

Namespace Rudeus

Classes

[Utils](#)

ユーティリティクラス

Interfaces

[Utils](#)

ユーティリティクラスのインターフェース

Interface IUtils

Namespace: [Rudeus](#)

Assembly: RudeusBg.dll

ユーティリティクラスのインターフェース

```
public interface IUtils
```

Extension Methods

[AssertEx.IsInstanceOf<TExpected>\(object\)](#) , [AssertEx.IsInstanceOf<TExpected>\(object\)](#) ,
[AssertEx.IsInstanceOf<TExpected>\(object\)](#) , [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) ,
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) , [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) ,
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#) , [AssertEx.IsNotStructuralEqual\(object, object, string\)](#) ,
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#) , [AssertEx.IsStructuralEqual\(object, object, string\)](#) ,
[AssertEx.IsStructuralEqual\(object, object, string\)](#) , [AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

CompareVersionString(string, string, int)

ドット区切り数字列を比較する

```
public static abstract int CompareVersionString(string a, string b, int depth)
```

Parameters

a [string](#)[🔗]

対象文字列A

b [string](#)[🔗]

対象文字列B

depth [int](#)[🔗]

比較する列数 (0なら全て)

Returns

[int](#)

A-Bの符号 (1, 0, -1)

ConcatStudentNumberFromMail(string)

メールアドレスから学籍番号を抽出する

```
public static abstract string ConcatStudentNumberFromMail(string mailAddress)
```

Parameters

mailAddress [string](#)

sXXXXXXX@s.do-johodai.ac.jpのメールアドレス

Returns

[string](#)

XXXXXXXの数字部分

CopyDirectory(string, string, bool)

ディレクトリを再帰的にコピーする

```
public static abstract void CopyDirectory(string sourceDir, string destDir, bool recursive)
```

Parameters

sourceDir [string](#)

destDir [string](#)

recursive [bool](#)

IsStudentMailAddress(string)

メールアドレスが学生用のものかどうかを判定する

```
public static abstract bool IsStudentMailAddress(string mailAddress)
```

Parameters

mailAddress [string](#)

メールアドレス

Returns

[bool](#)

ParseArgs(string[])

key=value形式の引数をパースする

```
public static abstract Dictionary<string, string> ParseArgs(string[] args)
```

Parameters

args [string](#)[]

Returns

[Dictionary](#) <[string](#), [string](#)>

RegisterDeviceAndSetData()

廃止予定 デバイスを登録し、データを設定する

```
public static abstract void RegisterDeviceAndSetData()
```

Class Utils

Namespace: [Rudeus](#)

Assembly: RudeusBg.dll

ユーティリティクラス

```
public class Utils : IUtils
```

Inheritance

[object](#) ← [Utils](#)

Implements

[IUtils](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Fields

KyoumuUrl

```
public static readonly string KyoumuUrl
```

Field Value

[string](#)

Polite3Url

```
public static readonly string Polite3Url
```

Field Value

[string](#)

WebPortalUrl

```
public static readonly string WebPortalUrl
```

Field Value

[string](#)

Properties

LM

```
public static ILocalMachine LM { get; set; }
```

Property Value

Methods

CompareVersionString(string, string, int)

ドット区切り数字列を比較する

```
public static int CompareVersionString(string a, string b, int depth = 0)
```

Parameters

a [string](#)

対象文字列A

b [string](#)

対象文字列B

depth [int](#)

比較する列数 (0なら全て)

Returns

[int](#)

A-Bの符号 (1, 0, -1)

ConcatStudentNumberFromMail(string)

s9999999@s.do-johodai.ac.jp -> 9999999

```
public static string ConcatStudentNumberFromMail(string mailAddress)
```

Parameters

mailAddress [string](#)

Returns

[string](#)

Exceptions

[Exception](#)

CopyDirectory(string, string, bool)

ディレクトリを再帰的にコピーする

```
public static void CopyDirectory(string sourceDir, string destinationDir, bool recursive)
```

Parameters

sourceDir [string](#)

destinationDir [string](#)

recursive [bool](#)

IsStudentMailAddress(string)

s9999999@s.do-johodai.ac.jp -> true

```
public static bool IsStudentMailAddress(string mailAddress)
```

Parameters

mailAddress [string](#)

Returns

[bool](#)

ParseArgs(string[])

key=value形式の引数をパースする

```
public static Dictionary<string, string> ParseArgs(string[] args)
```

Parameters

args [string](#) []

Returns

[Dictionary](#) <[string](#), [string](#)>

RegisterDeviceAndSetData()

廃止予定 デバイスを登録し、データを設定する

```
public static void RegisterDeviceAndSetData()
```

Namespace Rudeus.API

Classes

[RequestClient](#)

Interfaces

[IRequestClient](#)

Interface IRequestClient

Namespace: [Rudeus.API](#)

Assembly: RudeusBg.dll

```
public interface IRequestClient
```

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#).

Methods

Request(HttpRequestMessage)

HttpResponseMessage **Request**(HttpRequestMessage message)

Parameters

message [HttpRequestMessage](#)

Returns

[HttpResponseMessage](#) 

RequestString(HttpRequestMessage)

```
string RequestString(HttpRequestMessage message)
```

Parameters

message [HttpRequestMessage](#) 

Returns

[string](#) 

Class RequestClient

Namespace: [Rudeus.API](#)

Assembly: RudeusBg.dll

```
public class RequestClient : IRequestClient
```

Inheritance

[object](#) ← RequestClient

Implements

[IRequestClient](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Constructors

RequestClient(string)

```
public RequestClient(string endpoint)
```

Parameters

endpoint [string](#)↗

Methods

Request(HttpRequestMessage)

```
public HttpResponseMessage Request(HttpRequestMessage message)
```

Parameters

message [HttpRequestMessage](#)↗

Returns

[HttpResponseMessage](#)↗

RequestString(HttpRequestMessage)

```
public string RequestString(HttpRequestMessage message)
```

Parameters

message [HttpRequestMessage](#)↗

Returns

[string](#)↗

Namespace Rudeus.Model

Classes

[FakeSettings](#)

レジストリの設定を保持するモデルのFake実装

[LocalMachine](#)

ローカルマシンの情報を取得するモデル

[Settings](#)

レジストリに設定を保持するモデル

Interfaces

[ILocalCertificate](#)

ローカルマシンの証明書を操作するモデルのインターフェース

[ILocalMachine](#)

ローカルマシンの情報を取得するモデルのインターフェース

[ISettings](#)

設定をレジストリに保管するモデル

Class FakeSettings

Namespace: [Rudeus.Model](#)

Assembly: RudeusBg.dll

レジストリの設定を保持するモデルのFake実装

```
public class FakeSettings : ISettings
```

Inheritance

[object](#) ← FakeSettings

Implements

[ISettings](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Properties

AccessTokenP

```
public string AccessTokenP { get; set; }
```

Property Value

[string](#)

CurrentVersionP

```
public string CurrentVersionP { get; set; }
```

Property Value

[string](#)

DeviceIdP

```
public string DeviceIdP { get; set; }
```

Property Value

[string](#)

FirstHostnameP

```
public string FirstHostnameP { get; set; }
```

Property Value

[string](#)

HostnameP

```
public string HostnameP { get; set; }
```

Property Value

[string](#) 

LabelIdP

```
public string LabelIdP { get; set; }
```

Property Value

[string](#) 

LastDirNameP

```
public string LastDirNameP { get; set; }
```

Property Value

[string](#) 

LastVersionDirPathP

```
public string LastVersionDirPathP { get; set; }
```

Property Value

[string](#) 

LastVersionExeNameP

```
public string LastVersionExeNameP { get; set; }
```

Property Value

[string](#) 

LastVersionExePathP

```
public string LastVersionExePathP { get; set; }
```

Property Value

[string](#) 

LatestDirNameP

```
public string LatestDirNameP { get; set; }
```

Property Value

[string](#) 

LatestVersionDirPathP

```
public string LatestVersionDirPathP { get; set; }
```

Property Value

[string](#) 

LatestVersionExeNameP

```
public string LatestVersionExeNameP { get; set; }
```

Property Value

[string](#) 

LatestVersionExePathP

```
public string LatestVersionExePathP { get; set; }
```

Property Value

[string](#) 

LatestVersionStatusP

```
public string LatestVersionStatusP { get; set; }
```

Property Value

[string](#) 

UpdatingChannelP

```
public string UpdatingChannelP { get; set; }
```

Property Value

[string](#) 

UsernameP

```
public string UsernameP { get; set; }
```

Property Value

[string](#)

Methods

Create()

```
public static ISettings Create()
```

Returns

[ISettings](#)

GetInstance()

```
public static ISettings GetInstance()
```

Returns

[ISettings](#)

IsBetaChannelP()

```
public bool IsBetaChannelP()
```

Returns

[bool](#)

IsDevelopChannelP()

```
public bool IsDevelopChannelP()
```

Returns

[bool](#)

IsLatestVersionStatusDownloadedP()

```
public bool IsLatestVersionStatusDownloadedP()
```

Returns

[bool](#)

IsLatestVersionStatusOkP()

```
public bool IsLatestVersionStatusOkP()
```

Returns

[bool](#)

IsLatestVersionStatusUnlaunchableP()

```
public bool IsLatestVersionStatusUnlaunchableP()
```

Returns

[bool](#)

IsStableChannelP()

```
public bool IsStableChannelP()
```

Returns

[bool](#)

IsTestChannelP()

```
public bool IsTestChannelP()
```

Returns

[bool](#)

