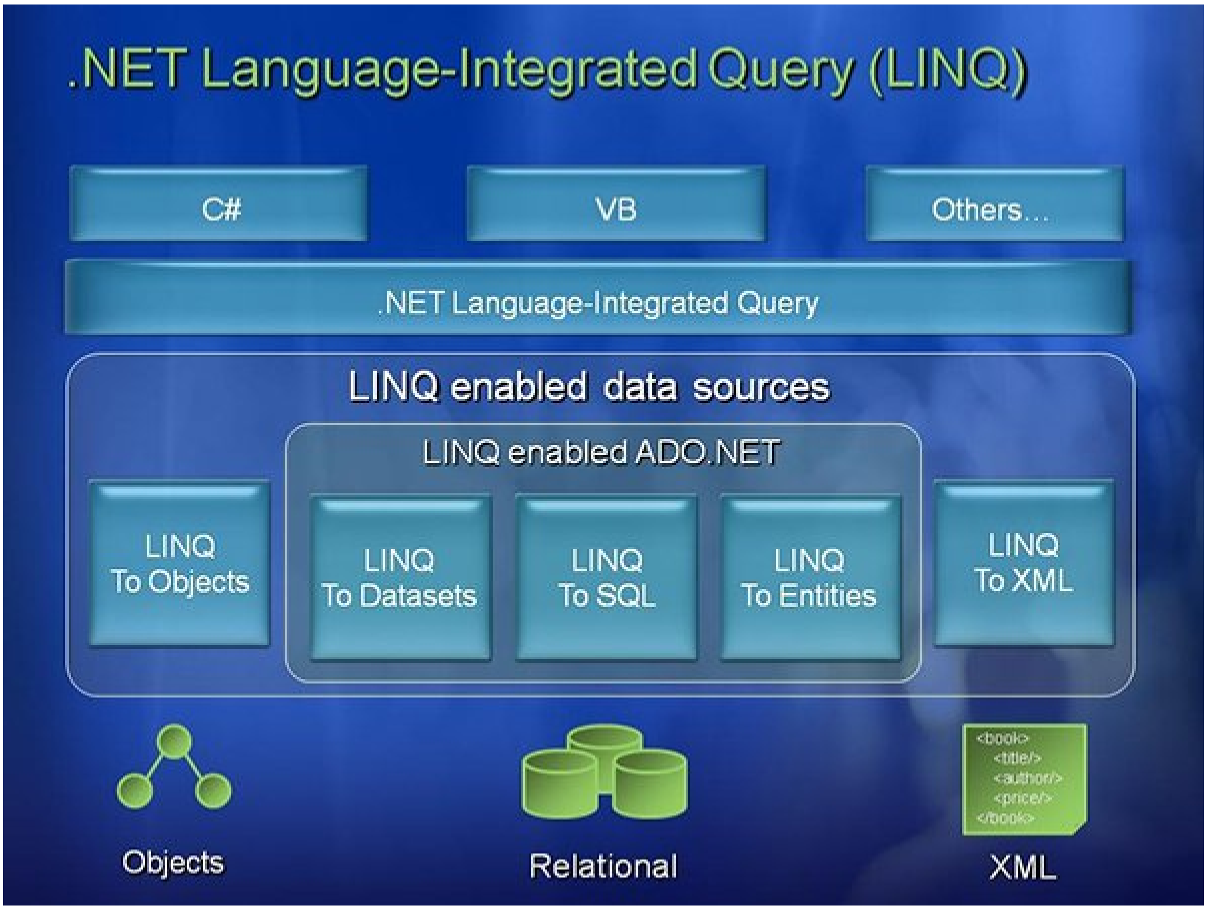
禪

LINQ [illegible]

[illegible]) [illegible] (LINQ



[illegible] XML [illegible] XML [illegible] XML [illegible] XML [illegible] XML [illegible].

[illegible] XML [illegible]

XElements [illegible] XDocuments [illegible] Parse() [illegible] XML [illegible] XElements [illegible] Load() [illegible] XML [illegible]. [illegible]. [illegible] 1 [illegible].

