












Woodstocks.WoodstocksIMS.Data.CSV Namespace

The Woodstocks.WoodstocksIMS.Data namespace contains the data access layer components for the WoodstocksIMS.

Classes

	Class	Description
	CSVDataRecord	Represents a csv data record.
	CSVHeader	Represents the header record from a csv file.
	CSVParser	A CSV Parser that converts a csv string into a List(T) .
	CSVReader	A CSVReader that is used to read records from a csv file.
	CSVRecord	A base abstract class for CSV Records.
	CSVWriter	A CSVWriter that is used to write CSVRecords to a file.
	FieldDoesNotExistException	The exception that is raised when an attempt is made to access a named field that does not exist.
	MissingValueException	The exception that is raised when a value is missing from a CSVRecord .
	ToyExporterCSV	An exporter to export Wood Stocks Toy stock data to a csv data file.
	ToyImporterCSV	An importer to import toy data from a csv data file.

Enumerations

	Enumeration	Description
	CSVParser.TrimOption	An enumeration that defines the values of trimming options.

CSVDataRecord Class

Represents a csv data record.

Inheritance Hierarchy

[System.Object](#)

[System.Collections.ObjectModel.Collection\(String\)](#)

[Woodstocks.WoodstocksIMS.Data.CSV.CSVRecord](#)

Woodstocks.WoodstocksIMS.Data.CSV.CSVDataRecord

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

```
C#  
public class CSVDataRecord : CSVRecord
```





```
VB  
Public Class CSVDataRecord  
    Inherits CSVRecord
```

```
C++  
public ref class CSVDataRecord : public CSVRecord
```


```
F#  
type CSVDataRecord =  
    class  
        inherit CSVRecord  
    end
```


















The **CSVDataRecord** type exposes the following members.

Constructors





	Name	Description
	CSVDataRecord()	Initialises a CSVDataRecord .
	CSVDataRecord(List(String))	Initialises a CSVDataRecord .
	CSVDataRecord(CSVHeader)	Initialises a CSVDataRecord .
	CSVDataRecord(CSVHeader, List(String))	Initialises a CSVDataRecord .

Methods

	Name	Description
	Add	Adds an object to the end of the Collection(T) . (Inherited from

		Collection(String).
	Clear	Removes all elements from the Collection(T) . (Inherited from Collection(String).)
	ClearItems	Removes all elements from the Collection(T) . (Inherited from Collection(String).)
	Contains	Determines whether an element is in the Collection(T) . (Inherited from Collection(String).)
	CopyTo	Copies the entire Collection(T) to a compatible one-dimensional Array , starting at the specified index of the target array. (Inherited from Collection(String).)
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the Collection(T) . (Inherited from Collection(String).)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire Collection(T) . (Inherited from Collection(String).)
	Insert	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String).)
	InsertItem	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String).)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object.)
	Remove	Removes the first occurrence of a specific object from the Collection(T) . (Inherited from Collection(String).)
	RemoveAt	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String).)
	RemoveItem	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String).)
	SetItem	Replaces the element at the specified index. (Inherited from Collection(String).)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Properties

	Name	Description
	Count	Gets the number of elements actually contained in the Collection(T) . (Inherited from Collection(String).)
	Item(Int32)	Gets or sets the element at the specified index. (Inherited from Collection(String).)
	Item(String)	Gets a value for a named field of the record.
	Items	Gets a IList(T) wrapper around the Collection(T) . (Inherited from Collection(String).)





A Sandcastle Documented Class Library

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVDataRecord Constructor

Overload List

	Name	Description
	CSVDataRecord()	Initialises a CSVDataRecord .
	CSVDataRecord(List<String>)	Initialises a CSVDataRecord .
	CSVDataRecord(CSVHeader)	Initialises a CSVDataRecord .
	CSVDataRecord(CSVHeader, List<String>)	Initialises a CSVDataRecord .

See Also

[CSVDataRecord Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVDataRecord Constructor

Initialises a [CSVDataRecord](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVDataRecord( )
```

VB

```
Public Sub New
```

C++

```
public:  
CSVDataRecord( )
```

F#

```
new : unit -> CSVDataRecord
```

See Also

[CSVDataRecord Class](#)

[CSVDataRecord Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVDataRecord Constructor (List(String))

Initialises a [CSVDataRecord](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVDataRecord(  
    List<string> values  
)
```

VB

```
Public Sub New (  
    values As List(Of String)  
)
```

C++

```
public:  
CSVDataRecord(  
    List<String^>^ values  
)
```

F#

```
new :  
    values : List<string> -> CSVDataRecord
```

Parameters

values

Type: [System.Collections.Generic.List\(String\)](#)

The values of the record.

See Also

[CSVDataRecord Class](#)

[CSVDataRecord Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVDataRecord Constructor (CSVHeader)

Initialises a [CSVDataRecord](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVDataRecord(  
    CSVHeader header  
)
```

VB

```
Public Sub New (  
    header As CSVHeader  
)
```

C++

```
public:  
CSVDataRecord(  
    CSVHeader^ header  
)
```

F#

```
new :  
    header : CSVHeader -> CSVDataRecord
```

Parameters

header

Type: [Woodstocks.WoodstocksIMS.Data.CSV.CSVHeader](#)

A header, containing the names of fields, for the record.

See Also

[CSVDataRecord Class](#)

[CSVDataRecord Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVDataRecord Constructor (CSVHeader, List(String))

Initialises a [CSVDataRecord](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVDataRecord(  
    CSVHeader header,  
    List<string> values  
)
```

VB

```
Public Sub New (  
    header As CSVHeader,  
    values As List(Of String)  
)
```

C++

```
public:  
CSVDataRecord(  
    CSVHeader^ header,  
    List<String^>^ values  
)
```

F#

```
new :  
    header : CSVHeader *  
    values : List<string> -> CSVDataRecord
```

Parameters

header

Type: [Woodstocks.WoodstocksIMS.Data.CSV.CSVHeader](#)

A [CSVHeader](#) that defines the fields for the record.

values

Type: [System.Collections.Generic.List\(String\)](#)

The values of the record.

See Also

[CSVDataRecord Class](#)



















[CSVDataRecord Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVDataRecord.CSVDataRecord Methods

The [CSVDataRecord](#) type exposes the following members.

Methods

	Name	Description
	Add	Adds an object to the end of the Collection(T) . (Inherited from Collection(String) .)
	Clear	Removes all elements from the Collection(T) . (Inherited from Collection(String) .)
	ClearItems	Removes all elements from the Collection(T) . (Inherited from Collection(String) .)
	Contains	Determines whether an element is in the Collection(T) . (Inherited from Collection(String) .)
	CopyTo	Copies the entire Collection(T) to a compatible one-dimensional Array , starting at the specified index of the target array. (Inherited from Collection(String) .)
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetEnumerator	Returns an enumerator that iterates through the Collection(T) . (Inherited from Collection(String) .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire Collection(T) . (Inherited from Collection(String) .)
	Insert	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String) .)
	InsertItem	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String) .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Remove	Removes the first occurrence of a specific object from the Collection(T) . (Inherited from Collection(String) .)
	RemoveAt	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String) .)
	RemoveItem	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String) .)
	SetItem	Replaces the element at the specified index. (Inherited from Collection(String) .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

See Also

[CSVDataRecord Class](#)





A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVDataRecord.CSVDataRecord Properties

The [CSVDataRecord](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements actually contained in the Collection(T) . (Inherited from Collection(String) .)
	Item(Int32)	Gets or sets the element at the specified index. (Inherited from Collection(String) .)
	Item(String)	Gets a value for a named field of the record.
	Items	Gets a List(T) wrapper around the Collection(T) . (Inherited from Collection(String) .)



See Also

[CSVDataRecord Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVDataRecord.Item Property

Overload List

	Name	Description
	Item(Int32)	Gets or sets the element at the specified index. (Inherited from Collection(String) .)
	Item(String)	Gets a value for a named field of the record.

See Also

[CSVDataRecord Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVDataRecord.Item Property (String)

Gets a value for a named field of the record.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public string this[
    string fieldName
] { get; set; }
```

VB

```
Public Default Property Item (
    fieldName As String
) As String
    Get
    Set
```

C++

```
public:
property String^ default[String^ fieldName] {
    String^ get (String^ fieldName);
    void set (String^ fieldName, String^ value);
}
```

F#

```
member Item : string with get, set
```

Parameters

fieldName

Type: [System.String](#)

The name of the field of the record for which the value should be returned.

Return Value

Type: [String](#)

The value of the named field.

Remarks

A **FieldNameArgumentIsNullException** is thrown if a null value or an empty string is passed as the argument of the *fieldName* parameter. A [FieldDoesNotExistException](#) is thrown if an attempt is made to read the value of a field and the field name for the record does not exist.

See Also

[CSVDataRecord Class](#)

A Sandcastle Documented Class Library

[Item Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVHeader Class

Represents the header record from a csv file.

Inheritance Hierarchy

[System.Object](#)

[System.Collections.ObjectModel.Collection\(String\)](#)

[Woodstocks.WoodstocksIMS.Data.CSV.CSVRecord](#)

Woodstocks.WoodstocksIMS.Data.CSV.CSVHeader

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

```
C#  
public class CSVHeader : CSVRecord
```




```
VB  
Public Class CSVHeader  
    Inherits CSVRecord
```

```
C++  
public ref class CSVHeader : public CSVRecord
```


```
F#  
type CSVHeader =  
    class  
        inherit CSVRecord  
    end
```



















The **CSVHeader** type exposes the following members.

Constructors





	Name	Description
	CSVHeader()	Initialises a CSVHeader .
	CSVHeader(List(String))	Initialises a CSVHeader .
	CSVHeader(String[])	Initialises a CSVHeader .

Methods

	Name	Description
	Add	Adds an object to the end of the Collection(T) . (Inherited from Collection(String) .)

	Clear	Removes all elements from the Collection(T) . (Inherited from Collection(String) .)
	ClearItems	Removes all elements from the Collection(T) . (Inherited from Collection(String) .)
	Contains	Determines whether an element is in the Collection(T) . (Inherited from Collection(String) .)
	CopyTo	Copies the entire Collection(T) to a compatible one-dimensional Array , starting at the specified index of the target array. (Inherited from Collection(String) .)
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetEnumerator	Returns an enumerator that iterates through the Collection(T) . (Inherited from Collection(String) .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire Collection(T) . (Inherited from Collection(String) .)
	Insert	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String) .)
	InsertItem	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String) .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Remove	Removes the first occurrence of a specific object from the Collection(T) . (Inherited from Collection(String) .)
	RemoveAt	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String) .)
	RemoveItem	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String) .)
	SetItem	Replaces the element at the specified index. (Inherited from Collection(String) .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Properties

	Name	Description
	Count	Gets the number of elements actually contained in the Collection(T) . (Inherited from Collection(String) .)
	Item(Int32)	Gets or sets the element at the specified index. (Inherited from Collection(String) .)
	Item(String)	Gets the value of a field.
	Items	Gets a IList(T) wrapper around the Collection(T) . (Inherited from Collection(String) .)




A Sandcastle Documented Class Library

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVHeader Constructor

Overload List

	Name	Description
	CSVHeader()	Initialises a CSVHeader .
	CSVHeader(List(String))	Initialises a CSVHeader .
	CSVHeader(String[])	Initialises a CSVHeader .

See Also

[CSVHeader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVHeader Constructor

Initialises a [CSVHeader](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVHeader ( )
```

VB

```
Public Sub New
```

C++

```
public:  
CSVHeader ( )
```

F#

```
new : unit -> CSVHeader
```

See Also

[CSVHeader Class](#)

[CSVHeader Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVHeader Constructor (List(String))

Initialises a [CSVHeader](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVHeader(  
    List<string> fieldNames  
)
```

VB

```
Public Sub New (  
    fieldNames As List(Of String)  
)
```

C++

```
public:  
CSVHeader(  
    List<String^>^ fieldNames  
)
```

F#

```
new :  
    fieldNames : List<string> -> CSVHeader
```

Parameters

fieldNames

Type: [System.Collections.Generic.List\(String\)](#)

The values of the [CSVHeader](#). The values are the names of the fields for data within the file.

See Also

[CSVHeader Class](#)

[CSVHeader Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVHeader Constructor (String[])

Initialises a [CSVHeader](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVHeader(  
    params string[] fieldNames  
)
```

VB

```
Public Sub New (  
    ParamArray fieldNames As String()  
)
```

C++

```
public:  
CSVHeader(  
    ... array<String^>^ fieldNames  
)
```

F#

```
new :  
    fieldNames : string[] -> CSVHeader
```

Parameters

fieldNames

Type: [System.String\[\]](#)

The values of the [CSVHeader](#). The values are the names of the fields for data within the file.

See Also

[CSVHeader Class](#)
















[CSVHeader Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVHeader.CSVHeader Methods

The [CSVHeader](#) type exposes the following members.

Methods

	Name	Description
	Add	Adds an object to the end of the Collection(T) . (Inherited from Collection(String) .)
	Clear	Removes all elements from the Collection(T) . (Inherited from Collection(String) .)
	ClearItems	Removes all elements from the Collection(T) . (Inherited from Collection(String) .)
	Contains	Determines whether an element is in the Collection(T) . (Inherited from Collection(String) .)
	CopyTo	Copies the entire Collection(T) to a compatible one-dimensional Array , starting at the specified index of the target array. (Inherited from Collection(String) .)
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetEnumerator	Returns an enumerator that iterates through the Collection(T) . (Inherited from Collection(String) .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire Collection(T) . (Inherited from Collection(String) .)
	Insert	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String) .)
	InsertItem	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String) .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Remove	Removes the first occurrence of a specific object from the Collection(T) . (Inherited from Collection(String) .)
	RemoveAt	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String) .)
	RemoveItem	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String) .)
	SetItem	Replaces the element at the specified index. (Inherited from Collection(String) .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

See Also

[CSVHeader Class](#)





A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVHeader.CSVHeader Properties

The [CSVHeader](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements actually contained in the Collection(T) . (Inherited from Collection(String) .)
	Item(Int32)	Gets or sets the element at the specified index. (Inherited from Collection(String) .)
	Item(String)	Gets the value of a field.
	Items	Gets a IList(T) wrapper around the Collection(T) . (Inherited from Collection(String) .)



See Also

[CSVHeader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVHeader.Item Property

Overload List

	Name	Description
	Item(Int32)	Gets or sets the element at the specified index. (Inherited from Collection(String) .)
	Item(String)	Gets the value of a field.

See Also

[CSVHeader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVHeader.Item Property (String)

Gets the value of a field.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public int this[
    string fieldName
] { get; }
```

VB

```
Public ReadOnly Default Property Item (
    fieldName As String
) As Integer
    Get
```

C++

```
public:
property int default[String^ fieldName] {
    int get (String^ fieldName);
}
```

F#

```
member Item : int with get
```

Parameters

fieldName

Type: [System.String](#)

The field name for which the value is to be retrieved.

Return Value

Type: [Int32](#)

The value of the field.

See Also

[CSVHeader Class](#)

[Item Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVParser Class

A CSV Parser that converts a csv string into a [List\(T\)](#).

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Data.CSV.CSVParser

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

```
C#  
public class CSVParser
```


```
VB  
Public Class CSVParser
```

```
C++  
public ref class CSVParser
```







```
F#  
type CSVParser = class end
```







The **CSVParser** type exposes the following members.

Constructors

	Name	Description
	CSVParser	Initializes a new instance of the CSVParser class

Methods

	Name	Description
	Detokenize	Detokenises a set of decomposed csv values.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetTokenCount	Gets the number of tokenised values within List{T} of decomposed csv values.
	GetType	Gets the Type of the current instance. (Inherited from Object .)

	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Parse	Parses a csv string into a List(T) of strings that contain the component values of the csv string.
	StripWhitespace	Removes the white space of values contained in values according to a specified trimming option.
	Tokenize	Replaces an escaped csv value with a csv token.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TrimEscapeCharacter	Removes the escape character from a set of values from a csv value string.

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVParser Constructor

Initializes a new instance of the [CSVParser](#) class

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVParser()
```

VB

```
Public Sub New
```

C++

```
public:  
CSVParser()
```

F#

```
new : unit -> CSVParser
```

See Also













[CSVParser Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVParser.CSVParser Methods

The [CSVParser](#) type exposes the following members.

Methods

	Name	Description
	Detokenize	Detokenises a set of decomposed csv values.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetTokenCount	Gets the number of tokenised values within List{T} of decomposed csv values.
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Parse	Parses a csv string into a List(T) of strings that contain the component values of the csv string.
	StripWhitespace	Removes the white space of values contained in values according to a specified trimming option.
	Tokenize	Replaces an escaped csv value with a csv token.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TrimEscapeCharacter	Removes the escape character from a set of values from a csv value string.

See Also

[CSVParser Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVParser.Detokenize Method

Detokenises a set of decomposed csv values.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual List<string> Detokenize(  
    List<string> tokenizedValues,  
    Queue<string> replacements  
)
```

VB

```
Protected Overridable Function Detokenize (  
    tokenizedValues As List(Of String),  
    replacements As Queue(Of String)  
) As List(Of String)
```

C++

```
protected:  
virtual List<String^>^ Detokenize(  
    List<String^>^ tokenizedValues,  
    Queue<String^>^ replacements  
)
```

F#

```
abstract Detokenize :  
    tokenizedValues : List<string> *  
    replacements : Queue<string> -> List<string>  
override Detokenize :  
    tokenizedValues : List<string> *  
    replacements : Queue<string> -> List<string>
```

Parameters

tokenizedValues

Type: [System.Collections.Generic.List\(String\)](#)

The decomposed List{T} of csv values.

replacements

Type: [System.Collections.Generic.Queue\(String\)](#)

The values to replace each csv token.

Return Value

Type: [List\(String\)](#)

The detokenized set of csv values.

Remarks

This method replaces each csv tokenized value with the original value before tokenization.

See Also

[CSVParser Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVParser.GetTokenCount Method

Gets the number of tokenised values within List{T} of decomposed csv values.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual int GetTokenCount(  
    List<string> tokenizedValues  
)
```

VB

```
Protected Overridable Function GetTokenCount (  
    tokenizedValues As List(Of String)  
) As Integer
```

C++

```
protected:  
virtual int GetTokenCount(  
    List<String^>^ tokenizedValues  
)
```

F#

```
abstract GetTokenCount :  
    tokenizedValues : List<string> -> int  
override GetTokenCount :  
    tokenizedValues : List<string> -> int
```

Parameters

tokenizedValues

Type: [System.Collections.Generic.List\(String\)](#)

The decomposed set of csv values.

Return Value

Type: [Int32](#)

The number of tokenised values contained within the List{T}.

See Also

[CSVParser Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVParser.Parse Method

Parses a csv string into a [List\(T\)](#) of strings that contain the component values of the csv string.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public List<string> Parse(  
    string input,  
    bool removeEscapeCharacter  
)
```

VB

```
Public Function Parse (  
    input As String,  
    removeEscapeCharacter As Boolean  
) As List(Of String)
```

C++

```
public:  
List<String^>^ Parse(  
    String^ input,  
    bool removeEscapeCharacter  
)
```

F#

```
member Parse :  
    input : string *  
    removeEscapeCharacter : bool -> List<string>
```

Parameters

input

Type: [System.String](#)

The csv string that should be parsed into a string List.

removeEscapeCharacter

Type: [System.Boolean](#)

Indicates whether escape characters (i.e double quotes should be removed from escaped values within the csv input string.

Return Value

Type: [List\(String\)](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Data.CSV.CSVParser.Parse(System.String,System.Boolean)"]

Remarks

This method parses a csv string into a List that contains the individual values of a the string. To ensure correct parsing of a csv value string, including those with escaped values, the string is tokenized prior to splitting it into its component values. Once the string is tokenized it is split at the character specified by csvSeparator. Once split into its component values, each tokenized component is replaced restoring the component value to its pre-tokenized value.

See Also

[CSVParser Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVParser.StripWhitespace Method

Removes the white space of values contained in values according to a specified trimming option.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void StripWhitespace(  
    List<string> values,  
    CSVParser.TrimOption trimOption  
)
```

VB

```
Protected Overridable Sub StripWhitespace (  
    values As List(Of String),  
    trimOption As CSVParser.TrimOption  
)
```

C++

```
protected:  
virtual void StripWhitespace(  
    List<String^>^ values,  
    CSVParser.TrimOption trimOption  
)
```

F#

```
abstract StripWhitespace :  
    values : List<string> *  
    trimOption : CSVParser.TrimOption -> unit  
override StripWhitespace :  
    values : List<string> *  
    trimOption : CSVParser.TrimOption -> unit
```

Parameters

values

Type: [System.Collections.Generic.List\(String\)](#)

The values that are to have white space removed.

trimOption

Type: [Woodstocks.WoodstocksIMS.Data.CSV.CSVParser.TrimOption](#)

The trimming option that specifies how trimming is to occur. [CSVParser.TrimOption](#)

See Also

[CSVParser Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVParser.Tokenize Method

Replaces an escaped csv value with a csv token.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual string Tokenize(  
    string csv,  
    out Queue<string> replaced  
)
```

VB

```
Protected Overridable Function Tokenize (  
    csv As String,  
    <OutAttribute> ByRef replaced As Queue(Of String)  
) As String
```

C++

```
protected:  
virtual String^ Tokenize(  
    String^ csv,  
    [OutAttribute] Queue<String^>^% replaced  
)
```

F#

```
abstract Tokenize :  
    csv : string *  
    replaced : Queue<string> byref -> string  
override Tokenize :  
    csv : string *  
    replaced : Queue<string> byref -> string
```

Parameters

csv

Type: [System.String](#)

replaced

Type: [System.Collections.Generic.Queue](#)([String](#))

Return Value

Type: [String](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Data.CSV.CSVParser.Tokenize(System.String,System.Collections.Generic.Queue{System.String}@)"]

Remarks

This method is used to replace any escaped csv values that are delimited with double quotes with a csv token.

See Also

[CSVParser Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVParser.TrimEscapeCharacter Method

Removes the escape character from a set of values from a csv value string.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual Queue<string> TrimEscapeCharacter(  
    Queue<string> escapedValues  
)
```

VB

```
Protected Overridable Function TrimEscapeCharacter (  
    escapedValues As Queue(Of String)  
) As Queue(Of String)
```

C++

```
protected:  
virtual Queue<String^>^ TrimEscapeCharacter(  
    Queue<String^>^ escapedValues  
)
```

F#

```
abstract TrimEscapeCharacter :  
    escapedValues : Queue<string> -> Queue<string>  
override TrimEscapeCharacter :  
    escapedValues : Queue<string> -> Queue<string>
```

Parameters

escapedValues

Type: [System.Collections.Generic.Queue\(String\)](#)

The individual values in the csv value string.

Return Value

Type: [Queue\(String\)](#)

The value set with any escape caharacter removed.

See Also

[CSVParser Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVParser.TrimOption Enumeration

An enumeration that defines the values of trimming options.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public enum TrimOption
```

VB

```
Public Enumeration TrimOption
```

C++

```
public enum class TrimOption
```

F#

```
type TrimOption
```

Members

Member name	Value	Description
None	0	Used to indicate no trimming should occur.
LeadingWhitespace	1	Used to indicate that only leading white space should be trimmed.
TrailingWhitespace	2	Used to indicate that only trailing white space should be trimmed.
LeadingAndTrailingWhitespace	3	Used to indicate that both leading and trailing white space should be trimmed.

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader Class

A CSVReader that is used to read records from a csv file.

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Data.CSV.CSVReader

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class CSVReader : IDisposable
```

VB

```
Public Class CSVReader
    Implements IDisposable
```

C++


```
public ref class CSVReader : IDisposable
```

F#






```
type CSVReader =
    class
        interface IDisposable
    end
```












The **CSVReader** type exposes the following members.

Constructors






	Name	Description
	CSVReader	Initialises a CSVReader for reading from a file containing csv records.

Methods

	Name	Description
	Close	Closes the file that the CSVReader has open.
	Dispose()	Implements IDisposable.Dispose() for the CSVReader .
	Dispose(Boolean)	Disposes of the resources of the CSVReader .
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations

		before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetRecordCount	Gets the number of csv records that are contained in the file.
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IsRecordIncomplete	Tests if the input string is a complete csv record.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Open	Opens the file that the CSVReader should read from.
	ReadDataRecord	Reads a data record from a csv file and returns the data as a CSVDataRecord .
	ReadHeader	Reads the header record from a file containing csv records.
	ReadRecord	Reads a record from a file and returns the record as a string.
	SetNextRecord	Sets the value of the nextRecord field.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Properties

	Name	Description
	AllowEmptyStringValues	Gets or Sets whether the CSVReader should allow empty string values in a CSVRecord (either CSVHeader or CSVDataRecord
	FilePath	Gets the path of the file that the CSVReader should read data from.
	IsOpen	Gets whether the file is open or not.
	NextRecord	Gets the zero-based index for the next record to be read. A value of -1 is returned if the current record is the last readable record.
	Records	Returns the number of records available to be read by the CSVReader .

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader Constructor

Initialises a CSVReader for reading from a file containing csv records.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVReader(  
    string filepath,  
    FileMode mode,  
    bool allowEmptyStringValues  
)
```

VB

```
Public Sub New (  
    filepath As String,  
    mode As FileMode,  
    allowEmptyStringValues As Boolean  
)
```

C++

```
public:  
CSVReader(  
    String^ filepath,  
    FileMode mode,  
    bool allowEmptyStringValues  
)
```

F#

```
new :  
    filepath : string *  
    mode : FileMode *  
    allowEmptyStringValues : bool -> CSVReader
```

Parameters

filepath

Type: [System.String](#)

The file path of the file that the CSVReader should read records from.

mode

Type: [System.IO.FileMode](#)

The file mode in which the CSVReader should open the file.

allowEmptyStringValues

Type: [System.Boolean](#)

Indicates whether the reader when reading values should detect a missing value, denoted by an empty string. If set to true then upon detecting a missing value the reader will raise a [MissingValueException](#)

See Also

















[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.CSVReader Methods

The [CSVReader](#) type exposes the following members.

Methods

	Name	Description
	Close	Closes the file that the CSVReader has open.
	Dispose()	Implements IDisposable.Dispose() for the CSVReader .
	Dispose(Boolean)	Disposes of the resources of the CSVReader .
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetRecordCount	Gets the number of csv records that are contained in the file.
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IsRecordIncomplete	Tests if the input string is a complete csv record.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Open	Opens the file that the CSVReader should read from.
	ReadDataRecord	Reads a data record from a csv file and returns the data as a CSVDataRecord .
	ReadHeader	Reads the header record from a file containing csv records.
	ReadRecord	Reads a record from a file and returns the record as a string.
	SetNextRecord	Sets the value of the nextRecord field.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.Close Method

Closes the file that the CSVReader has open.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Close()
```

VB

```
Public Sub Close
```

C++

```
public:  
void Close()
```

F#

```
member Close : unit -> unit
```



See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.Dispose Method

Overload List

	Name	Description
	Dispose()	Implements IDisposable.Dispose() for the CSVReader .
	Dispose(Boolean)	Disposes of the resources of the CSVReader .

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.Dispose Method

Implements [IDisposable.Dispose\(\)](#) for the [CSVReader](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Dispose()
```

VB

```
Public Sub Dispose
```

C++

```
public:  
virtual void Dispose() sealed
```

F#

```
abstract Dispose : unit -> unit  
override Dispose : unit -> unit
```

Implements

[IDisposable.Dispose\(\)](#)

See Also

[CSVReader Class](#)

[Dispose Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.Dispose Method (Boolean)

Disposes of the resources of the [CSVReader](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Protected Overridable Sub Dispose (  
    disposing As Boolean  
)
```

C++

```
protected:  
virtual void Dispose(  
    bool disposing  
)
```

F#

```
abstract Dispose :  
    disposing : bool -> unit  
override Dispose :  
    disposing : bool -> unit
```

Parameters

disposing

Type: [System.Boolean](#)

See Also

[CSVReader Class](#)

[Dispose Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.GetRecordCount Method

Gets the number of csv records that are contained in the file.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual int GetRecordCount()
```

VB

```
Protected Overridable Function GetRecordCount As Integer
```

C++

```
protected:  
virtual int GetRecordCount()
```

F#

```
abstract GetRecordCount : unit -> int  
override GetRecordCount : unit -> int
```

Return Value

Type: [Int32](#)

The number of records, including any header record, that is contained in the file.

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.IsRecordIncomplete Method

Tests if the input string is a complete csv record.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual bool IsRecordIncomplete(  
    string input  
)
```

VB

```
Protected Overridable Function IsRecordIncomplete (  
    input As String  
) As Boolean
```

C++

```
protected:  
virtual bool IsRecordIncomplete(  
    String^ input  
)
```

F#

```
abstract IsRecordIncomplete :  
    input : string -> bool  
override IsRecordIncomplete :  
    input : string -> bool
```

Parameters

input

Type: [System.String](#)

The string containing the csv that is to be tested.

Return Value

Type: [Boolean](#)

True if the record is complete, otherwise returns false.

Remarks

A csv record is regarded as incomplete if it ends with a partially escaped csv value (i.e. The value commences with a double quote (") but no matching double quote occurs before the end of the string).

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.Open Method

Opens the file that the CSVReader should read from.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Open( )
```

VB

```
Public Sub Open
```

C++

```
public:  
void Open( )
```

F#

```
member Open : unit -> unit
```

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.ReadDataRecord Method

Reads a data record from a csv file and returns the data as a [CSVDataRecord](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVDataRecord ReadDataRecord(  
    int recordIndex,  
    bool removeEscapeCharacter  
)
```

VB

```
Public Function ReadDataRecord (  
    recordIndex As Integer,  
    removeEscapeCharacter As Boolean  
) As CSVDataRecord
```

C++

```
public:  
CSVDataRecord^ ReadDataRecord(  
    int recordIndex,  
    bool removeEscapeCharacter  
)
```

F#

```
member ReadDataRecord :  
    recordIndex : int *  
    removeEscapeCharacter : bool -> CSVDataRecord
```

Parameters

recordIndex

Type: [System.Int32](#)

The index of the record to be read from the file.

removeEscapeCharacter

Type: [System.Boolean](#)

Indicates whether escape characters (i.e double quotes should be removed from escaped values within the csv input string.

Return Value

Type: [CSVDataRecord](#)

A [CSVDataRecord](#) that contains the record data.

A Sandcastle Documented Class Library

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.ReadHeader Method

Reads the header record from a file containing csv records.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVHeader ReadHeader(  
    bool removeEscapeCharacter  
)
```

VB

```
Public Function ReadHeader (  
    removeEscapeCharacter As Boolean  
) As CSVHeader
```

C++

```
public:  
CSVHeader^ ReadHeader(  
    bool removeEscapeCharacter  
)
```

F#

```
member ReadHeader :  
    removeEscapeCharacter : bool -> CSVHeader
```

Parameters

removeEscapeCharacter

Type: [System.Boolean](#)

[Missing <param name="removeEscapeCharacter"/> documentation for "M:Woodstocks.WoodstocksIMS.Data.CSV.CSVReader.ReadHeader(System.Boolean)"]

Return Value

Type: [CSVHeader](#)

A [CSVHeader](#) that contains the values of the header or null if the header record could not be read.

Remarks

The method reads the first record of the file and therefore assumes that the optional header is the first record in the file containing csv records.

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.ReadRecord Method

Reads a record from a file and returns the record as a string.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual string ReadRecord(  
    int recordIndex  
)
```

VB

```
Protected Overridable Function ReadRecord (  
    recordIndex As Integer  
) As String
```

C++

```
protected:  
virtual String^ ReadRecord(  
    int recordIndex  
)
```

F#

```
abstract ReadRecord :  
    recordIndex : int -> string  
override ReadRecord :  
    recordIndex : int -> string
```

Parameters

recordIndex

Type: [System.Int32](#)

A zero-based index value that indicates the record that should be read from the file.

Return Value

Type: [String](#)

The record as a string.

Remarks

The method reads lines from the file, ignoring empty lines, until the record wanted from the file is read. When the method reads the record that is wanted the method returns the record as a string.

The method ensures that a line read from the file constitutes a complete record by calling the `IsRecordComplete` method which checks that the line constitutes a complete record for a csv record.

See Also

[CSVReader Class](#)

A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.SetNextRecord Method

Sets the value of the nextRecord field.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected void SetNextRecord(  
    int recordIndex  
)
```

VB

```
Protected Sub SetNextRecord (  
    recordIndex As Integer  
)
```

C++

```
protected:  
void SetNextRecord(  
    int recordIndex  
)
```

F#

```
member SetNextRecord :  
    recordIndex : int -> unit
```

Parameters

recordIndex

Type: [System.Int32](#)

The zero-based index of the next record that the reader should read from the file.

See Also






[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.CSVReader Properties

The [CSVReader](#) type exposes the following members.

Properties

	Name	Description
	AllowEmptyStringValues	Gets or Sets whether the CSVReader should allow empty string values in a CSVRecord (either CSVHeader or CSVDataRecord)
	FilePath	Gets the path of the file that the CSVReader should read data from.
	IsOpen	Gets whether the file is open or not.
	NextRecord	Gets the zero-based index for the next record to be read. A value of -1 is returned if the current record is the last readable record.
	Records	Returns the number of records available to be read by the CSVReader .

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.AllowEmptyStringValues Property

Gets or Sets whether the [CSVReader](#) should allow empty string values in a CSVRecord (either [CSVHeader](#) or [CSVDataRecord](#))

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Public Property AllowEmptyStringValues As Boolean
    Get
    Set
```

C++

```
public:
property bool AllowEmptyStringValues {
    bool get ();
    void set (bool value);
}
```

F#

```
member AllowEmptyStringValues : bool with get, set
```

Property Value

Type: [Boolean](#)

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.FilePath Property

Gets the path of the file that the CSVReader should read data from.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Public Property FilePath As String  
    Get  
    Set
```

C++

```
public:  
property String^ FilePath {  
    String^ get ();  
    void set (String^ value);  
}
```

F#

```
member FilePath : string with get, set
```

Property Value

Type: [String](#)

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.IsOpen Property

Gets whether the file is open or not.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool IsOpen { get; }
```

VB

```
Public ReadOnly Property IsOpen As Boolean  
    Get
```

C++

```
public:  
property bool IsOpen {  
    bool get ();  
}
```

F#

```
member IsOpen : bool with get
```

Property Value

Type: [Boolean](#)

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.NextRecord Property

Gets the zero-based index for the next record to be read. A value of -1 is returned if the current record is the last readable record.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public int NextRecord { get; }
```

VB

```
Public ReadOnly Property NextRecord As Integer  
    Get
```

C++

```
public:  
property int NextRecord {  
    int get ();  
}
```

F#

```
member NextRecord : int with get
```

Property Value

Type: [Int32](#)

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVReader.Records Property

Returns the number of records available to be read by the [CSVReader](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public int Records { get; }
```

VB

```
Public ReadOnly Property Records As Integer  
    Get
```

C++

```
public:  
property int Records {  
    int get ();  
}
```

F#

```
member Records : int with get
```

Property Value

Type: [Int32](#)

See Also

[CSVReader Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVRecord Class

A base abstract class for CSV Records.

Inheritance Hierarchy

[System.Object](#)

[System.Collections.ObjectModel.Collection\(String\)](#)

Woodstocks.WoodstocksIMS.Data.CSV.CSVRecord

[Woodstocks.WoodstocksIMS.Data.CSV.CSVDataRecord](#)

[Woodstocks.WoodstocksIMS.Data.CSV.CSVHeader](#)

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public abstract class CSVRecord : Collection<string>
```

VB

```
Public MustInherit Class CSVRecord  
    Inherits Collection(Of String)
```

C++




```
public ref class CSVRecord abstract : public Collection<String^>
```

F#


```
[<AbstractClassAttribute>]  
type CSVRecord =  
    class  
        inherit Collection<string>  
    end
```


















The **CSVRecord** type exposes the following members.

Constructors




	Name	Description
	CSVRecord()	Initialises a new instance of the CSVRecord .
	CSVRecord(List(String))	Initialises a new instance of the CSVRecord .
	CSVRecord(String[])	Initialises a new instance of a CSVRecord .

Methods

	Name	Description
	Add	Adds an object to the end of the Collection(T) . (Inherited from

		Collection(String).
	Clear	Removes all elements from the Collection(T) . (Inherited from Collection(String).)
	ClearItems	Removes all elements from the Collection(T) . (Inherited from Collection(String).)
	Contains	Determines whether an element is in the Collection(T) . (Inherited from Collection(String).)
	CopyTo	Copies the entire Collection(T) to a compatible one-dimensional Array , starting at the specified index of the target array. (Inherited from Collection(String).)
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object.)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object.)
	GetEnumerator	Returns an enumerator that iterates through the Collection(T) . (Inherited from Collection(String).)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire Collection(T) . (Inherited from Collection(String).)
	Insert	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String).)
	InsertItem	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String).)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object.)
	Remove	Removes the first occurrence of a specific object from the Collection(T) . (Inherited from Collection(String).)
	RemoveAt	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String).)
	RemoveItem	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String).)
	SetItem	Replaces the element at the specified index. (Inherited from Collection(String).)
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Properties

	Name	Description
	Count	Gets the number of elements actually contained in the Collection(T) . (Inherited from Collection(String).)
	Item	Gets or sets the element at the specified index. (Inherited from Collection(String).)
	Items	Gets a IList(T) wrapper around the Collection(T) . (Inherited from Collection(String).)




A Sandcastle Documented Class Library

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVRecord Constructor

Overload List

	Name	Description
	CSVRecord()	Initialises a new instance of the CSVRecord .
	CSVRecord(List(String))	Initialises a new instance of the CSVRecord .
	CSVRecord(String[])	Initialises a new instance of a CSVRecord .

See Also

[CSVRecord Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVRecord Constructor

Initialises a new instance of the [CSVRecord](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVRecord( )
```

VB

```
Public Sub New
```

C++

```
public:  
CSVRecord( )
```

F#

```
new : unit -> CSVRecord
```

See Also

[CSVRecord Class](#)

[CSVRecord Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVRecord Constructor (List(String))

Initialises a new instance of the [CSVRecord](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVRecord(  
    List<string> values  
)
```

VB

```
Public Sub New (  
    values As List(Of String)  
)
```

C++

```
public:  
CSVRecord(  
    List<String^>^ values  
)
```

F#

```
new :  
    values : List<string> -> CSVRecord
```

Parameters

values

Type: [System.Collections.Generic.List\(String\)](#)

The initial values of the record.

See Also

[CSVRecord Class](#)

[CSVRecord Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVRecord Constructor (String[])

Initialises a new instance of a [CSVRecord](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVRecord(  
    params string[] values  
)
```

VB

```
Public Sub New (  
    ParamArray values As String()  
)
```

C++

```
public:  
CSVRecord(  
    ... array<String^>^ values  
)
```

F#

```
new :  
    values : string[] -> CSVRecord
```

Parameters

values

Type: [System.String\[\]](#)

The initial values of the record.

See Also

[CSVRecord Class](#)

















[CSVRecord Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVRecord.CSVRecord Methods

The [CSVRecord](#) type exposes the following members.

Methods

	Name	Description
	Add	Adds an object to the end of the Collection(T) . (Inherited from Collection(String) .)
	Clear	Removes all elements from the Collection(T) . (Inherited from Collection(String) .)
	ClearItems	Removes all elements from the Collection(T) . (Inherited from Collection(String) .)
	Contains	Determines whether an element is in the Collection(T) . (Inherited from Collection(String) .)
	CopyTo	Copies the entire Collection(T) to a compatible one-dimensional Array , starting at the specified index of the target array. (Inherited from Collection(String) .)
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetEnumerator	Returns an enumerator that iterates through the Collection(T) . (Inherited from Collection(String) .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IndexOf	Searches for the specified object and returns the zero-based index of the first occurrence within the entire Collection(T) . (Inherited from Collection(String) .)
	Insert	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String) .)
	InsertItem	Inserts an element into the Collection(T) at the specified index. (Inherited from Collection(String) .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Remove	Removes the first occurrence of a specific object from the Collection(T) . (Inherited from Collection(String) .)
	RemoveAt	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String) .)
	RemoveItem	Removes the element at the specified index of the Collection(T) . (Inherited from Collection(String) .)
	SetItem	Replaces the element at the specified index. (Inherited from Collection(String) .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

See Also

[CSVRecord Class](#)




A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVRecord.CSVRecord Properties

The [CSVRecord](#) type exposes the following members.

Properties

	Name	Description
	Count	Gets the number of elements actually contained in the Collection(T) . (Inherited from Collection(String) .)
	Item	Gets or sets the element at the specified index. (Inherited from Collection(String) .)
	Items	Gets a IList(T) wrapper around the Collection(T) . (Inherited from Collection(String) .)

See Also

[CSVRecord Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter Class

A CSVWriter that is used to write CSVRecords to a file.

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Data.CSV.CSVWriter

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class CSVWriter : IDisposable
```

VB

```
Public Class CSVWriter
    Implements IDisposable
```

C++


```
public ref class CSVWriter : IDisposable
```

F#






```
type CSVWriter =
    class
        interface IDisposable
    end
```








The **CSVWriter** type exposes the following members.

Constructors



	Name	Description
	CSVWriter	Initialises a CSVWriter .

Methods

	Name	Description
	Close	Closes the file the CSVWriter, and its associated file.
	Dispose()	Implements the IDisposable.Dispose() method.
	Dispose(Boolean)	Disposes of the resources that are utilised by the CSVWriter .
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations

		before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Open	Opens the file to which data is to be written.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	WriteCSVRecord	Writes a CSVRecord to the file associated with this CSVWriter .
	WriteToFile	Writes the output string to the associated file.

Properties

	Name	Description
	Filepath	Gets the filepath for the file that the CSVWriter should write data to.
	IsOpen	Gets the status of the file. Returns true if the file is open.

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter Constructor

Initialises a [CSVWriter](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public CSVWriter(  
    string filepath,  
    FileMode fileMode  
)
```

VB

```
Public Sub New (  
    filepath As String,  
    fileMode As FileMode  
)
```

C++

```
public:  
CSVWriter(  
    String^ filepath,  
    FileMode fileMode  
)
```

F#

```
new :  
    filepath : string *  
    fileMode : FileMode -> CSVWriter
```

Parameters

filepath

Type: [System.String](#)

The filepath of the file that data should be written to.

fileMode

Type: [System.IO.FileMode](#)

The mode that the file should be opened in.

See Also













[CSVWriter Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.CSVWriter Methods

The [CSVWriter](#) type exposes the following members.

Methods

	Name	Description
	Close	Closes the file the CSVWriter, and its associated file.
	Dispose()	Implements the IDisposable.Dispose() method.
	Dispose(Boolean)	Disposes of the resources that are utilised by the CSVWriter .
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Open	Opens the file to which data is to be written.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	WriteCSVRecord	Writes a CSVRecord to the file associated with this CSVWriter .
	WriteToFile	Writes the output string to the associated file.

See Also

[CSVWriter Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.Close Method

Closes the file the CSVWriter, and its associated file.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void Close()
```

VB

```
Public Overridable Sub Close
```

C++

```
public:  
virtual void Close()
```

F#

```
abstract Close : unit -> unit  
override Close : unit -> unit
```



See Also

[CSVWriter Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.Dispose Method

Overload List

	Name	Description
	Dispose()	Implements the IDisposable.Dispose() method.
	Dispose(Boolean)	Disposes of the resources that are utilised by the CSVWriter .

See Also

[CSVWriter Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.Dispose Method

Implements the IDisposable.Dispose() method.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Dispose()
```

VB

```
Public Sub Dispose
```

C++

```
public:  
virtual void Dispose() sealed
```

F#

```
abstract Dispose : unit -> unit  
override Dispose : unit -> unit
```

Implements

[IDisposable.Dispose\(\)](#)

Remarks

The implementation calls the protected virtual Dispose() method as per the Dispose pattern.

See Also

[CSVWriter Class](#)

[Dispose Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.Dispose Method (Boolean)

Disposes of the resources that are utilised by the [CSVWriter](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Protected Overridable Sub Dispose (  
    disposing As Boolean  
)
```

C++

```
protected:  
virtual void Dispose(  
    bool disposing  
)
```

F#

```
abstract Dispose :  
    disposing : bool -> unit  
override Dispose :  
    disposing : bool -> unit
```

Parameters

disposing

Type: [System.Boolean](#)

Indicates if the resources are being disposed. True if the resources should be disposed.

See Also

[CSVWriter Class](#)

[Dispose Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.Open Method

Opens the file to which data is to be written.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void Open ( )
```

VB

```
Public Overridable Sub Open
```

C++

```
public:  
virtual void Open ( )
```

F#

```
abstract Open : unit -> unit  
override Open : unit -> unit
```

See Also

[CSVWriter Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.WriteCSVRecord Method

Writes a [CSVRecord](#) to the file associated with this [CSVWriter](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void WriteCSVRecord(  
    CSVRecord record  
)
```

VB

```
Public Sub WriteCSVRecord (  
    record As CSVRecord  
)
```

C++

```
public:  
void WriteCSVRecord(  
    CSVRecord^ record  
)
```

F#

```
member WriteCSVRecord :  
    record : CSVRecord -> unit
```

Parameters

record

Type: [Woodstocks.WoodstocksIMS.Data.CSV.CSVRecord](#)

See Also

[CSVWriter Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.WriteToFile Method

Writes the output string to the associated file.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void WriteToFile(  
    string output  
)
```

VB

```
Protected Overridable Sub WriteToFile (  
    output As String  
)
```

C++

```
protected:  
virtual void WriteToFile(  
    String^ output  
)
```

F#

```
abstract WriteToFile :  
    output : string -> unit  
override WriteToFile :  
    output : string -> unit
```

Parameters

output

Type: [System.String](#)

The string that is to be written to the file associated with the CSVWriter.

See Also



[CSVWriter Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.CSVWriter Properties

The [CSVWriter](#) type exposes the following members.

Properties

	Name	Description
	Filepath	Gets the filepath for the file that the CSVWriter should write data to.
	IsOpen	Gets the status of the file. Returns true if the file is open.

See Also

[CSVWriter Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.Filepath Property

Gets the filepath for the file that the CSVWriter should write data to.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public string Filepath { get; }
```

VB

```
Public ReadOnly Property Filepath As String  
    Get
```

C++

```
public:  
property String^ Filepath {  
    String^ get ();  
}
```

F#

```
member Filepath : string with get
```

Property Value

Type: [String](#)

See Also

[CSVWriter Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

CSVWriter.IsOpen Property

Gets the status of the file. Returns true if the file is open.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool IsOpen { get; }
```

VB

```
Public ReadOnly Property IsOpen As Boolean  
    Get
```

C++

```
public:  
property bool IsOpen {  
    bool get ();  
}
```

F#

```
member IsOpen : bool with get
```

Property Value

Type: [Boolean](#)

See Also

[CSVWriter Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

FieldDoesNotExistException Class

The exception that is raised when an attempt is made to access a named field that does not exist.

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

Woodstocks.WoodstocksIMS.Data.CSV.FieldDoesNotExistException

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class FieldDoesNotExistException : Exception
```

VB

```
Public Class FieldDoesNotExistException  
    Inherits Exception
```

C++

```
public ref class FieldDoesNotExistException : public Exception
```

F#









```
type FieldDoesNotExistException =  
    class  
        inherit Exception  
    end
```

The **FieldDoesNotExistException** type exposes the following members.









Constructors

	Name	Description
	FieldDoesNotExistException()	Initialises a FieldDoesNotExistException .
	FieldDoesNotExistException(String)	Initialises a FieldDoesNotExistException .
	FieldDoesNotExistException(SerializationInfo, StreamingContext)	Initialises a FieldDoesNotExistException .
	FieldDoesNotExistException(String, Exception)	Initialises a FieldDoesNotExistException .


Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

A Sandcastle Documented Class Library

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

FieldDoesNotExistException Constructor

Overload List

	Name	Description
	FieldDoesNotExistException()	Initialises a FieldDoesNotExistException .
	FieldDoesNotExistException(String)	Initialises a FieldDoesNotExistException .
	FieldDoesNotExistException(SerializationInfo, StreamingContext)	Initialises a FieldDoesNotExistException .
	FieldDoesNotExistException(String, Exception)	Initialises a FieldDoesNotExistException .

See Also

[FieldDoesNotExistException Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

FieldDoesNotExistException Constructor

Initialises a [FieldDoesNotExistException](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public FieldDoesNotExistException()
```

VB

```
Public Sub New
```

C++

```
public:  
FieldDoesNotExistException()
```

F#

```
new : unit -> FieldDoesNotExistException
```

See Also

[FieldDoesNotExistException Class](#)

[FieldDoesNotExistException Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

FieldDoesNotExistException Constructor (String)

Initialises a [FieldDoesNotExistException](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public FieldDoesNotExistException(  
    string message  
)
```

VB

```
Public Sub New (  
    message As String  
)
```

C++

```
public:  
FieldDoesNotExistException(  
    String^ message  
)
```

F#

```
new :  
    message : string -> FieldDoesNotExistException
```

Parameters

message

Type: [System.String](#)

[Missing <param name="message"/> documentation for

"M:Woodstocks.WoodstocksIMS.Data.CSV.FieldDoesNotExistException.#ctor(System.String)"]

See Also

[FieldDoesNotExistException Class](#)

[FieldDoesNotExistException Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

FieldDoesNotExistException Constructor (SerializationInfo, StreamingContext)

Initialises a [FieldDoesNotExistException](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public FieldDoesNotExistException(  
    SerializationInfo info,  
    StreamingContext context  
)
```

VB

```
Public Sub New (  
    info As SerializationInfo,  
    context As StreamingContext  
)
```

C++

```
public:  
FieldDoesNotExistException(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

F#

```
new :  
    info : SerializationInfo *  
    context : StreamingContext -> FieldDoesNotExistException
```

Parameters

info

Type: [System.Runtime.Serialization.SerializationInfo](#)

[Missing <param name="info"/> documentation for

"M:Woodstocks.WoodstocksIMS.Data.CSV.FieldDoesNotExistException.#ctor(System.Runtime.Serialization.SerializationInfo,System.Runtime.Serialization.StreamingContext)"]

context

Type: [System.Runtime.Serialization.StreamingContext](#)

[Missing <param name="context"/> documentation for

"M:Woodstocks.WoodstocksIMS.Data.CSV.FieldDoesNotExistException.#ctor(System.Runtime.Serialization.SerializationInfo,System.Runtime.Serialization.StreamingContext)"]

See Also

[FieldDoesNotExistException Class](#)

[FieldDoesNotExistException Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

FieldDoesNotExistException Constructor (String, Exception)

Initialises a [FieldDoesNotExistException](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public FieldDoesNotExistException(  
    string message,  
    Exception innerException  
)
```

VB

```
Public Sub New (  
    message As String,  
    innerException As Exception  
)
```

C++

```
public:  
FieldDoesNotExistException(  
    String^ message,  
    Exception^ innerException  
)
```

F#

```
new :  
    message : string *  
    innerException : Exception -> FieldDoesNotExistException
```

Parameters

message

Type: [System.String](#)

[Missing <param name="message"/> documentation for

"M:Woodstocks.WoodstocksIMS.Data.CSV.FieldDoesNotExistException.#ctor(System.String,System.Exception)"]

innerException

Type: [System.Exception](#)

[Missing <param name="innerException"/> documentation for

"M:Woodstocks.WoodstocksIMS.Data.CSV.FieldDoesNotExistException.#ctor(System.String,System.Exception)"]

See Also

[FieldDoesNotExistException Class](#)

A Sandcastle Documented Class Library









[FieldDoesNotExistException Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

FieldDoesNotExistException.FieldDoesNotExistException Methods

The [FieldDoesNotExistException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

See Also









[FieldDoesNotExistException Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

FieldDoesNotExistException.FieldDoesNotExistException Properties

The [FieldDoesNotExistException](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

See Also


[FieldDoesNotExistException Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

FieldDoesNotExistException.FieldDoesNotExistException Events

The [FieldDoesNotExistException](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[FieldDoesNotExistException Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

MissingValueException Class

The exception that is raised when a value is missing from a [CSVRecord](#).

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

Woodstocks.WoodstocksIMS.Data.CSV.MissingValueException

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class MissingValueException : Exception
```

VB

```
Public Class MissingValueException  
    Inherits Exception
```

C++





```
public ref class MissingValueException : public Exception
```

F#


```
type MissingValueException =  
    class  
        inherit Exception  
    end
```








The **MissingValueException** type exposes the following members.

Constructors









	Name	Description
	MissingValueException()	Initializes a MissingValueException .
	MissingValueException(String)	Initializes a MissingValueException .
	MissingValueException(SerializationInfo, StreamingContext)	Initializes a MissingValueException .
	MissingValueException(String, Exception)	Initializes a MissingValueException .

Methods


	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)

	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
 Message	Gets a message that describes the current exception. (Inherited from Exception .)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
 StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Events





Name	Description
 SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

MissingValueException Constructor

Overload List

	Name	Description
	MissingValueException()	Initializes a MissingValueException .
	MissingValueException(String)	Initializes a MissingValueException .
	MissingValueException(SerializationInfo, StreamingContext)	Initializes a MissingValueException .
	MissingValueException(String, Exception)	Initializes a MissingValueException .

See Also

[MissingValueException Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

MissingValueException Constructor

Initializes a [MissingValueException](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public MissingValueException()
```

VB

```
Public Sub New
```

C++

```
public:  
MissingValueException()
```

F#

```
new : unit -> MissingValueException
```

See Also

[MissingValueException Class](#)

[MissingValueException Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

MissingValueException Constructor (String)

Initializes a [MissingValueException](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public MissingValueException(  
    string message  
)
```

VB

```
Public Sub New (  
    message As String  
)
```

C++

```
public:  
MissingValueException(  
    String^ message  
)
```

F#

```
new :  
    message : string -> MissingValueException
```

Parameters

message

Type: [System.String](#)

An error message for the exception.

See Also

[MissingValueException Class](#)

[MissingValueException Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

MissingValueException Constructor (SerializationInfo, StreamingContext)

Initializes a [MissingValueException](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public MissingValueException(  
    SerializationInfo info,  
    StreamingContext context  
)
```

VB

```
Public Sub New (  
    info As SerializationInfo,  
    context As StreamingContext  
)
```

C++

```
public:  
MissingValueException(  
    SerializationInfo^ info,  
    StreamingContext context  
)
```

F#

```
new :  
    info : SerializationInfo *  
    context : StreamingContext -> MissingValueException
```

Parameters

info

Type: [System.Runtime.Serialization.SerializationInfo](#)

context

Type: [System.Runtime.Serialization.StreamingContext](#)

See Also

[MissingValueException Class](#)

[MissingValueException Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

MissingValueException Constructor (String, Exception)

Initializes a [MissingValueException](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public MissingValueException(  
    string message,  
    Exception innerException  
)
```

VB

```
Public Sub New (  
    message As String,  
    innerException As Exception  
)
```

C++

```
public:  
MissingValueException(  
    String^ message,  
    Exception^ innerException  
)
```

F#

```
new :  
    message : string *  
    innerException : Exception -> MissingValueException
```

Parameters

message

Type: [System.String](#)

An error message for the exception.

innerException

Type: [System.Exception](#)

The exception that raised this exception.

See Also

[MissingValueException Class](#)









[MissingValueException Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

MissingValueException.MissingValueException Methods

The [MissingValueException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

See Also









[MissingValueException Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

MissingValueException.MissingValueException Properties

The [MissingValueException](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

See Also


[MissingValueException Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

MissingValueException.MissingValueException Events

The [MissingValueException](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[MissingValueException Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV Class

An exporter to export Wood Stocks [Toy](#) stock data to a csv data file.

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Data.CSV.ToyExporterCSV

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class ToyExporterCSV : IWoodstocksToyExporter
```

VB

```
Public Class ToyExporterCSV
    Implements IWoodstocksToyExporter
```

C++


```
public ref class ToyExporterCSV : IWoodstocksToyExporter
```

F#




```
type ToyExporterCSV =
    class
        interface IWoodstocksToyExporter
    end
```
















The **ToyExporterCSV** type exposes the following members.

Constructors


	Name	Description
	ToyExporterCSV	Initialises a ToyExporterCSV .

Methods



	Name	Description
	AppendCSVFileExtension	Checks whehter a file name, assumed to include, the path of the file has a .csv extension. Appends the .csv extension if the filename string does not have the .csv extension.
	BackgroundWorker RunWorkerCompleted	Handles the RunWorkerCompleted event of the BackgroundWorker used to carry out the exportation.
	Close	Closes the exporter.

	CopyDataForExport	Creates a copy of the data that is to be exported to the csv file.
	CreateBackgroundWorker	Creates a BackgroundWorker to be used to export data asynchronously.
	CreateBackupFile	Creates a backup file for the file to which data is to be exported.
	DoExport	The method that is called by a BackgroundWorker to perform an asynchronous exportation of IToy data to a csv file.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	ExportAsync	Exports data asynchronously to a csv data file.
	ExportCancel	Cancels an asynchronous export of toy data by the exporter.
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnExportCompleted	Handles the completion event of the BackgroundWorker by raising the ExportCompleted event.
	OnExportProgressChanged	Raises the ExportProgressChanged event.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	UpdateProgress	Handles the progress changed event of the background worker asynchronously exporting data.

Properties

	Name	Description
	IsBusy	Indicates if the exporter is busy carrying out an asynchronous exportation of toy data.

Events

	Name	Description
	ExportCompleted	Event that is raised upon completion of exportation.
	ExportProgressChanged	Event that is raised upon progress of exportation.

A Sandcastle Documented Class Library

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV Constructor

Initialises a [ToyExporterCSV](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public ToyExporterCSV( )
```

VB

```
Public Sub New
```

C++

```
public:  
ToyExporterCSV( )
```

F#

```
new : unit -> ToyExporterCSV
```

See Also


















[ToyExporterCSV Class](#)


[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.ToyExporterCSV Methods

The [ToyExporterCSV](#) type exposes the following members.

Methods

	Name	Description
	AppendCSVFileExtension	Checks whehter a file name, assumed to include, the path of the file has a .csv extension. Appends the .csv extension if the filename string does not have the .csv extension.
	BackgroundWorker_RunWorkerCompleted	Handles the RunWorkerCompleted event of the BackgroundWorker used to carry out the exportation.
	Close	Closes the exporter.
	CopyDataForExport	Creates a copy of the data that is to be exported to the csv file.
	CreateBackgroundWorker	Creates a BackgroundWorker to be used to export data asynchronously.
	CreateBackupFile	Creates a backup file for the file to which data is to be exported.
	DoExport	The method that is called by a BackgroundWorker to perform an asynchronous exportation of IToy data to a csv file.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	ExportAsync	Exports data asynchronously to a csv data file.
	ExportCancel	Cancels an asynchronous export of toy data by the exporter.
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnExportCompleted	Handles the completion event of the BackgroundWorker by raising the ExportCompleted event.
	OnExportProgressChanged	Raises the ExportProgressChanged event.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

	UpdateProgress	Handles the progress changed event of the background worker asynchronously exporting data.
---	--------------------------------	--

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.AppendCSVFileExtension Method

Checks whehter a file name, assumed to include, the path of the file has a .csv extension. Appends the .csv extension if the filename string does not have the .csv extension.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected string AppendCSVFileExtension(  
    string filename  
)
```

VB

```
Protected Function AppendCSVFileExtension (  
    filename As String  
) As String
```

C++

```
protected:  
String^ AppendCSVFileExtension(  
    String^ filename  
)
```

F#

```
member AppendCSVFileExtension :  
    filename : string -> string
```

Parameters

filename

Type: [System.String](#)

The file name, including the path, to the file.

Return Value

Type: [String](#)

A file name, with the .csv extension appended, if it does not contain a .csv extension, otherwise the original filename.

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.BackgroundWorker_RunWorkerCompleted Method

Handles the RunWorkerCompleted event of the [BackgroundWorker](#) used to carry out the exportation.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void BackgroundWorker_RunWorkerCompleted(  
    Object sender,  
    RunWorkerCompletedEventArgs e  
)
```

VB

```
Protected Overridable Sub BackgroundWorker_RunWorkerCompleted (  
    sender As Object,  
    e As RunWorkerCompletedEventArgs  
)
```

C++

```
protected:  
virtual void BackgroundWorker_RunWorkerCompleted(  
    Object^ sender,  
    RunWorkerCompletedEventArgs^ e  
)
```

F#

```
abstract BackgroundWorker_RunWorkerCompleted :  
    sender : Object *  
    e : RunWorkerCompletedEventArgs -> unit  
override BackgroundWorker_RunWorkerCompleted :  
    sender : Object *  
    e : RunWorkerCompletedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The BackgroundWorker carrying out the exportation.

e

Type: [System.ComponentModel.RunWorkerCompletedEventArgs](#)

The event data for the event.

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.Close Method

Closes the exporter.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Close()
```

VB

```
Public Sub Close
```

C++

```
public:  
virtual void Close() sealed
```

F#

```
abstract Close : unit -> unit  
override Close : unit -> unit
```

Implements

[IWoodstocksToyExporter.Close\(\)](#)

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.CopyDataForExport Method

Creates a copy of the data that is to be exported to the csv file.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected IToys CopyDataForExport(  
    IToys toys  
)
```

VB

```
Protected Function CopyDataForExport (  
    toys As IToys  
) As IToys
```

C++

```
protected:  
IToys^ CopyDataForExport(  
    IToys^ toys  
)
```

F#

```
member CopyDataForExport :  
    toys : IToys -> IToys
```

Parameters

toys

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The data to be exported to the file.