SetBetaChannelP()

```
public void SetBetaChannelP()
```

SetDevelopChannelP()

```
public void SetDevelopChannelP()
```

SetLatestVersionStatusDownloadedP()

```
public void SetLatestVersionStatusDownloadedP()
```

SetLatestVersionStatusOkP()

```
public void SetLatestVersionStatusOkP()
```

SetLatestVersionStatusUnlaunchableP()

```
public void SetLatestVersionStatusUnlaunchableP()
```

SetStableChannelP()

```
public void SetStableChannelP()
```

SetTestChannelP()

```
public void SetTestChannelP()
```


Interface ILocalCertificate

Namespace: [Rudeus.Model](#)

Assembly: RudeusBg.dll

ローカルマシンの証明書を操作するモデルのインターフェース

```
public interface ILocalCertificate
```

Extension Methods

[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

GetCertificate(string)

```
X509Certificate2? GetCertificate(string issuer)
```

Parameters

issuer [string](#) 

Returns

[X509Certificate2](#) 

GetInstance()

```
public static abstract ILocalCertificate GetInstance()
```

Returns

[ILocalCertificate](#)

InstallCertificateIntoRoot(string)

```
void InstallCertificateIntoRoot(string path)
```

Parameters

path [string](#)

InstallPkcs12IntoMy(string, string)

```
void InstallPkcs12IntoMy(string path, string password)
```

Parameters

path [string](#)

password [string](#)

UninstallCertificateFromString(string)

```
void UninstallCertificateFromString(string certificate)
```

Parameters

certificate [string](#)

Interface ILocalMachine

Namespace: [Rudeus.Model](#)

Assembly: RudeusBg.dll

ローカルマシンの情報を取得するモデルのインターフェース

```
public interface ILocalMachine
```

Extension Methods

[AssertEx.IsInstanceOf<TExpected>\(object\)](#) , [AssertEx.IsInstanceOf<TExpected>\(object\)](#) ,
[AssertEx.IsInstanceOf<TExpected>\(object\)](#) , [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) ,
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) , [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) ,
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#) , [AssertEx.IsNotStructuralEqual\(object, object, string\)](#) ,
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#) , [AssertEx.IsStructuralEqual\(object, object, string\)](#) ,
[AssertEx.IsStructuralEqual\(object, object, string\)](#) , [AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

GetDeviceId()

```
string GetDeviceId()
```

Returns

[string](#) 

GetHostname()

```
string GetHostname()
```

Returns

[string](#) 

GetInstance()

```
public static abstract ILocalMachine GetInstance()
```

Returns

[ILocalMachine](#)

GetPhysicalRamInfo()

```
string GetPhysicalRamInfo()
```

Returns

[string](#) 

GetSpec()

```
string GetSpec()
```

Returns

[string](#) 

GetStorageDeviceIdList()

```
List<string> GetStorageDeviceIdList()
```

Returns

[List](#)  <[string](#)  >

GetWinVersion()

```
string GetWinVersion()
```

Returns

[string](#) 

Interface ISettings

Namespace: [Rudeus.Model](#)

Assembly: RudeusBg.dll

設定をレジストリに保管するモデル

```
public interface ISettings
```

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#).

Properties

AccessTokenP

```
string AccessTokenP { get; set; }
```

Property Value

[string](#)

CurrentVersionP

```
string CurrentVersionP { get; set; }
```

Property Value

[string](#)

DeviceIdP

```
string DeviceIdP { get; set; }
```

Property Value

[string](#)

FirstHostnameP

```
string FirstHostnameP { get; set; }
```

Property Value

[string](#)

HostnameP

```
string HostnameP { get; set; }
```

Property Value

[string](#)

LabelIdP

```
string LabelIdP { get; set; }
```

Property Value

[string](#)

LastDirNameP

```
string LastDirNameP { get; set; }
```

Property Value

[string](#)

LastVersionDirPathP

```
string LastVersionDirPathP { get; set; }
```

Property Value

[string](#)

LastVersionExeNameP

```
string LastVersionExeNameP { get; set; }
```

Property Value

[string](#)

LastVersionExePathP


```
string LastVersionExePathP { get; }
```

Property Value

[string](#) 

LatestDirNameP

```
string LatestDirNameP { get; set; }
```

Property Value

[string](#) 

LatestVersionDirPathP

```
string LatestVersionDirPathP { get; set; }
```

Property Value

[string](#) 

LatestVersionExeNameP

```
string LatestVersionExeNameP { get; set; }
```

Property Value

[string](#) 

LatestVersionExePathP

```
string LatestVersionExePathP { get; }
```

Property Value

[string](#) 

LatestVersionStatusP

```
string LatestVersionStatusP { get; set; }
```

Property Value

[string](#) 

UpdatingChannelP

```
string UpdatingChannelP { get; set; }
```

Property Value

[string](#) 

UsernameP

```
string UsernameP { get; set; }
```

Property Value

[string](#) 

Methods

IsBetaChannelP()

```
bool IsBetaChannelP()
```

Returns

[bool](#)

IsDevelopChannelP()

```
bool IsDevelopChannelP()
```

Returns

[bool](#)

IsLatestVersionStatusDownloadedP()

```
bool IsLatestVersionStatusDownloadedP()
```

Returns

[bool](#)

IsLatestVersionStatusOkP()

```
bool IsLatestVersionStatusOkP()
```

Returns

[bool](#)

IsLatestVersionStatusUnlaunchableP()

```
bool IsLatestVersionStatusUnlaunchableP()
```

Returns

[bool](#)

IsStableChannelP()

```
bool IsStableChannelP()
```

Returns

[bool](#)

IsTestChannelP()

```
bool IsTestChannelP()
```

Returns

[bool](#)

SetBetaChannelP()

```
void SetBetaChannelP()
```

SetDevelopChannelP()

```
void SetDevelopChannelP()
```

SetLatestVersionStatusDownloadedP()

```
void SetLatestVersionStatusDownloadedP()
```

SetLatestVersionStatusOkP()

```
void SetLatestVersionStatusOkP()
```

SetLatestVersionStatusUnlaunchableP()

```
void SetLatestVersionStatusUnlaunchableP()
```

SetStableChannelP()

```
void SetStableChannelP()
```

SetTestChannelP()

```
void SetTestChannelP()
```

Class LocalMachine

Namespace: [Rudeus.Model](#)

Assembly: RudeusBg.dll

ローカルマシンの情報を取得するモデル

```
public class LocalMachine : ILocalMachine
```

Inheritance

[object](#) ← LocalMachine

Implements

[ILocalMachine](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

GetDeviceId()

```
public string GetDeviceId()
```

Returns

[string](#) 

GetHostname()

```
public string GetHostname()
```

Returns

[string](#) 

GetInstance()

```
public static ILocalMachine GetInstance()
```

Returns

[ILocalMachine](#)

GetPhysicalRamInfo()

```
public string GetPhysicalRamInfo()
```

Returns

[string](#) 

GetSpec()

```
public string GetSpec()
```

Returns

[string](#) 

GetStorageDeviceIdList()

```
public List<string> GetStorageDeviceIdList()
```

Returns

[List](#)  [<string](#)  [>](#)

GetWinVersion()

```
public string GetWinVersion()
```

Returns

[string](#) 

Class Settings

Namespace: [Rudeus.Model](#)

Assembly: RudeusBg.dll

レジストリに設定を保持するモデル

```
public class Settings : ISettings
```

Inheritance

[object](#) ← Settings

Implements

[ISettings](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Constructors

Settings()

```
public Settings()
```

Settings(string)

```
public Settings(string registryKey)
```

Parameters

registryKey [string](#) 

Fields

AccessTokenKey

```
public static string AccessTokenKey
```

Field Value

[string](#) 

CurrentVersionKey

```
public static string CurrentVersionKey
```

Field Value

[string](#) 

DeviceIdKey

```
public static string DeviceIdKey
```

Field Value

[string](#) 

FirstHostnameKey

```
public static string FirstHostnameKey
```

Field Value

[string](#) 

HostnameKey

```
public static string HostnameKey
```

Field Value

[string](#) 

LabelIdKey

```
public static string LabelIdKey
```

Field Value

[string](#) 

LatestVersionStatusKey

```
public static string LatestVersionStatusKey
```

Field Value

[string](#) 

NetworkIFListKey

```
public string NetworkIFListKey
```

Field Value

[string](#) 

UpdateChannelKey

```
public static string UpdateChannelKey
```

Field Value

[string](#) 

UsernameKey

```
public static string UsernameKey
```

Field Value

[string](#) 

Properties

AccessToken

```
public static string AccessToken { get; set; }
```

Property Value

[string](#) 

AccessTokenP

```
public string AccessTokenP { get; set; }
```

Property Value

[string](#) 

CurrentVersion

```
public static string CurrentVersion { get; set; }
```

Property Value

[string](#) 

CurrentVersionP

```
public string CurrentVersionP { get; set; }
```

Property Value

[string](#) 

DeviceId

```
public static string DeviceId { get; set; }
```

Property Value

[string](#) 

DeviceIdP

```
public string DeviceIdP { get; set; }
```

Property Value

[string](#) 

FirstHostname

```
public static string FirstHostname { get; set; }
```

Property Value

[string](#) 

FirstHostnameP

```
public string FirstHostnameP { get; set; }
```

Property Value

[string](#) 

Hostname

```
public static string Hostname { get; set; }
```

Property Value

[string](#) 

HostnameP

```
public string HostnameP { get; set; }
```

Property Value

[string](#) 

InitedLabelId

```
public static string InitedLabelId { get; }
```

Property Value

[string](#) 

InitedUsername

```
public static string InitedUsername { get; }
```

Property Value

[string](#) 

LabelId

```
public static string LabelId { get; set; }
```

Property Value

[string](#) 

