



CS 2530

XML

LINQ to XML

- LINQ to XML is a LINQ-enabled, in-memory XML programming interface that enables you to work with XML from within the .NET Framework programming languages.
- Namespace: `System.Xml.Linq`

LINQ to XML

- Similar to DOM:
 - Brings XML document into memory
 - You can query and modify the document,
 - You can save it to a file
- Different from DOM:
 - Different object model that is lighter weight and easier to work with

LINQ to XML

- Examples of what you can do using LINQ to XML:
 - Load XML from files or stream
 - Serialize XML to files or streams
 - Create XML trees
 - Manipulate the in-memory XML tree

Class inheritance

- XNode

add, delete, transform nodes

- XComment

- XContainer (abstract base class)

find previous or next sibling, parent, descendants, ..

- XDocument

- XElement (has property Value)

- XText

- . . .

XNode

- Represents concept of a node
e.g. element, comment, text

XDocument

Represents an XML document

- Save .. saves XDocument to file, stream, ..
- Load .. loads XDocument from file, stream, ..
- Descendants .. returns child nodes in document order.
when passed a name, it filters the list.

Uses [deferred execution](#)

XElement

- Represents an XML element
- It has:
 - XName
 - attributes (optional)
 - content (optional)
- Property Value gets / sets the text content (string)

XAttribute

- Name / value pair associated with an XML element
- Additional information
- Not a node in the tree
- Attributes have a name unique to element

Creating XML trees:

- Pass contents of the element or attribute as arguments to the constructor
- Functional Construction:

Construct new XML trees by using query results as parameters to XElement and XAttribute constructors

Manipulating XML trees:

- Manipulate the in-memory XML tree similar to DOM manipulation; for small changes in large XML trees using Add, Remove, ReplaceWith, . . .
- Functional Construction
 - Often faster to write, more robust, easier to maintain
 - Construct new XML trees by using query results as parameters to XElement and XAttribute constructors

Manipulating XML trees:

- Manipulate the in-memory XML tree similar to DOM manipulation;
for small changes in large XML trees
using Add, Remove, ReplaceWith, SetValue
- Functional Construction:
Often faster to write, more robust, easier to maintain
Construct new XML trees by using query results as
parameters to XElement and XAttribute constructors