Return Value

Type: [IToys](#)

A copy of the data to be exported.

Remarks

A copy of the data is made prior to exporting the data to ensure that the data that is to be exported is not modified whilst it is being exported.

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.CreateBackgroundWorker Method

Creates a [BackgroundWorker](#) to be used to export data asynchronously.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void CreateBackgroundWorker( )
```

VB

```
Protected Overridable Sub CreateBackgroundWorker
```

C++

```
protected:  
virtual void CreateBackgroundWorker( )
```

F#

```
abstract CreateBackgroundWorker : unit -> unit  
override CreateBackgroundWorker : unit -> unit
```

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.CreateBackupFile Method

Creates a backup file for the file to which data is to be exported.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void CreateBackupFile(  
    string filename  
)
```

VB

```
Protected Overridable Sub CreateBackupFile (  
    filename As String  
)
```

C++

```
protected:  
virtual void CreateBackupFile(  
    String^ filename  
)
```

F#

```
abstract CreateBackupFile :  
    filename : string -> unit  
override CreateBackupFile :  
    filename : string -> unit
```

Parameters

filename

Type: [System.String](#)

The file name, including path, of the file that is to be backed up.

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.DoExport Method

The method that is called by a [BackgroundWorker](#) to perform an asynchronous exportation of [IToy](#) data to a csv file.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void DoExport(  
    Object sender,  
    DoWorkEventArgs e  
)
```

VB

```
Protected Overridable Sub DoExport (  
    sender As Object,  
    e As DoWorkEventArgs  
)
```

C++

```
protected:  
virtual void DoExport(  
    Object^ sender,  
    DoWorkEventArgs^ e  
)
```

F#

```
abstract DoExport :  
    sender : Object *  
    e : DoWorkEventArgs -> unit  
override DoExport :  
    sender : Object *  
    e : DoWorkEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The worker performing the export.

e

Type: [System.ComponentModel.DoWorkEventArgs](#)

Event data passed by the worker.

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.ExportAsync Method

Exports data asynchronously to a csv data file.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ExportAsync(  
    string filename,  
    IToys data  
)
```

VB

```
Public Sub ExportAsync (  
    filename As String,  
    data As IToys  
)
```

C++

```
public:  
virtual void ExportAsync(  
    String^ filename,  
    IToys^ data  
) sealed
```

F#

```
abstract ExportAsync :  
    filename : string *  
    data : IToys -> unit  
override ExportAsync :  
    filename : string *  
    data : IToys -> unit
```

Parameters

filename

Type: [System.String](#)

The name of the file that data should be exported to, resulting in the data in the csv file being updated.

data

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The data to be written to the file.

Implements

[IWoodstocksToyExporter.ExportAsync\(String, IToys\)](#)

A Sandcastle Documented Class Library

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.ExportCancel Method

Cancels an asynchronous export of toy data by the exporter.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ExportCancel()
```

VB

```
Public Sub ExportCancel
```

C++

```
public:  
virtual void ExportCancel() sealed
```

F#

```
abstract ExportCancel : unit -> unit  
override ExportCancel : unit -> unit
```

Implements

[IWoodstocksToyExporter.ExportCancel\(\)](#)

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.OnExportCompleted Method

Handles the completion event of the BackgroundWorker by raising the [ExportCompleted](#) event.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void OnExportCompleted(  
    AsyncCompletedEventArgs e  
)
```

VB

```
Protected Overridable Sub OnExportCompleted (  
    e As AsyncCompletedEventArgs  
)
```

C++

```
protected:  
virtual void OnExportCompleted(  
    AsyncCompletedEventArgs^ e  
)
```

F#

```
abstract OnExportCompleted :  
    e : AsyncCompletedEventArgs -> unit  
override OnExportCompleted :  
    e : AsyncCompletedEventArgs -> unit
```

Parameters

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

The completion event data sent by the worker.

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.OnExportProgressChanged Method

Raises the [ExportProgressChanged](#) event.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void OnExportProgressChanged(  
    ProgressChangedEventArgs e  
)
```

VB

```
Protected Overridable Sub OnExportProgressChanged (  
    e As ProgressChangedEventArgs  
)
```

C++

```
protected:  
virtual void OnExportProgressChanged(  
    ProgressChangedEventArgs^ e  
)
```

F#

```
abstract OnExportProgressChanged :  
    e : ProgressChangedEventArgs -> unit  
override OnExportProgressChanged :  
    e : ProgressChangedEventArgs -> unit
```

Parameters

e

Type: [System.ComponentModel.ProgressChangedEventArgs](#)

Event data that indicates the progress of the operation.

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.UpdateProgress Method

Handles the progress changed event of the background worker asynchronously exporting data.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void UpdateProgress(  
    Object sender,  
    ProgressChangedEventArgs e  
)
```

VB

```
Protected Overridable Sub UpdateProgress (  
    sender As Object,  
    e As ProgressChangedEventArgs  
)
```

C++

```
protected:  
virtual void UpdateProgress(  
    Object^ sender,  
    ProgressChangedEventArgs^ e  
)
```

F#

```
abstract UpdateProgress :  
    sender : Object *  
    e : ProgressChangedEventArgs -> unit  
override UpdateProgress :  
    sender : Object *  
    e : ProgressChangedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The [BackgroundWorker](#) that raised the event.

e

Type: [System.ComponentModel.ProgressChangedEventArgs](#)

The progress of the exportation.

See Also


[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.ToyExporterCSV Properties

The [ToyExporterCSV](#) type exposes the following members.

Properties

	Name	Description
	IsBusy	Indicates if the exporter is busy carrying out an asynchronous exportation of toy data.

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.IsBusy Property

Indicates if the exporter is busy carrying out an asynchronous exportation of toy data.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool IsBusy { get; }
```

VB

```
Public ReadOnly Property IsBusy As Boolean  
    Get
```

C++

```
public:  
virtual property bool IsBusy {  
    bool get () sealed;  
}
```

F#

```
abstract IsBusy : bool with get  
override IsBusy : bool with get
```

Return Value

Type: [Boolean](#)

True if the exporter is busy exporting data, otherwise false.

Implements

[IWoodstocksToyExporter.IsBusy](#)

See Also



[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.ToyExporterCSV Events

The [ToyExporterCSV](#) type exposes the following members.

Events

	Name	Description
	ExportCompleted	Event that is raised upon completion of exportation.
	ExportProgressChanged	Event that is raised upon progress of exportation.

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.ExportCompleted Event

Event that is raised upon completion of exportation.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event AsyncCompletedEventHandler ExportCompleted
```

VB

```
Public Event ExportCompleted As AsyncCompletedEventHandler
```

C++

```
public:  
virtual event AsyncCompletedEventHandler^ ExportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>  
override ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

Implements

[IWoodstocksToyExporter.ExportCompleted](#)

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyExporterCSV.ExportProgressChanged Event

Event that is raised upon progress of exportation.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event ProgressChangedEventHandler ExportProgressChanged
```

VB

```
Public Event ExportProgressChanged As ProgressChangedEventHandler
```

C++

```
public:  
virtual event ProgressChangedEventHandler^ ExportProgressChanged {  
    void add (ProgressChangedEventHandler^ value);  
    void remove (ProgressChangedEventHandler^ value);  
}
```

F#

```
abstract ExportProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>  
override ExportProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>
```

Value

Type: [System.ComponentModel.ProgressChangedEventHandler](#)

Implements

[IWoodstocksToyExporter.ExportProgressChanged](#)

See Also

[ToyExporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV Class

An importer to import toy data from a csv data file.

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Data.CSV.ToyImporterCSV

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class ToyImporterCSV : IWoodstocksToyImporter,
    IDisposable
```

VB

```
Public Class ToyImporterCSV
    Implements IWoodstocksToyImporter, IDisposable
```

C++


```
public ref class ToyImporterCSV : IWoodstocksToyImporter,
    IDisposable
```

F#




```
type ToyImporterCSV =
    class
        interface IWoodstocksToyImporter
        interface IDisposable
    end
```
















The **ToyImporterCSV** type exposes the following members.

Constructors



	Name	Description
	ToyImporterCSV	Initialises a ToyImporterCSV .

Methods

	Name	Description
	Close	Closes the importer.
	CompleteImport	Handles the RunbackgroundWorkerCompleted event of the BackgroundbackgroundWorker performing an asynchronous import.
	Dispose()	Performs application-defined tasks associated with freeing, releasing, or

		resetting unmanaged resources.
	Dispose(Boolean)	Implements the IDisposable.Dispose() method.
	DoImport	Performs an asynchronous importation of toy data from a csv data file.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetToys	Retrieves the imported data from the ToyImporterCSV .
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ImportAsync	Imports toy data asynchronously.
	ImportCancel	Cancels an asynchronous import.
	IsBusy	Returns whether the importer is busy whilst carrying out an asynchronous import.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnImportCompleted	Raises the ImportCompleted event of the ToyImporterCSV .
	OnImportProgressChanged	Raises the ImportProgressChanged event of the ToyImporterCSV .
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	UpdateProgress	Handles the ProgressChanged event raised by the BackgroundbackgroundWorker carrying out an asynchronous import.

Events

	Name	Description
	ImportCompleted	Raised upon completion of an asynchronous operation.
	ImportProgressChanged	The event when progress is made on an asynchronous import.

See Also

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV Constructor

Initialises a [ToyImporterCSV](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public ToyImporterCSV( )
```

VB

```
Public Sub New
```

C++

```
public:  
ToyImporterCSV( )
```

F#

```
new : unit -> ToyImporterCSV
```

See Also



















[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.ToyImporterCSV Methods

The [ToyImporterCSV](#) type exposes the following members.

Methods

	Name	Description
	Close	Closes the importer.
	CompletemImport	Handles the RunbackgroundWorkerCompleted event of the BackgroundbackgroundWorker performing an asynchronous import.
	Dispose()	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Dispose(Boolean)	Implements the IDisposable.Dispose() method.
	DoImport	Performs an asynchronous importation of toy data from a csv data file.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetToys	Retrieves the imported data from the ToyImporterCSV .
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ImportAsync	Imports toy data asynchronously.
	ImportCancel	Cancels an asynchronous import.
	IsBusy	Returns whether the importer is busy wilst carrying out an asynhronous import.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnImportCompleted	Raises the ImportCompleted event of the ToyImporterCSV .
	OnImportProgressChanged	Raises the ImportProgressChanged event of the ToyImporterCSV .
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	UpdateProgress	Handles the ProgressChanged event raised by the BackgroundbackgroundWorker carrying out an asynchronous import.

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.Close Method

Closes the importer.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Close()
```

VB

```
Public Sub Close
```

C++

```
public:  
virtual void Close() sealed
```

F#

```
abstract Close : unit -> unit  
override Close : unit -> unit
```

Implements

[IWoodstocksToyImporter.Close\(\)](#)

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.CompleteImport Method

Handles the RunbackgroundWorkerCompleted event of the BackgroundbackgroundWorker performing an asynchronous import.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void CompleteImport(  
    Object sender,  
    RunWorkerCompletedEventArgs e  
)
```

VB

```
Protected Overridable Sub CompleteImport (  
    sender As Object,  
    e As RunWorkerCompletedEventArgs  
)
```

C++

```
protected:  
virtual void CompleteImport(  
    Object^ sender,  
    RunWorkerCompletedEventArgs^ e  
)
```

F#

```
abstract CompleteImport :  
    sender : Object *  
    e : RunWorkerCompletedEventArgs -> unit  
override CompleteImport :  
    sender : Object *  
    e : RunWorkerCompletedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The BackgroundbackgroundWorker that raised the event.

e

Type: [System.ComponentModel.RunWorkerCompletedEventArgs](#)

The event data sent by the BackgroundWorker.

Remarks

The event is handled by raising the ImportCompleted event of the [ToyImporterCSV](#).

A Sandcastle Documented Class Library



See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.Dispose Method

Overload List

	Name	Description
	Dispose()	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.
	Dispose(Boolean)	Implements the IDisposable.Dispose() method.

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.Dispose Method

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Dispose()
```

VB

```
Public Sub Dispose
```

C++

```
public:  
virtual void Dispose() sealed
```

F#

```
abstract Dispose : unit -> unit  
override Dispose : unit -> unit
```

Implements

[IDisposable.Dispose\(\)](#)

See Also

[ToyImporterCSV Class](#)

[Dispose Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.Dispose Method (Boolean)

Implements the IDisposable.Dispose() method.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Protected Overridable Sub Dispose (  
    disposing As Boolean  
)
```

C++

```
protected:  
virtual void Dispose(  
    bool disposing  
)
```

F#

```
abstract Dispose :  
    disposing : bool -> unit  
override Dispose :  
    disposing : bool -> unit
```

Parameters

disposing

Type: [System.Boolean](#)

True to release both managed and unmanaged resources; false to release only unmanaged resources

See Also

[ToyImporterCSV Class](#)

[Dispose Overload](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.DoImport Method

Performs an asynchronous importation of toy data from a csv data file.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void DoImport(  
    Object sender,  
    DoWorkEventArgs e  
)
```

VB

```
Protected Overridable Sub DoImport (  
    sender As Object,  
    e As DoWorkEventArgs  
)
```

C++

```
protected:  
virtual void DoImport(  
    Object^ sender,  
    DoWorkEventArgs^ e  
)
```

F#

```
abstract DoImport :  
    sender : Object *  
    e : DoWorkEventArgs -> unit  
override DoImport :  
    sender : Object *  
    e : DoWorkEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

A [BackgroundWorker](#) that invoked the method asynchronously.

e

Type: [System.ComponentModel.DoWorkEventArgs](#)

Event data passed by the BackgroundWorker.

Remarks

The Argument member of the event data contains the source (filepath) of the csv data file from which toy data should be imported.

A Sandcastle Documented Class Library

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.GetToys Method

Retrieves the imported data from the [ToyImporterCSV](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IToys GetToys()
```

VB

```
Public Function GetToys As IToys
```

C++

```
public:  
virtual IToys^ GetToys() sealed
```

F#

```
abstract GetToys : unit -> IToys  
override GetToys : unit -> IToys
```

Return Value

Type: [IToys](#)

An interface reference to the toy data imported, or null if no data was imported. The returned value should be checked for null before attempting to use the data.

Implements

[IWoodstocksToyImporter.GetToys\(\)](#)

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.ImportAsync Method

Imports toy data asynchronously.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ImportAsync(  
    string source  
)
```

VB

```
Public Sub ImportAsync (  
    source As String  
)
```

C++

```
public:  
virtual void ImportAsync(  
    String^ source  
) sealed
```

F#

```
abstract ImportAsync :  
    source : string -> unit  
override ImportAsync :  
    source : string -> unit
```

Parameters

source

Type: [System.String](#)

The source from which data should be imported.

Implements

[IWoodstocksToyImporter.ImportAsync\(String\)](#)

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.ImportCancel Method

Cancels an asynchronous import.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ImportCancel()
```

VB

```
Public Sub ImportCancel
```

C++

```
public:  
virtual void ImportCancel() sealed
```

F#

```
abstract ImportCancel : unit -> unit  
override ImportCancel : unit -> unit
```

Implements

[IWoodstocksToyImporter.ImportCancel\(\)](#)

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.IsBusy Method

Returns whether the importer is busy whilst carrying out an asynchronous import.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool IsBusy()
```

VB

```
Public Function IsBusy As Boolean
```

C++

```
public:  
virtual bool IsBusy() sealed
```

F#

```
abstract IsBusy : unit -> bool  
override IsBusy : unit -> bool
```

Return Value

Type: [Boolean](#)

True if the importer is busy carrying out an asynchronous import. False if the importer is not busy.

Implements

[IWoodstocksToyImporter.IsBusy\(\)](#)

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.OnImportCompleted Method

Raises the ImportCompleted event of the [ToyImporterCSV](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void OnImportCompleted(  
    AsyncCompletedEventArgs e  
)
```

VB

```
Protected Overridable Sub OnImportCompleted (  
    e As AsyncCompletedEventArgs  
)
```

C++

```
protected:  
virtual void OnImportCompleted(  
    AsyncCompletedEventArgs^ e  
)
```

F#

```
abstract OnImportCompleted :  
    e : AsyncCompletedEventArgs -> unit  
override OnImportCompleted :  
    e : AsyncCompletedEventArgs -> unit
```

Parameters

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.OnImportProgressChanged Method

Raises the [ImportProgressChanged](#) event of the [ToyImporterCSV](#).

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void OnImportProgressChanged(  
    ProgressChangedEventArgs e  
)
```

VB

```
Protected Overridable Sub OnImportProgressChanged (  
    e As ProgressChangedEventArgs  
)
```

C++

```
protected:  
virtual void OnImportProgressChanged(  
    ProgressChangedEventArgs^ e  
)
```

F#

```
abstract OnImportProgressChanged :  
    e : ProgressChangedEventArgs -> unit  
override OnImportProgressChanged :  
    e : ProgressChangedEventArgs -> unit
```

Parameters

e

Type: [System.ComponentModel.ProgressChangedEventArgs](#)

The progress change data for the event.

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.UpdateProgress Method

Handles the ProgressChanged event raised by the BackgroundbackgroundWorker carrying out an asynchronous import.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void UpdateProgress(  
    Object sender,  
    ProgressChangedEventArgs e  
)
```

VB

```
Protected Overridable Sub UpdateProgress (  
    sender As Object,  
    e As ProgressChangedEventArgs  
)
```

C++

```
protected:  
virtual void UpdateProgress(  
    Object^ sender,  
    ProgressChangedEventArgs^ e  
)
```

F#

```
abstract UpdateProgress :  
    sender : Object *  
    e : ProgressChangedEventArgs -> unit  
override UpdateProgress :  
    sender : Object *  
    e : ProgressChangedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The BackgroundbackgroundWorker that raised the event.

e

Type: [System.ComponentModel.ProgressChangedEventArgs](#)

A [ProgressChangedEventArgs](#) containing the progress change data.

Remarks

Handles the backgroundWorkers progress by raising the ImportProgressChanged event of the [ToyImporterCSV](#).

A Sandcastle Documented Class Library

See Also



[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.ToyImporterCSV Events

The [ToyImporterCSV](#) type exposes the following members.

Events

	Name	Description
	ImportCompleted	Raised upon completion of an asynchronous operation.
	ImportProgressChanged	The event when progress is made on an asynchronous import.

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.ImportCompleted Event

Raised upon completion of an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event ImportCompletedHandler ImportCompleted
```

VB

```
Public Event ImportCompleted As ImportCompletedHandler
```

C++

```
public:
virtual event ImportCompletedHandler^ ImportCompleted {
    void add (ImportCompletedHandler^ value);
    void remove (ImportCompletedHandler^ value);
}
```

F#

```
abstract ImportCompleted : IEvent<ImportCompletedHandler,
    EventArgs>
override ImportCompleted : IEvent<ImportCompletedHandler,
    EventArgs>
```

Value

Type: [Woodstocks.WoodstocksIMS.Domain.ImportCompletedHandler](#)

Implements

[IWoodstocksToyImporter.ImportCompleted](#)

Remarks

The [Error](#) should be checked to ensure that an error did not occur during the operation. If no error has occurred during the operation the result of the import operation can be retrieved by calling the [GetToys\(\)](#) method.

See Also

[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)

ToyImporterCSV.ImportProgressChanged Event

The event when progress is made on an asynchronous import.

Namespace: [Woodstocks.WoodstocksIMS.Data.CSV](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event ImportProgressChanged ImportProgressChanged
```

VB

```
Public Event ImportProgressChanged As ImportProgressChanged
```

C++

```
public:  
virtual event ImportProgressChanged^ ImportProgressChanged {  
    void add (ImportProgressChanged^ value);  
    void remove (ImportProgressChanged^ value);  
}
```

F#

```
abstract ImportProgressChanged : IEvent<ImportProgressChanged,  
    EventArgs>  
override ImportProgressChanged : IEvent<ImportProgressChanged,  
    EventArgs>
```

Value

Type: [Woodstocks.WoodstocksIMS.Domain.ImportProgressChanged](#)

Implements

[IWoodstocksToyImporter.ImportProgressChanged](#)

See Also















[ToyImporterCSV Class](#)

[Woodstocks.WoodstocksIMS.Data.CSV Namespace](#)







Woodstocks.WoodstocksIMS.Domain Namespace

The Woodstocks.WoodstocksIMS.Domain namespace contains the domain layer components for the WoodstocksIMS.





Classes

	Class	Description
	DuplicateToyException	Exception that is raised when it is detected that a duplicate toy is being added to a Toys collection.
	ImportToysCompletedEventArgs	Defines event data for the ImportCompleted event.
	InvalidCurrentCountException	Defines an exception that is raised when the value to be used for the current count of a Toy is invalid.
	InvalidDescriptionException	An exception that is raised when the value for a description of a Toy is invalid.
	InvalidItemCodeException	Defines an exception that is raised when the item code for a Toy is invalid.
	InvalidOnOrderException	Defines an exception to be raised when an OnOrder value is invalid.
	Toy	Represents a Toy sold by Wood Stocks.
	ToyChangedEventArgs	Contains the event data for a ToyChanged
	Toys	A collection of Toys.
	UnsavedDataException	Exception that occurs when stock data imported into WoodstocksIMS will be discarded without changes being saved.
	WoodstocksDataConverter	Contains data conversion methods.
	WoodstocksIMS	Implementation of the Wood Stocks Inventory Management System.
	WoodstocksIMSState	An abstract class to define state objects for the WoodstocksIMS .
	WoodstocksToyValidator	A static class that contains methods for performing validation of Wood Stocks toy data.




Interfaces

	Interface	Description
	IToy	Defines the interface of a Toy.
	IToys	Defines an interface for a collection of IToy .
	IWoodstocksIMS	Defines an interface to the WoodstocksIMS .
	IWoodstocksIMSClient	Defines a client interface for the WoodstocksIMS .
	IWoodstocksToyExporter	Defines methods for an exporter to export toy data.
	IWoodstocksToyImporter	Defines an interface to import toy data into the WoodstocksIMS .

Delegates

	Delegate	Description
	ImportCompletedHandler	Defines a delegate to handle the ImportCompleted event.
	ImportProgressChanged	Defines a delegate to handle the ImportProgressChanged event.
	ImportToysCompletedEventHandler	Defines an delegate to handle the ImportToysCompleted event
	ToyChangedEventHandler	Defines a delegate to handle the Changed event.

Enumerations

	Enumeration	Description
	OnOrder	Represents the order status of a Toy .
	SortOrder	Defines an enumeration for sort ordering.
	ToyField	Defines named constants for the fields of an IToy .

DuplicateToyException Class

Exception that is raised when it is detected that a duplicate toy is being added to a [Toys](#) collection.

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

Woodstocks.WoodstocksIMS.Domain.DuplicateToyException

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class DuplicateToyException : Exception
```

VB

```
Public Class DuplicateToyException  
    Inherits Exception
```

C++


```
public ref class DuplicateToyException : public Exception
```

F#




```
type DuplicateToyException =  
    class  
        inherit Exception  
    end
```






The **DuplicateToyException** type exposes the following members.

Constructors









	Name	Description
	DuplicateToyException	Initialises a DuplicateToyException .

Methods


	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)

	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

DuplicateToyException Constructor

Initialises a [DuplicateToyException](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public DuplicateToyException(  
    string itemCode  
)
```

VB

```
Public Sub New (  
    itemCode As String  
)
```

C++

```
public:  
DuplicateToyException(  
    String^ itemCode  
)
```

F#

```
new :  
    itemCode : string -> DuplicateToyException
```

Parameters

itemCode

Type: [System.String](#)

See Also









[DuplicateToyException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

DuplicateToyException.DuplicateToyException Methods

The [DuplicateToyException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

See Also









[DuplicateToyException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

DuplicateToyException.DuplicateToyException Properties

The [DuplicateToyException](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

See Also


[DuplicateToyException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

DuplicateToyException.DuplicateToyException Events

The [DuplicateToyException](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[DuplicateToyException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ImportCompletedHandler Delegate

Defines a delegate to handle the [ImportCompleted](#) event.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public delegate void ImportCompletedHandler(  
    IWoodstocksToyImporter sender,  
    AsyncCompletedEventArgs e  
)
```

VB

```
Public Delegate Sub ImportCompletedHandler (  
    sender As IWoodstocksToyImporter,  
    e As AsyncCompletedEventArgs  
)
```

C++

```
public delegate void ImportCompletedHandler(  
    IWoodstocksToyImporter^ sender,  
    AsyncCompletedEventArgs^ e  
)
```

F#

```
type ImportCompletedHandler =  
    delegate of  
        sender : IWoodstocksToyImporter *  
        e : AsyncCompletedEventArgs -> unit
```

Parameters

sender

Type: [Woodstocks.WoodstocksIMS.Domain.IWoodstocksToyImporter](#)

The [IWoodstocksToyImporter](#) that raised the event.

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

The event data for the [ImportCompleted](#) event.

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ImportProgressChanged Delegate

Defines a delegate to handle the [ImportProgressChanged](#) event.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public delegate void ImportProgressChanged(  
    IWoodstocksToyImporter sender,  
    ProgressChangedEventArgs e  
)
```

VB

```
Public Delegate Sub ImportProgressChanged (  
    sender As IWoodstocksToyImporter,  
    e As ProgressChangedEventArgs  
)
```

C++

```
public delegate void ImportProgressChanged(  
    IWoodstocksToyImporter^ sender,  
    ProgressChangedEventArgs^ e  
)
```

F#

```
type ImportProgressChanged =  
    delegate of  
        sender : IWoodstocksToyImporter *  
        e : ProgressChangedEventArgs -> unit
```

Parameters

sender

Type: [Woodstocks.WoodstocksIMS.Domain.IWoodstocksToyImporter](#)

The [IWoodstocksToyImporter](#) that raised the event.

e

Type: [System.ComponentModel.ProgressChangedEventArgs](#)

The event data for the [ImportProgressChanged](#) event.

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ImportToysCompletedEventArgs Class

Defines event data for the [ImportCompleted](#) event.

Inheritance Hierarchy

[System.Object](#)

[System.EventArgs](#)

[System.ComponentModel.AsyncCompletedEventArgs](#)

Woodstocks.WoodstocksIMS.Domain.ImportToysCompletedEventArgs

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

```
C#  
public class ImportToysCompletedEventArgs : AsyncCompletedEventArgs
```



```
VB  
Public Class ImportToysCompletedEventArgs  
    Inherits AsyncCompletedEventArgs
```

```
C++  
public ref class ImportToysCompletedEventArgs : public  
AsyncCompletedEventArgs
```


```
F#  
type ImportToysCompletedEventArgs =  
    class  
        inherit AsyncCompletedEventArgs  
    end
```







The **ImportToysCompletedEventArgs** type exposes the following members.

Constructors





	Name	Description
	ImportToysCompletedEventArgs(Exception, Boolean, Object)	Initialises a ImportToysCompletedEventArgs .
	ImportToysCompletedEventArgs(Exception, Boolean, Object, IToys)	Initialises a ImportToysCompletedEventArgs .

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object .

		(Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	RaiseExceptionIfNecessary	Raises a user-supplied exception if an asynchronous operation failed. (Inherited from AsyncCompletedEventArgs .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Properties



	Name	Description
	Cancelled	Gets a value indicating whether an asynchronous operation has been canceled. (Inherited from AsyncCompletedEventArgs .)
	Error	Gets a value indicating which error occurred during an asynchronous operation. (Inherited from AsyncCompletedEventArgs .)
	Result	Gets the result of an toy importation.
	UserState	Gets the unique identifier for the asynchronous task. (Inherited from AsyncCompletedEventArgs .)

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ImportToysCompletedEventArgs Constructor

Overload List

	Name	Description
	ImportToysCompletedEventArgs(Exception, Boolean, Object)	Initialises a ImportToysCompletedEventArgs .
	ImportToysCompletedEventArgs(Exception, Boolean, Object, IToys)	Initialises a ImportToysCompletedEventArgs .

See Also

[ImportToysCompletedEventArgs Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ImportToysCompletedEventArgs Constructor (Exception, Boolean, Object)

Initialises a [ImportToysCompletedEventArgs](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public ImportToysCompletedEventArgs(  
    Exception error,  
    bool cancelled,  
    Object userState  
)
```

VB

```
Public Sub New (  
    error As Exception,  
    cancelled As Boolean,  
    userState As Object  
)
```

C++

```
public:  
ImportToysCompletedEventArgs(  
    Exception^ error,  
    bool cancelled,  
    Object^ userState  
)
```

F#

```
new :  
    error : Exception *  
    cancelled : bool *  
    userState : Object -> ImportToysCompletedEventArgs
```

Parameters

error

Type: [System.Exception](#)

An [Exception](#) for any exception that has been raised during importation.

cancelled

Type: [System.Boolean](#)

A value that indicates if the importation was cancelled.

userState

Type: [System.Object](#)

A Sandcastle Documented Class Library

A value for identifying an import request

See Also

[ImportToysCompletedEventArgs Class](#)

[ImportToysCompletedEventArgs Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ImportToysCompletedEventArgs Constructor (Exception, Boolean, Object, IToys)

Initialises a [ImportToysCompletedEventArgs](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public ImportToysCompletedEventArgs(  
    Exception error,  
    bool cancelled,  
    Object userState,  
    IToys result  
)
```

VB

```
Public Sub New (  
    error As Exception,  
    cancelled As Boolean,  
    userState As Object,  
    result As IToys  
)
```

C++

```
public:  
ImportToysCompletedEventArgs(  
    Exception^ error,  
    bool cancelled,  
    Object^ userState,  
    IToys^ result  
)
```

F#

```
new :  
    error : Exception *  
    cancelled : bool *  
    userState : Object *  
    result : IToys -> ImportToysCompletedEventArgs
```

Parameters

error

Type: [System.Exception](#)

An [Exception](#) for any exception that has been raised during importation.

cancelled

Type: [System.Boolean](#)

A Sandcastle Documented Class Library

A value that indicates if the importation was cancelled.

userState

Type: [System.Object](#)

A value for identifying an import request

result

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The result of importation (i.e. the toy data imported).

See Also

[ImportToysCompletedEventArgs Class](#)








[ImportToysCompletedEventArgs Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ImportToysCompletedEventArgs.ImportToysCompletedEventArgs Methods

The [ImportToysCompletedEventArgs](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	RaiseExceptionIfNecessary	Raises a user-supplied exception if an asynchronous operation failed. (Inherited from AsyncCompletedEventArgs .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

See Also





[ImportToysCompletedEventArgs Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ImportToysCompletedEventArgs.ImportToysCompletedEventArgs Properties

The [ImportToysCompletedEventArgs](#) type exposes the following members.

Properties

	Name	Description
	Cancelled	Gets a value indicating whether an asynchronous operation has been canceled. (Inherited from AsyncCompletedEventArgs .)
	Error	Gets a value indicating which error occurred during an asynchronous operation. (Inherited from AsyncCompletedEventArgs .)
	Result	Gets the result of an toy importation.
	UserState	Gets the unique identifier for the asynchronous task. (Inherited from AsyncCompletedEventArgs .)

See Also

[ImportToysCompletedEventArgs Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ImportToysCompletedEventArgs.Result Property

Gets the result of an toy importation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IToys Result { get; }
```

VB

```
Public ReadOnly Property Result As IToys  
    Get
```

C++

```
public:  
property IToys^ Result {  
    IToys^ get ();  
}
```

F#

```
member Result : IToys with get
```

Property Value

Type: [IToys](#)

See Also

[ImportToysCompletedEventArgs Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ImportToysCompletedEventHandler Delegate

Defines an delegate to handle the ImportToysCompleted event

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public delegate void ImportToysCompletedEventHandler(  
    Object sender,  
    AsyncCompletedEventArgs e  
)
```

VB

```
Public Delegate Sub ImportToysCompletedEventHandler (  
    sender As Object,  
    e As AsyncCompletedEventArgs  
)
```

C++

```
public delegate void ImportToysCompletedEventHandler(  
    Object^ sender,  
    AsyncCompletedEventArgs^ e  
)
```

F#

```
type ImportToysCompletedEventHandler =  
    delegate of  
        sender : Object *  
        e : AsyncCompletedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidCurrentCountException Class

Defines an exception that is raised when the value to be used for the current count of a [Toy](#) is invalid.

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

Woodstocks.WoodstocksIMS.Domain.InvalidCurrentCountException

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class InvalidCurrentCountException : Exception
```

VB

```
Public Class InvalidCurrentCountException  
    Inherits Exception
```

C++




```
public ref class InvalidCurrentCountException : public Exception
```

F#



```
type InvalidCurrentCountException =  
    class  
        inherit Exception  
    end
```







The **InvalidCurrentCountException** type exposes the following members.

Constructors









	Name	Description
	InvalidCurrentCountException()	Initialises a InvalidCurrentCountException .
	InvalidCurrentCountException(String)	Initialises a InvalidCurrentCountException .
	InvalidCurrentCountException(String, Exception)	Initialises a InvalidCurrentCountException .

Methods


	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations

		before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Events




	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidCurrentCountException Constructor

Overload List

	Name	Description
	InvalidCurrentCountException()	Initialises a InvalidCurrentCountException .
	InvalidCurrentCountException(String)	Initialises a InvalidCurrentCountException .
	InvalidCurrentCountException(String, Exception)	Initialises a InvalidCurrentCountException .

See Also

[InvalidCurrentCountException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidCurrentCountException Constructor

Initialises a [InvalidCurrentCountException](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public InvalidCurrentCountException()
```

VB

```
Public Sub New
```

C++

```
public:  
InvalidCurrentCountException()
```

F#

```
new : unit -> InvalidCurrentCountException
```

See Also

[InvalidCurrentCountException Class](#)

[InvalidCurrentCountException Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidCurrentCountException Constructor (String)

Initialises a [InvalidCurrentCountException](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public InvalidCurrentCountException(  
    string message  
)
```

VB

```
Public Sub New (  
    message As String  
)
```

C++

```
public:  
InvalidCurrentCountException(  
    String^ message  
)
```

F#

```
new :  
    message : string -> InvalidCurrentCountException
```

Parameters

message

Type: [System.String](#)

A message for the exception.

See Also

[InvalidCurrentCountException Class](#)

[InvalidCurrentCountException Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidCurrentCountException Constructor (String, Exception)

Initialises a [InvalidCurrentCountException](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public InvalidCurrentCountException(  
    string message,  
    Exception innerException  
)
```

VB

```
Public Sub New (  
    message As String,  
    innerException As Exception  
)
```

C++

```
public:  
InvalidCurrentCountException(  
    String^ message,  
    Exception^ innerException  
)
```

F#

```
new :  
    message : string *  
    innerException : Exception -> InvalidCurrentCountException
```

Parameters

message

Type: [System.String](#)

A message for the exception.

innerException

Type: [System.Exception](#)

See Also

[InvalidCurrentCountException Class](#)









[InvalidCurrentCountException Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidCurrentCountException.InvalidCurrentCountException Methods

The [InvalidCurrentCountException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

See Also









[InvalidCurrentCountException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidCurrentCountException.InvalidCurrentCountException Properties

The [InvalidCurrentCountException](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

See Also


[InvalidCurrentCountException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidCurrentCountException.InvalidCurrentCountException Events

The [InvalidCurrentCountException](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[InvalidCurrentCountException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidDescriptionException Class

An exception that is raised when the value for a description of a [Toy](#) is invalid.

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

Woodstocks.WoodstocksIMS.Domain.InvalidDescriptionException

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class InvalidDescriptionException : Exception
```

VB

```
Public Class InvalidDescriptionException  
    Inherits Exception
```

C++


```
public ref class InvalidDescriptionException : public Exception
```

F#




```
type InvalidDescriptionException =  
    class  
        inherit Exception  
    end
```






The **InvalidDescriptionException** type exposes the following members.

Constructors









	Name	Description
	InvalidDescriptionException	Initializes a new instance of the InvalidDescriptionException class

Methods


	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)

	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidDescriptionException Constructor

Initializes a new instance of the [InvalidDescriptionException](#) class

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public InvalidDescriptionException()
```

VB

```
Public Sub New
```

C++

```
public:  
InvalidDescriptionException()
```

F#

```
new : unit -> InvalidDescriptionException
```

See Also









[InvalidDescriptionException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidDescriptionException.InvalidDescriptionException Methods

The [InvalidDescriptionException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

See Also









[InvalidDescriptionException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidDescriptionException.InvalidDescriptionException Properties

The [InvalidDescriptionException](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

See Also


[InvalidDescriptionException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidDescriptionException.InvalidDescriptionException Events

The [InvalidDescriptionException](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[InvalidDescriptionException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidItemCodeException Class

Defines an exception that is raised when the item code for a [Toy](#) is invalid.

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

Woodstocks.WoodstocksIMS.Domain.InvalidItemCodeException

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class InvalidItemCodeException : Exception
```

VB

```
Public Class InvalidItemCodeException  
    Inherits Exception
```

C++


```
public ref class InvalidItemCodeException : public Exception
```

F#




```
type InvalidItemCodeException =  
    class  
        inherit Exception  
    end
```






The **InvalidItemCodeException** type exposes the following members.

Constructors









	Name	Description
	InvalidItemCodeException	Initializes a new instance of the InvalidItemCodeException class

Methods


	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)

	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidItemCodeException Constructor

Initializes a new instance of the [InvalidItemCodeException](#) class

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public InvalidItemCodeException()
```

VB

```
Public Sub New
```

C++

```
public:  
InvalidItemCodeException()
```

F#

```
new : unit -> InvalidItemCodeException
```

See Also









[InvalidItemCodeException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidItemCodeException.InvalidItemCodeException Methods

The [InvalidItemCodeException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

See Also









[InvalidItemCodeException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidItemCodeException.InvalidItemCodeException Properties

The [InvalidItemCodeException](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

See Also


[InvalidItemCodeException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidItemCodeException.InvalidItemCodeException Events

The [InvalidItemCodeException](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[InvalidItemCodeException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidOnOrderException Class

Defines an exception to be raised when an [OnOrder](#) value is invalid.

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

Woodstocks.WoodstocksIMS.Domain.InvalidOnOrderException

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class InvalidOnOrderException : Exception
```

VB

```
Public Class InvalidOnOrderException  
    Inherits Exception
```

C++


```
public ref class InvalidOnOrderException : public Exception
```

F#




```
type InvalidOnOrderException =  
    class  
        inherit Exception  
    end
```






The **InvalidOnOrderException** type exposes the following members.

Constructors









	Name	Description
	InvalidOnOrderException	Initialises an InvalidOnOrderException .

Methods


	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)

	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidOnOrderException Constructor

Initialises an [InvalidOnOrderException](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public InvalidOnOrderException()
```

VB

```
Public Sub New
```

C++

```
public:  
InvalidOnOrderException()
```

F#

```
new : unit -> InvalidOnOrderException
```

See Also









[InvalidOnOrderException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidOrderException.InvalidOrderException Methods

The [InvalidOrderException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

See Also









[InvalidOrderException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidOrderException.InvalidOrderException Properties

The [InvalidOrderException](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

See Also


[InvalidOrderException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

InvalidOnOrderException.InvalidOnOrderException Events

The [InvalidOnOrderException](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[InvalidOnOrderException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToy Interface

Defines the interface of a Toy.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public interface IToy : IComparable<IToy>,
    IComparable, IEquatable<IToy>
```

VB

```
Public Interface IToy
    Inherits IComparable(Of IToy), IComparable, IEquatable(Of IToy)
```

C++




```
public interface class IToy : IComparable<IToy^>,
    IComparable, IEquatable<IToy^>
```

F#


```
type IToy =
    interface
        interface IComparable<IToy>
        interface IComparable
        interface IEquatable<IToy>
    end
```

The **IToy** type exposes the following members.





Methods

	Name	Description
	CompareTo(Object)	Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object. (Inherited from IComparable .)
	CompareTo(T)	Compares the current object with another object of the same type. (Inherited from IComparable(IToy) .)
	Equals	Indicates whether the current object is equal to another object of the same type. (Inherited from IEquatable(IToy) .)


Properties

	Name	Description
	CurrentCount	Gets the current count for a IToy .

A Sandcastle Documented Class Library

	InitialCount	Gets the initial count for a IToy .
	ItemCode	Gets the item code for the IToy .
	ItemDescription	Gets the description of the IToy .
	OnOrderStatus	Gets the on order status of an Toy

Events

	Name	Description
	Changed	Event that is raised by a IToy when its count has changed.




See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToy.IToy Methods

The [IToy](#) type exposes the following members.

Methods

	Name	Description
	CompareTo(Object)	Compares the current instance with another object of the same type and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object. (Inherited from IComparable .)
	CompareTo(T)	Compares the current object with another object of the same type. (Inherited from IComparable(IToy) .)
	Equals	Indicates whether the current object is equal to another object of the same type. (Inherited from IEquatable(IToy) .)

See Also






[IToy Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToy.IToy Properties

The [IToy](#) type exposes the following members.

Properties

	Name	Description
	CurrentCount	Gets the current count for a IToy .
	InitialCount	Gets the initial count for a IToy .
	ItemCode	Gets the item code for the IToy .
	ItemDescription	Gets the description of the IToy .
	OnOrderStatus	Gets the on order status of an Toy

See Also

[IToy Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToy.CurrentCount Property

Gets the current count for a [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
string CurrentCount { get; set; }
```

VB

```
Property CurrentCount As String  
    Get  
    Set
```

C++

```
property String^ CurrentCount {  
    String^ get ();  
    void set (String^ value);  
}
```

F#

```
abstract CurrentCount : string with get, set
```

Property Value

Type: [String](#)

See Also

[IToy Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToy.InitialCount Property

Gets the initial count for a [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
string InitialCount { get; }
```

VB

```
ReadOnly Property InitialCount As String  
    Get
```

C++

```
property String^ InitialCount {  
    String^ get ();  
}
```

F#

```
abstract InitialCount : string with get
```

Property Value

Type: [String](#)

See Also

[IToy Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToy.ItemCode Property

Gets the item code for the [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
string ItemCode { get; }
```

VB

```
ReadOnly Property ItemCode As String  
    Get
```

C++

```
property String^ ItemCode {  
    String^ get ();  
}
```

F#

```
abstract ItemCode : string with get
```

Property Value

Type: [String](#)

See Also

[IToy Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToy.ItemDescription Property

Gets the description of the [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
string ItemDescription { get; }
```

VB

```
ReadOnly Property ItemDescription As String  
    Get
```

C++

```
property String^ ItemDescription {  
    String^ get ();  
}
```

F#

```
abstract ItemDescription : string with get
```

Property Value

Type: [String](#)

See Also

[IToy Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToy.OnOrderStatus Property

Gets the on order status of an [Toy](#)

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
string OnOrderStatus { get; }
```

VB

```
ReadOnly Property OnOrderStatus As String  
    Get
```

C++

```
property String^ OnOrderStatus {  
    String^ get ();  
}
```

F#

```
abstract OnOrderStatus : string with get
```

Property Value

Type: [String](#)

See Also


[IToy Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToy.IToy Events

The [IToy](#) type exposes the following members.

Events

	Name	Description
	Changed	Event that is raised by a IToy when its count has changed.

See Also

[IToy Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToy.Changed Event

Event that is raised by a [IToy](#) when its count has changed.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event ToyChangedEventHandler Changed
```

VB

```
Event Changed As ToyChangedEventHandler
```

C++

```
event ToyChangedEventHandler^ Changed {  
    void add (ToyChangedEventHandler^ value);  
    void remove (ToyChangedEventHandler^ value);  
}
```

F#

```
abstract Changed : IEvent<ToyChangedEventHandler,  
    EventArgs>
```

Value

Type: [Woodstocks.WoodstocksIMS.Domain.ToyChangedEventHandler](#)

See Also

[IToy Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys Interface

Defines an interface for a collection of [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public interface IToys : IList<IToy>,
    ICollection<IToy>, IEnumerable<IToy>, IList,
    ICollection, IEnumerable
```

VB

```
Public Interface IToys
    Inherits IList(Of IToy), ICollection(Of IToy),
    IEnumerable(Of IToy), IList, ICollection, IEnumerable
```

C++







```
public interface class IToys : IList<IToy^>,
    ICollection<IToy^>, IEnumerable<IToy^>, IList,
    ICollection, IEnumerable
```













F#

```
type IToys =
    interface
        interface IList<IToy>
        interface ICollection<IToy>
        interface IEnumerable<IToy>
        interface IList
        interface ICollection
        interface IEnumerable
    end
```







The **IToys** type exposes the following members.

Methods


	Name	Description
	Add(Object)	Adds an item to the IList . (Inherited from IList .)
	Add(T)	Adds an item to the ICollection(T) . (Inherited from ICollection(IToy) .)
	Clear	Removes all IToy from the collection.
	Contains(Object)	Determines whether the IList contains a specific value. (Inherited from IList .)
	Contains(T)	Determines whether the ICollection(T) contains a specific value. (Inherited from ICollection(IToy) .)
	CopyTo(Array, Int32)	Copies the elements of the ICollection to an Array , starting at a particular Array index. (Inherited from ICollection .)

	CopyTo(T[], Int32)	Copies the elements of the ICollection(T) to an Array , starting at a particular Array index. (Inherited from ICollection(IToy) .)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable(IToy) .)
	IndexOf(Object)	Determines the index of a specific item in the IList . (Inherited from IList .)
	IndexOf(T)	Determines the index of a specific item in the IList(T) . (Inherited from IList(IToy) .)
	Insert(Int32, T)	Inserts an item to the IList(T) at the specified index. (Inherited from IList(IToy) .)
	Insert(Int32, Object)	Inserts an item to the IList at the specified index. (Inherited from IList .)
	Remove(Object)	Removes the first occurrence of a specific object from the IList . (Inherited from IList .)
	Remove(T)	Removes the first occurrence of a specific object from the ICollection(T) . (Inherited from ICollection(IToy) .)
	RemoveAt	Removes the IToy from the collection located at index.
	SortByCurrentCount	Sorts the collection of IToy by current count in the specified sort order.
	SortByItemCode	Sorts the collection of IToy by item code in the specified sort order..
	SortByOnOrder	Sorts the collection of IToy by on order status in the specified sort order.

Properties

	Name	Description
	Count	The total number of IToys in the collection.
	IsFixedSize	Gets a value indicating whether the IList has a fixed size. (Inherited from IList .)
	IsReadOnly	Gets a value indicating whether the ICollection(T) is read-only. (Inherited from ICollection(IToy) .)
	IsSynchronized	Gets a value indicating whether access to the ICollection is synchronized (thread safe). (Inherited from ICollection .)
	Item	Gets the IToy located at the indexed position specified by index.
	SyncRoot	Gets an object that can be used to synchronize access to the ICollection . (Inherited from ICollection .)

Events

	Name	Description
	ToyChanged	Event raised when an item in the collection changes.


See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.IToys Methods

The [IToys](#) type exposes the following members.

Methods

	Name	Description
	Add(Object)	Adds an item to the IList . (Inherited from IList .)
	Add(T)	Adds an item to the ICollection(T) . (Inherited from ICollection(IToy) .)
	Clear	Removes all IToy from the collection.
	Contains(Object)	Determines whether the IList contains a specific value. (Inherited from IList .)
	Contains(T)	Determines whether the ICollection(T) contains a specific value. (Inherited from ICollection(IToy) .)
	CopyTo(Array, Int32)	Copies the elements of the ICollection to an Array , starting at a particular Array index. (Inherited from ICollection .)
	CopyTo(T[], Int32)	Copies the elements of the ICollection(T) to an Array , starting at a particular Array index. (Inherited from ICollection(IToy) .)
	GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable(IToy) .)
	IndexOf(Object)	Determines the index of a specific item in the IList . (Inherited from IList .)
	IndexOf(T)	Determines the index of a specific item in the IList(T) . (Inherited from IList(IToy) .)
	Insert(Int32, T)	Inserts an item to the IList(T) at the specified index. (Inherited from IList(IToy) .)
	Insert(Int32, Object)	Inserts an item to the IList at the specified index. (Inherited from IList .)
	Remove(Object)	Removes the first occurrence of a specific object from the IList . (Inherited from IList .)
	Remove(T)	Removes the first occurrence of a specific object from the ICollection(T) . (Inherited from ICollection(IToy) .)
	RemoveAt	Removes the IToy from the collection located at index.
	SortByCurrentCount	Sorts the collection of IToy by current count in the specified sort order.
	SortByItemCode	Sorts the collection of IToy by item code in the specified sort order..
	SortByOnOrder	Sorts the collection of IToy by on order status in the specified sort order.

See Also

[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.Clear Method

Removes all [IToy](#) from the collection.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void Clear()
```

VB

```
Sub Clear
```

C++

```
void Clear()
```

F#

```
abstract Clear : unit -> unit
```

Implements

[ICollection\(T\).Clear\(\)](#)

[IList.Clear\(\)](#)

See Also

[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.RemoveAt Method

Removes the [IToy](#) from the collection located at index.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void RemoveAt (
    int index
)
```

VB

```
Sub RemoveAt (
    index As Integer
)
```

C++

```
void RemoveAt (
    int index
)
```

F#

```
abstract RemoveAt :
    index : int -> unit
```

Parameters

index

Type: [System.Int32](#)

The position within the collection of the [IToy](#) to be removed from the collection.

Implements

[IList\(T\).RemoveAt\(Int32\)](#)

[IList.RemoveAt\(Int32\)](#)

See Also

[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.SortByCurrentCount Method

Sorts the collection of [IToy](#) by current count in the specified sort order.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void SortByCurrentCount(  
    SortOrder order  
)
```

VB

```
Sub SortByCurrentCount (  
    order As SortOrder  
)
```

C++

```
void SortByCurrentCount(  
    SortOrder order  
)
```

F#

```
abstract SortByCurrentCount :  
    order : SortOrder -> unit
```

Parameters

order

Type: [Woodstocks.WoodstocksIMS.Domain.SortOrder](#)

The sort order.

See Also

[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.SortByItemCode Method

Sorts the collection of [IToy](#) by item code in the specified sort order..

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void SortByItemCode(  
    SortOrder order  
)
```

VB

```
Sub SortByItemCode (  
    order As SortOrder  
)
```

C++

```
void SortByItemCode(  
    SortOrder order  
)
```

F#

```
abstract SortByItemCode :  
    order : SortOrder -> unit
```

Parameters

order

Type: [Woodstocks.WoodstocksIMS.Domain.SortOrder](#)

The sort order.

See Also

[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.SortByOnOrder Method

Sorts the collection of [IToy](#) by on order status in the specified sort order.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void SortByOnOrder(  
    SortOrder order  
)
```

VB

```
Sub SortByOnOrder (  
    order As SortOrder  
)
```

C++

```
void SortByOnOrder(  
    SortOrder order  
)
```

F#

```
abstract SortByOnOrder :  
    order : SortOrder -> unit
```

Parameters

order

Type: [Woodstocks.WoodstocksIMS.Domain.SortOrder](#)

The sort order.

See Also







[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.IToys Properties

The [IToys](#) type exposes the following members.

Properties

	Name	Description
	Count	The total number of IToys in the collection.
	IsFixedSize	Gets a value indicating whether the IList has a fixed size. (Inherited from IList .)
	IsReadOnly	Gets a value indicating whether the ICollection(T) is read-only. (Inherited from ICollection(IToy) .)
	IsSynchronized	Gets a value indicating whether access to the ICollection is synchronized (thread safe). (Inherited from ICollection .)
	Item	Gets the IToy located at the indexed position specified by index.
	SyncRoot	Gets an object that can be used to synchronize access to the ICollection . (Inherited from ICollection .)

See Also

[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.Count Property

The total number of [IToys](#) in the collection.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
ReadOnly Property Count As Integer  
    Get
```

C++

```
property int Count {  
    int get ();  
}
```

F#

```
abstract Count : int with get
```

Property Value

Type: [Int32](#)

Implements

[ICollection\(T\).Count](#)

[ICollection.Count](#)

See Also

[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.Item Property

Gets the [IToy](#) located at the indexed position specified by index.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
IToy this[  
    int index  
] { get; }
```

VB

```
ReadOnly Default Property Item (  
    index As Integer  
) As IToy  
    Get
```

C++

```
property IToy^ default[int index] {  
    IToy^ get (int index);  
}
```

F#

```
abstract Item : IToy with get
```

Parameters

index

Type: [System.Int32](#)

The zero-based index to the collection for the position of the [IToy](#) to be retrieved from the collection.

Return Value

Type: [IToy](#)

The [IToy](#)

Implements

[IList\(T\).Item\(Int32\)](#)

See Also


[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.IToys Events

The [IToys](#) type exposes the following members.

Events

	Name	Description
	ToyChanged	Event raised when an item in the collection changes.

See Also

[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IToys.ToyChanged Event

Event raised when an item in the collection changes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event ToyChangedEventHandler ToyChanged
```

VB

```
Event ToyChanged As ToyChangedEventHandler
```

C++

```
event ToyChangedEventHandler^ ToyChanged {  
    void add (ToyChangedEventHandler^ value);  
    void remove (ToyChangedEventHandler^ value);  
}
```

F#

```
abstract ToyChanged : IEvent<ToyChangedEventHandler,  
    EventArgs>
```

Value

Type: [Woodstocks.WoodstocksIMS.Domain.ToyChangedEventHandler](#)

See Also

[IToys Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS Interface

Defines an interface to the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public interface IWoodstocksIMS : IWoodstocksIMSCient
```

VB

```
Public Interface IWoodstocksIMS
    Inherits IWoodstocksIMSCient
```

C++










```
public interface class IWoodstocksIMS : IWoodstocksIMSCient
```

















F#

```
type IWoodstocksIMS =
    interface
        interface IWoodstocksIMSCient
    end
```




The **IWoodstocksIMS** type exposes the following members.

Methods



	Name	Description
	CancelAsync	Cancels an asynchronous operation. (Inherited from IWoodstocksIMSCient .)
	DiscardImportedToyData()	Discards imported toy data from the system. (Inherited from IWoodstocksIMSCient .)
	DiscardImportedToyData(Boolean)	Discards imported data from the system. This method is defined as a system facing interface method.
	ExportToysAsync	Exports modified toy data from the system. (Inherited from IWoodstocksIMSCient .)
	GetExportingState	Get the Exporting state of the system
	GetIdleState	Get the Idle state of the system.
	GetImportingState	Get the Importing state of the system.
	GetModifiedToys	Gets the toy data that has been modified and has not been saved.
	GetToyDataSource	Gets the source from which the system will, or has, imported toy data.


	GetToyExporter	Gets the toy exporter used for exporting data.
	GetToyImporter	Gets the toy importer of the system.
	GetToys	Gets the toy data currently imported into the WoodstocksIMS .
	GetUnsavedChanges	Gets whether the IWoodstocksIMS has imported stock data that has been modified but has not been saved.
	ImportToysAsync	Imports toy data into the WoodstocksIMS for use by the system. (Inherited from IWoodstocksIMSClient .)
	IsBusy	Indicates if the WoodstocksIMS is busy carrying out an asynchronous operation. (Inherited from IWoodstocksIMSClient .)
	OnExportCompleted	Raises the ExportCompleted event of the IWoodstocksIMS .
	OnImportCompleted	Raises the ImportCompleted event of the IWoodstocksIMS .
	OnProgressChanged	Raises the ProgressChanged event of the IWoodstocksIMS to indicate that progress of an asynchronous operation.
	SetModifiedToys	Sets the toy data that has been modified and has not been saved.
	SetState	Set the current state of the system.
	SetToyDataSource	Sets the source from which the system will, or has, imported toy data.
	SetToyExporter	Sets the toy exporter used for exporting data.
	SetToyImporter	Sets the toy importer of the system.
	SetToys	Sets the toy data in use by the WoodstocksIMS .
	UnsavedChanges	Gets whether the toy data contains unsaved changes. (Inherited from IWoodstocksIMSClient .)

Properties

	Name	Description
	ToyDataSource	Gets and Sets the data source from which toy data should be imported. (Inherited from IWoodstocksIMSClient .)
	ToyImporter	Gets and Sets the IWoodstocksToyImporter to be used by the system to import toy data.
	Toys	Retrieves IToys which references Wood Stocks toy data after importation. (Inherited from IWoodstocksIMSClient .)

Events

	Name	Description
	ExportCompleted	Raised when an asynchronous export completes. (Inherited from IWoodstocksIMSClient .)
	ImportCompleted	Raised when an asynchronous import completes. (Inherited from

		IWoodstocksIMSClient.)
	ProgressChanged	Raised when progress on an asynchronous operation is made. (Inherited from IWoodstocksIMSClient.)























See Also




[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.IWoodstocksIMS Methods

The [IWoodstocksIMS](#) type exposes the following members.

Methods

	Name	Description
	CancelAsync	Cancels an asynchronous operation. (Inherited from IWoodstocksIMSCient.)
	DiscardImportedToyData()	Discards imported toy data from the system. (Inherited from IWoodstocksIMSCient.)
	DiscardImportedToyData(Boolean)	Discards imported data from the system. This method is defined as a system facing interface method.
	ExportToysAsync	Exports modified toy data from the system. (Inherited from IWoodstocksIMSCient.)
	GetExportingState	Get the Exporting state of the system
	GetIdleState	Get the Idle state of the system.
	GetImportingState	Get the Importing state of the system.
	GetModifiedToys	Gets the toy data that has been modified and has not been saved.
	GetToyDataSource	Gets the source from which the system will, or has, imported toy data.
	GetToyExporter	Gets the toy exporter used for exporting data.
	GetToyImporter	Gets the toy importer of the system.
	GetToys	Gets the toy data currently imported into the WoodstocksIMS .
	GetUnsavedChanges	Gets whether the IWoodstocksIMS has imported stock data that has been modified but has not been saved.
	ImportToysAsync	Imports toy data into the WoodstocksIMS for use by the system. (Inherited from IWoodstocksIMSCient.)
	IsBusy	Indicates if the WoodstocksIMS is busy carrying out an asynchronous operation. (Inherited from IWoodstocksIMSCient.)
	OnExportCompleted	Raises the ExportCompleted event of the IWoodstocksIMS .
	OnImportCompleted	Raises the ImportCompleted event of the IWoodstocksIMS .
	OnProgressChanged	Raises the ProgressChanged event of the IWoodstocksIMS to indicate that progress of an asynchronous operation.
	SetModifiedToys	Sets the toy data that has been modified and has not been saved.
	SetState	Set the current state of the system.
	SetToyDataSource	Sets the source from which the system will, or has, imported toy data.
	SetToyExporter	Sets the toy exporter used for exporting data.

	SetToyImporter	Sets the toy importer of the system.
	SetToys	Sets the toy data in use by the WoodstocksIMS .
	UnsavedChanges	Gets whether the toy data contains unsaved changes. (Inherited from IWoodstocksIMSClient .)



See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.DiscardImportedToyData Method

Overload List

	Name	Description
	DiscardImportedToyData()	Discards imported toy data from the system. (Inherited from IWoodstocksIMSClient .)
	DiscardImportedToyData(Boolean)	Discards imported data from the system. This method is defined as a system facing interface method.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.DiscardImportedToyData Method (Boolean)

Discards imported data from the system. This method is defined as a system facing interface method.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void DiscardImportedToyData(  
    bool disposing  
)
```

VB

```
Sub DiscardImportedToyData (  
    disposing As Boolean  
)
```

C++

```
void DiscardImportedToyData(  
    bool disposing  
)
```

F#

```
abstract DiscardImportedToyData :  
    disposing : bool -> unit
```

Parameters

disposing

Type: [System.Boolean](#)

Indicates that the data should actually be disposed of.

See Also

[IWoodstocksIMS Interface](#)

[DiscardImportedToyData Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.GetExportingState Method

Get the Exporting state of the system

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
WoodstocksIMSState GetExportingState()
```

VB

```
Function GetExportingState As WoodstocksIMSState
```

C++

```
WoodstocksIMSState^ GetExportingState()
```

F#

```
abstract GetExportingState : unit -> WoodstocksIMSState
```

Return Value

Type: [WoodstocksIMSState](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.IWoodstocksIMS.GetExportingState"]

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.GetIdleState Method

Get the Idle state of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
WoodstocksIMSState GetIdleState()
```

VB

```
Function GetIdleState As WoodstocksIMSState
```

C++

```
WoodstocksIMSState^ GetIdleState()
```

F#

```
abstract GetIdleState : unit -> WoodstocksIMSState
```

Return Value

Type: [WoodstocksIMSState](#)

The idle state of the system.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.GetImportingState Method

Get the Importing state of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
WoodstocksIMSState GetImportingState()
```

VB

```
Function GetImportingState As WoodstocksIMSState
```

C++

```
WoodstocksIMSState^ GetImportingState()
```

F#

```
abstract GetImportingState : unit -> WoodstocksIMSState
```

Return Value

Type: [WoodstocksIMSState](#)

The Importing state of the system

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.GetModifiedToys Method

Gets the toy data that has been modified and has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
IToys GetModifiedToys()
```

VB

```
Function GetModifiedToys As IToys
```

C++

```
IToys^ GetModifiedToys()
```

F#

```
abstract GetModifiedToys : unit -> IToys
```

Return Value

Type: [IToys](#)

A reference to the modified toy data.