LabelIdP

```
public string LabelIdP { get; set; }
```

Property Value

[string](#) 

LastDirName

```
public static string LastDirName { get; set; }
```

Property Value

[string](#) 

LastDirNameP

```
public string LastDirNameP { get; set; }
```

Property Value

[string](#) 

LastVersionDirPath


```
public static string LastVersionDirPath { get; set; }
```

Property Value

[string](#) 

LastVersionDirPathP

```
public string LastVersionDirPathP { get; set; }
```

Property Value

[string](#) 

LastVersionExeName

```
public static string LastVersionExeName { get; set; }
```

Property Value

[string](#) 

LastVersionExeNameP

```
public string LastVersionExeNameP { get; set; }
```

Property Value

[string](#) 

LastVersionExePath

```
public static string LastVersionExePath { get; }
```

Property Value

[string](#) 

LastVersionExePathP

```
public string LastVersionExePathP { get; }
```

Property Value

[string](#) 

LatestDirName

```
public static string LatestDirName { get; set; }
```

Property Value

[string](#) 

LatestDirNameP

```
public string LatestDirNameP { get; set; }
```

Property Value

[string](#) 

LatestVersionDirPath

```
public static string LatestVersionDirPath { get; set; }
```

Property Value

[string](#)

LatestVersionDirPathP

```
public string LatestVersionDirPathP { get; set; }
```

Property Value

[string](#)

LatestVersionExeName

```
public static string LatestVersionExeName { get; set; }
```

Property Value

[string](#)

LatestVersionExeNameP

```
public string LatestVersionExeNameP { get; set; }
```

Property Value

[string](#)

LatestVersionExePath

```
public static string LatestVersionExePath { get; }
```

Property Value

[string](#)

LatestVersionExePathP

```
public string LatestVersionExePathP { get; }
```

Property Value

[string](#)

LatestVersionStatus

```
public static string LatestVersionStatus { get; set; }
```

Property Value

[string](#)

LatestVersionStatusP

```
public string LatestVersionStatusP { get; set; }
```

Property Value

[string](#)

NetworkIFList

```
public string NetworkIFList { get; set; }
```

Property Value

[string](#) 

UpdatingChannel

```
public static string UpdatingChannel { get; set; }
```

Property Value

[string](#) 

UpdatingChannelP

```
public string UpdatingChannelP { get; set; }
```

Property Value

[string](#) 

Username

```
public static string Username { get; set; }
```

Property Value

[string](#) 

UsernameP

```
public string UsernameP { get; set; }
```

Property Value

[string](#)

Methods

IsBetaChannel()

```
public static bool IsBetaChannel()
```

Returns

[bool](#)

IsBetaChannelP()

```
public bool IsBetaChannelP()
```

Returns

[bool](#)

IsDevelopChannel()

```
public static bool IsDevelopChannel()
```

Returns

[bool](#)

IsDevelopChannelP()

```
public bool IsDevelopChannelP()
```

Returns

[bool](#)

IsLatestVersionStatusDownloaded()

```
public static bool IsLatestVersionStatusDownloaded()
```

Returns

[bool](#)

IsLatestVersionStatusDownloadedP()

```
public bool IsLatestVersionStatusDownloadedP()
```

Returns

[bool](#)

IsLatestVersionStatusOk()

```
public static bool IsLatestVersionStatusOk()
```

Returns

[bool](#)

IsLatestVersionStatusOkP()

```
public bool IsLatestVersionStatusOkP()
```

Returns

[bool](#)

IsLatestVersionStatusUnlaunchable()

```
public static bool IsLatestVersionStatusUnlaunchable()
```

Returns

[bool](#)

IsLatestVersionStatusUnlaunchableP()

```
public bool IsLatestVersionStatusUnlaunchableP()
```

Returns

[bool](#)

IsStableChannel()

```
public static bool IsStableChannel()
```

Returns

[bool](#)

IsStableChannelP()


```
public bool IsStableChannelP()
```

Returns

[bool](#)

IsTestChannel()

```
public static bool IsTestChannel()
```

Returns

[bool](#)

IsTestChannelP()

```
public bool IsTestChannelP()
```

Returns

[bool](#)

SetBetaChannel()

```
public static void SetBetaChannel()
```

SetBetaChannelP()

```
public void SetBetaChannelP()
```

SetDevelopChannel()

```
public static void SetDevelopChannel()
```

SetDevelopChannelP()

```
public void SetDevelopChannelP()
```

SetLatestVersionStatusDownloaded()

```
public static void SetLatestVersionStatusDownloaded()
```

SetLatestVersionStatusDownloadedP()

```
public void SetLatestVersionStatusDownloadedP()
```

SetLatestVersionStatusOk()

```
public static void SetLatestVersionStatusOk()
```

SetLatestVersionStatusOkP()

```
public void SetLatestVersionStatusOkP()
```

SetLatestVersionStatusUnlaunchable()

```
public static void SetLatestVersionStatusUnlaunchable()
```

SetLatestVersionStatusUnlaunchableP()

```
public void SetLatestVersionStatusUnlaunchableP()
```

SetStableChannel()

```
public static void SetStableChannel()
```

SetStableChannelP()

```
public void SetStableChannelP()
```

SetTestChannel()

```
public static void SetTestChannel()
```

SetTestChannelP()

```
public void SetTestChannelP()
```

UpdateRegistryKey(string?)

```
public static void UpdateRegistryKey(string? registryKey = null)
```

Parameters

registryKey [string](#)[🔗](#)

Namespace Rudeus.Procedure

Classes

[CertificateInstaller](#)

ローカルマシンに証明書をインストールする手続き

[FakeProcedure](#)

IProcedureのFake実装

Interfaces

[IFakeProcedure](#)

[ILauncher](#)

LauncherはExitCodeを返すべき手続きだったので作成

[IProcedure](#)

ひとまとまりの処理を実行するインターフェース 例えば、アクセストークンの再発行、インストール済みアプリの送信など

Class CertificateInstaller

Namespace: [Rudeus.Procedure](#)

Assembly: RudeusBg.dll

ローカルマシンに証明書をインストールする手続き

```
public class CertificateInstaller : IProcedure
```

Inheritance

[object](#) ← CertificateInstaller

Implements

[IProcedure](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Constructors

CertificateInstaller(ILocalCertificate?)

```
public CertificateInstaller(ILocalCertificate? lc = null)
```

Parameters

lc [ILocalCertificate](#)

Properties

_localCertificate

```
public ILocalCertificate _localCertificate { get; set; }
```

Property Value

[ILocalCertificate](#)

Methods

Run()

処理を実行する

```
public Task Run()
```

Returns

[Task](#)

Class FakeProcedure

Namespace: [Rudeus.Procedure](#)

Assembly: RudeusBg.dll

IProcedureのFake実装

```
public class FakeProcedure : IFakeProcedure, IProcedure
```

Inheritance

[object](#) ← FakeProcedure

Implements

[IFakeProcedure](#), [IProcedure](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Constructors

FakeProcedure()

```
public FakeProcedure()
```

Properties

RunCount

Runメソッドが呼ばれた回数

```
public int RunCount { get; }
```

Property Value

[int](#)

Methods

Run()

```
public Task Run()
```

Returns

[Task](#)

Interface IFakeProcedure

Namespace: [Rudeus.Procedure](#)

Assembly: RudeusBg.dll

```
public interface IFakeProcedure : IProcedure
```

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T,T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T,T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T,T\)](#), [AssertEx.IsNot<T>\(T,T\)](#), [AssertEx.IsNot<T>\(T,T\)](#),
[AssertEx.IsNot<T>\(T,T\)](#), [AssertEx.IsNot<T>\(T,T,IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T,T,IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T,T,IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T,T\)](#), [AssertEx.IsSameReferenceAs<T>\(T,T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T,T\)](#), [AssertEx.Is<T>\(T,T\)](#), [AssertEx.Is<T>\(T,T\)](#), [AssertEx.Is<T>\(T,T\)](#),
[AssertEx.Is<T>\(T,T,IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T,T,IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T,T,IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T,Expression<Func<T,bool>>,string\)](#),
[AssertEx.Is<T>\(T,Expression<Func<T,bool>>,string\)](#),
[AssertEx.Is<T>\(T,Expression<Func<T,bool>>,string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object,object,string\)](#),
[AssertEx.IsNotStructuralEqual\(object,object,string\)](#), [AssertEx.IsNotStructuralEqual\(object,object,string\)](#),
[AssertEx.IsStructuralEqual\(object,object,string\)](#), [AssertEx.IsStructuralEqual\(object,object,string\)](#),
[AssertEx.IsStructuralEqual\(object,object,string\)](#).

Properties

RunCount

```
int RunCount { get; }
```

Property Value


[int](#)

Methods

Run()

Task **Run()**

Returns

[Task](#)

Interface ILauncher

Namespace: [Rudeus.Procedure](#)

Assembly: RudeusLauncher.dll

LauncherはExitCodeを返すべき手続きだったので作成

```
public interface ILauncher : IProcedure
```

Inherited Members

[IProcedure.Run\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#).

Properties

ExitCode

```
int ExitCode { get; set; }
```

Property Value

[int](#)

Interface IProcedure

Namespace: [Rudeus.Procedure](#)

Assembly: RudeusBg.dll

ひとまとまりの処理を実行するインターフェース 例えば、アクセストークンの再発行、インストール済みアプリの送信など

```
public interface IProcedure
```

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#).

