



## ข้อดี

- พัฒนาระบบงานได้เร็วขึ้น
- ง่ายขึ้น
- ส่งมอบงานเร็วขึ้น
- ค่าใช้จ่ายน้อยลง
- เรียก service ที่พัฒนาแล้วมาใช้ใหม่
- การ implement service ถ้ามีโค้ดอนุญาตแล้วก็ใช่องเดิมมาใช้งานโดยไม่ต้องเขียนโค้ด

## Service Reuse

## ข้อเสีย

- ต้องปรับปรุงขั้นตอนการพัฒนาให้ดีด้วยการทำงานใหม่น้อยลง
- เขียน code น้อยลง
- ถ้าระบบงานซับซ้อนมากๆ อาจจะทำได้ลำบาก





01



Display XML บนเว็บ

Create XML



## STEP 01 สร้างไฟล์ชื่อ test.xml และบันทึกไว้ในโฟล์เดอร์ App\_Data

```
<?xml version="1.0" encoding="utf-8" ?>
<mydatabase>
  <customer>
    <CustomerID>C001</CustomerID>
    <Name>Nipat Jongsawat</Name>
    <Email>nipatj@yahoo.com</Email>
    <CountryCode>TH</CountryCode>
    <Budget>1,000,000</Budget>
    <Used>600,000</Used>
  </customer>
  <customer>
    <CustomerID>C002</CustomerID>
    <Name>Wichian premchaiswadi</Name>
    <Email>wichian@siam.edu</Email>
    <CountryCode>EN</CountryCode>
    <Budget>2,000,000</Budget>
    <Used>800,000</Used>
  </customer>
```

```
<customer>
  <CustomerID>C003</CustomerID>
  <Name>Anucha Tungkasthan</Name>
  <Email>aimdala@hotmail.com</Email>
  <CountryCode>US</CountryCode>
  <Budget>3,000,000</Budget>
  <Used>600,000</Used>
</customer>
<customer>
  <CustomerID>C004</CustomerID>
  <Name>Walisa Romsaiyud</Name>
  <Email>walisar@gmail.com</Email>
  <CountryCode>US</CountryCode>
  <Budget>4,000,000</Budget>
  <Used>100,000</Used>
</customer>
</mydatabase>
```



## STEP 02 สร้างไฟล์ชื่อ TestXml.aspx

เพิ่ม Code ในหน้า “Source”

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="TestXml.aspx.cs" Inherits="TestXml" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Test Reading XML</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Xml ID="myXml" runat="server"></asp:Xml>
        </div>
    </form>
</body>
</html>
```

## STEP 03

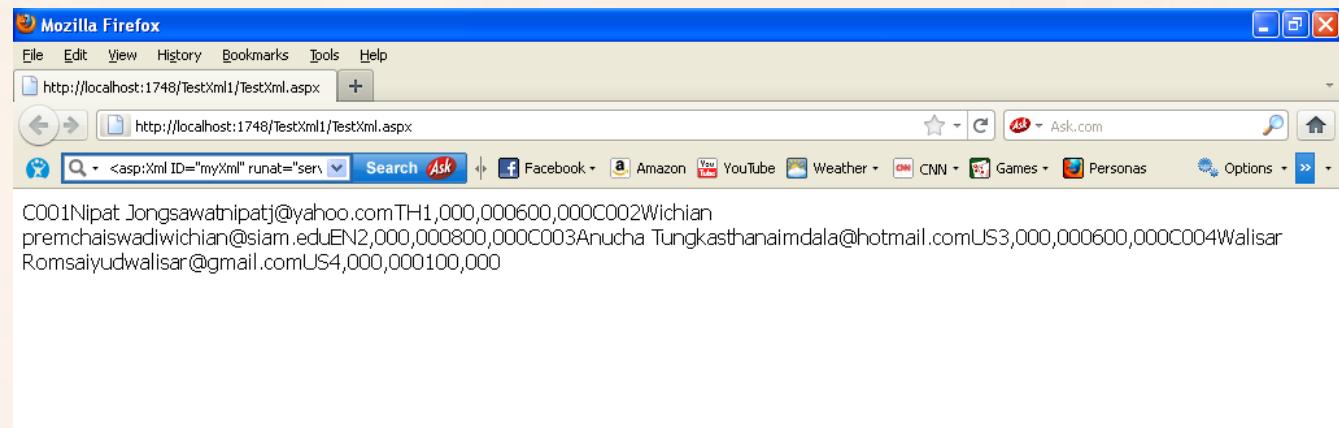
เพิ่ม Code ใหม่ไฟล์ “TestXml.aspx.cs”

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Xml;
using System.Xml.Xsl;

public partial class TestXml : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        XmlDocument doc = new XmlDocument();
        String XmlURL = Server.MapPath("~/App_Data/test.xml");
        doc.Load(XmlURL);
        myXml.Document = doc;
    }
}
```

## STEP 04

ตั้งค่า TestXml.aspx เป็น Start page และกดลง Run  
โปรแกรม ผลลัพธ์แสดงดังนี้





02

Login In โดยใช้ XML

## STEP 01

สร้างไฟล์ชื่อ AllUser.xml  
บันทึกไว้ในโฟล์เดอร์ App\_Data

```
<?xml version="1.0" encoding="utf-8" ?>
<Staff>
  <User>
    <username>admin</username>
    <password>admin</password>
  </User>
  <User>
    <username>nipat</username>
    <password>nipat</password>
  </User>
</Staff>
```

## STEP 02

สร้างไฟล์ชื่อ Login.aspx

2.1 เลือก Control “login”  
กำหนด format ของ Control

The screenshot shows a login interface with a blue header bar labeled "Log In". Below it are two input fields: "User Name:" and "Password:", each accompanied by a mandatory asterisk (\*). Underneath the password field is a checkbox labeled "Remember me next time." To the right of the password field is a "Log In" button.

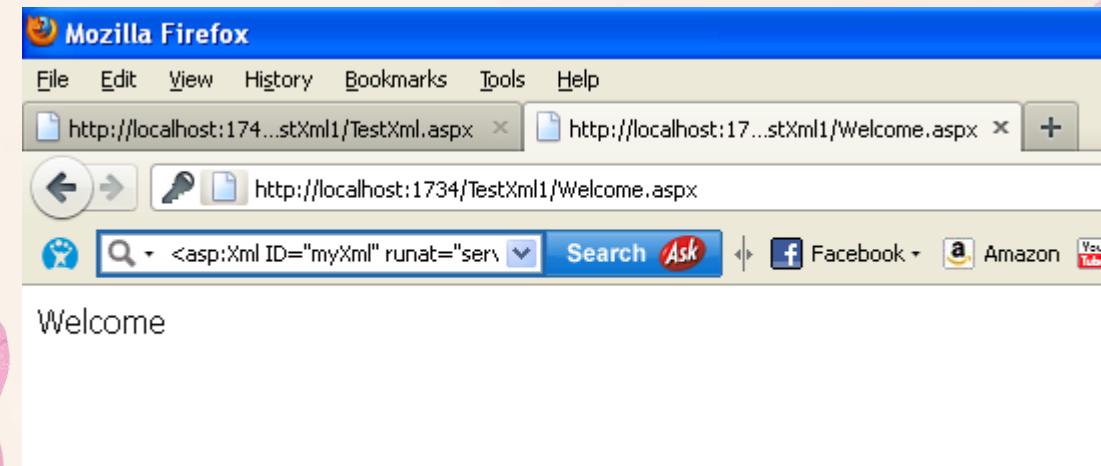
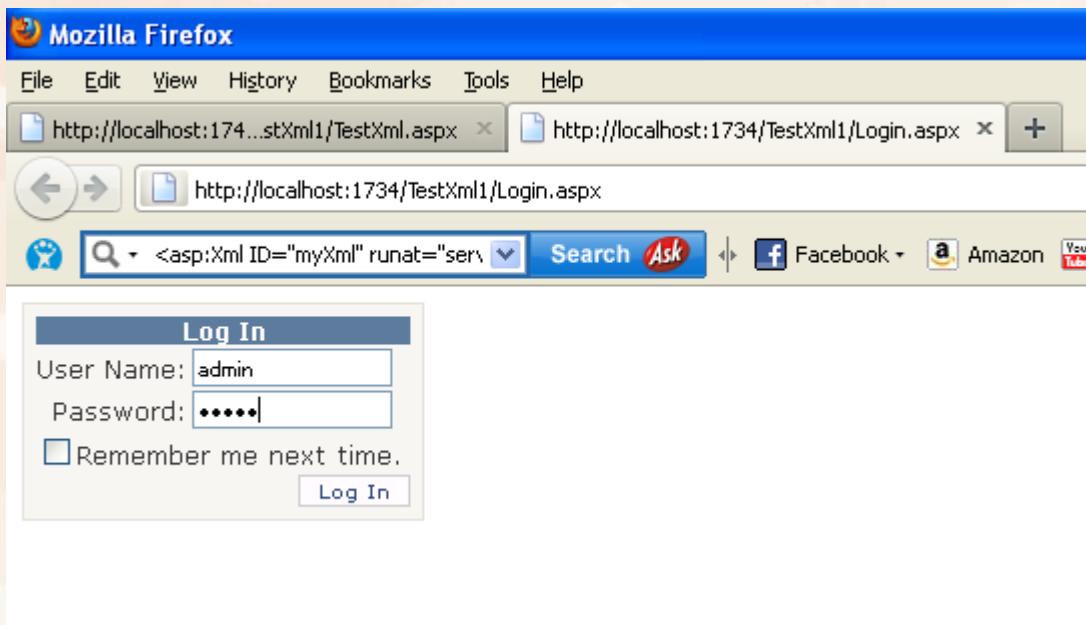
## STEP 03

