Advanced .NET Framework 2.0 Using C# 2005: Part 3 of 3

Meet the expert: Jan Fransen has been working to educate people in the best ways to use Microsoft products for more than 15 years. She's written both data-based and document-based applications for large and small companies and has helped hundreds of Office users and developers get started on the right foot through public and onsite training classes. As a writer, Jan has developed training courseware, contributed to books on Microsoft Office, written white papers for publication on MSDN, and created samples designed to help developers get up to speed quickly on new Microsoft products and features.

F. Scott Barker has over 15 years of experience as a software developer in Visual Basic .NET and most recently in C#. He has been a Microsoft MVP and worked at Microsoft as a member of the Access and FoxPro teams. Scott is a seasoned trainer and frequent speaker at Microsoft conferences worldwide. He has also been a contributing editor and writer for various technical magazines and has written several software development books.

Runtime: 08:19:37

Course description: In this volume, you'll learn about the .NET Framework 2.0 system types like Generic and Reference types. Work with groups of data using various classes like Collection, Hashtable, ArrayList, and Dictionary. Improve type safety and performance using generic collections and manage data better by using specialized collections. You'll learn about object serialization and working with XML formatted data. Manage files and folders using the System.IO namespace. See how to work with external data using the Stream classes.

Course outline:

.NET Framework Class Library
- Introduction
- Overview
- System Types
- Value Type / Ref. Types
- Built-in Value Types
- Number Types
- Demo: Numeric Types
- ToString Method
- Demo: Numeric ToString
- Instance vs. Static
- Nullable Type
- Demo: Numeric Static Methods
- Demo: Nullable Type
- Dates and Times
- Demo: Date/Time Properties
- Demo: Date/Time ToString
- Date/Time Instance Methods
- Date/Time Static Methods
- Summary

Reference Types
- Introduction
- Work w/Reference Types
- Copy a Value Type
- Copy a Reference Type
- Work with Strings
- Stringbuilder Class
- Demo: ReferenceTypes
- Demo: String Formatting
- Demo: Concatenate Strings
- Demo: Stringbuilder
- Overview: Arrays
- Array Methods
- Iterating Arrays
- Demo: Work with Arrays
- Handling Exceptions
- Demo: Work w/Exceptions
- Summary

Conversion
- Introduction
- Implicit Conversions
- Convert with Casts
- Convert w/.NET Framework
- Boxing / Unboxing
- Demo: Wide/Narrow Convert
- Demo: ToString
- Demo: Parse/TryParse
- Summary

Custom Types
- Introduction
- Overview: Custom Types
- Value vs Ref. Type
- Constants
- Enumerations
- Demo: Constants
- Demo: Enumerations
- More Complex Enum.
- Structures
- Constructors
- Properties
- Add Methods to Structure
- Property vs Method
- Demo: Simple Structure
- More Complex Structure
- Summary

Creating Classes
- Introduction
- Overview
- Class Constructors
- Properties / Methods
- Demo: A Simple Class
- Inherited Members
- Overloading Methods
- Instance / Shared members
- ApplicationException Class
- Handle Exceptions
- Demo: Exceptions
- Demo: Static Property
- Demo: Static Method
- Summary

Advanced Custom Types
- Introduction
- Using Attributes
- Built-in Attributes
- TypeForwardedToAttribute
- Build Your Attributes
- Implement Interfaces
- Use IComparable
- Use IDisposable
- Use IConvertible
- Use ICloneable
- Use IEquatable
- Use IFormattable
- Create Custom Interfaces
- View Custom Interface
- Summary

Delegates and Events
- Introduction
- Overview: Delegates
- Demo: Delegates
- Overview: Events
- Custom Event Handler

(Continued on page 2)

Advanced .NET Framework 2.0 Using C# 2005: Part 3 of 3

- Generic Classes
- Built-in Generic Classes
- Create Generic Types
- Consume Generic Type
- Add Constraints
- Demo: Generics/Constraints
- Summary

Simple Collections
- Introduction
- System.Collections
- Overview: ArrayList
- Demo: ArrayList
- Common Functionality
- IEnumerable Interface
- IEnumerator Interface
- ICollection Interface
- IList Interface
- CollectionBase Class
- ReadOnlyCollectionBase
- Implement Custom Collection
- Sequential Lists
- Queue Class
- Stack Class
- Demo: Queue
- Demo: Stack
- Demo: Queue and Stack
- Summary

Dictionaries and Sorting
- Introduction
- Dictionary Classes
- Handle Key/Value Pairs
- Hashtable
- GetEnumerator
- Determine Duplicates
- IEqualityComparer
- Demo: Hashtable
- Demo: IEqualityComparer
- SortedList
- Sort Collections
- Comparer Class
- Implement ICompare
- Demo: Sorting Lists
- Summary

