsoft.com/dotnet/api/system.object)
    - BoolExtensions
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
  - BoolExtensions (Coree.NETStandard.Extensions.Bool.BoolEx
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Extensions (Coree.NETStandard.Extensions.html).Bool (Coree.NETStandard.Extensions.Bool.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class BoolExtensions
```

Methods

IsFalse(bool?)

Checks if the nullable boolean value is false.

Declaration

```
public static bool IsFalse(this bool? nullableBool)
```

Parameters

Type	Name	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)?	*nullableBool*	The nullable boolean value.

Returns

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	True if the value is false; otherwise, false.

IsNull(bool?)

Checks if the nullable boolean value is null.

Declaration

```
public static bool IsNull(this bool? nullableBool)
```

Parameters

Type	Name	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)?	*nullableBool*	The nullable boolean value.

Returns

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	True if the value is null; otherwise, false.

IsTrue(bool?)

Checks if the nullable boolean value is true.

Declaration

```
public static bool IsTrue(this bool? nullableBool)
```

Parameters

Type	Name	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)?	*nullableBool*	The nullable boolean value.

Returns

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	True if the value is true; otherwise, false.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
- Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
  - BoolExtensions (Coree.NETStandard.Extensions.Bool.BoolEx
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace Coree.NETStandard.Extensions.Enumerable

Enter here to filter...

Classes

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
  EnumerableExtension (Coree.NETStandard.Extensions.Enumerable
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

EnumerableExtension (Coree.NETStandard.Extensions.Enumerable.EnumerableExtension.html)

Provides extension methods for IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) to enhance and simplify operations on collections.

Class EnumerableExtension

Provides extension methods for IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) to enhance and simplify operations on collections.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ EnumerableExtension

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Extensions (Coree.NETStandard.Extensions.html).Enumerable (Coree.NETStandard.Extensions.Enumerable.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class EnumerableExtension
```

Remarks

This static class contains a collection of utility methods that extend the functionality of the IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) interface. These methods offer convenient ways to perform common operations on enumerable collections, such as filtering, transformation, and aggregation, with a focus on improving code readability and efficiency. This class is declared as partial to allow for easy extension and organization of its methods across multiple files, facilitating maintainability and scalability of the utility functions provided.

Methods

RemoveNullEntries<T>(IEnumerable<T?>)

Filters out null entries from an IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1).

Declaration

```
public static IEnumerable<T> RemoveNullEntries<T>(this IEnumerable<T?> enumerable) where T
: class?
```

Parameters

Type	Name	Description
IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <T>	*enumerable*	The enumerable collection to filter.

Returns

Type	Description
IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <T>	An IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) without null elements.

Type Parameters

Name	Description
T	The type of elements in the collection, constrained to reference types.

Remarks

This extension method filters out each element in the given enumerable, returning only those elements that are not null. It is applicable to collections of reference types.

RemoveNullOrWhiteSpaceEntries(IEnumerable<string?>)

Filters out null, empty, and whitespace-only entries from an IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1).

Declaration

```
public static IEnumerable<string> RemoveNullOrWhiteSpaceEntries(this IEnumerable<string?>
    enumerable)
```

Parameters

Type	Name	Description
IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) <string (https://learn.microsoft.com/dotnet/api/system.string)>	*enumerable*	The enumerable collection of strings to filter.

Returns

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
  EnumerableExtension (Coree.NETStandard.Extensions.Enumerable
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Description
IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) < string (https://learn.microsoft.com/dotnet/api/system.string) >	An IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) consisting of non-empty and non-whitespace strings.

Remarks

This extension method iterates over each element in the given enumerable, returning only those elements that are not null, not empty, and not made up solely of whitespace characters.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
- Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
  EnumerableExtension (Coree.NETStandard.Extensions.Enumerable
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace Coree.NETStandard.Extensions.Stringbuilder

Enter here to filter...

Classes

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
- Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
  - StringBuilderExtension (Coree.NETStandard.Extensions.Stringbuilde
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

StringBuilderExtension (Coree.NETStandard.Extensions.Stringbuilder.StringBuilderExtension.html)

Provides extension methods for the StringBuilder (https://learn.microsoft.com/dotnet/api/system.text.stringbuilder) class.

Class StringBuilderExtension

Provides extension methods for the StringBuilder (https://learn.microsoft.com/dotnet/api/system.text.stringbuilder) class.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ StringBuilderExtension

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Extensions (Coree.NETStandard.Extensions.html).Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class StringBuilderExtension
```

Methods

AppendLineOrDefault(StringBuilder, string?)

Appends a new line to the StringBuilder (https://learn.microsoft.com/dotnet/api/system.text.stringbuilder) instance. If the provided string is not null, empty, or whitespace, it appends the string followed by a new line; otherwise, it only appends a new line.

Declaration

```
public static void AppendLineOrDefault(this StringBuilder stringbuilder, string? value)
```

Parameters

Type	Name	Description
StringBuilder (https://learn.microsoft.com/dotnet/api/system.text.stringbuilder)	*stringbuilder*	The StringBuilder (https://learn.microsoft.com/dotnet/api/system.text.stringbuilder) instance on which the method is called.
string (https://learn.microsoft.com/dotnet/api/system.string)	*value*	The string to append before the new line. If the string is null, empty, or consists only of white-space characters, only a new line is appended.

+ Coree.NETStandard.Abstractions
(Coree.NETStandard.Abstractions.html)

+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)

+ Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)

+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)

+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)

+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht

- Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h

  StringBuilderExtension
  (Coree.NETStandard.Extensions.Stringbuilde

+ Coree.NETStandard.Extensions.Strings
(Coree.NETStandard.Extensions.Strings.html)

+ Coree.NETStandard.HostedServicesCollection
(Coree.NETStandard.HostedServicesCollection

+ Coree.NETStandard.Serilog
(Coree.NETStandard.Serilog.html)

+ Coree.NETStandard.Services
(Coree.NETStandard.Services.html)

+ Coree.NETStandard.Services.RuntimeInsights
(Coree.NETStandard.Services.RuntimeInsights.

+ Coree.NETStandard.SpectreConsole
(Coree.NETStandard.SpectreConsole.html)

Namespace Coree.NETStandard.Extensions.Strings

Enter here to filter...

Classes

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
  - StringExtensions (Coree.NETStandard.Extensions.Strings.Strin
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

StringExtensions (Coree.NETStandard.Extensions.Strings.StringExtensions.html)

Provides extension methods for string operations, enhancing the built-in string manipulation capabilities.

Class StringExtensions

Provides extension methods for string operations, enhancing the built-in string manipulation capabilities.

Enter here to filter...

Inheritance

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
  - object (https://learn.microsoft.com/dotnet/api/system.object)
    - StringExtensions
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
  - StringExtensions (Coree.NETStandard.Extensions.Strings.Strin
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Extensions (Coree.NETStandard.Extensions.html).Strings (Coree.NETStandard.Extensions.Strings.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class StringExtensions
```

Methods

IsValidJson(string)

Validates whether the specified string is valid JSON.

Declaration

```
public static bool IsValidJson(this string jsonString)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*jsonString*	The JSON string to validate.

Returns

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	true if the string is valid JSON; otherwise, false.

IsValidXml(string)

Validates whether the specified string is well-formed XML.

Declaration

```
public static bool IsValidXml(this string xmlString)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*xmlString*	The XML string to validate.

Returns

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	true if the string is a well-formed XML; otherwise, false.

SplitWith(string?, char[], StringSplitOptions)

Splits the input string by an array of characters, treating them as a single connected delimiter, with options to control the split operation.

Declaration

```
public static string[] SplitWith(this string? input, char[] delimiter, StringSplitOptions
    options = StringSplitOptions.None)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*input*	The string to be split.
char (https://learn.microsoft.com/dotnet/api/system.char) []	*delimiter*	The array of characters to form the delimiter.
StringSplitOptions (https://learn.microsoft.com/dotnet/api/system.stringsplitoptions)	*options*	Specifies whether to omit empty or whitespace-only substrings from the resulting array.

Returns

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string) []	An array of substrings that are delimited by the specified sequence of characters.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
  StringExtensions (Coree.NETStandard.Extensions.Strings.Strin
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

SplitWith(string?, string, StringSplitOptions)

Splits the input string by a specified string delimiter, with options to control the split operation.

Declaration

```
public static string[] SplitWith(this string? input, string delimiter, StringSplitOptions
options = StringSplitOptions.None)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*input*	The string to be split.
string (https://learn.microsoft.com/dotnet/api/system.string)	*delimiter*	The string delimiter to split by.
StringSplitOptions (https://learn.microsoft.com/dotnet/api/system.stringsplitoptions)	*options*	Specifies whether to omit empty or whitespace-only substrings from the resulting array.

Returns

Type	Description
string[]	An array of substrings that are delimited by the specified string.

SplitWith(string?, string[], StringSplitOptions)

Splits the input string by multiple string delimiters, with options to control the split operation.

Declaration

```
public static string[] SplitWith(this string? input, string[] delimiters, StringSplitOptio
ns options = StringSplitOptions.None)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*input*	The string to be split.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
StringExtensions
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string) []	*delimiters*	The array of string delimiters to split by.
StringSplitOptions (https://learn.microsoft.com/dotnet/api/system.stringsplitoptions)	*options*	Specifies whether to omit empty or whitespace-only substrings from the resulting array.

Returns

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string) []	An array of substrings that are delimited by any of the specified strings.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
- Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
  - StringExtensions (Coree.NETStandard.Extensions.Strings.Strin
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace Coree.NETStandard.HostedServicesCollection

Enter here to filter...

Classes

HostedServicesCollectionConfig<T>
(Coree.NETStandard.HostedServicesCollection.HostedServicesCollectionConfig-1.html)

Represents a thread-safe collection of configuration items for hosted services. This collection allows for the addition of configuration items and provides a method to fetch and remove items sequentially, supporting concurrent operations.

ServiceCollectionExtensions
(Coree.NETStandard.HostedServicesCollection.ServiceCollectionExtensions.html)

Provides extension methods for IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) to facilitate the registration of hosted services with specific configurations.

ServiceProviderExtensions
(Coree.NETStandard.HostedServicesCollection.ServiceProviderExtensions.html)

Provides a set of extension methods for IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider). These methods extend the functionality of IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider) to offer enhanced and simplified service resolution capabilities, enabling the retrieval and management of service instances in a more flexible manner.

Interfaces

IHostedServicesCollectionConfig<T>
(Coree.NETStandard.HostedServicesCollection.IHostedServicesCollectionConfig-1.html)

Defines a contract for a configuration collection for hosted services, allowing the enqueuing of configuration items and retrieval of the next configuration item.

+ Coree.NETStandard.Abstractions
(Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings
(Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection
(Coree.NETStandard.HostedServicesCollection
HostedServicesCollectionConfig<T>
(Coree.NETStandard.HostedServicesCollecti
1.html)
IHostedServicesCollectionConfig<T>
(Coree.NETStandard.HostedServicesCollecti
html)
ServiceCollectionExtensions
(Coree.NETStandard.HostedServicesCollecti
ServiceProviderExtensions
(Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog
(Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services
(Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights
(Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole
(Coree.NETStandard.SpectreConsole.html)

Class HostedServicesCollectionConfig<T>

Represents a thread-safe collection of configuration items for hosted services. This collection allows for the addition of configuration items and provides a method to fetch and remove items sequentially, supporting concurrent operations.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ HostedServicesCollectionConfig<T>

Implements

IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollection.IHostedServicesCollectionConfig-1.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class HostedServicesCollectionConfig<T> : IHostedServicesCollectionConfig<T> where T : new()
```

Type Parameters

Name	Description
T	The type of the configuration items, which must have a parameterless constructor.