สร้างไฟล์ชื่อ Welcome.aspx และพิมพ์  
ข้อความ “welcome to this page”

## STEP 04 Double Click ที่ Control “Login” และพิมพ์ source code ดังนี้

```
1  using System.Web;          28
2  using System.Xml;          27
3  using System.Web.Security;  28
4  using System.Web.UI;        29
5  using System.Web.UI.WebControls; 30
6  using System.Web.UI.WebControls.WebParts; 31
7  using System.Web.UI.HtmlControls; 32
8
9  public partial class Login : System.Web.UI.Page
10 {
11     protected void Page_Load(object sender, EventArgs e)
12     {
13     }
14     protected void Login1_Authenticate(object sender, AuthenticateEventArgs e)
15     {
16         string username;
17         string pwd;
18         string CurrentUser = "";
19         string CurrentPwd = "";
20         bool LoginStatus = false;
21         username = Login1.UserName;
22         pwd = Login1.Password;
23
24         XmlDocument xd = new XmlDocument();
25         xd.Load(Server.MapPath("~/App_Data/AllUser.xml"));
26
27         XmlNodeList xnl = xd.GetElementsByTagName("User");
28
29         foreach (XmlNode xn in xnl)
30         {
31             XmlNodeList cxnl = xn.ChildNodes;
32
33             foreach (XmlNode cxn in cxnl)
34             {
35                 if (cxn.Name == "username")
36                 {
37                     if (cxn.InnerText == username)
38                     {
39                         CurrentUser = username;
40                     }
41                 }
42                 if (cxn.Name == "password")
43                 {
44                     if (cxn.InnerText == pwd)
45                     {
46                         CurrentPwd = pwd;
47                     }
48                 }
49             }
50         }
51
52         if ((CurrentUser != "") & (CurrentPwd != ""))
53         {
54             LoginStatus = true;
55         }
56     }
57
58     if (LoginStatus == true)
59     {
60         Session["UserAuthentication"] = username;
61         Session.Timeout = 1;
62
63         Response.Redirect("Welcome.aspx");
64     }
65     else
66     {
67         Session["UserAuthentication"] = "";
68     }
69 }
70 }
```

## STEP 05 ตั้งค่า Login.aspx เป็น Start page แล้วทดลอง Run โปรแกรมผลลัพธ์แสดงดังนี้





03

# Display XML data ໂຄນໄຈ DataGrid

## STEP 01

สร้างไฟล์ชื่อ NewEmployees.xml และบันทึกไว้ในโฟล์เดอร์ App\_Data

```
<?xml version="1.0" encoding="utf-8" ?>
<employees>
    <employee id="1">
        <firstName>Nancy</firstName>
        <lastName>Lee</lastName>
        <city>Seattle</city>
        <state>WA</state>
        <zipCode>98122</zipCode>
    </employee>
    <employee id="2">
        <firstName>Jason</firstName>
        <lastName>Wang</lastName>
        <city>Vancouver</city>
        <state>WA</state>
        <zipCode>98123</zipCode>
    </employee>
</employees>
```

## STEP 02

สร้างไฟล์ชื่อ GridXML.aspx

2.1 เลือก Control “GridView” จาก Toolbox “Data”

Column0	Column1	Column2
abc	abc	abc

2.2 กำหนด ID ของ Control “GridView” เป็นชื่อ “dgEmployees”

2.3 สร้างปุ่มที่มีข้อความ “Display data from XML” และกำหนด ID ของ Control Button “btnDisplay”

Display data from XML

2.4 Double Click ปุ่ม btnDisplay

## STEP 03 พิมพ์ code ดังต่อไปนี้

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Data;
8  using System.Data.OleDb;
9
10 public partial class GridXML : System.Web.UI.Page
11 {
12     protected void Page_Load(object sender, EventArgs e)
13     {
14
15     }
16     protected void btnDisplay_Click(object sender, EventArgs e)
17     {
18         DataSet objDataSet = new DataSet();
19         objDataSet.ReadXml(Server.MapPath("~/App_Data/NewEmployees.xml"));
20         dgEmployees.DataSource = objDataSet.Tables[0].DefaultView;
21         dgEmployees.DataBind();
22     }
23 }
```

## STEP 04 ตั้งค่า GridXML.aspx เป็น Start page แล้วกดลง Run โปรแกรม

firstName	lastName	city	state	zipCode	id
Nancy	Lee	Seattle	WA	98122	1
Jason	Wang	Vancouver	WA	98123	2

Display data from XML

# Phone.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<contacts>
  <contact>
    <name>Nipat</name>
    <phone>081-111-2345</phone>
  </contact>
  <contact>
    <name>Anucha</name>
    <phone>084-234-5678</phone>
  </contact>
</contacts>
```

<b>name</b>	<b>phone</b>
Nipat	081-111-2345
Anucha	084-234-5678

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7
8  using System.Xml;
9  using System.Data;
10
11 public partial class _Default : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15         DataSet ds = new DataSet();
16         ds.ReadXml(Server.MapPath("~/Phone.xml"));
17         GridView1.DataSource = ds;
18         GridView1.DataBind();
19     }
20 }
```



04

Write XML data

# STEP 01

## สร้างไฟล์ชื่อ WriteXml.aspx

- 1.1 สร้างปุ่มที่มีข้อความ “Create XML” และกำหนด ID ของ Control Button “btnCreateXml”

Create XML

- 1.2 Double Click ปุ่ม btnCreateXml

```
26  
27     int i = 0;  
28     for (i = 1; i <= 3; i++)  
29     {  
30         myXmlWriter.WriteStartElement("Product");// see next line  
31         myXmlWriter.WriteAttributeString("Id", i.ToString());// <Product id="">  
32         //<price>price element text for ID 1</price>  
33         myXmlWriter.WriteString("price", "price element text for ID " + i.ToString());  
34         // <weight>price element text for ID 1</price>  
35         myXmlWriter.WriteString("weight", "weight element text for ID " + i.ToString());  
36         // <description>description element text for ID 3</description> | Just an example of  
37         // how to write raw data  
38         myXmlWriter.WriteRaw(Environment.NewLine + " <description>description element text for ID " + i.ToString() + "</description>" + Environment.NewLine);  
39         myXmlWriter.WriteEndElement(); //</Product>  
40     }  
41     myXmlWriter.WriteEndElement(); // </Products>  
42     myXmlWriter.Close();  
43 }  
44 }  
45 }
```

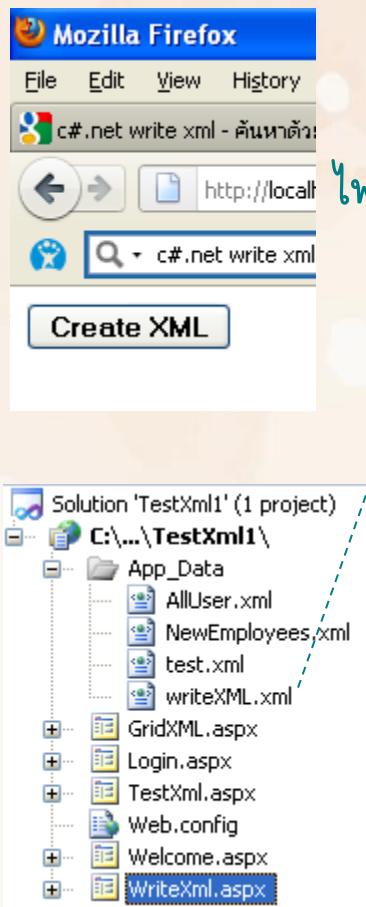
# STEP 02

## พิมพ์ source code ดังต่อไปนี้