Methods

Run()

処理を実行する

Task **Run()**

Returns

[Task](#) 

Namespace RudeusBg

Classes

[Worker](#)

Interfaces

[IWorker](#)

Interface IWorker

Namespace: [RudeusBg](#)

Assembly: RudeusBg.dll

```
public interface IWorker
```

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T,T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T,T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T,T\)](#), [AssertEx.IsNot<T>\(T,T\)](#), [AssertEx.IsNot<T>\(T,T\)](#),
[AssertEx.IsNot<T>\(T,T\)](#), [AssertEx.IsNot<T>\(T,T,IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T,T,IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T,T,IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T,T\)](#), [AssertEx.IsSameReferenceAs<T>\(T,T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T,T\)](#), [AssertEx.Is<T>\(T,T\)](#), [AssertEx.Is<T>\(T,T\)](#), [AssertEx.Is<T>\(T,T\)](#),
[AssertEx.Is<T>\(T,T,IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T,T,IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T,T,IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T,Expression<Func<T,bool>>,string\)](#),
[AssertEx.Is<T>\(T,Expression<Func<T,bool>>,string\)](#),
[AssertEx.Is<T>\(T,Expression<Func<T,bool>>,string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object,object,string\)](#),
[AssertEx.IsNotStructuralEqual\(object,object,string\)](#), [AssertEx.IsNotStructuralEqual\(object,object,string\)](#),
[AssertEx.IsStructuralEqual\(object,object,string\)](#), [AssertEx.IsStructuralEqual\(object,object,string\)](#),
[AssertEx.IsStructuralEqual\(object,object,string\)](#).

Properties

AccessTokenValidator

```
IProcedure AccessTokenValidator { get; set; }
```

Property Value

[IProcedure](#)

ScheduledRelularExecuter

```
IProcedure ScheduledRelularExecuter { get; set; }
```

Property Value

[IProcedure](#)

UserLoginExecuter

```
IProcedure UserLoginExecuter { get; set; }
```

Property Value

[IProcedure](#)

Class Worker

Namespace: [RudeusBg](#)

Assembly: RudeusBg.dll

```
public class Worker : BackgroundService, IHostedService, IDisposable, IWorker
```

Inheritance

[object](#) ← [BackgroundService](#) ← Worker

Implements

[IHostedService](#), [IDisposable](#), [IWorker](#)

Inherited Members

[BackgroundService.StartAsync\(CancellationToken\)](#),
[BackgroundService.StopAsync\(CancellationToken\)](#), [BackgroundService.Dispose\(\)](#),
[BackgroundService.ExecuteTask](#), [object.Equals\(object\)](#), [object.Equals\(object, object\)](#),
[object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#),
[object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#)

[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#).

Constructors

Worker(ILogger<Worker>, IProcedure?, IProcedure?,
IProcedure?)

```
public Worker(ILogger<Worker> logger, IProcedure? at = null, IProcedure? sr = null,  
IProcedure? ul = null)
```

Parameters

logger [ILogger](#) <[Worker](#)>

at [IProcedure](#)

sr [IProcedure](#)

ul [IProcedure](#)

Properties

AccessTokenValidator

```
public IProcedure AccessTokenValidator { get; set; }
```

Property Value

[IProcedure](#)

ScheduledRelularExecuter

```
public IProcedure ScheduledRelularExecuter { get; set; }
```

Property Value

[IProcedure](#)

UserLoginExecuter

```
public IProcedure UserLoginExecuter { get; set; }
```

Property Value

[IProcedure](#)

settings

```
public static Settings settings { get; set; }
```

Property Value

[Settings](#)

Methods

ExecuteAsync(CancellationTokens)

This method is called when the [IHostedService](#) starts. The implementation should return a task that represents the lifetime of the long running operation(s) being performed.

```
protected override Task ExecuteAsync(CancellationTokens stoppingToken)
```

Parameters

stoppingToken [CancellationTokens](#)

Triggered when [StopAsync\(CancellationTokens\)](#) is called.

Returns

[Task](#)

A [Task](#) that represents the long running operations.

Remarks

See [Worker Services in .NET](#) for implementation guidelines.

Namespace RudeusBgForm

Classes

[TaskTrayForm](#)

Class TaskTrayForm

Namespace: [RudeusBgForm](#)

Assembly: RudeusBgForm.dll

```
public class TaskTrayForm : Form, IDropTarget, ISynchronizeInvoke, IWin32Window,
IBindableComponent, IComponent, IDisposable, IContainerControl
```

Inheritance

[object](#) ← [MarshalByRefObject](#) ← [Component](#) ← [Control](#) ← [ScrollableControl](#) ← [ContainerControl](#) ← [Form](#) ← TaskTrayForm

Implements

[IDropTarget](#), [ISynchronizeInvoke](#), [IWin32Window](#), [IBindableComponent](#), [IComponent](#), [IDisposable](#), [IContainerControl](#)

Inherited Members

[Form.SetVisibleCore\(bool\)](#), [Form.Activate\(\)](#), [Form.ActivateMdiChild\(Form\)](#), [Form.AddOwnedForm\(Form\)](#), [Form.AdjustFormScrollbars\(bool\)](#), [Form.Close\(\)](#), [Form.CreateAccessibilityInstance\(\)](#), [Form.CreateControlsInstance\(\)](#), [Form.CreateHandle\(\)](#), [Form.DefWndProc\(ref Message\)](#), [Form.ProcessMnemonic\(char\)](#), [Form.CenterToParent\(\)](#), [Form.CenterToScreen\(\)](#), [Form.LayoutMdi\(MdiLayout\)](#), [Form.OnActivated\(EventArgs\)](#), [Form.OnBackgroundImageChanged\(EventArgs\)](#), [Form.OnBackgroundImageLayoutChanged\(EventArgs\)](#), [Form.OnClosing\(CancelEventArgs\)](#), [Form.OnClosed\(EventArgs\)](#), [Form.OnFormClosing\(FormClosingEventArgs\)](#), [Form.OnFormClosed\(FormClosedEventArgs\)](#), [Form.OnCreateControl\(\)](#), [Form.OnDeactivate\(EventArgs\)](#), [Form.OnEnabledChanged\(EventArgs\)](#), [Form.OnEnter\(EventArgs\)](#), [Form.OnFontChanged\(EventArgs\)](#), [Form.OnGotFocus\(EventArgs\)](#), [Form.OnHandleCreated\(EventArgs\)](#), [Form.OnHandleDestroyed\(EventArgs\)](#), [Form.OnHelpButtonClicked\(CancelEventArgs\)](#), [Form.OnLayout\(LayoutEventArgs\)](#), [Form.OnLoad\(EventArgs\)](#), [Form.OnMaximizedBoundsChanged\(EventArgs\)](#), [Form.OnMaximumSizeChanged\(EventArgs\)](#), [Form.OnMinimumSizeChanged\(EventArgs\)](#), [Form.OnInputLanguageChanged\(InputLanguageChangedEventArgs\)](#), [Form.OnInputLanguageChanging\(InputLanguageChangingEventArgs\)](#), [Form.OnVisibleChanged\(EventArgs\)](#), [Form.OnMdiChildActivate\(EventArgs\)](#), [Form.OnMenuStart\(EventArgs\)](#), [Form.OnMenuComplete\(EventArgs\)](#), [Form.OnPaint\(PaintEventArgs\)](#), [Form.OnResize\(EventArgs\)](#), [Form.OnDpiChanged\(DpiChangedEventArgs\)](#), [Form.OnGetDpiScaledSize\(int, int, ref Size\)](#), [Form.OnRightToLeftLayoutChanged\(EventArgs\)](#), [Form.OnShown\(EventArgs\)](#),