Constructors

HostedServicesCollectionConfig()

Initializes a new instance of the HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollection.HostedServicesCollectionConfig-1.html) class.

Declaration

```
public HostedServicesCollectionConfig()
```

Methods

Enqueue(T)

Adds a new configuration item to the end of the queue.

Declaration

```
public void Enqueue(T item)
```

Parameters

Type	Name	Description
T	*item*	The configuration item to be enqueued.

FetchNextConfig()

Attempts to remove and return the configuration item at the beginning of the queue. If the queue is empty, a new instance of the configuration item type is returned.

Declaration

```
public T FetchNextConfig()
```

Returns

Type	Description
T	The next configuration item if available; otherwise, a new instance of the configuration item type.

Implements

IHostedServicesCollectionConfig<T>
(Coree.NETStandard.HostedServicesCollection.IHostedServicesCollectionConfig-1.html)

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
  HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti
  IHostedServicesCollectionConfig<T> 1.html)
  IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti
  1.html)
  ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
  ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Interface IHostedServicesCollectionConfig<T>

Enter here to filter...

Defines a contract for a configuration collection for hosted services, allowing the enqueuing of configuration items and retrieval of the next configuration item.

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public interface IHostedServicesCollectionConfig<T>
```

Type Parameters

Name	Description
T	The type of the configuration item.

Methods

Enqueue(T)

Enqueues a new configuration item into the collection.

Declaration

```
void Enqueue(T item)
```

Parameters

Type	Name	Description
T	item	The configuration item to be added to the queue.

FetchNextConfig()

Fetches the next configuration item from the queue, removing it from the collection. If the queue is empty, returns a new instance of the configuration item type.

Declaration

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
  HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
  ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
T FetchNextConfig()
```

Returns

Type	Description
T	The next configuration item if available; otherwise, a new instance of the configuration item type.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
  - HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  - IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  - ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
  - ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class ServiceCollectionExtensions

Provides extension methods for IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) to facilitate the registration of hosted services with specific configurations.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ ServiceCollectionExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class ServiceCollectionExtensions
```

Methods

AddHostedServicesCollection<T, K>(IServiceCollection, IConfiguration, string)

Registers a collection of hosted services of type `T` with their configurations loaded from a specified section of the application's IConfiguration (https://learn.microsoft.com/dotnet/api/microsoft.extensions.configuration.iconfiguration).

Declaration

```
public static IServiceCollection AddHostedServicesCollection<T, K>(this IServiceCollection
services, IConfiguration configuration, string sectionName) where T : class, IHostedServic
e where K : class, new()
```

Parameters

Type	Name	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	*services*	The IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) to add the services to.

Enter here to filter...
+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
  HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollectio 1.html)
  IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollectio 1.html)
  ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollectio
  ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollectio
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Name	Description
IConfiguration (https://learn.microsoft.com/dotnet/api/microsoft.extensions.configuration.iconfiguration)	*configuration*	The application's configuration.
string (https://learn.microsoft.com/dotnet/api/system.string)	*sectionName*	The name of the configuration section from which to load the configurations.

Returns

Type	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	The original IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) instance with the hosted services registered.

Type Parameters

Name	Description
T	The type of the hosted service to register. Must implement IHostedService (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostedservice).
K	The configuration type for the hosted service. Must be a class with a parameterless constructor.

Remarks

Configurations are bound to new instances of *K* and each hosted service instance is registered with a singleton service descriptor.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
  HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
  ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
//appsettings.json
"ServiceConfigurations": {
    "FooService": {
      "Name": "Service1"
    },
    "BarService": {
      "Name": "Service2"
    }
  }
services.AddHostedServicesCollection<MyBackgroundSrv, MyBackgroundSrvConfig>(context.Confi
guration, "ServiceConfigurations");

public class MyBackgroundSrvConfig
{
    public Guid Guid { get; } = Guid.NewGuid();  public string? Name { get; set; }
}

public class MyBackgroundSrv : BackgroundService
{
    private readonly MyBackgroundSrvConfig options; private readonly ILogger<MyBackgroundS
ervice> logger;

    public MyBackgroundSrv(ILogger<MyBackgroundService> logger, IHostedServicesCollectionC
onfig<MyBackgroundSrvConfig> options)
    {
        this.logger = logger; this.options = options.FetchNextConfig();
    }

    protected override async Task ExecuteAsync(CancellationToken stoppingToken)
    {
        while (!stoppingToken.IsCancellationRequested)
        {
            try {
                logger.LogInformation("MyBackgroundSrv running: {Name} with {guid} at {tim
e}", options.Name, options.Guid.ToString(), DateTimeOffset.Now);
                await Task.Delay(5000, stoppingToken);
            }
            catch (TaskCanceledException)
            { logger.LogInformation("MyBackgroundSrv cancel: {Name} with {guid} at {tim
e}", options.Name, options.Guid.ToString(), DateTimeOffset.Now); }
        }
    }
}
```

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
  HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti
  1.html)
  IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti
  1.html)
  ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
  ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

AddHostedServicesCollection<T, K>(IServiceCollection, List<K>)

Registers a collection of hosted services of type `T` with their configurations specified in a list of `K`.

Declaration

```
public static IServiceCollection AddHostedServicesCollection<T, K>(this IServiceCollection
services, List<K> values) where T : class, IHostedService where K : class, new()
```

Parameters

Type	Name	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	*services*	The IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) to add the services to.
List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) <K>	*values*	A list of configuration instances for the hosted services.

Returns

Type	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	The original IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) instance with the hosted services registered.

Type Parameters

Name	Description
T	The type of the hosted service to register. Must implement IHostedService (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostedservice).
K	The configuration type for the hosted service. Must be a class with a parameterless constructor.

Remarks

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
  HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
  ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
services.AddHostedServicesCollection<MyBackgroundSrv, MyBackgroundSrvConfig>(new List<MyBa
ckgroundSrvConfig>() {
    new() { Name = "Service1"},
    new() { Name = "Service2" }
});

public class MyBackgroundSrvConfig
{
    public Guid Guid { get; } = Guid.NewGuid();
    public string? Name { get; set; }
}

public class MyBackgroundSrv : BackgroundService
{
    private readonly MyBackgroundSrvConfig options;
    private readonly ILogger<MyBackgroundService> logger;

    public MyBackgroundSrv(ILogger<MyBackgroundService> logger, IHostedServicesCollectionC
onfig<MyBackgroundSrvConfig> options)
    {
        this.logger = logger; this.options = options.FetchNextConfig();
    }

    protected override async Task ExecuteAsync(CancellationToken stoppingToken)
    {
        while (!stoppingToken.IsCancellationRequested)
        {
            try
            {
                logger.LogInformation("MyBackgroundSrv running: {Name} with {guid} at {tim
e}", options.Name, options.Guid.ToString(), DateTimeOffset.Now);
                await Task.Delay(5000, stoppingToken);
            }
            catch (TaskCanceledException)
            {
                logger.LogInformation("MyBackgroundSrv cancel: {Name} with {guid} at {tim
e}", options.Name, options.Guid.ToString(), DateTimeOffset.Now);
            }
        }
    }
}
```

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
  HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti
  1.html)
  IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti
  1.html)
  ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
  ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

AddHostedServicesCollection<T, K>(IServiceCollection, K[])

Registers a collection of hosted services of type `T` with their configurations specified in an array of `K`.

Declaration

```
public static IServiceCollection AddHostedServicesCollection<T, K>(this IServiceCollection
services, K[] values) where T : class, IHostedService where K : class, new()
```

Parameters

Type	Name	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	*services*	The IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) to add the services to.
K[]	*values*	An array of configuration instances for the hosted services.

Returns

Type	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	The original IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) instance with the hosted services registered.

Type Parameters

Name	Description
T	The type of the hosted service to register. Must implement IHostedService (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostedservice).
K	The configuration type for the hosted service. Must be a class with a parameterless constructor.

Remarks

Each hosted service instance is registered with a singleton service descriptor.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
  HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
  ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
services.AddHostedServicesCollection<MyBackgroundSrv, MyBackgroundSrvConfig>(new MyBackgro
undSrvConfig[] {
    new() { Name = "Service1"},
    new() { Name = "Service2" }
});
public class MyBackgroundSrvConfig
{
    public Guid Guid { get; } = Guid.NewGuid();
    public string? Name { get; set; }
}
public class MyBackgroundSrv : BackgroundService
{
    private readonly MyBackgroundSrvConfig options;
    private readonly ILogger<MyBackgroundService> logger;
    public MyBackgroundSrv(ILogger<MyBackgroundService> logger, IHostedServicesCollectionC
onfig<MyBackgroundSrvConfig> options)
    {
        this.logger = logger; this.options = options.FetchNextConfig();
    }
    protected override async Task ExecuteAsync(CancellationToken stoppingToken)
    {
        while (!stoppingToken.IsCancellationRequested)
        {
            try
            {
                logger.LogInformation("MyBackgroundSrv running: {Name} with {guid} at {tim
e}", options.Name, options.Guid.ToString(), DateTimeOffset.Now);
                await Task.Delay(5000, stoppingToken);
            }
            catch (TaskCanceledException)
            {
                logger.LogInformation("MyBackgroundSrv cancel: {Name} with {guid} at {tim
e}", options.Name, options.Guid.ToString(), DateTimeOffset.Now);
            }
        }
    }
}
```

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
  HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti
  1.html)
  IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti
  1.html)
  ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
  ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class ServiceProviderExtensions

Provides a set of extension methods for IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider). These methods extend the functionality of IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider) to offer enhanced and simplified service resolution capabilities, enabling the retrieval and management of service instances in a more flexible manner.