```
1     using System;  
2     using System.Collections.Generic;  
3     using System.Linq;  
4     using System.Web;  
5     using System.Web.UI;  
6     using System.Web.UI.WebControls;  
7     using System.Xml;  
8  
9     public partial class WriteXml : System.Web.UI.Page  
10    {  
11        protected void Page_Load(object sender, EventArgs e)  
12        {  
13        }  
14        protected void btnCreateXml_Click(object sender, EventArgs e)  
15        { // *** XmlTextWriter  
16            XmlTextWriter myXmlWriter = new  
17            XmlTextWriter(Server.MapPath("~/App_Data/writeXML.xml"), null);  
18            myXmlWriter.Formatting = Formatting.Indented;  
19            myXmlWriter.Indentation = 4;  
20  
21            // <!--http://forum.coolwebawards.com-->  
22            myXmlWriter.WriteComment("http://forum.coolwebawards.com");  
23            // <Products>  
24            myXmlWriter.WriteStartElement("Products");  
25        }
```

# STEP 03

ตั้งค่า WriteXML.aspx เป็น Start page แล้วกดลง Run โปรแกรม



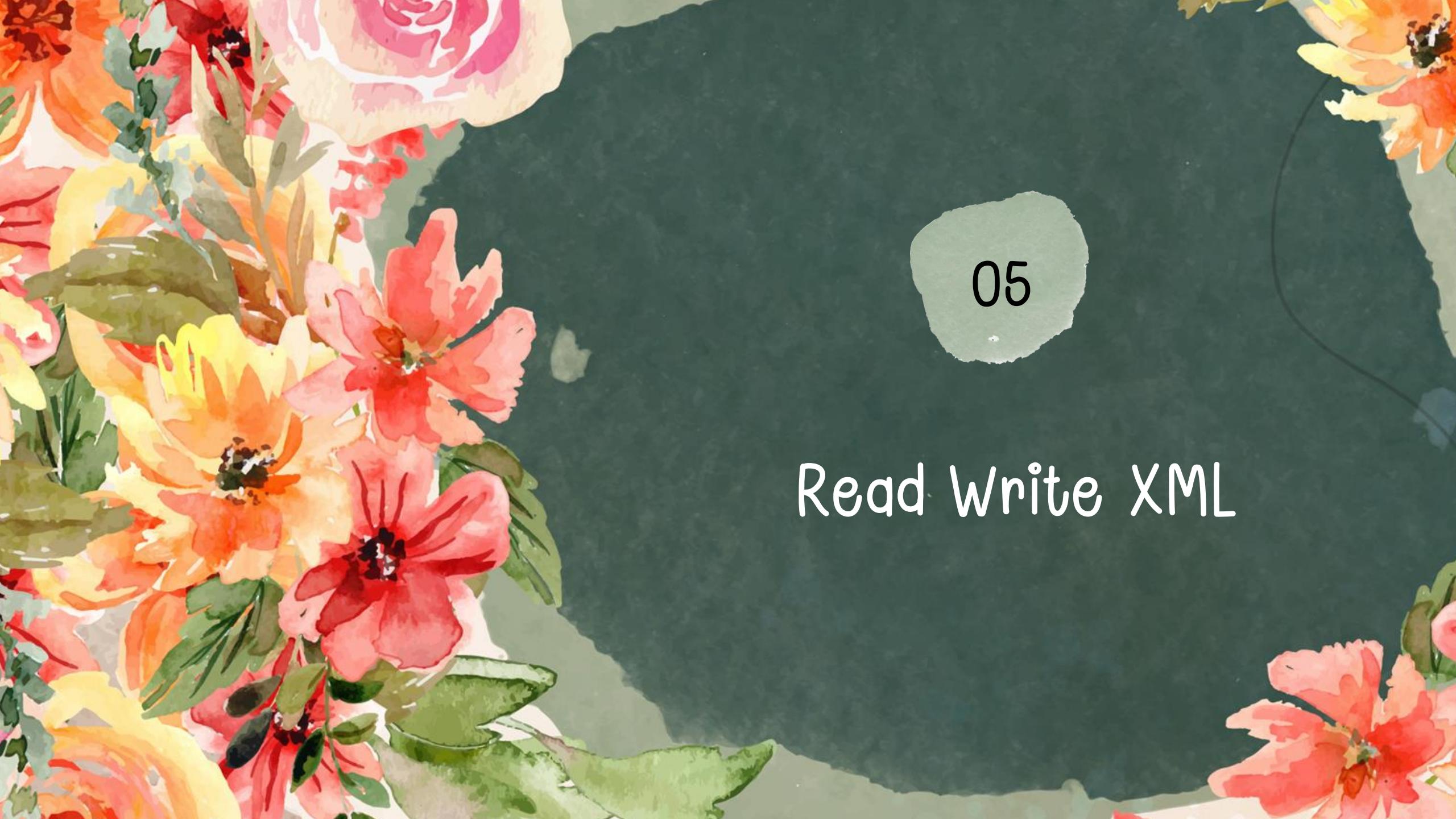
ข้อมูลในไฟล์ writeXML.xml

```
<!--http://forum.coolwebawards.com-->
<Products>
  <Product Id="1">
    <price>price element text for ID 1</price>
    <weight>weight element text for ID 1</weight>
    <description>description element text for ID 1</description>
  </Product>
  <Product Id="2">
    <price>price element text for ID 2</price>
    <weight>weight element text for ID 2</weight>
    <description>description element text for ID 2</description>
  </Product>
  <Product Id="3">
    <price>price element text for ID 3</price>
    <weight>weight element text for ID 3</weight>
    <description>description element text for ID 3</description>
  </Product>
</Products>
```

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.UI;
6 using System.Web.UI.WebControls;
7 using System.Text;
8 using System.Xml;
9
10 public partial class _Default : System.Web.UI.Page
11 {
12     protected void Page_Load(object sender, EventArgs e)
13     {
14
15     }
16     protected void btnCreateXML_Click(object sender, EventArgs e)
17     {
18         XmlWriter xmlWriter = XmlWriter.Create("C:/Users/nipat/Documents/Visual Studio 2010/WebSites/27_WriteXML/test.xml");
19
20         xmlWriter.WriteStartDocument();
21         xmlWriter.WriteStartElement("users");
22
23         xmlWriter.WriteStartElement("user");
24         xmlWriter.WriteAttributeString("age", "35");
25         xmlWriter.WriteString("Anucha Tungkasthan");
26         xmlWriter.WriteEndElement();
27
28         xmlWriter.WriteStartElement("user");
29         xmlWriter.WriteAttributeString("age", "44");
30         xmlWriter.WriteString("Pittaya Pompuang");
31
32         xmlWriter.WriteEndDocument();
33         xmlWriter.Close();
34     }
35 }
```

## Test.xml

```
<?xml version="1.0" encoding="utf-8"?>
<users>
    <user age="35">Anucha Tungkasthan</user>
    <user age="44">Pittaya Pompuang</user>
</users>
```



05

## Read Write XML

# Employee.xml

```
<?xml version="1.0" encoding="utf-8"?>
<!--Programmatically writing XML-->
<Employees>
    <Employee type="Permanent">
        <ID>100</ID>
        <FirstName>Nipat</FirstName>
        <LastName>Jongsawat</LastName>
        <Dept>IT</Dept>
    </Employee>
    <Employee type="Temporary">
        <ID>101</ID>
        <FirstName>Anucha</FirstName>
        <LastName>Tungkasthan</LastName>
        <Dept>Computer Engineering</Dept>
    </Employee>
```

```
<Employee type="Permamnet">
    <ID>102</ID>
    <FirstName>Walisa</FirstName>
    <LastName>Romsaiyud</LastName>
    <Dept>IT</Dept>
</Employee>
<Employee type="Temporary">
    <ID>103</ID>
    <FirstName>Sarayut</FirstName>
    <LastName>Intarasema</LastName>
    <Dept>Computer Engineering</Dept>
</Employee>
</Employees>
```

```
1  using System;                                26
2  using System.Collections.Generic;              27
3  using System.Linq;                            28
4  using System.Web;                            29
5  using System.Web.UI;                          30
6  using System.Web.UI.WebControls;             31
7  using System.Xml;                            32
8  using System.Text;                            33
9  using System.Data;                           34
10
11 public partial class WriteXMLFile : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15
16     }
17
18     public DataTable GetEmployees()
19     {
20         DataTable dt = new DataTable();
21
22         DataColumn dc = new DataColumn();
23         dc.ColumnName = "ID";
24         dt.Columns.Add(dc);
25
26         dc = new DataColumn();
27         dc.ColumnName = "FirstName";
28         dt.Columns.Add(dc);
29
30         dc = new DataColumn();
31         dc.ColumnName = "LastName";
32         dt.Columns.Add(dc);
33
34         dc = new DataColumn();
35         dc.ColumnName = "Type";
36         dt.Columns.Add(dc);
37
38         dc = new DataColumn();
39         dc.ColumnName = "Dept";
40         dt.Columns.Add(dc);
41
42         DataRow dr = dt.NewRow();
43         dr[0] = "100";
44         dr[1] = "Nipat";
45         dr[2] = "Jongsawat";
46         dr[3] = "Permanent";
47         dr[4] = "IT";
48         dt.Rows.Add(dr);
49
50         dr = dt.NewRow();
51         dr[0] = "101";
52         dr[1] = "Anucha";
53         dr[2] = "Tungkasthan";
54         dr[3] = "Temporary";
55         dr[4] = "Computer Engineering";
56         dt.Rows.Add(dr);
57
58         dr = dt.NewRow();
59         dr[0] = "102";
60         dr[1] = "Walisa";
61         dr[2] = "Romsaiyud";
62         dr[3] = "Permamnet";
63         dr[4] = "IT";
64         dt.Rows.Add(dr);
65
66         dr = dt.NewRow();
67         dr[0] = "103";
68         dr[1] = "Sarayut";
69         dr[2] = "Intarasema";
70         dr[3] = "Temporary";
71         dr[4] = "Computer Engineering";
72         dt.Rows.Add(dr);
73
74         return dt;
75 }
```

```
74 }
75 protected void btnWriteXML_Click(object sender, EventArgs e)
76 {
77     XmlTextWriter xmlwriter = new XmlTextWriter(Server.MapPath("App_Data/Employee.xml"), Encoding.UTF8);
78     xmlwriter.Formatting = Formatting.Indented;
79     xmlwriter.WriteStartDocument();
80     xmlwriter.WriteComment("Programmatically writing XML");
81     xmlwriter.WriteLineElement("Employees");
82     DataTable dt = GetEmployees();
83     for (int i = 0; i < dt.Rows.Count; i++)
84     {
85         xmlwriter.WriteLineElement("Employee");
86         xmlwriter.WriteLineAttributeString("type", dt.Rows[i]["Type"].ToString());
87         xmlwriter.WriteLineString("ID", dt.Rows[i]["ID"].ToString());
88         xmlwriter.WriteLineString("FirstName", dt.Rows[i]["FirstName"].ToString());
89         xmlwriter.WriteLineString("LastName", dt.Rows[i]["LastName"].ToString());
90         xmlwriter.WriteLineString("Dept", dt.Rows[i]["Dept"].ToString());
91         xmlwriter.WriteLineEndElement();
92     }
93     xmlwriter.WriteLineEndElement();
94     xmlwriter.WriteLineEndDocument();
95     xmlwriter.Flush();
96     xmlwriter.Close();
97 }
```

```
98 }
99 protected void btnReadXML_Click(object sender, EventArgs e)
100 {
101     using (XmlTextReader reader = new XmlTextReader(Server.MapPath("App_Data/Employee.xml")))
102     {
103         while (reader.Read())
104         {
105             if (reader.NodeType == XmlNodeType.Text)
106             {
107                 //Response.Write(reader.Value);
108                 //Response.Write("<br>");
109                 lstDisplayXML.Items.Add(reader.Value);
110             }
111         }
112     }
113 }
```