Generic Collections
- Introduction
- Generic Collections
- Generic List Class
- Use Predicates
- Demo: Array List Issues
- Demo: Generic List
- Demo: Iterate Generic List
- Demo: Find / Predicates
- Summary

Additional Generic Collections
- Introduction
- Generic Dictionary Class
- Sorted Classes
- Queues and Stacks
- Generic Queue Class
- Generic Stack Class
- Demo: Generic Dictionary
- Demo: Generic SortedList
- Demo: Generic Queue
- Demo: Generic Stack
- Generic LinkedList
- LinkedList Methods
- LinkedListNode Class
- Demo: Generic LinkedList
- Summary

Specialized Collections
- Introduction
- Generic Collection Classes
- Generic Collection Enums
- Demo: Gen Collections
- Specialized Collections
- StringCollection
- StringDictionary
- NameValueCollection
- Demo: StringCollection
- Demo: StringDictionary
- Demo: NameValueCollection
- BitArray
- BitVector32
- Demo: BitArray
- Demo: BitVector32
- Summary

Serializing Objects
- Introduction
- Overview: Serialization
- Serialization Formats
- Start with Serialization
- Specify a Formatter
- Steps to Serialize
- Demo: Serialize Strings
- Serialize a String
- Steps to Deserialize
- Serialize Complex Types
- Deserialize Complex Types
- Catch Exceptions
- Demo: Deserialize Strings
- Demo: Complex Types
- Summary

Serializing Custom Objects
- Introduction
- Serialize Custom Class
- Deserialize Custom Class
- Serialize Custom Colls
- Deserialize Custom Colls.
- Demo: Custom Class
- Demo: Custom Collections
- Formatting for SOAP
- Version Compatibility
- OptionalField Attribute
- Make Non-Serializable
- Deserialization Callback
- Demo: Use SOAP Formatter
- Demo: Non-Serialized
- Serialization Tips
- Summary

Serialize Objects with XML
- Introduction
- XML Serialization
- Simple XML Serialization
- Steps to Serialize XML
- Steps to Deserialize XML
- Serialize a String
- Deserialize a String
- Complex Data Types
- Catching Exceptions
- Serialize Complex Data
- Deserialize Complex Data
- Summary

Serialize Classes with XML
- Introduction
- Serialize Classes
- Deserialize Classes
- Serialize Collections
- Deserialize Collections
- Demo: Classes
- Demo: Collections
- Summary

Datasets and XML
- Introduction
- Overview
- Write Datasets
- Load Datasets
- Demo: Save Dataset
- Demo: Load Dataset
- Use XML Schemas
- Create Matching Class
- XML Schema Definition Tool
- Demo: Schema Def Tool
- Custom Serialization
- Implement ISerializable
- Serializing Methods
- SerializationInfo
- Custom Deserialization
- Serialization Events
- Demo: Custom Serialization
- Demo: Custom Deserialization
- Summary

Files and Folders
- Introduction
- Overview
- Work with Drives
- Demo: Drives
- Directories and Folders
- Demo: Directories
- FileSystemWatcher
- Files and Paths
- Monitor File System
- Demo: File/FileInfo Class
- Demo: FileSystemWatcher
- Parse with Path Class
- Demo: Path Class
- Demo: File and FileInfo
- Summary

Text Streams
- Introduction
- TextReader/TextWriter
- StringReader
- StringWriter
- Demo: StringReader/Writer
- Read From/Write To
- Stream Base Class
- FileStream Class
- Demo: FileStream Class
- StreamReader/StreamWriter
- BinaryReader/BinaryWriter
- Demo: StreamReader/Writer
- Demo: BinaryReader/Writer
- MemoryStream
- BufferedStream
- Demo: MemoryStream
- Demo: BufferedStream
- Summary

Compress / Decompress
- Introduction

(Continued on page 3)

Advanced .NET Framework 2.0 Using C# 2005: Part 3 of 3

page 3

• Overview
• GZipStream/DeflateStream
• Steps to Compress
• Demo: Compress File
• Steps to Decompress
• Demo: Decompress File
• Summary

Isolated Storage (IS)
• Introduction
• Overview: IS
• IsolatedStorageFile Class
• Retrieve IS for Machine
• Work with Folders
• Demo: Work with Folders
• Work with Files
• Demo: Work with Files
• Demo: Create/Store Files
• Demo: Open/Read Files
• Access from Other Apps
• IS File Permission Attrib
• Demo: Compress File in IS
• Summary