Remarks

This method is intended as a system method and therefore does not appear on the client interface.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.GetToyDataSource Method

Gets the source from which the system will, or has, imported toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
string GetToyDataSource()
```

VB

```
Function GetToyDataSource As String
```

C++

```
String^ GetToyDataSource()
```

F#

```
abstract GetToyDataSource : unit -> string
```

Return Value

Type: [String](#)

The source of the toy data.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.GetToyExporter Method

Gets the toy exporter used for exporting data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
IWoodstocksToyExporter GetToyExporter()
```

VB

```
Function GetToyExporter As IWoodstocksToyExporter
```

C++

```
IWoodstocksToyExporter^ GetToyExporter()
```

F#

```
abstract GetToyExporter : unit -> IWoodstocksToyExporter
```

Return Value

Type: [IWoodstocksToyExporter](#)

[Missing <returns> documentation for "M:Woodstocks.WoodstocksIMS.Domain.IWoodstocksIMS.GetToyExporter"]

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.GetToyImporter Method

Gets the toy importer of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
IWoodstocksToyImporter GetToyImporter()
```

VB

```
Function GetToyImporter As IWoodstocksToyImporter
```

C++

```
IWoodstocksToyImporter^ GetToyImporter()
```

F#

```
abstract GetToyImporter : unit -> IWoodstocksToyImporter
```

Return Value

Type: [IWoodstocksToyImporter](#)

The systems toy importer.

Remarks

This method is to be implemented to provide for an implementation of the ToyImporter property.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.GetToys Method

Gets the toy data currently imported into the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
IToys GetToys ( )
```

VB

```
Function GetToys As IToys
```

C++

```
IToys^ GetToys ( )
```

F#

```
abstract GetToys : unit -> IToys
```

Return Value

Type: [IToys](#)

A reference to the imported toy data

Remarks

This method is intended as a system method and therefore does not appear on the client interface.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.GetUnsavedChanges Method

Gets whether the [IWoodstocksIMS](#) has imported stock data that has been modified but has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
bool GetUnsavedChanges()
```

VB

```
Function GetUnsavedChanges As Boolean
```

C++

```
bool GetUnsavedChanges()
```

F#

```
abstract GetUnsavedChanges : unit -> bool
```

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.IWoodstocksIMS.GetUnsavedChanges"]

Remarks

This method is to be implemented for "internal system" use. It exists, primarily, to allow state objects to return the result from the system to clients who have called the UnsavedChanges method on the client interface.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.OnExportCompleted Method

Raises the [ExportCompleted](#) event of the [IWoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void OnExportCompleted(  
    Object sender,  
    AsyncCompletedEventArgs e  
)
```

VB

```
Sub OnExportCompleted (  
    sender As Object,  
    e As AsyncCompletedEventArgs  
)
```

C++

```
void OnExportCompleted(  
    Object^ sender,  
    AsyncCompletedEventArgs^ e  
)
```

F#

```
abstract OnExportCompleted :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The [IWoodstocksIMS](#) that raised the event.

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

The event data for the [ExportCompleted](#) event.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.OnImportCompleted Method

Raises the [ImportCompleted](#) event of the [IWoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void OnImportCompleted(  
    Object sender,  
    AsyncCompletedEventArgs e  
)
```

VB

```
Sub OnImportCompleted (  
    sender As Object,  
    e As AsyncCompletedEventArgs  
)
```

C++

```
void OnImportCompleted(  
    Object^ sender,  
    AsyncCompletedEventArgs^ e  
)
```

F#

```
abstract OnImportCompleted :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The [IWoodstocksIMS](#) that raised the event.

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

The event data for the [ImportCompleted](#) event.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.OnProgressChanged Method

Raises the [ProgressChanged](#) event of the [IWoodstocksIMS](#) to indicate that progress of an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void OnProgressChanged(  
    Object sender,  
    ProgressChangedEventArgs e  
)
```

VB

```
Sub OnProgressChanged (  
    sender As Object,  
    e As ProgressChangedEventArgs  
)
```

C++

```
void OnProgressChanged(  
    Object^ sender,  
    ProgressChangedEventArgs^ e  
)
```

F#

```
abstract OnProgressChanged :  
    sender : Object *  
    e : ProgressChangedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The [IWoodstocksIMS](#) that raised the event.

e

Type: [System.ComponentModel.ProgressChangedEventArgs](#)

The event data for the [ProgressChanged](#) event.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.SetModifiedToys Method

Sets the toy data that has been modified and has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void SetModifiedToys(  
    IToys modified  
)
```

VB

```
Sub SetModifiedToys (  
    modified As IToys  
)
```

C++

```
void SetModifiedToys(  
    IToys^ modified  
)
```

F#

```
abstract SetModifiedToys :  
    modified : IToys -> unit
```

Parameters

modified

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The modified toy data.

Return Value

Type:

A reference to the modified toy data.

Remarks

This method is intended as a system method and therefore does not appear on the client interface.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.SetState Method

Set the current state of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void SetState(  
    WoodstocksIMSState newState  
)
```

VB

```
Sub SetState (  
    newState As WoodstocksIMSState  
)
```

C++

```
void SetState(  
    WoodstocksIMSState^ newState  
)
```

F#

```
abstract SetState :  
    newState : WoodstocksIMSState -> unit
```

Parameters

newState

Type: [Woodstocks.WoodstocksIMS.Domain.WoodstocksIMSState](#)

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.SetToyDataSource Method

Sets the source from which the system will, or has, imported toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void SetToyDataSource(  
    string source  
)
```

VB

```
Sub SetToyDataSource (  
    source As String  
)
```

C++

```
void SetToyDataSource(  
    String^ source  
)
```

F#

```
abstract SetToyDataSource :  
    source : string -> unit
```

Parameters

source

Type: [System.String](#)

The source of the toy data.

Remarks

This method is intended as a system method and therefore does not appear on the client interface.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.SetToyExporter Method

Sets the toy exporter used for exporting data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void SetToyExporter(  
    IWoodstocksToyExporter toyExporter  
)
```

VB

```
Sub SetToyExporter (  
    toyExporter As IWoodstocksToyExporter  
)
```

C++

```
void SetToyExporter(  
    IWoodstocksToyExporter^ toyExporter  
)
```

F#

```
abstract SetToyExporter :  
    toyExporter : IWoodstocksToyExporter -> unit
```

Parameters

toyExporter

Type: [Woodstocks.WoodstocksIMS.Domain.IWoodstocksToyExporter](#)

The exporter that the system should use to export toy data.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.SetToyImporter Method

Sets the toy importer of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void SetToyImporter(  
    IWoodstocksToyImporter toyImporter  
)
```

VB

```
Sub SetToyImporter (  
    toyImporter As IWoodstocksToyImporter  
)
```

C++

```
void SetToyImporter(  
    IWoodstocksToyImporter^ toyImporter  
)
```

F#

```
abstract SetToyImporter :  
    toyImporter : IWoodstocksToyImporter -> unit
```

Parameters

toyImporter

Type: [Woodstocks.WoodstocksIMS.Domain.IWoodstocksToyImporter](#)

The importer that the system should be set to use.

Remarks

This method is to be implemented to provide for an implementation of the ToyImporter property.

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.SetToys Method

Sets the toy data in use by the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void SetToys(  
    IToys toys  
)
```

VB

```
Sub SetToys (  
    toys As IToys  
)
```

C++

```
void SetToys(  
    IToys^ toys  
)
```

F#

```
abstract SetToys :  
    toys : IToys -> unit
```

Parameters

toys

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The toy data to be used by the system.

See Also




[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.IWoodstocksIMS Properties

The [IWoodstocksIMS](#) type exposes the following members.

Properties

	Name	Description
	ToyDataSource	Gets and Sets the data source from which toy data should be imported. (Inherited from IWoodstocksIMSClient .)
	ToyImporter	Gets and Sets the IWoodstocksToyImporter to be used by the system to import toy data.
	Toys	Retrieves IToys which references Wood Stocks toy data after importation. (Inherited from IWoodstocksIMSClient .)

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.ToyImporter Property

Gets and Sets the [IWoodstocksToyImporter](#) to be used by the system to import toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
IWoodstocksToyImporter ToyImporter { get; set; }
```

VB

```
Property ToyImporter As IWoodstocksToyImporter  
    Get  
    Set
```

C++

```
property IWoodstocksToyImporter^ ToyImporter {  
    IWoodstocksToyImporter^ get ();  
    void set (IWoodstocksToyImporter^ value);  
}
```

F#

```
abstract ToyImporter : IWoodstocksToyImporter with get, set
```

Property Value

Type: [IWoodstocksToyImporter](#)

Remarks

This property is intended as a system property that that is unavailable to clients of the system. As a result it does not appear on the client interface.

See Also




[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMS.IWoodstocksIMS Events

The [IWoodstocksIMS](#) type exposes the following members.

Events

	Name	Description
	ExportCompleted	Raised when an asynchronous export completes. (Inherited from IWoodstocksIMSClient .)
	ImportCompleted	Raised when an asynchronous import completes. (Inherited from IWoodstocksIMSClient .)
	ProgressChanged	Raised when progress on an asynchronous operation is made. (Inherited from IWoodstocksIMSClient .)

See Also

[IWoodstocksIMS Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient Interface

Defines a client interface for the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public interface IWoodstocksIMSClient
```

VB

```
Public Interface IWoodstocksIMSClient
```

C++







```
public interface class IWoodstocksIMSClient
```

F#



```
type IWoodstocksIMSClient = interface end
```

The **IWoodstocksIMSClient** type exposes the following members.


Methods



	Name	Description
	CancelAsync	Cancels an asynchronous operation.
	DiscardImportedToyData	Discards imported toy data from the system.
	ExportToysAsync	Exports modified toy data from the system.
	ImportToysAsync	Imports toy data into the WoodstocksIMS for use by the system.
	IsBusy	Indicates if the WoodstocksIMS is busy carrying out an asynchronous operation.
	UnsavedChanges	Gets whether the toy data contains unsaved changes.

Properties

	Name	Description
	ToyDataSource	Gets and Sets the data source from which toy data should be imported.
	Toys	Retrieves IToys which references Wood Stocks toy data after importation.

Events

	Name	Description
	ExportCompleted	Raised when an asynchronous export completes.

	ImportCompleted	Raised when an asynchronous import completes.
	ProgressChanged	Raised when progress on an asynchronous operation is made.







See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.IWoodstocksIMSClient Methods

The [IWoodstocksIMSClient](#) type exposes the following members.

Methods

	Name	Description
	CancelAsync	Cancels an asynchronous operation.
	DiscardImportedToyData	Discards imported toy data from the system.
	ExportToysAsync	Exports modified toy data from the system.
	ImportToysAsync	Imports toy data into the WoodstocksIMS for use by the system.
	IsBusy	Indicates if the WoodstocksIMS is busy carrying out an asynchronous operation.
	UnsavedChanges	Gets whether the toy data contains unsaved changes.

See Also

[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.CancelAsync Method

Cancels an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void CancelAsync()
```

VB

```
Sub CancelAsync
```

C++

```
void CancelAsync()
```

F#

```
abstract CancelAsync : unit -> unit
```

See Also

[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.DiscardImportedToyData Method

Discards imported toy data from the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void DiscardImportedToyData()
```

VB

```
Sub DiscardImportedToyData
```

C++

```
void DiscardImportedToyData()
```

F#

```
abstract DiscardImportedToyData : unit -> unit
```

See Also

[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.ExportToysAsync Method

Exports modified toy data from the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void ExportToysAsync( )
```

VB

```
Sub ExportToysAsync
```

C++

```
void ExportToysAsync( )
```

F#

```
abstract ExportToysAsync : unit -> unit
```

See Also

[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.ImportToysAsync Method

Imports toy data into the [WoodstocksIMS](#) for use by the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void ImportToysAsync(  
    string source  
)
```

VB

```
Sub ImportToysAsync (  
    source As String  
)
```

C++

```
void ImportToysAsync(  
    String^ source  
)
```

F#

```
abstract ImportToysAsync :  
    source : string -> unit
```

Parameters

source

Type: [System.String](#)

The source from which toy data should be retrieved.

Remarks

The source is the filepath to the csv data file containing the Wood Stocks toy data.

See Also

[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSCient.IsBusy Method

Indicates if the [WoodstocksIMS](#) is busy carrying out an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
bool IsBusy()
```

VB

```
Function IsBusy As Boolean
```

C++

```
bool IsBusy()
```

F#

```
abstract IsBusy : unit -> bool
```

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.IWoodstocksIMSCient.IsBusy"]

See Also

[IWoodstocksIMSCient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.UnsavedChanges Method

Gets whether the toy data contains unsaved changes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
bool UnsavedChanges()
```

VB

```
Function UnsavedChanges As Boolean
```

C++

```
bool UnsavedChanges()
```

F#

```
abstract UnsavedChanges : unit -> bool
```

Return Value

Type: [Boolean](#)

True if the toy data contains unsaved changes, false if it does not.

See Also



[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.IWoodstocksIMSClient Properties

The [IWoodstocksIMSClient](#) type exposes the following members.

Properties

	Name	Description
	ToyDataSource	Gets and Sets the data source from which toy data should be imported.
	Toys	Retrieves IToys which references Wood Stocks toy data after importation.

See Also

[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.ToyDataSource Property

Gets and Sets the data source from which toy data should be imported.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
string ToyDataSource { get; set; }
```

VB

```
Property ToyDataSource As String  
    Get  
    Set
```

C++

```
property String^ ToyDataSource {  
    String^ get ();  
    void set (String^ value);  
}
```

F#

```
abstract ToyDataSource : string with get, set
```

Property Value

Type: [String](#)

See Also

[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.Toys Property

Retrieves [IToys](#) which references Wood Stocks toy data after importation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
IToys Toys { get; }
```

VB

```
ReadOnly Property Toys As IToys  
    Get
```

C++

```
property IToys^ Toys {  
    IToys^ get ();  
}
```

F#

```
abstract Toys : IToys with get
```

Return Value

Type: [IToys](#)

See Also




[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSCClient.IWoodstocksIMSCClient Events

The [IWoodstocksIMSCClient](#) type exposes the following members.

Events

	Name	Description
	ExportCompleted	Raised when an asynchronous export completes.
	ImportCompleted	Raised when an asynchronous import completes.
	ProgressChanged	Raised when progress on an asynchronous operation is made.

See Also

[IWoodstocksIMSCClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.ExportCompleted Event

Raised when an asynchronous export completes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event AsyncCompletedEventHandler ExportCompleted
```

VB

```
Event ExportCompleted As AsyncCompletedEventHandler
```

C++

```
event AsyncCompletedEventHandler^ ExportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

See Also

[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.ImportCompleted Event

Raised when an asynchronous import completes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event AsyncCompletedEventHandler ImportCompleted
```

VB

```
Event ImportCompleted As AsyncCompletedEventHandler
```

C++

```
event AsyncCompletedEventHandler^ ImportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ImportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

See Also

[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksIMSClient.ProgressChanged Event

Raised when progress on an asynchronous operation is made.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event ProgressChangedEventHandler ProgressChanged
```

VB

```
Event ProgressChanged As ProgressChangedEventHandler
```

C++

```
event ProgressChangedEventHandler^ ProgressChanged {  
    void add (ProgressChangedEventHandler^ value);  
    void remove (ProgressChangedEventHandler^ value);  
}
```

F#

```
abstract ProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>
```

Value

Type: [System.ComponentModel.ProgressChangedEventHandler](#)

See Also

[IWoodstocksIMSClient Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyExporter Interface

Defines methods for an exporter to export toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public interface IWoodstocksToyExporter
```

VB

```
Public Interface IWoodstocksToyExporter
```

C++




```
public interface class IWoodstocksToyExporter
```

F#


```
type IWoodstocksToyExporter = interface end
```

The **IWoodstocksToyExporter** type exposes the following members.



Methods

	Name	Description
	Close	Closes the exporter.
	ExportAsync	Exports toy data to the specified destination.
	ExportCancel	Cancels an asynchronous exportation of toy data.

Properties

	Name	Description
	IsBusy	Indicates if the exporter is busy carrying out an exportation.

Events

	Name	Description
	ExportCompleted	Event that is raised upon completion of exportation.
	ExportProgressChanged	Event that is raised upon progress of exportation.




See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyExporter.IWoodstocksToyExporter Methods

The [IWoodstocksToyExporter](#) type exposes the following members.

Methods

	Name	Description
	Close	Closes the exporter.
	ExportAsync	Exports toy data to the specified destination.
	ExportCancel	Cancels an asynchronous exportation of toy data.

See Also

[IWoodstocksToyExporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyExporter.Close Method

Closes the exporter.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void Close()
```

VB

```
Sub Close
```

C++

```
void Close()
```

F#

```
abstract Close : unit -> unit
```

See Also

[IWoodstocksToyExporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyExporter.ExportAsync Method

Exports toy data to the specified destination.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void ExportAsync(  
    string filename,  
    IToys data  
)
```

VB

```
Sub ExportAsync (  
    filename As String,  
    data As IToys  
)
```

C++

```
void ExportAsync(  
    String^ filename,  
    IToys^ data  
)
```

F#

```
abstract ExportAsync :  
    filename : string *  
    data : IToys -> unit
```

Parameters

filename

Type: [System.String](#)

The file name, including the path, of the file.

data

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The toy data to be exported.

See Also

[IWoodstocksToyExporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyExporter.ExportCancel Method

Cancels an asynchronous exportation of toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void ExportCancel()
```

VB

```
Sub ExportCancel
```

C++

```
void ExportCancel()
```

F#

```
abstract ExportCancel : unit -> unit
```

See Also


[IWoodstocksToyExporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyExporter.IWoodstocksToyExporter Properties

The [IWoodstocksToyExporter](#) type exposes the following members.

Properties

	Name	Description
	IsBusy	Indicates if the exporter is busy carrying out an exportation.

See Also

[IWoodstocksToyExporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyExporter.IsBusy Property

Indicates if the exporter is busy carrying out an exportation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
bool IsBusy { get; }
```

VB

```
ReadOnly Property IsBusy As Boolean  
    Get
```

C++

```
property bool IsBusy {  
    bool get ();  
}
```

F#

```
abstract IsBusy : bool with get
```

Return Value

Type: [Boolean](#)

See Also



[IWoodstocksToyExporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyExporter.IWoodstocksToyExporter Events

The [IWoodstocksToyExporter](#) type exposes the following members.

Events

	Name	Description
	ExportCompleted	Event that is raised upon completion of exportation.
	ExportProgressChanged	Event that is raised upon progress of exportation.

See Also

[IWoodstocksToyExporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyExporter.ExportCompleted Event

Event that is raised upon completion of exportation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event AsyncCompletedEventHandler ExportCompleted
```

VB

```
Event ExportCompleted As AsyncCompletedEventHandler
```

C++

```
event AsyncCompletedEventHandler^ ExportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

See Also

[IWoodstocksToyExporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyExporter.ExportProgressChanged Event

Event that is raised upon progress of exportation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event ProgressChangedEventHandler ExportProgressChanged
```

VB

```
Event ExportProgressChanged As ProgressChangedEventHandler
```

C++

```
event ProgressChangedEventHandler^ ExportProgressChanged {  
    void add (ProgressChangedEventHandler^ value);  
    void remove (ProgressChangedEventHandler^ value);  
}
```

F#

```
abstract ExportProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>
```

Value

Type: [System.ComponentModel.ProgressChangedEventHandler](#)

See Also

[IWoodstocksToyExporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyImporter Interface

Defines an interface to import toy data into the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public interface IWoodstocksToyImporter : IDisposable
```

VB

```
Public Interface IWoodstocksToyImporter
    Inherits IDisposable
```

C++







```
public interface class IWoodstocksToyImporter : IDisposable
```

F#



```
type IWoodstocksToyImporter =
    interface
        interface IDisposable
    end
```

The **IWoodstocksToyImporter** type exposes the following members.

Methods

	Name	Description
	Close	Closes the importer.
	Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable .)
	GetToys	Gets the data for toys that are imported by the importer.
	ImportAsync	Imports toy data asynchronously.
	ImportCancel	Cancels an asynchronous import.
	IsBusy	Returns whether the importer is busy whilst carrying out an asynchronous import.

Events

	Name	Description
	ImportCompleted	Raised upon completion of an asynchronous operation.
	ImportProgressChanged	The event when progress is made on an asynchronous import.

A Sandcastle Documented Class Library







See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyImporter.IWoodstocksToyImporter Methods

The [IWoodstocksToyImporter](#) type exposes the following members.

Methods

	Name	Description
	Close	Closes the importer.
	Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable .)
	GetToys	Gets the data for toys that are imported by the importer.
	ImportAsync	Imports toy data asynchronously.
	ImportCancel	Cancels an asynchronous import.
	IsBusy	Returns whether the importer is busy whilst carrying out an asynchronous import.

See Also

[IWoodstocksToyImporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyImporter.Close Method

Closes the importer.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void Close()
```

VB

```
Sub Close
```

C++

```
void Close()
```

F#

```
abstract Close : unit -> unit
```

See Also

[IWoodstocksToyImporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyImporter.GetToys Method

Gets the data for toys that are imported by the importer.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
IToys GetToys()
```

VB

```
Function GetToys As IToys
```

C++

```
IToys^ GetToys()
```

F#

```
abstract GetToys : unit -> IToys
```

Return Value

Type: [IToys](#)

The toy data that was imported.

See Also

[IWoodstocksToyImporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyImporter.ImportAsync Method

Imports toy data asynchronously.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void ImportAsync(  
    string source  
)
```

VB

```
Sub ImportAsync (  
    source As String  
)
```

C++

```
void ImportAsync(  
    String^ source  
)
```

F#

```
abstract ImportAsync :  
    source : string -> unit
```

Parameters

source

Type: [System.String](#)

The source from which data should be imported.

See Also

[IWoodstocksToyImporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyImporter.ImportCancel Method

Cancels an asynchronous import.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void ImportCancel()
```

VB

```
Sub ImportCancel
```

C++

```
void ImportCancel()
```

F#

```
abstract ImportCancel : unit -> unit
```

See Also

[IWoodstocksToyImporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyImporter.IsBusy Method

Returns whether the importer is busy whilst carrying out an asynchronous import.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
bool IsBusy()
```

VB

```
Function IsBusy As Boolean
```

C++

```
bool IsBusy()
```

F#

```
abstract IsBusy : unit -> bool
```

Return Value

Type: [Boolean](#)

True if the importer is busy carrying out an asynchronous import. False if the importer is not busy.

See Also



[IWoodstocksToyImporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyImporter.IWoodstocksToyImporter Events

The [IWoodstocksToyImporter](#) type exposes the following members.

Events

	Name	Description
	ImportCompleted	Raised upon completion of an asynchronous operation.
	ImportProgressChanged	The event when progress is made on an asynchronous import.

See Also

[IWoodstocksToyImporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyImporter.ImportCompleted Event

Raised upon completion of an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event ImportCompletedHandler ImportCompleted
```

VB

```
Event ImportCompleted As ImportCompletedHandler
```

C++

```
event ImportCompletedHandler^ ImportCompleted {  
    void add (ImportCompletedHandler^ value);  
    void remove (ImportCompletedHandler^ value);  
}
```

F#

```
abstract ImportCompleted : IEvent<ImportCompletedHandler,  
    EventArgs>
```

Value

Type: [Woodstocks.WoodstocksIMS.Domain.ImportCompletedHandler](#)

Remarks

The [Error](#) should be checked to ensure that an error did not occur during the operation. If no error has occurred during the operation the result of the import operation can be retrieved by calling the [GetToys\(\)](#) method.

See Also

[IWoodstocksToyImporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

IWoodstocksToyImporter.ImportProgressChanged Event

The event when progress is made on an asynchronous import.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event ImportProgressChanged ImportProgressChanged
```

VB

```
Event ImportProgressChanged As ImportProgressChanged
```

C++

```
event ImportProgressChanged^ ImportProgressChanged {  
    void add (ImportProgressChanged^ value);  
    void remove (ImportProgressChanged^ value);  
}
```

F#

```
abstract ImportProgressChanged : IEvent<ImportProgressChanged,  
    EventArgs>
```

Value

Type: [Woodstocks.WoodstocksIMS.Domain.ImportProgressChanged](#)

See Also

[IWoodstocksToyImporter Interface](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

OnOrder Enumeration

Represents the order status of a [Toy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public enum OnOrder
```

VB

```
Public Enumeration OnOrder
```

C++

```
public enum class OnOrder
```

F#

```
type OnOrder
```

Members

Member name	Value	Description
No	1	Represents that the Toy is not on order.
Yes	2	Represents that the Toy is on order.

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

SortOrder Enumeration

Defines an enumeration for sort ordering.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public enum SortOrder
```

VB

```
Public Enumeration SortOrder
```

C++

```
public enum class SortOrder
```

F#

```
type SortOrder
```

Members

	Member name	Value	Description
	Ascending	1	An ascending sort.
	Descending	2	A descending sort.

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy Class

Represents a Toy sold by Wood Stocks.

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Domain.Toy

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class Toy : IToy, IComparable<IToy>,
    IComparable, IEquatable<IToy>
```

VB

```
Public Class Toy
    Implements IToy, IComparable(Of IToy),
    IComparable, IEquatable(Of IToy)
```

C++




```
public ref class Toy : IToy, IComparable<IToy^>,
    IComparable, IEquatable<IToy^>
```

F#














```
type Toy =
    class
        interface IToy
        interface IComparable<IToy>
        interface IComparable
        interface IEquatable<IToy>
    end
```

The **Toy** type exposes the following members.






Constructors

	Name	Description
	Toy(IToy)	A copy constructor to initialise a toy.
	Toy(String, String, Int32, OnOrder)	Initialises a Toy that is stocked and sold by Wood Stocks.
	Toy(String, String, String, String)	Initialises a Toy that is stocked and sold by Wood Stocks.


Methods

	Name	Description
	CompareTo(Object)	Compares the current instance with another object and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.
	CompareTo(IToy)	Compares the current IToy with another IToy .
	Equals(Object)	Overrides the Equals(Object) method to provide results comparable to the Equals(T) implementation. (Overrides Object.Equals(Object) .)
	Equals(IToy)	Indicates whether the current Toy is equal to another Toy .
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetCurrentCount	Gets the current count of the Toy .
	GetHashCode	Serves as a hash function for a Toy . (Overrides Object.GetHashCode() .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnChanged	Raises the Changed event.
	SetCurrentCount(Int32)	Sets the current count for the Toy .
	SetCurrentCount(String)	Sets the current count for the Toy .
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Properties

	Name	Description
	CurrentCount	Gets the current count for a IToy .
	InitialCount	Gets the initial count for a IToy .
	ItemCode	Gets the item code for the IToy .
	ItemDescription	Gets the description of the IToy .
	OnOrderStatus	Gets the on order status of an Toy

Events




	Name	Description
	Changed	Event that is raised by a IToy when its count has changed.

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy Constructor

Overload List

	Name	Description
	Toy(IToy)	A copy constructor to initialise a toy.
	Toy(String, String, Int32, OnOrder)	Initialises a Toy that is stocked and sold by Wood Stocks.
	Toy(String, String, String, String)	Initialises a Toy that is stocked and sold by Wood Stocks.

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy Constructor (IToy)

A copy constructor to initialise a toy.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public Toy(  
    IToy toy  
)
```

VB

```
Public Sub New (  
    toy As IToy  
)
```

C++

```
public:  
Toy(  
    IToy^ toy  
)
```

F#

```
new :  
    toy : IToy -> Toy
```

Parameters

toy

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

The existing toy to copy.

See Also

[Toy Class](#)

[Toy Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy Constructor (String, String, Int32, OnOrder)

Initialises a [Toy](#) that is stocked and sold by Wood Stocks.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public Toy(  
    string itemCode,  
    string itemDescription,  
    int currentCount,  
    OnOrder onOrder  
)
```

VB

```
Public Sub New (  
    itemCode As String,  
    itemDescription As String,  
    currentCount As Integer,  
    onOrder As OnOrder  
)
```

C++

```
public:  
Toy(  
    String^ itemCode,  
    String^ itemDescription,  
    int currentCount,  
    OnOrder onOrder  
)
```

F#

```
new :  
    itemCode : string *  
    itemDescription : string *  
    currentCount : int *  
    onOrder : OnOrder -> Toy
```

Parameters

itemCode

Type: [System.String](#)

The item code of the toy.

itemDescription

Type: [System.String](#)

A description for the toy.

currentCount

Type: [System.Int32](#)

The amount of the toy that Wood Stocks currently has in stock.

onOrder

Type: [Woodstocks.WoodstocksIMS.Domain.OnOrder](#)

Indicates whether the toy is on order.

See Also

[Toy Class](#)

[Toy Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy Constructor (String, String, String, String)

Initialises a [Toy](#) that is stocked and sold by Wood Stocks.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public Toy(  
    string itemCode,  
    string itemDescription,  
    string currentCount,  
    string onOrder  
)
```

VB

```
Public Sub New (  
    itemCode As String,  
    itemDescription As String,  
    currentCount As String,  
    onOrder As String  
)
```

C++

```
public:  
Toy(  
    String^ itemCode,  
    String^ itemDescription,  
    String^ currentCount,  
    String^ onOrder  
)
```

F#

```
new :  
    itemCode : string *  
    itemDescription : string *  
    currentCount : string *  
    onOrder : string -> Toy
```

Parameters

itemCode

Type: [System.String](#)

The item code of the toy.

itemDescription

Type: [System.String](#)

A description of the toy.

A Sandcastle Documented Class Library

currentCount

Type: [System.String](#)

The current count of items in stock.

onOrder

Type: [System.String](#)

Indicates whether the item is on order or not.

See Also

[Toy Class](#)














[Toy Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.Toy Methods

The [Toy](#) type exposes the following members.

Methods

	Name	Description
	CompareTo(Object)	Compares the current instance with another object and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.
	CompareTo(IToy)	Compares the current IToy with another IToy .
	Equals(Object)	Overrides the Equals(Object) method to provide results comparable to the Equals(T) implementation. (Overrides Object.Equals(Object) .)
	Equals(IToy)	Indicates whether the current Toy is equal to another Toy .
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetCurrentCount	Gets the current count of the Toy .
	GetHashCode	Serves as a hash function for a Toy . (Overrides Object.GetHashCode() .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnChanged	Raises the Changed event.
	SetCurrentCount(Int32)	Sets the current count for the Toy .
	SetCurrentCount(String)	Sets the current count for the Toy .
	ToString	Returns a string that represents the current object. (Inherited from Object .)



See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.CompareTo Method

Overload List

	Name	Description
	CompareTo(Object)	Compares the current instance with another object and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.
	CompareTo(IToy)	Compares the current IToy with another IToy .

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.CompareTo Method (Object)

Compares the current instance with another object and returns an integer that indicates whether the current instance precedes, follows, or occurs in the same position in the sort order as the other object.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public int CompareTo(  
    Object obj  
)
```

VB

```
Public Function CompareTo (  
    obj As Object  
) As Integer
```

C++

```
public:  
virtual int CompareTo(  
    Object^ obj  
) sealed
```

F#

```
abstract CompareTo :  
    obj : Object -> int  
override CompareTo :  
    obj : Object -> int
```

Parameters

obj

Type: [System.Object](#)

An object to compare the current [IToy](#) with.

Return Value

Type: [Int32](#)

1 if the current [IToy](#) follows *obj* in the sort order. 0 if the current toy occurs in the same position as *obj*. -

1 if the current toy precedes *obj* in the sort order.

Implements

[IComparable.CompareTo\(Object\)](#)

Exceptions

Exception	Condition
ArgumentException	obj is not an IToy .

Remarks

By definition, any object compares greater than (or follows) null. Thus if the other object is null then this method returns 1. This method ensures results are consistent with the implementation of [IComparable\(T\)](#) by attempting to convert other to an [IToy](#). If the conversion fails an [ArgumentException](#) is thrown. Otherwise the CompareTo method of the generic interface is invoked.

See Also

[Toy Class](#)

[CompareTo Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.CompareTo Method (IToy)

Compares the current [IToy](#) with another [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public int CompareTo(  
    IToy other  
)
```

VB

```
Public Function CompareTo (  
    other As IToy  
) As Integer
```

C++

```
public:  
virtual int CompareTo(  
    IToy^ other  
) sealed
```

F#

```
abstract CompareTo :  
    other : IToy -> int  
override CompareTo :  
    other : IToy -> int
```

Parameters

other

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

Another toy to compare the current toy with

Return Value

Type: [Int32](#)

Returns -1 if the current toy should precede other, 0 if the two toys occur in the same order, and 1 if the current toy follows other in a sort order

Implements

[IComparable\(T\).CompareTo\(T\)](#)

Remarks

The comparison of two [IToys](#) is conducted on the basis of their item codes. Casing for the comparison is ignored. This ensures that an toy with the item code A0001 precedes a toy with a0002, for example. It is assumed casing withing an item code is non-significant.

See Also



[Toy Class](#)

[CompareTo Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.Equals Method

Overload List

	Name	Description
	Equals(Object)	Overrides the Equals(Object) method to provide results comparable to the Equals(T) implementation. (Overrides Object.Equals(Object) .)
	Equals(IToy)	Indicates whether the current Toy is equal to another Toy .

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.Equals Method (Object)

Overrides the [Equals\(Object\)](#) method to provide results comparable to the [Equals\(T\)](#) implementation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public override bool Equals(  
    Object other  
)
```

VB

```
Public Overrides Function Equals (  
    other As Object  
) As Boolean
```

C++

```
public:  
virtual bool Equals(  
    Object^ other  
) override
```

F#

```
abstract Equals :  
    other : Object -> bool  
override Equals :  
    other : Object -> bool
```

Parameters

other

Type: [System.Object](#)

Return Value

Type: [Boolean](#)

True if two [Toy](#) are equal, otherwise false.

See Also

[Toy Class](#)

[Equals Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.Equals Method (IToy)

Indicates whether the current [IToy](#) is equal to another [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool Equals(  
    IToy other  
)
```

VB

```
Public Function Equals (  
    other As IToy  
) As Boolean
```

C++

```
public:  
virtual bool Equals(  
    IToy^ other  
) sealed
```

F#

```
abstract Equals :  
    other : IToy -> bool  
override Equals :  
    other : IToy -> bool
```

Parameters

other

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

A [IToy](#) to compare the current [IToy](#) with

Return Value

Type: [Boolean](#)

True if the current [IToy](#) is equal to the other [IToy](#), otherwise false.

Implements

[IEquatable\(T\).Equals\(T\)](#)

Remarks

The current [IToy](#) is equal to the other [IToy](#) if the item codes of the two objects are the same. It is assumed that the item code for an [IToy](#) should be unique to an [IToy](#) and if equal to any other [IToy](#) then the two objects refer to the same toy sold by Wood Stocks.

A Sandcastle Documented Class Library

See Also

[Toy Class](#)

[Equals Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.GetCurrentCount Method

Gets the current count of the [Toy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual string GetCurrentCount()
```

VB

```
Protected Overridable Function GetCurrentCount As String
```

C++

```
protected:  
virtual String^ GetCurrentCount()
```

F#

```
abstract GetCurrentCount : unit -> string  
override GetCurrentCount : unit -> string
```

Return Value

Type: [String](#)

[Missing <returns> documentation for "M:Woodstocks.WoodstocksIMS.Domain.Toy.GetCurrentCount"]

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.GetHashCode Method

Serves as a hash function for a [Toy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public override int GetHashCode()
```

VB

```
Public Overrides Function GetHashCode As Integer
```

C++

```
public:  
virtual int GetHashCode() override
```

F#

```
abstract GetHashCode : unit -> int  
override GetHashCode : unit -> int
```

Return Value

Type: [Int32](#)

A hash code for the [Toy](#).

Remarks

The hash code of the item code for a [Toy](#) is utilised for the hash code of a [Toy](#). The item code is immutable, as it cannot be changed following instantiation of a [Toy](#). If the item code becomes changable then this method would need to be altered because the item code hash could no longer be reliably used as a hash code of a [Toy](#).

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.OnChanged Method

Raises the [Changed](#) event.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void OnChanged(  
    ToyChangedEventArgs e  
)
```

VB

```
Protected Overridable Sub OnChanged (  
    e As ToyChangedEventArgs  
)
```

C++

```
protected:  
virtual void OnChanged(  
    ToyChangedEventArgs^ e  
)
```

F#

```
abstract OnChanged :  
    e : ToyChangedEventArgs -> unit  
override OnChanged :  
    e : ToyChangedEventArgs -> unit
```

Parameters

e

Type: [Woodstocks.WoodstocksIMS.Domain.ToyChangedEventArgs](#)

An [ToyChangedEventArgs](#) that contains the event data.

Remarks

The [ToyChangedEventArgs](#) event data for the event contains the initial and current counts for a toy. This enables detection as to whether the current count differs from the initial count of a [Toy](#).



See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.SetCurrentCount Method

Overload List

	Name	Description
	SetCurrentCount(Int32)	Sets the current count for the Toy .
	SetCurrentCount(String)	Sets the current count for the Toy .

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.SetCurrentCount Method (Int32)

Sets the current count for the [Toy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void SetCurrentCount(  
    int count  
)
```

VB

```
Protected Overridable Sub SetCurrentCount (  
    count As Integer  
)
```

C++

```
protected:  
virtual void SetCurrentCount(  
    int count  
)
```

F#

```
abstract SetCurrentCount :  
    count : int -> unit  
override SetCurrentCount :  
    count : int -> unit
```

Parameters

count

Type: [System.Int32](#)

The current count of the [Toy](#).

See Also

[Toy Class](#)

[SetCurrentCount Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.SetCurrentCount Method (String)

Sets the current count for the [Toy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void SetCurrentCount(  
    string count  
)
```

VB

```
Protected Overridable Sub SetCurrentCount (  
    count As String  
)
```

C++

```
protected:  
virtual void SetCurrentCount(  
    String^ count  
)
```

F#

```
abstract SetCurrentCount :  
    count : string -> unit  
override SetCurrentCount :  
    count : string -> unit
```

Parameters

count

Type: [System.String](#)

The current count of the [Toy](#).

Remarks

This method throws an [InvalidCurrentCountException](#) if the value of count cannot be converted into a integer value.

See Also

[Toy Class](#)






[SetCurrentCount Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.Toy Properties

The [Toy](#) type exposes the following members.

Properties

	Name	Description
	CurrentCount	Gets the current count for a IToy .
	InitialCount	Gets the initial count for a IToy .
	ItemCode	Gets the item code for the IToy .
	ItemDescription	Gets the description of the IToy .
	OnOrderStatus	Gets the on order status of an Toy

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.CurrentCount Property

Gets the current count for a [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Public Property CurrentCount As String  
    Get  
    Set
```

C++

```
public:  
virtual property String^ CurrentCount {  
    String^ get () sealed;  
    void set (String^ value) sealed;  
}
```

F#

```
abstract CurrentCount : string with get, set  
override CurrentCount : string with get, set
```

Property Value

Type: [String](#)

Implements

[IToy.CurrentCount](#)

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.InitialCount Property

Gets the initial count for a [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public string InitialCount { get; }
```

VB

```
Public ReadOnly Property InitialCount As String  
    Get
```

C++

```
public:  
virtual property String^ InitialCount {  
    String^ get () sealed;  
}
```

F#

```
abstract InitialCount : string with get  
override InitialCount : string with get
```

Property Value

Type: [String](#)

Implements

[IToy.InitialCount](#)

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.ItemCode Property

Gets the item code for the [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public string ItemCode { get; }
```

VB

```
Public ReadOnly Property ItemCode As String  
    Get
```

C++

```
public:  
virtual property String^ ItemCode {  
    String^ get () sealed;  
}
```

F#

```
abstract ItemCode : string with get  
override ItemCode : string with get
```

Property Value

Type: [String](#)

Implements

[IToy.ItemCode](#)

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.ItemDescription Property

Gets the description of the [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public string ItemDescription { get; }
```

VB

```
Public ReadOnly Property ItemDescription As String  
    Get
```

C++

```
public:  
virtual property String^ ItemDescription {  
    String^ get () sealed;  
}
```

F#

```
abstract ItemDescription : string with get  
override ItemDescription : string with get
```

Property Value

Type: [String](#)

Implements

[IToy.ItemDescription](#)

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.OnOrderStatus Property

Gets the on order status of an [Toy](#)

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public string OnOrderStatus { get; }
```

VB

```
Public ReadOnly Property OnOrderStatus As String  
    Get
```

C++

```
public:  
virtual property String^ OnOrderStatus {  
    String^ get () sealed;  
}
```

F#

```
abstract OnOrderStatus : string with get  
override OnOrderStatus : string with get
```

Property Value

Type: [String](#)

Implements

[IToy.OnOrderStatus](#)

See Also


[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.Toy Events

The [Toy](#) type exposes the following members.

Events

	Name	Description
	Changed	Event that is raised by a IToy when its count has changed.

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toy.Changed Event

Event that is raised by a [IToy](#) when its count has changed.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event ToyChangedEventHandler Changed
```

VB

```
Public Event Changed As ToyChangedEventHandler
```

C++

```
public:
virtual event ToyChangedEventHandler^ Changed {
    void add (ToyChangedEventHandler^ value);
    void remove (ToyChangedEventHandler^ value);
}
```

F#

```
abstract Changed : IEvent<ToyChangedEventHandler,
    EventArgs>
override Changed : IEvent<ToyChangedEventHandler,
    EventArgs>
```

Value

Type: [Woodstocks.WoodstocksIMS.Domain.ToyChangedEventHandler](#)

Implements

[IToy.Changed](#)

See Also

[Toy Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ToyChangedEventArgs Class

Contains the event data for a [ToyChanged](#)

Inheritance Hierarchy

[System.Object](#)

[System.EventArgs](#)

Woodstocks.WoodstocksIMS.Domain.ToyChangedEventArgs

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class ToyChangedEventArgs : EventArgs
```

VB

```
Public Class ToyChangedEventArgs  
    Inherits EventArgs
```

C++


```
public ref class ToyChangedEventArgs : public EventArgs
```

F#




```
type ToyChangedEventArgs =  
    class  
        inherit EventArgs  
    end
```




The **ToyChangedEventArgs** type exposes the following members.

Constructors



	Name	Description
	ToyChangedEventArgs	Initializes a new instance of the ToyChangedEventArgs class

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)

	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

Properties

	Name	Description
	Field	Gets the field (or property) of the Toy that has changed.
	Toy	Gets the Toy that has changed.

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ToyChangedEventArgs Constructor

Initializes a new instance of the [ToyChangedEventArgs](#) class

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public ToyChangedEventArgs(  
    IToy toy,  
    ToyField field  
)
```

VB

```
Public Sub New (  
    toy As IToy,  
    field As ToyField  
)
```

C++

```
public:  
ToyChangedEventArgs(  
    IToy^ toy,  
    ToyField field  
)
```

F#

```
new :  
    toy : IToy *  
    field : ToyField -> ToyChangedEventArgs
```

Parameters

toy

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

field

Type: [Woodstocks.WoodstocksIMS.Domain.ToyField](#)

See Also







[ToyChangedEventArgs Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ToyChangedEventArgs.ToyChangedEventArgs Methods

The [ToyChangedEventArgs](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

See Also



[ToyChangedEventArgs Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ToyChangedEventArgs.ToyChangedEventArgs Properties

The [ToyChangedEventArgs](#) type exposes the following members.

Properties

	Name	Description
	Field	Gets the field (or property) of the Toy that has changed.
	Toy	Gets the Toy that has changed.

See Also

[ToyChangedEventArgs Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ToyChangedEventArgs.Field Property

Gets the field (or property) of the [Toy](#) that has changed.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public ToyField Field { get; }
```

VB

```
Public ReadOnly Property Field As ToyField  
    Get
```

C++

```
public:  
property ToyField Field {  
    ToyField get ();  
}
```

F#

```
member Field : ToyField with get
```

Property Value

Type: [ToyField](#)

See Also

[ToyChangedEventArgs Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ToyChangedEventArgs.Toy Property

Gets the **Toy** that has changed.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IToy Toy { get; }
```

VB

```
Public ReadOnly Property Toy As IToy  
    Get
```

C++

```
public:  
property IToy^ Toy {  
    IToy^ get ();  
}
```

F#

```
member Toy : IToy with get
```

Property Value

Type: [IToy](#)

See Also

[ToyChangedEventArgs Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ToyChangedEventHandler Delegate

Defines a delegate to handle the [Changed](#) event.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public delegate void ToyChangedEventHandler(  
    IToy sender,  
    ToyChangedEventArgs e  
)
```

VB

```
Public Delegate Sub ToyChangedEventHandler (  
    sender As IToy,  
    e As ToyChangedEventArgs  
)
```

C++

```
public delegate void ToyChangedEventHandler(  
    IToy^ sender,  
    ToyChangedEventArgs^ e  
)
```

F#

```
type ToyChangedEventHandler =  
    delegate of  
        sender : IToy *  
        e : ToyChangedEventArgs -> unit
```

Parameters

sender

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

The [IToy](#) that has changed.

e

Type: [Woodstocks.WoodstocksIMS.Domain.ToyChangedEventArgs](#)

An [ToyChangedEventArgs](#) that contains the event data.

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

ToyField Enumeration

Defines named constants for the fields of an [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public enum ToyField
```

VB

```
Public Enumeration ToyField
```

C++

```
public enum class ToyField
```

F#

```
type ToyField
```

Members

Member name	Value	Description
ItemCode	1	Identifies the ItemCode value.
ItemDescription	2	Identifies the ItemDescription value.
CurrentCount	3	Identifies the CurrentCount value.
InitialCount	4	Identifies the InitialCount value.
OnOrder	5	Identifies the OnOrderStatus value.

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys Class

A collection of Toys.

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Domain.Toys

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class Toys : IToys, IList<IToy>,
    ICollection<IToy>, IEnumerable<IToy>, IList,
    ICollection, IEnumerable
```

VB

```
Public Class Toys
    Implements IToys, IList(Of IToy),
    ICollection(Of IToy), IEnumerable(Of IToy), IList,
    ICollection, IEnumerable
```

C++



```
public ref class Toys : IToys, IList<IToy^>,
    ICollection<IToy^>, IEnumerable<IToy^>, IList,
    ICollection, IEnumerable
```

F#






```
type Toys =
    class
        interface IToys
        interface IList<IToy>
        interface ICollection<IToy>
        interface IEnumerable<IToy>
        interface IList
        interface ICollection
        interface IEnumerable
    end
```







The **Toys** type exposes the following members.

Constructors







	Name	Description
	Toys()	Initialises a Toys collection.
	Toys(IToys)	Initialise a Toys collection.

Methods


	Name	Description
	Add(Object)	Adds an item to the IList .
	Add(IToy)	Adds an item to the ICollection(T) .
	Clear	Removes all IToy from the collection.
	Contains(Object)	Determines whether the IList contains a specific value.
	Contains(IToy)	Determines whether the ICollection(T) contains a specific value.
	CopyTo(Array, Int32)	Copies the elements of the ICollection to an Array , starting at a particular Array index.
	CopyTo(IToy[], Int32)	Copies the elements of the ICollection(T) to an Array , starting at a particular Array index.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetEnumerator	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IndexOf(Object)	Determines the index of a specific item in the IList .
	IndexOf(IToy)	Determines the index of a specific item in the IList(T) .
	Insert(Int32, Object)	Inserts an item to the IList at the specified index.
	Insert(Int32, IToy)	Inserts an item to the IList(T) at the specified index.
	ItemChanged	Handles the Changed event of a toy in the collection when it is raised.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnToyChanged	Raises the ToyChanged event of the collection when a IToy within the collection changes.
	Remove(Object)	Removes the first occurrence of a specific object from the IList .
	Remove(IToy)	Removes the first occurrence of a specific object from the ICollection(T) .
	RemoveAt	Removes the IToy from the collection located at index.
	SortByCurrentCount	Sorts the collection of IToy by current count in the specified sort order.
	SortByCurrentCountAscending	Sorts the collection, in ascending order, by the Current Count for a

		IToy.
	SortByCurrentCountDescending	The implemenation to sort a collection of Toys by the current count of the toy in descending order.
	SortByItemCode	Sorts the collection of IToy by item code in the specified sort order..
	SortByOnOrder	Sorts the collection of IToy by on order status in the specified sort order.
	SortByOnOrderAscending	The implemenation to sort a collection of Toys by on order status in ascending order.
	SortByOnOrderDescending	The implemenation to sort a collection of Toys by on order status in descending order.
	ToString	Returns a string that represents the current object. (Inherited from Object.)

Properties

	Name	Description
	Count	The total number of IToys in the collection.
	IsFixedSize	Gets a value indicating whether the IList has a fixed size.
	IsReadOnly	Gets a value indicating whether the ICollection(T) is read-only.
	IsSynchronized	Gets a value indicating whether access to the ICollection is synchronized (thread safe).
	Item	Gets the IToy located at the indexed position specified by index.
	SyncRoot	Gets an object that can be used to synchronize access to the ICollection .

Events



	Name	Description
	ToyChanged	Event raised when an item in the collection changes.

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys Constructor

Overload List

	Name	Description
	Toys()	Initialises a Toys collection.
	Toys(IToys)	Initialise a Toys collection.

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys Constructor

Initialises a [Toys](#) collection.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public Toys ( )
```

VB

```
Public Sub New
```

C++

```
public :  
Toys ( )
```

F#

```
new : unit -> Toys
```

See Also

[Toys Class](#)

[Toys Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys Constructor (IToys)

Initialise a [Toys](#) collection.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public Toys(  
    IToys toys  
)
```

VB

```
Public Sub New (  
    toys As IToys  
)
```

C++

```
public:  
Toys(  
    IToys^ toys  
)
```

F#

```
new :  
    toys : IToys -> Toys
```

Parameters

toys

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

See Also

[Toys Class](#)











[Toys Overload](#)








[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Toys Methods

The [Toys](#) type exposes the following members.

Methods

	Name	Description
	Add(Object)	Adds an item to the IList .
	Add(IToy)	Adds an item to the ICollection(T) .
	Clear	Removes all IToy from the collection.
	Contains(Object)	Determines whether the IList contains a specific value.
	Contains(IToy)	Determines whether the ICollection(T) contains a specific value.
	CopyTo(Array, Int32)	Copies the elements of the ICollection to an Array , starting at a particular Array index.
	CopyTo(IToy[], Int32)	Copies the elements of the ICollection(T) to an Array , starting at a particular Array index.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetEnumerator	Returns an enumerator that iterates through the collection.
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	IndexOf(Object)	Determines the index of a specific item in the IList .
	IndexOf(IToy)	Determines the index of a specific item in the IList(T) .
	Insert(Int32, Object)	Inserts an item to the IList at the specified index.
	Insert(Int32, IToy)	Inserts an item to the IList(T) at the specified index.
	ItemChanged	Handles the Changed event of a toy in the collection when it is raised.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnToyChanged	Raises the ToyChanged event of the collection when a IToy within the collection changes.
	Remove(Object)	Removes the first occurrence of a specific object from the IList .
	Remove(IToy)	Removes the first occurrence of a specific object from the ICollection(T) .
	RemoveAt	Removes the IToy from the collection located at index.
	SortByCurrentCount	Sorts the collection of IToy by current count in the specified sort

		order.
	SortByCurrentCountAscending	Sorts the collection, in ascending order, by the Current Count for a IToy .
	SortByCurrentCountDescending	The implemenation to sort a collection of Toys by the current count of the toy in descending order.
	SortByItemCode	Sorts the collection of IToy by item code in the specified sort order..
	SortByOnOrder	Sorts the collection of IToy by on order status in the specified sort order.
	SortByOnOrderAscending	The implemenation to sort a collection of Toys by on order status in ascending order.
	SortByOnOrderDescending	The implemenation to sort a collection of Toys by on order status in descending order.
	ToString	Returns a string that represents the current object. (Inherited from Object .)


See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Add Method

Overload List

	Name	Description
	Add(Object)	Adds an item to the IList .
	Add(IToy)	Adds an item to the ICollection(T) .

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Add Method (Object)

Adds an item to the [IList](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public int Add(  
    Object value  
)
```

VB

```
Public Function Add (  
    value As Object  
) As Integer
```

C++

```
public:  
virtual int Add(  
    Object^ value  
) sealed
```

F#

```
abstract Add :  
    value : Object -> int  
override Add :  
    value : Object -> int
```

Parameters

value

Type: [System.Object](#)

The object to add to the [IList](#).

Return Value

Type: [Int32](#)

The position into which the new element was inserted, or -1 to indicate that the item was not inserted into the collection,

Implements

[IList.Add\(Object\)](#)

Exceptions

Exception	Condition
NotSupportedException	The IList is read-only.-or- The IList has a fixed size.

See Also

[Toys Class](#)

[Add Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Add Method (IToy)

Adds an item to the [ICollection\(T\)](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Add(  
    IToy item  
)
```

VB

```
Public Sub Add (  
    item As IToy  
)
```

C++

```
public:  
virtual void Add(  
    IToy^ item  
) sealed
```

F#

```
abstract Add :  
    item : IToy -> unit  
override Add :  
    item : IToy -> unit
```

Parameters

item

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

The object to add to the [ICollection\(T\)](#).

Implements

[ICollection\(T\).Add\(T\)](#)

Exceptions

Exception	Condition
NotSupportedException	The ICollection(T) is read-only.

See Also

[Toys Class](#)

[Add Overload](#)

A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Clear Method

Removes all [IToy](#) from the collection.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Clear()
```

VB

```
Public Sub Clear
```

C++

```
public:  
virtual void Clear() sealed
```

F#

```
abstract Clear : unit -> unit  
override Clear : unit -> unit
```

Implements

[IToys.Clear\(\)](#)

[ICollection\(T\).Clear\(\)](#)

[IList.Clear\(\)](#)

Exceptions

Exception	Condition
NotSupportedException	The ICollection(T) is read-only.



See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Contains Method

Overload List

	Name	Description
	Contains(Object)	Determines whether the IList contains a specific value.
	Contains(IToy)	Determines whether the ICollection(T) contains a specific value.

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Contains Method (Object)

Determines whether the [IList](#) contains a specific value.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool Contains(  
    Object value  
)
```

VB

```
Public Function Contains (  
    value As Object  
) As Boolean
```

C++

```
public:  
virtual bool Contains(  
    Object^ value  
) sealed
```

F#

```
abstract Contains :  
    value : Object -> bool  
override Contains :  
    value : Object -> bool
```

Parameters

value

Type: [System.Object](#)

The object to locate in the [IList](#).

Return Value

Type: [Boolean](#)

true if the [Object](#) is found in the [IList](#); otherwise, false.

Implements

[IList.Contains\(Object\)](#)

See Also

[Toys Class](#)

[Contains Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Contains Method (IToy)

Determines whether the [ICollection\(T\)](#) contains a specific value.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool Contains(  
    IToy item  
)
```

VB

```
Public Function Contains (  
    item As IToy  
) As Boolean
```

C++

```
public:  
virtual bool Contains(  
    IToy^ item  
) sealed
```

F#

```
abstract Contains :  
    item : IToy -> bool  
override Contains :  
    item : IToy -> bool
```

Parameters

item

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

The object to locate in the [ICollection\(T\)](#).

Return Value

Type: [Boolean](#)

true if *item* is found in the [ICollection\(T\)](#); otherwise, false.

Implements

[ICollection\(T\).Contains\(T\)](#)

See Also

[Toys Class](#)

[Contains Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.CopyTo Method

Overload List

	Name	Description
	CopyTo(Array, Int32)	Copies the elements of the ICollection to an Array , starting at a particular Array index.
	CopyTo(IToy[], Int32)	Copies the elements of the ICollection(T) to an Array , starting at a particular Array index.

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.CopyTo Method (Array, Int32)

Copies the elements of the [ICollection](#) to an [Array](#), starting at a particular [Array](#) index.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void CopyTo(  
    Array array,  
    int index  
)
```

VB

```
Public Sub CopyTo (  
    array As Array,  
    index As Integer  
)
```

C++

```
public:  
virtual void CopyTo(  
    Array^ array,  
    int index  
) sealed
```

F#

```
abstract CopyTo :  
    array : Array *  
    index : int -> unit  
override CopyTo :  
    array : Array *  
    index : int -> unit
```

Parameters

array

Type: [System.Array](#)

The one-dimensional [Array](#) that is the destination of the elements copied from [ICollection](#). The [Array](#) must have zero-based indexing.

index

Type: [System.Int32](#)

The zero-based index in *array* at which copying begins.

Implements

[ICollection.CopyTo\(Array, Int32\)](#)

Exceptions

Exception	Condition
ArgumentNullException	<i>array</i> is null.
ArgumentOutOfRangeException	<i>index</i> is less than zero.
ArgumentException	<i>array</i> is multidimensional.-or- The number of elements in the source ICollection is greater than the available space from <i>index</i> to the end of the destination <i>array</i> .-or-The type of the source ICollection cannot be cast automatically to the type of the destination <i>array</i> .

Remarks

This method is not implemented because it is not currently needed in the solution.

See Also

[Toys Class](#)

[CopyTo Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.CopyTo Method (IToy[], Int32)

Copies the elements of the [ICollection\(T\)](#) to an [Array](#), starting at a particular [Array](#) index.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void CopyTo(  
    IToy[] array,  
    int arrayIndex  
)
```

VB

```
Public Sub CopyTo (  
    array As IToy(),  
    arrayIndex As Integer  
)
```

C++

```
public:  
virtual void CopyTo(  
    array<IToy^>^ array,  
    int arrayIndex  
) sealed
```

F#

```
abstract CopyTo :  
    array : IToy[] *  
    arrayIndex : int -> unit  
override CopyTo :  
    array : IToy[] *  
    arrayIndex : int -> unit
```

Parameters

array

Type: [Woodstocks.WoodstocksIMS.Domain.IToy\[\]](#)

The one-dimensional [Array](#) that is the destination of the elements copied from [ICollection\(T\)](#). The [Array](#) must have zero-based indexing.

arrayIndex

Type: [System.Int32](#)

The zero-based index in *array* at which copying begins.

Implements

[ICollection\(T\).CopyTo\(T\[\], Int32\)](#)

Exceptions

Exception	Condition
ArgumentNullException	<i>array</i> is null.
ArgumentOutOfRangeException	<i>arrayIndex</i> is less than 0.
ArgumentException	The number of elements in the source ICollection(T) is greater than the available space from <i>arrayIndex</i> to the end of the destination <i>array</i> .

See Also

[Toys Class](#)

[CopyTo Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.GetEnumerator Method

Returns an enumerator that iterates through the collection.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IEnumerator<IToy> GetEnumerator()
```

VB

```
Public Function GetEnumerator As IEnumerator(Of IToy)
```

C++

```
public:  
virtual IEnumerator<IToy^> GetEnumerator() sealed
```

F#

```
abstract GetEnumerator : unit -> IEnumerator<IToy>  
override GetEnumerator : unit -> IEnumerator<IToy>
```

Return Value

Type: [IEnumerator\(IToy\)](#)

A [IEnumerator\(T\)](#) that can be used to iterate through the collection.

Implements

[IEnumerable\(T\).GetEnumerator\(\)](#)



See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.IndexOf Method

Overload List

	Name	Description
	IndexOf(Object)	Determines the index of a specific item in the IList .
	IndexOf(IToy)	Determines the index of a specific item in the IList(T) .

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.IndexOf Method (Object)

Determines the index of a specific item in the [IList](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public int IndexOf(  
    Object value  
)
```

VB

```
Public Function IndexOf (  
    value As Object  
) As Integer
```

C++

```
public:  
virtual int IndexOf(  
    Object^ value  
) sealed
```

F#

```
abstract IndexOf :  
    value : Object -> int  
override IndexOf :  
    value : Object -> int
```

Parameters

value

Type: [System.Object](#)

The object to locate in the [IList](#).

Return Value

Type: [Int32](#)

The index of *value* if found in the list; otherwise, -1.

Implements

[IList.IndexOf\(Object\)](#)

See Also

[Toys Class](#)

[IndexOf Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.IndexOf Method (IToy)

Determines the index of a specific item in the [IList\(T\)](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public int IndexOf(  
    IToy item  
)
```

VB

```
Public Function IndexOf (  
    item As IToy  
) As Integer
```

C++

```
public:  
virtual int IndexOf(  
    IToy^ item  
) sealed
```

F#

```
abstract IndexOf :  
    item : IToy -> int  
override IndexOf :  
    item : IToy -> int
```

Parameters

item

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

The object to locate in the [IList\(T\)](#).

Return Value

Type: [Int32](#)

The index of *item* if found in the list; otherwise, -1.

Implements

[IList\(T\).IndexOf\(T\)](#)

See Also



[Toys Class](#)

[IndexOf Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Insert Method

Overload List

	Name	Description
	Insert(Int32, Object)	Inserts an item to the IList at the specified index.
	Insert(Int32, IToy)	Inserts an item to the IList(T) at the specified index.

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Insert Method (Int32, Object)

Inserts an item to the [IList](#) at the specified index.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Insert(  
    int index,  
    Object value  
)
```

VB

```
Public Sub Insert (  
    index As Integer,  
    value As Object  
)
```

C++

```
public:  
virtual void Insert(  
    int index,  
    Object^ value  
) sealed
```

F#

```
abstract Insert :  
    index : int *  
    value : Object -> unit  
override Insert :  
    index : int *  
    value : Object -> unit
```

Parameters

index

Type: [System.Int32](#)

The zero-based index at which *value* should be inserted.

value

Type: [System.Object](#)

The object to insert into the [IList](#).

Implements

[IList.Insert\(Int32, Object\)](#)

Exceptions

Exception	Condition
ArgumentOutOfRangeException	<i>index</i> is not a valid index in the IList .
NotSupportedException	The IList is read-only.-or- The IList has a fixed size.
NullReferenceException	<i>value</i> is null reference in the IList .

See Also

[Toys Class](#)

[Insert Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Insert Method (Int32, IToy)

Inserts an item to the [IList\(T\)](#) at the specified index.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Insert(  
    int index,  
    IToy item  
)
```

VB

```
Public Sub Insert (  
    index As Integer,  
    item As IToy  
)
```

C++

```
public:  
virtual void Insert(  
    int index,  
    IToy^ item  
) sealed
```

F#

```
abstract Insert :  
    index : int *  
    item : IToy -> unit  
override Insert :  
    index : int *  
    item : IToy -> unit
```

Parameters

index

Type: [System.Int32](#)

The zero-based index at which *item* should be inserted.

item

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

The object to insert into the [IList\(T\)](#).