[Form.OnTextChanged\(EventArgs\)](#), [Form.ProcessCmdKey\(ref Message, Keys\)](#),
[Form.ProcessDialogKey\(Keys\)](#), [Form.ProcessDialogChar\(char\)](#),
[Form.ProcessKeyPreview\(ref Message\)](#), [Form.ProcessTabKey\(bool\)](#),
[Form.RemoveOwnedForm\(Form\)](#), [Form.Select\(bool, bool\)](#),
[Form.ScaleMinMaxSize\(float, float, bool\)](#), [Form.GetScaledBounds\(Rectangle, SizeF, BoundsSpecified\)](#),
[Form.ScaleControl\(SizeF, BoundsSpecified\)](#), [Form.SetBoundsCore\(int, int, int, int, BoundsSpecified\)](#),
[Form.SetClientSizeCore\(int, int\)](#), [Form.SetDesktopBounds\(int, int, int, int\)](#),
[Form.SetDesktopLocation\(int, int\)](#), [Form.Show\(IWin32Window\)](#), [Form.ShowDialog\(\)](#),
[Form.ShowDialog\(IWin32Window\)](#), [Form.ToString\(\)](#), [Form.UpdateDefaultButton\(\)](#),
[Form.OnResizeBegin\(EventArgs\)](#), [Form.OnResizeEnd\(EventArgs\)](#),
[Form.OnStyleChanged\(EventArgs\)](#), [Form.ValidateChildren\(\)](#),
[Form.ValidateChildren\(ValidationConstraints\)](#), [Form.WndProc\(ref Message\)](#), [Form.AcceptButton](#),
[Form.ActiveForm](#), [Form.ActiveMdiChild](#), [Form.AllowTransparency](#), [Form.AutoScroll](#),
[Form.AutoSize](#), [Form.AutoSizeMode](#), [Form.AutoValidate](#), [Form.BackColor](#),
[Form.FormBorderStyle](#), [Form.CancelButton](#), [Form.ClientSize](#), [Form.ControlBox](#),
[Form.CreateParams](#), [Form.DefaultImeMode](#), [Form.DefaultSize](#), [Form.DesktopBounds](#),
[Form.DesktopLocation](#), [Form.DialogResult](#), [Form.HelpButton](#), [Form.Icon](#), [Form.IsMdiChild](#),
[Form.IsMdiContainer](#), [Form.IsRestrictedWindow](#), [Form.KeyPreview](#), [Form.Location](#),
[Form.MaximizedBounds](#), [Form.MaximumSize](#), [Form.MainMenuStrip](#), [Form.MinimumSize](#),
[Form.MaximizeBox](#), [Form.MdiChildren](#), [Form.MdiChildrenMinimizedAnchorBottom](#),
[Form.MdiParent](#), [Form.MinimizeBox](#), [Form.Modal](#), [Form.Opacity](#), [Form.OwnedForms](#),
[Form.Owner](#), [Form.RestoreBounds](#), [Form.RightToLeftLayout](#), [Form.ShowInTaskbar](#),
[Form.ShowIcon](#), [Form.ShowWithoutActivation](#), [Form.Size](#), [Form.SizeGripStyle](#),
[Form.StartPosition](#), [Form.Text](#), [Form.TopLevel](#), [Form.TopMost](#), [Form.TransparencyKey](#),
[Form.WindowState](#), [Form.AutoSizeChanged](#), [Form.AutoValidateChanged](#),
[Form.HelpButtonClicked](#), [Form.MaximizedBoundsChanged](#), [Form.MaximumSizeChanged](#),
[Form.MinimumSizeChanged](#), [Form.Activated](#), [Form.Deactivate](#), [Form.FormClosing](#),
[Form.FormClosed](#), [Form.Load](#), [Form.MdiChildActivate](#), [Form.MenuComplete](#),
[Form.MenuStart](#), [Form.InputLanguageChanged](#), [Form.InputLanguageChanging](#),
[Form.RightToLeftLayoutChanged](#), [Form.Shown](#), [Form.DpiChanged](#), [Form.ResizeBegin](#),
[Form.ResizeEnd](#), [ContainerControl.OnAutoValidateChanged\(EventArgs\)](#),
[ContainerControl.OnMove\(EventArgs\)](#), [ContainerControl.OnParentChanged\(EventArgs\)](#),
[ContainerControl.PerformAutoScale\(\)](#), [ContainerControl.RescaleConstantsForDpi\(int, int\)](#),
[ContainerControl.Validate\(\)](#), [ContainerControl.Validate\(bool\)](#),
[ContainerControl.AutoScaleDimensions](#), [ContainerControl.AutoScaleFactor](#),
[ContainerControl.AutoScaleMode](#), [ContainerControl.BindingContext](#),
[ContainerControl.CanEnableIme](#), [ContainerControl.ActiveControl](#),
[ContainerControl.CurrentAutoScaleDimensions](#), [ContainerControl.ParentForm](#),
[ScrollableControl.ScrollStateAutoScrolling](#), [ScrollableControl.ScrollStateHScrollVisible](#),
[ScrollableControl.ScrollStateVScrollVisible](#), [ScrollableControl.ScrollStateUserHasScrolled](#),

[ScrollableControl.ScrollStateFullDrag](#) , [ScrollableControl.GetScrollState\(int\)](#) ,
[ScrollableControl.OnMouseWheel\(MouseEventArgs\)](#) ,
[ScrollableControl.OnRightToLeftChanged\(EventArgs\)](#) ,
[ScrollableControl.OnPaintBackground\(PaintEventArgs\)](#) ,
[ScrollableControl.OnPaddingChanged\(EventArgs\)](#) , [ScrollableControl.SetDisplayRectLocation\(int, int\)](#) ,
[ScrollableControl.ScrollControlIntoView\(Control\)](#) , [ScrollableControl.ScrollToControl\(Control\)](#) ,
[ScrollableControl.OnScroll\(ScrollEventArgs\)](#) , [ScrollableControl.SetAutoScrollMargin\(int, int\)](#) ,
[ScrollableControl.SetScrollState\(int, bool\)](#) , [ScrollableControl.AutoScrollMargin](#) ,
[ScrollableControl.AutoScrollPosition](#) , [ScrollableControl.AutoScrollMinSize](#) ,
[ScrollableControl.DisplayRectangle](#) , [ScrollableControl.HScroll](#) , [ScrollableControl.HorizontalScroll](#) ,
[ScrollableControl.VScroll](#) , [ScrollableControl.VerticalScroll](#) , [ScrollableControl.Scroll](#) ,
[Control.GetAccessibilityObjectById\(int\)](#) , [Control.SetAutoSizeMode\(AutoSizeMode\)](#) ,
[Control.GetAutoSizeMode\(\)](#) , [Control.GetPreferredSize\(Size\)](#) ,
[Control.AccessibilityNotifyClients\(AccessibleEvents, int\)](#) ,
[Control.AccessibilityNotifyClients\(AccessibleEvents, int, int\)](#) , [Control.BeginInvoke\(Delegate\)](#) ,
[Control.BeginInvoke\(Action\)](#) , [Control.BeginInvoke\(Delegate, params object\[\]\)](#) ,
[Control.BringToFront\(\)](#) , [Control.Contains\(Control\)](#) , [Control.CreateGraphics\(\)](#) ,
[Control.CreateControl\(\)](#) , [Control.DestroyHandle\(\)](#) , [Control.DoDragDrop\(object, DragDropEffects\)](#) ,
[Control.DoDragDrop\(object, DragDropEffects, Bitmap, Point, bool\)](#) ,
[Control.DrawToBitmap\(Bitmap, Rectangle\)](#) , [Control.EndInvoke\(IAsyncResult\)](#) , [Control.FindForm\(\)](#) ,
[Control.GetTopLevel\(\)](#) , [Control.RaiseKeyEvent\(object, KeyEventArgs\)](#) ,
[Control.RaiseMouseEvent\(object, MouseEventArgs\)](#) , [Control.Focus\(\)](#) ,
[Control.FromChildHandle\(nint\)](#) , [Control.FromHandle\(nint\)](#) ,
[Control.GetChildAtPoint\(Point, GetChildAtPointSkip\)](#) , [Control.GetChildAtPoint\(Point\)](#) ,
[Control.GetContainerControl\(\)](#) , [Control.GetNextControl\(Control, bool\)](#) ,
[Control.GetStyle\(ControlStyles\)](#) , [Control.Hide\(\)](#) , [Control.InitLayout\(\)](#) , [Control.Invalidate\(Region\)](#) ,
[Control.Invalidate\(Region, bool\)](#) , [Control.Invalidate\(\)](#) , [Control.Invalidate\(bool\)](#) ,
[Control.Invalidate\(Rectangle\)](#) , [Control.Invalidate\(Rectangle, bool\)](#) , [Control.Invoke\(Action\)](#) ,
[Control.Invoke\(Delegate\)](#) , [Control.Invoke\(Delegate, params object\[\]\)](#) ,
[Control.Invoke<T>\(Func<T>\)](#) , [Control.InvokePaint\(Control, PaintEventArgs\)](#) ,
[Control.InvokePaintBackground\(Control, PaintEventArgs\)](#) , [Control.IsKeyLocked\(Keys\)](#) ,
[Control.IsInputChar\(char\)](#) , [Control.IsInputKey\(Keys\)](#) , [Control.IsMnemonic\(char, string\)](#) ,
[Control.LogicalToDeviceUnits\(int\)](#) , [Control.LogicalToDeviceUnits\(Size\)](#) ,
[Control.ScaleBitmapLogicalToDevice\(ref Bitmap\)](#) , [Control.NotifyInvalidate\(Rectangle\)](#) ,
[Control.InvokeOnClick\(Control, EventArgs\)](#) , [Control.OnAutoSizeChanged\(EventArgs\)](#) ,
[Control.OnBackColorChanged\(EventArgs\)](#) , [Control.OnBindingContextChanged\(EventArgs\)](#) ,
[Control.OnCausesValidationChanged\(EventArgs\)](#) , [Control.OnContextMenuStripChanged\(EventArgs\)](#) ,
[Control.OnCursorChanged\(EventArgs\)](#) , [Control.OnDataContextChanged\(EventArgs\)](#) ,
[Control.OnDockChanged\(EventArgs\)](#) , [Control.OnForeColorChanged\(EventArgs\)](#) ,
[Control.OnNotifyMessage\(Message\)](#) , [Control.OnParentBackColorChanged\(EventArgs\)](#) ,