Inheritance

[object](https://learn.microsoft.com/dotnet/api/system.object) (https://learn.microsoft.com/dotnet/api/system.object)
→ ServiceProviderExtensions

Namespace: Coree.NETStandard (Coree.NETStandard.html).HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class ServiceProviderExtensions
```

Methods

GetHostedServiceCollectionOfTypeLazy<T>(IServiceProvider)

Provides a lazy-initialized collection of hosted services of the specified type `T`, designed to be used in scenarios where direct service resolution in constructors may lead to race conditions.

Declaration

```
public static Lazy<IEnumerable<T>> GetHostedServiceCollectionOfTypeLazy<T>(this IServicePr
ovider serviceProvider)
```

Parameters

Type	Name	Description
IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider)	*serviceProvider*	The IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider) from which to retrieve the services.

Returns

▼ Enter here to filter...
+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti
1.html)
IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti
1.html)
ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Description
Lazy (https://learn.microsoft.com/dotnet/api/system.lazy-1)<IEnumerable (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1)<T>>	A Lazy<T> (https://learn.microsoft.com/dotnet/api/system.lazy-1) object that, when evaluated, provides an IEnumerable<T> (https://learn.microsoft.com/dotnet/api/system.collections.generic.ienumerable-1) representing the collection of services of the specified type `T`. If no services of the specified type are registered, an empty enumerable is returned. This allows safe and deferred resolution of services, ideal for use within method implementations to avoid race conditions in service stacks.

Type Parameters

Name	Description
T	The type of the bound service to retrieve. This type parameter is contravariant.

Remarks

This method is particularly useful in complex dependency scenarios, where some services may not be fully initialized at the time of a constructor's execution. By deferring the service resolution to the actual execution time of a method, it mitigates the risk of encountering race conditions within the service initialization stack. It leverages Lazy<T> (https://learn.microsoft.com/dotnet/api/system.lazy-1) to ensure that the service collection is resolved only when it is first needed, thereby enhancing reliability and performance by avoiding unnecessary upfront service resolution and instantiation.

Examples

This example demonstrates using GetHostedServiceCollectionOfTypeLazy<T>(IServiceProvider) within a service constructor to defer the resolution of dependent services, mitigating race conditions:

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.html)
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection.html)
  - HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollection.ServiceProviderExtensions.html#Coree_NETStandard_HostedServicesCollection_ServiceProviderExtensions_GetHostedServiceCollectionOfTypeLazy__1_System_IServiceProvider_)
  - IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollection.html)
  - ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollection.html)
  - ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollection.html)
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
public class MyService :BackgroundService
{
    private readonly Lazy<IEnumerable<MyDependentService>> _dependentServices;

    public MyService(IServiceProvider serviceProvider)
    {
        _dependentServices = serviceProvider.GetHostedServiceCollectionOfTypeLazy<MyDependentService>();
    }

    protected override async Task ExecuteAsync(CancellationToken stoppingToken)
    {
        while (!stoppingToken.IsCancellationRequested)
        {
            // The dependent services are resolved here, when first needed, avoiding race conditions.
            foreach(var service in _dependentServices.Value)
            {
                // Interact with the service
            }
        }
    }
}
```

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
- Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
  - HostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  - IHostedServicesCollectionConfig<T> (Coree.NETStandard.HostedServicesCollecti 1.html)
  - ServiceCollectionExtensions (Coree.NETStandard.HostedServicesCollecti
  - ServiceProviderExtensions (Coree.NETStandard.HostedServicesCollecti
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace Coree.NETStandard.Serilog

Classes

Enter here to filter...

ConditionalLevelSink (Coree.NETStandard.Serilog.ConditionalLevelSink.html)

LoggerSinkConfigurationExtensions (Coree.NETStandard.Serilog.LoggerSinkConfigurationExtensions.html)

OutputTemplates (Coree.NETStandard.Serilog.OutputTemplates.html)

Defines output templates for Serilog logging.

ServiceCollectionExtensions (Coree.NETStandard.Serilog.ServiceCollectionExtensions.html)

Extension methods for configuring logging services.

SourceContextShortEnricher (Coree.NETStandard.Serilog.SourceContextShortEnricher.html)

A Serilog enricher that adds the "SourceContextShort" property of log events by trimming generic type indicators, simplifying namespaces, and optionally removing specified suffixes. This can make the log source context more readable and concise in log outputs.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
  - ConditionalLevelSink (Coree.NETStandard.Serilog.ConditionalLeve
  - LoggerSinkConfigurationExtensions (Coree.NETStandard.Serilog.LoggerSinkCon
  - OutputTemplates (Coree.NETStandard.Serilog.OutputTemplate
  - ServiceCollectionExtensions (Coree.NETStandard.Serilog.ServiceCollectio
  - SourceContextShortEnricher (Coree.NETStandard.Serilog.SourceContextS
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class ConditionalLevelSink

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ ConditionalLevelSink

Implements

ILogEventSink

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Serilog (Coree.NETStandard.Serilog.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class ConditionalLevelSink : ILogEventSink
```

Constructors

ConditionalLevelSink(ILogEventSink, Dictionary<string, LogEventLevel>?)

Declaration

```
public ConditionalLevelSink(ILogEventSink innerSink, Dictionary<string, LogEventLevel>? co
nditionalLevel)
```

Parameters

Type	Name	Description
ILogEventSink	*innerSink*	
Dictionary<string (https://learn.microsoft.com/dotnet/api/system.string), LogEventLevel>? (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)	*conditionalLevel*	

Methods

Emit(LogEvent)

Emit the provided log event to the sink.

Declaration

```
public void Emit(LogEvent logEvent)
```

Parameters

Type	Name	Description
LogEvent	*logEvent*	The log event to write.

Implements

Serilog.Core.ILogEventSink

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
  - ConditionalLevelSink (Coree.NETStandard.Serilog.ConditionalLeve
  - LoggerSinkConfigurationExtensions (Coree.NETStandard.Serilog.LoggerSinkCon
  - OutputTemplates (Coree.NETStandard.Serilog.OutputTemplate
  - ServiceCollectionExtensions (Coree.NETStandard.Serilog.ServiceCollectio
  - SourceContextShortEnricher (Coree.NETStandard.Serilog.SourceContextS
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class LoggerSinkConfigurationExtensions

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ LoggerSinkConfigurationExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Serilog (Coree.NETStandard.Serilog.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class LoggerSinkConfigurationExtensions
```

Methods

ConsoleConditionalLevel(LoggerSinkConfiguration, LogEventLevel, string, IFormatProvider?, LoggingLevelSwitch?, LogEventLevel?, ConsoleTheme?, bool, object?, Dictionary<string, LogEventLevel>?)

Declaration

```
public static LoggerConfiguration ConsoleConditionalLevel(this LoggerSinkConfiguration log
gerSinkConfiguration, LogEventLevel restrictedToMinimumLevel = LogEventLevel.Verbose, string o
utputTemplate = "[{Timestamp:HH:mm:ss} {Level:u3}] {Message:lj}{NewLine}{Exception}", IFor
matProvider? formatProvider = null, LoggingLevelSwitch? levelSwitch = null, LogEventLevel?
standardErrorFromLevel = null, ConsoleTheme? theme = null, bool applyThemeToRedirectedOutp
ut = false, object? syncRoot = null, Dictionary<string, LogEventLevel>? conditionalLevel =
null)
```

Parameters

Type	Name	Description
LoggerSinkConfiguration (Coree.NETStandard.Serilog.LoggerSinkConfiguration.html)	*loggerConfiguration*	
LogEventLevel (Coree.NETStandard.Serilog.LogEventLevel.html)	*restrictedToMinimumLevel*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*outputTemplate*	
IFormatProvider (https://learn.microsoft.com/dotnet/api/system.iformatprovider)	*formatProvider*	
LoggingLevelSwitch (Coree.NETStandard.Serilog.LoggingLevelSwitch.html)	*levelSwitch*	
LogEventLevel? (Coree.NETStandard.Serilog.LogEventLevel.html)	*standardErrorFromLevel*	

Filter: Enter here to filter...

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.html)
+ Coree.NETStandard.Extensions.LoggingLevel (Coree.NETStandard.Extensions.LoggingLevel.html)
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServiceCollection (Coree.NETStandard.HostedServiceCollection.html)
- Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
  - ConditionalLevelSink (Coree.NETStandard.Serilog.ConditionalLeve
  - LoggerSinkConfigurationExtensions (Coree.NETStandard.Serilog.LoggerSinkCon
  - OutputTemplates (Coree.NETStandard.Serilog.OutputTemplate
  - ServiceCollectionExtensions (Coree.NETStandard.Serilog.ServiceCollectio
  - SourceContextShortEnricher (Coree.NETStandard.Serilog.SourceContextS
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Name	Description
ConsoleTheme	*theme*	
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*applyThemeToRedirectedOutput*	
object (https://learn.microsoft.com/dotnet/api/system.object)	*syncRoot*	
Dictionary<string (https://learn.microsoft.com/dotnet/api/system.string), LogEventLevel> (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2)	*conditionalLevel*	

Returns

Type	Description
LoggerConfiguration	

DebugConditionalLevel(LoggerSinkConfiguration, LogEventLevel, string, IFormatProvider?, LoggingLevelSwitch?, Dictionary<string, LogEventLevel>?)

Declaration

```
public static LoggerConfiguration DebugConditionalLevel(this LoggerSinkConfiguration logge
rSinkConfiguration, LogEventLevel restrictedToMinimumLevel = LogEventLevel.Verbose, string out
putTemplate = "[{Timestamp:HH:mm:ss} {Level:u3}] {Message:lj}{NewLine}{Exception}", IForma
tProvider? formatProvider = null, LoggingLevelSwitch? levelSwitch = null, Dictionary<strin
g, LogEventLevel>? conditionalLevel = null)
```

Parameters

Type	Name	Description
LoggerSinkConfiguration (https://learn.microsoft.com/dotnet/api/system.string)	*loggerConfiguration*	
LogEventLevel	*restrictedToMinimumLevel*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*outputTemplate*	
IFormatProvider (https://learn.microsoft.com/dotnet/api/system.iformatprovider)	*formatProvider*	
LoggingLevelSwitch	*levelSwitch*	
Dictionary<string (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2), LogEventLevel>	*conditionalLevel*	

Returns

Type	Description
LoggerConfiguration	

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
  - ConditionalLevelSink (Coree.NETStandard.Serilog.ConditionalLeve
  - LoggerSinkConfigurationExtensions (Coree.NETStandard.Serilog.LoggerSinkCon
  - OutputTemplates (Coree.NETStandard.Serilog.OutputTemplate
  - ServiceCollectionExtensions (Coree.NETStandard.Serilog.ServiceCollectio
  - SourceContextShortEnricher (Coree.NETStandard.Serilog.SourceContextS
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class OutputTemplates

Defines output templates for Serilog logging.

Enter here to filter...

Inheritance

+ Coree.NETStandard.Abstractions
(Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings
(Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection
(Coree.NETStandard.HostedServicesCollection
- Coree.NETStandard.Serilog
(Coree.NETStandard.Serilog.html)
ConditionalLevelSink
(Coree.NETStandard.Serilog.ConditionalLeve
LoggerSinkConfigurationExtensions
(Coree.NETStandard.Serilog.LoggerSinkCon
OutputTemplates
(Coree.NETStandard.Serilog.OutputTemplate
ServiceCollectionExtensions
(Coree.NETStandard.Serilog.ServiceCollectio
SourceContextShortEnricher
(Coree.NETStandard.Serilog.SourceContextS
+ Coree.NETStandard.Services
(Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights
(Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole
(Coree.NETStandard.SpectreConsole.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Serilog (Coree.NETStandard.Serilog.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class OutputTemplates
```

Methods

Default()

Provides the standard output template.

Declaration

```
public static string Default()
```

Returns

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

Remarks

The returned template string is "{Timestamp:HH:mm:ss.ffff} {Level:u3} | {SourceContext} | {EnvironmentUserName} | {EnvironmentName} | {Message:l}{NewLine}{Exception}".

DefaultShort()

Provides a shortened version of the output template.

Declaration

```
public static string DefaultShort()
```

Returns

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

Remarks

The returned template string is "{Timestamp:HH:mm:ss.ffff} | {Level:u3} | {SourceContextShort} | {Message:l} {NewLine}{Exception}".

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
  - ConditionalLevelSink (Coree.NETStandard.Serilog.ConditionalLeve
  - LoggerSinkConfigurationExtensions (Coree.NETStandard.Serilog.LoggerSinkCon
  - OutputTemplates (Coree.NETStandard.Serilog.OutputTemplate
  - ServiceCollectionExtensions (Coree.NETStandard.Serilog.ServiceCollectio
  - SourceContextShortEnricher (Coree.NETStandard.Serilog.SourceContextS
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class ServiceCollectionExtensions

Extension methods for configuring logging services.

Enter here to filter...

Inheritance

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
  - ConditionalLevelSink (Coree.NETStandard.Serilog.ConditionalLeve
  - LoggerSinkConfigurationExtensions (Coree.NETStandard.Serilog.LoggerSinkConf
  - OutputTemplates (Coree.NETStandard.Serilog.OutputTemplate
  - ServiceCollectionExtensions (Coree.NETStandard.Serilog.ServiceCollectio
  - SourceContextMinimum (Coree.NETStandard.Serilog.SourceContextS
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Serilog (Coree.NETStandard.Serilog.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class ServiceCollectionExtensions
```

Methods

AddLoggingCoreeNETStandard(IServiceCollection, bool, Dictionary<string, LogEventLevel>?)