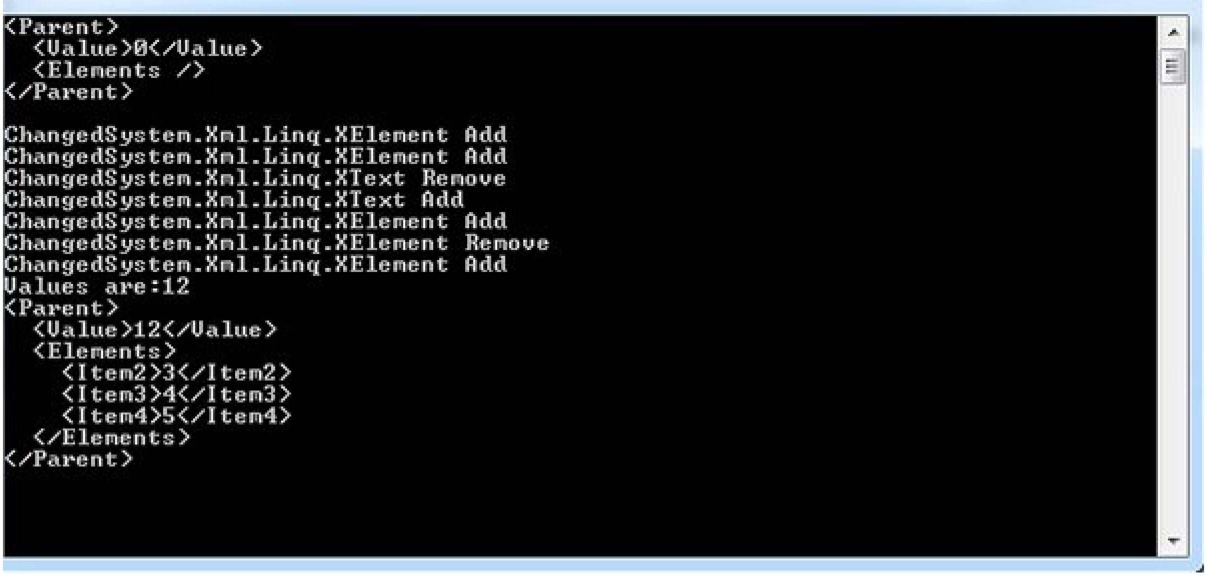
```
static void Main(string[] args)
{
```

```
            string myElement =
            @"<Language ID ='1'>
<HighLevel>Yes</HighLevel>
<Programming>CSharp</Programming>
</Language>";
            XElement newElement = XElement.Parse(myElement);
            Console.WriteLine(newElement);
            Console.WriteLine();
XDocument Doc = XDocument.Load("myxml.xml");
            Console.WriteLine(Doc);
        }
```

[illegible] XML

[illegible] XML [illegible] XObject [illegible] XObject [illegible] 2 [illegible] XML [illegible] 1 [illegible] 2 [illegible].

```
        static void Main(string[] args)
        {
            XElement root = new XElement("Parent",
    new XElement("Value", "0"),
    new XElement("Elements"));
  XElement total = root.Element("Value");
 XElement items = root.Element("Elements");
            items.Changed += (object sender, XObjectChangeEventArgs objectchangeevent) =>
            {
                switch (objectchangeevent.ObjectChange)
                {
                    case XObjectChange.Add:
                        if (sender is XElement)
 total.Value = ((int)total + (int)(XElement)sender).ToString();
                        if (sender is XText)
 total.Value = ((int)total + (int)((XText)sender).Parent).ToString();
                        break;
                    case XObjectChange.Remove:
                        if (sender is XElement)
  total.Value = ((int)total - (int)(XElement)sender).ToString();
                        if (sender is XText)
total.Value = ((int)total - Int32.Parse(((XText)sender).Value)).ToString();
                        break;
                }
                Console.WriteLine("Changed{0}{1}",sender.GetType().ToString(),
objectchangeevent.ObjectChange.ToString());
            };
            Console.WriteLine(root);
            Console.ReadLine();
            items.SetElementValue("Item1", 1);
            items.SetElementValue(“Item2”, 2);
            items.SetElementValue(“Item2”, 3);
            items.SetElementValue(“Item3”, 4);
           items.SetElementValue(“Item1”, null);
            items.SetElementValue(“Item4”, 5);
Console.WriteLine(“Values are:{0}”, (int)total);
            Console.WriteLine(root);
            Console.ReadLine();
        }
```



```
<Parent>
  <Value>0</Value>
  <Elements />
</Parent>

ChangedSystem.Xml.Linq.XElement Add
ChangedSystem.Xml.Linq.XElement Add
ChangedSystem.Xml.Linq.XText Remove
ChangedSystem.Xml.Linq.XText Add
ChangedSystem.Xml.Linq.XElement Add
ChangedSystem.Xml.Linq.XElement Remove
ChangedSystem.Xml.Linq.XElement Add
Values are:12
<Parent>
  <Value>12</Value>
  <Elements>
    <Item2>3</Item2>
    <Item3>4</Item3>
    <Item4>5</Item4>
  </Elements>
</Parent>
```

[illegible] 1 - [illegible].