[Control.OnParentBackgroundImageChanged\(EventArgs\)](#),
[Control.OnParentBindingContextChanged\(EventArgs\)](#), [Control.OnParentCursorChanged\(EventArgs\)](#),
[Control.OnParentDataContextChanged\(EventArgs\)](#), [Control.OnParentEnabledChanged\(EventArgs\)](#),
[Control.OnParentFontChanged\(EventArgs\)](#), [Control.OnParentForeColorChanged\(EventArgs\)](#),
[Control.OnParentRightToLeftChanged\(EventArgs\)](#), [Control.OnParentVisibleChanged\(EventArgs\)](#),
[Control.OnPrint\(PaintEventArgs\)](#), [Control.OnTabIndexChanged\(EventArgs\)](#),
[Control.OnTabStopChanged\(EventArgs\)](#), [Control.OnClick\(EventArgs\)](#),
[Control.OnClientSizeChanged\(EventArgs\)](#), [Control.OnControlAdded\(ControlEventArgs\)](#),
[Control.OnControlRemoved\(ControlEventArgs\)](#), [Control.OnLocationChanged\(EventArgs\)](#),
[Control.OnDoubleClick\(EventArgs\)](#), [Control.OnDragEnter\(DragEventArgs\)](#),
[Control.OnDragOver\(DragEventArgs\)](#), [Control.OnDragLeave\(EventArgs\)](#),
[Control.OnDragDrop\(DragEventArgs\)](#), [Control.OnGiveFeedback\(GiveFeedbackEventArgs\)](#),
[Control.InvokeGotFocus\(Control, EventArgs\)](#), [Control.OnHelpRequested\(HelpEventArgs\)](#),
[Control.OnInvalidated\(InvalidateEventArgs\)](#), [Control.OnKeyDown\(KeyEventArgs\)](#),
[Control.OnKeyPress\(KeyPressEventArgs\)](#), [Control.OnKeyUp\(KeyEventArgs\)](#),
[Control.OnLeave\(EventArgs\)](#), [Control.InvokeLostFocus\(Control, EventArgs\)](#),
[Control.OnLostFocus\(EventArgs\)](#), [Control.OnMarginChanged\(EventArgs\)](#),
[Control.OnMouseDoubleClick\(MouseEventArgs\)](#), [Control.OnMouseClick\(MouseEventArgs\)](#),
[Control.OnMouseCaptureChanged\(EventArgs\)](#), [Control.OnMouseDown\(MouseEventArgs\)](#),
[Control.OnMouseEnter\(EventArgs\)](#), [Control.OnMouseLeave\(EventArgs\)](#),
[Control.OnDpiChangedBeforeParent\(EventArgs\)](#), [Control.OnDpiChangedAfterParent\(EventArgs\)](#),
[Control.OnMouseHover\(EventArgs\)](#), [Control.OnMouseMove\(MouseEventArgs\)](#),
[Control.OnMouseUp\(MouseEventArgs\)](#),
[Control.OnQueryContinueDrag\(QueryContinueDragEventArgs\)](#),
[Control.OnRegionChanged\(EventArgs\)](#), [Control.OnPreviewKeyDown\(PreviewKeyDownEventArgs\)](#),
[Control.OnSizeChanged\(EventArgs\)](#), [Control.OnChangeUICues\(UICuesEventArgs\)](#),
[Control.OnSystemColorsChanged\(EventArgs\)](#), [Control.OnValidating\(CancelEventArgs\)](#),
[Control.OnValidated\(EventArgs\)](#), [Control.PerformLayout\(\)](#), [Control.PerformLayout\(Control, string\)](#),
[Control.PointToClient\(Point\)](#), [Control.PointToScreen\(Point\)](#),
[Control.PreProcessMessage\(ref Message\)](#), [Control.PreProcessControlMessage\(ref Message\)](#),
[Control.ProcessKeyEventArgs\(ref Message\)](#), [Control.ProcessKeyMessage\(ref Message\)](#),
[Control.RaiseDragEvent\(object, DragEventArgs\)](#), [Control.RaisePaintEvent\(object, PaintEventArgs\)](#),
[Control.RecreateHandle\(\)](#), [Control.RectangleToClient\(Rectangle\)](#),
[Control.RectangleToScreen\(Rectangle\)](#), [Control.ReflectMessage\(nint, ref Message\)](#),
[Control.Refresh\(\)](#), [Control.ResetMouseEventArgs\(\)](#), [Control.ResetText\(\)](#), [Control.ResumeLayout\(\)](#),
[Control.ResumeLayout\(bool\)](#), [Control.Scale\(SizeF\)](#), [Control.Select\(\)](#),
[Control.SelectNextControl\(Control, bool, bool, bool, bool\)](#), [Control.SendToBack\(\)](#),
[Control.SetBounds\(int, int, int, int\)](#), [Control.SetBounds\(int, int, int, int, BoundsSpecified\)](#),
[Control.SizeFromClientSize\(Size\)](#), [Control.SetStyle\(ControlStyles, bool\)](#), [Control.SetTopLevel\(bool\)](#),
[Control.RtlTranslateAlignment\(HorizontalAlignment\)](#),

[Control.RtlTranslateAlignment\(LeftRightAlignment\)](#),
[Control.RtlTranslateAlignment\(ContentAlignment\)](#),
[Control.RtlTranslateHorizontal\(HorizontalAlignment\)](#),
[Control.RtlTranslateLeftRight\(LeftRightAlignment\)](#), [Control.RtlTranslateContent\(ContentAlignment\)](#),
[Control.Show\(\)](#), [Control.SuspendLayout\(\)](#), [Control.Update\(\)](#), [Control.UpdateBounds\(\)](#),
[Control.UpdateBounds\(int, int, int, int\)](#), [Control.UpdateBounds\(int, int, int, int, int, int\)](#),
[Control.UpdateZOrder\(\)](#), [Control.UpdateStyles\(\)](#), [Control.OnImeModeChanged\(EventArgs\)](#),
[Control.AccessibilityObject](#), [Control.AccessibleDefaultActionDescription](#),
[Control.AccessibleDescription](#), [Control.AccessibleName](#), [Control.AccessibleRole](#),
[Control.AllowDrop](#), [Control.Anchor](#), [Control.AutoScrollOffset](#), [Control.LayoutEngine](#),
[Control.DataContext](#), [Control.BackgroundImage](#), [Control.BackgroundImageLayout](#),
[Control.Bottom](#), [Control.Bounds](#), [Control.CanFocus](#), [Control.CanRaiseEvents](#),
[Control.CanSelect](#), [Control.Capture](#), [Control.CausesValidation](#),
[Control.CheckForIllegalCrossThreadCalls](#), [Control.ClientRectangle](#), [Control.CompanyName](#),
[Control.ContainsFocus](#), [Control.ContextMenuStrip](#), [Control.Controls](#), [Control.Created](#),
[Control.Cursor](#), [Control.DataBindings](#), [Control.DefaultBackColor](#), [Control.DefaultCursor](#),
[Control.DefaultFont](#), [Control.DefaultForeColor](#), [Control.DefaultMargin](#),
[Control.DefaultMaximumSize](#), [Control.DefaultMinimumSize](#), [Control.DefaultPadding](#),
[Control.DeviceDpi](#), [Control.IsDisposed](#), [Control.Disposing](#), [Control.Dock](#),
[Control.DoubleBuffered](#), [Control.Enabled](#), [Control.Focused](#), [Control.Font](#), [Control.FontHeight](#),
[Control.ForeColor](#), [Control.Handle](#), [Control.HasChildren](#), [Control.Height](#),
[Control.IsHandleCreated](#), [Control.InvokeRequired](#), [Control.IsAccessible](#),
[Control.IsAncestorSiteInDesignMode](#), [Control.IsMirrored](#), [Control.Left](#), [Control.Margin](#),
[Control.ModifierKeys](#), [Control.MouseButtons](#), [Control.MousePosition](#), [Control.Name](#),
[Control.Parent](#), [Control.ProductName](#), [Control.ProductVersion](#), [Control.RecreatingHandle](#),
[Control.Region](#), [Control.RenderRightToLeft](#), [Control.ResizeRedraw](#), [Control.Right](#),
[Control.RightToLeft](#), [Control.ScaleChildren](#), [Control.Site](#), [Control.TabIndex](#), [Control.TabStop](#),
[Control.Tag](#), [Control.Top](#), [Control.TopLevelControl](#), [Control.ShowKeyboardCues](#),
[Control.ShowFocusCues](#), [Control.UseWaitCursor](#), [Control.Visible](#), [Control.Width](#),
[Control.PreferredSize](#), [Control.Padding](#), [Control.ImeMode](#), [Control.ImeModeBase](#),
[Control.PropagatingImeMode](#), [Control.BackColorChanged](#), [Control.BackgroundImageChanged](#),
[Control.BackgroundImageLayoutChanged](#), [Control.BindingContextChanged](#),
[Control.CausesValidationChanged](#), [Control.ClientSizeChanged](#),
[Control.ContextMenuStripChanged](#), [Control.CursorChanged](#), [Control.DockChanged](#),
[Control.EnabledChanged](#), [Control.FontChanged](#), [Control.ForeColorChanged](#),
[Control.LocationChanged](#), [Control.MarginChanged](#), [Control.RegionChanged](#),
[Control.RightToLeftChanged](#), [Control.SizeChanged](#), [Control.TabIndexChanged](#),
[Control.TabStopChanged](#), [Control.TextChanged](#), [Control.VisibleChanged](#), [Control.Click](#),
[Control.ControlAdded](#), [Control.ControlRemoved](#), [Control.DataContextChanged](#),
[Control.DragDrop](#), [Control.DragEnter](#), [Control.DragOver](#), [Control.DragLeave](#),

[Control.GiveFeedback](#), [Control.HandleCreated](#), [Control.HandleDestroyed](#),
[Control.HelpRequested](#), [Control.Invalidated](#), [Control.PaddingChanged](#), [Control.Paint](#),
[Control.QueryContinueDrag](#), [Control.QueryAccessibilityHelp](#), [Control.DoubleClick](#),
[Control.Enter](#), [Control.GotFocus](#), [Control.KeyDown](#), [Control.KeyPress](#), [Control.KeyUp](#),
[Control.Layout](#), [Control.Leave](#), [Control.LostFocus](#), [Control.MouseClick](#),
[Control.MouseDoubleClick](#), [Control.MouseCaptureChanged](#), [Control.MouseDown](#),
[Control.MouseEnter](#), [Control.MouseLeave](#), [Control.DpiChangedBeforeParent](#),
[Control.DpiChangedAfterParent](#), [Control.MouseHover](#), [Control.MouseMove](#), [Control.MouseUp](#),
[Control.MouseWheel](#), [Control.Move](#), [Control.PreviewKeyDown](#), [Control.Resize](#),
[Control.ChangeUICues](#), [Control.StyleChanged](#), [Control.SystemColorsChanged](#),
[Control.Validating](#), [Control.Validated](#), [Control.ParentChanged](#), [Control.ImeModeChanged](#),
[Component.Dispose\(\)](#), [Component.GetService\(Type\)](#), [Component.Container](#),
[Component.DesignMode](#), [Component.Events](#), [Component.Disposed](#),
[MarshalByRefObject.GetLifetimeService\(\)](#), [MarshalByRefObject.InitializeLifetimeService\(\)](#),
[MarshalByRefObject.MemberwiseClone\(bool\)](#), [object.Equals\(object\)](#), [object.Equals\(object, object\)](#),
[object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#),
[object.ReferenceEquals\(object, object\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Constructors

TaskTrayForm()

```
public TaskTrayForm()
```


Methods

Dispose(bool)

Clean up any resources being used.

```
protected override void Dispose(bool disposing)
```

Parameters

disposing [bool](#)

true if managed resources should be disposed; otherwise, false.