Implements

[IList\(T\).Insert\(Int32, T\)](#)

Exceptions

Exception	Condition
ArgumentOutOfRangeException	<i>index</i> is not a valid index in the IList(T) .
NotSupportedException	The IList(T) is read-only.

See Also

[Toys Class](#)

[Insert Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.ItemChanged Method

Handles the [Changed](#) event of a toy in the collection when it is raised.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void ItemChanged(  
    IToy sender,  
    ToyChangedEventArgs e  
)
```

VB

```
Protected Overridable Sub ItemChanged (  
    sender As IToy,  
    e As ToyChangedEventArgs  
)
```

C++

```
protected:  
virtual void ItemChanged(  
    IToy^ sender,  
    ToyChangedEventArgs^ e  
)
```

F#

```
abstract ItemChanged :  
    sender : IToy *  
    e : ToyChangedEventArgs -> unit  
override ItemChanged :  
    sender : IToy *  
    e : ToyChangedEventArgs -> unit
```

Parameters

sender

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

The toy for which the [Changed](#) event has been raised.

e

Type: [Woodstocks.WoodstocksIMS.Domain.ToyChangedEventArgs](#)

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.OnToyChanged Method

Raises the [ToyChanged](#) event of the collection when a [IToy](#) within the collection changes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void OnToyChanged(  
    ToyChangedEventArgs e  
)
```

VB

```
Protected Overridable Sub OnToyChanged (  
    e As ToyChangedEventArgs  
)
```

C++

```
protected:  
virtual void OnToyChanged(  
    ToyChangedEventArgs^ e  
)
```

F#

```
abstract OnToyChanged :  
    e : ToyChangedEventArgs -> unit  
override OnToyChanged :  
    e : ToyChangedEventArgs -> unit
```

Parameters

e

Type: [Woodstocks.WoodstocksIMS.Domain.ToyChangedEventArgs](#)

Event data that indicates the the [IToy](#) that changed and the nature of the change.



See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Remove Method

Overload List

	Name	Description
	Remove(Object)	Removes the first occurrence of a specific object from the IList .
	Remove(IToy)	Removes the first occurrence of a specific object from the ICollection(T) .

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Remove Method (Object)

Removes the first occurrence of a specific object from the [IList](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Remove(  
    Object value  
)
```

VB

```
Public Sub Remove (  
    value As Object  
)
```

C++

```
public:  
virtual void Remove(  
    Object^ value  
) sealed
```

F#

```
abstract Remove :  
    value : Object -> unit  
override Remove :  
    value : Object -> unit
```

Parameters

value

Type: [System.Object](#)

The object to remove from the [IList](#).

Implements

[IList.Remove\(Object\)](#)

Exceptions

Exception	Condition
NotSupportedException	The IList is read-only.-or- The IList has a fixed size.

See Also

[Toys Class](#)

[Remove Overload](#)

A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Remove Method (IToy)

Removes the first occurrence of a specific object from the [ICollection\(T\)](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool Remove(  
    IToy item  
)
```

VB

```
Public Function Remove (  
    item As IToy  
) As Boolean
```

C++

```
public:  
virtual bool Remove(  
    IToy^ item  
) sealed
```

F#

```
abstract Remove :  
    item : IToy -> bool  
override Remove :  
    item : IToy -> bool
```

Parameters

item

Type: [Woodstocks.WoodstocksIMS.Domain.IToy](#)

The object to remove from the [ICollection\(T\)](#).

Return Value

Type: [Boolean](#)

true if *item* was successfully removed from the [ICollection\(T\)](#); otherwise, false. This method also returns false if *item* is not found in the original [ICollection\(T\)](#).

Implements

[ICollection\(T\).Remove\(T\)](#)

Exceptions

Exception	Condition
NotSupportedException	The ICollection(T) is read-only.

See Also

[Toys Class](#)

[Remove Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.RemoveAt Method

Removes the [IToy](#) from the collection located at index.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void RemoveAt(  
    int index  
)
```

VB

```
Public Sub RemoveAt (  
    index As Integer  
)
```

C++

```
public:  
virtual void RemoveAt(  
    int index  
) sealed
```

F#

```
abstract RemoveAt :  
    index : int -> unit  
override RemoveAt :  
    index : int -> unit
```

Parameters

index

Type: [System.Int32](#)

The position within the collection of the [IToy](#) to be removed from the collection.

Implements

[IToys.RemoveAt\(Int32\)](#)

[IList\(T\).RemoveAt\(Int32\)](#)

[IList.RemoveAt\(Int32\)](#)

Exceptions

Exception	Condition
ArgumentOutOfRangeException	<i>index</i> is not a valid index in the IList(T) .
NotSupportedException	The IList(T) is read-only.

A Sandcastle Documented Class Library

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.SortByCurrentCount Method

Sorts the collection of [IToy](#) by current count in the specified sort order.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SortByCurrentCount(  
    SortOrder order  
)
```

VB

```
Public Sub SortByCurrentCount (  
    order As SortOrder  
)
```

C++

```
public:  
virtual void SortByCurrentCount(  
    SortOrder order  
) sealed
```

F#

```
abstract SortByCurrentCount :  
    order : SortOrder -> unit  
override SortByCurrentCount :  
    order : SortOrder -> unit
```

Parameters

order

Type: [Woodstocks.WoodstocksIMS.Domain.SortOrder](#)

The sort order.

Implements

[IToys.SortByCurrentCount\(SortOrder\)](#)

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.SortByCurrentCountAscending Method

Sorts the collection, in ascending order, by the Current Count for a [IToy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void SortByCurrentCountAscending()
```

VB

```
Protected Overridable Sub SortByCurrentCountAscending
```

C++

```
protected:  
virtual void SortByCurrentCountAscending()
```

F#

```
abstract SortByCurrentCountAscending : unit -> unit  
override SortByCurrentCountAscending : unit -> unit
```

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.SortByCurrentCountDescending Method

The implementation to sort a collection of [Toys](#) by the current count of the toy in descending order.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void SortByCurrentCountDescending()
```

VB

```
Protected Overridable Sub SortByCurrentCountDescending
```

C++

```
protected:  
virtual void SortByCurrentCountDescending()
```

F#

```
abstract SortByCurrentCountDescending : unit -> unit  
override SortByCurrentCountDescending : unit -> unit
```

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.SortByItemCode Method

Sorts the collection of [IToy](#) by item code in the specified sort order..

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SortByItemCode(  
    SortOrder order  
)
```

VB

```
Public Sub SortByItemCode (  
    order As SortOrder  
)
```

C++

```
public:  
virtual void SortByItemCode(  
    SortOrder order  
) sealed
```

F#

```
abstract SortByItemCode :  
    order : SortOrder -> unit  
override SortByItemCode :  
    order : SortOrder -> unit
```

Parameters

order

Type: [Woodstocks.WoodstocksIMS.Domain.SortOrder](#)

The sort order.

Implements

[IToys.SortByItemCode\(SortOrder\)](#)

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.SortByOnOrder Method

Sorts the collection of [IToy](#) by on order status in the specified sort order.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SortByOnOrder(  
    SortOrder order  
)
```

VB

```
Public Sub SortByOnOrder (  
    order As SortOrder  
)
```

C++

```
public:  
virtual void SortByOnOrder(  
    SortOrder order  
) sealed
```

F#

```
abstract SortByOnOrder :  
    order : SortOrder -> unit  
override SortByOnOrder :  
    order : SortOrder -> unit
```

Parameters

order

Type: [Woodstocks.WoodstocksIMS.Domain.SortOrder](#)

The sort order.

Implements

[IToys.SortByOnOrder\(SortOrder\)](#)

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.SortByOnOrderAscending Method

The implementation to sort a collection of [Toys](#) by on order status in ascending order.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected void SortByOnOrderAscending()
```

VB

```
Protected Sub SortByOnOrderAscending
```

C++

```
protected:  
void SortByOnOrderAscending()
```

F#

```
member SortByOnOrderAscending : unit -> unit
```

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.SortByOnOrderDescending Method

The implementation to sort a collection of [Toys](#) by on order status in descending order.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected void SortByOnOrderDescending()
```

VB

```
Protected Sub SortByOnOrderDescending
```

C++

```
protected:  
void SortByOnOrderDescending()
```

F#

```
member SortByOnOrderDescending : unit -> unit
```

See Also







[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Toys Properties

The [Toys](#) type exposes the following members.

Properties

	Name	Description
	Count	The total number of IToys in the collection.
	IsFixedSize	Gets a value indicating whether the IList has a fixed size.
	IsReadOnly	Gets a value indicating whether the ICollection(T) is read-only.
	IsSynchronized	Gets a value indicating whether access to the ICollection is synchronized (thread safe).
	Item	Gets the IToy located at the indexed position specified by index.
	SyncRoot	Gets an object that can be used to synchronize access to the ICollection .

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Count Property

The total number of [IToys](#) in the collection.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Public ReadOnly Property Count As Integer  
    Get
```

C++

```
public:  
virtual property int Count {  
    int get () sealed;  
}
```

F#

```
abstract Count : int with get  
override Count : int with get
```

Return Value

Type: [Int32](#)

The number of elements contained in the [ICollection\(T\)](#).

Implements

[IToys.Count](#)

[ICollection\(T\).Count](#)

[ICollection.Count](#)

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.IsFixedSize Property

Gets a value indicating whether the [IList](#) has a fixed size.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool IsFixedSize { get; }
```

VB

```
Public ReadOnly Property IsFixedSize As Boolean  
    Get
```

C++

```
public:  
virtual property bool IsFixedSize {  
    bool get () sealed;  
}
```

F#

```
abstract IsFixedSize : bool with get  
override IsFixedSize : bool with get
```

Return Value

Type: [Boolean](#)

true if the [IList](#) has a fixed size; otherwise, false.

Implements

[IList.IsFixedSize](#)

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.IsReadOnly Property

Gets a value indicating whether the [ICollection\(T\)](#) is read-only.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool IsReadOnly { get; }
```

VB

```
Public ReadOnly Property IsReadOnly As Boolean  
    Get
```

C++

```
public:  
virtual property bool IsReadOnly {  
    bool get () sealed;  
}
```

F#

```
abstract IsReadOnly : bool with get  
override IsReadOnly : bool with get
```

Return Value

Type: [Boolean](#)

true if the [ICollection\(T\)](#) is read-only; otherwise, false.

Implements

[ICollection\(T\).IsReadOnly](#)

[IList.IsReadOnly](#)

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.IsSynchronized Property

Gets a value indicating whether access to the [ICollection](#) is synchronized (thread safe).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool IsSynchronized { get; }
```

VB

```
Public ReadOnly Property IsSynchronized As Boolean  
    Get
```

C++

```
public:  
virtual property bool IsSynchronized {  
    bool get () sealed;  
}
```

F#

```
abstract IsSynchronized : bool with get  
override IsSynchronized : bool with get
```

Return Value

Type: [Boolean](#)

true if access to the [ICollection](#) is synchronized (thread safe); otherwise, false.

Implements

[ICollection.IsSynchronized](#)

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Item Property

Gets the [IToy](#) located at the indexed position specified by index.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Public Default Property Item (
    index As Integer
) As IToy
    Get
    Set
```

C++

```
public:
virtual property IToy^ default[int index] {
    IToy^ get (int index) sealed;
    void set (int index, IToy^ value) sealed;
}
```

F#

```
abstract Item : IToy with get, set
override Item : IToy with get, set
```

Parameters

index

Type: [System.Int32](#)

The zero-based index to the collection for the position of the [IToy](#) to be retrieved from the collection.

Return Value

Type: [IToy](#)

The [IToy](#)

Implements

[IToys.Item\(Int32\)](#)

[IList\(T\).Item\(Int32\)](#)

Exceptions

Exception	Condition
ArgumentOutOfRangeException	<i>index</i> is not a valid index in the IList(T) .
NotSupportedException	The property is set and the IList(T) is read-only.

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.SyncRoot Property

Gets an object that can be used to synchronize access to the [ICollection](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public Object SyncRoot { get; }
```

VB

```
Public ReadOnly Property SyncRoot As Object  
    Get
```

C++

```
public:  
virtual property Object^ SyncRoot {  
    Object^ get () sealed;  
}
```

F#

```
abstract SyncRoot : Object with get  
override SyncRoot : Object with get
```

Return Value

Type: [Object](#)

An object that can be used to synchronize access to the [ICollection](#).

Implements

[ICollection.SyncRoot](#)

Remarks

This method is not implemented because it is not currently needed in the solution.

See Also


[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.Toys Events

The [Toys](#) type exposes the following members.

Events

	Name	Description
	ToyChanged	Event raised when an item in the collection changes.

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

Toys.ToyChanged Event

Event raised when an item in the collection changes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event ToyChangedEventHandler ToyChanged
```

VB

```
Public Event ToyChanged As ToyChangedEventHandler
```

C++

```
public:
virtual event ToyChangedEventHandler^ ToyChanged {
    void add (ToyChangedEventHandler^ value);
    void remove (ToyChangedEventHandler^ value);
}
```

F#

```
abstract ToyChanged : IEvent<ToyChangedEventHandler,
    EventArgs>
override ToyChanged : IEvent<ToyChangedEventHandler,
    EventArgs>
```

Value

Type: [Woodstocks.WoodstocksIMS.Domain.ToyChangedEventHandler](#)

Implements

[IToys.ToyChanged](#)

See Also

[Toys Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

UnsavedDataException Class

Exception that occurs when stock data imported into [WoodstocksIMS](#) will be discarded without changes being saved.

Inheritance Hierarchy

[System.Object](#)

[System.Exception](#)

Woodstocks.WoodstocksIMS.Domain.UnsavedDataException

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

```
C#  
public class UnsavedDataException : Exception
```


```
VB  
Public Class UnsavedDataException  
    Inherits Exception
```

```
C++  
public ref class UnsavedDataException : public Exception
```




```
F#  
type UnsavedDataException =  
    class  
        inherit Exception  
    end
```






The **UnsavedDataException** type exposes the following members.

Constructors









	Name	Description
	UnsavedDataException	Initializes a new instance of the UnsavedDataException class

Methods


	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause

		of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

UnsavedDataException Constructor

Initializes a new instance of the [UnsavedDataException](#) class

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public UnsavedDataException()
```

VB

```
Public Sub New
```

C++

```
public:  
UnsavedDataException()
```

F#

```
new : unit -> UnsavedDataException
```

See Also









[UnsavedDataException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

UnsavedDataException.UnsavedDataException Methods

The [UnsavedDataException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

See Also









[UnsavedDataException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

UnsavedDataException.UnsavedDataException Properties

The [UnsavedDataException](#) type exposes the following members.

Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

See Also


[UnsavedDataException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

UnsavedDataException.UnsavedDataException Events

The [UnsavedDataException](#) type exposes the following members.

Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

See Also

[UnsavedDataException Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksDataConverter Class

Contains data conversion methods.

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Domain.WoodstocksDataConverter

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public static class WoodstocksDataConverter
```

VB

```
Public NotInheritable Class WoodstocksDataConverter
```

C++



```
public ref class WoodstocksDataConverter abstract sealed
```

F#

```
[<AbstractClassAttribute>]  
[<SealedAttribute>]  
type WoodstocksDataConverter = class end
```

The **WoodstocksDataConverter** type exposes the following members.

Methods

	Name	Description
	OnOrderToString	Converts an OnOrder value to a string value.
	ToOnOrder	Converts a value string into a OnOrder value.



See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksDataConverter.WoodstocksDataConverter Methods

The [WoodstocksDataConverter](#) type exposes the following members.

Methods

	Name	Description
	OnOrderToString	Converts an OnOrder value to a string value.
	ToOnOrder	Converts a value string into a OnOrder value.

See Also

[WoodstocksDataConverter Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksDataConverter.OnOrderToString Method

Converts an OnOrder value to a string value.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public static string OnOrderToString(  
    OnOrder onOrder  
)
```

VB

```
Public Shared Function OnOrderToString (  
    onOrder As OnOrder  
) As String
```

C++

```
public:  
static String^ OnOrderToString(  
    OnOrder onOrder  
)
```

F#

```
static member OnOrderToString :  
    onOrder : OnOrder -> string
```

Parameters

onOrder

Type: [Woodstocks.WoodstocksIMS.Domain.OnOrder](#)

The [OnOrder](#) value that is to be converted to an string value.

Return Value

Type: [String](#)

"Yes" if the OnOrder value is [Yes](#), "No" if the OnOrder value is [No](#)

Remarks

This method throws an [ArgumentException](#) if the argument passed to the method is not a value of [OnOrder](#).

See Also

[WoodstocksDataConverter Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksDataConverter.ToOnOrder Method

Converts a value string into a OnOrder value.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public static OnOrder ToOnOrder(  
    string value  
)
```

VB

```
Public Shared Function ToOnOrder (  
    value As String  
) As OnOrder
```

C++

```
public:  
static OnOrder ToOnOrder(  
    String^ value  
)
```

F#

```
static member ToOnOrder :  
    value : string -> OnOrder
```

Parameters

value

Type: [System.String](#)

The value that is to be converted to an OnOrder value.

Return Value

Type: [OnOrder](#)

The corresponding [OnOrder](#) value of the string if the value can be converted successfully. Throws an `InvalidCastException` if the value cannot be converted.

See Also

[WoodstocksDataConverter Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS Class

Implementation of the Wood Stocks Inventory Management System. The Wood Stocks Inventory Management System.

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Domain.WoodstocksIMS

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class WoodstocksIMS : IWoodstocksIMS,
    IWoodstocksIMSClient
```

VB

```
Public Class WoodstocksIMS
    Implements IWoodstocksIMS, IWoodstocksIMSClient
```

C++






```
public ref class WoodstocksIMS : IWoodstocksIMS,
    IWoodstocksIMSClient
```



























F#


```
type WoodstocksIMS =
    class
        interface IWoodstocksIMS
        interface IWoodstocksIMSClient
    end
```

The **WoodstocksIMS** type exposes the following members.




Methods

	Name	Description
	CancelAsync	Cancels an asynchronous operation.
	DiscardImportedToyData()	Discards imported toy data from the system.
	DiscardImportedToyData(Boolean)	The system facing method that is used to discard imported toy data from the system.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	ExportToysAsync	Exports modified toy data from the system.




	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetApplication	Creates the WoodstocksIMS for the application when called if not null and returns a client interface reference to the system.
	GetExportingState	Get the Exporting state of the system
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetIdleState	Get the Idle state of the system.
	GetImportingState	Get the Importing state of the system.
	GetModifiedToys	Gets the toy data that has been modified and has not been saved.
	GetToyDataSource	Gets the source from which the system will, or has, imported toy data.
	GetToyExporter	Gets the toy exporter used for exporting data.
	GetToyImporter	Gets the toy importer of the system.
	GetToys	Gets the toy data currently imported into the WoodstocksIMS .
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	GetUnsavedChanges	Gets whether the IWoodstocksIMS has imported stock data that has been modified but has not been saved.
	ImportToysAsync	Imports toy data into the WoodstocksIMS for use by the system.
	IsBusy	Indicates if the WoodstocksIMS is busy carrying out an asynchronous operation.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnExportCompleted	Raises the ExportCompleted event of the WoodstocksIMS .
	OnImportCompleted	Raises the ImportCompleted event of the WoodstocksIMS .
	OnProgressChanged	Raises the ProgressChanged event of the WoodstocksIMS .
	SetModifiedToys	Sets the toy data that has been modified and has not been saved.
	SetState	Set the current state of the system.
	SetToyDataSource	Sets the source from which the system will, or has, imported toy data.
	SetToyExporter	Sets the toy exporter used for exporting data.
	SetToyImporter	Sets the toy importer of the system.
	SetToys	Sets the toy data in use by the WoodstocksIMS .
	ToString	Returns a string that represents the current object. (Inherited

		from Object .)
	UnsavedChanges	Gets whether the WoodstocksIMS has data imported that is modified and has not been saved

Properties

	Name	Description
	ToyDataSource	Gets and Sets the data source from which toy data should be imported.
	ToyImporter	Gets and Sets the IWoodstocksToyImporter to be used by the system to import toy data.
	Toys	Retrieves IToys which references Wood Stocks toy data after importation.

Events

	Name	Description
	ExportCompleted	Raised when an asynchronous export completes.
	ImportCompleted	Raised when an asynchronous import completes.
	ProgressChanged	Raised when progress on an asynchronous operation is made.























See Also











[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.WoodstocksIMS Methods

The [WoodstocksIMS](#) type exposes the following members.

Methods

	Name	Description
	CancelAsync	Cancels an asynchronous operation.
	DiscardImportedToyData()	Discards imported toy data from the system.
	DiscardImportedToyData(Boolean)	The system facing method that is used to discard imported toy data from teh system.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	ExportToysAsync	Exports modified toy data from the system.
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetApplication	Creates the WoodstocksIMS for the application when called if not null and returns a client interface reference to the system.
	GetExportingState	Get the Exporting state of the system
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetIdleState	Get the Idle state of the system.
	GetImportingState	Get the Importing state of the system.
	GetModifiedToys	Gets the toy data that has been modified and has not been saved.
	GetToyDataSource	Gets the source from which the system will, or has, imported toy data.
	GetToyExporter	Gets the toy exporter used for exporting data.
	GetToyImporter	Gets the toy importer of the system.
	GetToys	Gets the toy data currently imported into the WoodstocksIMS .
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	GetUnsavedChanges	Gets whether the IWoodstocksIMS has imported stock data that has been modified but has not been saved.
	ImportToysAsync	Imports toy data into the WoodstocksIMS for use by the system.
	IsBusy	Indicates if the WoodstocksIMS is busy carrying out an asynchronous operation.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnExportCompleted	Raises the ExportCompleted event of the WoodstocksIMS .

	OnImportCompleted	Raises the ImportCompleted event of the WoodstocksIMS .
	OnProgressChanged	Raises the ProgressChanged event of the WoodstocksIMS .
	SetModifiedToys	Sets the toy data that has been modified and has not been saved.
	SetState	Set the current state of the system.
	SetToyDataSource	Sets the source from which the system will, or has, imported toy data.
	SetToyExporter	Sets the toy exporter used for exporting data.
	SetToyImporter	Sets the toy importer of the system.
	SetToys	Sets the toy data in use by the WoodstocksIMS .
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	UnsavedChanges	Gets whether the WoodstocksIMS has data imported that is modified and has not been saved

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.CancelAsync Method

Cancels an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void CancelAsync()
```

VB

```
Public Sub CancelAsync
```

C++

```
public:  
virtual void CancelAsync() sealed
```

F#

```
abstract CancelAsync : unit -> unit  
override CancelAsync : unit -> unit
```

Implements

[IWoodstocksIMSClient.CancelAsync\(\)](#)



See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstockslMS.DiscardImportedToyData Method

Overload List

	Name	Description
	DiscardImportedToyData()	Discards imported toy data from the system.
	DiscardImportedToyData(Boolean)	The system facing method that is used to discard imported toy data from teh system.

See Also

[WoodstockslMS Class](#)

[Woodstocks.WoodstockslMS.Domain Namespace](#)

WoodstocksIMS.DiscardImportedToyData Method

Discards imported toy data from the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void DiscardImportedToyData()
```

VB

```
Public Sub DiscardImportedToyData
```

C++

```
public:  
virtual void DiscardImportedToyData() sealed
```

F#

```
abstract DiscardImportedToyData : unit -> unit  
override DiscardImportedToyData : unit -> unit
```

Implements

[IWoodstocksIMSClient.DiscardImportedToyData\(\)](#)

See Also

[WoodstocksIMS Class](#)

[DiscardImportedToyData Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstockIMS.DiscardImportedToyData Method (Boolean)

The system facing method that is used to discard imported toy data from teh system.

Namespace: [Woodstocks.WoodstockIMS.Domain](#)

Assembly: WoodstockIMSLib (in WoodstockIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void DiscardImportedToyData(  
    bool disposing  
)
```

VB

```
Public Overridable Sub DiscardImportedToyData (  
    disposing As Boolean  
)
```

C++

```
public:  
virtual void DiscardImportedToyData(  
    bool disposing  
)
```

F#

```
abstract DiscardImportedToyData :  
    disposing : bool -> unit  
override DiscardImportedToyData :  
    disposing : bool -> unit
```

Parameters

disposing

Type: [System.Boolean](#)

Indicates if the system should discard the data that is imported. If true, the data is discarded.

Implements

[IWoodstockIMS.DiscardImportedToyData\(Boolean\)](#)

See Also

[WoodstockIMS Class](#)

[DiscardImportedToyData Overload](#)

[Woodstocks.WoodstockIMS.Domain Namespace](#)

WoodstockslMS.ExportToysAsync Method

Exports modified toy data from the system.

Namespace: [Woodstocks.WoodstockslMS.Domain](#)

Assembly: WoodstockslMSLib (in WoodstockslMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ExportToysAsync()
```

VB

```
Public Sub ExportToysAsync
```

C++

```
public:  
virtual void ExportToysAsync() sealed
```

F#

```
abstract ExportToysAsync : unit -> unit  
override ExportToysAsync : unit -> unit
```

Implements

[IWoodstockslMSClient.ExportToysAsync\(\)](#)

See Also

[WoodstockslMS Class](#)

[Woodstocks.WoodstockslMS.Domain Namespace](#)

WoodstocksIMS.GetApplication Method

Creates the [WoodstocksIMS](#) for the application when called if not null and returns a client interface reference to the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public static IWoodstocksIMSClient GetApplication()
```

VB

```
Public Shared Function GetApplication As IWoodstocksIMSClient
```

C++

```
public:  
static IWoodstocksIMSClient^ GetApplication()
```

F#

```
static member GetApplication : unit -> IWoodstocksIMSClient
```

Return Value

Type: [IWoodstocksIMSClient](#)

A client interface to the [WoodstocksIMS](#).

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.GetExportingState Method

Get the Exporting state of the system

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public WoodstocksIMSState GetExportingState()
```

VB

```
Public Function GetExportingState As WoodstocksIMSState
```

C++

```
public:  
virtual WoodstocksIMSState^ GetExportingState() sealed
```

F#

```
abstract GetExportingState : unit -> WoodstocksIMSState  
override GetExportingState : unit -> WoodstocksIMSState
```

Return Value

Type: [WoodstocksIMSState](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMS.GetExportingState"]

Implements

[IWoodstocksIMS.GetExportingState\(\)](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.GetIdleState Method

Get the Idle state of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public WoodstocksIMSState GetIdleState()
```

VB

```
Public Function GetIdleState As WoodstocksIMSState
```

C++

```
public:  
virtual WoodstocksIMSState^ GetIdleState() sealed
```

F#

```
abstract GetIdleState : unit -> WoodstocksIMSState  
override GetIdleState : unit -> WoodstocksIMSState
```

Return Value

Type: [WoodstocksIMSState](#)

The idle state of the system.

Implements

[IWoodstocksIMS.GetIdleState\(\)](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.GetImportingState Method

Get the Importing state of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public WoodstocksIMSState GetImportingState()
```

VB

```
Public Function GetImportingState As WoodstocksIMSState
```

C++

```
public:  
virtual WoodstocksIMSState^ GetImportingState() sealed
```

F#

```
abstract GetImportingState : unit -> WoodstocksIMSState  
override GetImportingState : unit -> WoodstocksIMSState
```

Return Value

Type: [WoodstocksIMSState](#)

The Importing state of the system

Implements

[IWoodstocksIMS.GetImportingState\(\)](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.GetModifiedToys Method

Gets the toy data that has been modified and has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IToys GetModifiedToys()
```

VB

```
Public Function GetModifiedToys As IToys
```

C++

```
public:  
virtual IToys^ GetModifiedToys() sealed
```

F#

```
abstract GetModifiedToys : unit -> IToys  
override GetModifiedToys : unit -> IToys
```

Return Value

Type: [IToys](#)

A reference to the modified toy data.

Implements

[IWoodstocksIMS.GetModifiedToys\(\)](#)

Remarks

This method is intended as a system method and therefore does not appear on the client interface.

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.GetToyDataSource Method

Gets the source from which the system will, or has, imported toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public string GetToyDataSource()
```

VB

```
Public Function GetToyDataSource As String
```

C++

```
public:  
virtual String^ GetToyDataSource() sealed
```

F#

```
abstract GetToyDataSource : unit -> string  
override GetToyDataSource : unit -> string
```

Return Value

Type: [String](#)

The source of the toy data.

Implements

[IWoodstocksIMS.GetToyDataSource\(\)](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.GetToyExporter Method

Gets the toy exporter used for exporting data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IWoodstocksToyExporter GetToyExporter()
```

VB

```
Public Function GetToyExporter As IWoodstocksToyExporter
```

C++

```
public:  
virtual IWoodstocksToyExporter^ GetToyExporter() sealed
```

F#

```
abstract GetToyExporter : unit -> IWoodstocksToyExporter  
override GetToyExporter : unit -> IWoodstocksToyExporter
```

Return Value

Type: [IWoodstocksToyExporter](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMS.GetToyExporter"]

Implements

[IWoodstocksIMS.GetToyExporter\(\)](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.GetToyImporter Method

Gets the toy importer of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual IWoodstocksToyImporter GetToyImporter()
```

VB

```
Public Overridable Function GetToyImporter As IWoodstocksToyImporter
```

C++

```
public:  
virtual IWoodstocksToyImporter^ GetToyImporter()
```

F#

```
abstract GetToyImporter : unit -> IWoodstocksToyImporter  
override GetToyImporter : unit -> IWoodstocksToyImporter
```

Return Value

Type: [IWoodstocksToyImporter](#)

The systems toy importer.

Implements

[IWoodstocksIMS.GetToyImporter\(\)](#)

Remarks

This method is to be implemented to provide for an implementation of the ToyImporter property.

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.GetToys Method

Gets the toy data currently imported into the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IToys GetToys()
```

VB

```
Public Function GetToys As IToys
```

C++

```
public:  
virtual IToys^ GetToys() sealed
```

F#

```
abstract GetToys : unit -> IToys  
override GetToys : unit -> IToys
```

Return Value

Type: [IToys](#)

A reference to the imported toy data

Implements

[IWoodstocksIMS.GetToys\(\)](#)

Remarks

This method is intended as a system method and therefore does not appear on the client interface.

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.GetUnsavedChanges Method

Gets whether the [IWoodstocksIMS](#) has imported stock data that has been modified but has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool GetUnsavedChanges()
```

VB

```
Public Function GetUnsavedChanges As Boolean
```

C++

```
public:  
virtual bool GetUnsavedChanges() sealed
```

F#

```
abstract GetUnsavedChanges : unit -> bool  
override GetUnsavedChanges : unit -> bool
```

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for "M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMS.GetUnsavedChanges"]

Implements

[IWoodstocksIMS.GetUnsavedChanges\(\)](#)

Remarks

This method is to be implemented for "internal system" use. It exists, primarily, to allow state objects to return the result from the system to clients who have called the UnsavedChanges method on the client interface.

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.ImportToysAsync Method

Imports toy data into the [WoodstocksIMS](#) for use by the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ImportToysAsync(  
    string source  
)
```

VB

```
Public Sub ImportToysAsync (  
    source As String  
)
```

C++

```
public:  
virtual void ImportToysAsync(  
    String^ source  
) sealed
```

F#

```
abstract ImportToysAsync :  
    source : string -> unit  
override ImportToysAsync :  
    source : string -> unit
```

Parameters

source

Type: [System.String](#)

The source from which toy data should be retrieved.

Implements

[IWoodstocksIMSClient.ImportToysAsync\(String\)](#)

Remarks

The source is the filepath to the csv data file containing the Wood Stocks toy data.

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.IsBusy Method

Indicates if the [WoodstocksIMS](#) is busy carrying out an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool IsBusy()
```

VB

```
Public Function IsBusy As Boolean
```

C++

```
public:  
virtual bool IsBusy() sealed
```

F#

```
abstract IsBusy : unit -> bool  
override IsBusy : unit -> bool
```

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for "M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMS.IsBusy"]

Implements

[IWoodstocksIMSClient.IsBusy\(\)](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.OnExportCompleted Method

Raises the [ExportCompleted](#) event of the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void OnExportCompleted(  
    Object sender,  
    AsyncCompletedEventArgs e  
)
```

VB

```
Public Sub OnExportCompleted (  
    sender As Object,  
    e As AsyncCompletedEventArgs  
)
```

C++

```
public:  
virtual void OnExportCompleted(  
    Object^ sender,  
    AsyncCompletedEventArgs^ e  
) sealed
```

F#

```
abstract OnExportCompleted :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit  
override OnExportCompleted :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The [WoodstocksIMS](#) that raised the event.

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

The [AsyncCompletedEventArgs](#) data sent from the asynchronous thread that carried out the operation.

Implements

[IWoodstocksIMS.OnExportCompleted\(Object, AsyncCompletedEventArgs\)](#)

A Sandcastle Documented Class Library

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.OnImportCompleted Method

Raises the [ImportCompleted](#) event of the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void OnImportCompleted(  
    Object sender,  
    AsyncCompletedEventArgs e  
)
```

VB

```
Public Sub OnImportCompleted (  
    sender As Object,  
    e As AsyncCompletedEventArgs  
)
```

C++

```
public:  
virtual void OnImportCompleted(  
    Object^ sender,  
    AsyncCompletedEventArgs^ e  
) sealed
```

F#

```
abstract OnImportCompleted :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit  
override OnImportCompleted :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The [WoodstocksIMS](#) that raised the event.

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

The [AsyncCompletedEventArgs](#) data sent from the asynchronous thread that carried out the operation.

Implements

[IWoodstocksIMS.OnImportCompleted\(Object, AsyncCompletedEventArgs\)](#)

A Sandcastle Documented Class Library

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.OnProgressChanged Method

Raises the [ProgressChanged](#) event of the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void OnProgressChanged(  
    Object sender,  
    ProgressChangedEventArgs e  
)
```

VB

```
Public Sub OnProgressChanged (  
    sender As Object,  
    e As ProgressChangedEventArgs  
)
```

C++

```
public:  
virtual void OnProgressChanged(  
    Object^ sender,  
    ProgressChangedEventArgs^ e  
) sealed
```

F#

```
abstract OnProgressChanged :  
    sender : Object *  
    e : ProgressChangedEventArgs -> unit  
override OnProgressChanged :  
    sender : Object *  
    e : ProgressChangedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The [WoodstocksIMS](#) that raised the event.

e

Type: [System.ComponentModel.ProgressChangedEventArgs](#)

The [ProgressChangedEventArgs](#) data for the event.

Implements

[IWoodstocksIMS.OnProgressChanged\(Object, ProgressChangedEventArgs\)](#)

A Sandcastle Documented Class Library

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.SetModifiedToys Method

Sets the toy data that has been modified and has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SetModifiedToys(  
    IToys modified  
)
```

VB

```
Public Sub SetModifiedToys (  
    modified As IToys  
)
```

C++

```
public:  
virtual void SetModifiedToys(  
    IToys^ modified  
) sealed
```

F#

```
abstract SetModifiedToys :  
    modified : IToys -> unit  
override SetModifiedToys :  
    modified : IToys -> unit
```

Parameters

modified

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The modified toy data.

Return Value

Type:

A reference to the modified toy data.

Implements

[IWoodstocksIMS.SetModifiedToys\(IToys\)](#)

Remarks

This method is intended as a system method and therefore does not appear on the client interface.

See Also

[WoodstocksIMS Class](#)

A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.SetState Method

Set the current state of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SetState(  
    WoodstocksIMSState newState  
)
```

VB

```
Public Sub SetState (  
    newState As WoodstocksIMSState  
)
```

C++

```
public:  
virtual void SetState(  
    WoodstocksIMSState^ newState  
) sealed
```

F#

```
abstract SetState :  
    newState : WoodstocksIMSState -> unit  
override SetState :  
    newState : WoodstocksIMSState -> unit
```

Parameters

newState

Type: [Woodstocks.WoodstocksIMS.Domain.WoodstocksIMSState](#)

[Missing <param name="newState"/> documentation for "M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMS.SetState(Woodstocks.WoodstocksIMS.Domain.WoodstocksIMSState)"]

Implements

[IWoodstocksIMS.SetState\(WoodstocksIMSState\)](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.SetToyDataSource Method

Sets the source from which the system will, or has, imported toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SetToyDataSource(  
    string source  
)
```

VB

```
Public Sub SetToyDataSource (  
    source As String  
)
```

C++

```
public:  
virtual void SetToyDataSource(  
    String^ source  
) sealed
```

F#

```
abstract SetToyDataSource :  
    source : string -> unit  
override SetToyDataSource :  
    source : string -> unit
```

Parameters

source

Type: [System.String](#)

The source of the toy data.

Implements

[IWoodstocksIMS.SetToyDataSource\(String\)](#)

Remarks

This method is intended as a system method and therefore does not appear on the client interface.

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.SetToyExporter Method

Sets the toy exporter used for exporting data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SetToyExporter(  
    IWoodstocksToyExporter toyExporter  
)
```

VB

```
Public Sub SetToyExporter (  
    toyExporter As IWoodstocksToyExporter  
)
```

C++

```
public:  
virtual void SetToyExporter(  
    IWoodstocksToyExporter^ toyExporter  
) sealed
```

F#

```
abstract SetToyExporter :  
    toyExporter : IWoodstocksToyExporter -> unit  
override SetToyExporter :  
    toyExporter : IWoodstocksToyExporter -> unit
```

Parameters

toyExporter

Type: [Woodstocks.WoodstocksIMS.Domain.IWoodstocksToyExporter](#)

The exporter that the system should use to export toy data.

Implements

[IWoodstocksIMS.SetToyExporter\(IWoodstocksToyExporter\)](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.SetToyImporter Method

Sets the toy importer of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SetToyImporter(  
    IWoodstocksToyImporter toyImporter  
)
```

VB

```
Public Sub SetToyImporter (  
    toyImporter As IWoodstocksToyImporter  
)
```

C++

```
public:  
virtual void SetToyImporter(  
    IWoodstocksToyImporter^ toyImporter  
) sealed
```

F#

```
abstract SetToyImporter :  
    toyImporter : IWoodstocksToyImporter -> unit  
override SetToyImporter :  
    toyImporter : IWoodstocksToyImporter -> unit
```

Parameters

toyImporter

Type: [Woodstocks.WoodstocksIMS.Domain.IWoodstocksToyImporter](#)

The importer that the system should be set to use.

Implements

[IWoodstocksIMS.SetToyImporter\(IWoodstocksToyImporter\)](#)

Remarks

This method is to be implemented to provide for an implementation of the ToyImporter property.

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.SetToys Method

Sets the toy data in use by the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SetToys(  
    IToys toys  
)
```

VB

```
Public Sub SetToys (  
    toys As IToys  
)
```

C++

```
public:  
virtual void SetToys(  
    IToys^ toys  
) sealed
```

F#

```
abstract SetToys :  
    toys : IToys -> unit  
override SetToys :  
    toys : IToys -> unit
```

Parameters

toys

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The toy data to be used by the system.

Implements

[IWoodstocksIMS.SetToys\(IToys\)](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.UnsavedChanges Method

Gets whether the [WoodstocksIMS](#) has data imported that is modified and has not been saved

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool UnsavedChanges( )
```

VB

```
Public Function UnsavedChanges As Boolean
```

C++

```
public:  
virtual bool UnsavedChanges( ) sealed
```

F#

```
abstract UnsavedChanges : unit -> bool  
override UnsavedChanges : unit -> bool
```

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMS.UnsavedChanges"]

Implements

[IWoodstocksIMSClient.UnsavedChanges\(\)](#)

See Also




[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.WoodstocksIMS Properties

The [WoodstocksIMS](#) type exposes the following members.

Properties

	Name	Description
	ToyDataSource	Gets and Sets the data source from which toy data should be imported.
	ToyImporter	Gets and Sets the IWoodstocksToyImporter to be used by the system to import toy data.
	Toys	Retrieves IToys which references Wood Stocks toy data after importation.

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.ToyDataSource Property

Gets and Sets the data source from which toy data should be imported.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Public Property ToyDataSource As String  
    Get  
    Set
```

C++

```
public:  
virtual property String^ ToyDataSource {  
    String^ get () sealed;  
    void set (String^ value) sealed;  
}
```

F#

```
abstract ToyDataSource : string with get, set  
override ToyDataSource : string with get, set
```

Property Value

Type: [String](#)

Implements

[IWoodstocksIMSClient.ToyDataSource](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.ToyImporter Property

Gets and Sets the [IWoodstocksToyImporter](#) to be used by the system to import toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IWoodstocksToyImporter ToyImporter { get; set; }
```

VB

```
Public Property ToyImporter As IWoodstocksToyImporter
    Get
    Set
```

C++

```
public:
virtual property IWoodstocksToyImporter^ ToyImporter {
    IWoodstocksToyImporter^ get () sealed;
    void set (IWoodstocksToyImporter^ value) sealed;
}
```

F#

```
abstract ToyImporter : IWoodstocksToyImporter with get, set
override ToyImporter : IWoodstocksToyImporter with get, set
```

Property Value

Type: [IWoodstocksToyImporter](#)

Implements

[IWoodstocksIMS.ToyImporter](#)

Remarks

This property is intended as a system property that that is unavailable to clients of the system. As a result it does not appear on the client interface.

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.Toys Property

Retrieves [IToys](#) which references Wood Stocks toy data after importation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IToys Toys { get; }
```

VB

```
Public ReadOnly Property Toys As IToys  
    Get
```

C++

```
public:  
virtual property IToys^ Toys {  
    IToys^ get () sealed;  
}
```

F#

```
abstract Toys : IToys with get  
override Toys : IToys with get
```

Return Value

Type: [IToys](#)

Implements

[IWoodstocksIMSCient.Toys](#)

See Also




[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstockslMS.WoodstockslMS Events

The [WoodstockslMS](#) type exposes the following members.

Events

	Name	Description
	ExportCompleted	Raised when an asynchronous export completes.
	ImportCompleted	Raised when an asynchronous import completes.
	ProgressChanged	Raised when progress on an asynchronous operation is made.

See Also

[WoodstockslMS Class](#)

[Woodstocks.WoodstockslMS.Domain Namespace](#)

WoodstocksIMS.ExportCompleted Event

Raised when an asynchronous export completes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event AsyncCompletedEventHandler ExportCompleted
```

VB

```
Public Event ExportCompleted As AsyncCompletedEventHandler
```

C++

```
public:  
virtual event AsyncCompletedEventHandler^ ExportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>  
override ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

Implements

[IWoodstocksIMSClient.ExportCompleted](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.ImportCompleted Event

Raised when an asynchronous import completes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event AsyncCompletedEventHandler ImportCompleted
```

VB

```
Public Event ImportCompleted As AsyncCompletedEventHandler
```

C++

```
public:  
virtual event AsyncCompletedEventHandler^ ImportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ImportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>  
override ImportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

Implements

[IWoodstocksIMSClient.ImportCompleted](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMS.ProgressChanged Event

Raised when progress on an asynchronous operation is made.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event ProgressChangedEventHandler ProgressChanged
```

VB

```
Public Event ProgressChanged As ProgressChangedEventHandler
```

C++

```
public:  
virtual event ProgressChangedEventHandler^ ProgressChanged {  
    void add (ProgressChangedEventHandler^ value);  
    void remove (ProgressChangedEventHandler^ value);  
}
```

F#

```
abstract ProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>  
override ProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>
```

Value

Type: [System.ComponentModel.ProgressChangedEventHandler](#)

Implements

[IWoodstocksIMSClient.ProgressChanged](#)

See Also

[WoodstocksIMS Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState Class

An abstract class to define state objects for the [WoodstocksIMS](#).

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Domain.WoodstocksIMSState

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public abstract class WoodstocksIMSState : IWoodstocksIMS,
    IWoodstocksIMSClient
```

VB

```
Public MustInherit Class WoodstocksIMSState
    Implements IWoodstocksIMS, IWoodstocksIMSClient
```

C++


```
public ref class WoodstocksIMSState abstract : IWoodstocksIMS,
    IWoodstocksIMSClient
```

F#




```
[<AbstractClassAttribute>]
type WoodstocksIMSState =
    class
        interface IWoodstocksIMS
        interface IWoodstocksIMSClient
    end
```



























The **WoodstocksIMSState** type exposes the following members.



Constructors

	Name	Description
	WoodstocksIMSState	Initialises a WoodstocksIMSState .




Methods

	Name	Description
	CancelAsync	Cancels an asynchronous operation.
	DiscardImportedToyData()	Discards imported toy data from the system.
	DiscardImportedToyData(Boolean)	Discards imported data from the system. This method is defined




		as a system facing interface method.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	ExportToysAsync	Exports modified toy data from the system.
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetExportingState	Get the Exporting state of the system
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetIdleState	Get the Idle state of the system.
	GetImportingState	Get the Importing state of the system.
	GetModifiedToys	Gets the toy data that has been modified and has not been saved.
	GetToyDataSource	Gets the source from which the system will, or has, imported toy data.
	GetToyExporter	Gets the toy exporter used for exporting data.
	GetToyImporter	Gets the toy importer of the system.
	GetToys	Gets the toy data currently imported into the WoodstocksIMS .
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	GetUnsavedChanges	Gets whether the IWoodstocksIMS has imported stock data that has been modified but has not been saved.
	ImportToysAsync	Imports toy data into the WoodstocksIMS for use by the system.
	IsBusy	
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnExportCompleted	Raises the ExportCompleted event of the IWoodstocksIMS .
	OnImportCompleted	Raises the ImportCompleted event of the IWoodstocksIMS .
	OnProgressChanged	Raises the ProgressChanged event of the IWoodstocksIMS to indicate that progress of an asynchronous operation.
	SetModifiedToys	Sets the toy data that has been modified and has not been saved.
	SetState	Set the current state of the system.
	SetToyDataSource	Sets the source from which the system will, or has, imported toy data.
	SetToyExporter	Sets the toy exporter used for exporting data.
	SetToyImporter	Sets the toy importer of the system.
	SetToys	Sets the toy data in use by the WoodstocksIMS .

	ToString	Returns a string that represents the current object. (Inherited from Object .)
	UnsavedChanges	Gets whether the toy data contains unsaved changes.

Properties

	Name	Description
	ToyDataSource	Gets and Sets the data source from which toy data should be imported.
	ToyImporter	Gets and Sets the IWoodstocksToyImporter to be used by the system to import toy data.
	Toys	Retrieves IToys which references Wood Stocks toy data after importation.

Events

	Name	Description
	ExportCompleted	Raised when an asynchronous export completes.
	ImportCompleted	Raised when an asynchronous import completes.
	ProgressChanged	Raised when progress on an asynchronous operation is made.

Remarks

A state object for the [WoodstocksIMS](#) defines the functionality of the system according systems state. It is intended that derived classes of this class define a specific set of functionality for a particular state. For example the **IdleState** defines the systems functionality when the system is in an idle state.

See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState Constructor

Initialises a [WoodstocksIMSState](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public WoodstocksIMSState(  
    IWoodstocksIMS woodstocksIMS  
)
```

VB

```
Public Sub New (  
    woodstocksIMS As IWoodstocksIMS  
)
```

C++

```
public:  
WoodstocksIMSState(  
    IWoodstocksIMS^ woodstocksIMS  
)
```

F#

```
new :  
    woodstocksIMS : IWoodstocksIMS -> WoodstocksIMSState
```

Parameters

woodstocksIMS

Type: [Woodstocks.WoodstocksIMS.Domain.IWoodstocksIMS](#)

A reference to the [WoodstocksIMS](#).

See Also

[WoodstocksIMSState Class](#)









[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.WoodstocksIMSState Methods

The [WoodstocksIMSState](#) type exposes the following members.

Methods

	Name	Description
	CancelAsync	Cancels an asynchronous operation.
	DiscardImportedToyData()	Discards imported toy data from the system.
	DiscardImportedToyData(Boolean)	Discards imported data from the system. This method is defined as a system facing interface method.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	ExportToysAsync	Exports modified toy data from the system.
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetExportingState	Get the Exporting state of the system
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetIdleState	Get the Idle state of the system.
	GetImportingState	Get the Importing state of the system.
	GetModifiedToys	Gets the toy data that has been modified and has not been saved.
	GetToyDataSource	Gets the source from which the system will, or has, imported toy data.
	GetToyExporter	Gets the toy exporter used for exporting data.
	GetToyImporter	Gets the toy importer of the system.
	GetToys	Gets the toy data currently imported into the WoodstocksIMS .
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	GetUnsavedChanges	Gets whether the IWoodstocksIMS has imported stock data that has been modified but has not been saved.
	ImportToysAsync	Imports toy data into the WoodstocksIMS for use by the system.
	IsBusy	
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	OnExportCompleted	Raises the ExportCompleted event of the IWoodstocksIMS .
	OnImportCompleted	Raises the ImportCompleted event of the IWoodstocksIMS .
	OnProgressChanged	Raises the ProgressChanged event of the IWoodstocksIMS to indicate that progress of an asynchronous operation.

	SetModifiedToys	Sets the toy data that has been modified and has not been saved.
	SetState	Set the current state of the system.
	SetToyDataSource	Sets the source from which the system will, or has, imported toy data.
	SetToyExporter	Sets the toy exporter used for exporting data.
	SetToyImporter	Sets the toy importer of the system.
	SetToys	Sets the toy data in use by the WoodstocksIMS .
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	UnsavedChanges	Gets whether the toy data contains unsaved changes.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.CancelAsync Method

Cancels an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void CancelAsync()
```

VB

```
Public Overridable Sub CancelAsync
```

C++

```
public:  
virtual void CancelAsync()
```

F#

```
abstract CancelAsync : unit -> unit  
override CancelAsync : unit -> unit
```

Implements

[IWoodstocksIMSClient.CancelAsync\(\)](#)

Remarks

The default implementation is to throw an [InvalidOperationException](#). This is to ensure that the system will only allow states in which a cancellation can be performed will occur. If a system state should allow cancellation then this method should be overridden in the derived state class.



See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.DiscardImportedToyData Method

Overload List

	Name	Description
	DiscardImportedToyData()	Discards imported toy data from the system.
	DiscardImportedToyData(Boolean)	Discards imported data from the system. This method is defined as a system facing interface method.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.DiscardImportedToyData Method

Discards imported toy data from the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void DiscardImportedToyData()
```

VB

```
Public Overridable Sub DiscardImportedToyData
```

C++

```
public:  
virtual void DiscardImportedToyData()
```

F#

```
abstract DiscardImportedToyData : unit -> unit  
override DiscardImportedToyData : unit -> unit
```

Implements

[IWoodstocksIMSClient.DiscardImportedToyData\(\)](#)

Remarks

The default implementation throws the `InvalidOperationException` so as to ensure states that allow discarding of imported data provide the required implementation.

See Also

[WoodstocksIMSState Class](#)

[DiscardImportedToyData Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.DiscardImportedToyData Method (Boolean)

Discards imported data from the system. This method is defined as a system facing interface method.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void DiscardImportedToyData(  
    bool disposing  
)
```

VB

```
Public Overridable Sub DiscardImportedToyData (  
    disposing As Boolean  
)
```

C++

```
public:  
virtual void DiscardImportedToyData(  
    bool disposing  
)
```

F#

```
abstract DiscardImportedToyData :  
    disposing : bool -> unit  
override DiscardImportedToyData :  
    disposing : bool -> unit
```

Parameters

disposing

Type: [System.Boolean](#)

Indicates that the data should actually be disposed of.

Implements

[IWoodstocksIMS.DiscardImportedToyData\(Boolean\)](#)

Remarks

The default implementation throws the `InvalidOperationException` so as to ensure states that allow discarding of imported data provide the required implementation.

See Also

[WoodstocksIMSState Class](#)

[DiscardImportedToyData Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.ExportToysAsync Method

Exports modified toy data from the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void ExportToysAsync( )
```

VB

```
Public Overridable Sub ExportToysAsync
```

C++

```
public:  
virtual void ExportToysAsync( )
```

F#

```
abstract ExportToysAsync : unit -> unit  
override ExportToysAsync : unit -> unit
```

Implements

[IWoodstocksIMSClient.ExportToysAsync\(\)](#)

Remarks

The default implementation provided by this method is to throw an [InvalidOperationException](#). This is to ensure that the system will only allow exportation to be commenced whilst the system is in a state that allows exportation. If a particular system state should allow importation then this class should be overridden in the derived state class.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.GetExportingState Method

Get the Exporting state of the system

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public WoodstocksIMSState GetExportingState()
```

VB

```
Public Function GetExportingState As WoodstocksIMSState
```

C++

```
public:  
virtual WoodstocksIMSState^ GetExportingState() sealed
```

F#

```
abstract GetExportingState : unit -> WoodstocksIMSState  
override GetExportingState : unit -> WoodstocksIMSState
```

Return Value

Type: [WoodstocksIMSState](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMSState.GetExportingState"]

Implements

[IWoodstocksIMS.GetExportingState\(\)](#)

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.GetIdleState Method

Get the Idle state of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public WoodstocksIMSState GetIdleState()
```

VB

```
Public Function GetIdleState As WoodstocksIMSState
```

C++

```
public:  
virtual WoodstocksIMSState^ GetIdleState() sealed
```

F#

```
abstract GetIdleState : unit -> WoodstocksIMSState  
override GetIdleState : unit -> WoodstocksIMSState
```

Return Value

Type: [WoodstocksIMSState](#)

The idle state of the system.

Implements

[IWoodstocksIMS.GetIdleState\(\)](#)

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.GetImportingState Method

Get the Importing state of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public WoodstocksIMSState GetImportingState()
```

VB

```
Public Function GetImportingState As WoodstocksIMSState
```

C++

```
public:  
virtual WoodstocksIMSState^ GetImportingState() sealed
```

F#

```
abstract GetImportingState : unit -> WoodstocksIMSState  
override GetImportingState : unit -> WoodstocksIMSState
```

Return Value

Type: [WoodstocksIMSState](#)

The Importing state of the system

Implements

[IWoodstocksIMS.GetImportingState\(\)](#)

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.GetModifiedToys Method

Gets the toy data that has been modified and has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual IToys GetModifiedToys()
```

VB

```
Public Overridable Function GetModifiedToys As IToys
```

C++

```
public:  
virtual IToys^ GetModifiedToys()
```

F#

```
abstract GetModifiedToys : unit -> IToys  
override GetModifiedToys : unit -> IToys
```

Return Value

Type: [IToys](#)

A reference to the modified toy data.

Implements

[IWoodstocksIMS.GetModifiedToys\(\)](#)

Remarks

This method is intended as a system method and therefore does not appear on the client interface.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.GetToyDataSource Method

Gets the source from which the system will, or has, imported toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual string GetToyDataSource()
```

VB

```
Public Overridable Function GetToyDataSource As String
```

C++

```
public:  
virtual String^ GetToyDataSource()
```

F#

```
abstract GetToyDataSource : unit -> string  
override GetToyDataSource : unit -> string
```

Return Value

Type: [String](#)

The source of the toy data.

Implements

[IWoodstocksIMS.GetToyDataSource\(\)](#)

Remarks

The default implementation throws an [InvalidOperationException](#). This is to prevent a state by default from accessing the toy data source value. If a particular state should allow the toy data source value to be retrieved then the derived state class for the state should override this method to provide the required functionality.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.GetToyExporter Method

Gets the toy exporter used for exporting data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IWoodstocksToyExporter GetToyExporter()
```

VB

```
Public Function GetToyExporter As IWoodstocksToyExporter
```

C++

```
public:  
virtual IWoodstocksToyExporter^ GetToyExporter() sealed
```

F#

```
abstract GetToyExporter : unit -> IWoodstocksToyExporter  
override GetToyExporter : unit -> IWoodstocksToyExporter
```

Return Value

Type: [IWoodstocksToyExporter](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMSState.GetToyExporter"]

Implements

[IWoodstocksIMS.GetToyExporter\(\)](#)

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.GetToyImporter Method

Gets the toy importer of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual IWoodstocksToyImporter GetToyImporter()
```

VB

```
Public Overridable Function GetToyImporter As IWoodstocksToyImporter
```

C++

```
public:  
virtual IWoodstocksToyImporter^ GetToyImporter()
```

F#

```
abstract GetToyImporter : unit -> IWoodstocksToyImporter  
override GetToyImporter : unit -> IWoodstocksToyImporter
```

Return Value

Type: [IWoodstocksToyImporter](#)

The systems toy importer.

Implements

[IWoodstocksIMS.GetToyImporter\(\)](#)

Remarks

By default this method throws a [InvalidOperationException](#). This is to ensure that by default states cannot access the systems toy importer. If a particular state should have access to the toy importer then this method should be overridden by the derived state class that defines the state to provide the required functionality.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.GetToys Method

Gets the toy data currently imported into the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual IToys GetToys()
```

VB

```
Public Overridable Function GetToys As IToys
```

C++

```
public:  
virtual IToys^ GetToys()
```

F#

```
abstract GetToys : unit -> IToys  
override GetToys : unit -> IToys
```

Return Value

Type: [IToys](#)

A reference to the imported toy data

Implements

[IWoodstocksIMS.GetToys\(\)](#)

Remarks

The default implementation of this method throws an [InvalidOperationException](#). This is to ensure that by default states do not allow access to toy data. If a particular state should allow toy data to be accessed then this method should be overridden in the derived state class.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.GetUnsavedChanges Method

Gets whether the [IWoodstocksIMS](#) has imported stock data that has been modified but has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual bool GetUnsavedChanges ( )
```

VB

```
Public Overridable Function GetUnsavedChanges As Boolean
```

C++

```
public:  
virtual bool GetUnsavedChanges ( )
```

F#

```
abstract GetUnsavedChanges : unit -> bool  
override GetUnsavedChanges : unit -> bool
```

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMSState.GetUnsavedChanges"]

Implements

[IWoodstocksIMS.GetUnsavedChanges\(\)](#)

Remarks

The default implementation returns the result from the system by utilising the system implementation.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.ImportToysAsync Method

Imports toy data into the [WoodstocksIMS](#) for use by the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void ImportToysAsync(  
    string source  
)
```

VB

```
Public Overridable Sub ImportToysAsync (  
    source As String  
)
```

C++

```
public:  
virtual void ImportToysAsync(  
    String^ source  
)
```

F#

```
abstract ImportToysAsync :  
    source : string -> unit  
override ImportToysAsync :  
    source : string -> unit
```

Parameters

source

Type: [System.String](#)

The source from which toy data should be retrieved.

Implements

[IWoodstocksIMSClient.ImportToysAsync\(String\)](#)

Remarks

The default implementation provided by this method is to throw an [InvalidOperationException](#). This is to ensure that the system will only allow importation to be initiated whilst the system is in a state that will allow importation. If a particular system state should allow importation then this class should be overridden in the derived state class.

See Also

[WoodstocksIMSState Class](#)

A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.IsBusy Method

[Missing <summary> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMSState.IsBusy"]

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual bool IsBusy()
```

VB

```
Public Overridable Function IsBusy As Boolean
```

C++

```
public:  
virtual bool IsBusy()
```

F#

```
abstract IsBusy : unit -> bool  
override IsBusy : unit -> bool
```

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Domain.WoodstocksIMSState.IsBusy"]

Implements

[IWoodstocksIMSClient.IsBusy\(\)](#)

Remarks

The default implementation of this method is return true to indicate that the system is busy whilst carrying out an asynchronous operation. The method is implemented to return true, because unless the system is in its idle state if the system is carrying out an asynchronous operation then the system will be busy. The default implementation is intended to minimise the requirement to override this method when by derived state classes that define particular states.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.OnExportCompleted Method

Raises the [ExportCompleted](#) event of the [IWoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void OnExportCompleted(  
    Object sender,  
    AsyncCompletedEventArgs e  
)
```

VB

```
Public Overridable Sub OnExportCompleted (  
    sender As Object,  
    e As AsyncCompletedEventArgs  
)
```

C++

```
public:  
virtual void OnExportCompleted(  
    Object^ sender,  
    AsyncCompletedEventArgs^ e  
)
```

F#

```
abstract OnExportCompleted :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit  
override OnExportCompleted :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The [IWoodstocksIMS](#) that raised the event.

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

The event data for the [ExportCompleted](#) event.

Implements

[IWoodstocksIMS.OnExportCompleted\(Object, AsyncCompletedEventArgs\)](#)

A Sandcastle Documented Class Library

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.OnImportCompleted Method

Raises the [ImportCompleted](#) event of the [IWoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void OnImportCompleted(  
    Object sender,  
    AsyncCompletedEventArgs e  
)
```

VB

```
Public Overridable Sub OnImportCompleted (  
    sender As Object,  
    e As AsyncCompletedEventArgs  
)
```

C++

```
public:  
virtual void OnImportCompleted(  
    Object^ sender,  
    AsyncCompletedEventArgs^ e  
)
```

F#

```
abstract OnImportCompleted :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit  
override OnImportCompleted :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The [IWoodstocksIMS](#) that raised the event.

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

The event data for the [ImportCompleted](#) event.

Implements

[IWoodstocksIMS.OnImportCompleted\(Object, AsyncCompletedEventArgs\)](#)

A Sandcastle Documented Class Library

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstockslMSState.OnProgressChanged Method

Raises the [ProgressChanged](#) event of the [IWoodstockslMS](#) to indicate that progress of an asynchronous operation.