Adds and configures Serilog logging to the specified IServiceCollection, incorporating a conditional log level feature.

Declaration

```
public static IServiceCollection AddLoggingCoreeNETStandard(this IServiceCollection servic
es, bool simplifyNamespace = true, Dictionary<string, LogEventLevel>? conditionalLevel = n
ull)
```

Parameters

Type	Name	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	*services*	The IServiceCollection to which logging services are added.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*simplifyNamespace*	If true, simplifies the namespace to only include the last component.

Type	Name	Description
Dictionary (https://learn.microsoft.com/dotnet/api/system.collections.generic.dictionary-2) <string (https://learn.microsoft.com/dotnet/api/system.string), LogEventLevel>	*conditionalLevel*	An optional dictionary mapping source contexts to LogEventLevels, allowing for dynamic log level overrides. Example usage: `new Dictionary<string, LogEventLevel> { ["Coree.SomeService"] = LogEventLevel.Debug, };` This parameter enables conditional log level adjustment based on the source context, offering fine-grained control over log verbosity without direct reference to the source contexts at runtime.

Returns

Type	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	The original IServiceCollection, supporting method chaining.

Remarks

Configures Serilog as the primary logging provider, leveraging a LoggingLevelSwitch for default level control. It enriches log entries with contextual information and enables output to both console and debug sinks, using a concise output format. This setup ensures that logs are both informative and manageable, tailored to development and production environments.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.StringBuilder (Coree.NETStandard.Extensions.StringBuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
  - ConditionalLevelSink (Coree.NETStandard.Serilog.ConditionalLeve
  - LoggerSinkConfigurationExtensions (Coree.NETStandard.Serilog.LoggerSinkCon
  - OutputTemplates (Coree.NETStandard.Serilog.OutputTemplate
  - ServiceCollectionExtensions (Coree.NETStandard.Serilog.ServiceCollectio
  - SourceContextShortEnricher (Coree.NETStandard.Serilog.SourceContextS
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class SourceContextShortEnricher

A Serilog enricher that adds the "SourceContextShort" property of log events by trimming generic type indicators, simplifying namespaces, and optionally removing specified suffixes. This can make the log source context more readable and concise in log outputs.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ SourceContextShortEnricher

Implements

ILogEventEnricher

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Serilog (Coree.NETStandard.Serilog.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class SourceContextShortEnricher : ILogEventEnricher
```

Constructors

SourceContextShortEnricher()

Initializes a new instance of the SourceContextShortEnricher (Coree.NETStandard.Serilog.SourceContextShortEnricher.html) class with default settings. By default, it trims generic type indicators, simplifies namespaces to their last component, applies a default padding, and removes common suffixes like "Service", "Services", "AsyncCommand", and "Command".

Declaration

```
public SourceContextShortEnricher()
```

SourceContextShortEnricher(bool, bool, int, string[]?)

Initializes a new instance of the SourceContextShortEnricher (Coree.NETStandard.Serilog.SourceContextShortEnricher.html) class with customizable settings.

Declaration

```
public SourceContextShortEnricher(bool trimGenericTypeIndicator = true, bool simplifyNames
pace = true, int sourceContextPadding = 15, string[]? suffixesToRemove = null)
```

Parameters

Type	Name	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*trimGenericTypeIndicator*	If true, trims the generic type indicator from the source context.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*simplifyNamespace*	If true, simplifies the namespace to only include the last component.
int (https://learn.microsoft.com/dotnet/api/system.int32)	*sourceContextPadding*	The padding applied to the modified source context for alignment in logs.
string (https://learn.microsoft.com/dotnet/api/system.string) []	*suffixesToRemove*	An array of suffixes to remove from the source context, enhancing readability.

Methods

Enrich(LogEvent, ILogEventPropertyFactory)

A Serilog enricher that adds the "SourceContextShort" property of log events by trimming generic type indicators, simplifying namespaces, and optionally removing specified suffixes. This can make the log source context more readable and concise in log outputs.

Declaration

```
public void Enrich(LogEvent logEvent, ILogEventPropertyFactory propertyFactory)
```

Parameters

Type	Name	Description
LogEvent	*logEvent*	The log event to enrich.
ILogEventPropertyFactory	*propertyFactory*	The factory used to create new log event properties.

Implements

Serilog.Core.ILogEventEnricher

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
- Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
  - ConditionalLevelSink (Coree.NETStandard.Serilog.ConditionalLeve
  - LoggerSinkConfigurationExtensions (Coree.NETStandard.Serilog.LoggerSinkCon
  - OutputTemplates (Coree.NETStandard.Serilog.OutputTemplate
  - ServiceCollectionExtensions (Coree.NETStandard.Serilog.ServiceCollectio
  - SourceContextShortEnricher (Coree.NETStandard.Serilog.SourceContextS
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Namespace Coree.NETStandard.Services

Classes

Enter here to filter...

FileService (Coree.NETStandard.Services.FileService.html)

Defines a service for file system operations.

ProcessRunResult (Coree.NETStandard.Services.ProcessRunResult.html)

Represents the result of running an external process.

ProcessService (Coree.NETStandard.Services.ProcessService.html)

Provides functionality for managing and executing external processes. This service supports a variety of operations including running processes with options for cancellation, timeouts, and potentially more features in the future.

RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsightsService.html)

Interfaces

IFileService (Coree.NETStandard.Services.IFileService.html)

Defines a service for file system operations.

IProcessService (Coree.NETStandard.Services.IProcessService.html)

Defines a service for running external processes with support for cancellation and timeouts.

IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsights.html)

Enums

ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExitCodeState.html)

Represents various outcome states for process execution operations within the service. This enum is designed to be flexible and applicable to a range of process-related methods, providing a standardized way to communicate success, failure, and other states across different types of process executions.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessServic
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExit
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRe
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

+ Coree.NETStandard.Abstractions
(Coree.NETStandard.Abstractions.html)

+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)

+ Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)

+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)

+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)

+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht

+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h

+ Coree.NETStandard.Extensions.Strings
(Coree.NETStandard.Extensions.Strings.html)

+ Coree.NETStandard.HostedServicesCollection
(Coree.NETStandard.HostedServicesCollection

+ Coree.NETStandard.Serilog
(Coree.NETStandard.Serilog.html)

- Coree.NETStandard.Services
(Coree.NETStandard.Services.html)

FileService
(Coree.NETStandard.Services.FileService.htm
IFileService
(Coree.NETStandard.Services.IFileService.htm
IProcessService
(Coree.NETStandard.Services.IProcessServic
IRuntimeInsights
(Coree.NETStandard.Services.IRuntimeInsigh
ProcessRunExitCodeState
(Coree.NETStandard.Services.ProcessRunExi
ProcessRunResult
(Coree.NETStandard.Services.ProcessRunRes
ProcessService
(Coree.NETStandard.Services.ProcessService
RuntimeInsightsService
(Coree.NETStandard.Services.RuntimeInsigh

Class FileService

Defines a service for file system operations.

Enter here to filter...

Inheritance

object (https://learn.microsoft.com/dotnet/api/system.object)

↳ FileService

Implements

IFileService (Coree.NETStandard.Services.IFileService.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Services (Coree.NETStandard.Services.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class FileService : IFileService
```

Constructors

FileService(ILogger<FileService>)

Defines a service for file system operations.

Declaration

```
public FileService(ILogger<FileService> logger)
```

Parameters

Type	Name	Description
ILogger (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.ilogger-1)<FileService (Coree.NETStandard.Services.FileService.html)>	*logger*	

Methods

GetCorrectCasedPath(string?)

Declaration

```
public string? GetCorrectCasedPath(string? path)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*path*	

Returns

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

GetCorrectCasedPathAsync(string?, CancellationToken)

Corrects the casing of a specified file or directory path based on the actual casing on the file system.

Declaration

```
public Task<string> GetCorrectCasedPathAsync(string? path, CancellationToken cancellation
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*path*	The path to correct. Can be either a file or directory path.
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task<string> (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)	The path with corrected casing if the file or directory exists. Returns null for null or empty input, returns the original path if the path does not exist or if the file system is case-sensitive and not NTFS or FAT.

Remarks

This method checks the existence of the path and then corrects its casing to match the case used in the file system. It considers the file system's case sensitivity: on NTFS or FAT (case-insensitive file systems), it corrects the path casing; otherwise, it assumes a case-sensitive file system and returns the original path if the drive format is not recognized. The method processes each component of the path to ensure the entire path is correctly cased from root to leaf.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  FileService (Coree.NETStandard.Services.FileService.htm
  IFileService (Coree.NETStandard.Services.IFileService.htm
  IProcessService (Coree.NETStandard.Services.IProcessService
  IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExi
  ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  ProcessService (Coree.NETStandard.Services.ProcessService
  RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

IsCommandAvailable(string?)

Declaration

```
public string? IsCommandAvailable(string? path)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*path*	

Returns

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

IsCommandAvailableAsync(string?, CancellationToken)

Declaration

```
public Task<string?> IsCommandAvailableAsync(string? command, CancellationToken cancellati
onToken)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*command*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)< string (https://learn.microsoft.com/dotnet/api/system.string)>	

IsExecutableFilePresent(string?, string)

Declaration

```
public string? IsExecutableFilePresent(string? command, string path)
```

Parameters

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessService
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExit
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*command*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*path*	

Returns

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

IsExecutableFilePresentAsync(string?, string)

Declaration

```
public Task<string?> IsExecutableFilePresentAsync(string? command, string path)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*command*	
string (https://learn.microsoft.com/dotnet/api/system.string)	*path*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1) < string (https://learn.microsoft.com/dotnet/api/system.string) >	

Implements

FileService (Coree.NETStandard.Services.FileService.htm

IFileService (Coree.NETStandard.Services.IFileService.html)

IFileService (Coree.NETStandard.Services.IFileService.htm

IProcessService (Coree.NETStandard.Services.IProcessService

IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh

ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExi

ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes

ProcessService (Coree.NETStandard.Services.ProcessService

RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)

Interface IFileService

Defines a service for file system operations.

Enter here to filter...

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessService
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExi
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Services (Coree.NETStandard.Services.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public interface IFileService
```

Methods

GetCorrectCasedPath(string?)

Declaration

```
string? GetCorrectCasedPath(string? path)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*path*	

Returns

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

GetCorrectCasedPathAsync(string?, CancellationToken)

Declaration

```
Task<string?> GetCorrectCasedPathAsync(string? path, CancellationToken cancellationToken)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*path*	

Type	Name	Description
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)<string (https://learn.microsoft.com/dotnet/api/system.string)>	

IsCommandAvailable(string?)