[illegible] XML

[illegible] XElement [illegible] System.Xml.Linq.XElementSave [illegible] XDocument [illegible] System.Xml.Linq.XDocument.Save [illegible] XML [illegible]. [illegible] XmlWriter [illegible] XML [illegible]. [illegible] Save [illegible]. [illegible] Save [illegible].

Save [illegible] XDocument [illegible] TextWriter [illegible] XmlWriter [illegible].

Save [illegible] XElement [illegible] TextWriter، XMLTextWriter [illegible] XmlWriter [illegible]. [illegible] 3 [illegible] XElement [illegible].

```
using System.Xml.Linq;
using System.IO;
        static void Main(string[] args)
        {
            XElement root = new XElement(“Parent”,
      new XElement(“Child”, “child content”) );
            root.Save(“MyXml.xml”);
            string str = File.ReadAllText(“MyXml.xml”);
            Console.WriteLine(str);
        }
```

[illegible] XML

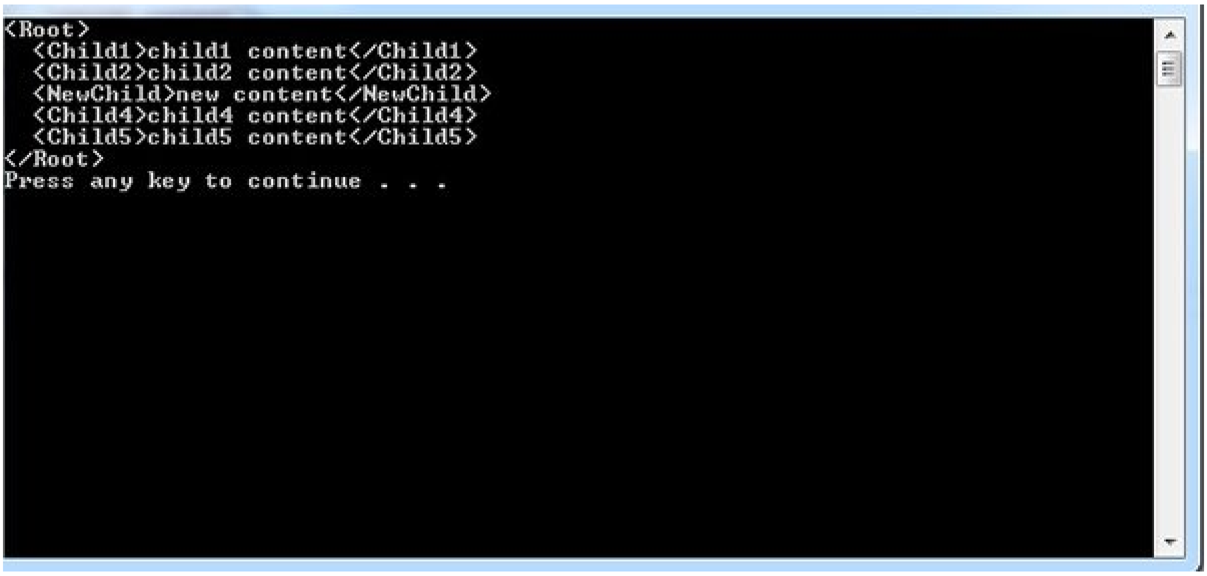
[illegible] XML [illegible]. [illegible] XML [illegible]. [illegible] XmlWriterSettings [illegible] OmitXmlDeclaration [illegible] true [illegible]. [illegible] true [illegible] XML [illegible]. [illegible] Create [illegible] XmlWriter [illegible]. [illegible] StringBuiler [illegible] XmlSettings [illegible] XmlWriter [illegible]. [illegible] XML [illegible] XmlWriter [illegible] Save [illegible]. [illegible] 4 [illegible].

```
        static void Main(string[] args)
        {
   StringBuilder sb = new StringBuilder();
           XmlWriterSettings xmlwritersettings = new XmlWriterSettings();
 xmlwritersettings.OmitXmlDeclaration = true;

           using (XmlWriter xmlwriter = XmlWriter.Create(sb, xmlwritersettings))
           {
            XElement root = new XElement("Parent",new XElement("Child", "child content"));
              root.Save(xmlwriter);
           }
           Console.WriteLine(sb.ToString());
        }
        }
```