# Car.xml

```
<?xml version="1.0" encoding="utf-8"?>
<!--Programmatically writing XML-->
<Cars>
    <Car type="Type: Sedan">
        <ID>ID: 005</ID>
        <ModelName>Model: Toyota</ModelName>
        <Year>Year: 2015</Year>
        <Price>Price: 560000</Price>
        <PayInterest>Total Interest:
        21560</PayInterest>
    </Car>
</Cars>
```

```
1 using System;                                26 dc = new DataColumn();                      50 dr[3] = ddlType.Text;
2 using System.Collections.Generic;            27 dc.ColumnName = "ModelName";                51 dr[4] = txtPrice.Text;
3 using System.Linq;                            28 dt.Columns.Add(dc);                      52
4 using System.Web;                            29
5 using System.Web.UI;                         30 dc = new DataColumn();                      53 double year = Convert.ToDouble(ddlYear.Text);
6 using System.Web.UI.WebControls;             31 dc.ColumnName = "Year";                     54 double price = Convert.ToDouble(txtPrice.Text);
7 using System.Xml;                            32 dt.Columns.Add(dc);                      55
8 using System.Text;                           33
9 using System.Data;                           34 dc = new DataColumn();                      56 localhost.PayUsedCar myService = new localhost.PayUsedCar();
10
11 public partial class WriteXMLFile : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15
16     }
17
18     public DataTable GetEmployees()
19     {
20         DataTable dt = new DataTable();
21
22         DataColumn dc = new DataColumn();          46 dc = new DataColumn();                      66 dr[5] = PayInterest;
23         dc.ColumnName = "ID";                      47 dc.ColumnName = "PayInterest";              67
24         dt.Columns.Add(dc);                      48 DataRow dr = dt.NewRow();                  68 dt.Rows.Add(dr);
25
26         dc = new DataColumn();                      49 dr[0] = txtId.Text;                      69
27         dc.ColumnName = "ModelName";                50 dr[1] = ddlmodelName.Text;                 70
28         dt.Columns.Add(dc);                      51 dr[2] = ddlYear.Text;                     71
29
30         dc = new DataColumn();                      52
31         dc.ColumnName = "Year";                   53 double year = Convert.ToDouble(ddlYear.Text);
32         dt.Columns.Add(dc);                      54 double price = Convert.ToDouble(txtPrice.Text);
33
34         dc = new DataColumn();                      55 localhost.PayUsedCar myService = new localhost.PayUsedCar();
35         dc.ColumnName = "Type";                   56 double payInterest = myService.payInterestUsedCar(year, price);
36         dt.Columns.Add(dc);                      57 String PayInterest = Convert.ToString(payInterest);
37
38         dc = new DataColumn();                      58
39         dc.ColumnName = "Price";                  59 lblOne.Text = Convert.ToString(payInterest / 12);
40         dt.Columns.Add(dc);                      60 lblTwo.Text = Convert.ToString(payInterest / 24);
41
42         dc = new DataColumn();                      61 lblThree.Text = Convert.ToString(payInterest / 36);
43         dc.ColumnName = "PayInterest";           62 lblFour.Text = Convert.ToString(payInterest / 48);
44         dt.Columns.Add(dc);                      63 lblFive.Text = Convert.ToString(payInterest / 60);
45
46         dc = new DataColumn();                      64 lblSix.Text = Convert.ToString(payInterest / 72);
47         dc.ColumnName = "PayInterest";           65
48         dt.Columns.Add(dc);                      66
49
50         DataRow dr = dt.NewRow();                  67
51         dr[0] = txtId.Text;                      68
52         dr[1] = ddlmodelName.Text;               69
53         dr[2] = ddlYear.Text;                    70
54
55         dt.Rows.Add(dr);                        71
56
57         return dt;                            72
58     }                                         73 }
```

```

44 protected void btnReadXML_Click(object sender, EventArgs e)
45 {
46     using (XmlTextReader reader = new XmlTextReader(Server.MapPath("App_Data/Cars.xml")))
47     {
48         while (reader.Read())
49         {
50             if (reader.NodeType == XmlNodeType.Text)
51             {
52                 //Response.Write(reader.Value);
53                 //Response.Write("<br>");
54                 lstDisplayXML.Items.Add(reader.Value);
55             }
56         }
57     }
58 }
59
60 protected void btnWriteXML_Click1(object sender, EventArgs e)
61 {
62     XmlTextWriter xmlwriter = new XmlTextWriter(Server.MapPath("App_Data/Cars.xml"), Encoding.UTF8);
63     xmlwriter.Formatting = Formatting.Indented;
64     xmlwriter.WriteStartDocument();
65     xmlwriter.WriteComment("Programmatically writing XML");
66     xmlwriter.WriteStartElement("Cars");
67     DataTable dt = GetEmployees();
68
69     for (int i = 0; i < dt.Rows.Count; i++)
70     {
71         xmlwriter.WriteStartElement("Car");
72         xmlwriter.WriteAttributeString("type", "Type: " + dt.Rows[i]["Type"].ToString());
73         xmlwriter.WriteLineString("ID", "ID: " + dt.Rows[i]["ID"].ToString());
74         xmlwriter.WriteLineString("ModelName", "Model: " + dt.Rows[i]["ModelName"].ToString());
75         xmlwriter.WriteLineString("Year", "Year: " + dt.Rows[i]["Year"].ToString());
76         xmlwriter.WriteLineString("Price", "Price: " + dt.Rows[i]["Price"].ToString());
77         xmlwriter.WriteLineString("PayInterest", "Total Interest: " + dt.Rows[i]["PayInterest"].ToString());
78         xmlwriter.WriteEndElement();
79     }
80     xmlwriter.WriteEndElement();
81     xmlwriter.WriteEndDocument();
82     xmlwriter.Flush();
83     xmlwriter.Close();
84 }
85
86
87
88
89
90
91
92
93
94
95
96

```



```

87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113

```

Enter the Used Car data:

Id:	<input type="text"/>
Model Name:	<input type="text"/> Toyota
Year:	<input type="text"/> 2015
Type:	<input type="text"/> Sedan
Price:	<input type="text"/>

1) Click "Add" to add a used car

2) Click "Display" to display used cars' info

Display Used Cars' Info here!  
 (Interest Rate: 3.85% if >2010; 4.15% if 2005->2010; 4.5% if <2005)

Pay Interest per Month:  
 1 year = Baht  
 2 year = Baht  
 3 year = Baht  
 4 year = Baht  
 5 year = Baht  
 6 year = Baht