Declaration

```
string? IsCommandAvailable(string? command)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*command*	

Returns

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

IsCommandAvailableAsync(string?, CancellationToken)

Declaration

```
Task<string?> IsCommandAvailableAsync(string? command, CancellationToken cancellationToken)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*command*	
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	

Returns

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  FileService (Coree.NETStandard.Services.FileService.htm
  IFileService (Coree.NETStandard.Services.IFileService.htm
  IProcessService (Coree.NETStandard.Services.IProcessService
  IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExit
  ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  ProcessService (Coree.NETStandard.Services.ProcessService
  RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1) < string (https://learn.microsoft.com/dotnet/api/system.string) >	

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessServic
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExi
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Interface IProcessService

Defines a service for running external processes with support for cancellation and timeouts.

Enter here to filter...

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Services (Coree.NETStandard.Services.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public interface IProcessService
```

Methods

RunProcessWithCancellationSupportAsync(string, string, string, bool, CancellationToken, TimeSpan?)

Runs an external process asynchronously with options for cancellation and timeout.

Declaration

```
Task<ProcessRunResult> RunProcessWithCancellationSupportAsync(string fileName, string argu
ments, string workingDirectory, bool killOnCancel = false, CancellationToken cancellationW
aitRequest = default, TimeSpan? timeout = null)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*fileName*	The name of the executable to run.
string (https://learn.microsoft.com/dotnet/api/system.string)	*arguments*	The arguments to pass to the executable.
string (https://learn.microsoft.com/dotnet/api/system.string)	*workingDirectory*	The working directory for the process.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*killOnCancel*	Indicates whether the process should be killed if a cancellation is requested.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessService
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExit
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Type	Name	Description
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationWaitRequest*	The cancellation token that can be used to request cancellation of the operation.
TimeSpan (https://learn.microsoft.com/dotnet/api/system.timespan) ?	*timeout*	Optional timeout after which the process will be killed if not completed.

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1) < ProcessRunResult (Coree.NETStandard.Services.ProcessRunResult.html) >	A task that represents the asynchronous operation, containing the result of the process execution.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessService
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExit
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Interface IRuntimeInsights

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Services (Coree.NETStandard.Services.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public interface IRuntimeInsights
```

Properties

CallingAssemblyName

Declaration

```
string? CallingAssemblyName { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

CurrentDirectory

Declaration

```
string? CurrentDirectory { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

CurrentProcessMainModuleFilename

Declaration

```
string? CurrentProcessMainModuleFilename { get; }
```

Property Value

Enter here to filter...
+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
FileService (Coree.NETStandard.Services.FileService.htm
IFileService (Coree.NETStandard.Services.IFileService.htm
IProcessService (Coree.NETStandard.Services.IProcessServic
IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExi
ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
ProcessService (Coree.NETStandard.Services.ProcessService
RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

CurrentProcessMainModuleName

Declaration

```
string? CurrentProcessMainModuleName { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

EntryAssemblyName

Declaration

```
string? EntryAssemblyName { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

EnvironmentCommandLine

Declaration

```
string? EnvironmentCommandLine { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

ExecutingAssemblyName

Declaration

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessServic
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExi
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

```
string? ExecutingAssemblyName { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

OSArchitecture

Declaration

```
string? OSArchitecture { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

ProcessArchitecture

Declaration

```
string? ProcessArchitecture { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

Username

Declaration

```
string? Username { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessServic
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExit
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessServic
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExi
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Enum ProcessRunExitCodeState

Represents various outcome states for process execution operations within the service. This enum is designed to be flexible and applicable to a range of process-related methods, providing a standardized way to communicate success, failure, and other states across different types of process executions.

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Services (Coree.NETStandard.Services.html)

Assembly: Coree.NETStandard.dll

Syntax

```
[Flags]
public enum ProcessRunExitCodeState
```

Fields

Name	Description
IsCanceled	Indicates the operation was canceled, either through user action or a timeout. This state reflects an intentional interruption of the process before completion.
IsFailedStart	Indicates the process failed to start, which could result from various issues such as executable not found, insufficient permissions, or invalid arguments.
IsValid	Indicates a valid exit code has been set, suggesting the process has concluded and an outcome is available for evaluation. This state is intended to signify the end of an operation, whether successful or not.
IsValidErrorCode	Indicates the process ran but completed with an error, typically represented by a non-zero exit code. This state signifies an unsuccessful execution according to the process's own criteria.
IsValidSuccess	Indicates a successful execution where the process completed as expected without any errors, typically represented by an exit code of 0.
None	The default state, indicating no specific outcome has been determined. This state is used before an operation is performed or when an operation's outcome does not fit any other defined state.

Class ProcessRunResult

Represents the result of running an external process.

Enter here to filter...

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ ProcessRunResult

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Services (Coree.NETStandard.Services.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class ProcessRunResult
```

Properties

Arguments

Gets or sets the arguments passed to the executable.

Declaration

```
public string? Arguments { get; set; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

Commandline

Gets or sets the command line used to start the process.

Declaration

```
public string? Commandline { get; set; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

ExitCode

Gets or sets the exit code returned by the process.

Declaration

```
public int ExitCode { get; set; }
```

Property Value

Type	Description
int (https://learn.microsoft.com/dotnet/api/system.int32)	

ExitCodeState

Gets or sets the state of the exit code indicating the outcome of the process run.

Declaration

```
public ProcessRunExitCodeState ExitCodeState { get; set; }
```

Property Value

Type	Description
ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExitCodeState.html)	

Filename

Gets or sets the filename of the executable that was run.

Declaration

```
public string? Filename { get; set; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

Output

Gets or sets the output (both stdout and stderr) of the process.

Declaration

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessService
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeIns
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExit
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh
- Coree.NETStandard.Services.RuntimeInsights

```
public string? Output { get; set; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessServic
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExi
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Class ProcessService

Provides functionality for managing and executing external processes. This service supports a variety of operations including running processes with options for cancellation, timeouts, and potentially more features in the future.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ ProcessService

Implements

IProcessService (Coree.NETStandard.Services.IProcessService.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Services (Coree.NETStandard.Services.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class ProcessService : IProcessService
```

Constructors

ProcessService(ILogger<ProcessService>)

Initializes a new instance of the ProcessService (Coree.NETStandard.Services.ProcessService.html) class, setting up logging capabilities to monitor and log process execution details. This constructor ensures the necessary dependencies are in place for the service to operate.

Declaration

```
public ProcessService(ILogger<ProcessService> logger)
```

Parameters

Type	Name	Description
ILogger<ProcessService> (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.ILogger-1)	*logger*	The logger used to log information, warnings, and errors related to process management and execution.

Exceptions

Type	Condition
ArgumentNullException (https://learn.microsoft.com/dotnet/api/system.argumentnullexception)	Thrown if the provided `logger` is null.

Methods

RunProcessWithCancellationSupportAsync(string, string, string, bool, CancellationToken, TimeSpan?)

Runs an external process asynchronously with options for cancellation and timeout.

Declaration

```
public Task<ProcessRunResult> RunProcessWithCancellationSupportAsync(string fileName, string arguments, string workingDirectory, bool killOnCancel = false, CancellationToken cancellationWaitRequest = default, TimeSpan? timeout = null)
```

Parameters

Type	Name	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	*fileName*	The name of the executable to run.
string (https://learn.microsoft.com/dotnet/api/system.string)	*arguments*	The arguments to pass to the executable.
string (https://learn.microsoft.com/dotnet/api/system.string)	*workingDirectory*	The working directory for the process.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*killOnCancel*	Indicates whether the process should be killed if a cancellation is requested.
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationWaitRequest*	The cancellation token that can be used to request cancellation of the operation.
TimeSpan? (https://learn.microsoft.com/dotnet/api/system.timespan)	*timeout*	Optional timeout after which the process will be killed if not completed.

Returns

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Abstractions.Models (Coree.NETStandard.Abstractions.Models.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  FileService (Coree.NETStandard.Services.FileService.htm
  IFileService (Coree.NETStandard.Services.IFileService.htm
  IProcessService (Coree.NETStandard.Services.IProcessService
  IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExit
  ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  ?
  ProcessService (Coree.NETStandard.Services.ProcessService
  RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1) < ProcessRunResult (Coree.NETStandard.Services.ProcessRunResult.html) >	A task that represents the asynchronous operation, containing the result of the process execution.

Implements

IProcessService (Coree.NETStandard.Services.IProcessService.html)

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessServic
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExi
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Class RuntimeInsightsService

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ RuntimeInsightsService

Implements

IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsights.html)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Services (Coree.NETStandard.Services.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class RuntimeInsightsService : IRuntimeInsights
```

Constructors

RuntimeInsightsService(ILogger<IRuntimeInsights>)

Declaration

```
public RuntimeInsightsService(ILogger<IRuntimeInsights> logger)
```

Parameters

Type	Name	Description
ILogger (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.ilogger-1)<IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsights.html)>	*logger*	

Properties

CallingAssemblyName

Declaration

```
public string? CallingAssemblyName { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

CurrentDirectory

Declaration

```
public string? CurrentDirectory { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

CurrentProcessMainModuleFilename

Declaration

```
public string? CurrentProcessMainModuleFilename { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

CurrentProcessMainModuleName

Declaration

```
public string? CurrentProcessMainModuleName { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

EntryAssemblyName

Declaration

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessServic
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExi
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

```
public string? EntryAssemblyName { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

EnvironmentCommandLine

Declaration

```
public string? EnvironmentCommandLine { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

ExecutingAssemblyName

Declaration

```
public string? ExecutingAssemblyName { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

OSArchitecture

Declaration

```
public string? OSArchitecture { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessServic
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExit
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

ProcessArchitecture

Declaration

```
public string? ProcessArchitecture { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

Username

Declaration

```
public string? Username { get; }
```

Property Value

Type	Description
string (https://learn.microsoft.com/dotnet/api/system.string)	

Implements

IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsights.html)

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
- Coree.NETStandard.Services (Coree.NETStandard.Services.html)
  - FileService (Coree.NETStandard.Services.FileService.htm
  - IFileService (Coree.NETStandard.Services.IFileService.htm
  - IProcessService (Coree.NETStandard.Services.IProcessService
  - IRuntimeInsights (Coree.NETStandard.Services.IRuntimeInsigh
  - ProcessRunExitCodeState (Coree.NETStandard.Services.ProcessRunExit
  - ProcessRunResult (Coree.NETStandard.Services.ProcessRunRes
  - ProcessService (Coree.NETStandard.Services.ProcessService
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh

Namespace Coree.NETStandard.Services.RuntimeInsights

Enter here to filter...

Classes

RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsights.RuntimeInsightsService.html)

Represents a service for RuntimeInsightsService methods with optional dependency injection support. This service implements the IRuntimeInsightsService interface, providing methods reflection of the running system and environment. This service inherits from DependencySingleton<PInvokeService>, which supports both dependency injection (DI) and non-DI scenarios

Interfaces

IRuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsights.IRuntimeInsightsService.html)

Represents a service for retrieving runtime insights.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
  - IRuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Interface IRuntimeInsightsService

Represents a service for retrieving runtime insights.

Enter here to filter...