Namespace: [Woodstocks.WoodstockslMS.Domain](#)

Assembly: WoodstockslMSLib (in WoodstockslMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void OnProgressChanged(  
    Object sender,  
    ProgressChangedEventArgs e  
)
```

VB

```
Public Overridable Sub OnProgressChanged (  
    sender As Object,  
    e As ProgressChangedEventArgs  
)
```

C++

```
public:  
virtual void OnProgressChanged(  
    Object^ sender,  
    ProgressChangedEventArgs^ e  
)
```

F#

```
abstract OnProgressChanged :  
    sender : Object *  
    e : ProgressChangedEventArgs -> unit  
override OnProgressChanged :  
    sender : Object *  
    e : ProgressChangedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The [IWoodstockslMS](#) that raised the event.

e

Type: [System.ComponentModel.ProgressChangedEventArgs](#)

The event data for the [ProgressChanged](#) event.

Implements

[IWoodstockslMS.OnProgressChanged\(Object, ProgressChangedEventArgs\)](#)

A Sandcastle Documented Class Library

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.SetModifiedToys Method

Sets the toy data that has been modified and has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void SetModifiedToys(  
    IToys modified  
)
```

VB

```
Public Overridable Sub SetModifiedToys (  
    modified As IToys  
)
```

C++

```
public:  
virtual void SetModifiedToys(  
    IToys^ modified  
)
```

F#

```
abstract SetModifiedToys :  
    modified : IToys -> unit  
override SetModifiedToys :  
    modified : IToys -> unit
```

Parameters

modified

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The modified toy data.

Return Value

Type:

A reference to the modified toy data.

Implements

[IWoodstocksIMS.SetModifiedToys\(IToys\)](#)

Remarks

This method is intended as a system method and therefore does not appear on the client interface.

See Also

[WoodstocksIMSState Class](#)

A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.SetState Method

Set the current state of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SetState(  
    WoodstocksIMSState newState  
)
```

VB

```
Public Sub SetState (  
    newState As WoodstocksIMSState  
)
```

C++

```
public:  
virtual void SetState(  
    WoodstocksIMSState^ newState  
) sealed
```

F#

```
abstract SetState :  
    newState : WoodstocksIMSState -> unit  
override SetState :  
    newState : WoodstocksIMSState -> unit
```

Parameters

newState

Type: [Woodstocks.WoodstocksIMS.Domain.WoodstocksIMSState](#)

Implements

[IWoodstocksIMS.SetState\(WoodstocksIMSState\)](#)

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.SetToyDataSource Method

Sets the source from which the system will, or has, imported toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void SetToyDataSource(  
    string source  
)
```

VB

```
Public Overridable Sub SetToyDataSource (  
    source As String  
)
```

C++

```
public:  
virtual void SetToyDataSource(  
    String^ source  
)
```

F#

```
abstract SetToyDataSource :  
    source : string -> unit  
override SetToyDataSource :  
    source : string -> unit
```

Parameters

source

Type: [System.String](#)

The source of the toy data.

Implements

[IWoodstocksIMS.SetToyDataSource\(String\)](#)

Remarks

The default implementation throws an [InvalidOperationException](#). This is to prevent a state by default from being able to set the toy data source. If a particular state should allow the toy data source to be set then the derived state class should override this method to provide the required functionality.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.SetToyExporter Method

Sets the toy exporter used for exporting data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void SetToyExporter(  
    IWoodstocksToyExporter toyExporter  
)
```

VB

```
Public Overridable Sub SetToyExporter (  
    toyExporter As IWoodstocksToyExporter  
)
```

C++

```
public:  
virtual void SetToyExporter(  
    IWoodstocksToyExporter^ toyExporter  
)
```

F#

```
abstract SetToyExporter :  
    toyExporter : IWoodstocksToyExporter -> unit  
override SetToyExporter :  
    toyExporter : IWoodstocksToyExporter -> unit
```

Parameters

toyExporter

Type: [Woodstocks.WoodstocksIMS.Domain.IWoodstocksToyExporter](#)

The exporter that the system should use to export toy data.

Implements

[IWoodstocksIMS.SetToyExporter\(IWoodstocksToyExporter\)](#)

Remarks

By default this method throws a [InvalidOperationException](#). This is to ensure that by default states cannot set a toy exporter. If a particular state should set the toy exporter then this method should be overridden by the derived state class that defines the state to provide the required functionality.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.SetToyImporter Method

Sets the toy importer of the system.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void SetToyImporter(  
    IWoodstocksToyImporter toyImporter  
)
```

VB

```
Public Overridable Sub SetToyImporter (  
    toyImporter As IWoodstocksToyImporter  
)
```

C++

```
public:  
virtual void SetToyImporter(  
    IWoodstocksToyImporter^ toyImporter  
)
```

F#

```
abstract SetToyImporter :  
    toyImporter : IWoodstocksToyImporter -> unit  
override SetToyImporter :  
    toyImporter : IWoodstocksToyImporter -> unit
```

Parameters

toyImporter

Type: [Woodstocks.WoodstocksIMS.Domain.IWoodstocksToyImporter](#)

The importer that the system should be set to use.

Implements

[IWoodstocksIMS.SetToyImporter\(IWoodstocksToyImporter\)](#)

Remarks

By default this method throws a [InvalidOperationException](#). This is to ensure that by default states cannot set toy importer of the system. If a particular state should set the toy importer then this method should be overridden by the derived state class that defines the state to provide the required functionality.

See Also

[WoodstocksIMSState Class](#)

A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.SetToys Method

Sets the toy data in use by the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void SetToys(  
    IToys toys  
)
```

VB

```
Public Overridable Sub SetToys (  
    toys As IToys  
)
```

C++

```
public:  
virtual void SetToys(  
    IToys^ toys  
)
```

F#

```
abstract SetToys :  
    toys : IToys -> unit  
override SetToys :  
    toys : IToys -> unit
```

Parameters

toys

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The toy data to be used by the system.

Implements

[IWoodstocksIMS.SetToys\(IToys\)](#)

Remarks

The default implementation of this method causes an [InvalidOperationException](#) to be thrown. This is to prevent a state by default from setting the toy data being used by the system. If a particular state should allow the toy data to be set then the derived state class for the state should override this method to provide the required functionality.

See Also

[WoodstocksIMSState Class](#)

A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.UnsavedChanges Method

Gets whether the toy data contains unsaved changes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual bool UnsavedChanges()
```

VB

```
Public Overridable Function UnsavedChanges As Boolean
```

C++

```
public:  
virtual bool UnsavedChanges()
```

F#

```
abstract UnsavedChanges : unit -> bool  
override UnsavedChanges : unit -> bool
```

Return Value

Type: [Boolean](#)

True if the toy data contains unsaved changes, false if it does not.

Implements

[IWoodstocksIMSClient.UnsavedChanges\(\)](#)

Remarks

By default calling this method results in an `InvalidOperationException`. If a state should allow the retrieval of unsaved changes status, then the state should override this method and call the `GetUnsavedChanges` method.

See Also




[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.WoodstocksIMSState Properties

The [WoodstocksIMSState](#) type exposes the following members.

Properties

	Name	Description
	ToyDataSource	Gets and Sets the data source from which toy data should be imported.
	ToyImporter	Gets and Sets the IWoodstocksToyImporter to be used by the system to import toy data.
	Toys	Retrieves IToys which references Wood Stocks toy data after importation.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.ToyDataSource Property

Gets and Sets the data source from which toy data should be imported.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual string ToyDataSource { get; set; }
```

VB

```
Public Overridable Property ToyDataSource As String  
    Get  
    Set
```

C++

```
public:  
virtual property String^ ToyDataSource {  
    String^ get ();  
    void set (String^ value);  
}
```

F#

```
abstract ToyDataSource : string with get, set  
override ToyDataSource : string with get, set
```

Property Value

Type: [String](#)

Implements

[IWoodstocksIMSClient.ToyDataSource](#)

Remarks

The default implementation of this property redirects the get operation to the [GetToyDataSource\(\)](#) method, whilst the set operation is redirected to the [SetToyDataSource\(String\)](#) method

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.ToyImporter Property

Gets and Sets the [IWoodstocksToyImporter](#) to be used by the system to import toy data.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual IWoodstocksToyImporter ToyImporter { get; set; }
```

VB

```
Public Overridable Property ToyImporter As IWoodstocksToyImporter  
    Get  
    Set
```

C++

```
public:  
virtual property IWoodstocksToyImporter^ ToyImporter {  
    IWoodstocksToyImporter^ get ();  
    void set (IWoodstocksToyImporter^ value);  
}
```

F#

```
abstract ToyImporter : IWoodstocksToyImporter with get, set  
override ToyImporter : IWoodstocksToyImporter with get, set
```

Property Value

Type: [IWoodstocksToyImporter](#)

Implements

[IWoodstocksIMS.ToyImporter](#)

Remarks

The default implementation of this property is to redirect the get or set operation to a method that gets or sets the toy importer. By default the methods to which the get or set operation are directed throw an [InvalidOperationException](#). This is to ensure that by default states do not alter the toy importer in use by the system. If a particular state should allow the toy importer to be retrieved or set then the appropriate method should be overridden in the derived state class that defines the state.

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.Toys Property

Retrieves [IToys](#) which references Wood Stocks toy data after importation.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual IToys Toys { get; }
```

VB

```
Public Overridable ReadOnly Property Toys As IToys  
    Get
```

C++

```
public:  
virtual property IToys^ Toys {  
    IToys^ get ();  
}
```

F#

```
abstract Toys : IToys with get  
override Toys : IToys with get
```

Return Value

Type: [IToys](#)

Implements

[IWoodstocksIMSCient.Toys](#)

Remarks

The default implementation provided for the Toys property is to throw an [InvalidOperationException](#). This is to ensure that toy data can only be accessed in states that allow the toy data to be accessed. If a particular state should allow for the toy data of the system to be accessed then this property should be overridden in a derived state class.

See Also




[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstockslMSState.WoodstockslMSState Events

The [WoodstockslMSState](#) type exposes the following members.

Events

	Name	Description
	ExportCompleted	Raised when an asynchronous export completes.
	ImportCompleted	Raised when an asynchronous import completes.
	ProgressChanged	Raised when progress on an asynchronous operation is made.

See Also

[WoodstockslMSState Class](#)

[Woodstocks.WoodstockslMS.Domain Namespace](#)

WoodstocksIMSState.ExportCompleted Event

Raised when an asynchronous export completes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event AsyncCompletedEventHandler ExportCompleted
```

VB

```
Public Event ExportCompleted As AsyncCompletedEventHandler
```

C++

```
public:  
virtual event AsyncCompletedEventHandler^ ExportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>  
override ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

Implements

[IWoodstocksIMSClient.ExportCompleted](#)

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.ImportCompleted Event

Raised when an asynchronous import completes.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event AsyncCompletedEventHandler ImportCompleted
```

VB

```
Public Event ImportCompleted As AsyncCompletedEventHandler
```

C++

```
public:  
virtual event AsyncCompletedEventHandler^ ImportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ImportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>  
override ImportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

Implements

[IWoodstocksIMSClient.ImportCompleted](#)

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksIMSState.ProgressChanged Event

Raised when progress on an asynchronous operation is made.

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event ProgressChangedEventHandler ProgressChanged
```

VB

```
Public Event ProgressChanged As ProgressChangedEventHandler
```

C++

```
public:  
virtual event ProgressChangedEventHandler^ ProgressChanged {  
    void add (ProgressChangedEventHandler^ value);  
    void remove (ProgressChangedEventHandler^ value);  
}
```

F#

```
abstract ProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>  
override ProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>
```

Value

Type: [System.ComponentModel.ProgressChangedEventHandler](#)

Implements

[IWoodstocksIMSClient.ProgressChanged](#)

See Also

[WoodstocksIMSState Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksToyValidator Class

A static class that contains methods for performing validation of Wood Stocks toy data.

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Domain.WoodstocksToyValidator

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public static class WoodstocksToyValidator
```

VB

```
Public NotInheritable Class WoodstocksToyValidator
```

C++






```
public ref class WoodstocksToyValidator abstract sealed
```

F#

```
[<AbstractClassAttribute>]  
[<SealedAttribute>]  
type WoodstocksToyValidator = class end
```

The **WoodstocksToyValidator** type exposes the following members.

Methods

	Name	Description
	IsValidCurrentCount	Performs validation of a string value to be used for the current count of a Toy .
	IsValidDescription	Performs validation of a string value to be used for the item description of a Toy .
	IsValidItemCode	Performs validation of a string value to be used for the item code of a Toy .
	IsValidOnOrder(Int32)	Performs validation of a int value to be used for the OnOrder status of a Toy .
	IsValidOnOrder(String)	Performs validation of a string value to be used for the OnOrder status of a Toy .











See Also

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksToyValidator.WoodstocksToyValidator Methods

The [WoodstocksToyValidator](#) type exposes the following members.

Methods

	Name	Description
 	IsValidCurrentCount	Performs validation of a string value to be used for the current count of a Toy .
 	IsValidDescription	Performs validation of a string value to be used for the item description of a Toy .
 	IsValidItemCode	Performs validation of a string value to be used for the item code of a Toy .
 	IsValidOnOrder(Int32)	Performs validation of a int value to be used for the OnOrder status of a Toy .
 	IsValidOnOrder(String)	Performs validation of a string value to be used for the OnOrder status of a Toy .

See Also

[WoodstocksToyValidator Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksToyValidator.IsValidCurrentCount Method

Performs validation of a string value to be used for the current count of a [Toy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public static bool IsValidCurrentCount(  
    string value  
)
```

VB

```
Public Shared Function IsValidCurrentCount (  
    value As String  
) As Boolean
```

C++

```
public:  
static bool IsValidCurrentCount(  
    String^ value  
)
```

F#

```
static member IsValidCurrentCount :  
    value : string -> bool
```

Parameters

value

Type: [System.String](#)

The value that is to be validated.

Return Value

Type: [Boolean](#)

True if the value is valid.

See Also

[WoodstocksToyValidator Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksToyValidator.IsValidDescription Method

Performs validation of a string value to be used for the item description of a [Toy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public static bool IsValidDescription(  
    string value  
)
```

VB

```
Public Shared Function IsValidDescription (  
    value As String  
) As Boolean
```

C++

```
public:  
static bool IsValidDescription(  
    String^ value  
)
```

F#

```
static member IsValidDescription :  
    value : string -> bool
```

Parameters

value

Type: [System.String](#)

The value that is to be validated.

Return Value

Type: [Boolean](#)

True if the value is valid.

See Also

[WoodstocksToyValidator Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksToyValidator.IsValidItemCode Method

Performs validation of a string value to be used for the item code of a [Toy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public static bool IsValidItemCode(  
    string value  
)
```

VB

```
Public Shared Function IsValidItemCode (  
    value As String  
) As Boolean
```

C++

```
public:  
static bool IsValidItemCode(  
    String^ value  
)
```

F#

```
static member IsValidItemCode :  
    value : string -> bool
```

Parameters

value

Type: [System.String](#)

The value that is to be validated.

Return Value

Type: [Boolean](#)

True if the value is valid.



See Also

[WoodstocksToyValidator Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksToyValidator.IsValidOnOrder Method

Overload List

	Name	Description
	IsValidOnOrder(Int32)	Performs validation of a int value to be used for the OnOrder status of a Toy .
	IsValidOnOrder(String)	Performs validation of a string value to be used for the OnOrder status of a Toy .

See Also

[WoodstocksToyValidator Class](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksToyValidator.IsValidOnOrder Method (Int32)

Performs validation of a int value to be used for the OnOrder status of a [Toy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public static bool IsValidOnOrder(  
    int value  
)
```

VB

```
Public Shared Function IsValidOnOrder (  
    value As Integer  
) As Boolean
```

C++

```
public:  
static bool IsValidOnOrder(  
    int value  
)
```

F#

```
static member IsValidOnOrder :  
    value : int -> bool
```

Parameters

value

Type: [System.Int32](#)

The value that is to be validated.

Return Value

Type: [Boolean](#)

True if the value is valid.

See Also

[WoodstocksToyValidator Class](#)

[IsValidOnOrder Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)

WoodstocksToyValidator.IsValidOnOrder Method (String)

Performs validation of a string value to be used for the OnOrder status of a [Toy](#).

Namespace: [Woodstocks.WoodstocksIMS.Domain](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public static bool IsValidOnOrder(  
    string value  
)
```

VB

```
Public Shared Function IsValidOnOrder (  
    value As String  
) As Boolean
```

C++

```
public:  
static bool IsValidOnOrder(  
    String^ value  
)
```

F#

```
static member IsValidOnOrder :  
    value : string -> bool
```

Parameters

value

Type: [System.String](#)

The value that is to be validated.

Return Value

Type: [Boolean](#)

True if the value is valid.

See Also

[WoodstocksToyValidator Class](#)




[IsValidOnOrder Overload](#)

[Woodstocks.WoodstocksIMS.Domain Namespace](#)



Woodstocks.WoodstocksIMS.Presentation Namespace

The Woodstocks.WoodstocksIMS.Presentation namespace contains the presentation layer components for the WoodstocksIMS.

Classes

	Class	Description
	ProgressDialog	A progress dialog box.
	WoodstocksIMSController	A controller for the WoodstocksIMS .
	WoodstocksIMSForm	A view for the WoodstocksIMS .

Interfaces

	Interface	Description
	IWoodstocksIMSController	Defines the interface of a controller for a WoodstocksIMS .
	IWoodstocksIMSView	An interface for a View within the Wood Stocks Inventory Management System.

IWoodstocksIMSController Interface

Defines the interface of a controller for a [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public interface IWoodstocksIMSController
```

VB

```
Public Interface IWoodstocksIMSController
```

C++












```
public interface class IWoodstocksIMSController
```

F#




```
type IWoodstocksIMSController = interface end
```

The **IWoodstocksIMSController** type exposes the following members.

Methods

	Name	Description
	CancelAsync	Informs the WoodstocksIMS to cancel an asynchronous operation.
	DiscardImportedToyData	Causes the WoodstocksIMS to discard toy data that has been imported into the system.
	ExportToysAsync	Requests the WoodstocksIMS to export Toy data asynchronously into the system.
	GetCancellationStatus	Gets the cancellation status of an operation.
	GetErrorStatus	Gets the error reported by the WoodstocksIMS whne an error occurs.
	GetToys	Gets the toy data from the WoodstocksIMS.
	ImportToysAsync	Requests the WoodstocksIMS to import Toy data asynchronously into the system.
	ResetCancellationStatus	Resets the cancellation status of the cotnroller.
	ResetErrorStatus	Resets the error status reported by the controller
	SetView	Set the controller's view.
	UnsavedChanges	Gets whether the IWoodstocksIMS has imported data that has not been saved.

Events

	Name	Description
	ExportCompleted	Raised by the controller when an asynchronous exportation has completed.
	ImportCompleted	Raised by the controller when an asynchronous importation has completed.
	ProgressChanged	Raised by the controller when progress on an asynchronous operation.












See Also

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.IWoodstocksIMSController Methods

The [IWoodstocksIMSController](#) type exposes the following members.

Methods

	Name	Description
	CancelAsync	Informs the WoodstocksIMS to cancel an asynchronous operation.
	DiscardImportedToyData	Causes the WoodstocksIMS to discard toy data that has been imported into the system.
	ExportToysAsync	Requests the WoodstocksIMS to export Toy data asynchronously into the system.
	GetCancellationStatus	Gets the cancellation status of an operation.
	GetErrorStatus	Gets the error reported by the WoodstocksIMS whne an error occurs.
	GetToys	Gets the toy data from the WoodstocksIMS.
	ImportToysAsync	Requests the WoodstocksIMS to import Toy data asynchronously into the system.
	ResetCancellationStatus	Resets the cancellation status of the cotnroller.
	ResetErrorStatus	Resets the error status reported by the controller
	SetView	Set the controller's view.
	UnsavedChanges	Gets whether the IWoodstocksIMS has imported data that has not been saved.

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.CancelAsync Method

Informs the [WoodstocksIMS](#) to cancel an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void CancelAsync()
```

VB

```
Sub CancelAsync
```

C++

```
void CancelAsync()
```

F#

```
abstract CancelAsync : unit -> unit
```

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.DiscardImportedToyData Method

Causes the [WoodstocksIMS](#) to discard toy data that has been imported into the system.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void DiscardImportedToyData()
```

VB

```
Sub DiscardImportedToyData
```

C++

```
void DiscardImportedToyData()
```

F#

```
abstract DiscardImportedToyData : unit -> unit
```

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.ExportToysAsync Method

Requests the [WoodstocksIMS](#) to export [Toy](#) data asynchronously into the system.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void ExportToysAsync ( )
```

VB

```
Sub ExportToysAsync
```

C++

```
void ExportToysAsync ( )
```

F#

```
abstract ExportToysAsync : unit -> unit
```

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.GetCancellationStatus Method

Gets the cancellation status of an operation.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
bool GetCancellationStatus()
```

VB

```
Function GetCancellationStatus As Boolean
```

C++

```
bool GetCancellationStatus()
```

F#

```
abstract GetCancellationStatus : unit -> bool
```

Return Value

Type: [Boolean](#)

True if an operation has been cancelled, otherwise false.

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.GetErrorStatus Method

Gets the error reported by the WoodstocksIMS whne an error occurs.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
Exception GetErrorStatus()
```

VB

```
Function GetErrorStatus As Exception
```

C++

```
Exception^ GetErrorStatus()
```

F#

```
abstract GetErrorStatus : unit -> Exception
```

Return Value

Type: [Exception](#)

The [Exception](#) that reports the error.

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.GetToys Method

Gets the toy data from the WoodstocksIMS.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
IToys GetToys()
```

VB

```
Function GetToys As IToys
```

C++

```
IToys^ GetToys()
```

F#

```
abstract GetToys : unit -> IToys
```

Return Value

Type: [IToys](#)

The imported toy data.

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.ImportToysAsync Method

Requests the [WoodstocksIMS](#) to import [Toy](#) data asynchronously into the system.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void ImportToysAsync(  
    string source  
)
```

VB

```
Sub ImportToysAsync (  
    source As String  
)
```

C++

```
void ImportToysAsync(  
    String^ source  
)
```

F#

```
abstract ImportToysAsync :  
    source : string -> unit
```

Parameters

source

Type: [System.String](#)

The source from which [Toy](#) data should be imported.

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.ResetCancellationStatus Method

Resets the cancellation status of the cotnroller.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void ResetCancellationStatus()
```

VB

```
Sub ResetCancellationStatus
```

C++

```
void ResetCancellationStatus()
```

F#

```
abstract ResetCancellationStatus : unit -> unit
```

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.ResetErrorStatus Method

Resets the error status reported by the controller

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void ResetErrorStatus()
```

VB

```
Sub ResetErrorStatus
```

C++

```
void ResetErrorStatus()
```

F#

```
abstract ResetErrorStatus : unit -> unit
```

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.SetView Method

Set the controller's view.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void SetView(  
    IWoodstocksIMSView view  
)
```

VB

```
Sub SetView (  
    view As IWoodstocksIMSView  
)
```

C++

```
void SetView(  
    IWoodstocksIMSView^ view  
)
```

F#

```
abstract SetView :  
    view : IWoodstocksIMSView -> unit
```

Parameters

view

Type: [Woodstocks.WoodstocksIMS.Presentation.IWoodstocksIMSView](#)

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.UnsavedChanges Method

Gets whether the [IWoodstocksIMS](#) has imported data that has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
bool UnsavedChanges( )
```

VB

```
Function UnsavedChanges As Boolean
```

C++

```
bool UnsavedChanges( )
```

F#

```
abstract UnsavedChanges : unit -> bool
```

Return Value

Type: [Boolean](#)

True if the system has modified data that has not been saved, otherwise false.

See Also




[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.IWoodstocksIMSController Events

The [IWoodstocksIMSController](#) type exposes the following members.

Events

	Name	Description
	ExportCompleted	Raised by the controller when an asynchronous exportation has completed.
	ImportCompleted	Raised by the controller when an asynchronous importation has completed.
	ProgressChanged	Raised by the controller when progress on an asynchronous operation.

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.ExportCompleted Event

Raised by the controller when an asynchronous exportation has completed.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event AsyncCompletedEventHandler ExportCompleted
```

VB

```
Event ExportCompleted As AsyncCompletedEventHandler
```

C++

```
event AsyncCompletedEventHandler^ ExportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.ImportCompleted Event

Raised by the controller when an asynchronous importation has completed.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event AsyncCompletedEventHandler ImportCompleted
```

VB

```
Event ImportCompleted As AsyncCompletedEventHandler
```

C++

```
event AsyncCompletedEventHandler^ ImportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ImportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSController.ProgressChanged Event

Raised by the controller when progress on an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
event ProgressChangedEventHandler ProgressChanged
```

VB

```
Event ProgressChanged As ProgressChangedEventHandler
```

C++

```
event ProgressChangedEventHandler^ ProgressChanged {  
    void add (ProgressChangedEventHandler^ value);  
    void remove (ProgressChangedEventHandler^ value);  
}
```

F#

```
abstract ProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>
```

Value

Type: [System.ComponentModel.ProgressChangedEventHandler](#)

See Also

[IWoodstocksIMSController Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSView Interface

An interface for a View within the Wood Stocks Inventory Management System.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public interface IWoodstocksIMSView
```

VB

```
Public Interface IWoodstocksIMSView
```

C++






```
public interface class IWoodstocksIMSView
```

F#

```
type IWoodstocksIMSView = interface end
```

The **IWoodstocksIMSView** type exposes the following members.

Methods

	Name	Description
	DisableImport	Disables import option of the View.
	EnableImport	Enables import option of the View.
	NotifyExportCompletion	Notifies the user that exportation has completed.
	NotifyImportCompletion	Notifies the user that importation has completed.
	Reset	Resets the view.






See Also

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSVIEW.IWoodstocksIMSVIEW Methods

The [IWoodstocksIMSVIEW](#) type exposes the following members.

Methods

	Name	Description
	DisableImport	Disables import option of the View.
	EnableImport	Enables import option of the View.
	NotifyExportCompletion	Notifies the user that exportation has completed.
	NotifyImportCompletion	Notifies the user that importation has completed.
	Reset	Resets the view.

See Also

[IWoodstocksIMSVIEW Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSView.DisableImport Method

Disables import option of the View.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void DisableImport()
```

VB

```
Sub DisableImport
```

C++

```
void DisableImport()
```

F#

```
abstract DisableImport : unit -> unit
```

See Also

[IWoodstocksIMSView Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSView.EnableImport Method

Enables import option of the View.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void EnableImport()
```

VB

```
Sub EnableImport
```

C++

```
void EnableImport()
```

F#

```
abstract EnableImport : unit -> unit
```

See Also

[IWoodstocksIMSView Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSView.NotifyExportCompletion Method

Notifies the user that exportation has completed.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void NotifyExportCompletion(  
    Exception ex,  
    bool cancelled  
)
```

VB

```
Sub NotifyExportCompletion (  
    ex As Exception,  
    cancelled As Boolean  
)
```

C++

```
void NotifyExportCompletion(  
    Exception^ ex,  
    bool cancelled  
)
```

F#

```
abstract NotifyExportCompletion :  
    ex : Exception *  
    cancelled : bool -> unit
```

Parameters

ex

Type: [System.Exception](#)

Any exception that occurred during exportation.

cancelled

Type: [System.Boolean](#)

Whether the exportation has been cancelled.

See Also

[IWoodstocksIMSView Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSView.NotifyImportCompletion Method

Notifies the user that importation has completed.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void NotifyImportCompletion(  
    Exception ex,  
    bool cancelled,  
    IToys result  
)
```

VB

```
Sub NotifyImportCompletion (  
    ex As Exception,  
    cancelled As Boolean,  
    result As IToys  
)
```

C++

```
void NotifyImportCompletion(  
    Exception^ ex,  
    bool cancelled,  
    IToys^ result  
)
```

F#

```
abstract NotifyImportCompletion :  
    ex : Exception *  
    cancelled : bool *  
    result : IToys -> unit
```

Parameters

ex

Type: [System.Exception](#)

Exception that occurred during the import operation.

cancelled

Type: [System.Boolean](#)

Indicates whether the operation was cancelled by the user.

result

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The result of the import operation.

Remarks

If the operation was cancelled or an exception occurred whilst completing the operation then null should be passed as the argument of the result parameter.

If an exception occurs during the operation then the Exception should be passed as the argument for the ex parameter, otherwise null should be passed for ex.

If the operation was cancelled then true should be passed as the argument for the cancelled parameter, otherwise false should be passed for cancelled.

See Also

[IWoodstockIMSView Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

IWoodstocksIMSVIEW.Reset Method

Resets the view.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
void Reset ( )
```

VB

```
Sub Reset
```

C++

```
void Reset ( )
```

F#

```
abstract Reset : unit -> unit
```

See Also

[IWoodstocksIMSVIEW Interface](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

ProgressDialog Class

A progress dialog box.

Inheritance Hierarchy

[System.Object](#)

[System.MarshalByRefObject](#)

[System.ComponentModel.Component](#)

[System.Windows.Forms.Control](#)

[System.Windows.Forms.ScrollableControl](#)

[System.Windows.Forms.ContainerControl](#)

[System.Windows.Forms.Form](#)

Woodstocks.WoodstocksIMS.Presentation.ProgressDialog

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class ProgressDialog : Form
```

VB

```
Public Class ProgressDialog  
    Inherits Form
```

C++


```
public ref class ProgressDialog : public Form
```

F#

```
type ProgressDialog =  
    class  
        inherit Form  
    end
```

















The **ProgressDialog** type exposes the following members.



















Constructors


















	Name	Description
	ProgressDialog	Initialises the ProgressDialog .













Methods









	Name	Description
	AccessibilityNotifyClients(AccessibleEvents,	Notifies the accessibility client applications of the


























	Int32)	specified AccessibleEvents for the specified child control. (Inherited from Control .)
	AccessibilityNotifyClients(AccessibleEvents, Int32, Int32)	Notifies the accessibility client applications of the specified AccessibleEvents for the specified child control . (Inherited from Control .)
	Activate	Activates the form and gives it focus. (Inherited from Form .)
	ActivateMdiChild	Activates the MDI child of a form. (Inherited from Form .)
	AddOwnedForm	Adds an owned form to this form. (Inherited from Form .)
	AdjustFormScrollbars	Adjusts the scroll bars on the container based on the current control positions and the control currently selected. (Inherited from Form .)
	ApplyAutoScaling	Obsolete. Resizes the form according to the current value of the AutoScaleBaseSize property and the size of the current font. (Inherited from Form .)
	BeginInvoke(Delegate)	Executes the specified delegate asynchronously on the thread that the control's underlying handle was created on. (Inherited from Control .)
	BeginInvoke(Delegate, Object[])	Executes the specified delegate asynchronously with the specified arguments, on the thread that the control's underlying handle was created on. (Inherited from Control .)
	BringToFront	Brings the control to the front of the z-order. (Inherited from Control .)
	CenterToParent	Centers the position of the form within the bounds of the parent form. (Inherited from Form .)
	CenterToScreen	Centers the form on the current screen. (Inherited from Form .)
	Close	Closes the form. (Inherited from Form .)
	Completed	Event Listener that can be used to handle task completion events.
	Contains	Retrieves a value indicating whether the specified control is a child of the control. (Inherited from Control .)
	CreateAccessibilityInstance	Creates a new accessibility object for the control. (Inherited from Control .)
	CreateControl	Forces the creation of the visible control, including the creation of the handle and any visible child controls.

























		(Inherited from Control .)
	CreateControlsInstance	(Inherited from Form .)
	CreateGraphics	Creates the Graphics for the control. (Inherited from Control .)
	CreateHandle	Creates the handle for the form. If a derived class overrides this function, it must call the base implementation. (Inherited from Form .)
	CreateObjRef	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object. (Inherited from MarshalByRefObject .)
	DefWndProc	(Inherited from Form .)
	DestroyHandle	Destroys the handle associated with the control. (Inherited from Control .)
	Dispose()	Releases all resources used by the Component . (Inherited from Component .)
	Dispose(Boolean)	Clean up any resources being used. (Overrides Form.Dispose(Boolean) .)
	DoDragDrop	Begins a drag-and-drop operation. (Inherited from Control .)
	DrawToBitmap	Supports rendering to the specified bitmap. (Inherited from Control .)
	EndInvoke	Retrieves the return value of the asynchronous operation represented by the IAsyncResult passed. (Inherited from Control .)
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Releases unmanaged resources and performs other cleanup operations before the Component is reclaimed by garbage collection. (Inherited from Component .)
	FindForm	Retrieves the form that the control is on. (Inherited from Control .)
	Focus	Sets input focus to the control. (Inherited from Control .)
	GetAccessibilityObjectById	Retrieves the specified AccessibleObject . (Inherited from Control .)
	GetAutoSizeMode	Retrieves a value indicating how a control will behave when its AutoSize property is enabled. (Inherited from Control .)
	GetChildAtPoint(Point)	Retrieves the child control that is located at the



















		specified coordinates. (Inherited from Control .)
	GetChildAtPoint(Point, GetChildAtPointSkip)	Retrieves the child control that is located at the specified coordinates, specifying whether to ignore child controls of a certain type. (Inherited from Control .)
	GetContainerControl	Returns the next ContainerControl up the control's chain of parent controls. (Inherited from Control .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetLifetimeService	Retrieves the current lifetime service object that controls the lifetime policy for this instance. (Inherited from MarshalByRefObject .)
	GetNextControl	Retrieves the next control forward or back in the tab order of child controls. (Inherited from Control .)
	GetPreferredSize	Retrieves the size of a rectangular area into which a control can be fitted. (Inherited from Control .)
	GetScaledBounds	(Inherited from Form .)
	GetScrollState	Determines whether the specified flag has been set. (Inherited from ScrollableControl .)
	GetService	Returns an object that represents a service provided by the Component or by its Container . (Inherited from Component .)
	GetStyle	Retrieves the value of the specified control style bit for the control. (Inherited from Control .)
	GetTopLevel	Determines if the control is a top-level control. (Inherited from Control .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	Hide	Conceals the control from the user. (Inherited from Control .)
	InitializeLifetimeService	Obtains a lifetime service object to control the lifetime policy for this instance. (Inherited from MarshalByRefObject .)
	InitLayout	Called after the control has been added to another container. (Inherited from Control .)
	Invalidate()	Invalidates the entire surface of the control and causes the control to be redrawn. (Inherited from Control .)
	Invalidate(Region)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and













		causes a paint message to be sent to the control. (Inherited from Control.)
	Invalidate(Boolean)	Invalidates a specific region of the control and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control.)
	Invalidate(Rectangle)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. (Inherited from Control.)
	Invalidate(Region, Boolean)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control.)
	Invalidate(Rectangle, Boolean)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control.)
	Invoke(Delegate)	Executes the specified delegate on the thread that owns the control's underlying window handle. (Inherited from Control.)
	Invoke(Delegate, Object[])	Executes the specified delegate, on the thread that owns the control's underlying window handle, with the specified list of arguments. (Inherited from Control.)
	InvokeGotFocus	Raises the GotFocus event for the specified control. (Inherited from Control.)
	InvokeLostFocus	Raises the LostFocus event for the specified control. (Inherited from Control.)
	InvokeOnClick	Raises the Click event for the specified control. (Inherited from Control.)
	InvokePaint	Raises the Paint event for the specified control. (Inherited from Control.)
	InvokePaintBackground	Raises the PaintBackground event for the specified control. (Inherited from Control.)
	IsInputChar	Determines if a character is an input character that the control recognizes. (Inherited from Control.)





















	IsInputKey	Determines whether the specified key is a regular input key or a special key that requires preprocessing. (Inherited from Control .)
	LayoutMdi	Arranges the multiple-document interface (MDI) child forms within the MDI parent form. (Inherited from Form .)
	MemberwiseClone()	Creates a shallow copy of the current Object . (Inherited from Object .)
	MemberwiseClone(Boolean)	Creates a shallow copy of the current MarshalByRefObject object. (Inherited from MarshalByRefObject .)
	NotifyInvalidade	Raises the Invalidated event with a specified region of the control to invalidate. (Inherited from Control .)
	OnActivated	Raises the Activated event. (Inherited from Form .)
	OnAutoSizeChanged	Raises the AutoSizeChanged event. (Inherited from Control .)
	OnAutoValidateChanged	Raises the AutoValidateChanged event. (Inherited from ContainerControl .)
	OnBackColorChanged	Raises the BackColorChanged event. (Inherited from Control .)
	OnBackgroundImageChanged	Raises the BackgroundImageChanged event. (Inherited from Form .)
	OnBackgroundImageLayoutChanged	Raises the BackgroundImageLayoutChanged event. (Inherited from Form .)
	OnBindingContextChanged	Raises the BindingContextChanged event. (Inherited from Control .)
	OnCausesValidationChanged	Raises the CausesValidationChanged event. (Inherited from Control .)
	OnChangeUICues	Raises the ChangeUICues event. (Inherited from Control .)
	OnClick	Raises the Click event. (Inherited from Control .)
	OnClientSizeChanged	Raises the ClientSizeChanged event. (Inherited from Control .)
	OnClosed	Raises the Closed event. (Inherited from Form .)
	OnClosing	Raises the Closing event. (Inherited from Form .)
	OnContextMenuChanged	Raises the ContextMenuChanged event. (Inherited from Control .)
	OnContextMenuStripChanged	Raises the ContextMenuStripChanged event. (Inherited from Control .)
	OnControlAdded	Raises the ControlAdded event. (Inherited from















		Control.)
	OnControlRemoved	Raises the ControlRemoved event. (Inherited from Control.)
	OnCreateControl	Raises the CreateControl event. (Inherited from Form.)
	OnCursorChanged	Raises the CursorChanged event. (Inherited from Control.)
	OnDeactivate	Raises the Deactivate event. (Inherited from Form.)
	OnDockChanged	Raises the DockChanged event. (Inherited from Control.)
	OnDoubleClick	Raises the DoubleClick event. (Inherited from Control.)
	OnDragDrop	Raises the DragDrop event. (Inherited from Control.)
	OnDragEnter	Raises the DragEnter event. (Inherited from Control.)
	OnDragLeave	Raises the DragLeave event. (Inherited from Control.)
	OnDragOver	Raises the DragOver event. (Inherited from Control.)
	OnEnabledChanged	(Inherited from Form.)
	OnEnter	Raises the Enter event. (Inherited from Form.)
	OnFontChanged	(Inherited from Form.)
	OnForeColorChanged	Raises the ForeColorChanged event. (Inherited from Control.)
	OnFormClosed	Raises the FormClosed event. (Inherited from Form.)
	OnFormClosing	Raises the FormClosing event. (Inherited from Form.)
	OnGiveFeedback	Raises the GiveFeedback event. (Inherited from Control.)
	OnGotFocus	Raises the GotFocus event. (Inherited from Control.)
	OnHandleCreated	(Inherited from Form.)
	OnHandleDestroyed	(Inherited from Form.)
	OnHelpButtonClicked	Raises the HelpButtonClicked event. (Inherited from Form.)
	OnHelpRequested	Raises the HelpRequested event. (Inherited from Control.)
	OnImeModeChanged	Raises the ImeModeChanged event. (Inherited from Control.)
	OnInputLanguageChanged	Raises the InputLanguageChanged event. (Inherited from Form.)
	OnInputLanguageChanging	Raises the InputLanguageChanging event. (Inherited from Form.)





















	OnInvalidated	Raises the Invalidated event. (Inherited from Control .)
	OnKeyDown	Raises the KeyDown event. (Inherited from Control .)
	OnKeyPress	Raises the KeyPress event. (Inherited from Control .)
	OnKeyUp	Raises the KeyUp event. (Inherited from Control .)
	OnLayout	Raises the Layout event. (Inherited from Form .)
	OnLeave	Raises the Leave event. (Inherited from Control .)
	OnLoad	Raises the Load event. (Inherited from Form .)
	OnLocationChanged	Raises the LocationChanged event. (Inherited from Control .)
	OnLostFocus	Raises the LostFocus event. (Inherited from Control .)
	OnMarginChanged	Raises the MarginChanged event. (Inherited from Control .)
	OnMaximizedBoundsChanged	Raises the MaximizedBoundsChanged event. (Inherited from Form .)
	OnMaximumSizeChanged	Raises the MaximumSizeChanged event. (Inherited from Form .)
	OnMdiChildActivate	Raises the MdiChildActivate event. (Inherited from Form .)
	OnMenuComplete	Raises the MenuComplete event. (Inherited from Form .)
	OnMenuStart	Raises the MenuStart event. (Inherited from Form .)
	OnMinimumSizeChanged	Raises the MinimumSizeChanged event. (Inherited from Form .)
	OnMouseCaptureChanged	Raises the MouseCaptureChanged event. (Inherited from Control .)
	OnMouseClicked	Raises the MouseClicked event. (Inherited from Control .)
	OnMouseDoubleClick	Raises the MouseDoubleClick event. (Inherited from Control .)
	OnMouseDown	Raises the MouseDown event. (Inherited from Control .)
	OnMouseEnter	Raises the MouseEnter event. (Inherited from Control .)
	OnMouseHover	Raises the MouseHover event. (Inherited from Control .)
	OnMouseLeave	Raises the MouseLeave event. (Inherited from Control .)
	OnMouseMove	Raises the MouseMove event. (Inherited from Control .)





















	OnMouseUp	Raises the MouseUp event. (Inherited from Control .)
	OnMouseWheel	Raises the MouseWheel event. (Inherited from ScrollableControl .)
	OnMove	Raises the Move event. (Inherited from Control .)
	OnNotifyMessage	Notifies the control of Windows messages. (Inherited from Control .)
	OnPaddingChanged	Raises the PaddingChanged event. (Inherited from ScrollableControl .)
	OnPaint	(Inherited from Form .)
	OnPaintBackground	Paints the background of the control. (Inherited from ScrollableControl .)
	OnParentBackColorChanged	Raises the BackColorChanged event when the BackColor property value of the control's container changes. (Inherited from Control .)
	OnParentBackgroundImageChanged	Raises the BackgroundImageChanged event when the BackgroundImage property value of the control's container changes. (Inherited from Control .)
	OnParentBindingContextChanged	Raises the BindingContextChanged event when the BindingContext property value of the control's container changes. (Inherited from Control .)
	OnParentChanged	(Inherited from ContainerControl .)
	OnParentCursorChanged	Raises the CursorChanged event. (Inherited from Control .)
	OnParentEnabledChanged	Raises the EnabledChanged event when the Enabled property value of the control's container changes. (Inherited from Control .)
	OnParentFontChanged	Raises the FontChanged event when the Font property value of the control's container changes. (Inherited from Control .)
	OnParentForeColorChanged	Raises the ForeColorChanged event when the ForeColor property value of the control's container changes. (Inherited from Control .)
	OnParentRightToLeftChanged	Raises the RightToLeftChanged event when the RightToLeft property value of the control's container changes. (Inherited from Control .)
	OnParentVisibleChanged	Raises the VisibleChanged event when the Visible property value of the control's container changes. (Inherited from Control .)
	OnPreviewKeyDown	Raises the PreviewKeyDown event. (Inherited from Control .)
	OnPrint	Raises the Paint event. (Inherited from Control .)




	OnQueryContinueDrag	Raises the QueryContinueDrag event. (Inherited from Control .)
	OnRegionChanged	Raises the RegionChanged event. (Inherited from Control .)
	OnResize	(Inherited from Form .)
	OnResizeBegin	Raises the ResizeBegin event. (Inherited from Form .)
	OnResizeEnd	Raises the ResizeEnd event. (Inherited from Form .)
	OnRightToLeftChanged	(Inherited from ScrollableControl .)
	OnRightToLeftLayoutChanged	Raises the RightToLeftLayoutChanged event. (Inherited from Form .)
	OnScroll	Raises the Scroll event. (Inherited from ScrollableControl .)
	OnShown	Raises the Shown event. (Inherited from Form .)
	OnSizeChanged	Raises the SizeChanged event. (Inherited from Control .)
	OnStyleChanged	(Inherited from Form .)
	OnSystemColorsChanged	Raises the SystemColorsChanged event. (Inherited from Control .)
	OnTabIndexChanged	Raises the TabIndexChanged event. (Inherited from Control .)
	OnTabStopChanged	Raises the TabStopChanged event. (Inherited from Control .)
	OnTextChanged	(Inherited from Form .)
	OnValidated	Raises the Validated event. (Inherited from Control .)
	OnValidating	Raises the Validating event. (Inherited from Control .)
	OnVisibleChanged	Raises the VisibleChanged event. (Inherited from Form .)
	PerformAutoScale	Performs scaling of the container control and its children. (Inherited from ContainerControl .)
	PerformLayout()	Forces the control to apply layout logic to all its child controls. (Inherited from Control .)
	PerformLayout(Control, String)	Forces the control to apply layout logic to all its child controls. (Inherited from Control .)
	PointToClient	Computes the location of the specified screen point into client coordinates. (Inherited from Control .)
	PointToScreen	Computes the location of the specified client point into screen coordinates. (Inherited from Control .)
	PreProcessControlMessage	Preprocesses keyboard or input messages within the message loop before they are dispatched. (Inherited

		from Control .)
	PreProcessMessage	Preprocesses keyboard or input messages within the message loop before they are dispatched. (Inherited from Control .)
	ProcessCmdKey	Processes a command key. (Inherited from Form .)
	ProcessDialogChar	Processes a dialog character. (Inherited from Form .)
	ProcessDialogKey	Processes a dialog box key. (Inherited from Form .)
	ProcessKeyEventArgs	Processes a key message and generates the appropriate control events. (Inherited from Control .)
	ProcessKeyMessage	Processes a keyboard message. (Inherited from Control .)
	ProcessKeyPreview	(Inherited from Form .)
	ProcessMnemonic	Processes a mnemonic character. (Inherited from Form .)
	ProcessTabKey	(Inherited from Form .)
	ProgressChanged	Event Listener that can be used to handle progress changes.
	RaiseDragEvent	Raises the appropriate drag event. (Inherited from Control .)
	RaiseKeyEvent	Raises the appropriate key event. (Inherited from Control .)
	RaiseMouseEvent	Raises the appropriate mouse event. (Inherited from Control .)
	RaisePaintEvent	Raises the appropriate paint event. (Inherited from Control .)
	RecreateHandle	Forces the re-creation of the handle for the control. (Inherited from Control .)
	RectangleToClient	Computes the size and location of the specified screen rectangle in client coordinates. (Inherited from Control .)
	RectangleToScreen	Computes the size and location of the specified client rectangle in screen coordinates. (Inherited from Control .)
	Refresh	Forces the control to invalidate its client area and immediately redraw itself and any child controls. (Inherited from Control .)
	RemoveOwnedForm	Removes an owned form from this form. (Inherited from Form .)
	ResetBackColor	Resets the BackColor property to its default value. (Inherited from Control .)















	ResetBindings	Causes a control bound to the BindingSource to reread all the items in the list and refresh their displayed values. (Inherited from Control .)
	ResetCursor	Resets the Cursor property to its default value. (Inherited from Control .)
	ResetFont	Resets the Font property to its default value. (Inherited from Control .)
	ResetForeColor	Resets the ForeColor property to its default value. (Inherited from Control .)
	ResetImeMode	Resets the ImeMode property to its default value. (Inherited from Control .)
	ResetMouseEventArgs	Resets the control to handle the MouseLeave event. (Inherited from Control .)
	ResetRightToLeft	Resets the RightToLeft property to its default value. (Inherited from Control .)
	ResetText	Resets the Text property to its default value. (Inherited from Control .)
	ResumeLayout()	Resumes usual layout logic. (Inherited from Control .)
	ResumeLayout(Boolean)	Resumes usual layout logic, optionally forcing an immediate layout of pending layout requests. (Inherited from Control .)
	RtlTranslateAlignment(HorizontalAlignment)	Converts the specified HorizontalAlignment to the appropriate HorizontalAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateAlignment(LeftRightAlignment)	Converts the specified LeftRightAlignment to the appropriate LeftRightAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateAlignment(ContentAlignment)	Converts the specified ContentAlignment to the appropriate ContentAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateContent	Converts the specified ContentAlignment to the appropriate ContentAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateHorizontal	Converts the specified HorizontalAlignment to the appropriate HorizontalAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateLeftRight	Converts the specified LeftRightAlignment to the appropriate LeftRightAlignment to support right-to-left text. (Inherited from Control .)
	Scale(Single)	Obsolete. Scales the control and any child controls. (Inherited





















		from Control .)
	Scale(SizeF)	Scales the control and all child controls by the specified scaling factor. (Inherited from Control .)
	Scale(Single, Single)	Obsolete. Scales the entire control and any child controls. (Inherited from Control .)
	ScaleControl	Scales the location, size, padding, and margin of a control. (Inherited from Form .)
	ScaleCore	Performs scaling of the form. (Inherited from Form .)
	ScrollControlIntoView	Scrolls the specified child control into view on an auto-scroll enabled control. (Inherited from ScrollableControl .)
	ScrollToControl	Calculates the scroll offset to the specified child control. (Inherited from ScrollableControl .)
	Select()	Activates the control. (Inherited from Control .)
	Select(Boolean, Boolean)	Selects this form, and optionally selects the next or previous control. (Inherited from Form .)
	SelectNextControl	Activates the next control. (Inherited from Control .)
	SendToBack	Sends the control to the back of the z-order. (Inherited from Control .)
	SetAutoScrollMargin	Sets the size of the auto-scroll margins. (Inherited from ScrollableControl .)
	SetAutoSizeMode	Sets a value indicating how a control will behave when its AutoSize property is enabled. (Inherited from Control .)
	SetBounds(Int32, Int32, Int32, Int32)	Sets the bounds of the control to the specified location and size. (Inherited from Control .)
	SetBounds(Int32, Int32, Int32, Int32, BoundsSpecified)	Sets the specified bounds of the control to the specified location and size. (Inherited from Control .)
	SetBoundsCore	(Inherited from Form .)
	SetClientSizeCore	Sets the client size of the form. This will adjust the bounds of the form to make the client size the requested size. (Inherited from Form .)
	SetDesktopBounds	Sets the bounds of the form in desktop coordinates. (Inherited from Form .)
	SetDesktopLocation	Sets the location of the form in desktop coordinates. (Inherited from Form .)
	SetDisplayRectLocation	Positions the display window to the specified value. (Inherited from ScrollableControl .)
	SetScrollState	Sets the specified scroll state flag. (Inherited from





















		ScrollableControl.)
	SetStyle	Sets a specified ControlStyles flag to either true or false. (Inherited from Control.)
	SetTopLevel	Sets the control as the top-level control. (Inherited from Control.)
	SetVisibleCore	(Inherited from Form.)
	Show()	Displays the control to the user. (Inherited from Control.)
	Show(IWin32Window)	Shows the form with the specified owner to the user. (Inherited from Form.)
	ShowDialog()	Shows the form as a modal dialog box. (Inherited from Form.)
	ShowDialog(IWin32Window)	Shows the form as a modal dialog box with the specified owner. (Inherited from Form.)
	SizeFromClientSize	Determines the size of the entire control from the height and width of its client area. (Inherited from Control.)
	SuspendLayout	Temporarily suspends the layout logic for the control. (Inherited from Control.)
	ToString	Gets a string representing the current instance of the form. (Inherited from Form.)
	Update	Causes the control to redraw the invalidated regions within its client area. (Inherited from Control.)
	UpdateBounds()	Updates the bounds of the control with the current size and location. (Inherited from Control.)
	UpdateBounds(Int32, Int32, Int32, Int32)	Updates the bounds of the control with the specified size and location. (Inherited from Control.)
	UpdateBounds(Int32, Int32, Int32, Int32, Int32, Int32)	Updates the bounds of the control with the specified size, location, and client size. (Inherited from Control.)
	UpdateDefaultButton	Updates which button is the default button. (Inherited from Form.)
	UpdateProgress	Updates the progress displayed by the dialog window.
	UpdateStyles	Forces the assigned styles to be reapplied to the control. (Inherited from Control.)
	UpdateZOrder	Updates the control in its parent's z-order. (Inherited from Control.)
	Validate()	Verifies the value of the control losing focus by causing the Validating and Validated events to occur, in that order. (Inherited from ContainerControl.)
	Validate(Boolean)	Verifies the value of the control that is losing focus;





















		conditionally dependent on whether automatic validation is turned on. (Inherited from ContainerControl.)
	ValidateChildren()	(Inherited from Form.)
	ValidateChildren(ValidationConstraints)	(Inherited from Form.)
	WndProc	(Inherited from Form.)




















Properties
















Name	Description
 AcceptButton	Gets or sets the button on the form that is clicked when the user presses the ENTER key. (Inherited from Form.)
 AccessibilityObject	Gets the AccessibleObject assigned to the control. (Inherited from Control.)
 AccessibleDefaultActionDescription	Gets or sets the default action description of the control for use by accessibility client applications. (Inherited from Control.)
 AccessibleDescription	Gets or sets the description of the control used by accessibility client applications. (Inherited from Control.)
 AccessibleName	Gets or sets the name of the control used by accessibility client applications. (Inherited from Control.)
 AccessibleRole	Gets or sets the accessible role of the control (Inherited from Control.)
 ActiveControl	Gets or sets the active control on the container control. (Inherited from ContainerControl.)
 ActiveMdiChild	Gets the currently active multiple-document interface (MDI) child window. (Inherited from Form.)
 AllowDrop	Gets or sets a value indicating whether the control can accept data that the user drags onto it. (Inherited from Control.)
 AllowTransparency	Gets or sets a value indicating whether the opacity of the form can be adjusted. (Inherited from Form.)
 Anchor	Gets or sets the edges of the container to which a control is bound and determines how a control is resized with its parent. (Inherited from Control.)
 AutoScale	Obsolete. Gets or sets a value indicating whether the form adjusts its size to fit the height of the font used on the form and scales its controls. (Inherited from Form.)
 AutoScaleBaseSize	Gets or sets the base size used for autoscaling of the form. (Inherited from Form.)
 AutoScaleDimensions	Gets or sets the dimensions that the control was designed to. (Inherited from ContainerControl.)





















	AutoScaleFactor	Gets the scaling factor between the current and design-time automatic scaling dimensions. (Inherited from ContainerControl .)
	AutoScaleMode	Gets or sets the automatic scaling mode of the control. (Inherited from ContainerControl .)
	AutoScroll	Gets or sets a value indicating whether the form enables autoscrolling. (Inherited from Form .)
	AutoScrollMargin	Gets or sets the size of the auto-scroll margin. (Inherited from ScrollableControl .)
	AutoScrollMinSize	Gets or sets the minimum size of the auto-scroll. (Inherited from ScrollableControl .)
	AutoScrollOffset	Gets or sets where this control is scrolled to in ScrollControlIntoView(Control) . (Inherited from Control .)
	AutoScrollPosition	Gets or sets the location of the auto-scroll position. (Inherited from ScrollableControl .)
	AutoSize	Resize the form according to the setting of AutoSizeMode . (Inherited from Form .)
	AutoSizeMode	Gets or sets the mode by which the form automatically resizes itself. (Inherited from Form .)
	AutoValidate	(Inherited from Form .)
	BackColor	(Inherited from Form .)
	BackgroundImage	Gets or sets the background image displayed in the control. (Inherited from Control .)
	BackgroundImageLayout	Gets or sets the background image layout as defined in the ImageLayout enumeration. (Inherited from Control .)
	BindingContext	(Inherited from ContainerControl .)
	Bottom	Gets the distance, in pixels, between the bottom edge of the control and the top edge of its container's client area. (Inherited from Control .)
	Bounds	Gets or sets the size and location of the control including its nonclient elements, in pixels, relative to the parent control. (Inherited from Control .)
	CancelButton	Gets or sets the button control that is clicked when the user presses the ESC key. (Inherited from Form .)
	CanEnableIme	Gets a value indicating whether the ImeMode property can be set to an active value, to enable IME support. (Inherited from ContainerControl .)
	CanFocus	Gets a value indicating whether the control can receive focus. (Inherited from Control .)
	CanRaiseEvents	Determines if events can be raised on the control. (Inherited


















		from Control .)
	CanSelect	Gets a value indicating whether the control can be selected. (Inherited from Control .)
	Capture	Gets or sets a value indicating whether the control has captured the mouse. (Inherited from Control .)
	CausesValidation	Gets or sets a value indicating whether the control causes validation to be performed on any controls that require validation when it receives focus. (Inherited from Control .)
	ClientRectangle	Gets the rectangle that represents the client area of the control. (Inherited from Control .)
	ClientSize	Gets or sets the size of the client area of the form. (Inherited from Form .)
	CompanyName	Gets the name of the company or creator of the application containing the control. (Inherited from Control .)
	Container	Gets the IContainer that contains the Component . (Inherited from Component .)
	ContainsFocus	Gets a value indicating whether the control, or one of its child controls, currently has the input focus. (Inherited from Control .)
	ContextMenu	Gets or sets the shortcut menu associated with the control. (Inherited from Control .)
	ContextMenuStrip	Gets or sets the ContextMenuStrip associated with this control. (Inherited from Control .)
	ControlBox	Gets or sets a value indicating whether a control box is displayed in the caption bar of the form. (Inherited from Form .)
	Controls	Gets the collection of controls contained within the control. (Inherited from Control .)
	Created	Gets a value indicating whether the control has been created. (Inherited from Control .)
	CreateParams	(Inherited from Form .)
	CurrentAutoScaleDimensions	Gets the current run-time dimensions of the screen. (Inherited from ContainerControl .)
	Cursor	Gets or sets the cursor that is displayed when the mouse pointer is over the control. (Inherited from Control .)
	DataBindings	Gets the data bindings for the control. (Inherited from Control .)
	DefaultCursor	Gets or sets the default cursor for the control. (Inherited from Control .)
	DefaultImeMode	Gets the default Input Method Editor (IME) mode supported by the control. (Inherited from Form .)
	DefaultMargin	Gets the space, in pixels, that is specified by default between controls. (Inherited from Control .)

 DefaultMaximumSize	Gets the length and height, in pixels, that is specified as the default maximum size of a control. (Inherited from Control .)
 DefaultMinimumSize	Gets the length and height, in pixels, that is specified as the default minimum size of a control. (Inherited from Control .)
 DefaultPadding	Gets the internal spacing, in pixels, of the contents of a control. (Inherited from Control .)
 DefaultSize	(Inherited from Form .)
 DesignMode	Gets a value that indicates whether the Component is currently in design mode. (Inherited from Component .)
 DesktopBounds	Gets or sets the size and location of the form on the Windows desktop. (Inherited from Form .)
 DesktopLocation	Gets or sets the location of the form on the Windows desktop. (Inherited from Form .)
 DialogResult	Gets or sets the dialog result for the form. (Inherited from Form .)
 DisplayRectangle	Gets the rectangle that represents the virtual display area of the control. (Inherited from ScrollableControl .)
 Disposing	Gets a value indicating whether the base Control class is in the process of disposing. (Inherited from Control .)
 Dock	Gets or sets which control borders are docked to its parent control and determines how a control is resized with its parent. (Inherited from Control .)
 DockPadding	Gets the dock padding settings for all edges of the control. (Inherited from ScrollableControl .)
 DoubleBuffered	Gets or sets a value indicating whether this control should redraw its surface using a secondary buffer to reduce or prevent flicker. (Inherited from Control .)
 Enabled	Gets or sets a value indicating whether the control can respond to user interaction. (Inherited from Control .)
 Events	Gets the list of event handlers that are attached to this Component . (Inherited from Component .)
 Focused	Gets a value indicating whether the control has input focus. (Inherited from Control .)
 Font	Gets or sets the font of the text displayed by the control. (Inherited from Control .)
 FontHeight	Gets or sets the height of the font of the control. (Inherited from Control .)
 ForeColor	Gets or sets the foreground color of the control. (Inherited from Control .)
 FormBorderStyle	Gets or sets the border style of the form. (Inherited from Form .)


 Handle	Gets the window handle that the control is bound to. (Inherited from Control .)
 HasChildren	Gets a value indicating whether the control contains one or more child controls. (Inherited from Control .)
 Height	Gets or sets the height of the control. (Inherited from Control .)
 HelpButton	Gets or sets a value indicating whether a Help button should be displayed in the caption box of the form. (Inherited from Form .)
 HorizontalScroll	Gets the characteristics associated with the horizontal scroll bar. (Inherited from ScrollableControl .)
 HScroll	Gets or sets a value indicating whether the horizontal scroll bar is visible. (Inherited from ScrollableControl .)
 Icon	Gets or sets the icon for the form. (Inherited from Form .)
 ImeMode	Gets or sets the Input Method Editor (IME) mode of the control. (Inherited from Control .)
 ImeModeBase	Gets or sets the IME mode of a control. (Inherited from Control .)
 InvokeRequired	Gets a value indicating whether the caller must call an invoke method when making method calls to the control because the caller is on a different thread than the one the control was created on. (Inherited from Control .)
 IsAccessible	Gets or sets a value indicating whether the control is visible to accessibility applications. (Inherited from Control .)
 IsDisposed	Gets a value indicating whether the control has been disposed of. (Inherited from Control .)
 IsHandleCreated	Gets a value indicating whether the control has a handle associated with it. (Inherited from Control .)
 IsMdiChild	Gets a value indicating whether the form is a multiple-document interface (MDI) child form. (Inherited from Form .)
 IsMdiContainer	Gets or sets a value indicating whether the form is a container for multiple-document interface (MDI) child forms. (Inherited from Form .)
 IsMirrored	Gets a value indicating whether the control is mirrored. (Inherited from Control .)
 IsRestrictedWindow	Gets a value indicating whether the form can use all windows and user input events without restriction. (Inherited from Form .)
 KeyPreview	Gets or sets a value indicating whether the form will receive key events before the event is passed to the control that has focus. (Inherited from Form .)
 LayoutEngine	Gets a cached instance of the control's layout engine. (Inherited from Control .)






















 Left	Gets or sets the distance, in pixels, between the left edge of the control and the left edge of its container's client area. (Inherited from Control .)
 Location	Gets or sets the Point that represents the upper-left corner of the Form in screen coordinates. (Inherited from Form .)
 MainMenuStrip	Gets or sets the primary menu container for the form. (Inherited from Form .)
 Margin	Gets or sets the space between controls. (Inherited from Form .)
 MaximizeBox	Gets or sets a value indicating whether the Maximize button is displayed in the caption bar of the form. (Inherited from Form .)
 MaximizedBounds	Gets and sets the size of the form when it is maximized. (Inherited from Form .)
 MaximumSize	Gets the maximum size the form can be resized to. (Inherited from Form .)
 MdiChildren	Gets an array of forms that represent the multiple-document interface (MDI) child forms that are parented to this form. (Inherited from Form .)
 MdiParent	Gets or sets the current multiple-document interface (MDI) parent form of this form. (Inherited from Form .)
 Menu	Gets or sets the MainMenu that is displayed in the form. (Inherited from Form .)
 MergedMenu	Gets the merged menu for the form. (Inherited from Form .)
 MinimizeBox	Gets or sets a value indicating whether the Minimize button is displayed in the caption bar of the form. (Inherited from Form .)
 MinimumSize	Gets or sets the minimum size the form can be resized to. (Inherited from Form .)
 Modal	Gets a value indicating whether this form is displayed modally. (Inherited from Form .)
 Name	Gets or sets the name of the control. (Inherited from Control .)
 Opacity	Gets or sets the opacity level of the form. (Inherited from Form .)
 OwnedForms	Gets an array of Form objects that represent all forms that are owned by this form. (Inherited from Form .)
 Owner	Gets or sets the form that owns this form. (Inherited from Form .)
 Padding	Gets or sets padding within the control. (Inherited from Control .)
 Parent	Gets or sets the parent container of the control. (Inherited from Control .)
 ParentForm	Gets the form that the container control is assigned to. (Inherited from ContainerControl .)





















	PreferredSize	Gets the size of a rectangular area into which the control can fit. (Inherited from Control .)
	ProductName	Gets the product name of the assembly containing the control. (Inherited from Control .)
	ProductVersion	Gets the version of the assembly containing the control. (Inherited from Control .)
	RecreatingHandle	Gets a value indicating whether the control is currently re-creating its handle. (Inherited from Control .)
	Region	Gets or sets the window region associated with the control. (Inherited from Control .)
	RenderRightToLeft	Obsolete. This property is now obsolete. (Inherited from Control .)
	ResizeRedraw	Gets or sets a value indicating whether the control redraws itself when resized. (Inherited from Control .)
	RestoreBounds	Gets the location and size of the form in its normal window state. (Inherited from Form .)
	Right	Gets the distance, in pixels, between the right edge of the control and the left edge of its container's client area. (Inherited from Control .)
	RightToLeft	Gets or sets a value indicating whether control's elements are aligned to support locales using right-to-left fonts. (Inherited from Control .)
	RightToLeftLayout	Gets or sets a value indicating whether right-to-left mirror placement is turned on. (Inherited from Form .)
	ScaleChildren	Gets a value that determines the scaling of child controls. (Inherited from Control .)
	ShowFocusCues	Gets a value indicating whether the control should display focus rectangles. (Inherited from Control .)
	ShowIcon	Gets or sets a value indicating whether an icon is displayed in the caption bar of the form. (Inherited from Form .)
	ShowInTaskbar	Gets or sets a value indicating whether the form is displayed in the Windows taskbar. (Inherited from Form .)
	ShowKeyboardCues	Gets a value indicating whether the user interface is in the appropriate state to show or hide keyboard accelerators. (Inherited from Control .)
	ShowWithoutActivation	Gets a value indicating whether the window will be activated when it is shown. (Inherited from Form .)
	Site	Gets or sets the site of the control. (Inherited from Control .)
	Size	Gets or sets the size of the form. (Inherited from Form .)
	SizeGripStyle	Gets or sets the style of the size grip to display in the lower-right






















		corner of the form. (Inherited from Form.)
	StartPosition	Gets or sets the starting position of the form at run time. (Inherited from Form.)
	TabIndex	Gets or sets the tab order of the control within its container. (Inherited from Form.)
	TabStop	Gets or sets a value indicating whether the user can give the focus to this control using the TAB key. (Inherited from Form.)
	Tag	Gets or sets the object that contains data about the control. (Inherited from Control.)
	Text	(Inherited from Form.)
	Top	Gets or sets the distance, in pixels, between the top edge of the control and the top edge of its container's client area. (Inherited from Control.)
	TopLevel	Gets or sets a value indicating whether to display the form as a top-level window. (Inherited from Form.)
	TopLevelControl	Gets the parent control that is not parented by another Windows Forms control. Typically, this is the outermost Form that the control is contained in. (Inherited from Control.)
	TopMost	Gets or sets a value indicating whether the form should be displayed as a topmost form. (Inherited from Form.)
	TransparencyKey	Gets or sets the color that will represent transparent areas of the form. (Inherited from Form.)
	UseWaitCursor	Gets or sets a value indicating whether to use the wait cursor for the current control and all child controls. (Inherited from Control.)
	VerticalScroll	Gets the characteristics associated with the vertical scroll bar. (Inherited from ScrollableControl.)
	Visible	Gets or sets a value indicating whether the control and all its child controls are displayed. (Inherited from Control.)
	VScroll	Gets or sets a value indicating whether the vertical scroll bar is visible. (Inherited from ScrollableControl.)
	Width	Gets or sets the width of the control. (Inherited from Control.)
	WindowState	Gets or sets a value that indicates whether form is minimized, maximized, or normal. (Inherited from Form.)
	WindowTarget	This property is not relevant for this class. (Inherited from Control.)












Events




Name	Description
 Activated	Occurs when the form is activated in code or by the user. (Inherited

		from Form .)
	AutoSizeChanged	Occurs when the AutoSize property changes. (Inherited from Form .)
	AutoValidateChanged	Occurs when the AutoValidate property changes. (Inherited from Form .)
	BackColorChanged	Occurs when the value of the BackColor property changes. (Inherited from Control .)
	BackgroundImageChanged	Occurs when the value of the BackgroundImage property changes. (Inherited from Control .)
	BackgroundImageLayoutChanged	Occurs when the BackgroundImageLayout property changes. (Inherited from Control .)
	BindingContextChanged	Occurs when the value of the BindingContext property changes. (Inherited from Control .)
	CausesValidationChanged	Occurs when the value of the CausesValidation property changes. (Inherited from Control .)
	ChangeUICues	Occurs when the focus or keyboard user interface (UI) cues change. (Inherited from Control .)
	Click	Occurs when the control is clicked. (Inherited from Control .)
	ClientSizeChanged	Occurs when the value of the ClientSize property changes. (Inherited from Control .)
	Closed	Occurs when the form is closed. (Inherited from Form .)
	Closing	Occurs when the form is closing. (Inherited from Form .)
	ContextMenuChanged	Occurs when the value of the ContextMenu property changes. (Inherited from Control .)
	ContextMenuStripChanged	Occurs when the value of the ContextMenuStrip property changes. (Inherited from Control .)
	ControlAdded	Occurs when a new control is added to the Control.ControlCollection . (Inherited from Control .)
	ControlRemoved	Occurs when a control is removed from the Control.ControlCollection . (Inherited from Control .)
	CursorChanged	Occurs when the value of the Cursor property changes. (Inherited from Control .)
	Deactivate	Occurs when the form loses focus and is no longer the active form. (Inherited from Form .)
	Disposed	Occurs when the component is disposed by a call to the Dispose() method. (Inherited from Component .)
	DockChanged	Occurs when the value of the Dock property changes. (Inherited from Control .)
	DoubleClick	Occurs when the control is double-clicked. (Inherited from

	Control.)
 DragDrop	Occurs when a drag-and-drop operation is completed. (Inherited from Control.)
 DragEnter	Occurs when an object is dragged into the control's bounds. (Inherited from Control.)
 DragLeave	Occurs when an object is dragged out of the control's bounds. (Inherited from Control.)
 DragOver	Occurs when an object is dragged over the control's bounds. (Inherited from Control.)
 EnabledChanged	Occurs when the Enabled property value has changed. (Inherited from Control.)
 Enter	Occurs when the control is entered. (Inherited from Control.)
 FontChanged	Occurs when the Font property value changes. (Inherited from Control.)
 ForeColorChanged	Occurs when the ForeColor property value changes. (Inherited from Control.)
 FormClosed	Occurs after the form is closed. (Inherited from Form.)
 FormClosing	Occurs before the form is closed. (Inherited from Form.)
 GiveFeedback	Occurs during a drag operation. (Inherited from Control.)
 GotFocus	Occurs when the control receives focus. (Inherited from Control.)
 HandleCreated	Occurs when a handle is created for the control. (Inherited from Control.)
 HandleDestroyed	Occurs when the control's handle is in the process of being destroyed. (Inherited from Control.)
 HelpButtonClicked	Occurs when the Help button is clicked. (Inherited from Form.)
 HelpRequested	Occurs when the user requests help for a control. (Inherited from Control.)
 ImeModeChanged	Occurs when the ImeMode property has changed. (Inherited from Control.)
 InputLanguageChanged	Occurs after the input language of the form has changed. (Inherited from Form.)
 InputLanguageChanging	Occurs when the user attempts to change the input language for the form. (Inherited from Form.)
 Invalidated	Occurs when a control's display requires redrawing. (Inherited from Control.)
 KeyDown	Occurs when a key is pressed while the control has focus. (Inherited from Control.)
 KeyPress	Occurs when a key is pressed while the control has focus. (Inherited from Control.)

 KeyUp	Occurs when a key is released while the control has focus. (Inherited from Control .)
 Layout	Occurs when a control should reposition its child controls. (Inherited from Control .)
 Leave	Occurs when the input focus leaves the control. (Inherited from Control .)
 Load	Occurs before a form is displayed for the first time. (Inherited from Form .)
 LocationChanged	Occurs when the Location property value has changed. (Inherited from Control .)
 LostFocus	Occurs when the control loses focus. (Inherited from Control .)
 MarginChanged	Occurs when the Margin property changes. (Inherited from Form .)
 MaximizedBoundsChanged	Occurs when the value of the MaximizedBounds property has changed. (Inherited from Form .)
 MaximumSizeChanged	Occurs when the value of the MaximumSize property has changed. (Inherited from Form .)
 MdiChildActivate	Occurs when a multiple-document interface (MDI) child form is activated or closed within an MDI application. (Inherited from Form .)
 MenuComplete	Occurs when the menu of a form loses focus. (Inherited from Form .)
 MenuStart	Occurs when the menu of a form receives focus. (Inherited from Form .)
 MinimumSizeChanged	Occurs when the value of the MinimumSize property has changed. (Inherited from Form .)
 MouseCaptureChanged	Occurs when the control loses mouse capture. (Inherited from Control .)
 MouseClicked	Occurs when the control is clicked by the mouse. (Inherited from Control .)
 MouseDoubleClick	Occurs when the control is double clicked by the mouse. (Inherited from Control .)
 MouseDown	Occurs when the mouse pointer is over the control and a mouse button is pressed. (Inherited from Control .)
 MouseEnter	Occurs when the mouse pointer enters the control. (Inherited from Control .)
 MouseHover	Occurs when the mouse pointer rests on the control. (Inherited from Control .)
 MouseLeave	Occurs when the mouse pointer leaves the control. (Inherited from Control .)
 MouseMove	Occurs when the mouse pointer is moved over the control.

		(Inherited from Control.)
	MouseUp	Occurs when the mouse pointer is over the control and a mouse button is released. (Inherited from Control.)
	MouseWheel	Occurs when the mouse wheel moves while the control has focus. (Inherited from Control.)
	Move	Occurs when the control is moved. (Inherited from Control.)
	PaddingChanged	Occurs when the control's padding changes. (Inherited from Control.)
	Paint	Occurs when the control is redrawn. (Inherited from Control.)
	ParentChanged	Occurs when the Parent property value changes. (Inherited from Control.)
	PreviewKeyDown	Occurs before the KeyDown event when a key is pressed while focus is on this control. (Inherited from Control.)
	QueryAccessibilityHelp	Occurs when AccessibleObject is providing help to accessibility applications. (Inherited from Control.)
	QueryContinueDrag	Occurs during a drag-and-drop operation and enables the drag source to determine whether the drag-and-drop operation should be canceled. (Inherited from Control.)
	RegionChanged	Occurs when the value of the Region property changes. (Inherited from Control.)
	Resize	Occurs when the control is resized. (Inherited from Control.)
	ResizeBegin	Occurs when a form enters resizing mode. (Inherited from Form.)
	ResizeEnd	Occurs when a form exits resizing mode. (Inherited from Form.)
	RightToLeftChanged	Occurs when the RightToLeft property value changes. (Inherited from Control.)
	RightToLeftLayoutChanged	Occurs after the value of the RightToLeftLayout property changes. (Inherited from Form.)
	Scroll	Occurs when the user or code scrolls through the client area. (Inherited from ScrollableControl.)
	Shown	Occurs whenever the form is first displayed. (Inherited from Form.)
	SizeChanged	Occurs when the Size property value changes. (Inherited from Control.)
	StyleChanged	Occurs when the control style changes. (Inherited from Control.)
	SystemColorsChanged	Occurs when the system colors change. (Inherited from Control.)
	TabIndexChanged	Occurs when the value of the TabIndex property changes. (Inherited from Form.)
	TabStopChanged	Occurs when the TabStop property changes. (Inherited from Form.)
	TextChanged	Occurs when the Text property value changes. (Inherited from Control.)

 Validated	Occurs when the control is finished validating. (Inherited from Control .)
 Validating	Occurs when the control is validating. (Inherited from Control .)
 VisibleChanged	Occurs when the Visible property value changes. (Inherited from Control .)

See Also

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

ProgressDialog Constructor

Initialises the [ProgressDialog](#).

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public ProgressDialog(  
    string caption  
)
```