[illegible] XElement [illegible]

[illegible] XElement [illegible] RemoveAll() [illegible]. [illegible] 5 [illegible]. [illegible]. [illegible] </ Root> [illegible]. [illegible] 5:

```
static void Main(string[] args)
        {
XElement root = new XElement("Root",
   new XAttribute("Att1", 1),
   new XAttribute("Att2", 2),
   new XAttribute("Att3", 3),
   new XElement("Child1", 1),
   new XElement("Child2", 2),
   new XElement("Child3", 3)
);
root.RemoveAll();
Console.WriteLine(root);
        }
```



```
<Root>
  <Child1>child1 content</Child1>
  <Child2>child2 content</Child2>
  <NewChild>new content</NewChild>
  <Child4>child4 content</Child4>
  <Child5>child5 content</Child5>
</Root>
Press any key to continue . . .
```

[illegible] 2 - [illegible]

[illegible]

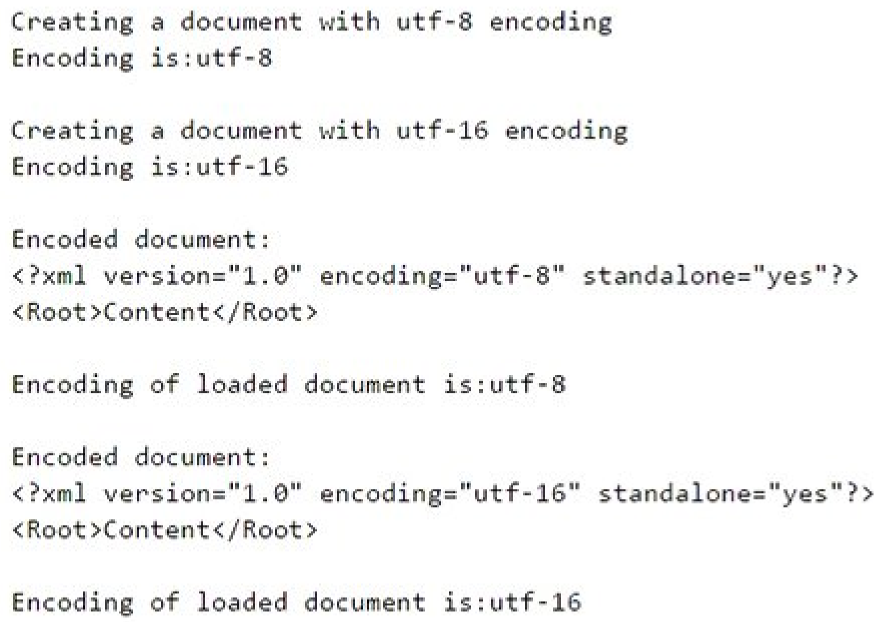
[illegible]. [illegible] ReplaceWith [illegible] XNode [illegible]. 6 [illegible]. 2 [illegible] 6 [illegible].

```
static void Main(string[] args)
        {
    XElement xmlTree = new XElement("Root",
    new XElement("Child1", "child1 content"),
    new XElement("Child2", "child2 content"),
    new XElement("Child3", "child3 content"),
    new XElement("Child4", "child4 content"),
    new XElement("Child5", "child5 content")

);
XElement child3 = xmlTree.Element("Child3");
child3.ReplaceWith(
    new XElement("NewChild", "new content")
);
Console.WriteLine(xmlTree);
        }
```

[illegible]