Namespace RudeusBgFormTest

Classes

[UnitTest1](#)

Class UnitTest1

Namespace: [RudeusBgFormTest](#)








Assembly: RudeusBgFormTest.dll

```
public class UnitTest1
```










































Inheritance

[object](#)  ← UnitTest1

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#) , [AssertEx.AsDynamic<T>\(T\)](#) , [AssertEx.AsDynamic<T>\(T\)](#) ,
[AssertEx.IsNotNull<T>\(T\)](#) , [AssertEx.IsNotNull<T>\(T\)](#) , [AssertEx.IsNotNull<T>\(T\)](#) ,
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#) , [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#) ,
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#) , [AssertEx.IsNot<T>\(T, T\)](#) , [AssertEx.IsNot<T>\(T, T\)](#) ,
[AssertEx.IsNot<T>\(T, T\)](#) , [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#) ,
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#) , [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#) ,
[AssertEx.IsNull<T>\(T\)](#) , [AssertEx.IsNull<T>\(T\)](#) , [AssertEx.IsNull<T>\(T\)](#) ,
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#) , [AssertEx.IsSameReferenceAs<T>\(T, T\)](#) ,
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#) , [AssertEx.Is<T>\(T, T\)](#) , [AssertEx.Is<T>\(T, T\)](#) ,
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#) , [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#) ,
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#) , [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#) ,
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#) ,
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#) , [AssertEx.IsInstanceOf<TExpected>\(object\)](#) ,
[AssertEx.IsInstanceOf<TExpected>\(object\)](#) , [AssertEx.IsInstanceOf<TExpected>\(object\)](#) ,
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) , [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) ,
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) , [AssertEx.IsNotStructuralEqual\(object, object, string\)](#) ,
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#) , [AssertEx.IsNotStructuralEqual\(object, object, string\)](#) ,
[AssertEx.IsStructuralEqual\(object, object, string\)](#) , [AssertEx.IsStructuralEqual\(object, object, string\)](#) ,
[AssertEx.IsStructuralEqual\(object, object, string\)](#) 

Methods

Test1()

```
[Fact]  
public void Test1()
```


Namespace RudeusBgInitializer

Classes

[Program](#)

Class Program

Namespace: [RudeusBgInitializer](#)








Assembly: RudeusBgInitializer.dll

```
public class Program
```








































Inheritance

[object](#)  ← Program

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#) , [AssertEx.AsDynamic<T>\(T\)](#) , [AssertEx.AsDynamic<T>\(T\)](#) ,
[AssertEx.IsNotNull<T>\(T\)](#) , [AssertEx.IsNotNull<T>\(T\)](#) , [AssertEx.IsNotNull<T>\(T\)](#) ,
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#) , [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#) ,
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#) , [AssertEx.IsNot<T>\(T, T\)](#) , [AssertEx.IsNot<T>\(T, T\)](#) ,
[AssertEx.IsNot<T>\(T, T\)](#) , [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#) ,
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#) , [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#) ,
[AssertEx.IsNull<T>\(T\)](#) , [AssertEx.IsNull<T>\(T\)](#) , [AssertEx.IsNull<T>\(T\)](#) ,
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#) , [AssertEx.IsSameReferenceAs<T>\(T, T\)](#) ,
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#) , [AssertEx.Is<T>\(T, T\)](#) , [AssertEx.Is<T>\(T, T\)](#) ,
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#) , [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#) ,
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#) , [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#) ,
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#) ,
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#) , [AssertEx.IsInstanceOf<TExpected>\(object\)](#) ,
[AssertEx.IsInstanceOf<TExpected>\(object\)](#) , [AssertEx.IsInstanceOf<TExpected>\(object\)](#) ,
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) , [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) ,
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#) , [AssertEx.IsNotStructuralEqual\(object, object, string\)](#) ,
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#) , [AssertEx.IsNotStructuralEqual\(object, object, string\)](#) ,
[AssertEx.IsStructuralEqual\(object, object, string\)](#) , [AssertEx.IsStructuralEqual\(object, object, string\)](#) ,
[AssertEx.IsStructuralEqual\(object, object, string\)](#) 

Properties

CheckLogMessage

```
public static bool? CheckLogMessage { get; set; }
```

Property Value

[bool](#)[?]

_certificateInstaller

```
public static IProcedure _certificateInstaller { get; set; }
```

Property Value

[IProcedure](#)

_registryInitializer

```
public static IProcedure _registryInitializer { get; set; }
```

Property Value

[IProcedure](#)

_serverRegister

```
public static IProcedure _serverRegister { get; set; }
```

Property Value

[IProcedure](#)

_taskInitializer

```
public static IPProcedure _taskInitializer { get; set; }
```

Property Value

[IPProcedure](#)

Methods

Main(string[])

```
public static void Main(string[] args)
```

Parameters

args [string](#)[]

MainAsync(string[])

```
public static Task MainAsync(string[] args)
```

Parameters

args [string](#)[]

Returns

[Task](#)

Namespace RudeusBgInitializerTest

Classes

[ProgramTests](#)

[TaskInitializerTests](#)

Class ProgramTests

Namespace: [RudeusBgInitializerTest](#)

Assembly: RudeusBgInitializerTest.dll

```
public class ProgramTests
```

Inheritance

[object](#) ← ProgramTests

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#).

Methods

TestMain()

```
[Fact]  
public void TestMain()
```

Class TaskInitializerTests

Namespace: [RudeusBgInitializerTest](#)

Assembly: RudeusBgInitializerTest.dll

```
public class TaskInitializerTests
```

Inheritance

[object](#) ← TaskInitializerTests

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Namespace RudeusBgTest

Classes

[UnitTest1](#)

Class UnitTest1

Namespace: [RudeusBgTest](#)








Assembly: RudeusBgTest.dll

```
public class UnitTest1
```

Inheritance

[object](#)  ← UnitTest1

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#).

Methods

Test1()

```
[Fact]  
public void Test1()
```

Namespace RudeusLauncherTest

Classes

[LauncherTests](#)

[ProgramTests](#)

[UpdaterTests](#)

Class LauncherTests

Namespace: [RudeusLauncherTest](#)

Assembly: RudeusLauncherTest.dll

```
public class LauncherTests
```

Inheritance

[object](#) ← LauncherTests

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

TestRun()

```
[Fact]  
public Task TestRun()
```

Returns

[Task](#)

TestRun2()

```
[Fact]  
public Task TestRun2()
```

Returns

[Task](#)

TestRun3()

```
[Fact]  
public Task TestRun3()
```

Returns

[Task](#)

Class ProgramTests

Namespace: [RudeusLauncherTest](#)

Assembly: RudeusLauncherTest.dll

```
public class ProgramTests
```

Inheritance

[object](#) ← ProgramTests

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

Test1()

```
[Fact]  
public void Test1()
```


Class UpdaterTests

Namespace: [RudeusLauncherTest](#)

Assembly: RudeusLauncherTest.dll

```
public class UpdaterTests
```

Inheritance

[object](#) ← UpdaterTests

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Namespace RudeusSharedLibTest

Classes

[ConstantsTests](#)

[UnitTestSample](#)

[UtilsTests](#)

Class ConstantsTests

Namespace: [RudeusSharedLibTest](#)








Assembly: RudeusSharedLibTest.dll

```
public class ConstantsTests
```

Inheritance

[object](#)  ← ConstantsTests

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#).