VB

```
Public Sub New (  
    caption As String  
)
```

C++

```
public:  
ProgressDialog(  
    String^ caption  
)
```

F#

```
new :  
    caption : string -> ProgressDialog
```

Parameters

caption

Type: [System.String](#)

The title for the dialog window.

See Also
















[ProgressDialog Class](#)


















[Woodstocks.WoodstocksIMS.Presentation Namespace](#)



















ProgressDialog.ProgressDialog Methods













The [ProgressDialog](#) type exposes the following members.





















Methods



























	Name	Description
	AccessibilityNotifyClients(AccessibleEvents, Int32)	Notifies the accessibility client applications of the specified AccessibleEvents for the specified child control. (Inherited from Control .)
	AccessibilityNotifyClients(AccessibleEvents, Int32, Int32)	Notifies the accessibility client applications of the specified AccessibleEvents for the specified child control . (Inherited from Control .)
	Activate	Activates the form and gives it focus. (Inherited from Form .)
	ActivateMdiChild	Activates the MDI child of a form. (Inherited from Form .)
	AddOwnedForm	Adds an owned form to this form. (Inherited from Form .)
	AdjustFormScrollbars	Adjusts the scroll bars on the container based on the current control positions and the control currently selected. (Inherited from Form .)
	ApplyAutoScaling	Obsolete. Resizes the form according to the current value of the AutoScaleBaseSize property and the size of the current font. (Inherited from Form .)
	BeginInvoke(Delegate)	Executes the specified delegate asynchronously on the thread that the control's underlying handle was created on. (Inherited from Control .)
	BeginInvoke(Delegate, Object[])	Executes the specified delegate asynchronously with the specified arguments, on the thread that the control's underlying handle was created on. (Inherited from Control .)
	BringToFront	Brings the control to the front of the z-order. (Inherited from Control .)
	CenterToParent	Centers the position of the form within the bounds of the parent form. (Inherited from Form .)
	CenterToScreen	Centers the form on the current screen. (Inherited from Form .)
	Close	Closes the form. (Inherited from Form .)
	Completed	Event Listener that can be used to handle task completion events.
	Contains	Retrieves a value indicating whether the specified

























		control is a child of the control. (Inherited from Control .)
	CreateAccessibilityInstance	Creates a new accessibility object for the control. (Inherited from Control .)
	CreateControl	Forces the creation of the visible control, including the creation of the handle and any visible child controls. (Inherited from Control .)
	CreateControlsInstance	(Inherited from Form .)
	CreateGraphics	Creates the Graphics for the control. (Inherited from Control .)
	CreateHandle	Creates the handle for the form. If a derived class overrides this function, it must call the base implementation. (Inherited from Form .)
	CreateObjRef	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object. (Inherited from MarshalByRefObject .)
	DefWndProc	(Inherited from Form .)
	DestroyHandle	Destroys the handle associated with the control. (Inherited from Control .)
	Dispose()	Releases all resources used by the Component . (Inherited from Component .)
	Dispose(Boolean)	Clean up any resources being used. (Overrides Form.Dispose(Boolean) .)
	DoDragDrop	Begins a drag-and-drop operation. (Inherited from Control .)
	DrawToBitmap	Supports rendering to the specified bitmap. (Inherited from Control .)
	EndInvoke	Retrieves the return value of the asynchronous operation represented by the IAsyncResult passed. (Inherited from Control .)
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	Finalize	Releases unmanaged resources and performs other cleanup operations before the Component is reclaimed by garbage collection. (Inherited from Component .)
	FindForm	Retrieves the form that the control is on. (Inherited from Control .)
	Focus	Sets input focus to the control. (Inherited from Control .)



















	GetAccessibilityObjectById	Retrieves the specified AccessibleObject . (Inherited from Control .)
	GetAutoSizeMode	Retrieves a value indicating how a control will behave when its AutoSize property is enabled. (Inherited from Control .)
	GetChildAtPoint(Point)	Retrieves the child control that is located at the specified coordinates. (Inherited from Control .)
	GetChildAtPoint(Point, GetChildAtPointSkip)	Retrieves the child control that is located at the specified coordinates, specifying whether to ignore child controls of a certain type. (Inherited from Control .)
	GetContainerControl	Returns the next ContainerControl up the control's chain of parent controls. (Inherited from Control .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetLifetimeService	Retrieves the current lifetime service object that controls the lifetime policy for this instance. (Inherited from MarshalByRefObject .)
	GetNextControl	Retrieves the next control forward or back in the tab order of child controls. (Inherited from Control .)
	GetPreferredSize	Retrieves the size of a rectangular area into which a control can be fitted. (Inherited from Control .)
	GetScaledBounds	(Inherited from Form .)
	GetScrollState	Determines whether the specified flag has been set. (Inherited from ScrollableControl .)
	GetService	Returns an object that represents a service provided by the Component or by its Container . (Inherited from Component .)
	GetStyle	Retrieves the value of the specified control style bit for the control. (Inherited from Control .)
	GetTopLevel	Determines if the control is a top-level control. (Inherited from Control .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	Hide	Conceals the control from the user. (Inherited from Control .)
	InitializeLifetimeService	Obtains a lifetime service object to control the lifetime policy for this instance. (Inherited from MarshalByRefObject .)
	InitLayout	Called after the control has been added to another container. (Inherited from Control .)

























 Invalidate()	Invalidate the entire surface of the control and causes the control to be redrawn. (Inherited from Control .)
 Invalidate(Region)	Invalidate the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. (Inherited from Control .)
 Invalidate(Boolean)	Invalidate a specific region of the control and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control .)
 Invalidate(Rectangle)	Invalidate the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. (Inherited from Control .)
 Invalidate(Region, Boolean)	Invalidate the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control .)
 Invalidate(Rectangle, Boolean)	Invalidate the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control .)
 Invoke(Delegate)	Executes the specified delegate on the thread that owns the control's underlying window handle. (Inherited from Control .)
 Invoke(Delegate, Object[])	Executes the specified delegate, on the thread that owns the control's underlying window handle, with the specified list of arguments. (Inherited from Control .)
 InvokeGotFocus	Raises the GotFocus event for the specified control. (Inherited from Control .)
 InvokeLostFocus	Raises the LostFocus event for the specified control. (Inherited from Control .)
 InvokeOnClick	Raises the Click event for the specified control. (Inherited from Control .)
 InvokePaint	Raises the Paint event for the specified control.






















		(Inherited from Control .)
	InvokePaintBackground	Raises the PaintBackground event for the specified control. (Inherited from Control .)
	IsInputChar	Determines if a character is an input character that the control recognizes. (Inherited from Control .)
	IsInputKey	Determines whether the specified key is a regular input key or a special key that requires preprocessing. (Inherited from Control .)
	LayoutMdi	Arranges the multiple-document interface (MDI) child forms within the MDI parent form. (Inherited from Form .)
	MemberwiseClone()	Creates a shallow copy of the current Object . (Inherited from Object .)
	MemberwiseClone(Boolean)	Creates a shallow copy of the current MarshalByRefObject object. (Inherited from MarshalByRefObject .)
	NotifyInvalidate	Raises the Invalidated event with a specified region of the control to invalidate. (Inherited from Control .)
	OnActivated	Raises the Activated event. (Inherited from Form .)
	OnAutoSizeChanged	Raises the AutoSizeChanged event. (Inherited from Control .)
	OnAutoValidateChanged	Raises the AutoValidateChanged event. (Inherited from ContainerControl .)
	OnBackColorChanged	Raises the BackColorChanged event. (Inherited from Control .)
	OnBackgroundImageChanged	Raises the BackgroundImageChanged event. (Inherited from Form .)
	OnBackgroundImageLayoutChanged	Raises the BackgroundImageLayoutChanged event. (Inherited from Form .)
	OnBindingContextChanged	Raises the BindingContextChanged event. (Inherited from Control .)
	OnCausesValidationChanged	Raises the CausesValidationChanged event. (Inherited from Control .)
	OnChangeUICues	Raises the ChangeUICues event. (Inherited from Control .)
	OnClick	Raises the Click event. (Inherited from Control .)
	OnClientSizeChanged	Raises the ClientSizeChanged event. (Inherited from Control .)
	OnClosed	Raises the Closed event. (Inherited from Form .)
	OnClosing	Raises the Closing event. (Inherited from Form .)
















	OnContextMenuChanged	Raises the ContextMenuChanged event. (Inherited from Control .)
	OnContextMenuStripChanged	Raises the ContextMenuStripChanged event. (Inherited from Control .)
	OnControlAdded	Raises the ControlAdded event. (Inherited from Control .)
	OnControlRemoved	Raises the ControlRemoved event. (Inherited from Control .)
	OnCreateControl	Raises the CreateControl event. (Inherited from Form .)
	OnCursorChanged	Raises the CursorChanged event. (Inherited from Control .)
	OnDeactivate	Raises the Deactivate event. (Inherited from Form .)
	OnDockChanged	Raises the DockChanged event. (Inherited from Control .)
	OnDoubleClick	Raises the DoubleClick event. (Inherited from Control .)
	OnDragDrop	Raises the DragDrop event. (Inherited from Control .)
	OnDragEnter	Raises the DragEnter event. (Inherited from Control .)
	OnDragLeave	Raises the DragLeave event. (Inherited from Control .)
	OnDragOver	Raises the DragOver event. (Inherited from Control .)
	OnEnabledChanged	(Inherited from Form .)
	OnEnter	Raises the Enter event. (Inherited from Form .)
	OnFontChanged	(Inherited from Form .)
	OnForeColorChanged	Raises the ForeColorChanged event. (Inherited from Control .)
	OnFormClosed	Raises the FormClosed event. (Inherited from Form .)
	OnFormClosing	Raises the FormClosing event. (Inherited from Form .)
	OnGiveFeedback	Raises the GiveFeedback event. (Inherited from Control .)
	OnGotFocus	Raises the GotFocus event. (Inherited from Control .)
	OnHandleCreated	(Inherited from Form .)
	OnHandleDestroyed	(Inherited from Form .)
	OnHelpButtonClicked	Raises the HelpButtonClicked event. (Inherited from Form .)
	OnHelpRequested	Raises the HelpRequested event. (Inherited from Control .)
	OnImeModeChanged	Raises the ImeModeChanged event. (Inherited from

















		Control.)
	OnInputLanguageChanged	Raises the InputLanguageChanged event. (Inherited from Form.)
	OnInputLanguageChanging	Raises the InputLanguageChanging event. (Inherited from Form.)
	OnInvalidated	Raises the Invalidated event. (Inherited from Control.)
	OnKeyDown	Raises the KeyDown event. (Inherited from Control.)
	OnKeyPress	Raises the KeyPress event. (Inherited from Control.)
	OnKeyUp	Raises the KeyUp event. (Inherited from Control.)
	OnLayout	Raises the Layout event. (Inherited from Form.)
	OnLeave	Raises the Leave event. (Inherited from Control.)
	OnLoad	Raises the Load event. (Inherited from Form.)
	OnLocationChanged	Raises the LocationChanged event. (Inherited from Control.)
	OnLostFocus	Raises the LostFocus event. (Inherited from Control.)
	OnMarginChanged	Raises the MarginChanged event. (Inherited from Control.)
	OnMaximizedBoundsChanged	Raises the MaximizedBoundsChanged event. (Inherited from Form.)
	OnMaximumSizeChanged	Raises the MaximumSizeChanged event. (Inherited from Form.)
	OnMdiChildActivate	Raises the MdiChildActivate event. (Inherited from Form.)
	OnMenuComplete	Raises the MenuComplete event. (Inherited from Form.)
	OnMenuStart	Raises the MenuStart event. (Inherited from Form.)
	OnMinimumSizeChanged	Raises the MinimumSizeChanged event. (Inherited from Form.)
	OnMouseCaptureChanged	Raises the MouseCaptureChanged event. (Inherited from Control.)
	OnMouseClicked	Raises the MouseClicked event. (Inherited from Control.)
	OnMouseDoubleClick	Raises the MouseDoubleClick event. (Inherited from Control.)
	OnMouseDown	Raises the MouseDown event. (Inherited from Control.)
	OnMouseEnter	Raises the MouseEnter event. (Inherited from Control.)
	OnMouseHover	Raises the MouseHover event. (Inherited from





















		Control.)
	OnMouseLeave	Raises the MouseLeave event. (Inherited from Control.)
	OnMouseMove	Raises the MouseMove event. (Inherited from Control.)
	OnMouseUp	Raises the MouseUp event. (Inherited from Control.)
	OnMouseWheel	Raises the MouseWheel event. (Inherited from ScrollableControl.)
	OnMove	Raises the Move event. (Inherited from Control.)
	OnNotifyMessage	Notifies the control of Windows messages. (Inherited from Control.)
	OnPaddingChanged	Raises the PaddingChanged event. (Inherited from ScrollableControl.)
	OnPaint	(Inherited from Form.)
	OnPaintBackground	Paints the background of the control. (Inherited from ScrollableControl.)
	OnParentBackColorChanged	Raises the BackColorChanged event when the BackColor property value of the control's container changes. (Inherited from Control.)
	OnParentBackgroundImageChanged	Raises the BackgroundImageChanged event when the BackgroundImage property value of the control's container changes. (Inherited from Control.)
	OnParentBindingContextChanged	Raises the BindingContextChanged event when the BindingContext property value of the control's container changes. (Inherited from Control.)
	OnParentChanged	(Inherited from ContainerControl.)
	OnParentCursorChanged	Raises the CursorChanged event. (Inherited from Control.)
	OnParentEnabledChanged	Raises the EnabledChanged event when the Enabled property value of the control's container changes. (Inherited from Control.)
	OnParentFontChanged	Raises the FontChanged event when the Font property value of the control's container changes. (Inherited from Control.)
	OnParentForeColorChanged	Raises the ForeColorChanged event when the ForeColor property value of the control's container changes. (Inherited from Control.)
	OnParentRightToLeftChanged	Raises the RightToLeftChanged event when the RightToLeft property value of the control's container changes. (Inherited from Control.)







	OnParentVisibleChanged	Raises the VisibleChanged event when the Visible property value of the control's container changes. (Inherited from Control .)
	OnPreviewKeyDown	Raises the PreviewKeyDown event. (Inherited from Control .)
	OnPrint	Raises the Paint event. (Inherited from Control .)
	OnQueryContinueDrag	Raises the QueryContinueDrag event. (Inherited from Control .)
	OnRegionChanged	Raises the RegionChanged event. (Inherited from Control .)
	OnResize	(Inherited from Form .)
	OnResizeBegin	Raises the ResizeBegin event. (Inherited from Form .)
	OnResizeEnd	Raises the ResizeEnd event. (Inherited from Form .)
	OnRightToLeftChanged	(Inherited from ScrollableControl .)
	OnRightToLeftLayoutChanged	Raises the RightToLeftLayoutChanged event. (Inherited from Form .)
	OnScroll	Raises the Scroll event. (Inherited from ScrollableControl .)
	OnShown	Raises the Shown event. (Inherited from Form .)
	OnSizeChanged	Raises the SizeChanged event. (Inherited from Control .)
	OnStyleChanged	(Inherited from Form .)
	OnSystemColorsChanged	Raises the SystemColorsChanged event. (Inherited from Control .)
	OnTabIndexChanged	Raises the TabIndexChanged event. (Inherited from Control .)
	OnTabStopChanged	Raises the TabStopChanged event. (Inherited from Control .)
	OnTextChanged	(Inherited from Form .)
	OnValidated	Raises the Validated event. (Inherited from Control .)
	OnValidating	Raises the Validating event. (Inherited from Control .)
	OnVisibleChanged	Raises the VisibleChanged event. (Inherited from Form .)
	PerformAutoScale	Performs scaling of the container control and its children. (Inherited from ContainerControl .)
	PerformLayout()	Forces the control to apply layout logic to all its child controls. (Inherited from Control .)
	PerformLayout(Control, String)	Forces the control to apply layout logic to all its child controls. (Inherited from Control .)

	PointToClient	Computes the location of the specified screen point into client coordinates. (Inherited from Control .)
	PointToScreen	Computes the location of the specified client point into screen coordinates. (Inherited from Control .)
	PreProcessControlMessage	Preprocesses keyboard or input messages within the message loop before they are dispatched. (Inherited from Control .)
	PreProcessMessage	Preprocesses keyboard or input messages within the message loop before they are dispatched. (Inherited from Control .)
	ProcessCmdKey	Processes a command key. (Inherited from Form .)
	ProcessDialogChar	Processes a dialog character. (Inherited from Form .)
	ProcessDialogKey	Processes a dialog box key. (Inherited from Form .)
	ProcessKeyEventArgs	Processes a key message and generates the appropriate control events. (Inherited from Control .)
	ProcessKeyMessage	Processes a keyboard message. (Inherited from Control .)
	ProcessKeyPreview	(Inherited from Form .)
	ProcessMnemonic	Processes a mnemonic character. (Inherited from Form .)
	ProcessTabKey	(Inherited from Form .)
	ProgressChanged	Event Listener that can be used to handle progress changes.
	RaiseDragEvent	Raises the appropriate drag event. (Inherited from Control .)
	RaiseKeyEvent	Raises the appropriate key event. (Inherited from Control .)
	RaiseMouseEvent	Raises the appropriate mouse event. (Inherited from Control .)
	RaisePaintEvent	Raises the appropriate paint event. (Inherited from Control .)
	RecreateHandle	Forces the re-creation of the handle for the control. (Inherited from Control .)
	RectangleToClient	Computes the size and location of the specified screen rectangle in client coordinates. (Inherited from Control .)
	RectangleToScreen	Computes the size and location of the specified client rectangle in screen coordinates. (Inherited from Control .)
	Refresh	Forces the control to invalidate its client area and

		immediately redraw itself and any child controls. (Inherited from Control .)
	RemoveOwnedForm	Removes an owned form from this form. (Inherited from Form .)
	ResetBackColor	Resets the BackColor property to its default value. (Inherited from Control .)
	ResetBindings	Causes a control bound to the BindingSource to reread all the items in the list and refresh their displayed values. (Inherited from Control .)
	ResetCursor	Resets the Cursor property to its default value. (Inherited from Control .)
	ResetFont	Resets the Font property to its default value. (Inherited from Control .)
	ResetForeColor	Resets the ForeColor property to its default value. (Inherited from Control .)
	ResetIMEMode	Resets the IMEMode property to its default value. (Inherited from Control .)
	ResetMouseEventArgs	Resets the control to handle the MouseLeave event. (Inherited from Control .)
	ResetRightToLeft	Resets the RightToLeft property to its default value. (Inherited from Control .)
	ResetText	Resets the Text property to its default value. (Inherited from Control .)
	ResumeLayout()	Resumes usual layout logic. (Inherited from Control .)
	ResumeLayout(Boolean)	Resumes usual layout logic, optionally forcing an immediate layout of pending layout requests. (Inherited from Control .)
	RtlTranslateAlignment(HorizontalAlignment)	Converts the specified HorizontalAlignment to the appropriate HorizontalAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateAlignment(LeftRightAlignment)	Converts the specified LeftRightAlignment to the appropriate LeftRightAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateAlignment(ContentAlignment)	Converts the specified ContentAlignment to the appropriate ContentAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateContent	Converts the specified ContentAlignment to the appropriate ContentAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateHorizontal	Converts the specified HorizontalAlignment to the appropriate HorizontalAlignment to support right-to-

		left text. (Inherited from Control .)
	RtlTranslateLeftRight	Converts the specified LeftRightAlignment to the appropriate LeftRightAlignment to support right-to-left text. (Inherited from Control .)
	Scale(Single)	Obsolete. Scales the control and any child controls. (Inherited from Control .)
	Scale(SizeF)	Scales the control and all child controls by the specified scaling factor. (Inherited from Control .)
	Scale(Single, Single)	Obsolete. Scales the entire control and any child controls. (Inherited from Control .)
	ScaleControl	Scales the location, size, padding, and margin of a control. (Inherited from Form .)
	ScaleCore	Performs scaling of the form. (Inherited from Form .)
	ScrollControlIntoView	Scrolls the specified child control into view on an auto-scroll enabled control. (Inherited from ScrollableControl .)
	ScrollToControl	Calculates the scroll offset to the specified child control. (Inherited from ScrollableControl .)
	Select()	Activates the control. (Inherited from Control .)
	Select(Boolean, Boolean)	Selects this form, and optionally selects the next or previous control. (Inherited from Form .)
	SelectNextControl	Activates the next control. (Inherited from Control .)
	SendToBack	Sends the control to the back of the z-order. (Inherited from Control .)
	SetAutoScrollMargin	Sets the size of the auto-scroll margins. (Inherited from ScrollableControl .)
	SetAutoSizeMode	Sets a value indicating how a control will behave when its AutoSize property is enabled. (Inherited from Control .)
	SetBounds(Int32, Int32, Int32, Int32)	Sets the bounds of the control to the specified location and size. (Inherited from Control .)
	SetBounds(Int32, Int32, Int32, Int32, BoundsSpecified)	Sets the specified bounds of the control to the specified location and size. (Inherited from Control .)
	SetBoundsCore	(Inherited from Form .)
	SetClientSizeCore	Sets the client size of the form. This will adjust the bounds of the form to make the client size the requested size. (Inherited from Form .)
	SetDesktopBounds	Sets the bounds of the form in desktop coordinates.

		(Inherited from Form .)
	SetDesktopLocation	Sets the location of the form in desktop coordinates. (Inherited from Form .)
	SetDisplayRectLocation	Positions the display window to the specified value. (Inherited from ScrollableControl .)
	SetScrollState	Sets the specified scroll state flag. (Inherited from ScrollableControl .)
	SetStyle	Sets a specified ControlStyles flag to either true or false. (Inherited from Control .)
	SetTopLevel	Sets the control as the top-level control. (Inherited from Control .)
	SetVisibleCore	(Inherited from Form .)
	Show()	Displays the control to the user. (Inherited from Control .)
	Show(IWin32Window)	Shows the form with the specified owner to the user. (Inherited from Form .)
	ShowDialog()	Shows the form as a modal dialog box. (Inherited from Form .)
	ShowDialog(IWin32Window)	Shows the form as a modal dialog box with the specified owner. (Inherited from Form .)
	SizeFromClientSize	Determines the size of the entire control from the height and width of its client area. (Inherited from Control .)
	SuspendLayout	Temporarily suspends the layout logic for the control. (Inherited from Control .)
	ToString	Gets a string representing the current instance of the form. (Inherited from Form .)
	Update	Causes the control to redraw the invalidated regions within its client area. (Inherited from Control .)
	UpdateBounds()	Updates the bounds of the control with the current size and location. (Inherited from Control .)
	UpdateBounds(Int32, Int32, Int32, Int32)	Updates the bounds of the control with the specified size and location. (Inherited from Control .)
	UpdateBounds(Int32, Int32, Int32, Int32, Int32, Int32)	Updates the bounds of the control with the specified size, location, and client size. (Inherited from Control .)
	UpdateDefaultButton	Updates which button is the default button. (Inherited from Form .)
	UpdateProgress	Updates the progress displayed by the dialog window.
	UpdateStyles	Forces the assigned styles to be reapplied to the control. (Inherited from Control .)

	UpdateZOrder	Updates the control in its parent's z-order. (Inherited from Control .)
	Validate()	Verifies the value of the control losing focus by causing the Validating and Validated events to occur, in that order. (Inherited from ContainerControl .)
	Validate(Boolean)	Verifies the value of the control that is losing focus; conditionally dependent on whether automatic validation is turned on. (Inherited from ContainerControl .)
	ValidateChildren()	(Inherited from Form .)
	ValidateChildren(ValidationConstraints)	(Inherited from Form .)
	WndProc	(Inherited from Form .)

See Also

[ProgressDialog Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

ProgressDialog.Completed Method

Event Listener that can be used to handle task completion events.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void Completed(  
    Object sender,  
    AsyncCompletedEventArgs e  
)
```

VB

```
Public Sub Completed (  
    sender As Object,  
    e As AsyncCompletedEventArgs  
)
```

C++

```
public:  
void Completed(  
    Object^ sender,  
    AsyncCompletedEventArgs^ e  
)
```

F#

```
member Completed :  
    sender : Object *  
    e : AsyncCompletedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

The sender of the event.

e

Type: [System.ComponentModel.AsyncCompletedEventArgs](#)

Event data sent by the sender.



See Also

[ProgressDialog Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

ProgressDialog.Dispose Method

Overload List

	Name	Description
	Dispose()	Releases all resources used by the Component . (Inherited from Component .)
	Dispose(Boolean)	Clean up any resources being used. (Overrides Form.Dispose(Boolean) .)

See Also

[ProgressDialog Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

ProgressDialog.Dispose Method (Boolean)

Clean up any resources being used.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Protected Overrides Sub Dispose (  
    disposing As Boolean  
)
```

C++

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

F#

```
abstract Dispose :  
    disposing : bool -> unit  
override Dispose :  
    disposing : bool -> unit
```

Parameters

disposing

Type: [System.Boolean](#)

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

See Also

[ProgressDialog Class](#)

[Dispose Overload](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

ProgressDialog.ProgressChanged Method

Event Listener that can be used to handle progress changes.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ProgressChanged(  
    Object sender,  
    ProgressChangedEventArgs e  
)
```

VB

```
Public Sub ProgressChanged (  
    sender As Object,  
    e As ProgressChangedEventArgs  
)
```

C++

```
public:  
void ProgressChanged(  
    Object^ sender,  
    ProgressChangedEventArgs^ e  
)
```

F#

```
member ProgressChanged :  
    sender : Object *  
    e : ProgressChangedEventArgs -> unit
```

Parameters

sender

Type: [System.Object](#)

[Missing <param name="sender"/> documentation for

"M:Woodstocks.WoodstocksIMS.Presentation.ProgressDialog.ProgressChanged(System.Object,System.ComponentModel.ProgressChangedEventArgs)"]

e

Type: [System.ComponentModel.ProgressChangedEventArgs](#)

[Missing <param name="e"/> documentation for

"M:Woodstocks.WoodstocksIMS.Presentation.ProgressDialog.ProgressChanged(System.Object,System.ComponentModel.ProgressChangedEventArgs)"]

A Sandcastle Documented Class Library

See Also

[ProgressDialog Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

ProgressDialog.UpdateProgress Method

Updates the progress displayed by the dialog window.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void UpdateProgress(  
    int progressPercentage  
)
```

VB

```
Public Sub UpdateProgress (  
    progressPercentage As Integer  
)
```

C++

```
public:  
void UpdateProgress(  
    int progressPercentage  
)
```

F#

```
member UpdateProgress :  
    progressPercentage : int -> unit
```

Parameters

progressPercentage

Type: [System.Int32](#)

[Missing <param name="progressPercentage"/> documentation for
"M:Woodstocks.WoodstocksIMS.Presentation.ProgressDialog.UpdateProgress(System.Int32)"]

See Also

















[ProgressDialog Class](#)





















[Woodstocks.WoodstocksIMS.Presentation Namespace](#)






















ProgressDialog.ProgressDialog Properties


















The [ProgressDialog](#) type exposes the following members.




















Properties






















Name	Description
 AcceptButton	Gets or sets the button on the form that is clicked when the user presses the ENTER key. (Inherited from Form .)
 AccessibilityObject	Gets the AccessibleObject assigned to the control. (Inherited from Control .)
 AccessibleDefaultActionDescription	Gets or sets the default action description of the control for use by accessibility client applications. (Inherited from Control .)
 AccessibleDescription	Gets or sets the description of the control used by accessibility client applications. (Inherited from Control .)
 AccessibleName	Gets or sets the name of the control used by accessibility client applications. (Inherited from Control .)
 AccessibleRole	Gets or sets the accessible role of the control (Inherited from Control .)
 ActiveControl	Gets or sets the active control on the container control. (Inherited from ContainerControl .)
 ActiveMdiChild	Gets the currently active multiple-document interface (MDI) child window. (Inherited from Form .)
 AllowDrop	Gets or sets a value indicating whether the control can accept data that the user drags onto it. (Inherited from Control .)
 AllowTransparency	Gets or sets a value indicating whether the opacity of the form can be adjusted. (Inherited from Form .)
 Anchor	Gets or sets the edges of the container to which a control is bound and determines how a control is resized with its parent. (Inherited from Control .)
 AutoScale	Obsolete. Gets or sets a value indicating whether the form adjusts its size to fit the height of the font used on the form and scales its controls. (Inherited from Form .)
 AutoScaleBaseSize	Gets or sets the base size used for autoscaling of the form. (Inherited from Form .)
 AutoScaleDimensions	Gets or sets the dimensions that the control was designed to. (Inherited from ContainerControl .)
 AutoScaleFactor	Gets the scaling factor between the current and design-time automatic scaling dimensions. (Inherited from ContainerControl .)
 AutoScaleMode	Gets or sets the automatic scaling mode of the control. (Inherited from ContainerControl .)









 AutoScroll	Gets or sets a value indicating whether the form enables autoscrolling. (Inherited from Form .)
 AutoScrollMargin	Gets or sets the size of the auto-scroll margin. (Inherited from ScrollableControl .)
 AutoScrollMinSize	Gets or sets the minimum size of the auto-scroll. (Inherited from ScrollableControl .)
 AutoScrollOffset	Gets or sets where this control is scrolled to in ScrollControlIntoView(Control) . (Inherited from Control .)
 AutoScrollPosition	Gets or sets the location of the auto-scroll position. (Inherited from ScrollableControl .)
 AutoSize	Resize the form according to the setting of AutoSizeMode . (Inherited from Form .)
 AutoSizeMode	Gets or sets the mode by which the form automatically resizes itself. (Inherited from Form .)
 AutoValidate	(Inherited from Form .)
 BackColor	(Inherited from Form .)
 BackgroundImage	Gets or sets the background image displayed in the control. (Inherited from Control .)
 BackgroundImageLayout	Gets or sets the background image layout as defined in the ImageLayout enumeration. (Inherited from Control .)
 BindingContext	(Inherited from ContainerControl .)
 Bottom	Gets the distance, in pixels, between the bottom edge of the control and the top edge of its container's client area. (Inherited from Control .)
 Bounds	Gets or sets the size and location of the control including its nonclient elements, in pixels, relative to the parent control. (Inherited from Control .)
 CancelButton	Gets or sets the button control that is clicked when the user presses the ESC key. (Inherited from Form .)
 CanEnableIme	Gets a value indicating whether the ImeMode property can be set to an active value, to enable IME support. (Inherited from ContainerControl .)
 CanFocus	Gets a value indicating whether the control can receive focus. (Inherited from Control .)
 CanRaiseEvents	Determines if events can be raised on the control. (Inherited from Control .)
 CanSelect	Gets a value indicating whether the control can be selected. (Inherited from Control .)
 Capture	Gets or sets a value indicating whether the control has captured the mouse. (Inherited from Control .)
















 CausesValidation	Gets or sets a value indicating whether the control causes validation to be performed on any controls that require validation when it receives focus. (Inherited from Control .)
 ClientRectangle	Gets the rectangle that represents the client area of the control. (Inherited from Control .)
 ClientSize	Gets or sets the size of the client area of the form. (Inherited from Form .)
 CompanyName	Gets the name of the company or creator of the application containing the control. (Inherited from Control .)
 Container	Gets the IContainer that contains the Component . (Inherited from Component .)
 ContainsFocus	Gets a value indicating whether the control, or one of its child controls, currently has the input focus. (Inherited from Control .)
 ContextMenu	Gets or sets the shortcut menu associated with the control. (Inherited from Control .)
 ContextMenuStrip	Gets or sets the ContextMenuStrip associated with this control. (Inherited from Control .)
 ControlBox	Gets or sets a value indicating whether a control box is displayed in the caption bar of the form. (Inherited from Form .)
 Controls	Gets the collection of controls contained within the control. (Inherited from Control .)
 Created	Gets a value indicating whether the control has been created. (Inherited from Control .)
 CreateParams	(Inherited from Form .)
 CurrentAutoScaleDimensions	Gets the current run-time dimensions of the screen. (Inherited from ContainerControl .)
 Cursor	Gets or sets the cursor that is displayed when the mouse pointer is over the control. (Inherited from Control .)
 DataBindings	Gets the data bindings for the control. (Inherited from Control .)
 DefaultCursor	Gets or sets the default cursor for the control. (Inherited from Control .)
 DefaultImeMode	Gets the default Input Method Editor (IME) mode supported by the control. (Inherited from Form .)
 DefaultMargin	Gets the space, in pixels, that is specified by default between controls. (Inherited from Control .)
 DefaultMaximumSize	Gets the length and height, in pixels, that is specified as the default maximum size of a control. (Inherited from Control .)
 DefaultMinimumSize	Gets the length and height, in pixels, that is specified as the default minimum size of a control. (Inherited from Control .)
 DefaultPadding	Gets the internal spacing, in pixels, of the contents of a control.

		(Inherited from Control .)
	DefaultSize	(Inherited from Form .)
	DesignMode	Gets a value that indicates whether the Component is currently in design mode. (Inherited from Component .)
	DesktopBounds	Gets or sets the size and location of the form on the Windows desktop. (Inherited from Form .)
	DesktopLocation	Gets or sets the location of the form on the Windows desktop. (Inherited from Form .)
	DialogResult	Gets or sets the dialog result for the form. (Inherited from Form .)
	DisplayRectangle	Gets the rectangle that represents the virtual display area of the control. (Inherited from ScrollableControl .)
	Disposing	Gets a value indicating whether the base Control class is in the process of disposing. (Inherited from Control .)
	Dock	Gets or sets which control borders are docked to its parent control and determines how a control is resized with its parent. (Inherited from Control .)
	DockPadding	Gets the dock padding settings for all edges of the control. (Inherited from ScrollableControl .)
	DoubleBuffered	Gets or sets a value indicating whether this control should redraw its surface using a secondary buffer to reduce or prevent flicker. (Inherited from Control .)
	Enabled	Gets or sets a value indicating whether the control can respond to user interaction. (Inherited from Control .)
	Events	Gets the list of event handlers that are attached to this Component . (Inherited from Component .)
	Focused	Gets a value indicating whether the control has input focus. (Inherited from Control .)
	Font	Gets or sets the font of the text displayed by the control. (Inherited from Control .)
	FontHeight	Gets or sets the height of the font of the control. (Inherited from Control .)
	ForeColor	Gets or sets the foreground color of the control. (Inherited from Control .)
	FormBorderStyle	Gets or sets the border style of the form. (Inherited from Form .)
	Handle	Gets the window handle that the control is bound to. (Inherited from Control .)
	HasChildren	Gets a value indicating whether the control contains one or more child controls. (Inherited from Control .)
	Height	Gets or sets the height of the control. (Inherited from Control .)

	HelpButton	Gets or sets a value indicating whether a Help button should be displayed in the caption box of the form. (Inherited from Form .)
	HorizontalScroll	Gets the characteristics associated with the horizontal scroll bar. (Inherited from ScrollableControl .)
	HScroll	Gets or sets a value indicating whether the horizontal scroll bar is visible. (Inherited from ScrollableControl .)
	Icon	Gets or sets the icon for the form. (Inherited from Form .)
	ImeMode	Gets or sets the Input Method Editor (IME) mode of the control. (Inherited from Control .)
	ImeModeBase	Gets or sets the IME mode of a control. (Inherited from Control .)
	InvokeRequired	Gets a value indicating whether the caller must call an invoke method when making method calls to the control because the caller is on a different thread than the one the control was created on. (Inherited from Control .)
	IsAccessible	Gets or sets a value indicating whether the control is visible to accessibility applications. (Inherited from Control .)
	IsDisposed	Gets a value indicating whether the control has been disposed of. (Inherited from Control .)
	IsHandleCreated	Gets a value indicating whether the control has a handle associated with it. (Inherited from Control .)
	IsMdiChild	Gets a value indicating whether the form is a multiple-document interface (MDI) child form. (Inherited from Form .)
	IsMdiContainer	Gets or sets a value indicating whether the form is a container for multiple-document interface (MDI) child forms. (Inherited from Form .)
	IsMirrored	Gets a value indicating whether the control is mirrored. (Inherited from Control .)
	IsRestrictedWindow	Gets a value indicating whether the form can use all windows and user input events without restriction. (Inherited from Form .)
	KeyPreview	Gets or sets a value indicating whether the form will receive key events before the event is passed to the control that has focus. (Inherited from Form .)
	LayoutEngine	Gets a cached instance of the control's layout engine. (Inherited from Control .)
	Left	Gets or sets the distance, in pixels, between the left edge of the control and the left edge of its container's client area. (Inherited from Control .)
	Location	Gets or sets the Point that represents the upper-left corner of the Form in screen coordinates. (Inherited from Form .)
	MainMenuStrip	Gets or sets the primary menu container for the form. (Inherited

		from Form.)
	Margin	Gets or sets the space between controls. (Inherited from Form.)
	MaximizeBox	Gets or sets a value indicating whether the Maximize button is displayed in the caption bar of the form. (Inherited from Form.)
	MaximizedBounds	Gets and sets the size of the form when it is maximized. (Inherited from Form.)
	MaximumSize	Gets the maximum size the form can be resized to. (Inherited from Form.)
	MdiChildren	Gets an array of forms that represent the multiple-document interface (MDI) child forms that are parented to this form. (Inherited from Form.)
	MdiParent	Gets or sets the current multiple-document interface (MDI) parent form of this form. (Inherited from Form.)
	Menu	Gets or sets the MainMenu that is displayed in the form. (Inherited from Form.)
	MergedMenu	Gets the merged menu for the form. (Inherited from Form.)
	MinimizeBox	Gets or sets a value indicating whether the Minimize button is displayed in the caption bar of the form. (Inherited from Form.)
	MinimumSize	Gets or sets the minimum size the form can be resized to. (Inherited from Form.)
	Modal	Gets a value indicating whether this form is displayed modally. (Inherited from Form.)
	Name	Gets or sets the name of the control. (Inherited from Control.)
	Opacity	Gets or sets the opacity level of the form. (Inherited from Form.)
	OwnedForms	Gets an array of Form objects that represent all forms that are owned by this form. (Inherited from Form.)
	Owner	Gets or sets the form that owns this form. (Inherited from Form.)
	Padding	Gets or sets padding within the control. (Inherited from Control.)
	Parent	Gets or sets the parent container of the control. (Inherited from Control.)
	ParentForm	Gets the form that the container control is assigned to. (Inherited from ContainerControl.)
	PreferredSize	Gets the size of a rectangular area into which the control can fit. (Inherited from Control.)
	ProductName	Gets the product name of the assembly containing the control. (Inherited from Control.)
	ProductVersion	Gets the version of the assembly containing the control.

		(Inherited from Control .)
	RecreatingHandle	Gets a value indicating whether the control is currently re-creating its handle. (Inherited from Control .)
	Region	Gets or sets the window region associated with the control. (Inherited from Control .)
	RenderRightToLeft	Obsolete. This property is now obsolete. (Inherited from Control .)
	ResizeRedraw	Gets or sets a value indicating whether the control redraws itself when resized. (Inherited from Control .)
	RestoreBounds	Gets the location and size of the form in its normal window state. (Inherited from Form .)
	Right	Gets the distance, in pixels, between the right edge of the control and the left edge of its container's client area. (Inherited from Control .)
	RightToLeft	Gets or sets a value indicating whether control's elements are aligned to support locales using right-to-left fonts. (Inherited from Control .)
	RightToLeftLayout	Gets or sets a value indicating whether right-to-left mirror placement is turned on. (Inherited from Form .)
	ScaleChildren	Gets a value that determines the scaling of child controls. (Inherited from Control .)
	ShowFocusCues	Gets a value indicating whether the control should display focus rectangles. (Inherited from Control .)
	ShowIcon	Gets or sets a value indicating whether an icon is displayed in the caption bar of the form. (Inherited from Form .)
	ShowInTaskbar	Gets or sets a value indicating whether the form is displayed in the Windows taskbar. (Inherited from Form .)
	ShowKeyboardCues	Gets a value indicating whether the user interface is in the appropriate state to show or hide keyboard accelerators. (Inherited from Control .)
	ShowWithoutActivation	Gets a value indicating whether the window will be activated when it is shown. (Inherited from Form .)
	Site	Gets or sets the site of the control. (Inherited from Control .)
	Size	Gets or sets the size of the form. (Inherited from Form .)
	SizeGripStyle	Gets or sets the style of the size grip to display in the lower-right corner of the form. (Inherited from Form .)
	StartPosition	Gets or sets the starting position of the form at run time. (Inherited from Form .)
	TabIndex	Gets or sets the tab order of the control within its container. (Inherited from Form .)

 TabStop	Gets or sets a value indicating whether the user can give the focus to this control using the TAB key. (Inherited from Form .)
 Tag	Gets or sets the object that contains data about the control. (Inherited from Control .)
 Text	(Inherited from Form .)
 Top	Gets or sets the distance, in pixels, between the top edge of the control and the top edge of its container's client area. (Inherited from Control .)
 TopLevel	Gets or sets a value indicating whether to display the form as a top-level window. (Inherited from Form .)
 TopLevelControl	Gets the parent control that is not parented by another Windows Forms control. Typically, this is the outermost Form that the control is contained in. (Inherited from Control .)
 TopMost	Gets or sets a value indicating whether the form should be displayed as a topmost form. (Inherited from Form .)
 TransparencyKey	Gets or sets the color that will represent transparent areas of the form. (Inherited from Form .)
 UseWaitCursor	Gets or sets a value indicating whether to use the wait cursor for the current control and all child controls. (Inherited from Control .)
 VerticalScroll	Gets the characteristics associated with the vertical scroll bar. (Inherited from ScrollableControl .)
 Visible	Gets or sets a value indicating whether the control and all its child controls are displayed. (Inherited from Control .)
 VScroll	Gets or sets a value indicating whether the vertical scroll bar is visible. (Inherited from ScrollableControl .)
 Width	Gets or sets the width of the control. (Inherited from Control .)
 WindowState	Gets or sets a value that indicates whether form is minimized, maximized, or normal. (Inherited from Form .)
 WindowTarget	This property is not relevant for this class. (Inherited from Control .)

See Also

[ProgressDialog Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)













ProgressDialog.ProgressDialog Events





















The [ProgressDialog](#) type exposes the following members.







Events

Name	Description
 Activated	Occurs when the form is activated in code or by the user. (Inherited from Form .)
 AutoSizeChanged	Occurs when the AutoSize property changes. (Inherited from Form .)
 AutoValidateChanged	Occurs when the AutoValidate property changes. (Inherited from Form .)
 BackColorChanged	Occurs when the value of the BackColor property changes. (Inherited from Control .)
 BackgroundImageChanged	Occurs when the value of the BackgroundImage property changes. (Inherited from Control .)
 BackgroundImageLayoutChanged	Occurs when the BackgroundImageLayout property changes. (Inherited from Control .)
 BindingContextChanged	Occurs when the value of the BindingContext property changes. (Inherited from Control .)
 CausesValidationChanged	Occurs when the value of the CausesValidation property changes. (Inherited from Control .)
 ChangeUICues	Occurs when the focus or keyboard user interface (UI) cues change. (Inherited from Control .)
 Click	Occurs when the control is clicked. (Inherited from Control .)
 ClientSizeChanged	Occurs when the value of the ClientSize property changes. (Inherited from Control .)
 Closed	Occurs when the form is closed. (Inherited from Form .)
 Closing	Occurs when the form is closing. (Inherited from Form .)
 ContextMenuChanged	Occurs when the value of the ContextMenu property changes. (Inherited from Control .)
 ContextMenuStripChanged	Occurs when the value of the ContextMenuStrip property changes. (Inherited from Control .)
 ControlAdded	Occurs when a new control is added to the Control.ControlCollection . (Inherited from Control .)
 ControlRemoved	Occurs when a control is removed from the Control.ControlCollection . (Inherited from Control .)
 CursorChanged	Occurs when the value of the Cursor property changes. (Inherited from Control .)
 Deactivate	Occurs when the form loses focus and is no longer the active form. (Inherited from Form .)

 Disposed	Occurs when the component is disposed by a call to the Dispose() method. (Inherited from Component .)
 DockChanged	Occurs when the value of the Dock property changes. (Inherited from Control .)
 DoubleClick	Occurs when the control is double-clicked. (Inherited from Control .)
 DragDrop	Occurs when a drag-and-drop operation is completed. (Inherited from Control .)
 DragEnter	Occurs when an object is dragged into the control's bounds. (Inherited from Control .)
 DragLeave	Occurs when an object is dragged out of the control's bounds. (Inherited from Control .)
 DragOver	Occurs when an object is dragged over the control's bounds. (Inherited from Control .)
 EnabledChanged	Occurs when the Enabled property value has changed. (Inherited from Control .)
 Enter	Occurs when the control is entered. (Inherited from Control .)
 FontChanged	Occurs when the Font property value changes. (Inherited from Control .)
 ForeColorChanged	Occurs when the ForeColor property value changes. (Inherited from Control .)
 FormClosed	Occurs after the form is closed. (Inherited from Form .)
 FormClosing	Occurs before the form is closed. (Inherited from Form .)
 GiveFeedback	Occurs during a drag operation. (Inherited from Control .)
 GotFocus	Occurs when the control receives focus. (Inherited from Control .)
 HandleCreated	Occurs when a handle is created for the control. (Inherited from Control .)
 HandleDestroyed	Occurs when the control's handle is in the process of being destroyed. (Inherited from Control .)
 HelpButtonClicked	Occurs when the Help button is clicked. (Inherited from Form .)
 HelpRequested	Occurs when the user requests help for a control. (Inherited from Control .)
 ImeModeChanged	Occurs when the ImeMode property has changed. (Inherited from Control .)
 InputLanguageChanged	Occurs after the input language of the form has changed. (Inherited from Form .)
 InputLanguageChanging	Occurs when the user attempts to change the input language for the form. (Inherited from Form .)
 Invalidated	Occurs when a control's display requires redrawing. (Inherited from

	Control.)
 KeyDown	Occurs when a key is pressed while the control has focus. (Inherited from Control.)
 KeyPress	Occurs when a key is pressed while the control has focus. (Inherited from Control.)
 KeyUp	Occurs when a key is released while the control has focus. (Inherited from Control.)
 Layout	Occurs when a control should reposition its child controls. (Inherited from Control.)
 Leave	Occurs when the input focus leaves the control. (Inherited from Control.)
 Load	Occurs before a form is displayed for the first time. (Inherited from Form.)
 LocationChanged	Occurs when the Location property value has changed. (Inherited from Control.)
 LostFocus	Occurs when the control loses focus. (Inherited from Control.)
 MarginChanged	Occurs when the Margin property changes. (Inherited from Form.)
 MaximizedBoundsChanged	Occurs when the value of the MaximizedBounds property has changed. (Inherited from Form.)
 MaximumSizeChanged	Occurs when the value of the MaximumSize property has changed. (Inherited from Form.)
 MdiChildActivate	Occurs when a multiple-document interface (MDI) child form is activated or closed within an MDI application. (Inherited from Form.)
 MenuComplete	Occurs when the menu of a form loses focus. (Inherited from Form.)
 MenuStart	Occurs when the menu of a form receives focus. (Inherited from Form.)
 MinimumSizeChanged	Occurs when the value of the MinimumSize property has changed. (Inherited from Form.)
 MouseCaptureChanged	Occurs when the control loses mouse capture. (Inherited from Control.)
 MouseClicked	Occurs when the control is clicked by the mouse. (Inherited from Control.)
 MouseDoubleClick	Occurs when the control is double clicked by the mouse. (Inherited from Control.)
 MouseDown	Occurs when the mouse pointer is over the control and a mouse button is pressed. (Inherited from Control.)
 MouseEnter	Occurs when the mouse pointer enters the control. (Inherited from Control.)

 MouseHover	Occurs when the mouse pointer rests on the control. (Inherited from Control .)
 MouseLeave	Occurs when the mouse pointer leaves the control. (Inherited from Control .)
 MouseMove	Occurs when the mouse pointer is moved over the control. (Inherited from Control .)
 MouseUp	Occurs when the mouse pointer is over the control and a mouse button is released. (Inherited from Control .)
 MouseWheel	Occurs when the mouse wheel moves while the control has focus. (Inherited from Control .)
 Move	Occurs when the control is moved. (Inherited from Control .)
 PaddingChanged	Occurs when the control's padding changes. (Inherited from Control .)
 Paint	Occurs when the control is redrawn. (Inherited from Control .)
 ParentChanged	Occurs when the Parent property value changes. (Inherited from Control .)
 PreviewKeyDown	Occurs before the KeyDown event when a key is pressed while focus is on this control. (Inherited from Control .)
 QueryAccessibilityHelp	Occurs when AccessibleObject is providing help to accessibility applications. (Inherited from Control .)
 QueryContinueDrag	Occurs during a drag-and-drop operation and enables the drag source to determine whether the drag-and-drop operation should be canceled. (Inherited from Control .)
 RegionChanged	Occurs when the value of the Region property changes. (Inherited from Control .)
 Resize	Occurs when the control is resized. (Inherited from Control .)
 ResizeBegin	Occurs when a form enters resizing mode. (Inherited from Form .)
 ResizeEnd	Occurs when a form exits resizing mode. (Inherited from Form .)
 RightToLeftChanged	Occurs when the RightToLeft property value changes. (Inherited from Control .)
 RightToLeftLayoutChanged	Occurs after the value of the RightToLeftLayout property changes. (Inherited from Form .)
 Scroll	Occurs when the user or code scrolls through the client area. (Inherited from ScrollableControl .)
 Shown	Occurs whenever the form is first displayed. (Inherited from Form .)
 SizeChanged	Occurs when the Size property value changes. (Inherited from Control .)
 StyleChanged	Occurs when the control style changes. (Inherited from Control .)
 SystemColorsChanged	Occurs when the system colors change. (Inherited from Control .)

 TabIndexChanged	Occurs when the value of the TabIndex property changes. (Inherited from Form .)
 TabStopChanged	Occurs when the TabStop property changes. (Inherited from Form .)
 TextChanged	Occurs when the Text property value changes. (Inherited from Control .)
 Validated	Occurs when the control is finished validating. (Inherited from Control .)
 Validating	Occurs when the control is validating. (Inherited from Control .)
 VisibleChanged	Occurs when the Visible property value changes. (Inherited from Control .)

See Also

[ProgressDialog Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController Class

A controller for the [WoodstocksIMS](#).

Inheritance Hierarchy

[System.Object](#)

Woodstocks.WoodstocksIMS.Presentation.WoodstocksIMSController

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class WoodstocksIMSController : IWoodstocksIMSController
```

VB

```
Public Class WoodstocksIMSController
    Implements IWoodstocksIMSController
```

C++


```
public ref class WoodstocksIMSController : IWoodstocksIMSController
```

F#





```
type WoodstocksIMSController =
    class
        interface IWoodstocksIMSController
    end
```














The **WoodstocksIMSController** type exposes the following members.