06

Read Update XML

## data.xml

```
<?xml version="1.0" standalone="yes"?>
<Users>
  <user>
    <Id>1001</Id>
    <Name>Nipat Jongsawat</Name>
    <Dsng>Dr.</Dsng>
  </user>
  <user>
    <Id>1002</Id>
    <Name>Anucha Tungkasthan</Name>
    <Dsng>Dr.</Dsng>
  </user>
  <user>
    <Id>1003</Id>
    <Name>Walisa Romsaiyud</Name>
    <Dsng>Assistant Professor Dr.</Dsng>
  </user>
```

```
<user>
  <Id>1004</Id>
  <Name>Wichian Premchaiswadi</Name>
  <Dsng>Associate Professor Dr.</Dsng>
</user>
<user>
  <Id>1005</Id>
  <Name>Sarayut Intarasema</Name>
  <Dsng>A.jarn.</Dsng>
</user>
<user>
  <Id>1006</Id>
  <Name>somchai Ployyam</Name>
  <Dsng>Dr.</Dsng>
</user>
<user>
  <Id>1007</Id>
  <Name>Chai Somboon</Name>
  <Dsng>A.jarn.</Dsng>
</user>
</Users>
```

```
1  using System;
2  using System.Collections;
3  using System.Configuration;
4  using System.Data;
5  using System.Linq;
6  using System.Web;
7  using System.Web.Security;
8  using System.Web.UI;
9  using System.Web.UI.HtmlControls;
10 using System.Web.UI.WebControls;
11 using System.Web.UI.WebControls.WebParts;
12 using System.Xml.Linq;
13
14 namespace ReadingXMLfile
15 {
16     public partial class _Default : System.Web.UI.Page
17     {
18         protected void Page_Load(object sender, EventArgs e)
19         {
20             if (!IsPostBack)
21             {
22                 DataSet ds = new DataSet();
23                 ds.ReadXml(Server.MapPath("data.xml"));
24                 Session["Data"] = ds.Tables[0];
25                 Bind(); // binding the data to gridView(id=gvdata)
26             }
27         }
28     }
29 }
30
31 protected void btnSave_Click(object sender, EventArgs e)
32 {
33     DataTable dt = Session["Data"] as DataTable;
34     DataRow dr = dt.NewRow(); // creating new row to add new record into datatable
35
36     TextBox t1 = (TextBox)gvData.FooterRow.FindControl("txtId");
37     TextBox t2 = (TextBox)gvData.FooterRow.FindControl("txtName");
38     TextBox t3 = (TextBox)gvData.FooterRow.FindControl("txtDsng");
39
40     dr[0] = t1.Text;
41     dr[1] = t2.Text;
42     dr[2] = t3.Text;
43     dt.Rows.Add(dr);
44     dt.AcceptChanges();
45     dt.WriteXml(Server.MapPath("data.xml")); // updating the new record into XML File
46     Session["Data"] = dt;
47     Bind();
48     gvData.EditIndex = -1;
49 }
50
51 protected void btnCancel_Click(object sender, EventArgs e)
52 {
53     gvData.EditIndex = -1;
54     Bind();
55 }
56
57 protected void btnAdd_Click(object sender, EventArgs e)
58 {
59     gvData.FooterRow.Visible = true;
60 }
61
62 public void Bind()
63 {
64     gvData.DataSource = Session["Data"];
65     gvData.DataBind();
66 }
```

# ReadingXMLfile.csproj

```
<Project ToolsVersion="3.5" DefaultTargets="Build" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
<PropertyGroup>
<Configuration Condition=" '$(Configuration)' == "" >Debug</Configuration>
<Platform Condition=" '$(Platform)' == "" >AnyCPU</Platform>
<ProductVersion>9.0.30729</ProductVersion>
<SchemaVersion>2.0</SchemaVersion>
<ProjectGuid>{3AA6EFOF-16F7-40E8-BCCA-2C8B7FB414C6}</ProjectGuid>
<ProjectTypeGuids>{349c5851-65d1-11da-9384-00065b846f21};{f0e04ec0-301f-11d3-bf4b-00c04f79efbc}</ProjectTypeGuids>
<OutputType>Library</OutputType>
<AppDesignerFolder>Properties</AppDesignerFolder>
<RootNamespace>ReadingXMLfile</RootNamespace>
<AssemblyName>ReadingXMLfile</AssemblyName>
<TargetFrameworkVersion>v3.5</TargetFrameworkVersion>
</PropertyGroup>
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
<DebugSymbols>true</DebugSymbols>
<DebugType>full</DebugType>
<Optimize>false</Optimize>
<OutputPath>bin\</OutputPath>
<DefineConstants>DEBUG;TRACE</DefineConstants>
<ErrorReport>prompt</ErrorReport>
<WarningLevel>4</WarningLevel>
</PropertyGroup>
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
<DebugType>pdbsonly</DebugType>
<Optimize>true</Optimize>
<OutputPath>bin\</OutputPath>
<DefineConstants>TRACE</DefineConstants>
<ErrorReport>prompt</ErrorReport>
<WarningLevel>4</WarningLevel>
</PropertyGroup>
<ItemGroup>
<Reference Include="System" />
<Reference Include="System.Data" />
<Reference Include="System.Core" />
<RequiredTargetFramework>3.5</RequiredTargetFramework>
</Reference>
<Reference Include="System.Data.DataSetExtensions" />
<RequiredTargetFramework>3.5</RequiredTargetFramework>
</Reference>
<Reference Include="System.Web.Extensions" />
<RequiredTargetFramework>3.5</RequiredTargetFramework>
</Reference>
<Reference Include="System.Xml.Linq" />
<RequiredTargetFramework>3.5</RequiredTargetFramework>
</Reference>
</ItemGroup>
<ItemGroup>
<Content Include="data.xml" />
<Content Include="Default.aspx" />
<Content Include="Web.config" />
</ItemGroup>
<ItemGroup>
<Compile Include="Default.aspx.cs" />
<SubType>ASPXCodeBehind</SubType>
<DependentUpon>Default.aspx</DependentUpon>
</Compile>
<Compile Include="Default.aspx.designer.cs" />
<DependentUpon>Default.aspx</DependentUpon>
</Compile>
<Compile Include="Properties\AssemblyInfo.cs" />
</ItemGroup>
</Project>
```

# Reading XML file.csproj

```
<Reference Include="System.Drawing" />
<Reference Include="System.Web" />
<Reference Include="System.Xml" />
<Reference Include="System.Configuration" />
<Reference Include="System.Web.Services" />
<Reference Include="System.EnterpriseServices" />
<Reference Include="System.Web.Mobile" />
</ItemGroup>
<ItemGroup>
<Content Include="data.xml" />
<Content Include="Default.aspx" />
<Content Include="Web.config" />
</ItemGroup>
<ItemGroup>
<Compile Include="Default.aspx.cs">
<SubType>ASPXCodeBehind</SubType>
<DependentUpon>Default.aspx</DependentUpon>
</Compile>
<Compile Include="Default.aspx.designer.cs">
<DependentUpon>Default.aspx</DependentUpon>
</Compile>
<Compile Include="Properties\AssemblyInfo.cs" />
</ItemGroup>
<ItemGroup>
<Folder Include="App_Data\" />
</ItemGroup>
<Import Project="$(MSBuildBinPath)\Microsoft.CSharp.targets" />
<Import Project="$(MSBuildExtensionsPath)\Microsoft\VisualStudio\v9.0\WebApplications\Microsoft.WebApplication.targets" />
<ProjectExtensions>
<VisualStudio>
<FlavorProperties GUID="{349c5851-65d9-11da-9384-00065b846f21}">
<WebProjectProperties>
<UseIIS>False</UseIIS>
<AutoAssignPort>True</AutoAssignPort>
<DevelopmentServerPort>3707</DevelopmentServerPort>
<DevelopmentServerVPath>/</DevelopmentServerVPath>
<IISUrl>
</IISUrl>
<NTLMAuthentication>False</NTLMAuthentication>
<UseCustomServer>False</UseCustomServer>
<CustomServerUrl>
</CustomServerUrl>
<SaveServerSettingsInUserFile>False</SaveServerSettingsInUserFile>
</WebProjectProperties>
</FlavorProperties>
</VisualStudio>
</ProjectExtensions>
</Project>
```