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Services (Coree.NETStandard.Services.html).RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public interface IRuntimeInsightsService
```

Methods

IsDebugBuild()

Checks if the current build is a debug build.

Declaration

```
bool? IsDebugBuild()
```

Returns

Type	Description
bool? (https://learn.microsoft.com/dotnet/api/system.boolean)	True if the current build is a debug build; otherwise, false. Returns null if the determination cannot be made.

IsDebugBuildAsync(CancellationToken)

Checks if the current build is a debug build.

Declaration

```
Task<bool?> IsDebugBuildAsync(CancellationToken cancellationToken = default)
```

Parameters

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
  - IRuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh
  - IsDebugBuildAsync(CancellationToken) (Coree.NETStandard.Services.RuntimeInsigh
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Type	Name	Description
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	The cancellation token to cancel the operation.

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1) < bool (https://learn.microsoft.com/dotnet/api/system.boolean) >	A task representing the asynchronous operation. The task result contains true if the current build is a debug build; otherwise, false. If the operation is canceled or an exception occurs, returns null.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
  - IRuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh
  - RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Class RuntimeInsightsService

Represents a service for RuntimeInsightsService methods with optional dependency injection support. This service implements the IRuntimeInsightsService interface, providing methods reflection of the running system and environment. This service inherits from DependencySingleton<PInvokeService>, which supports both dependency injection (DI) and non-DI scenarios

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ DependencySingleton (Coree.NETStandard.Abstractions.DependencySingleton-1.html)<RuntimeInsights.RuntimeInsightsService>
↳ RuntimeInsightsService

Implements

IRuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsights.IRuntimeInsightsService.html)
IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)

Inherited Members

DependencySingleton<RuntimeInsightsService>.SetMinimumLogLevel(LogLevel) (Coree.NETStandard.Abstractions.DependencySingleton-1.html#Coree_NETStandard_Abstractions_DependencySingleton_1_SetMinimumLogLevel_Microsoft_Extensions_Logging_LogLevel_)
DependencySingleton<RuntimeInsightsService>.Instance (Coree.NETStandard.Abstractions.DependencySingleton-1.html#Coree_NETStandard_Abstractions_DependencySingleton_1_Instance)
DependencySingleton<RuntimeInsightsService>.logger (Coree.NETStandard.Abstractions.DependencySingleton-1.html#Coree_NETStandard_Abstractions_DependencySingleton_1_logger)
DependencySingleton<RuntimeInsightsService>.configuration (Coree.NETStandard.Abstractions.DependencySingleton-1.html#Coree_NETStandard_Abstractions_DependencySingleton_1_configuration)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).Services (Coree.NETStandard.Services.html).RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class RuntimeInsightsService : DependencySingleton<RuntimeInsightsService>, IRuntimeInsightsService, IDependencySingleton
```

Constructors

RuntimeInsightsService(ILogger<RuntimeInsightsService>, IConfiguration)

Initializes a new instance of the untimeInsightsService class with the specified logger and configuration.

Declaration

```
public RuntimeInsightsService(ILogger<RuntimeInsightsService> logger, IConfiguration confi
guration)
```

Parameters

Type	Name	Description
ILogger<RuntimeInsightsService> (https://learn.microsoft.com/dotnet/api/microsoft.extensions.logging.ilogger-1)	*logger*	The logger instance for logging messages.
IConfiguration (https://learn.microsoft.com/dotnet/api/microsoft.extensions.configuration.iconfiguration)	*configuration*	The configuration instance for accessing application settings.

Methods

IsDebugBuild()

Checks if the current build is a debug build.

Declaration

```
public bool? IsDebugBuild()
```

Returns

Type	Description
bool? (https://learn.microsoft.com/dotnet/api/system.boolean)	True if the current build is a debug build; otherwise, false. If the operation is canceled or an exception occurs, returns null.

IsDebugBuildAsync(CancellationToken)

Checks if the current build is a debug build.

Declaration

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
  IRuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh
  RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

```
public Task<bool?> IsDebugBuildAsync(CancellationToken cancellationToken = default)
```

Parameters

Type	Name	Description
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	The cancellation token to cancel the operation.

Returns

Type	Description
Task<bool?> (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task-1)	True if the current build is a debug build; otherwise, false. If the operation is canceled or an exception occurs, returns null.

IsDevelopmentBuild()

Checks if the current build is a debug build.

Declaration

```
public static bool IsDevelopmentBuild()
```

Returns

Type	Description
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	True if the current build is a debug build; otherwise, false.

Implements

IRuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsights.IRuntimeInsightsService.html)
IDependencySingleton (Coree.NETStandard.Abstractions.IDependencySingleton.html)

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
- Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
  IRuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh
  RuntimeInsightsService (Coree.NETStandard.Services.RuntimeInsigh
+ Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)

+ Coree.NETStandard.Abstractions
(Coree.NETStandard.Abstractions.html)

+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)

+ Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)

+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)

+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)

+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht

+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h

+ Coree.NETStandard.Extensions.Strings
(Coree.NETStandard.Extensions.Strings.html)

+ Coree.NETStandard.HostedServicesCollection
(Coree.NETStandard.HostedServicesCollection

+ Coree.NETStandard.Serilog
(Coree.NETStandard.Serilog.html)

+ Coree.NETStandard.Services
(Coree.NETStandard.Services.html)

- Coree.NETStandard.Services.RuntimeInsights
(Coree.NETStandard.Services.RuntimeInsights.

  IRuntimeInsightsService
  (Coree.NETStandard.Services.RuntimeInsigh
  RuntimeInsightsService
  (Coree.NETStandard.Services.RuntimeInsigh

+ Coree.NETStandard.SpectreConsole
(Coree.NETStandard.SpectreConsole.html)

Namespace Coree.NETStandard.Spectre Console

Enter here to filter...

Classes

+ Coree.NETStandard.Abstractions
(Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.B
(Coree.NETStandard.Extensions.B.html)
+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings
(Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection
(Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog
(Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services
(Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights
(Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole
(Coree.NETStandard.SpectreConsole.html)
CommandAppExtensions
(Coree.NETStandard.SpectreConsole.Comma
HostBuilderExtensions
(Coree.NETStandard.SpectreConsole.HostBu
ServiceCollectionExtensions
(Coree.NETStandard.SpectreConsole.Service
SpectreConsoleHostedService
(Coree.NETStandard.SpectreConsole.Spectre
SpectreConsoleHostedService.ExitCode
(Coree.NETStandard.SpectreConsole.Spectre
SpectreConsoleTypeRegistrar
(Coree.NETStandard.SpectreConsole.Spectre
SpectreConsoleTypeResolver
(Coree.NETStandard.SpectreConsole.Spectre

CommandAppExtensions
(Coree.NETStandard.SpectreConsole.CommandAppExtensions.html)

Provides extension methods for Spectre.Console.Cli.ICommandApp.

HostBuilderExtensions
(Coree.NETStandard.SpectreConsole.HostBuilderExtensions.html)

Provides extension methods for IHostBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostbuilder) to facilitate the integration of Spectre.Console's CommandApp into the application's host builder.

ServiceCollectionExtensions
(Coree.NETStandard.SpectreConsole.ServiceCollectionExtensions.html)

Provides extension methods for IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) to facilitate the integration of Spectre.Console's CommandApp.

SpectreConsoleHostedService
(Coree.NETStandard.SpectreConsole.SpectreConsoleHostedService.html)

Provides a hosted service for running Spectre.Console CLI applications within a .NET Core generic host. This class handles the application lifecycle, processes commands, and manages application exit codes based on execution results.

SpectreConsoleTypeRegistrar
(Coree.NETStandard.SpectreConsole.SpectreConsoleTypeRegistrar.html)

Enables type registration for Spectre.Console applications, supporting both IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider) and IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection).

SpectreConsoleTypeResolver
(Coree.NETStandard.SpectreConsole.SpectreConsoleTypeResolver.html)

Resolves types from an IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider), and optionally disposes it.

Enums

SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.SpectreConsoleHostedService.ExitCode.html)

Defines exit codes for the SpectreConsoleHostedService application. These exit codes communicate the outcome of the application's execution to the calling process, allowing for scripted interactions and error handling.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Class CommandAppExtensions

Provides extension methods for Spectre.Console.Cli.ICommandApp.

Enter here to filter...

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ CommandAppExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class CommandAppExtensions
```

Methods

GetCommandTypes(ICommandApp)

Retrieves a list of command types registered with the Spectre.Console.Cli.ICommandApp. This method uses reflection to access the private configurator field within the Spectre.Console.Cli.ICommandApp instance, then extracts and returns the types of the registered commands.

Declaration

```
public static List<Type> GetCommandTypes(this ICommandApp commandApp)
```

Parameters

Type	Name	Description
ICommandApp	commandApp	The Spectre.Console.Cli.ICommandApp instance to extract command types from.

Returns

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Type	Description
List (https://learn.microsoft.com/dotnet/api/system.collections.generic.list-1) < Type (https://learn.microsoft.com/dotnet/api/system.type) >	A list of Type (https://learn.microsoft.com/dotnet/api/system.type) objects representing the command types registered with the Spectre.Console.Cli.ICommandApp.

Remarks

This method relies on the internal structure of the Spectre.Console.Cli.ICommandApp implementation. If the implementation changes in future versions, this method may no longer work as expected.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Class HostBuilderExtensions

Provides extension methods for IHostBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostbuilder) to facilitate the integration of Spectre.Console's CommandApp into the application's host builder.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ HostBuilderExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class HostBuilderExtensions
```

Methods

UseSpectreConsole(IHostBuilder, Action<IConfigurator>, bool)

Configures the application's host builder to use Spectre.Console's Spectre.Console.Cli.CommandApp and related services. This setup enables the application to utilize Spectre.Console's command line interface tools, allowing for the definition, parsing, and execution of commands within a .NET Core application.

Declaration

```
public static IHostBuilder UseSpectreConsole(this IHostBuilder builder, Action<IConfigurator> configureCommandApp, bool suppressLifeTimeStatusMessages = true)
```

Parameters

Type	Name	Description
IHostBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostbuilder)	*builder*	The IHostBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostbuilder) to configure.

Type	Name	Description
Action (https://learn.microsoft.com/dotnet/api/system.action-1)<IConfigurator>	*configureCommandApp*	An Action<T> (https://learn.microsoft.com/dotnet/api/system.action-1) delegate that configures the Spectre.Console.Cli.CommandApp.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*suppressLifeTimeStatusMessages*	Supresses default output of the application host.

Returns

Type	Description
IHostBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostbuilder)	The IHostBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostbuilder) so that additional configuration calls can be chained.

Remarks

This method performs the following operations:

- Creates and configures a new instance of Spectre.Console.Cli.CommandApp using the provided `configureCommandApp` action.
- Scans for command types within the Spectre.Console.Cli.CommandApp and registers them as singletons in the service collection.
- Registers the Spectre.Console.Cli.CommandApp as a singleton service, configured to use a Spectre ConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.SpectreConsoleTypeRegistrar.html) for dependency injection.
- Adds a SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.SpectreConsoleHostedService.html) to the services collection to manage the lifecycle of the Spectre.Console.Cli.CommandApp within the host application.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

```
static async Task Main(string[] args)
{
    var builder = Host.CreateDefaultBuilder(args);
    builder.ConfigureServices(services =>
    {
        services.AddSingleton<ISomeServiceDependency, SomeServiceDependency>();
        services.AddHostedService<SomeOtherBackgroundService>();
    });
    builder.UseSpectreConsole(configCommandApp =>
    {
        configCommandApp.SetApplicationName("testapp");
        configCommandApp.AddCommand<MyAsyncCommand>("my-command");
    });
    await builder.Build().RunAsync();
}

public class MyAsyncCommand : AsyncCommand<MyAsyncCommand.Settings>
{
    private readonly ISomeServiceDependency dependencyService; private readonly IHostAppli
cationLifetime appLifetime;

    public MyAsyncCommand(ISomeServiceDependency dependencyService, IHostApplicationLifeti
me appLifetime)
    {
        this.dependencyService = dependencyService; this.appLifetime = appLifetime;
    }

    public class Settings : CommandSettings
    {
        [Description("The commandline setting")]
        [CommandArgument(0, "<Setting>")]
        public string? SomeSetting { get; init; }
    }

    public override async Task<int> ExecuteAsync(CommandContext context, Settings setting
s)
    {
        var exitCode = dependencyService.PerformConsoleAction(appLifetime.ApplicationStoppin
g);
        catch (Exception) { return (int)SpectreConsoleHostedService.ExitCode.CommandTermin
ated; }
        return (int)SpectreConsoleHostedService.ExitCode.SuccessAndContinue;
    }
}
```

Exceptions

Type	Condition
ArgumentNullException (https://learn.microsoft.com/dotnet/api/system.argumentnullexception)	Thrown if `builder` is null.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comm
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

UseSpectreConsoleSyncCommandExecutor(IHostBuilder, Action<IConfigurator>, bool)

Sets up a command execution environment within the application's host configuration, where commands are executed synchronously. This method creates an entry point for a command-line application that supports multiple commands, each running in a synchronous manner within its own task context. In the event of a service cancellation, the currently executing command will be terminated abruptly.