Constructors




	Name	Description
	WoodstocksIMSController	Initialises a WoodstocksIMSController .

Methods

	Name	Description
	CancelAsync	Informs the WoodstocksIMS to cancel an asynchronous operation.
	DiscardImportedToyData	Causes the WoodstocksIMS to discard toy data that has been imported into the system.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	ExportToysAsync	Requests the WoodstocksIMS to export Toy data asynchronously into the

		system.
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetCancellationStatus	Gets the cancellation status of an operation.
	GetErrorStatus	Gets the error reported by the WoodstocksIMS whne an error occurs.
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetToys	Gets the toy data from the WoodstocksIMS.
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ImportToysAsync	Requests the WoodstocksIMS to import Toy data asynchronously into the system.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ResetCancellationStatus	Resets the cancellation status of the cotnroller.
	ResetErrorStatus	Resets the error status reported by the controller
	SetView	Set the controller's view.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	UnsavedChanges	Gets whether the IWoodstocksIMS has imported data that has not been saved.

Events

	Name	Description
	ExportCompleted	Raised by the controller when an asynchronous exportation has completed.
	ImportCompleted	Raised by the controller when an asynchronous importation has completed.
	ProgressChanged	Raised by the controller when progress on an asynchronous operation.

See Also

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController Constructor

Initialises a [WoodstocksIMSController](#).

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public WoodstocksIMSController()
```

VB

```
Public Sub New
```

C++

```
public:  
WoodstocksIMSController()
```

F#

```
new : unit -> WoodstocksIMSController
```

See Also


















[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.WoodstocksIMSController Methods

The [WoodstocksIMSController](#) type exposes the following members.

Methods

	Name	Description
	CancelAsync	Informs the WoodstocksIMS to cancel an asynchronous operation.
	DiscardImportedToyData	Causes the WoodstocksIMS to discard toy data that has been imported into the system.
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	ExportToysAsync	Requests the WoodstocksIMS to export Toy data asynchronously into the system.
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetCancellationStatus	Gets the cancellation status of an operation.
	GetErrorStatus	Gets the error reported by the WoodstocksIMS whne an error occurs.
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetToys	Gets the toy data from the WoodstocksIMS.
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ImportToysAsync	Requests the WoodstocksIMS to import Toy data asynchronously into the system.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ResetCancellationStatus	Resets the cancellation status of the cotnroller.
	ResetErrorStatus	Resets the error status reported by the controller
	SetView	Set the controller's view.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	UnsavedChanges	Gets whether the IWoodstocksIMS has imported data that has not been saved.

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.CancelAsync Method

Informs the [WoodstocksIMS](#) to cancel an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void CancelAsync()
```

VB

```
Public Sub CancelAsync
```

C++

```
public:  
virtual void CancelAsync() sealed
```

F#

```
abstract CancelAsync : unit -> unit  
override CancelAsync : unit -> unit
```

Implements

[IWoodstocksIMSController.CancelAsync\(\)](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.DiscardImportedToyData Method

Causes the [WoodstocksIMS](#) to discard toy data that has been imported into the system.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void DiscardImportedToyData()
```

VB

```
Public Sub DiscardImportedToyData
```

C++

```
public:  
virtual void DiscardImportedToyData() sealed
```

F#

```
abstract DiscardImportedToyData : unit -> unit  
override DiscardImportedToyData : unit -> unit
```

Implements

[IWoodstocksIMSController.DiscardImportedToyData\(\)](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.ExportToysAsync Method

Requests the [WoodstocksIMS](#) to export [Toy](#) data asynchronously into the system.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ExportToysAsync ( )
```

VB

```
Public Sub ExportToysAsync
```

C++

```
public:  
virtual void ExportToysAsync ( ) sealed
```

F#

```
abstract ExportToysAsync : unit -> unit  
override ExportToysAsync : unit -> unit
```

Implements

[IWoodstocksIMSController.ExportToysAsync\(\)](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.GetCancellationStatus Method

Gets the cancellation status of an operation.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool GetCancellationStatus()
```

VB

```
Public Function GetCancellationStatus As Boolean
```

C++

```
public:  
virtual bool GetCancellationStatus() sealed
```

F#

```
abstract GetCancellationStatus : unit -> bool  
override GetCancellationStatus : unit -> bool
```

Return Value

Type: [Boolean](#)

True if an operation has been cancelled, otherwise false.

Implements

[IWoodstocksIMSController.GetCancellationStatus\(\)](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.GetErrorStatus Method

Gets the error reported by the WoodstocksIMS whne an error occurs.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public Exception GetErrorStatus()
```

VB

```
Public Function GetErrorStatus As Exception
```

C++

```
public:  
virtual Exception^ GetErrorStatus() sealed
```

F#

```
abstract GetErrorStatus : unit -> Exception  
override GetErrorStatus : unit -> Exception
```

Return Value

Type: [Exception](#)

The [Exception](#) that reports the error.

Implements

[IWoodstocksIMSController.GetErrorStatus\(\)](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.GetToys Method

Gets the toy data from the WoodstocksIMS.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public IToys GetToys()
```

VB

```
Public Function GetToys As IToys
```

C++

```
public:  
virtual IToys^ GetToys() sealed
```

F#

```
abstract GetToys : unit -> IToys  
override GetToys : unit -> IToys
```

Return Value

Type: [IToys](#)

The imported toy data.

Implements

[IWoodstocksIMSController.GetToys\(\)](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.ImportToysAsync Method

Requests the [WoodstocksIMS](#) to import [Toy](#) data asynchronously into the system.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ImportToysAsync(  
    string source  
)
```

VB

```
Public Sub ImportToysAsync (  
    source As String  
)
```

C++

```
public:  
virtual void ImportToysAsync(  
    String^ source  
) sealed
```

F#

```
abstract ImportToysAsync :  
    source : string -> unit  
override ImportToysAsync :  
    source : string -> unit
```

Parameters

source

Type: [System.String](#)

The source from which [Toy](#) data should be imported.

Implements

[IWoodstocksIMSController.ImportToysAsync\(String\)](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.ResetCancellationStatus Method

Resets the cancellation status of the cotnroller.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ResetCancellationStatus()
```

VB

```
Public Sub ResetCancellationStatus
```

C++

```
public:  
virtual void ResetCancellationStatus() sealed
```

F#

```
abstract ResetCancellationStatus : unit -> unit  
override ResetCancellationStatus : unit -> unit
```

Implements

[IWoodstocksIMSController.ResetCancellationStatus\(\)](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.ResetErrorStatus Method

Resets the error status reported by the controller

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void ResetErrorStatus()
```

VB

```
Public Sub ResetErrorStatus
```

C++

```
public:  
virtual void ResetErrorStatus() sealed
```

F#

```
abstract ResetErrorStatus : unit -> unit  
override ResetErrorStatus : unit -> unit
```

Implements

[IWoodstocksIMSController.ResetErrorStatus\(\)](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.SetView Method

Set the controller's view.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void SetView(  
    IWoodstocksIMSView view  
)
```

VB

```
Public Sub SetView (  
    view As IWoodstocksIMSView  
)
```

C++

```
public:  
virtual void SetView(  
    IWoodstocksIMSView^ view  
) sealed
```

F#

```
abstract SetView :  
    view : IWoodstocksIMSView -> unit  
override SetView :  
    view : IWoodstocksIMSView -> unit
```

Parameters

view

Type: [Woodstocks.WoodstocksIMS.Presentation.IWoodstocksIMSView](#)

Implements

[IWoodstocksIMSController.SetView\(IWoodstocksIMSView\)](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.UnsavedChanges Method

Gets whether the [IWoodstocksIMS](#) has imported data that has not been saved.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public bool UnsavedChanges()
```

VB

```
Public Function UnsavedChanges As Boolean
```

C++

```
public:  
virtual bool UnsavedChanges() sealed
```

F#

```
abstract UnsavedChanges : unit -> bool  
override UnsavedChanges : unit -> bool
```

Return Value

Type: [Boolean](#)

True if the system has modified data that has not been saved, otherwise false.

Implements

[IWoodstocksIMSController.UnsavedChanges\(\)](#)

See Also




[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstockslMSController.WoodstockslMSController Events

The [WoodstockslMSController](#) type exposes the following members.

Events

	Name	Description
	ExportCompleted	Raised by the controller when an asynchronous exportation has completed.
	ImportCompleted	Raised by the controller when an asynchronous importation has completed.
	ProgressChanged	Raised by the controller when progress on an asynchronous operation.

See Also

[WoodstockslMSController Class](#)

[Woodstocks.WoodstockslMS.Presentation Namespace](#)

WoodstocksIMSController.ExportCompleted Event

Raised by the controller when an asynchronous exportation has completed.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event AsyncCompletedEventHandler ExportCompleted
```

VB

```
Public Event ExportCompleted As AsyncCompletedEventHandler
```

C++

```
public:  
virtual event AsyncCompletedEventHandler^ ExportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>  
override ExportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

Implements

[IWoodstocksIMSController.ExportCompleted](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.ImportCompleted Event

Raised by the controller when an asynchronous importation has completed.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event AsyncCompletedEventHandler ImportCompleted
```

VB

```
Public Event ImportCompleted As AsyncCompletedEventHandler
```

C++

```
public:  
virtual event AsyncCompletedEventHandler^ ImportCompleted {  
    void add (AsyncCompletedEventHandler^ value);  
    void remove (AsyncCompletedEventHandler^ value);  
}
```

F#

```
abstract ImportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>  
override ImportCompleted : IEvent<AsyncCompletedEventHandler,  
    AsyncCompletedEventArgs>
```

Value

Type: [System.ComponentModel.AsyncCompletedEventHandler](#)

Implements

[IWoodstocksIMSController.ImportCompleted](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSController.ProgressChanged Event

Raised by the controller when progress on an asynchronous operation.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public event ProgressChangedEventHandler ProgressChanged
```

VB

```
Public Event ProgressChanged As ProgressChangedEventHandler
```

C++

```
public:  
virtual event ProgressChangedEventHandler^ ProgressChanged {  
    void add (ProgressChangedEventHandler^ value);  
    void remove (ProgressChangedEventHandler^ value);  
}
```

F#

```
abstract ProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>  
override ProgressChanged : IEvent<ProgressChangedEventHandler,  
    ProgressChangedEventArgs>
```

Value

Type: [System.ComponentModel.ProgressChangedEventHandler](#)

Implements

[IWoodstocksIMSController.ProgressChanged](#)

See Also

[WoodstocksIMSController Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm Class

A view for the [WoodstocksIMS](#).

Inheritance Hierarchy

[System.Object](#)

[System.MarshalByRefObject](#)

[System.ComponentModel.Component](#)

[System.Windows.Forms.Control](#)

[System.Windows.Forms.ScrollableControl](#)

[System.Windows.Forms.ContainerControl](#)

[System.Windows.Forms.Form](#)

Woodstocks.WoodstocksIMS.Presentation.WoodstocksIMSForm

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class WoodstocksIMSForm : Form,
    IWoodstocksIMSView
```

VB

```
Public Class WoodstocksIMSForm
    Inherits Form
    Implements IWoodstocksIMSView
```

C++

```
public ref class WoodstocksIMSForm : public Form,
    IWoodstocksIMSView
```

F#

















```
type WoodstocksIMSForm =
    class
        inherit Form
        interface IWoodstocksIMSView
    end
```




















The **WoodstocksIMSForm** type exposes the following members.


















Constructors














	Name	Description
	WoodstocksIMSForm	Initialises a FormView for the WoodstocksIMS .

















Methods






















	Name	Description
	AccessibilityNotifyClients(AccessibleEvents, Int32)	Notifies the accessibility client applications of the specified AccessibleEvents for the specified child control. (Inherited from Control .)
	AccessibilityNotifyClients(AccessibleEvents, Int32, Int32)	Notifies the accessibility client applications of the specified AccessibleEvents for the specified child control . (Inherited from Control .)
	Activate	Activates the form and gives it focus. (Inherited from Form .)
	ActivateMdiChild	Activates the MDI child of a form. (Inherited from Form .)
	AddOwnedForm	Adds an owned form to this form. (Inherited from Form .)
	AdjustFormScrollbars	Adjusts the scroll bars on the container based on the current control positions and the control currently selected. (Inherited from Form .)
	ApplyAutoScaling	Obsolete. Resizes the form according to the current value of the AutoScaleBaseSize property and the size of the current font. (Inherited from Form .)
	BeginInvoke(Delegate)	Executes the specified delegate asynchronously on the thread that the control's underlying handle was created on. (Inherited from Control .)
	BeginInvoke(Delegate, Object[])	Executes the specified delegate asynchronously with the specified arguments, on the thread that the control's underlying handle was created on. (Inherited from Control .)
	BringToFront	Brings the control to the front of the z-order. (Inherited from Control .)
	CenterToParent	Centers the position of the form within the bounds of the parent form. (Inherited from Form .)
	CenterToScreen	Centers the form on the current screen. (Inherited from Form .)
	Close	Closes the form. (Inherited from Form .)
	Contains	Retrieves a value indicating whether the specified control is a child of the control. (Inherited from Control .)
	CreateAccessibilityInstance	Creates a new accessibility object for the control. (Inherited from Control .)
	CreateControl	Forces the creation of the visible control, including the





























		creation of the handle and any visible child controls. (Inherited from Control .)
	CreateControlsInstance	(Inherited from Form .)
	CreateGraphics	Creates the Graphics for the control. (Inherited from Control .)
	CreateHandle	Creates the handle for the form. If a derived class overrides this function, it must call the base implementation. (Inherited from Form .)
	CreateObjRef	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object. (Inherited from MarshalByRefObject .)
	DefWndProc	(Inherited from Form .)
	DestroyHandle	Destroys the handle associated with the control. (Inherited from Control .)
	DisableImport	Disables import option of the View.
	DisplayStatus	Updates the status being displayed by the Form to the user.
	DisplayToys	Causes the form to display toy data in the DataGridView of the Form used to display toy data.
	DisplayUnsavedDataDialog	Displays a message box to the user informing them that imported data that has been modified has not been saved and asks user if they would like to save changes.
	Dispose()	Releases all resources used by the Component . (Inherited from Component .)
	Dispose(Boolean)	Clean up any resources being used. (Overrides Form.Dispose(Boolean) .)
	DoDragDrop	Begins a drag-and-drop operation. (Inherited from Control .)
	DrawToBitmap	Supports rendering to the specified bitmap. (Inherited from Control .)
	EnableImport	Enables import option of the View.
	EndInvoke	Retrieves the return value of the asynchronous operation represented by the IAsyncResult passed. (Inherited from Control .)
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)
	ExportToys	Exports toy data
	Finalize	Releases unmanaged resources and performs other

		cleanup operations before the Component is reclaimed by garbage collection. (Inherited from Component .)
	FindForm	Retrieves the form that the control is on. (Inherited from Control .)
	Focus	Sets input focus to the control. (Inherited from Control .)
	GetAccessibilityObjectById	Retrieves the specified AccessibleObject . (Inherited from Control .)
	GetAutoSizeMode	Retrieves a value indicating how a control will behave when its AutoSize property is enabled. (Inherited from Control .)
	GetChildAtPoint(Point)	Retrieves the child control that is located at the specified coordinates. (Inherited from Control .)
	GetChildAtPoint(Point, GetChildAtPointSkip)	Retrieves the child control that is located at the specified coordinates, specifying whether to ignore child controls of a certain type. (Inherited from Control .)
	GetContainerControl	Returns the next ContainerControl up the control's chain of parent controls. (Inherited from Control .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetLifetimeService	Retrieves the current lifetime service object that controls the lifetime policy for this instance. (Inherited from MarshalByRefObject .)
	GetNextControl	Retrieves the next control forward or back in the tab order of child controls. (Inherited from Control .)
	GetPreferredSize	Retrieves the size of a rectangular area into which a control can be fitted. (Inherited from Control .)
	GetScaledBounds	(Inherited from Form .)
	GetScrollState	Determines whether the specified flag has been set. (Inherited from ScrollableControl .)
	GetService	Returns an object that represents a service provided by the Component or by its Container . (Inherited from Component .)
	GetStatus	Gets the status that is displayed by the Form to the user.
	GetStyle	Retrieves the value of the specified control style bit for the control. (Inherited from Control .)
	GetTopLevel	Determines if the control is a top-level control. (Inherited from Control .)




















	GetType	Gets the Type of the current instance. (Inherited from Object .)
	Hide	Conceals the control from the user. (Inherited from Control .)
	ImportToys	Imports data into the WoodstocksIMS .
	InitializeLifetimeService	Obtains a lifetime service object to control the lifetime policy for this instance. (Inherited from MarshalByRefObject .)
	InitLayout	Called after the control has been added to another container. (Inherited from Control .)
	Invalidate()	Invalidates the entire surface of the control and causes the control to be redrawn. (Inherited from Control .)
	Invalidate(Region)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. (Inherited from Control .)
	Invalidate(Boolean)	Invalidates a specific region of the control and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control .)
	Invalidate(Rectangle)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. (Inherited from Control .)
	Invalidate(Region, Boolean)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control .)
	Invalidate(Rectangle, Boolean)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control .)
	Invoke(Delegate)	Executes the specified delegate on the thread that owns the control's underlying window handle. (Inherited from Control .)
	Invoke(Delegate, Object[])	Executes the specified delegate, on the thread that























		owns the control's underlying window handle, with the specified list of arguments. (Inherited from Control .)
	InvokeGotFocus	Raises the GotFocus event for the specified control. (Inherited from Control .)
	InvokeLostFocus	Raises the LostFocus event for the specified control. (Inherited from Control .)
	InvokeOnClick	Raises the Click event for the specified control. (Inherited from Control .)
	InvokePaint	Raises the Paint event for the specified control. (Inherited from Control .)
	InvokePaintBackground	Raises the PaintBackground event for the specified control. (Inherited from Control .)
	IsInputChar	Determines if a character is an input character that the control recognizes. (Inherited from Control .)
	IsInputKey	Determines whether the specified key is a regular input key or a special key that requires preprocessing. (Inherited from Control .)
	LayoutMdi	Arranges the multiple-document interface (MDI) child forms within the MDI parent form. (Inherited from Form .)
	MemberwiseClone()	Creates a shallow copy of the current Object . (Inherited from Object .)
	MemberwiseClone(Boolean)	Creates a shallow copy of the current MarshalByRefObject object. (Inherited from MarshalByRefObject .)
	NotifyDirectoryNotFound	Notifies the user that a DirectoryNotFoundException that has occurred.
	NotifyDuplicateToyException	Notifies the user that an duplicate toy has been added to the collection of toys.
	NotifyError	Notifies the user that an error has occurred.
	NotifyExportCancellation	Notifies the user that exportation has been cancelled.
	NotifyExportCompletion()	Notifies the user that exportation has completed.
	NotifyExportCompletion(Exception, Boolean)	Notifies the user that exportation has completed.
	NotifyFileNotFound	Notifies the user that a FileNotFoundException has occurred.
	NotifyImportCancellation	Notifies the user that importation has been cancelled.
	NotifyImportCompletion	Notifies the user that importation has completed.
	NotifyInvalidate	Raises the Invalidated event with a specified region of the control to invalidate. (Inherited from Control .)





















	NotifyUnknownException	Notifies the user that an unknown problem has occurred.
	OnActivated	Raises the Activated event. (Inherited from Form .)
	OnAutoSizeChanged	Raises the AutoSizeChanged event. (Inherited from Control .)
	OnAutoValidateChanged	Raises the AutoValidateChanged event. (Inherited from ContainerControl .)
	OnBackColorChanged	Raises the BackColorChanged event. (Inherited from Control .)
	OnBackgroundImageChanged	Raises the BackgroundImageChanged event. (Inherited from Form .)
	OnBackgroundImageLayoutChanged	Raises the BackgroundImageLayoutChanged event. (Inherited from Form .)
	OnBindingContextChanged	Raises the BindingContextChanged event. (Inherited from Control .)
	OnCausesValidationChanged	Raises the CausesValidationChanged event. (Inherited from Control .)
	OnChangeUICues	Raises the ChangeUICues event. (Inherited from Control .)
	OnClick	Raises the Click event. (Inherited from Control .)
	OnClientSizeChanged	Raises the ClientSizeChanged event. (Inherited from Control .)
	OnClosed	Raises the Closed event. (Inherited from Form .)
	OnClosing	Raises the Closing event. (Inherited from Form .)
	OnContextMenuChanged	Raises the ContextMenuChanged event. (Inherited from Control .)
	OnContextMenuStripChanged	Raises the ContextMenuStripChanged event. (Inherited from Control .)
	OnControlAdded	Raises the ControlAdded event. (Inherited from Control .)
	OnControlRemoved	Raises the ControlRemoved event. (Inherited from Control .)
	OnCreateControl	Raises the CreateControl event. (Inherited from Form .)
	OnCursorChanged	Raises the CursorChanged event. (Inherited from Control .)
	OnDeactivate	Raises the Deactivate event. (Inherited from Form .)
	OnDockChanged	Raises the DockChanged event. (Inherited from Control .)


















	OnDoubleClick	Raises the DoubleClick event. (Inherited from Control .)
	OnDragDrop	Raises the DragDrop event. (Inherited from Control .)
	OnDragEnter	Raises the DragEnter event. (Inherited from Control .)
	OnDragLeave	Raises the DragLeave event. (Inherited from Control .)
	OnDragOver	Raises the DragOver event. (Inherited from Control .)
	OnEnabledChanged	(Inherited from Form .)
	OnEnter	Raises the Enter event. (Inherited from Form .)
	OnFontChanged	(Inherited from Form .)
	OnForeColorChanged	Raises the ForeColorChanged event. (Inherited from Control .)
	OnFormClosed	Raises the FormClosed event. (Inherited from Form .)
	OnFormClosing	Raises the FormClosing event. (Inherited from Form .)
	OnGiveFeedback	Raises the GiveFeedback event. (Inherited from Control .)
	OnGotFocus	Raises the GotFocus event. (Inherited from Control .)
	OnHandleCreated	(Inherited from Form .)
	OnHandleDestroyed	(Inherited from Form .)
	OnHelpButtonClicked	Raises the HelpButtonClicked event. (Inherited from Form .)
	OnHelpRequested	Raises the HelpRequested event. (Inherited from Control .)
	OnImeModeChanged	Raises the ImeModeChanged event. (Inherited from Control .)
	OnInputLanguageChanged	Raises the InputLanguageChanged event. (Inherited from Form .)
	OnInputLanguageChanging	Raises the InputLanguageChanging event. (Inherited from Form .)
	OnInvalidated	Raises the Invalidated event. (Inherited from Control .)
	OnKeyDown	Raises the KeyDown event. (Inherited from Control .)
	OnKeyPress	Raises the KeyPress event. (Inherited from Control .)
	OnKeyUp	Raises the KeyUp event. (Inherited from Control .)
	OnLayout	Raises the Layout event. (Inherited from Form .)
	OnLeave	Raises the Leave event. (Inherited from Control .)
	OnLoad	Raises the Load event. (Inherited from Form .)
	OnLocationChanged	Raises the LocationChanged event. (Inherited from Control .)





















	OnLostFocus	Raises the LostFocus event. (Inherited from Control .)
	OnMarginChanged	Raises the MarginChanged event. (Inherited from Control .)
	OnMaximizedBoundsChanged	Raises the MaximizedBoundsChanged event. (Inherited from Form .)
	OnMaximumSizeChanged	Raises the MaximumSizeChanged event. (Inherited from Form .)
	OnMdiChildActivate	Raises the MdiChildActivate event. (Inherited from Form .)
	OnMenuComplete	Raises the MenuComplete event. (Inherited from Form .)
	OnMenuStart	Raises the MenuStart event. (Inherited from Form .)
	OnMinimumSizeChanged	Raises the MinimumSizeChanged event. (Inherited from Form .)
	OnMouseCaptureChanged	Raises the MouseCaptureChanged event. (Inherited from Control .)
	OnMouseClicked	Raises the MouseClicked event. (Inherited from Control .)
	OnMouseDoubleClick	Raises the MouseDoubleClick event. (Inherited from Control .)
	OnMouseDown	Raises the MouseDown event. (Inherited from Control .)
	OnMouseEnter	Raises the MouseEnter event. (Inherited from Control .)
	OnMouseHover	Raises the MouseHover event. (Inherited from Control .)
	OnMouseLeave	Raises the MouseLeave event. (Inherited from Control .)
	OnMouseMove	Raises the MouseMove event. (Inherited from Control .)
	OnMouseUp	Raises the MouseUp event. (Inherited from Control .)
	OnMouseWheel	Raises the MouseWheel event. (Inherited from ScrollableControl .)
	OnMove	Raises the Move event. (Inherited from Control .)
	OnNotifyMessage	Notifies the control of Windows messages. (Inherited from Control .)
	OnPaddingChanged	Raises the PaddingChanged event. (Inherited from ScrollableControl .)
	OnPaint	(Inherited from Form .)
	OnPaintBackground	Paints the background of the control. (Inherited from


















		ScrollableControl.)
	OnParentBackColorChanged	Raises the BackColorChanged event when the BackColor property value of the control's container changes. (Inherited from Control.)
	OnParentBackgroundImageChanged	Raises the BackgroundImageChanged event when the BackgroundImage property value of the control's container changes. (Inherited from Control.)
	OnParentBindingContextChanged	Raises the BindingContextChanged event when the BindingContext property value of the control's container changes. (Inherited from Control.)
	OnParentChanged	(Inherited from ContainerControl.)
	OnParentCursorChanged	Raises the CursorChanged event. (Inherited from Control.)
	OnParentEnabledChanged	Raises the EnabledChanged event when the Enabled property value of the control's container changes. (Inherited from Control.)
	OnParentFontChanged	Raises the FontChanged event when the Font property value of the control's container changes. (Inherited from Control.)
	OnParentForeColorChanged	Raises the ForeColorChanged event when the ForeColor property value of the control's container changes. (Inherited from Control.)
	OnParentRightToLeftChanged	Raises the RightToLeftChanged event when the RightToLeft property value of the control's container changes. (Inherited from Control.)
	OnParentVisibleChanged	Raises the VisibleChanged event when the Visible property value of the control's container changes. (Inherited from Control.)
	OnPreviewKeyDown	Raises the PreviewKeyDown event. (Inherited from Control.)
	OnPrint	Raises the Paint event. (Inherited from Control.)
	OnQueryContinueDrag	Raises the QueryContinueDrag event. (Inherited from Control.)
	OnRegionChanged	Raises the RegionChanged event. (Inherited from Control.)
	OnResize	(Inherited from Form.)
	OnResizeBegin	Raises the ResizeBegin event. (Inherited from Form.)
	OnResizeEnd	Raises the ResizeEnd event. (Inherited from Form.)
	OnRightToLeftChanged	(Inherited from ScrollableControl.)
	OnRightToLeftLayoutChanged	Raises the RightToLeftLayoutChanged event.

		(Inherited from Form .)
	OnScroll	Raises the Scroll event. (Inherited from ScrollableControl .)
	OnShown	Raises the Shown event. (Inherited from Form .)
	OnSizeChanged	Raises the SizeChanged event. (Inherited from Control .)
	OnStyleChanged	(Inherited from Form .)
	OnSystemColorsChanged	Raises the SystemColorsChanged event. (Inherited from Control .)
	OnTabIndexChanged	Raises the TabIndexChanged event. (Inherited from Control .)
	OnTabStopChanged	Raises the TabStopChanged event. (Inherited from Control .)
	OnTextChanged	(Inherited from Form .)
	OnValidated	Raises the Validated event. (Inherited from Control .)
	OnValidating	Raises the Validating event. (Inherited from Control .)
	OnVisibleChanged	Raises the VisibleChanged event. (Inherited from Form .)
	PerformAutoScale	Performs scaling of the container control and its children. (Inherited from ContainerControl .)
	PerformLayout()	Forces the control to apply layout logic to all its child controls. (Inherited from Control .)
	PerformLayout(Control, String)	Forces the control to apply layout logic to all its child controls. (Inherited from Control .)
	PointToClient	Computes the location of the specified screen point into client coordinates. (Inherited from Control .)
	PointToScreen	Computes the location of the specified client point into screen coordinates. (Inherited from Control .)
	PreProcessControlMessage	Preprocesses keyboard or input messages within the message loop before they are dispatched. (Inherited from Control .)
	PreProcessMessage	Preprocesses keyboard or input messages within the message loop before they are dispatched. (Inherited from Control .)
	ProcessCmdKey	Processes a command key. (Inherited from Form .)
	ProcessDialogChar	Processes a dialog character. (Inherited from Form .)
	ProcessDialogKey	Processes a dialog box key. (Inherited from Form .)
	ProcessKeyEventArgs	Processes a key message and generates the appropriate control events. (Inherited from Control .)


	ProcessKeyMessage	Processes a keyboard message. (Inherited from Control .)
	ProcessKeyPreview	(Inherited from Form .)
	ProcessMnemonic	Processes a mnemonic character. (Inherited from Form .)
	ProcessTabKey	(Inherited from Form .)
	RaiseDragEvent	Raises the appropriate drag event. (Inherited from Control .)
	RaiseKeyEvent	Raises the appropriate key event. (Inherited from Control .)
	RaiseMouseEvent	Raises the appropriate mouse event. (Inherited from Control .)
	RaisePaintEvent	Raises the appropriate paint event. (Inherited from Control .)
	RecreateHandle	Forces the re-creation of the handle for the control. (Inherited from Control .)
	RectangleToClient	Computes the size and location of the specified screen rectangle in client coordinates. (Inherited from Control .)
	RectangleToScreen	Computes the size and location of the specified client rectangle in screen coordinates. (Inherited from Control .)
	Refresh	Forces the control to invalidate its client area and immediately redraw itself and any child controls. (Inherited from Control .)
	RemoveOwnedForm	Removes an owned form from this form. (Inherited from Form .)
	Reset	Resets the view.
	ResetBackColor	Resets the BackColor property to its default value. (Inherited from Control .)
	ResetBindings	Causes a control bound to the BindingSource to reread all the items in the list and refresh their displayed values. (Inherited from Control .)
	ResetCursor	Resets the Cursor property to its default value. (Inherited from Control .)
	ResetFont	Resets the Font property to its default value. (Inherited from Control .)
	ResetForeColor	Resets the ForeColor property to its default value. (Inherited from Control .)
	ResetImeMode	Resets the ImeMode property to its default value.




















		(Inherited from Control .)
	ResetMouseEventArgs	Resets the control to handle the MouseLeave event. (Inherited from Control .)
	ResetRightToLeft	Resets the RightToLeft property to its default value. (Inherited from Control .)
	ResetText	Resets the Text property to its default value. (Inherited from Control .)
	ResumeLayout()	Resumes usual layout logic. (Inherited from Control .)
	ResumeLayout(Boolean)	Resumes usual layout logic, optionally forcing an immediate layout of pending layout requests. (Inherited from Control .)
	RtlTranslateAlignment(HorizontalAlignment)	Converts the specified HorizontalAlignment to the appropriate HorizontalAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateAlignment(LeftRightAlignment)	Converts the specified LeftRightAlignment to the appropriate LeftRightAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateAlignment(ContentAlignment)	Converts the specified ContentAlignment to the appropriate ContentAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateContent	Converts the specified ContentAlignment to the appropriate ContentAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateHorizontal	Converts the specified HorizontalAlignment to the appropriate HorizontalAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateLeftRight	Converts the specified LeftRightAlignment to the appropriate LeftRightAlignment to support right-to-left text. (Inherited from Control .)
	Scale(Single)	Obsolete. Scales the control and any child controls. (Inherited from Control .)
	Scale(SizeF)	Scales the control and all child controls by the specified scaling factor. (Inherited from Control .)
	Scale(Single, Single)	Obsolete. Scales the entire control and any child controls. (Inherited from Control .)
	ScaleControl	Scales the location, size, padding, and margin of a control. (Inherited from Form .)
	ScaleCore	Performs scaling of the form. (Inherited from Form .)
	ScrollControlIntoView	Scrolls the specified child control into view on an




















		auto-scroll enabled control. (Inherited from ScrollableControl.)
	ScrollToControl	Calculates the scroll offset to the specified child control. (Inherited from ScrollableControl.)
	Select()	Activates the control. (Inherited from Control.)
	Select(Boolean, Boolean)	Selects this form, and optionally selects the next or previous control. (Inherited from Form.)
	SelectNextControl	Activates the next control. (Inherited from Control.)
	SendToBack	Sends the control to the back of the z-order. (Inherited from Control.)
	SetAutoScrollMargin	Sets the size of the auto-scroll margins. (Inherited from ScrollableControl.)
	SetAutoSizeMode	Sets a value indicating how a control will behave when its AutoSize property is enabled. (Inherited from Control.)
	SetBounds(Int32, Int32, Int32, Int32)	Sets the bounds of the control to the specified location and size. (Inherited from Control.)
	SetBounds(Int32, Int32, Int32, Int32, BoundsSpecified)	Sets the specified bounds of the control to the specified location and size. (Inherited from Control.)
	SetBoundsCore	(Inherited from Form.)
	SetClientSizeCore	Sets the client size of the form. This will adjust the bounds of the form to make the client size the requested size. (Inherited from Form.)
	SetDesktopBounds	Sets the bounds of the form in desktop coordinates. (Inherited from Form.)
	SetDesktopLocation	Sets the location of the form in desktop coordinates. (Inherited from Form.)
	SetDisplayRectLocation	Positions the display window to the specified value. (Inherited from ScrollableControl.)
	SetScrollState	Sets the specified scroll state flag. (Inherited from ScrollableControl.)
	SetStyle	Sets a specified ControlStyles flag to either true or false. (Inherited from Control.)
	SetTopLevel	Sets the control as the top-level control. (Inherited from Control.)
	SetVisibleCore	(Inherited from Form.)
	Show()	Displays the control to the user. (Inherited from Control.)
	Show(IWin32Window)	Shows the form with the specified owner to the user. (Inherited from Form.)






















	ShowDialog()	Shows the form as a modal dialog box. (Inherited from Form .)
	ShowDialog(IWin32Window)	Shows the form as a modal dialog box with the specified owner. (Inherited from Form .)
	SizeFromClientSize	Determines the size of the entire control from the height and width of its client area. (Inherited from Control .)
	SuspendLayout	Temporarily suspends the layout logic for the control. (Inherited from Control .)
	ToString	Gets a string representing the current instance of the form. (Inherited from Form .)
	Update	Causes the control to redraw the invalidated regions within its client area. (Inherited from Control .)
	UpdateBounds()	Updates the bounds of the control with the current size and location. (Inherited from Control .)
	UpdateBounds(Int32, Int32, Int32, Int32)	Updates the bounds of the control with the specified size and location. (Inherited from Control .)
	UpdateBounds(Int32, Int32, Int32, Int32, Int32, Int32)	Updates the bounds of the control with the specified size, location, and client size. (Inherited from Control .)
	UpdateDefaultButton	Updates which button is the default button. (Inherited from Form .)
	UpdateStyles	Forces the assigned styles to be reapplied to the control. (Inherited from Control .)
	UpdateZOrder	Updates the control in its parent's z-order. (Inherited from Control .)
	Validate()	Verifies the value of the control losing focus by causing the Validating and Validated events to occur, in that order. (Inherited from ContainerControl .)
	Validate(Boolean)	Verifies the value of the control that is losing focus; conditionally dependent on whether automatic validation is turned on. (Inherited from ContainerControl .)
	ValidateChildren()	(Inherited from Form .)
	ValidateChildren(ValidationConstraints)	(Inherited from Form .)
	WndProc	(Inherited from Form .)





















Properties




















	Name	Description
	AcceptButton	Gets or sets the button on the form that is clicked when the user presses the ENTER key. (Inherited from Form .)






















	AccessibilityObject	Gets the AccessibleObject assigned to the control. (Inherited from Control .)
	AccessibleDefaultActionDescription	Gets or sets the default action description of the control for use by accessibility client applications. (Inherited from Control .)
	AccessibleDescription	Gets or sets the description of the control used by accessibility client applications. (Inherited from Control .)
	AccessibleName	Gets or sets the name of the control used by accessibility client applications. (Inherited from Control .)
	AccessibleRole	Gets or sets the accessible role of the control (Inherited from Control .)
	ActiveControl	Gets or sets the active control on the container control. (Inherited from ContainerControl .)
	ActiveMdiChild	Gets the currently active multiple-document interface (MDI) child window. (Inherited from Form .)
	AllowDrop	Gets or sets a value indicating whether the control can accept data that the user drags onto it. (Inherited from Control .)
	AllowTransparency	Gets or sets a value indicating whether the opacity of the form can be adjusted. (Inherited from Form .)
	Anchor	Gets or sets the edges of the container to which a control is bound and determines how a control is resized with its parent. (Inherited from Control .)
	AutoScale	Obsolete. Gets or sets a value indicating whether the form adjusts its size to fit the height of the font used on the form and scales its controls. (Inherited from Form .)
	AutoScaleBaseSize	Gets or sets the base size used for autoscaling of the form. (Inherited from Form .)
	AutoScaleDimensions	Gets or sets the dimensions that the control was designed to. (Inherited from ContainerControl .)
	AutoScaleFactor	Gets the scaling factor between the current and design-time automatic scaling dimensions. (Inherited from ContainerControl .)
	AutoScaleMode	Gets or sets the automatic scaling mode of the control. (Inherited from ContainerControl .)
	AutoScroll	Gets or sets a value indicating whether the form enables autoscrolling. (Inherited from Form .)
	AutoScrollMargin	Gets or sets the size of the auto-scroll margin. (Inherited from ScrollableControl .)
	AutoScrollMinSize	Gets or sets the minimum size of the auto-scroll. (Inherited from ScrollableControl .)
	AutoScrollOffset	Gets or sets where this control is scrolled to in

		ScrollControlIntoView(Control) . (Inherited from Control .)
	AutoScrollPosition	Gets or sets the location of the auto-scroll position. (Inherited from ScrollableControl .)
	AutoSize	Resize the form according to the setting of AutoSizeMode . (Inherited from Form .)
	AutoSizeMode	Gets or sets the mode by which the form automatically resizes itself. (Inherited from Form .)
	AutoValidate	(Inherited from Form .)
	BackColor	(Inherited from Form .)
	BackgroundImage	Gets or sets the background image displayed in the control. (Inherited from Control .)
	BackgroundImageLayout	Gets or sets the background image layout as defined in the ImageLayout enumeration. (Inherited from Control .)
	BindingContext	(Inherited from ContainerControl .)
	Bottom	Gets the distance, in pixels, between the bottom edge of the control and the top edge of its container's client area. (Inherited from Control .)
	Bounds	Gets or sets the size and location of the control including its nonclient elements, in pixels, relative to the parent control. (Inherited from Control .)
	CancelButton	Gets or sets the button control that is clicked when the user presses the ESC key. (Inherited from Form .)
	CanEnableIme	Gets a value indicating whether the ImeMode property can be set to an active value, to enable IME support. (Inherited from ContainerControl .)
	CanFocus	Gets a value indicating whether the control can receive focus. (Inherited from Control .)
	CanRaiseEvents	Determines if events can be raised on the control. (Inherited from Control .)
	CanSelect	Gets a value indicating whether the control can be selected. (Inherited from Control .)
	Capture	Gets or sets a value indicating whether the control has captured the mouse. (Inherited from Control .)
	CausesValidation	Gets or sets a value indicating whether the control causes validation to be performed on any controls that require validation when it receives focus. (Inherited from Control .)
	ClientRectangle	Gets the rectangle that represents the client area of the control. (Inherited from Control .)
	ClientSize	Gets or sets the size of the client area of the form. (Inherited from Form .)












 CompanyName	Gets the name of the company or creator of the application containing the control. (Inherited from Control .)
 Container	Gets the IContainer that contains the Component . (Inherited from Component .)
 ContainsFocus	Gets a value indicating whether the control, or one of its child controls, currently has the input focus. (Inherited from Control .)
 ContextMenu	Gets or sets the shortcut menu associated with the control. (Inherited from Control .)
 ContextMenuStrip	Gets or sets the ContextMenuStrip associated with this control. (Inherited from Control .)
 ControlBox	Gets or sets a value indicating whether a control box is displayed in the caption bar of the form. (Inherited from Form .)
 Controls	Gets the collection of controls contained within the control. (Inherited from Control .)
 Created	Gets a value indicating whether the control has been created. (Inherited from Control .)
 CreateParams	(Inherited from Form .)
 CurrentAutoScaleDimensions	Gets the current run-time dimensions of the screen. (Inherited from ContainerControl .)
 Cursor	Gets or sets the cursor that is displayed when the mouse pointer is over the control. (Inherited from Control .)
 DataBindings	Gets the data bindings for the control. (Inherited from Control .)
 DefaultCursor	Gets or sets the default cursor for the control. (Inherited from Control .)
 DefaultImeMode	Gets the default Input Method Editor (IME) mode supported by the control. (Inherited from Form .)
 DefaultMargin	Gets the space, in pixels, that is specified by default between controls. (Inherited from Control .)
 DefaultMaximumSize	Gets the length and height, in pixels, that is specified as the default maximum size of a control. (Inherited from Control .)
 DefaultMinimumSize	Gets the length and height, in pixels, that is specified as the default minimum size of a control. (Inherited from Control .)
 DefaultPadding	Gets the internal spacing, in pixels, of the contents of a control. (Inherited from Control .)
 DefaultSize	(Inherited from Form .)
 DesignMode	Gets a value that indicates whether the Component is currently in design mode. (Inherited from Component .)
 DesktopBounds	Gets or sets the size and location of the form on the Windows desktop. (Inherited from Form .)

 DesktopLocation	Gets or sets the location of the form on the Windows desktop. (Inherited from Form .)
 DialogResult	Gets or sets the dialog result for the form. (Inherited from Form .)
 DisplayRectangle	Gets the rectangle that represents the virtual display area of the control. (Inherited from ScrollableControl .)
 Disposing	Gets a value indicating whether the base Control class is in the process of disposing. (Inherited from Control .)
 Dock	Gets or sets which control borders are docked to its parent control and determines how a control is resized with its parent. (Inherited from Control .)
 DockPadding	Gets the dock padding settings for all edges of the control. (Inherited from ScrollableControl .)
 DoubleBuffered	Gets or sets a value indicating whether this control should redraw its surface using a secondary buffer to reduce or prevent flicker. (Inherited from Control .)
 Enabled	Gets or sets a value indicating whether the control can respond to user interaction. (Inherited from Control .)
 Events	Gets the list of event handlers that are attached to this Component . (Inherited from Component .)
 Focused	Gets a value indicating whether the control has input focus. (Inherited from Control .)
 Font	Gets or sets the font of the text displayed by the control. (Inherited from Control .)
 FontHeight	Gets or sets the height of the font of the control. (Inherited from Control .)
 ForeColor	Gets or sets the foreground color of the control. (Inherited from Control .)
 FormBorderStyle	Gets or sets the border style of the form. (Inherited from Form .)
 Handle	Gets the window handle that the control is bound to. (Inherited from Control .)
 HasChildren	Gets a value indicating whether the control contains one or more child controls. (Inherited from Control .)
 Height	Gets or sets the height of the control. (Inherited from Control .)
 HelpButton	Gets or sets a value indicating whether a Help button should be displayed in the caption box of the form. (Inherited from Form .)
 HorizontalScroll	Gets the characteristics associated with the horizontal scroll bar. (Inherited from ScrollableControl .)
 HScroll	Gets or sets a value indicating whether the horizontal scroll bar is visible. (Inherited from ScrollableControl .)








 Icon	Gets or sets the icon for the form. (Inherited from Form .)
 ImeMode	Gets or sets the Input Method Editor (IME) mode of the control. (Inherited from Control .)
 ImeModeBase	Gets or sets the IME mode of a control. (Inherited from Control .)
 InvokeRequired	Gets a value indicating whether the caller must call an invoke method when making method calls to the control because the caller is on a different thread than the one the control was created on. (Inherited from Control .)
 IsAccessible	Gets or sets a value indicating whether the control is visible to accessibility applications. (Inherited from Control .)
 IsDisposed	Gets a value indicating whether the control has been disposed of. (Inherited from Control .)
 IsHandleCreated	Gets a value indicating whether the control has a handle associated with it. (Inherited from Control .)
 IsMdiChild	Gets a value indicating whether the form is a multiple-document interface (MDI) child form. (Inherited from Form .)
 IsMdiContainer	Gets or sets a value indicating whether the form is a container for multiple-document interface (MDI) child forms. (Inherited from Form .)
 IsMirrored	Gets a value indicating whether the control is mirrored. (Inherited from Control .)
 IsRestrictedWindow	Gets a value indicating whether the form can use all windows and user input events without restriction. (Inherited from Form .)
 KeyPreview	Gets or sets a value indicating whether the form will receive key events before the event is passed to the control that has focus. (Inherited from Form .)
 LayoutEngine	Gets a cached instance of the control's layout engine. (Inherited from Control .)
 Left	Gets or sets the distance, in pixels, between the left edge of the control and the left edge of its container's client area. (Inherited from Control .)
 Location	Gets or sets the Point that represents the upper-left corner of the Form in screen coordinates. (Inherited from Form .)
 MainMenuStrip	Gets or sets the primary menu container for the form. (Inherited from Form .)
 Margin	Gets or sets the space between controls. (Inherited from Form .)
 MaximizeBox	Gets or sets a value indicating whether the Maximize button is displayed in the caption bar of the form. (Inherited from Form .)
 MaximizedBounds	Gets and sets the size of the form when it is maximized. (Inherited from Form .)





















	MaximumSize	Gets the maximum size the form can be resized to. (Inherited from Form .)
	MdiChildren	Gets an array of forms that represent the multiple-document interface (MDI) child forms that are parented to this form. (Inherited from Form .)
	MdiParent	Gets or sets the current multiple-document interface (MDI) parent form of this form. (Inherited from Form .)
	Menu	Gets or sets the MainMenu that is displayed in the form. (Inherited from Form .)
	MergedMenu	Gets the merged menu for the form. (Inherited from Form .)
	MinimizeBox	Gets or sets a value indicating whether the Minimize button is displayed in the caption bar of the form. (Inherited from Form .)
	MinimumSize	Gets or sets the minimum size the form can be resized to. (Inherited from Form .)
	Modal	Gets a value indicating whether this form is displayed modally. (Inherited from Form .)
	Name	Gets or sets the name of the control. (Inherited from Control .)
	Opacity	Gets or sets the opacity level of the form. (Inherited from Form .)
	OwnedForms	Gets an array of Form objects that represent all forms that are owned by this form. (Inherited from Form .)
	Owner	Gets or sets the form that owns this form. (Inherited from Form .)
	Padding	Gets or sets padding within the control. (Inherited from Control .)
	Parent	Gets or sets the parent container of the control. (Inherited from Control .)
	ParentForm	Gets the form that the container control is assigned to. (Inherited from ContainerControl .)
	PreferredSize	Gets the size of a rectangular area into which the control can fit. (Inherited from Control .)
	ProductName	Gets the product name of the assembly containing the control. (Inherited from Control .)
	ProductVersion	Gets the version of the assembly containing the control. (Inherited from Control .)
	RecreatingHandle	Gets a value indicating whether the control is currently re-creating its handle. (Inherited from Control .)
	Region	Gets or sets the window region associated with the control. (Inherited from Control .)
	RenderRightToLeft	Obsolete. This property is now obsolete. (Inherited from Control .)




















	ResizeRedraw	Gets or sets a value indicating whether the control redraws itself when resized. (Inherited from Control .)
	RestoreBounds	Gets the location and size of the form in its normal window state. (Inherited from Form .)
	Right	Gets the distance, in pixels, between the right edge of the control and the left edge of its container's client area. (Inherited from Control .)
	RightToLeft	Gets or sets a value indicating whether control's elements are aligned to support locales using right-to-left fonts. (Inherited from Control .)
	RightToLeftLayout	Gets or sets a value indicating whether right-to-left mirror placement is turned on. (Inherited from Form .)
	ScaleChildren	Gets a value that determines the scaling of child controls. (Inherited from Control .)
	ShowFocusCues	Gets a value indicating whether the control should display focus rectangles. (Inherited from Control .)
	ShowIcon	Gets or sets a value indicating whether an icon is displayed in the caption bar of the form. (Inherited from Form .)
	ShowInTaskbar	Gets or sets a value indicating whether the form is displayed in the Windows taskbar. (Inherited from Form .)
	ShowKeyboardCues	Gets a value indicating whether the user interface is in the appropriate state to show or hide keyboard accelerators. (Inherited from Control .)
	ShowWithoutActivation	Gets a value indicating whether the window will be activated when it is shown. (Inherited from Form .)
	Site	Gets or sets the site of the control. (Inherited from Control .)
	Size	Gets or sets the size of the form. (Inherited from Form .)
	SizeGripStyle	Gets or sets the style of the size grip to display in the lower-right corner of the form. (Inherited from Form .)
	StartPosition	Gets or sets the starting position of the form at run time. (Inherited from Form .)
	TabIndex	Gets or sets the tab order of the control within its container. (Inherited from Form .)
	TabStop	Gets or sets a value indicating whether the user can give the focus to this control using the TAB key. (Inherited from Form .)
	Tag	Gets or sets the object that contains data about the control. (Inherited from Control .)
	Text	(Inherited from Form .)
	Top	Gets or sets the distance, in pixels, between the top edge of the control and the top edge of its container's client area. (Inherited

		from Control .)
	TopLevel	Gets or sets a value indicating whether to display the form as a top-level window. (Inherited from Form .)
	TopLevelControl	Gets the parent control that is not parented by another Windows Forms control. Typically, this is the outermost Form that the control is contained in. (Inherited from Control .)
	TopMost	Gets or sets a value indicating whether the form should be displayed as a topmost form. (Inherited from Form .)
	TransparencyKey	Gets or sets the color that will represent transparent areas of the form. (Inherited from Form .)
	UseWaitCursor	Gets or sets a value indicating whether to use the wait cursor for the current control and all child controls. (Inherited from Control .)
	VerticalScroll	Gets the characteristics associated with the vertical scroll bar. (Inherited from ScrollableControl .)
	Visible	Gets or sets a value indicating whether the control and all its child controls are displayed. (Inherited from Control .)
	VScroll	Gets or sets a value indicating whether the vertical scroll bar is visible. (Inherited from ScrollableControl .)
	Width	Gets or sets the width of the control. (Inherited from Control .)
	WindowState	Gets or sets a value that indicates whether form is minimized, maximized, or normal. (Inherited from Form .)
	WindowTarget	This property is not relevant for this class. (Inherited from Control .)













Events

	Name	Description
	Activated	Occurs when the form is activated in code or by the user. (Inherited from Form .)
	AutoSizeChanged	Occurs when the AutoSize property changes. (Inherited from Form .)
	AutoValidateChanged	Occurs when the AutoValidate property changes. (Inherited from Form .)
	BackColorChanged	Occurs when the value of the BackColor property changes. (Inherited from Control .)
	BackgroundImageChanged	Occurs when the value of the BackgroundImage property changes. (Inherited from Control .)
	BackgroundImageLayoutChanged	Occurs when the BackgroundImageLayout property changes. (Inherited from Control .)
	BindingContextChanged	Occurs when the value of the BindingContext property changes.

		(Inherited from Control .)
	CausesValidationChanged	Occurs when the value of the CausesValidation property changes. (Inherited from Control .)
	ChangeUICues	Occurs when the focus or keyboard user interface (UI) cues change. (Inherited from Control .)
	Click	Occurs when the control is clicked. (Inherited from Control .)
	ClientSizeChanged	Occurs when the value of the ClientSize property changes. (Inherited from Control .)
	Closed	Occurs when the form is closed. (Inherited from Form .)
	Closing	Occurs when the form is closing. (Inherited from Form .)
	ContextMenuChanged	Occurs when the value of the ContextMenu property changes. (Inherited from Control .)
	ContextMenuStripChanged	Occurs when the value of the ContextMenuStrip property changes. (Inherited from Control .)
	ControlAdded	Occurs when a new control is added to the Control.ControlCollection . (Inherited from Control .)
	ControlRemoved	Occurs when a control is removed from the Control.ControlCollection . (Inherited from Control .)
	CursorChanged	Occurs when the value of the Cursor property changes. (Inherited from Control .)
	Deactivate	Occurs when the form loses focus and is no longer the active form. (Inherited from Form .)
	Disposed	Occurs when the component is disposed by a call to the Dispose() method. (Inherited from Component .)
	DockChanged	Occurs when the value of the Dock property changes. (Inherited from Control .)
	DoubleClick	Occurs when the control is double-clicked. (Inherited from Control .)
	DragDrop	Occurs when a drag-and-drop operation is completed. (Inherited from Control .)
	DragEnter	Occurs when an object is dragged into the control's bounds. (Inherited from Control .)
	DragLeave	Occurs when an object is dragged out of the control's bounds. (Inherited from Control .)
	DragOver	Occurs when an object is dragged over the control's bounds. (Inherited from Control .)
	EnabledChanged	Occurs when the Enabled property value has changed. (Inherited from Control .)
	Enter	Occurs when the control is entered. (Inherited from Control .)

 FontChanged	Occurs when the Font property value changes. (Inherited from Control .)
 ForeColorChanged	Occurs when the ForeColor property value changes. (Inherited from Control .)
 FormClosed	Occurs after the form is closed. (Inherited from Form .)
 FormClosing	Occurs before the form is closed. (Inherited from Form .)
 GiveFeedback	Occurs during a drag operation. (Inherited from Control .)
 GotFocus	Occurs when the control receives focus. (Inherited from Control .)
 HandleCreated	Occurs when a handle is created for the control. (Inherited from Control .)
 HandleDestroyed	Occurs when the control's handle is in the process of being destroyed. (Inherited from Control .)
 HelpButtonClicked	Occurs when the Help button is clicked. (Inherited from Form .)
 HelpRequested	Occurs when the user requests help for a control. (Inherited from Control .)
 ImeModeChanged	Occurs when the ImeMode property has changed. (Inherited from Control .)
 InputLanguageChanged	Occurs after the input language of the form has changed. (Inherited from Form .)
 InputLanguageChanging	Occurs when the user attempts to change the input language for the form. (Inherited from Form .)
 Invalidated	Occurs when a control's display requires redrawing. (Inherited from Control .)
 KeyDown	Occurs when a key is pressed while the control has focus. (Inherited from Control .)
 KeyPress	Occurs when a key is pressed while the control has focus. (Inherited from Control .)
 KeyUp	Occurs when a key is released while the control has focus. (Inherited from Control .)
 Layout	Occurs when a control should reposition its child controls. (Inherited from Control .)
 Leave	Occurs when the input focus leaves the control. (Inherited from Control .)
 Load	Occurs before a form is displayed for the first time. (Inherited from Form .)
 LocationChanged	Occurs when the Location property value has changed. (Inherited from Control .)
 LostFocus	Occurs when the control loses focus. (Inherited from Control .)
 MarginChanged	Occurs when the Margin property changes. (Inherited from Form .)

 MaximizedBoundsChanged	Occurs when the value of the MaximizedBounds property has changed. (Inherited from Form .)
 MaximumSizeChanged	Occurs when the value of the MaximumSize property has changed. (Inherited from Form .)
 MdiChildActivate	Occurs when a multiple-document interface (MDI) child form is activated or closed within an MDI application. (Inherited from Form .)
 MenuComplete	Occurs when the menu of a form loses focus. (Inherited from Form .)
 MenuStart	Occurs when the menu of a form receives focus. (Inherited from Form .)
 MinimumSizeChanged	Occurs when the value of the MinimumSize property has changed. (Inherited from Form .)
 MouseCaptureChanged	Occurs when the control loses mouse capture. (Inherited from Control .)
 MouseClicked	Occurs when the control is clicked by the mouse. (Inherited from Control .)
 MouseDoubleClick	Occurs when the control is double clicked by the mouse. (Inherited from Control .)
 MouseDown	Occurs when the mouse pointer is over the control and a mouse button is pressed. (Inherited from Control .)
 MouseEnter	Occurs when the mouse pointer enters the control. (Inherited from Control .)
 MouseHover	Occurs when the mouse pointer rests on the control. (Inherited from Control .)
 MouseLeave	Occurs when the mouse pointer leaves the control. (Inherited from Control .)
 MouseMove	Occurs when the mouse pointer is moved over the control. (Inherited from Control .)
 MouseUp	Occurs when the mouse pointer is over the control and a mouse button is released. (Inherited from Control .)
 MouseWheel	Occurs when the mouse wheel moves while the control has focus. (Inherited from Control .)
 Move	Occurs when the control is moved. (Inherited from Control .)
 PaddingChanged	Occurs when the control's padding changes. (Inherited from Control .)
 Paint	Occurs when the control is redrawn. (Inherited from Control .)
 ParentChanged	Occurs when the Parent property value changes. (Inherited from Control .)
 PreviewKeyDown	Occurs before the KeyDown event when a key is pressed while

		focus is on this control. (Inherited from Control.)
	QueryAccessibilityHelp	Occurs when AccessibleObject is providing help to accessibility applications. (Inherited from Control.)
	QueryContinueDrag	Occurs during a drag-and-drop operation and enables the drag source to determine whether the drag-and-drop operation should be canceled. (Inherited from Control.)
	RegionChanged	Occurs when the value of the Region property changes. (Inherited from Control.)
	Resize	Occurs when the control is resized. (Inherited from Control.)
	ResizeBegin	Occurs when a form enters resizing mode. (Inherited from Form.)
	ResizeEnd	Occurs when a form exits resizing mode. (Inherited from Form.)
	RightToLeftChanged	Occurs when the RightToLeft property value changes. (Inherited from Control.)
	RightToLeftLayoutChanged	Occurs after the value of the RightToLeftLayout property changes. (Inherited from Form.)
	Scroll	Occurs when the user or code scrolls through the client area. (Inherited from ScrollableControl.)
	Shown	Occurs whenever the form is first displayed. (Inherited from Form.)
	SizeChanged	Occurs when the Size property value changes. (Inherited from Control.)
	StyleChanged	Occurs when the control style changes. (Inherited from Control.)
	SystemColorsChanged	Occurs when the system colors change. (Inherited from Control.)
	TabIndexChanged	Occurs when the value of the TabIndex property changes. (Inherited from Form.)
	TabStopChanged	Occurs when the TabStop property changes. (Inherited from Form.)
	TextChanged	Occurs when the Text property value changes. (Inherited from Control.)
	Validated	Occurs when the control is finished validating. (Inherited from Control.)
	Validating	Occurs when the control is validating. (Inherited from Control.)
	VisibleChanged	Occurs when the Visible property value changes. (Inherited from Control.)

See Also

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm Constructor

Initialises a FormView for the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public WoodstocksIMSForm( )
```

VB

```
Public Sub New
```

C++

```
public:  
WoodstocksIMSForm( )
```

F#

```
new : unit -> WoodstocksIMSForm
```

See Also















[WoodstocksIMSForm Class](#)




















[Woodstocks.WoodstocksIMS.Presentation Namespace](#)



















WoodstockMSForm.WoodstockMSForm Methods














The [WoodstockMSForm](#) type exposes the following members.





