07

# Read Write Delete XML

# emp.xml

```
<?xml version="1.0" standalone="yes"?>
<DocumentElement>
    <emptable>
        <eno>6</eno>
        <ename>rest</ename>
        <eadd>e</eadd>
        <esal>54321</esal>
    </emptable>
    <emptable>
        <eno>7</eno>
        <ename>test</ename>
        <eadd>test</eadd>
        <esal>23456</esal>
    </emptable>
    <emptable>
        <eno>8</eno>
        <ename>xxxx</ename>
        <eadd>xxxx</eadd>
        <esal>1234</esal>
    </emptable>
    <emptable>
        <eno>9</eno>
        <ename>asd</ename>
        <eadd>sdf</eadd>
        <esal>1234</esal>
    </emptable>
</DocumentElement>
```

```
1  using System;                                     26
2  using System.Data;                                27
3  using System.Configuration;                      28
4  using System.Web;                                 29
5  using System.Web.Security;                        30
6  using System.Web.UI;                             31
7  using System.Web.UI.WebControls;                 32
8  using System.Web.UI.WebControls.WebParts;        33
9  using System.Web.UI.HtmlControls;                34
10 using System.IO;                                  35
11
12 public partial class _Default : System.Web.UI.Page
13 {
14     protected void Page_Load(object sender, EventArgs e)
15     {
16         String st = Server.MapPath("emp.xml");
17         if (Page.IsPostBack == false)
18         {
19             if(File.Exists(st)==false)
20                 creatxml();
21             getxml();
22         }
23     }
24     private void creatxml()
25     {
26         DataTable tb = new DataTable("emptable");
27         tb.Columns.Add("eno", Type.GetType("System.Int32"));
28         tb.Columns.Add("ename", Type.GetType("System.String"));
29         tb.Columns.Add("eadd", Type.GetType("System.String"));
30         tb.Columns.Add("esal", Type.GetType("System.Int32"));
31         DataRow r = tb.NewRow();
32         r[0] = 1;
33         r[1] = "Nipat Jongsawat";
34         r[2] = "Samutprakarn Prapadaeng";
35         r[3] = 8500;
36         tb.Rows.Add(r);
37         String st = Server.MapPath("emp.xml");
38         tb.WriteXml(st);
39     }
40     private void getxml()
41     {
42         String st = Server.MapPath("emp.xml");
43         DataSet ds = new DataSet();
44         ds.ReadXml(st);
45         GridView1.DataSource = ds;
46         GridView1.DataBind();
47     }
48     protected void GridView1_RowCommand(object sender, GridViewCommandEventArgs e)
49     {
50 }
```

```
50 if (e.CommandName == "save")
51 {
52     String st = Server.MapPath("emp.xml");
53     DataSet ds = new DataSet();
54     ds.ReadXml(st);
55     DataRow r = ds.Tables[0].NewRow();
56     r[0] = Convert.ToInt32(((TextBox)(GridView1.FooterRow.FindControl("TextBox1"))).Text);
57     r[1] = ((TextBox)(GridView1.FooterRow.FindControl("TextBox3"))).Text;
58     r[2] = ((TextBox)(GridView1.FooterRow.FindControl("TextBox5"))).Text;
59     r[3] = Convert.ToInt32(((TextBox)(GridView1.FooterRow.FindControl("TextBox7"))).Text);
60     ds.Tables[0].Rows.Add(r);
61     ds.WriteXml(st);
62     getxml();
63 }
64 }
65 protected void GridView1_RowDeleting(object sender, GridViewDeleteEventArgs e)
66 {
67     String st = Server.MapPath("emp.xml");
68     DataSet ds = new DataSet();
69     ds.ReadXml(st);
70     ds.Tables[0].Rows.RemoveAt(e.RowIndex);
71     ds.WriteXml(st);
72     getxml();
73 }
```

```
74     protected void GridView1_RowEditing(object sender, GridViewEditEventArgs e)
75     {
76         GridView1.EditIndex = e.NewEditIndex;
77         getxml();
78     }
79     protected void GridView1_RowCancelingEdit(object sender, GridViewCancelEditEventArgs e)
80     {
81         GridView1.EditIndex = -1;
82         getxml();
83     }
84     protected void GridView1_RowUpdating(object sender, GridViewUpdateEventArgs e)
85     {
86         String st = Server.MapPath("emp.xml");
87         DataSet ds = new DataSet();
88         ds.ReadXml(st);
89         ds.Tables[0].Rows[e.RowIndex][1] = ((TextBox)(GridView1.Rows[e.RowIndex].FindControl("TextBox2"))).Text;
90         ds.Tables[0].Rows[e.RowIndex][2] = ((TextBox)(GridView1.Rows[e.RowIndex].FindControl("TextBox4"))).Text;
91         ds.Tables[0].Rows[e.RowIndex][3] = Convert.ToInt32(((TextBox)(GridView1.Rows[e.RowIndex].FindControl("TextBox6"))).Text);
92         GridView1.EditIndex = -1;
93         ds.WriteXml(st);
94         getxml();
95     }
96 }
```



08

# LinQ XML

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10
11 public partial class _1ReadXML : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16
17     //Display all XML elements
18     protected void btnDisplay_Click(object sender, EventArgs e)
19     {
20         XElement xelement = XElement.Load
21             (Server.MapPath("~/App_Data/Employees.xml"));
22         //IEnumerable --> Iterate data in memory
23         IEnumerable<XElement> employees = xelement.Elements();
24         // Read the entire XML
25         foreach (var employee in employees)
26         {
27             Response.Write(employee);
28         }
29     }
30 }
```

# 1ReadXML.aspx

1 Sam Male 423-555-0124 424-555-0545  
7A Cox Street Acampo CA 95220 USA  
2 Lucy Female 143-555-0763 434-555-0567  
Jess Bay Alta CA 95701 USA  
3 Kate Female 166-555-0231 233-555-0442  
23 Boxen Street Milford CA 96121 USA  
4 Chris Male 564-555-0122 442-555-0154  
124 Kutbay Montara CA 94037 USA

## Display XML

# 2ReadXMLElements.aspx

List of all Employee Names with sex :  
Sam Male 423-555-0124;

Lucy Female 143-555-0763;

Kate Female 166-555-0231;

Chris Male 564-555-0122

## Display

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10 
11 public partial class _2ReadXMLElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16 
17     //access a Single Element using LINQ to XML
18     protected void btnDisplay_Click(object sender, EventArgs e)
19     {
20         XElement xelement = XElement.Load
21             (Server.MapPath("~/App_Data/Employees.xml"));
22         IEnumerable<XElement> employees = xelement.Elements();
23         Response.Write("List of all Employee Names with sex : ");
24         foreach (var employee in employees)
25         {
26             Response.Write("<br>");
27             Response.Write(employee.Element("Name").Value + " "
28                         + employee.Element("Sex").Value + " "
29                         + employee.Element("Phone").Value +
30                         ";   ");
31             Response.Write("<br>");
32         }
33     }
34 }
```

## 3ReadXMLSpecificElements.aspx

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10
11 public partial class _3ReadXMLSpecificElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16
17     protected void btnDisplay_Click(object sender, EventArgs e)
18     {
19         XElement xelement = XElement.Load
20             (Server.MapPath("~/App_Data/Employees.xml"));
21         IEnumerable<XElement> employees = xelement.Elements();
22         Response.Write("List of all Employee Names along with their ID: ");
23         foreach (var employee in employees)
24         {
25             Response.Write(employee.Element("Name").Value +
26                         ",ID: " + employee.Element("EmpId").Value + " ; ");
27         }
28     }
29 }
```

List of all Employee Names along with their ID: Sam ,ID: 1 ; Lucy ,ID: 2 ; Kate  
,ID: 3 ; Chris ,ID: 4 ;

**Display**

## 5ReadXMLSpecificElements.aspx

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10
11 public partial class _5ReadXMLSpecificElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16
17     //access Specific Element having a Specific Attribute using LINQ to XML
18     protected void btnDisplay_Click(object sender, EventArgs e)
19     {
20         XElement xelement = XElement.Load(Server.MapPath("~/App_Data/Employees.xml"));
21
22         var homePhone = from phoneno in xelement.Elements("Employee")
23                          where (string)phoneno.Element("Phone")
24                            .Attribute("Type") == "Home"
25                          select phoneno;
26
27         Response.Write("List HomePhone Nos.");
28         foreach ( XElement xEle in homePhone)
29         {
30             Response.Write("<br>" + xEle.Element("Phone").Value + ",");
31         }
32     }
33 }
```

List HomePhone Nos.

423-555-0124,

143-555-0763,

166-555-0231,

564-555-0122,

**Display**

# 4ReadXMLSpecificElements.aspx

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10
11 public partial class _3ReadXMLSpecificElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16
17     //access all Elements having a Specific Attribute using LINQ to XML
18     protected void btnDisplay_Click(object sender, EventArgs e)
19     {
20         XElement xelement = XElement.Load
21             (Server.MapPath("~/App_Data/Employees.xml"));
22         var name = from nm in xelement.Elements("Employee")
23             where (string)nm.Element("Sex") == "Female"
24             select nm;
25         Response.Write("Details of Female Employees:");
26         foreach ( XElement xEle in name)
27             Response.Write("<br>" +xEle);
28     }
29 }
```

Details of Female Employees:  
2 Lucy Female 143-555-0763 434-555-0567  
Jess Bay Alta CA 95701 USA

3 Kate Female 166-555-0231 233-555-0442  
23 Boxen Street Milford CA 96121 USA

Display

Please enter "Male" or "Female" and click display:

Display

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10
11 public partial class _3ReadXMLSpecificElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16
17     //access all Elements having a Specific Attribute using LINQ to XML
18     protected void btnDisplay_Click(object sender, EventArgs e)
19     {
20         XElement xelement = XElement.Load
21             (Server.MapPath("~/Employees.xml"));
22         var name = from nm in xelement.Elements("Employee")
23             where (string)nm.Element("Sex") == txtEnter.Text
24             select nm;
25         Response.Write("Details of Male/Female Employees:");
26         foreach ( XElement xEle in name)
27             Response.Write("<br>" +xEle);
28     }
29 }
```