Declaration

```
public static IHostBuilder UseSpectreConsoleSyncCommandExecutor(this IHostBuilder builder,
    Action<IConfigurator> configureCommandApp, bool suppressLifeTimeStatusMessages = true)
```

Parameters

Type	Name	Description
IHostBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostbuilder)	*builder*	The host builder to configure, which is augmented with services and configurations necessary for managing command execution.
Action <IConfigurator> (https://learn.microsoft.com/dotnet/api/system.action-1)	*configureCommandApp*	A delegate that configures the command-line application, facilitating the registration and setup of individual commands.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*suppressLifeTimeStatusMessages*	Optional. Determines whether to suppress host lifetime status messages, with a default value of true. When set to false, these messages are made visible, offering insights into the lifecycle events of the host.

Returns

Type	Description
IHostBuilder (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostbuilder)	The host builder, now configured with the synchronous command execution environment, ready for further configuration or immediate use.

Remarks

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

```
static async Task Main(string[] args)
{
    var builder = Host.CreateDefaultBuilder(args);
    builder.ConfigureServices(services =>
    {
        services.AddSingleton<ISomeServiceDependency, SomeServiceDependency>();
        services.AddHostedService<SomeOtherBackgroundService>();
    });
    builder.UseSpectreConsoleSyncCommandExecutor(configCommandApp =>
    {
        configCommandApp.SetApplicationName("testapp");
        configCommandApp.AddCommand<MyAsyncCommand>("my-command");
    });
    await builder.Build().RunAsync();
}
public class MyAsyncCommand : AsyncCommand<MyAsyncCommand.Settings>
{
    private readonly ISomeServiceDependency dependencyService;
    private readonly IHostApplicationLifetime appLifetime;
    public MyAsyncCommand(ISomeServiceDependency dependencyService, IHostApplicationLifetime a
    ppLifetime)
    {
        this.dependencyService = dependencyService;
        this.appLifetime = appLifetime;
    }
    public class Settings : CommandSettings
    {
        [Description("The commandline setting")]
        [CommandArgument(0, "<Setting>")]
        public string? SomeSetting { get; init; }
    }
    public override async Task<int> ExecuteAsync(CommandContext context, Settings settings)
    {
        // Attempts to abort the command gracefully will fail because the command runs on a se
    parate task and not in the service host's managed context.
        try
        {
            await dependencyService.PerformConsoleAction(appLifetime.ApplicationStopping);
        }
        catch (Exception)
        {
            // This catch block is unreachable. The service host's abrupt termination of the c
    ommand prevents this block from being executed.
            return (int)SpectreConsoleHostedService.ExitCode.CommandTerminated;
        }
        // It's crucial to explicitly manage the service host's state after command execution
        completes, either by continuing operation or initiating shutdown.
        return (int)SpectreConsoleHostedService.ExitCode.SuccessAndContinue;
    }
```

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.StringBuilder (Coree.NETStandard.Extensions.StringBuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectr
  - SpectreConsoleHostedServiceExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

```
}
```

Exceptions

Type	Condition
ArgumentNullException (https://learn.microsoft.com/dotnet/api/system.argumentnullexception)	Thrown if the provided host builder is null, ensuring method reliability and preventing null reference errors.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Class ServiceCollectionExtensions

Provides extension methods for IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) to facilitate the integration of Spectre.Console's CommandApp.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ ServiceCollectionExtensions

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public static class ServiceCollectionExtensions
```

Methods

AddSpectreConsole(IServiceCollection, Action<IConfigurator>, bool)

Adds and configures Spectre.Console's Spectre.Console.Cli.CommandApp and related services to the specified IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection). This setup allows for utilizing Spectre.Console's command line interface tools within a .NET Core application, enabling the definition, parsing, and execution of commands.

Declaration

```
public static IServiceCollection AddSpectreConsole(this IServiceCollection services, Actio
    n<IConfigurator> configureCommandApp, bool suppressLifeTimeStatusMessages = true)
```

Parameters

Type	Name	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	*services*	The IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) to add services to.

Enter here to filter...
+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Type	Name	Description
Action (https://learn.microsoft.com/dotnet/api/system.action-1) <IConfigurator>	*configureCommandApp*	An Action (https://learn.microsoft.com/dotnet/api/system.action) delegate that is used to configure the Spectre.Console.Cli.CommandApp.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*suppressLifeTimeStatusMessages*	Supresses default output of the application host.

Returns

Type	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	The IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) so that additional calls can be chained.

Remarks

This method performs the following operations:

- Initializes a new instance of Spectre.Console.Cli.CommandApp and configures it using the provided `configureCommandApp` action.
- Scans for command types registered within the Spectre.Console.Cli.CommandApp and registers them as singletons in the service collection.
- Registers the Spectre.Console.Cli.CommandApp itself as a singleton service, configured to use a Spectre ConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.SpectreConsoleTypeRegistrar.html) for dependency injection.
- Adds a SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.SpectreConsoleHostedService.html), to manage the application lifecycle of the Spectre.Console.Cli.CommandApp.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

```
static async Task Main(string[] args)
{
    var builder = Host.CreateDefaultBuilder(args);
    builder.ConfigureServices(services =>
    {
        services.AddSpectreConsole(configCommandApp =>
        {
            configCommandApp.SetApplicationName("testapp");
            configCommandApp.AddCommand<MyAsyncCommand>("my-command");
        });
        services.AddSingleton<ISomeServiceDependency, SomeServiceDependency>();
        services.AddHostedService<SomeOtherBackgroundService>();
    });
    await builder.Build().RunAsync();
}

public class MyAsyncCommand : AsyncCommand<MyAsyncCommand.Settings>
{
    private readonly ISomeServiceDependency dependencyService; private readonly IHostAppli
cationLifetime appLifetime;

    public MyAsyncCommand(ISomeServiceDependency dependencyService, IHostApplicationLifeti
me appLifetime)
    {
        this.dependencyService = dependencyService; this.appLifetime = appLifetime;
    }

    public class Settings : CommandSettings
    {
        [Description("The commandline setting")]
        [CommandArgument(0, "<Setting>")]
        public string? SomeSetting { get; init; }
    }

    public override async Task<int> ExecuteAsync(CommandContext context, Settings setting
s)
    {
        try { var x = dependencyService.PerformConsoleAction(appLifetime.ApplicationStoppin
g); }
        catch (Exception) { return (int)SpectreConsoleHostedService.ExitCode.CommandTermin
ated; }
        return (int)SpectreConsoleHostedService.ExitCode.SuccessAndContinue;
    }
}
```

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comm
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

AddSpectreConsoleSyncCommandExecutor(IServiceCollection, Action<IConfigurator>, bool)

Registers a command execution environment with Spectre.Console integration into the application's service collection. This setup facilitates the execution of command-line commands synchronously within their own task contexts. Designed to support multiple commands, this configuration allows for abrupt termination of commands upon service cancellation, ensuring immediate response to stop requests.

Declaration

```
public static IServiceCollection AddSpectreConsoleSyncCommandExecutor(this IServiceCollect
ion services, Action<IConfigurator> configureCommandApp, bool suppressLifeTimeStatusMessag
es = true)
```

Parameters

Type	Name	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	*services*	The IServiceCollection to configure, which is augmented with the necessary services and configurations for Spectre.Console command execution.
Action (https://learn.microsoft.com/dotnet/api/system.action-1) <IConfigurator>	*configureCommandApp*	A delegate that configures the command-line application, facilitating the registration and setup of individual commands within the Spectre.Console framework.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*suppressLifeTimeStatusMessages*	Optional. Determines whether to suppress host lifetime status messages, with a default value of true. Setting this to false enables the visibility of these messages, providing insights into the lifecycle events of the host and command execution.

Returns

Type	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	The IServiceCollection, now configured with the Spectre.Console command execution environment, ready for further configuration or immediate use within the application's startup process.

Remarks

Integrating the SpectreConsole command executor into your services collection. Commands are executed synchronously within a separate task. Note that attempts to abort commands gracefully will face challenges due to their execution in separate tasks, which are not managed by the service host's context.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.Hosted
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - SpectreCommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - SpectreHostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - SpectreServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

```
static async Task Main(string[] args)
{
    var builder = Host.CreateDefaultBuilder(args);
    builder.ConfigureServices(services =>
    {
        services.AddSingleton<ISomeServiceDependency, SomeServiceDependency>();
        services.AddHostedService<SomeOtherBackgroundService>();
        services.AddSpectreConsoleSyncCommandExecutor(configureCommandApp =>
        {
            configureCommandApp.SetApplicationName("toolkit");
            configureCommandApp.AddCommand<MyAsyncCommand>("my-command");
        });
    });
    await builder.Build().RunAsync();
}

public class MyAsyncCommand : AsyncCommand<MyAsyncCommand.Settings>
{
    private readonly ISomeServiceDependency dependencyService;
    private readonly IHostApplicationLifetime appLifetime;

    public MyAsyncCommand(ISomeServiceDependency dependencyService, IHostApplicationLifeti
me appLifetime)
    {
        this.dependencyService = dependencyService;
        this.appLifetime = appLifetime;
    }

    public class Settings : CommandSettings
    {
        [Description("The commandline setting")]
        [CommandArgument(0, "<Setting>")]
        public string? SomeSetting { get; init; }
    }

    public override async Task<int> ExecuteAsync(CommandContext context, Settings setting
s)
    {
        // Attempts to abort the command gracefully will fail because the command runs on
a separate task and not in the service host's managed context.
        try {
            await dependencyService.PerformConsoleAction(appLifetime.ApplicationStopping);
        }
        catch (Exception)
        {
            // This catch block is unreachable. The service host's abrupt termination of t
he command prevents this block from being executed.
            return (int)SpectreConsoleHostedService.ExitCode.CommandTerminated;
        }
        // It's crucial to explicitly manage the service host's state after command execut
ion completes, either continuing operation or initiating shutdown.
        return (int)SpectreConsoleHostedService.ExitCode.SuccessAndContinue;
```

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectr
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

```
        }
    }
```


+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
    - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
    - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
    - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
    - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
    - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
    - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Class SpectreConsoleHostedService

Provides a hosted service for running Spectre.Console CLI applications within a .NET Core generic host. This class handles the application lifecycle, processes commands, and manages application exit codes based on command execution results.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)

↳ SpectreConsoleHostedService

Implements

IHostedService (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostedservice)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class SpectreConsoleHostedService : IHostedService
```

Constructors

SpectreConsoleHostedService(ICommandApp, IHostApplicationLifetime)

Initializes a new instance of the SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.SpectreConsoleHostedService.html) class with the specified Spectre.Console CLI application and application lifetime management services.