Methods



	Name	Description
	AccessibilityNotifyClients(AccessibleEvents, Int32)	Notifies the accessibility client applications of the specified AccessibleEvents for the specified child control. (Inherited from Control .)
	AccessibilityNotifyClients(AccessibleEvents, Int32, Int32)	Notifies the accessibility client applications of the specified AccessibleEvents for the specified child control . (Inherited from Control .)
	Activate	Activates the form and gives it focus. (Inherited from Form .)
	ActivateMdiChild	Activates the MDI child of a form. (Inherited from Form .)
	AddOwnedForm	Adds an owned form to this form. (Inherited from Form .)
	AdjustFormScrollbars	Adjusts the scroll bars on the container based on the current control positions and the control currently selected. (Inherited from Form .)
	ApplyAutoScaling	Obsolete. Resizes the form according to the current value of the AutoScaleBaseSize property and the size of the current font. (Inherited from Form .)
	BeginInvoke(Delegate)	Executes the specified delegate asynchronously on the thread that the control's underlying handle was created on. (Inherited from Control .)
	BeginInvoke(Delegate, Object[])	Executes the specified delegate asynchronously with the specified arguments, on the thread that the control's underlying handle was created on. (Inherited from Control .)
	BringToFront	Brings the control to the front of the z-order. (Inherited from Control .)
	CenterToParent	Centers the position of the form within the bounds of the parent form. (Inherited from Form .)
	CenterToScreen	Centers the form on the current screen. (Inherited from Form .)
	Close	Closes the form. (Inherited from Form .)
	Contains	Retrieves a value indicating whether the specified control is a child of the control. (Inherited from Control .)





























	CreateAccessibilityInstance	Creates a new accessibility object for the control. (Inherited from Control .)
	CreateControl	Forces the creation of the visible control, including the creation of the handle and any visible child controls. (Inherited from Control .)
	CreateControlsInstance	(Inherited from Form .)
	CreateGraphics	Creates the Graphics for the control. (Inherited from Control .)
	CreateHandle	Creates the handle for the form. If a derived class overrides this function, it must call the base implementation. (Inherited from Form .)
	CreateObjRef	Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object. (Inherited from MarshalByRefObject .)
	DefWndProc	(Inherited from Form .)
	DestroyHandle	Destroys the handle associated with the control. (Inherited from Control .)
	DisableImport	Disables import option of the View.
	DisplayStatus	Updates the status being displayed by the Form to the user.
	DisplayToys	Causes the form to display toy data in the DataGridView of the Form used to display toy data.
	DisplayUnsavedDataDialog	Displays a message box to the user informing them that imported data that has been modified has not been saved and asks user if they would like to save changes.
	Dispose()	Releases all resources used by the Component . (Inherited from Component .)
	Dispose(Boolean)	Clean up any resources being used. (Overrides Form.Dispose(Boolean) .)
	DoDragDrop	Begins a drag-and-drop operation. (Inherited from Control .)
	DrawToBitmap	Supports rendering to the specified bitmap. (Inherited from Control .)
	EnableImport	Enables import option of the View.
	EndInvoke	Retrieves the return value of the asynchronous operation represented by the IAsyncResult passed. (Inherited from Control .)
	Equals	Determines whether the specified Object is equal to the current Object . (Inherited from Object .)






















	ExportToys	Exports toy data
	Finalize	Releases unmanaged resources and performs other cleanup operations before the Component is reclaimed by garbage collection. (Inherited from Component .)
	FindForm	Retrieves the form that the control is on. (Inherited from Control .)
	Focus	Sets input focus to the control. (Inherited from Control .)
	GetAccessibilityObjectById	Retrieves the specified AccessibleObject . (Inherited from Control .)
	GetAutoSizeMode	Retrieves a value indicating how a control will behave when its AutoSize property is enabled. (Inherited from Control .)
	GetChildAtPoint(Point)	Retrieves the child control that is located at the specified coordinates. (Inherited from Control .)
	GetChildAtPoint(Point, GetChildAtPointSkip)	Retrieves the child control that is located at the specified coordinates, specifying whether to ignore child controls of a certain type. (Inherited from Control .)
	GetContainerControl	Returns the next ContainerControl up the control's chain of parent controls. (Inherited from Control .)
	GetHashCode	Serves as a hash function for a particular type. (Inherited from Object .)
	GetLifetimeService	Retrieves the current lifetime service object that controls the lifetime policy for this instance. (Inherited from MarshalByRefObject .)
	GetNextControl	Retrieves the next control forward or back in the tab order of child controls. (Inherited from Control .)
	GetPreferredSize	Retrieves the size of a rectangular area into which a control can be fitted. (Inherited from Control .)
	GetScaledBounds	(Inherited from Form .)
	GetScrollState	Determines whether the specified flag has been set. (Inherited from ScrollableControl .)
	GetService	Returns an object that represents a service provided by the Component or by its Container . (Inherited from Component .)
	GetStatus	Gets the status that is displayed by the Form to the user.
	GetStyle	Retrieves the value of the specified control style bit for the control. (Inherited from Control .)













	GetTopLevel	Determines if the control is a top-level control. (Inherited from Control .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	Hide	Conceals the control from the user. (Inherited from Control .)
	ImportToys	Imports data into the WoodstocksIMS .
	InitializeLifetimeService	Obtains a lifetime service object to control the lifetime policy for this instance. (Inherited from MarshalByRefObject .)
	InitLayout	Called after the control has been added to another container. (Inherited from Control .)
	Invalidate()	Invalidates the entire surface of the control and causes the control to be redrawn. (Inherited from Control .)
	Invalidate(Region)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. (Inherited from Control .)
	Invalidate(Boolean)	Invalidates a specific region of the control and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control .)
	Invalidate(Rectangle)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. (Inherited from Control .)
	Invalidate(Region, Boolean)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control .)
	Invalidate(Rectangle, Boolean)	Invalidates the specified region of the control (adds it to the control's update region, which is the area that will be repainted at the next paint operation), and causes a paint message to be sent to the control. Optionally, invalidates the child controls assigned to the control. (Inherited from Control .)
	Invoke(Delegate)	Executes the specified delegate on the thread that owns the control's underlying window handle.
























		(Inherited from Control .)
	Invoke(Delegate, Object[])	Executes the specified delegate, on the thread that owns the control's underlying window handle, with the specified list of arguments. (Inherited from Control .)
	InvokeGotFocus	Raises the GotFocus event for the specified control. (Inherited from Control .)
	InvokeLostFocus	Raises the LostFocus event for the specified control. (Inherited from Control .)
	InvokeOnClick	Raises the Click event for the specified control. (Inherited from Control .)
	InvokePaint	Raises the Paint event for the specified control. (Inherited from Control .)
	InvokePaintBackground	Raises the PaintBackground event for the specified control. (Inherited from Control .)
	IsInputChar	Determines if a character is an input character that the control recognizes. (Inherited from Control .)
	IsInputKey	Determines whether the specified key is a regular input key or a special key that requires preprocessing. (Inherited from Control .)
	LayoutMdi	Arranges the multiple-document interface (MDI) child forms within the MDI parent form. (Inherited from Form .)
	MemberwiseClone()	Creates a shallow copy of the current Object . (Inherited from Object .)
	MemberwiseClone(Boolean)	Creates a shallow copy of the current MarshalByRefObject object. (Inherited from MarshalByRefObject .)
	NotifyDirectoryNotFound	Notifies the user that a DirectoryNotFoundException has occurred.
	NotifyDuplicateToyException	Notifies the user that an duplicate toy has been added to the collection of toys.
	NotifyError	Notifies the user that an error has occurred.
	NotifyExportCancellation	Notifies the user that exportation has been cancelled.
	NotifyExportCompletion()	Notifies the user that exportation has completed.
	NotifyExportCompletion(Exception, Boolean)	Notifies the user that exportation has completed.
	NotifyFileNotFound	Notifies the user that a FileNotFoundException has occurred.
	NotifyImportCancellation	Notifies the user that importation has been cancelled.
	NotifyImportCompletion	Notifies the user that importation has completed.





















	NotifyInvalidate	Raises the Invalidated event with a specified region of the control to invalidate. (Inherited from Control .)
	NotifyUnknownException	Notifies the user that an unknown problem has occurred.
	OnActivated	Raises the Activated event. (Inherited from Form .)
	OnAutoSizeChanged	Raises the AutoSizeChanged event. (Inherited from Control .)
	OnAutoValidateChanged	Raises the AutoValidateChanged event. (Inherited from ContainerControl .)
	OnBackColorChanged	Raises the BackColorChanged event. (Inherited from Control .)
	OnBackgroundImageChanged	Raises the BackgroundImageChanged event. (Inherited from Form .)
	OnBackgroundImageLayoutChanged	Raises the BackgroundImageLayoutChanged event. (Inherited from Form .)
	OnBindingContextChanged	Raises the BindingContextChanged event. (Inherited from Control .)
	OnCausesValidationChanged	Raises the CausesValidationChanged event. (Inherited from Control .)
	OnChangeUICues	Raises the ChangeUICues event. (Inherited from Control .)
	OnClick	Raises the Click event. (Inherited from Control .)
	OnClientSizeChanged	Raises the ClientSizeChanged event. (Inherited from Control .)
	OnClosed	Raises the Closed event. (Inherited from Form .)
	OnClosing	Raises the Closing event. (Inherited from Form .)
	OnContextMenuChanged	Raises the ContextMenuChanged event. (Inherited from Control .)
	OnContextMenuStripChanged	Raises the ContextMenuStripChanged event. (Inherited from Control .)
	OnControlAdded	Raises the ControlAdded event. (Inherited from Control .)
	OnControlRemoved	Raises the ControlRemoved event. (Inherited from Control .)
	OnCreateControl	Raises the CreateControl event. (Inherited from Form .)
	OnCursorChanged	Raises the CursorChanged event. (Inherited from Control .)
	OnDeactivate	Raises the Deactivate event. (Inherited from Form .)

















	OnDockChanged	Raises the DockChanged event. (Inherited from Control .)
	OnDoubleClick	Raises the DoubleClick event. (Inherited from Control .)
	OnDragDrop	Raises the DragDrop event. (Inherited from Control .)
	OnDragEnter	Raises the DragEnter event. (Inherited from Control .)
	OnDragLeave	Raises the DragLeave event. (Inherited from Control .)
	OnDragOver	Raises the DragOver event. (Inherited from Control .)
	OnEnabledChanged	(Inherited from Form .)
	OnEnter	Raises the Enter event. (Inherited from Form .)
	OnFontChanged	(Inherited from Form .)
	OnForeColorChanged	Raises the ForeColorChanged event. (Inherited from Control .)
	OnFormClosed	Raises the FormClosed event. (Inherited from Form .)
	OnFormClosing	Raises the FormClosing event. (Inherited from Form .)
	OnGiveFeedback	Raises the GiveFeedback event. (Inherited from Control .)
	OnGotFocus	Raises the GotFocus event. (Inherited from Control .)
	OnHandleCreated	(Inherited from Form .)
	OnHandleDestroyed	(Inherited from Form .)
	OnHelpButtonClicked	Raises the HelpButtonClicked event. (Inherited from Form .)
	OnHelpRequested	Raises the HelpRequested event. (Inherited from Control .)
	OnImeModeChanged	Raises the ImeModeChanged event. (Inherited from Control .)
	OnInputLanguageChanged	Raises the InputLanguageChanged event. (Inherited from Form .)
	OnInputLanguageChanging	Raises the InputLanguageChanging event. (Inherited from Form .)
	OnInvalidated	Raises the Invalidated event. (Inherited from Control .)
	OnKeyDown	Raises the KeyDown event. (Inherited from Control .)
	OnKeyPress	Raises the KeyPress event. (Inherited from Control .)
	OnKeyUp	Raises the KeyUp event. (Inherited from Control .)
	OnLayout	Raises the Layout event. (Inherited from Form .)
	OnLeave	Raises the Leave event. (Inherited from Control .)
	OnLoad	Raises the Load event. (Inherited from Form .)






















	OnLocationChanged	Raises the LocationChanged event. (Inherited from Control .)
	OnLostFocus	Raises the LostFocus event. (Inherited from Control .)
	OnMarginChanged	Raises the MarginChanged event. (Inherited from Control .)
	OnMaximizedBoundsChanged	Raises the MaximizedBoundsChanged event. (Inherited from Form .)
	OnMaximumSizeChanged	Raises the MaximumSizeChanged event. (Inherited from Form .)
	OnMdiChildActivate	Raises the MdiChildActivate event. (Inherited from Form .)
	OnMenuComplete	Raises the MenuComplete event. (Inherited from Form .)
	OnMenuStart	Raises the MenuStart event. (Inherited from Form .)
	OnMinimumSizeChanged	Raises the MinimumSizeChanged event. (Inherited from Form .)
	OnMouseCaptureChanged	Raises the MouseCaptureChanged event. (Inherited from Control .)
	OnMouseClicked	Raises the MouseClicked event. (Inherited from Control .)
	OnMouseDoubleClick	Raises the MouseDoubleClick event. (Inherited from Control .)
	OnMouseDown	Raises the MouseDown event. (Inherited from Control .)
	OnMouseEnter	Raises the MouseEnter event. (Inherited from Control .)
	OnMouseHover	Raises the MouseHover event. (Inherited from Control .)
	OnMouseLeave	Raises the MouseLeave event. (Inherited from Control .)
	OnMouseMove	Raises the MouseMove event. (Inherited from Control .)
	OnMouseUp	Raises the MouseUp event. (Inherited from Control .)
	OnMouseWheel	Raises the MouseWheel event. (Inherited from ScrollableControl .)
	OnMove	Raises the Move event. (Inherited from Control .)
	OnNotifyMessage	Notifies the control of Windows messages. (Inherited from Control .)
	OnPaddingChanged	Raises the PaddingChanged event. (Inherited from ScrollableControl .)



















	OnPaint	(Inherited from Form .)
	OnPaintBackground	Paints the background of the control. (Inherited from ScrollableControl .)
	OnParentBackColorChanged	Raises the BackColorChanged event when the BackColor property value of the control's container changes. (Inherited from Control .)
	OnParentBackgroundImageChanged	Raises the BackgroundImageChanged event when the BackgroundImage property value of the control's container changes. (Inherited from Control .)
	OnParentBindingContextChanged	Raises the BindingContextChanged event when the BindingContext property value of the control's container changes. (Inherited from Control .)
	OnParentChanged	(Inherited from ContainerControl .)
	OnParentCursorChanged	Raises the CursorChanged event. (Inherited from Control .)
	OnParentEnabledChanged	Raises the EnabledChanged event when the Enabled property value of the control's container changes. (Inherited from Control .)
	OnParentFontChanged	Raises the FontChanged event when the Font property value of the control's container changes. (Inherited from Control .)
	OnParentForeColorChanged	Raises the ForeColorChanged event when the ForeColor property value of the control's container changes. (Inherited from Control .)
	OnParentRightToLeftChanged	Raises the RightToLeftChanged event when the RightToLeft property value of the control's container changes. (Inherited from Control .)
	OnParentVisibleChanged	Raises the VisibleChanged event when the Visible property value of the control's container changes. (Inherited from Control .)
	OnPreviewKeyDown	Raises the PreviewKeyDown event. (Inherited from Control .)
	OnPrint	Raises the Paint event. (Inherited from Control .)
	OnQueryContinueDrag	Raises the QueryContinueDrag event. (Inherited from Control .)
	OnRegionChanged	Raises the RegionChanged event. (Inherited from Control .)
	OnResize	(Inherited from Form .)
	OnResizeBegin	Raises the ResizeBegin event. (Inherited from Form .)
	OnResizeEnd	Raises the ResizeEnd event. (Inherited from Form .)

	OnRightToLeftChanged	(Inherited from ScrollableControl .)
	OnRightToLeftLayoutChanged	Raises the RightToLeftLayoutChanged event. (Inherited from Form .)
	OnScroll	Raises the Scroll event. (Inherited from ScrollableControl .)
	OnShown	Raises the Shown event. (Inherited from Form .)
	OnSizeChanged	Raises the SizeChanged event. (Inherited from Control .)
	OnStyleChanged	(Inherited from Form .)
	OnSystemColorsChanged	Raises the SystemColorsChanged event. (Inherited from Control .)
	OnTabIndexChanged	Raises the TabIndexChanged event. (Inherited from Control .)
	OnTabStopChanged	Raises the TabStopChanged event. (Inherited from Control .)
	OnTextChanged	(Inherited from Form .)
	OnValidated	Raises the Validated event. (Inherited from Control .)
	OnValidating	Raises the Validating event. (Inherited from Control .)
	OnVisibleChanged	Raises the VisibleChanged event. (Inherited from Form .)
	PerformAutoScale	Performs scaling of the container control and its children. (Inherited from ContainerControl .)
	PerformLayout()	Forces the control to apply layout logic to all its child controls. (Inherited from Control .)
	PerformLayout(Control, String)	Forces the control to apply layout logic to all its child controls. (Inherited from Control .)
	PointToClient	Computes the location of the specified screen point into client coordinates. (Inherited from Control .)
	PointToScreen	Computes the location of the specified client point into screen coordinates. (Inherited from Control .)
	PreProcessControlMessage	Preprocesses keyboard or input messages within the message loop before they are dispatched. (Inherited from Control .)
	PreProcessMessage	Preprocesses keyboard or input messages within the message loop before they are dispatched. (Inherited from Control .)
	ProcessCmdKey	Processes a command key. (Inherited from Form .)
	ProcessDialogChar	Processes a dialog character. (Inherited from Form .)
	ProcessDialogKey	Processes a dialog box key. (Inherited from Form .)

	ProcessKeyEventArgs	Processes a key message and generates the appropriate control events. (Inherited from Control .)
	ProcessKeyMessage	Processes a keyboard message. (Inherited from Control .)
	ProcessKeyPreview	(Inherited from Form .)
	ProcessMnemonic	Processes a mnemonic character. (Inherited from Form .)
	ProcessTabKey	(Inherited from Form .)
	RaiseDragEvent	Raises the appropriate drag event. (Inherited from Control .)
	RaiseKeyEvent	Raises the appropriate key event. (Inherited from Control .)
	RaiseMouseEvent	Raises the appropriate mouse event. (Inherited from Control .)
	RaisePaintEvent	Raises the appropriate paint event. (Inherited from Control .)
	RecreateHandle	Forces the re-creation of the handle for the control. (Inherited from Control .)
	RectangleToClient	Computes the size and location of the specified screen rectangle in client coordinates. (Inherited from Control .)
	RectangleToScreen	Computes the size and location of the specified client rectangle in screen coordinates. (Inherited from Control .)
	Refresh	Forces the control to invalidate its client area and immediately redraw itself and any child controls. (Inherited from Control .)
	RemoveOwnedForm	Removes an owned form from this form. (Inherited from Form .)
	Reset	Resets the view.
	ResetBackColor	Resets the BackColor property to its default value. (Inherited from Control .)
	ResetBindings	Causes a control bound to the BindingSource to reread all the items in the list and refresh their displayed values. (Inherited from Control .)
	ResetCursor	Resets the Cursor property to its default value. (Inherited from Control .)
	ResetFont	Resets the Font property to its default value. (Inherited from Control .)
	ResetForeColor	Resets the ForeColor property to its default value.

		(Inherited from Control .)
	ResetImeMode	Resets the ImeMode property to its default value. (Inherited from Control .)
	ResetMouseEventArgs	Resets the control to handle the MouseLeave event. (Inherited from Control .)
	ResetRightToLeft	Resets the RightToLeft property to its default value. (Inherited from Control .)
	ResetText	Resets the Text property to its default value. (Inherited from Control .)
	ResumeLayout()	Resumes usual layout logic. (Inherited from Control .)
	ResumeLayout(Boolean)	Resumes usual layout logic, optionally forcing an immediate layout of pending layout requests. (Inherited from Control .)
	RtlTranslateAlignment(HorizontalAlignment)	Converts the specified HorizontalAlignment to the appropriate HorizontalAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateAlignment(LeftRightAlignment)	Converts the specified LeftRightAlignment to the appropriate LeftRightAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateAlignment(ContentAlignment)	Converts the specified ContentAlignment to the appropriate ContentAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateContent	Converts the specified ContentAlignment to the appropriate ContentAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateHorizontal	Converts the specified HorizontalAlignment to the appropriate HorizontalAlignment to support right-to-left text. (Inherited from Control .)
	RtlTranslateLeftRight	Converts the specified LeftRightAlignment to the appropriate LeftRightAlignment to support right-to-left text. (Inherited from Control .)
	Scale(Single)	Obsolete. Scales the control and any child controls. (Inherited from Control .)
	Scale(SizeF)	Scales the control and all child controls by the specified scaling factor. (Inherited from Control .)
	Scale(Single, Single)	Obsolete. Scales the entire control and any child controls. (Inherited from Control .)
	ScaleControl	Scales the location, size, padding, and margin of a control. (Inherited from Form .)

	ScaleCore	Performs scaling of the form. (Inherited from Form .)
	ScrollControlIntoView	Scrolls the specified child control into view on an auto-scroll enabled control. (Inherited from ScrollableControl .)
	ScrollToControl	Calculates the scroll offset to the specified child control. (Inherited from ScrollableControl .)
	Select()	Activates the control. (Inherited from Control .)
	Select(Boolean, Boolean)	Selects this form, and optionally selects the next or previous control. (Inherited from Form .)
	SelectNextControl	Activates the next control. (Inherited from Control .)
	SendToBack	Sends the control to the back of the z-order. (Inherited from Control .)
	SetAutoScrollMargin	Sets the size of the auto-scroll margins. (Inherited from ScrollableControl .)
	SetAutoSizeMode	Sets a value indicating how a control will behave when its AutoSize property is enabled. (Inherited from Control .)
	SetBounds(Int32, Int32, Int32, Int32)	Sets the bounds of the control to the specified location and size. (Inherited from Control .)
	SetBounds(Int32, Int32, Int32, Int32, BoundsSpecified)	Sets the specified bounds of the control to the specified location and size. (Inherited from Control .)
	SetBoundsCore	(Inherited from Form .)
	SetClientSizeCore	Sets the client size of the form. This will adjust the bounds of the form to make the client size the requested size. (Inherited from Form .)
	SetDesktopBounds	Sets the bounds of the form in desktop coordinates. (Inherited from Form .)
	SetDesktopLocation	Sets the location of the form in desktop coordinates. (Inherited from Form .)
	SetDisplayRectLocation	Positions the display window to the specified value. (Inherited from ScrollableControl .)
	SetScrollState	Sets the specified scroll state flag. (Inherited from ScrollableControl .)
	SetStyle	Sets a specified ControlStyles flag to either true or false. (Inherited from Control .)
	SetTopLevel	Sets the control as the top-level control. (Inherited from Control .)
	SetVisibleCore	(Inherited from Form .)
	Show()	Displays the control to the user. (Inherited from Control .)

 Show(IWin32Window)	Shows the form with the specified owner to the user. (Inherited from Form .)
 ShowDialog()	Shows the form as a modal dialog box. (Inherited from Form .)
 ShowDialog(IWin32Window)	Shows the form as a modal dialog box with the specified owner. (Inherited from Form .)
 SizeFromClientSize	Determines the size of the entire control from the height and width of its client area. (Inherited from Control .)
 SuspendLayout	Temporarily suspends the layout logic for the control. (Inherited from Control .)
 ToString	Gets a string representing the current instance of the form. (Inherited from Form .)
 Update	Causes the control to redraw the invalidated regions within its client area. (Inherited from Control .)
 UpdateBounds()	Updates the bounds of the control with the current size and location. (Inherited from Control .)
 UpdateBounds(Int32, Int32, Int32, Int32)	Updates the bounds of the control with the specified size and location. (Inherited from Control .)
 UpdateBounds(Int32, Int32, Int32, Int32, Int32, Int32)	Updates the bounds of the control with the specified size, location, and client size. (Inherited from Control .)
 UpdateDefaultButton	Updates which button is the default button. (Inherited from Form .)
 UpdateStyles	Forces the assigned styles to be reapplied to the control. (Inherited from Control .)
 UpdateZOrder	Updates the control in its parent's z-order. (Inherited from Control .)
 Validate()	Verifies the value of the control losing focus by causing the Validating and Validated events to occur, in that order. (Inherited from ContainerControl .)
 Validate(Boolean)	Verifies the value of the control that is losing focus; conditionally dependent on whether automatic validation is turned on. (Inherited from ContainerControl .)
 ValidateChildren()	(Inherited from Form .)
 ValidateChildren(ValidationConstraints)	(Inherited from Form .)
 WndProc	(Inherited from Form .)

See Also

[WoodstockslMSForm Class](#)

A Sandcastle Documented Class Library

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.DisableImport Method

Disables import option of the View.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void DisableImport()
```

VB

```
Public Sub DisableImport
```

C++

```
public:  
virtual void DisableImport() sealed
```

F#

```
abstract DisableImport : unit -> unit  
override DisableImport : unit -> unit
```

Implements

[IWoodstocksIMSView.DisableImport\(\)](#)

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.DisplayStatus Method

Updates the status being displayed by the Form to the user.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void DisplayStatus(  
    string status  
)
```

VB

```
Protected Overridable Sub DisplayStatus (  
    status As String  
)
```

C++

```
protected:  
virtual void DisplayStatus(  
    String^ status  
)
```

F#

```
abstract DisplayStatus :  
    status : string -> unit  
override DisplayStatus :  
    status : string -> unit
```

Parameters

status

Type: [System.String](#)

A status to be displayed to the user.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.DisplayToys Method

Causes the form to display toy data in the DataGridView of the Form used to display toy data.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void DisplayToys(  
    IToys woodstocksToys  
)
```

VB

```
Protected Overridable Sub DisplayToys (  
    woodstocksToys As IToys  
)
```

C++

```
protected:  
virtual void DisplayToys(  
    IToys^ woodstocksToys  
)
```

F#

```
abstract DisplayToys :  
    woodstocksToys : IToys -> unit  
override DisplayToys :  
    woodstocksToys : IToys -> unit
```

Parameters

woodstocksToys

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The toy data to be displayed to the user.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.DisplayUnsavedDataDialog Method

Displays a message box to the user informing them that imported data that has been modified has not been saved and asks user if they would like to save changes.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected DialogResult DisplayUnsavedDataDialog()
```

VB

```
Protected Function DisplayUnsavedDataDialog As DialogResult
```

C++

```
protected:  
DialogResult DisplayUnsavedDataDialog()
```

F#

```
member DisplayUnsavedDataDialog : unit -> DialogResult
```

Return Value

Type: [DialogResult](#)

The result of the users action indicating if they wish to save, or not.



See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstockIMSForm.Dispose Method

Overload List

	Name	Description
	Dispose()	Releases all resources used by the Component . (Inherited from Component .)
	Dispose(Boolean)	Clean up any resources being used. (Overrides Form.Dispose(Boolean) .)

See Also

[WoodstockIMSForm Class](#)

[Woodstock.WoodstockIMS.Presentation Namespace](#)

WoodstocksIMSForm.Dispose Method (Boolean)

Clean up any resources being used.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

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

VB

```
Protected Overrides Sub Dispose (  
    disposing As Boolean  
)
```

C++

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

F#

```
abstract Dispose :  
    disposing : bool -> unit  
override Dispose :  
    disposing : bool -> unit
```

Parameters

disposing

Type: [System.Boolean](#)

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

See Also

[WoodstocksIMSForm Class](#)

[Dispose Overload](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.EnableImport Method

Enables import option of the View.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void EnableImport()
```

VB

```
Public Sub EnableImport
```

C++

```
public:  
virtual void EnableImport() sealed
```

F#

```
abstract EnableImport : unit -> unit  
override EnableImport : unit -> unit
```

Implements

[IWoodstocksIMSView.EnableImport\(\)](#)

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.ExportToys Method

Exports toy data

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void ExportToys()
```

VB

```
Protected Overridable Sub ExportToys
```

C++

```
protected:  
virtual void ExportToys()
```

F#

```
abstract ExportToys : unit -> unit  
override ExportToys : unit -> unit
```

Remarks

A progress window is created to report the progress status to the user, before invoking the controller method ExportToysAsync() to instruct the system to export data.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.GetStatus Method

Gets the status that is displayed by the Form to the user.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual string GetStatus()
```

VB

```
Protected Overridable Function GetStatus As String
```

C++

```
protected:  
virtual String^ GetStatus()
```

F#

```
abstract GetStatus : unit -> string  
override GetStatus : unit -> string
```

Return Value

Type: [String](#)

The status being displayed by the Form to the user.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.ImportToys Method

Imports data into the [WoodstocksIMS](#).

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void ImportToys()
```

VB

```
Protected Overridable Sub ImportToys
```

C++

```
protected:  
virtual void ImportToys()
```

F#

```
abstract ImportToys : unit -> unit  
override ImportToys : unit -> unit
```

Remarks

A progress window is created to report the progress status to the user, before invoking the controller method ImportToysAsync() to instruct the system to export data.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.NotifyDirectoryNotFound Method

Notifies the user that a [DirectoryNotFoundException](#) that has occurred.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected bool NotifyDirectoryNotFound(  
    DirectoryNotFoundException ex  
)
```

VB

```
Protected Function NotifyDirectoryNotFound (  
    ex As DirectoryNotFoundException  
) As Boolean
```

C++

```
protected:  
bool NotifyDirectoryNotFound(  
    DirectoryNotFoundException^ ex  
)
```

F#

```
member NotifyDirectoryNotFound :  
    ex : DirectoryNotFoundException -> bool
```

Parameters

ex

Type: [System.IO.DirectoryNotFoundException](#)

The DirectoryNotFoundException.

Return Value

Type: [Boolean](#)

True because the method notifies the user that the exception by has occurred.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.NotifyDuplicateToyException Method

Notifies the user that an duplicate toy has been added to the collection of toys.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected bool NotifyDuplicateToyException(  
    DuplicateToyException ex  
)
```

VB

```
Protected Function NotifyDuplicateToyException (  
    ex As DuplicateToyException  
) As Boolean
```

C++

```
protected:  
bool NotifyDuplicateToyException(  
    DuplicateToyException^ ex  
)
```

F#

```
member NotifyDuplicateToyException :  
    ex : DuplicateToyException -> bool
```

Parameters

ex

Type: [Woodstocks.WoodstocksIMS.Domain.DuplicateToyException](#)

Return Value

Type: [Boolean](#)

True to indicate that the method has handled the error, in this case by notifying the user.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.NotifyError Method

Notifies the user that an error has occurred.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected bool NotifyError(  
    Exception ex  
)
```

VB

```
Protected Function NotifyError (  
    ex As Exception  
) As Boolean
```

C++

```
protected:  
bool NotifyError(  
    Exception^ ex  
)
```

F#

```
member NotifyError :  
    ex : Exception -> bool
```

Parameters

ex

Type: [System.Exception](#)

The exception that has occurred.

Return Value

Type: [Boolean](#)

True, if the user has been notified of the error.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.NotifyExportCancellation Method

Notifies the user that exportation has been cancelled.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void NotifyExportCancellation()
```

VB

```
Protected Overridable Sub NotifyExportCancellation
```

C++

```
protected:  
virtual void NotifyExportCancellation()
```

F#

```
abstract NotifyExportCancellation : unit -> unit  
override NotifyExportCancellation : unit -> unit
```



See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.NotifyExportCompletion Method

Overload List

	Name	Description
	NotifyExportCompletion()	Notifies the user that exportation has completed.
	NotifyExportCompletion(Exception, Boolean)	Notifies the user that exportation has completed.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.NotifyExportCompletion Method

Notifies the user that exportation has completed.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void NotifyExportCompletion()
```

VB

```
Protected Overridable Sub NotifyExportCompletion
```

C++

```
protected:  
virtual void NotifyExportCompletion()
```

F#

```
abstract NotifyExportCompletion : unit -> unit  
override NotifyExportCompletion : unit -> unit
```

See Also

[WoodstocksIMSForm Class](#)

[NotifyExportCompletion Overload](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstockIMSForm.NotifyExportCompletion Method (Exception, Boolean)

Notifies the user that exportation has completed.

Namespace: [Woodstocks.WoodstockIMS.Presentation](#)

Assembly: WoodstockIMSLib (in WoodstockIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void NotifyExportCompletion(  
    Exception ex,  
    bool cancelled  
)
```

VB

```
Public Sub NotifyExportCompletion (  
    ex As Exception,  
    cancelled As Boolean  
)
```

C++

```
public:  
virtual void NotifyExportCompletion(  
    Exception^ ex,  
    bool cancelled  
) sealed
```

F#

```
abstract NotifyExportCompletion :  
    ex : Exception *  
    cancelled : bool -> unit  
override NotifyExportCompletion :  
    ex : Exception *  
    cancelled : bool -> unit
```

Parameters

ex

Type: [System.Exception](#)

Any exception that occurred during exportation.

cancelled

Type: [System.Boolean](#)

Whether the exportation was cancelled.

Implements

[IWoodstockIMSView.NotifyExportCompletion\(Exception, Boolean\)](#)

A Sandcastle Documented Class Library

See Also

[WoodstockIMSForm Class](#)

[NotifyExportCompletion Overload](#)

[Woodstocks.WoodstockIMS.Presentation Namespace](#)

WoodstocksIMSForm.NotifyFileNotFound Method

Notifies the user that a [FileNotFoundException](#) has occurred.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected bool NotifyFileNotFound(  
    FileNotFoundException ex  
)
```

VB

```
Protected Function NotifyFileNotFound (  
    ex As FileNotFoundException  
) As Boolean
```

C++

```
protected:  
bool NotifyFileNotFound(  
    FileNotFoundException^ ex  
)
```

F#

```
member NotifyFileNotFound :  
    ex : FileNotFoundException -> bool
```

Parameters

ex

Type: [System.IO.FileNotFoundException](#)

The FileNotFoundException that was raised by the application.

Return Value

Type: [Boolean](#)

True because the method notifies the user that the exception has occurred.

Remarks

The user is notified by displaying a message to the user.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.NotifyImportCancellation Method

Notifies the user that importation has been cancelled.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected virtual void NotifyImportCancellation()
```

VB

```
Protected Overridable Sub NotifyImportCancellation
```

C++

```
protected:  
virtual void NotifyImportCancellation()
```

F#

```
abstract NotifyImportCancellation : unit -> unit  
override NotifyImportCancellation : unit -> unit
```

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.NotifyImportCompletion Method

Notifies the user that importation has completed.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public void NotifyImportCompletion(  
    Exception ex,  
    bool cancelled,  
    IToys result  
)
```

VB

```
Public Sub NotifyImportCompletion (  
    ex As Exception,  
    cancelled As Boolean,  
    result As IToys  
)
```

C++

```
public:  
virtual void NotifyImportCompletion(  
    Exception^ ex,  
    bool cancelled,  
    IToys^ result  
) sealed
```

F#

```
abstract NotifyImportCompletion :  
    ex : Exception *  
    cancelled : bool *  
    result : IToys -> unit  
override NotifyImportCompletion :  
    ex : Exception *  
    cancelled : bool *  
    result : IToys -> unit
```

Parameters

ex

Type: [System.Exception](#)

Exception that occurred during the import operation.

cancelled

Type: [System.Boolean](#)

Indicates whether the operation was cancelled by the user.

result

Type: [Woodstocks.WoodstocksIMS.Domain.IToys](#)

The result of the import operation.

Implements

[IWoodstocksIMSView.NotifyImportCompletion\(Exception, Boolean, IToys\)](#)

Remarks

The implementation identifies whether the importation has been cancelled or whether an error has occurred during importation if it completes without being cancelled.

If the importation was cancelled the user is notified of the cancellation.

If the importation was not cancelled, but an error occurred during the importation, the user is notified of the error.

If the importation completed successfully without being cancelled then the user is notified of the successful completion of the importation.

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.NotifyUnknownException Method

Notifies the user that an unknown problem has occurred.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
protected bool NotifyUnknownException(  
    Exception ex  
)
```

VB

```
Protected Function NotifyUnknownException (  
    ex As Exception  
) As Boolean
```

C++

```
protected:  
bool NotifyUnknownException(  
    Exception^ ex  
)
```

F#

```
member NotifyUnknownException :  
    ex : Exception -> bool
```

Parameters

ex

Type: [System.Exception](#)

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for

"M:Woodstocks.WoodstocksIMS.Presentation.WoodstocksIMSForm.NotifyUnknownException(System.Exception)"]

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)

WoodstocksIMSForm.Reset Method

Resets the view.

Namespace: [Woodstocks.WoodstocksIMS.Presentation](#)

Assembly: WoodstocksIMSLib (in WoodstocksIMSLib.dll) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public virtual void Reset ( )
```

VB

```
Public Overridable Sub Reset
```

C++

```
public:  
virtual void Reset ( )
```

F#

```
abstract Reset : unit -> unit  
override Reset : unit -> unit
```

Implements

[IWoodstocksIMSView.Reset\(\)](#)

See Also














[WoodstocksIMSForm Class](#)





















[Woodstocks.WoodstocksIMS.Presentation Namespace](#)






















WoodstockslMSForm.WoodstockslMSForm Properties















The [WoodstockslMSForm](#) type exposes the following members.




















Properties






















Name	Description
 AcceptButton	Gets or sets the button on the form that is clicked when the user presses the ENTER key. (Inherited from Form .)
 AccessibilityObject	Gets the AccessibleObject assigned to the control. (Inherited from Control .)
 AccessibleDefaultActionDescription	Gets or sets the default action description of the control for use by accessibility client applications. (Inherited from Control .)
 AccessibleDescription	Gets or sets the description of the control used by accessibility client applications. (Inherited from Control .)
 AccessibleName	Gets or sets the name of the control used by accessibility client applications. (Inherited from Control .)
 AccessibleRole	Gets or sets the accessible role of the control (Inherited from Control .)
 ActiveControl	Gets or sets the active control on the container control. (Inherited from ContainerControl .)
 ActiveMdiChild	Gets the currently active multiple-document interface (MDI) child window. (Inherited from Form .)
 AllowDrop	Gets or sets a value indicating whether the control can accept data that the user drags onto it. (Inherited from Control .)
 AllowTransparency	Gets or sets a value indicating whether the opacity of the form can be adjusted. (Inherited from Form .)
 Anchor	Gets or sets the edges of the container to which a control is bound and determines how a control is resized with its parent. (Inherited from Control .)
 AutoScale	Obsolete. Gets or sets a value indicating whether the form adjusts its size to fit the height of the font used on the form and scales its controls. (Inherited from Form .)
 AutoScaleBaseSize	Gets or sets the base size used for autoscaling of the form. (Inherited from Form .)
 AutoScaleDimensions	Gets or sets the dimensions that the control was designed to. (Inherited from ContainerControl .)
 AutoScaleFactor	Gets the scaling factor between the current and design-time automatic scaling dimensions. (Inherited from ContainerControl .)
 AutoScaleMode	Gets or sets the automatic scaling mode of the control. (Inherited from ContainerControl .)


 AutoScroll	Gets or sets a value indicating whether the form enables autoscrolling. (Inherited from Form .)
 AutoScrollMargin	Gets or sets the size of the auto-scroll margin. (Inherited from ScrollableControl .)
 AutoScrollMinSize	Gets or sets the minimum size of the auto-scroll. (Inherited from ScrollableControl .)
 AutoScrollOffset	Gets or sets where this control is scrolled to in ScrollControlIntoView(Control) . (Inherited from Control .)
 AutoScrollPosition	Gets or sets the location of the auto-scroll position. (Inherited from ScrollableControl .)
 AutoSize	Resize the form according to the setting of AutoSizeMode . (Inherited from Form .)
 AutoSizeMode	Gets or sets the mode by which the form automatically resizes itself. (Inherited from Form .)
 AutoValidate	(Inherited from Form .)
 BackColor	(Inherited from Form .)
 BackgroundImage	Gets or sets the background image displayed in the control. (Inherited from Control .)
 BackgroundImageLayout	Gets or sets the background image layout as defined in the ImageLayout enumeration. (Inherited from Control .)
 BindingContext	(Inherited from ContainerControl .)
 Bottom	Gets the distance, in pixels, between the bottom edge of the control and the top edge of its container's client area. (Inherited from Control .)
 Bounds	Gets or sets the size and location of the control including its nonclient elements, in pixels, relative to the parent control. (Inherited from Control .)
 CancelButton	Gets or sets the button control that is clicked when the user presses the ESC key. (Inherited from Form .)
 CanEnableIme	Gets a value indicating whether the ImeMode property can be set to an active value, to enable IME support. (Inherited from ContainerControl .)
 CanFocus	Gets a value indicating whether the control can receive focus. (Inherited from Control .)
 CanRaiseEvents	Determines if events can be raised on the control. (Inherited from Control .)
 CanSelect	Gets a value indicating whether the control can be selected. (Inherited from Control .)
 Capture	Gets or sets a value indicating whether the control has captured the mouse. (Inherited from Control .)
















 CausesValidation	Gets or sets a value indicating whether the control causes validation to be performed on any controls that require validation when it receives focus. (Inherited from Control .)
 ClientRectangle	Gets the rectangle that represents the client area of the control. (Inherited from Control .)
 ClientSize	Gets or sets the size of the client area of the form. (Inherited from Form .)
 CompanyName	Gets the name of the company or creator of the application containing the control. (Inherited from Control .)
 Container	Gets the IContainer that contains the Component . (Inherited from Component .)
 ContainsFocus	Gets a value indicating whether the control, or one of its child controls, currently has the input focus. (Inherited from Control .)
 ContextMenu	Gets or sets the shortcut menu associated with the control. (Inherited from Control .)
 ContextMenuStrip	Gets or sets the ContextMenuStrip associated with this control. (Inherited from Control .)
 ControlBox	Gets or sets a value indicating whether a control box is displayed in the caption bar of the form. (Inherited from Form .)
 Controls	Gets the collection of controls contained within the control. (Inherited from Control .)
 Created	Gets a value indicating whether the control has been created. (Inherited from Control .)
 CreateParams	(Inherited from Form .)
 CurrentAutoScaleDimensions	Gets the current run-time dimensions of the screen. (Inherited from ContainerControl .)
 Cursor	Gets or sets the cursor that is displayed when the mouse pointer is over the control. (Inherited from Control .)
 DataBindings	Gets the data bindings for the control. (Inherited from Control .)
 DefaultCursor	Gets or sets the default cursor for the control. (Inherited from Control .)
 DefaultImeMode	Gets the default Input Method Editor (IME) mode supported by the control. (Inherited from Form .)
 DefaultMargin	Gets the space, in pixels, that is specified by default between controls. (Inherited from Control .)
 DefaultMaximumSize	Gets the length and height, in pixels, that is specified as the default maximum size of a control. (Inherited from Control .)
 DefaultMinimumSize	Gets the length and height, in pixels, that is specified as the default minimum size of a control. (Inherited from Control .)
 DefaultPadding	Gets the internal spacing, in pixels, of the contents of a control.

		(Inherited from Control .)
	DefaultSize	(Inherited from Form .)
	DesignMode	Gets a value that indicates whether the Component is currently in design mode. (Inherited from Component .)
	DesktopBounds	Gets or sets the size and location of the form on the Windows desktop. (Inherited from Form .)
	DesktopLocation	Gets or sets the location of the form on the Windows desktop. (Inherited from Form .)
	DialogResult	Gets or sets the dialog result for the form. (Inherited from Form .)
	DisplayRectangle	Gets the rectangle that represents the virtual display area of the control. (Inherited from ScrollableControl .)
	Disposing	Gets a value indicating whether the base Control class is in the process of disposing. (Inherited from Control .)
	Dock	Gets or sets which control borders are docked to its parent control and determines how a control is resized with its parent. (Inherited from Control .)
	DockPadding	Gets the dock padding settings for all edges of the control. (Inherited from ScrollableControl .)
	DoubleBuffered	Gets or sets a value indicating whether this control should redraw its surface using a secondary buffer to reduce or prevent flicker. (Inherited from Control .)
	Enabled	Gets or sets a value indicating whether the control can respond to user interaction. (Inherited from Control .)
	Events	Gets the list of event handlers that are attached to this Component . (Inherited from Component .)
	Focused	Gets a value indicating whether the control has input focus. (Inherited from Control .)
	Font	Gets or sets the font of the text displayed by the control. (Inherited from Control .)
	FontHeight	Gets or sets the height of the font of the control. (Inherited from Control .)
	ForeColor	Gets or sets the foreground color of the control. (Inherited from Control .)
	FormBorderStyle	Gets or sets the border style of the form. (Inherited from Form .)
	Handle	Gets the window handle that the control is bound to. (Inherited from Control .)
	HasChildren	Gets a value indicating whether the control contains one or more child controls. (Inherited from Control .)
	Height	Gets or sets the height of the control. (Inherited from Control .)

 HelpButton	Gets or sets a value indicating whether a Help button should be displayed in the caption box of the form. (Inherited from Form .)
 HorizontalScroll	Gets the characteristics associated with the horizontal scroll bar. (Inherited from ScrollableControl .)
 HScroll	Gets or sets a value indicating whether the horizontal scroll bar is visible. (Inherited from ScrollableControl .)
 Icon	Gets or sets the icon for the form. (Inherited from Form .)
 ImeMode	Gets or sets the Input Method Editor (IME) mode of the control. (Inherited from Control .)
 ImeModeBase	Gets or sets the IME mode of a control. (Inherited from Control .)
 InvokeRequired	Gets a value indicating whether the caller must call an invoke method when making method calls to the control because the caller is on a different thread than the one the control was created on. (Inherited from Control .)
 IsAccessible	Gets or sets a value indicating whether the control is visible to accessibility applications. (Inherited from Control .)
 IsDisposed	Gets a value indicating whether the control has been disposed of. (Inherited from Control .)
 IsHandleCreated	Gets a value indicating whether the control has a handle associated with it. (Inherited from Control .)
 IsMdiChild	Gets a value indicating whether the form is a multiple-document interface (MDI) child form. (Inherited from Form .)
 IsMdiContainer	Gets or sets a value indicating whether the form is a container for multiple-document interface (MDI) child forms. (Inherited from Form .)
 IsMirrored	Gets a value indicating whether the control is mirrored. (Inherited from Control .)
 IsRestrictedWindow	Gets a value indicating whether the form can use all windows and user input events without restriction. (Inherited from Form .)
 KeyPreview	Gets or sets a value indicating whether the form will receive key events before the event is passed to the control that has focus. (Inherited from Form .)
 LayoutEngine	Gets a cached instance of the control's layout engine. (Inherited from Control .)
 Left	Gets or sets the distance, in pixels, between the left edge of the control and the left edge of its container's client area. (Inherited from Control .)
 Location	Gets or sets the Point that represents the upper-left corner of the Form in screen coordinates. (Inherited from Form .)
 MainMenuStrip	Gets or sets the primary menu container for the form. (Inherited

		from Form .)
	Margin	Gets or sets the space between controls. (Inherited from Form .)
	MaximizeBox	Gets or sets a value indicating whether the Maximize button is displayed in the caption bar of the form. (Inherited from Form .)
	MaximizedBounds	Gets and sets the size of the form when it is maximized. (Inherited from Form .)
	MaximumSize	Gets the maximum size the form can be resized to. (Inherited from Form .)
	MdiChildren	Gets an array of forms that represent the multiple-document interface (MDI) child forms that are parented to this form. (Inherited from Form .)
	MdiParent	Gets or sets the current multiple-document interface (MDI) parent form of this form. (Inherited from Form .)
	Menu	Gets or sets the MainMenu that is displayed in the form. (Inherited from Form .)
	MergedMenu	Gets the merged menu for the form. (Inherited from Form .)
	MinimizeBox	Gets or sets a value indicating whether the Minimize button is displayed in the caption bar of the form. (Inherited from Form .)
	MinimumSize	Gets or sets the minimum size the form can be resized to. (Inherited from Form .)
	Modal	Gets a value indicating whether this form is displayed modally. (Inherited from Form .)
	Name	Gets or sets the name of the control. (Inherited from Control .)
	Opacity	Gets or sets the opacity level of the form. (Inherited from Form .)
	OwnedForms	Gets an array of Form objects that represent all forms that are owned by this form. (Inherited from Form .)
	Owner	Gets or sets the form that owns this form. (Inherited from Form .)
	Padding	Gets or sets padding within the control. (Inherited from Control .)
	Parent	Gets or sets the parent container of the control. (Inherited from Control .)
	ParentForm	Gets the form that the container control is assigned to. (Inherited from ContainerControl .)
	PreferredSize	Gets the size of a rectangular area into which the control can fit. (Inherited from Control .)
	ProductName	Gets the product name of the assembly containing the control. (Inherited from Control .)
	ProductVersion	Gets the version of the assembly containing the control.

		(Inherited from Control.)
	RecreatingHandle	Gets a value indicating whether the control is currently re-creating its handle. (Inherited from Control.)
	Region	Gets or sets the window region associated with the control. (Inherited from Control.)
	RenderRightToLeft	Obsolete. This property is now obsolete. (Inherited from Control.)
	ResizeRedraw	Gets or sets a value indicating whether the control redraws itself when resized. (Inherited from Control.)
	RestoreBounds	Gets the location and size of the form in its normal window state. (Inherited from Form.)
	Right	Gets the distance, in pixels, between the right edge of the control and the left edge of its container's client area. (Inherited from Control.)
	RightToLeft	Gets or sets a value indicating whether control's elements are aligned to support locales using right-to-left fonts. (Inherited from Control.)
	RightToLeftLayout	Gets or sets a value indicating whether right-to-left mirror placement is turned on. (Inherited from Form.)
	ScaleChildren	Gets a value that determines the scaling of child controls. (Inherited from Control.)
	ShowFocusCues	Gets a value indicating whether the control should display focus rectangles. (Inherited from Control.)
	ShowIcon	Gets or sets a value indicating whether an icon is displayed in the caption bar of the form. (Inherited from Form.)
	ShowInTaskbar	Gets or sets a value indicating whether the form is displayed in the Windows taskbar. (Inherited from Form.)
	ShowKeyboardCues	Gets a value indicating whether the user interface is in the appropriate state to show or hide keyboard accelerators. (Inherited from Control.)
	ShowWithoutActivation	Gets a value indicating whether the window will be activated when it is shown. (Inherited from Form.)
	Site	Gets or sets the site of the control. (Inherited from Control.)
	Size	Gets or sets the size of the form. (Inherited from Form.)
	SizeGripStyle	Gets or sets the style of the size grip to display in the lower-right corner of the form. (Inherited from Form.)
	StartPosition	Gets or sets the starting position of the form at run time. (Inherited from Form.)
	TabIndex	Gets or sets the tab order of the control within its container. (Inherited from Form.)

 TabStop	Gets or sets a value indicating whether the user can give the focus to this control using the TAB key. (Inherited from Form .)
 Tag	Gets or sets the object that contains data about the control. (Inherited from Control .)
 Text	(Inherited from Form .)
 Top	Gets or sets the distance, in pixels, between the top edge of the control and the top edge of its container's client area. (Inherited from Control .)
 TopLevel	Gets or sets a value indicating whether to display the form as a top-level window. (Inherited from Form .)
 TopLevelControl	Gets the parent control that is not parented by another Windows Forms control. Typically, this is the outermost Form that the control is contained in. (Inherited from Control .)
 TopMost	Gets or sets a value indicating whether the form should be displayed as a topmost form. (Inherited from Form .)
 TransparencyKey	Gets or sets the color that will represent transparent areas of the form. (Inherited from Form .)
 UseWaitCursor	Gets or sets a value indicating whether to use the wait cursor for the current control and all child controls. (Inherited from Control .)
 VerticalScroll	Gets the characteristics associated with the vertical scroll bar. (Inherited from ScrollableControl .)
 Visible	Gets or sets a value indicating whether the control and all its child controls are displayed. (Inherited from Control .)
 VScroll	Gets or sets a value indicating whether the vertical scroll bar is visible. (Inherited from ScrollableControl .)
 Width	Gets or sets the width of the control. (Inherited from Control .)
 WindowState	Gets or sets a value that indicates whether form is minimized, maximized, or normal. (Inherited from Form .)
 WindowTarget	This property is not relevant for this class. (Inherited from Control .)

See Also

[WoodstocksIMSForm Class](#)


[Woodstocks.WoodstocksIMS.Presentation Namespace](#)











WoodstockslMSForm.WoodstockslMSForm Events






















The [WoodstockslMSForm](#) type exposes the following members.







Events

Name	Description
 Activated	Occurs when the form is activated in code or by the user. (Inherited from Form .)
 AutoSizeChanged	Occurs when the AutoSize property changes. (Inherited from Form .)
 AutoValidateChanged	Occurs when the AutoValidate property changes. (Inherited from Form .)
 BackColorChanged	Occurs when the value of the BackColor property changes. (Inherited from Control .)
 BackgroundImageChanged	Occurs when the value of the BackgroundImage property changes. (Inherited from Control .)
 BackgroundImageLayoutChanged	Occurs when the BackgroundImageLayout property changes. (Inherited from Control .)
 BindingContextChanged	Occurs when the value of the BindingContext property changes. (Inherited from Control .)
 CausesValidationChanged	Occurs when the value of the CausesValidation property changes. (Inherited from Control .)
 ChangeUICues	Occurs when the focus or keyboard user interface (UI) cues change. (Inherited from Control .)
 Click	Occurs when the control is clicked. (Inherited from Control .)
 ClientSizeChanged	Occurs when the value of the ClientSize property changes. (Inherited from Control .)
 Closed	Occurs when the form is closed. (Inherited from Form .)
 Closing	Occurs when the form is closing. (Inherited from Form .)
 ContextMenuChanged	Occurs when the value of the ContextMenu property changes. (Inherited from Control .)
 ContextMenuStripChanged	Occurs when the value of the ContextMenuStrip property changes. (Inherited from Control .)
 ControlAdded	Occurs when a new control is added to the Control.ControlCollection . (Inherited from Control .)
 ControlRemoved	Occurs when a control is removed from the Control.ControlCollection . (Inherited from Control .)
 CursorChanged	Occurs when the value of the Cursor property changes. (Inherited from Control .)
 Deactivate	Occurs when the form loses focus and is no longer the active form. (Inherited from Form .)

 Disposed	Occurs when the component is disposed by a call to the Dispose() method. (Inherited from Component .)
 DockChanged	Occurs when the value of the Dock property changes. (Inherited from Control .)
 DoubleClick	Occurs when the control is double-clicked. (Inherited from Control .)
 DragDrop	Occurs when a drag-and-drop operation is completed. (Inherited from Control .)
 DragEnter	Occurs when an object is dragged into the control's bounds. (Inherited from Control .)
 DragLeave	Occurs when an object is dragged out of the control's bounds. (Inherited from Control .)
 DragOver	Occurs when an object is dragged over the control's bounds. (Inherited from Control .)
 EnabledChanged	Occurs when the Enabled property value has changed. (Inherited from Control .)
 Enter	Occurs when the control is entered. (Inherited from Control .)
 FontChanged	Occurs when the Font property value changes. (Inherited from Control .)
 ForeColorChanged	Occurs when the ForeColor property value changes. (Inherited from Control .)
 FormClosed	Occurs after the form is closed. (Inherited from Form .)
 FormClosing	Occurs before the form is closed. (Inherited from Form .)
 GiveFeedback	Occurs during a drag operation. (Inherited from Control .)
 GotFocus	Occurs when the control receives focus. (Inherited from Control .)
 HandleCreated	Occurs when a handle is created for the control. (Inherited from Control .)
 HandleDestroyed	Occurs when the control's handle is in the process of being destroyed. (Inherited from Control .)
 HelpButtonClicked	Occurs when the Help button is clicked. (Inherited from Form .)
 HelpRequested	Occurs when the user requests help for a control. (Inherited from Control .)
 ImeModeChanged	Occurs when the ImeMode property has changed. (Inherited from Control .)
 InputLanguageChanged	Occurs after the input language of the form has changed. (Inherited from Form .)
 InputLanguageChanging	Occurs when the user attempts to change the input language for the form. (Inherited from Form .)
 Invalidated	Occurs when a control's display requires redrawing. (Inherited from

	Control.)
 KeyDown	Occurs when a key is pressed while the control has focus. (Inherited from Control.)
 KeyPress	Occurs when a key is pressed while the control has focus. (Inherited from Control.)
 KeyUp	Occurs when a key is released while the control has focus. (Inherited from Control.)
 Layout	Occurs when a control should reposition its child controls. (Inherited from Control.)
 Leave	Occurs when the input focus leaves the control. (Inherited from Control.)
 Load	Occurs before a form is displayed for the first time. (Inherited from Form.)
 LocationChanged	Occurs when the Location property value has changed. (Inherited from Control.)
 LostFocus	Occurs when the control loses focus. (Inherited from Control.)
 MarginChanged	Occurs when the Margin property changes. (Inherited from Form.)
 MaximizedBoundsChanged	Occurs when the value of the MaximizedBounds property has changed. (Inherited from Form.)
 MaximumSizeChanged	Occurs when the value of the MaximumSize property has changed. (Inherited from Form.)
 MdiChildActivate	Occurs when a multiple-document interface (MDI) child form is activated or closed within an MDI application. (Inherited from Form.)
 MenuComplete	Occurs when the menu of a form loses focus. (Inherited from Form.)
 MenuStart	Occurs when the menu of a form receives focus. (Inherited from Form.)
 MinimumSizeChanged	Occurs when the value of the MinimumSize property has changed. (Inherited from Form.)
 MouseCaptureChanged	Occurs when the control loses mouse capture. (Inherited from Control.)
 MouseClicked	Occurs when the control is clicked by the mouse. (Inherited from Control.)
 MouseDoubleClick	Occurs when the control is double clicked by the mouse. (Inherited from Control.)
 MouseDown	Occurs when the mouse pointer is over the control and a mouse button is pressed. (Inherited from Control.)
 MouseEnter	Occurs when the mouse pointer enters the control. (Inherited from Control.)

 MouseHover	Occurs when the mouse pointer rests on the control. (Inherited from Control .)
 MouseLeave	Occurs when the mouse pointer leaves the control. (Inherited from Control .)
 MouseMove	Occurs when the mouse pointer is moved over the control. (Inherited from Control .)
 MouseUp	Occurs when the mouse pointer is over the control and a mouse button is released. (Inherited from Control .)
 MouseWheel	Occurs when the mouse wheel moves while the control has focus. (Inherited from Control .)
 Move	Occurs when the control is moved. (Inherited from Control .)
 PaddingChanged	Occurs when the control's padding changes. (Inherited from Control .)
 Paint	Occurs when the control is redrawn. (Inherited from Control .)
 ParentChanged	Occurs when the Parent property value changes. (Inherited from Control .)
 PreviewKeyDown	Occurs before the KeyDown event when a key is pressed while focus is on this control. (Inherited from Control .)
 QueryAccessibilityHelp	Occurs when AccessibleObject is providing help to accessibility applications. (Inherited from Control .)
 QueryContinueDrag	Occurs during a drag-and-drop operation and enables the drag source to determine whether the drag-and-drop operation should be canceled. (Inherited from Control .)
 RegionChanged	Occurs when the value of the Region property changes. (Inherited from Control .)
 Resize	Occurs when the control is resized. (Inherited from Control .)
 ResizeBegin	Occurs when a form enters resizing mode. (Inherited from Form .)
 ResizeEnd	Occurs when a form exits resizing mode. (Inherited from Form .)
 RightToLeftChanged	Occurs when the RightToLeft property value changes. (Inherited from Control .)
 RightToLeftLayoutChanged	Occurs after the value of the RightToLeftLayout property changes. (Inherited from Form .)
 Scroll	Occurs when the user or code scrolls through the client area. (Inherited from ScrollableControl .)
 Shown	Occurs whenever the form is first displayed. (Inherited from Form .)
 SizeChanged	Occurs when the Size property value changes. (Inherited from Control .)
 StyleChanged	Occurs when the control style changes. (Inherited from Control .)
 SystemColorsChanged	Occurs when the system colors change. (Inherited from Control .)

 TabIndexChanged	Occurs when the value of the TabIndex property changes. (Inherited from Form .)
 TabStopChanged	Occurs when the TabStop property changes. (Inherited from Form .)
 TextChanged	Occurs when the Text property value changes. (Inherited from Control .)
 Validated	Occurs when the control is finished validating. (Inherited from Control .)
 Validating	Occurs when the control is validating. (Inherited from Control .)
 VisibleChanged	Occurs when the Visible property value changes. (Inherited from Control .)

See Also

[WoodstocksIMSForm Class](#)

[Woodstocks.WoodstocksIMS.Presentation Namespace](#)