[illegible] XML [illegible] XDeclaration [illegible] XML [illegible] (Encoding) [illegible]. [illegible] 7 [illegible]. utf-8 [illegible] utf-16 [illegible]. [illegible]. [illegible] Encoding [illegible] XML [illegible]. 3 [illegible] 7 [illegible].

```
Console.WriteLine("Creating a document with utf-8 encoding");
XDocument encodedDoc8 = new XDocument(
    new XDeclaration("1.0", "utf-8", "yes"),
    new XElement("Root", "Content")
);
encodedDoc8.Save(“EncodedUtf8.xml”);
Console.WriteLine(“Encoding is:{0}”, encodedDoc8.Declaration.Encoding);
Console.WriteLine();

Console.WriteLine(“Creating a document with utf-16 encoding”);
XDocument encodedDoc16 = new XDocument(
    new XDeclaration(“1.0”, “utf-16”, “yes”),
    new XElement(“Root”, “Content”)
);
encodedDoc16.Save(“EncodedUtf16.xml”);
Console.WriteLine(“Encoding is:{0}”, encodedDoc16.Declaration.Encoding);
Console.WriteLine();
  XDocument newDoc8 = XDocument.Load(“EncodedUtf8.xml”);
Console.WriteLine(“Encoded document:”);
Console.WriteLine(File.ReadAllText(“EncodedUtf8.xml”));
Console.WriteLine();
Console.WriteLine(“Encoding of loaded document is:{0}”, newDoc8.Declaration.Encoding);
Console.WriteLine();
  XDocument newDoc16 = XDocument.Load(“EncodedUtf16.xml”);
Console.WriteLine(“Encoded document:”);
Console.WriteLine(File.ReadAllText(“EncodedUtf16.xml”));
Console.WriteLine();
Console.WriteLine(“Encoding of loaded document is:{0}”, newDoc16.Declaration.Encoding);
```



```
Creating a document with utf-8 encoding
Encoding is:utf-8

Creating a document with utf-16 encoding
Encoding is:utf-16

Encoded document:
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<Root>Content</Root>

Encoding of loaded document is:utf-8

Encoded document:
<?xml version="1.0" encoding="utf-16" standalone="yes"?>
<Root>Content</Root>

Encoding of loaded document is:utf-16
```

[illegible] 3 - [illegible] *Encoding* [illegible].

XML [illegible] XSLT [illegible]

XSLT ([illegible] Extensible Stylesheet Language Transformations) [illegible] XML [illegible]. [illegible]. XSLT [illegible] XML [illegible] XML [illegible] HTML [illegible]. [illegible] XML [illegible] XmlReader [illegible] createReader [illegible] XML [illegible]

XmlWriter [illegible]. [illegible] XSLT [illegible] XML [illegible]. 8 [illegible].

```
        static void Main(string[] args)
        {
            string xslstring= @”<?xml version=’1.0’?>
<xsl:stylesheet xmlns:xsl=’http://www.w3.org/1999/XSL/Transform’ version=’1.0’>
    <xsl:template match=’/Root’>
        <Parent>
            <A1>
            <xsl:value-of select=’Child1’/>
            </A1>
            <A2>
            <xsl:value-of select=’Child2’/>
            </A2>
        </Parent>
    </xsl:template>
</xsl:stylesheet>”;

            XDocument xmlTree = new XDocument(
                new XElement(“Root”,
    new XElement(“Item1”, “Item1 data”),
       new XElement(“Item2”, “Item2 data”)));

            XDocument newTree = new XDocument();
            using (XmlWriter writer = newTree.CreateWriter())
            {
                XslCompiledTransform xslt = new XslCompiledTransform();
                xslt.Load(XmlReader.Create(new StringReader(xslstring)));
                xslt.Transform(xmlTree.CreateReader(), writer);
            }
            Console.WriteLine(newTree);
        }
```

XslCompiledTransform [illegible] XML [illegible]. [illegible] XSLT [illegible]. Load [illegible] XslCompiledTransform [illegible] Create [illegible]. [illegible] Load [illegible] xslCompiledTransform [illegible] Load [illegible] XslTransform [illegible] XLST [illegible].

[illegible]:

[illegible]

[illegible]:

[illegible]

[illegible]:

11:28 - 17/03/1396

[illegible]:

LINQ - [illegible] - [illegible] - [illegible] - [illegible] - [illegible] - [illegible]

[illegible]

[illegible]: https://www.shabakeh-mag.com/workshop/7998/%D8%AA%DA%A9%D9%86%DB%8C%DA%A9%E2%80%8C%D9%87%D8%A7%DB%8C-%DA%A9%D8%A7%D8%B1%D8%A8%D8%B1%D8%AF%DB%8C-%D8%A7%D8%B3%D8%AA%D8%AE%D8%B1%D8%A7%D8%AC-%D8%AF%D8%A7%D8%AF%D9%87%E2%80%8C%D9%87%D8%A7-%D8%A8%D8%A7-%D9%84%DB%8C%D9%86%DA%A9-%D8%A8%D8%AE%D8%B4-%D9%BE%D9%86%D8%AC%D9%85