Class UnitTestSample

Namespace: [RudeusSharedLibTest](#)

Assembly: RudeusSharedLibTest.dll

```
public class UnitTestSample
```

Inheritance

[object](#) ← UnitTestSample

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

TestTest()

```
[Fact(DisplayName = "テストの書き方サンプル")]  
public void TestTest()
```

Class UtilsTests

Namespace: [RudeusSharedLibTest](#)

Assembly: RudeusSharedLibTest.dll

```
public class UtilsTests
```

Inheritance

[object](#) ← [UtilsTests](#)

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

TestCompareVersionString()

```
[Fact(DisplayName = "CompareVersionString")]  
public void TestCompareVersionString()
```

TestConcatStudentNumberFromMail()

```
[Fact(DisplayName = "ConcatStudentNumberFromMail")]  
public void TestConcatStudentNumberFromMail()
```

TestIsStudentMailAddress()

```
[Fact(DisplayName = "IsStudentMailAddress")]  
public void TestIsStudentMailAddress()
```

Namespace RudeusSharedLibTest.API

Classes

[RemoteAPITests](#)

Class RemoteAPITests

Namespace: [RudeusSharedLibTest.API](#)

Assembly: RudeusSharedLibTest.dll

```
public class RemoteAPITests
```

Inheritance

[object](#) ← RemoteAPITests

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

TestBuildHttpRequestMessage()

```
[Fact]
public void TestBuildHttpRequestMessage()
```

TestLoginDevice()

```
[Fact]
public void TestLoginDevice()
```

TestRegisterDevice()

```
[Fact]
public void TestRegisterDevice()
```

TestRegisterDeviceException()

```
[Fact]
public void TestRegisterDeviceException()
```

TestSendInstalledApps()

```
[Fact]
public void TestSendInstalledApps()
```

TestUpdateDevice()

```
[Fact]
public void TestUpdateDevice()
```

Namespace RudeusSharedLibTest.Procedure

Classes

[CertificateInstallerTests](#)

[FakeProcedureTests](#)

[RegistryInitializerTests](#)

Class CertificateInstallerTests

Namespace: [RudeusSharedLibTest.Procedure](#)

Assembly: RudeusSharedLibTest.dll

```
public class CertificateInstallerTests
```

Inheritance

[object](#) ← CertificateInstallerTests

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

TestRun()

```
[Fact]  
public void TestRun()
```

Class FakeProcedureTests

Namespace: [RudeusSharedLibTest.Procedure](#)

Assembly: RudeusSharedLibTest.dll

```
public class FakeProcedureTests
```

Inheritance

[object](#) ← FakeProcedureTests

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods


[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

TestRun()

```
[Fact]  
public Task TestRun()
```

Returns

[Task](#)

Class RegistryInitializerTests

Namespace: [RudeusSharedLibTest.Procedure](#)

Assembly: RudeusSharedLibTest.dll

```
public class RegistryInitializerTests
```

Inheritance

[object](#) ← RegistryInitializerTests

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Extension Methods

[AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#), [AssertEx.AsDynamic<T>\(T\)](#),
[AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#), [AssertEx.IsNotNull<T>\(T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsNotSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T\)](#),
[AssertEx.IsNot<T>\(T, T\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.IsNot<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#), [AssertEx.IsNull<T>\(T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.IsSameReferenceAs<T>\(T, T\)](#),
[AssertEx.IsSameReferenceAs<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#), [AssertEx.Is<T>\(T, T\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#),
[AssertEx.Is<T>\(T, T, IEqualityComparer<T>\)](#), [AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#),
[AssertEx.Is<T>\(T, Expression<Func<T, bool>>, string\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsInstanceOf<TExpected>\(object\)](#), [AssertEx.IsInstanceOf<TExpected>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotInstanceOf<TWrong>\(object\)](#),
[AssertEx.IsNotInstanceOf<TWrong>\(object\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsNotStructuralEqual\(object, object, string\)](#), [AssertEx.IsNotStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#), [AssertEx.IsStructuralEqual\(object, object, string\)](#),
[AssertEx.IsStructuralEqual\(object, object, string\)](#)

Methods

TestRun()


```
[Fact]  
public void TestRun()
```

Namespace Xunit

Classes

[AssertEx](#)

Class AssertEx

Namespace: [Xunit](#)








Assembly: RudeusBgInitializerTest.dll

```
[ContractVerification(false)]  
public static class AssertEx
```

Inheritance

[object](#)  ← AssertEx

Inherited Members

[object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  ,
[object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#)  , [object.ToString\(\)](#) 

Methods

AsDynamic<T>(T)

to DynamicAccessor that can call private method/field/property/indexer.

```
public static dynamic AsDynamic<T>(this T target)
```

Parameters

target T

Returns

dynamic

Type Parameters

T

IsFalse(bool)

Is(false)

```
public static void IsFalse(this bool value)
```

Parameters

value [bool](#)

IsInstanceOf<TExpected>(object)

Assert.IsType

```
public static TExpected IsInstanceOf<TExpected>(this object value)
```

Parameters

value [object](#)

Returns

TExpected

Type Parameters

TExpected

IsNotInstanceOf<TWrong>(object)

Assert.IsNotType

```
public static void IsNotInstanceOf<TWrong>(this object value)
```

Parameters

value [object](#)

Type Parameters

Wrong

IsNotNull<T>(T)

Assert.NotNull

```
public static void IsNotNull<T>(this T value)
```

Parameters

value T

Type Parameters

T

IsNotSameReferenceAs<T>(T, T)

Assert.NotSame

```
public static void IsNotSameReferenceAs<T>(this T actual, T notExpected)
```

Parameters

actual T

notExpected T

Type Parameters

T

IsNotStructuralEqual(object, object, string)

Assert by deep recursive value equality compare

```
public static void IsNotStructuralEqual(this object actual, object expected, string message = "")
```

Parameters

actual [object](#)

expected [object](#)

message [string](#)

IsNot<T>(IEnumerable<T>, IEnumerable<T>)

Assert.NotEqual(sequence value compare)

```
public static void IsNot<T>(this IEnumerable<T> actual, IEnumerable<T> notExpected)
```

Parameters

actual [IEnumerable](#)<T>

notExpected [IEnumerable](#)<T>

Type Parameters

T

IsNot<T>(IEnumerable<T>, IEnumerable<T>, IEqualityComparer<T>)

Assert.False(actual.SequenceEqual(notExpected, comparer))

```
public static void IsNot<T>(this IEnumerable<T> actual, IEnumerable<T> notExpected, IEqualityComparer<T> comparer)
```

Parameters

actual [IEnumerable](#)<T>

notExpected [IEnumerable](#) <T>

comparer [IEqualityComparer](#) <T>

Type Parameters

T

IsNot<T>(IEnumerable<T>, IEnumerable<T>, Func<T, T, bool>)

Assert.False(actual.SequenceEqual(notExpected, comparison))

```
public static void IsNot<T>(this IEnumerable<T> actual, IEnumerable<T> notExpected, Func<T, T, bool> equalityComparison)
```

Parameters

actual [IEnumerable](#) <T>

notExpected [IEnumerable](#) <T>

equalityComparison [Func](#) <T, T, [bool](#) >

Type Parameters

T

IsNot<T>(IEnumerable<T>, params T[])

Assert.NotEqual(sequence value compare)

```
public static void IsNot<T>(this IEnumerable<T> actual, params T[] notExpected)
```

Parameters

actual [IEnumerable](#) <T>

notExpected T[]

Type Parameters

T

IsNot<T>(T, T)

Assert.NotEqual, if T is IEnumerable then check value equality

```
public static void IsNot<T>(this T actual, T notExpected)
```

Parameters

actual T

notExpected T

Type Parameters

T

IsNot<T>(T, T, IEqualityComparer<T>)

Assert.NotEqual

```
public static void IsNot<T>(this T actual, T notExpected, IEqualityComparer<T> comparer)
```

Parameters

actual T

notExpected T

comparer [IEqualityComparer](#)<T>

Type Parameters

T

IsNull<T>(T)

Assert.Null

```
public static void IsNull<T>(this T value)
```

Parameters

value T

Type Parameters

T

IsSameReferenceAs<T>(T, T)

Assert.Same

```
public static void IsSameReferenceAs<T>(this T actual, T expected)
```

Parameters

actual T

expected T

Type Parameters

T

IsStructuralEqual(object, object, string)

Assert by deep recursive value equality compare

```
public static void IsStructuralEqual(this object actual, object expected, string message  
= "")
```

Parameters

actual [object](#)

expected [object](#)

message [string](#)

IsTrue(bool)

Is(true)

```
public static void IsTrue(this bool value)
```

Parameters

value [bool](#)

Is<T>(IEnumerable<T>, IEnumerable<T>)

Assert.Equal(sequence value compare)

```
public static void Is<T>(this IEnumerable<T> actual, IEnumerable<T> expected)
```

Parameters

actual [IEnumerable](#)<T>

expected [IEnumerable](#)<T>

Type Parameters

T

Is<T>(IEnumerable<T>, IEnumerable<T>, IEqualityComparer<T>)

Assert.True(actual.SequenceEqual(expected, comparer))

```
public static void Is<T>(this IEnumerable<T> actual, IEnumerable<T> expected,
    IEqualityComparer<T> comparer)
```

Parameters

actual [IEnumerable](#)<T>

expected [IEnumerable](#)<T>

comparer [IEqualityComparer](#)<T>

Type Parameters

T

Is<T>(IEnumerable<T>, IEnumerable<T>, Func<T, T, bool>)

Assert.True(actual.SequenceEqual(expected, comparison))

```
public static void Is<T>(this IEnumerable<T> actual, IEnumerable<T> expected, Func<T, T,
    bool> equalityComparison)
```

Parameters

actual [IEnumerable](#)<T>

expected [IEnumerable](#)<T>

equalityComparison [Func](#)<T, T, [bool](#)>

Type Parameters

T

Is<T>(IEnumerable<T>, params T[])

Assert.Equal(sequence value compare)

```
public static void Is<T>(this IEnumerable<T> actual, params T[] expected)
```

Parameters

actual [IEnumerable](#)<T>

expected T[]

Type Parameters

T

Is<T>(T, Expression<Func<T, bool>>, string)

Assert.True(predicate(value))

```
public static void Is<T>(this T value, Expression<Func<T, bool>> predicate, string message  
= "")
```

Parameters

value T

predicate [Expression](#)<[Func](#)<T, [bool](#)>>

message [string](#)

Type Parameters

T

Is<T>(T, T)

Assert.Equal, if T is IEnumerable then compare value equality

```
public static void Is<T>(this T actual, T expected)
```

Parameters

actual T

expected T

Type Parameters

T

Is<T>(T, T, IEqualityComparer<T>)

Assert.Equal

```
public static void Is<T>(this T actual, T expected, IEqualityComparer<T> comparer)
```

Parameters

actual T

expected T

comparer [IEqualityComparer](#)<T>

Type Parameters

T