Declaration

```
public SpectreConsoleHostedService(ICommandApp commandApp, IHostApplicationLifetime appLif
etime)
```

Parameters

Type	Name	Description
ICommandApp	*commandApp*	The Spectre.Console CLI application to be hosted.
IHostApplicationLifetime (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostapplicationlifetime)	*appLifetime*	The application lifetime management service.

Enter here to filter...

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

+ Coree.NETStandard.Classes
(Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient
(Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions
(Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool
(Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable
(Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder
(Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings
(Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection
(Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog
(Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services
(Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights
(Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole
(Coree.NETStandard.SpectreConsole.html)
CommandAppExtensions
(Coree.NETStandard.SpectreConsole.Comma
HostBuilderExtensions
(Coree.NETStandard.SpectreConsole.HostBu
ServiceCollectionExtensions
(Coree.NETStandard.SpectreConsole.Service
SpectreConsoleHostedService
(Coree.NETStandard.SpectreConsole.Spectre
SpectreConsoleHostedService.ExitCode
(Coree.NETStandard.SpectreConsole.Spectre
SpectreConsoleTypeRegistrar
(Coree.NETStandard.SpectreConsole.Spectre

Methods

StartAsync(CancellationToken)

Starts the hosted Spectre.Console CLI application asynchronously. Parses the command line arguments, executes the corresponding command, and sets the application exit code based on the outcome of the command execution. This method employs a specialized set of exit codes defined by the SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.SpectreConsoleHostedService.ExitCode.html) enum, which may deviate from standard exit code conventions to suit the specific behavior and requirements of this application. For example, it includes codes for successful command execution that leads to application continuation or termination, and for handling cases where help information is displayed or a command is not found.

Declaration

```
public Task StartAsync(CancellationToken cancellationToken)
```

Parameters

Type	Name	Description
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	A token that can be used to request cancellation of the operation.

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task)	A task that represents the asynchronous start operation.

StopAsync(CancellationToken)

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. Currently, this method completes immediately as no cleanup is needed.

Declaration

```
public Task StopAsync(CancellationToken cancellationToken)
```

Parameters

Type	Name	Description
CancellationToken (https://learn.microsoft.com/dotnet/api/system.threading.cancellationtoken)	*cancellationToken*	A token that can be used to request cancellation of the operation.

Returns

Type	Description
Task (https://learn.microsoft.com/dotnet/api/system.threading.tasks.task)	A task that represents the asynchronous stop operation.

Implements

IHostedService (https://learn.microsoft.com/dotnet/api/microsoft.extensions.hosting.ihostedservice)

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Enum SpectreConsoleHostedService.ExitCode

Enter here to filter...

Defines exit codes for the SpectreConsoleHostedService application. These exit codes communicate the outcome of the application's execution to the calling process, allowing for scripted interactions and error handling.

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public enum SpectreConsoleHostedService.ExitCode
```

Fields

Name	Description
CommandFailedToRun	Indicates that the application command failed to run. This can occur if initialization fails or if the application is aborted before it can run properly.
CommandNotFound	Indicates that the specified command could not be found. This exit code is used when a user inputs a command that does not match any known command patterns.
CommandTerminated	Indicates that the command was terminated, often due to an external interrupt signal (e.g., Ctrl+C). This code is reserved for scenarios where the application's execution is abruptly stopped.
HelpVersionDisplayed	Indicates that help or version information was displayed. This neutral, Linux-like exit code is used when the application successfully displays help or version information upon request.
SuccessAndCommandIsDone	Indicates that the command executed successfully and the application will not initiate a shutdown. This exit code signifies that the command was completed successfully, and the application remains active for further commands or operations. The application lifecycle is not affected, and `appLifetime.Stop` is not called. Note: If background services or other tasks running within the application encounter a failure, they need to set a failed exit code manually (e.g., `Environment.ExitCode = non-zero value`) before calling `appLifetime.Stop` to ensure the exit code accurately reflects the application state.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comm
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Name	Description
SuccessAndExit	Indicates that the command executed successfully and the application will initiate a shutdown. This exit code is used when a command completes successfully and dictates that the application should perform a clean shutdown by calling `appLifetime.Stop`. It signifies the end of the application's current lifecycle, allowing for graceful termination.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Class SpectreConsoleTypeRegistrar

Enables type registration for Spectre.Console applications, supporting both IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider) and IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection).

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ SpectreConsoleTypeRegistrar

Implements

ITypeRegistrar

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public class SpectreConsoleTypeRegistrar : ITypeRegistrar
```

Constructors

SpectreConsoleTypeRegistrar(IServiceCollection, bool)

Constructs with an IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection), for manual service collection setups, e.g., simple console apps.

Declaration

```
public SpectreConsoleTypeRegistrar(IServiceCollection serviceCollection, bool disposeServi
ceProvider)
```

Parameters

Type	Name	Description
IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection)	*serviceCollection*	Service collection for registrations.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*disposeServiceProvider*	True to dispose the service provider on CommandApp disposal.

SpectreConsoleTypeRegistrar(IServiceProvider, bool)

Constructs with an IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider), typically for hosted applications using `builder.Build()`.

True to dispose the service provider on CommandApp disposal.

Declaration

```
public SpectreConsoleTypeRegistrar(IServiceProvider serviceProvider, bool disposeServicePr
ovider)
```

Parameters

Type	Name	Description
IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider)	*serviceProvider*	Service provider for type resolution.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*disposeServiceProvider*	

Methods

Build()

Builds a type resolver based on the provided service provider or service collection.

Declaration

```
public ITypeResolver Build()
```

Returns

Type	Description
ITypeResolver	Type resolver for configured services.

Exceptions

Type	Condition
InvalidOperationException (https://learn.microsoft.com/dotnet/api/system.invalidoperationexception)	Thrown if no service provider or collection is provided.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Register(Type, Type)

Registers a service with its implementation in the service collection. Only for IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) usage.

Declaration

```
public void Register(Type service, Type implementation)
```

Parameters

Type	Name	Description
Type (https://learn.microsoft.com/dotnet/api/system.type)	*service*	Service type.
Type (https://learn.microsoft.com/dotnet/api/system.type)	*implementation*	Implementation type.

RegisterInstance(Type, object)

Registers a service instance in the service collection. Only for IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) usage.

Declaration

```
public void RegisterInstance(Type service, object implementation)
```

Parameters

Type	Name	Description
Type (https://learn.microsoft.com/dotnet/api/system.type)	*service*	Service type.
object (https://learn.microsoft.com/dotnet/api/system.object)	*implementation*	Service instance.

RegisterLazy(Type, Func<object>)

Registers a service with a factory function for deferred instantiation. Only for IServiceCollection (https://learn.microsoft.com/dotnet/api/microsoft.extensions.dependencyinjection.iservicecollection) usage.

Declaration

```
public void RegisterLazy(Type service, Func<object> func)
```

Parameters

Type	Name	Description
Type (https://learn.microsoft.com/dotnet/api/system.type)	*service*	Service type.

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Type	Name	Description
Func (https://learn.microsoft.com/dotnet/api/system.func-1)< object (https://learn.microsoft.com/dotnet/api/system.object)>	*func*	Factory function.

Exceptions

Type	Condition
ArgumentNullException (https://learn.microsoft.com/dotnet/api/system.argumentnullexception)	Thrown if `func` is null.

Implements

Spectre.Console.Cli.ITypeRegistrar

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Class SpectreConsoleTypeResolver

Resolves types from an IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider), and optionally disposes it.

Inheritance

↳ object (https://learn.microsoft.com/dotnet/api/system.object)
↳ SpectreConsoleTypeResolver

Implements

ITypeResolver
IDisposable (https://learn.microsoft.com/dotnet/api/system.idisposable)

Namespace: Coree (Coree.html).NETStandard (Coree.NETStandard.html).SpectreConsole (Coree.NETStandard.SpectreConsole.html)

Assembly: Coree.NETStandard.dll

Syntax

```
public sealed class SpectreConsoleTypeResolver : ITypeResolver, IDisposable
```

Constructors

SpectreConsoleTypeResolver(IServiceProvider, bool)

Initializes with a specified service provider.

Declaration

```
public SpectreConsoleTypeResolver(IServiceProvider provider, bool DisposeServiceProvider)
```

Parameters

Type	Name	Description
IServiceProvider (https://learn.microsoft.com/dotnet/api/system.iserviceprovider)	*provider*	Service provider for type resolution.
bool (https://learn.microsoft.com/dotnet/api/system.boolean)	*DisposeServiceProvider*	True to dispose the service provider on disposal.

Methods

Enter here to filter...
+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comm
HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre

Dispose()

Disposes the service provider if disposable.

Declaration

```
public void Dispose()
```

Resolve(Type?)

Resolves a service of the specified type.

Declaration

```
public object? Resolve(Type? type)
```

Parameters

Type	Name	Description
Type (https://learn.microsoft.com/dotnet/api/system.type)	*type*	Service type; returns null if type is null.

Returns

Type	Description
object (https://learn.microsoft.com/dotnet/api/system.object)	Service object or null if not found.

Implements

Spectre.Console.Cli.ITypeResolver

IDisposable (https://learn.microsoft.com/dotnet/api/system.idisposable)

+ Coree.NETStandard.Abstractions (Coree.NETStandard.Abstractions.html)
+ Coree.NETStandard.Classes (Coree.NETStandard.Classes.html)
+ Coree.NETStandard.CoreeHttpClient (Coree.NETStandard.CoreeHttpClient.html)
+ Coree.NETStandard.Extensions (Coree.NETStandard.Extensions.html)
+ Coree.NETStandard.Extensions.Bool (Coree.NETStandard.Extensions.Bool.html)
+ Coree.NETStandard.Extensions.Enumerable (Coree.NETStandard.Extensions.Enumerable.ht
+ Coree.NETStandard.Extensions.Stringbuilder (Coree.NETStandard.Extensions.Stringbuilder.h
+ Coree.NETStandard.Extensions.Strings (Coree.NETStandard.Extensions.Strings.html)
+ Coree.NETStandard.HostedServicesCollection (Coree.NETStandard.HostedServicesCollection
+ Coree.NETStandard.Serilog (Coree.NETStandard.Serilog.html)
+ Coree.NETStandard.Services (Coree.NETStandard.Services.html)
+ Coree.NETStandard.Services.RuntimeInsights (Coree.NETStandard.Services.RuntimeInsights.
- Coree.NETStandard.SpectreConsole (Coree.NETStandard.SpectreConsole.html)
  - CommandAppExtensions (Coree.NETStandard.SpectreConsole.Comma
  - HostBuilderExtensions (Coree.NETStandard.SpectreConsole.HostBu
  - ServiceCollectionExtensions (Coree.NETStandard.SpectreConsole.Service
  - SpectreConsoleHostedService (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleHostedService.ExitCode (Coree.NETStandard.SpectreConsole.Spectre
  - SpectreConsoleTypeRegistrar (Coree.NETStandard.SpectreConsole.Spectre