# 4ReadXMLSpecificElements\_Modified.aspx

## 4ReadXMLSpecificElements\_Modified1.aspx

Please choose "Male" or "Female" and click display:

Male  
Female

Display

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10
11 public partial class _3ReadXMLSpecificElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16 }
```

```
17 //access all Elements having a Specific Attribute using LINQ to XML
18 protected void btnDisplay_Click(object sender, EventArgs e)
19 {
20
21     XElement xelement = XElement.Load
22         (Server.MapPath("~/Employees.xml"));
23
24     var name = from nm in xelement.Elements("Employee")
25
26         where (string)nm.Element("Sex") == lstBoxEnter.Text
27
28         select nm;
29
30     Response.Write("Details of Male/Female Employees:");
31
32     foreach ( XElement xEle in name)
33
34         Response.Write("<br>" + xEle);
35
36     }
37 }
```

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10
11 public partial class _3ReadXMLSpecificElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16
17     //Find an Element within another Element using LINQ to XML
18     protected void btnDisplay_Click(object sender, EventArgs e)
19     {
20
21         XElement xelement = XElement.Load(Server.MapPath
22             ("~/App_Data/Employees.xml"));
23
24         var addresses = from address in xelement.Elements("Employee")
25
26             where (string)address.Element("Address").
27                 Element("City") == "Alta"
28             select address;
29
30         Response.Write("Details of Employees living in Alta City: ");
31
32         foreach ( XElement xEle in addresses)
33
34             Response.Write(xEle);
35
}

```

## 6FindXMLElements.aspx

Details of Employees living in Alta City: 2 Lucy Female 143-555-0763 434-555-0567  
*Jess Bay Alta CA 95701 USA*

**Display**

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10
11 public partial class _3ReadXMLSpecificElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15
16     }
17
18     //access all Elements having a Specific Attribute using LINQ to XML
19     protected void btnDisplay_Click(object sender, EventArgs e)
20     {
21
22         XElement xelement = XElement.Load
23             (Server.MapPath("~/Employees.xml"));
24
25         var name = from nm in xelement.Elements("Employee")
26             where (string)nm.Element("Sex") == txtEnter.Text
27             select nm;
28
29         Response.Write("Details of Male/Female Employees:");
30
31         foreach ( XElement xEle in name)
32
33             Response.Write("<br>" +xEle);
34
}

```

## 6FindXMLElements\_Modified.aspx

Choose the city and click display:

Acampo ▾

**Display**

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10
11 public partial class _3ReadXMLSpecificElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16
17     //Find Nested Elements (using Descendants Axis) using LINQ to XML
18     protected void btnDisplay_Click(object sender, EventArgs e)
19     {
20         XElement xelement = XElement.Load
21             (Server.MapPath("~/App_Data/Employees.xml"));
22         Response.Write("List of all Addresses: <br>");
23         foreach (XElement xEle in xelement.Descendants("Address"))
24         {
25             Response.Write((string)xEle + ", <br>");
26         }
27     }
28 }
```

## 7FindXMLNestedElements.aspx

List of all Addresses:  
7A Cox StreetAcampoCA95220USA,  
Jess BayAltaCA95701USA,  
23 Boxen StreetMilfordCA96121USA,  
124 KutbayMontaraCA94037USA,

**Display**

## 8SortXMLElements.aspx

List and Sort all Zip Codes: 94037, 95220, 95701, 96121,

**Display**

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10
11 public partial class _3ReadXMLSpecificElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16
17     //Find Nested Elements (using Descendants Axis) using LINQ to XML
18     protected void btnDisplay_Click(object sender, EventArgs e)
19     {
20         XElement xelement = XElement.Load(Server.MapPath("~/App_Data/Employees.xml"));
21         IEnumerable<string> codes = from code in xelement.Elements("Employee")
22                                         let zip = (string)code.Element("Address")
23                                         .Element("Zip")
24                                         orderby zip
25                                         select zip;
26         Response.Write("List and Sort all Zip Codes: ");
27         foreach (string zp in codes)
28             Response.Write(zp + ", ");
29     }
30 }
```

# ReadUsedCars.aspx

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10 using System.Data;
11
12 public partial class _1ReadXML : System.Web.UI.Page
13 {
14     protected void Page_Load(object sender, EventArgs e)
15     {
16         DataSet ds = new DataSet();
17         ds.ReadXml(Server.MapPath("~/App_Data/UsedCars.xml"));
18         GridView1.DataSource = ds;
19         GridView1.DataBind();
20     }
21     //Display all XML elements
```

```
22     protected void btnDisplay_Click(object sender, EventArgs e)
23     {
24         XElement xelement = XElement.Load
25             (Server.MapPath("~/App_Data/UsedCars.xml"));
26         //IEnumerable --> iterate data in memory
27         IEnumerable<XElement> cars = xelement.Elements();
28         // Read the entire XML
29         foreach (var car in cars)
30         {
31             Response.Write(car);
32         }
33     }
34     protected void btnSearch_Click(object sender, EventArgs e)
35     {
36         XElement xelement = XElement.Load
37             (Server.MapPath("~/App_Data/UsedCars.xml"));
38         var name = from nm in xelement.Elements("Car")
39                     where (string)nm.Element("Brand") == ddlBrand.Text
40                     select nm;
41         Response.Write("Details of Cars:");
42         foreach ( XElement xEle in name)
43             Response.Write("<br>" + xEle);
44     }
```

I22140014 Mazda CX-5 SUV Blue Benzene 2.0 Automatic 6 speeds 2014 N Y 1 CD 18 inch Max Y YYYY  
Y Automatic Y Y Y Y Y Y Keyless  
[Display all cars](#)

CarId	Brand	Model	Type	Color	Engine	CC	Gear	Year	Car_Id
I22140014	Mazda	CX-5	SUV	Blue	Benzene	2.0	Automatic	6 speeds	2014 0

Toyota

Search



09

# LinQ XML <Learning Service>





# LeasingService.asmx

## LeasingService

Click [here](#) for a complete list of operations.

### Leasing

#### Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
year:	8
price:	89

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Services;

6
7  namespace LeasingService
8  {
9      ///<summary>
10     ///</summary>
11     [WebService(Namespace = "http://tempuri.org/")]
12     [WebServiceBinding(ConformsTo = WsProfiles.BasicProfile1_1)]
13     [System.ComponentModel.ToolboxItem(false)]
```

```
15
16  public class LeasingService : System.Web.Services.WebService
17  {
18      [WebMethod]
19      public double Leasing(double year, double price)
20      {
21          //2014-2018 = 3.85; 2009-2013 = 3.14; < 2009 = 4.5;
22          if (year >= 2014 && year <= 2018)
23          {
24              double payInterest = (3.85 * price) / 100;
25              return payInterest;
26          }
27          else if (year >= 2009 && year <= 2013)
28          {
29              double payInterest = (4.14 * price) / 100;
30              return payInterest;
31          }
32          else
33          {
34              double payInterest = (4.5 * price) / 100;
35              return payInterest;
36          }
37      }
38  }
```



10

# LinQ XML <TorAutoCar>



```

<?xml version="1.0" encoding="utf-8"?>
<!--Programmatically writing XML-->
<Cars>
  <Car type="Type: Sedan">
    <CarID>001</CarID>
    <Brand>Toyota</Brand>
    <Model>Camry</Model>
    <Year>2013</Year>
    <Price>760000</Price>
    <Location>Ratchada Soi 24</Location>
    <SellingStatus>Not sold</SellingStatus>
  </Car>
  <Car type="Type: Sedan">
    <CarID>002</CarID>
    <Brand>Honda</Brand>
    <Model>Accord</Model>
    <Year>2009</Year>
    <Price>450000</Price>
    <Location>Kanchanapisek Bangyai</Location>
    <SellingStatus>Not sold</SellingStatus>
  </Car>
  <Car type="Type: SUV">
    <CarID>003</CarID>
    <Brand>Honda</Brand>
    <Model>CRV</Model>
    <Year>2013</Year>
    <Price>795000</Price>
    <Location>Ratchapruk</Location>
    <SellingStatus>Not sold</SellingStatus>
  </Car>
  <Car type="Type: SUV">
    <CarID>005</CarID>
    <Brand>Mazda</Brand>
    <Model>CX-5</Model>
    <Year>2016</Year>
    <Price>1250000</Price>
    <Location>Ratchapruk</Location>
    <SellingStatus>Not sold</SellingStatus>
  </Car>
  <Car type="Type: Sedan">
    <CarID>006</CarID>
    <Brand>Volvo</Brand>
    <Model>S80</Model>
    <Year>2008</Year>
    <Price>680000</Price>
    <Location>Ratanatibet</Location>
    <SellingStatus>Sold</SellingStatus>
  </Car>
  <Car type="Type: PickUp">
    <CarID>008</CarID>
    <Brand>Mitsubishi</Brand>
    <Model>Triton</Model>
    <Year>2011</Year>
    <Price>480000</Price>
    <Location>Ratchapruk</Location>
    <SellingStatus>Not sold</SellingStatus>
  </Car>
  <Car type="Type: PPV">
    <CarID>009</CarID>
    <Brand>Mitsubishi</Brand>
    <Model>Pajero Sport</Model>
    <Year>2009</Year>
    <Price>600000</Price>
    <Location>Kanchanapisek Bangyai</Location>
    <SellingStatus>Sold</SellingStatus>
  </Car>
</Cars>

```

## TorAutoCars.xml

# TorAutoCar.aspx



```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10 using System.Data;
11
12 public partial class TorAutoCar : System.Web.UI.Page
13 {
14     protected void Page_Load(object sender, EventArgs e)
15     {
16     }
17
18     protected void btnSearch_Click(object sender, EventArgs e)
19     {
20         XElement xelement = XElement.Load(Server.MapPath("~/App_Data/TorAutoCars.xml"));
21         var name = from nm in xelement.Elements("Car")
22             where (string)nm.Element("Location") == ddlLocation.Text
23             select nm;
24
25         Response.Write("ຮັບສ່ວນໃນລາຍກາ : " + ddlLocation.Text);
26
27         foreach (XElement xEle in name)
28             Response.Write("<br><br>" + xEle);
29     }
30
31     protected void btnBrand_Click(object sender, EventArgs e)
32     {
33         XElement xelement = XElement.Load(Server.MapPath("~/App_Data/TorAutoCars.xml"));
34         var name = from nm in xelement.Elements("Car")
35             where (string)nm.Element("Brand") == ddlBrand.Text
36             select nm;
37
38         Response.Write("ຮັບສ່ວນ : " + ddlBrand.Text);
39         foreach (XElement xEle in name)
40             Response.Write("<br><br>" + xEle);
41     }
42
43     protected void btnBrandSection_Click(object sender, EventArgs e)
44     {
45         XElement xelement = XElement.Load(Server.MapPath("~/App_Data/TorAutoCars.xml"));
46         var name = from nm in xelement.Elements("Car")
47             where (string)nm.Element("Brand") == ddlBrand.Text && (string)nm.Element("Location") == ddlLocation.Text
48             select nm;
49
50         Response.Write("ຮັບສ່ວນ : " + ddlBrand.Text);
51         foreach (XElement xEle in name)
52             Response.Write("<br><br>" + xEle);
53     }
54 }
```

```

50 protected void btnCompute_Click(object sender, EventArgs e)
51 {
52     localhost.LeasingService myService = new localhost.LeasingService();
53     double payInterest = myService.Leasing(Convert.ToDouble(ddlYear.Text), Convert.ToDouble(txtPrice.Text));
54     double price = Convert.ToDouble(txtPrice.Text);
55     double totalPrice = price + payInterest;
56     double downPayment = Convert.ToDouble(txtDown.Text);
57     double finalPrice = totalPrice - downPayment;
58     double fiveYears = finalPrice / 60;
59     string fiveYearsString = String.Format("{0:0,0.00}", fiveYears);
60     lbl5Years.Text = fiveYearsString;
61
62     double fourYears = finalPrice / 48;
63     string fourYearsString = String.Format("{0:0,0.00}", fourYears);
64     lbl4Years.Text = fourYearsString;
65
66     double threeYears = finalPrice / 36;
67     string threeYearsString = String.Format("{0:0,0.00}", threeYears);
68     lbl3Years.Text = threeYearsString;
69
70     double twoYears = finalPrice / 24;
71     string twoYearsString = String.Format("{0:0,0.00}", twoYears);
72     lbl2Years.Text = twoYearsString;
73

```

```

74 if (Convert.ToDouble(ddlYear.Text) >= 2014 && Convert.ToDouble(ddlYear.Text) <= 2018)
75 {
76     lblShowInterest.Text = "ดอกเบี้ย 3.85% ต่อปี";
77 }
78 else if (Convert.ToDouble(ddlYear.Text) >= 2009 && Convert.ToDouble(ddlYear.Text) <= 2013)
79 {
80     lblShowInterest.Text = "ดอกเบี้ย 4.14% ต่อปี";
81 }
82 else
83 {
84     lblShowInterest.Text = "ดอกเบี้ย 4.5% ต่อปี";
85 }
86 }
87 }

```

เลือกสาขา  ค้นหารถจากสาขาของ Tor  
 เลือกยี่ห้อรถ  ค้นหารถจากยี่ห้อ

ค่าน้ำหน้วยอดจัดไฟແນน์และค่าງวด

ปี	<input style="width: 100%; height: 25px; border: none; border-bottom: 1px solid black;" type="button" value="2017"/> dropdownlist	
ราคาเต็ม	<input style="width: 30%; height: 25px; border: 1px solid black;" type="text"/> บาท	
ราคากวน	<input style="width: 30%; height: 25px; border: 1px solid black;" type="text"/> บาท	
<b>ค่าน้ำหน้วยงวด</b>		

#### ดอกเบี้ย

ผ่อน 5 ปี หรือ 60 งวด (ผ่อนเดือนละ)	<input style="width: 100px; height: 25px; border: none; border-bottom: 1px solid black;" type="button" value="แสดงผล"/>	บาท
ผ่อน 4 ปี หรือ 48 งวด (ผ่อนเดือนละ)	<input style="width: 100px; height: 25px; border: none; border-bottom: 1px solid black;" type="button" value="แสดงผล"/>	บาท
ผ่อน 3 ปี หรือ 36 งวด (ผ่อนเดือนละ)	<input style="width: 100px; height: 25px; border: none; border-bottom: 1px solid black;" type="button" value="แสดงผล"/>	บาท
ผ่อน 2 ปี หรือ 24 งวด (ผ่อนเดือนละ)	<input style="width: 100px; height: 25px; border: none; border-bottom: 1px solid black;" type="button" value="แสดงผล"/>	บาท

# TorAutoCar.aspx





11

# LinQ XML <Register>



# Register.xml

```
1 <?xml version="1.0" encoding="utf-8" ?>
2 <Register>
3   <Student>
4     <StudentId>1163109060116</StudentId>
5     <StudentName>นายชินวัตร เลี้นตรง</StudentName>
6     <Status>Enrolled</Status>
7     <Faculty>Faculty of Science and Technology</Faculty>
8     <Program>Information Technology</Program>
9     <Advisor>อนุชา ตุงคำธาราน</Advisor>
10    <AdvisorMobile>0814999435</AdvisorMobile>
11    <Year>2565</Year>
12    <Semmester>2</Semmester>
13    <Subjects>
14      <Subject>
15        <SubjectID>09142302</SubjectID>
16        <SubjectName>Embedded System and Internet of Things Technology</SubjectName>
17        <Section>2</Section>
18      </Subject>
19      <Subject>
20        <SubjectID>09142316</SubjectID>
21        <SubjectName>Information Technology Project 1</SubjectName>
22        <Section>2</Section>
23      </Subject>
24
25    <Subject>
26      <SubjectID>09143322</SubjectID>
27      <SubjectName>Database Application Development Software</SubjectName>
28      <Section>2</Section>
29    </Subject>
30    <Subject>
31      <SubjectID>09143323</SubjectID>
32      <SubjectName>Database Architecture and Administration</SubjectName>
33      <Section>2</Section>
34    </Subject>
35    <Subject>
36      <SubjectID>09143324</SubjectID>
37      <SubjectName>Software Development Process</SubjectName>
38      <Section>2</Section>
39    </Subject>
40    <Subject>
41      <SubjectID>09143331</SubjectID>
42      <SubjectName>Web Services Technology</SubjectName>
43      <Section>2</Section>
44    </Subject>
```

# Register.xml

```
44 <Subject> 67
45   <SubjectID>09143491</SubjectID> 68
46     <SubjectName>Information technology for business</SubjectName> 69
47   <Section>2</Section> 70
48 </Subject> 71
49 <Subject> 72
50   <SubjectID>09144301</SubjectID> 73
51     <SubjectName>Preparation for Professional Experience in Information Technology</SubjectName> 74
52   <Section>1</Section> 75
53 </Subject> 76
54 </Subjects> 77
55 </Student> 78
56 <Student> 79
57   <StudentId>1163109060140</StudentId> 80
58   <StudentName>นายเกียรติพัฒน์ บุญกรงจ่าง</StudentName> 81
59   <Status>Enrolled</Status> 82
60   <Faculty>Faculty of Science and Technology</Faculty> 83
61   <Program>Information Technology</Program> 84
62   <Advisor>อนุชา ตุ้งคี้ช្នាក់</Advisor> 85
63   <AdvisorMobile>0814999435</AdvisorMobile> 86
64   <Year>2565</Year> 87
65   <Semmester>2</Semmester> 88
66 <Subjects> 89
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
```

# Register.xml

```
91 </Subject>
92 <Subject>
93   <SubjectID>09143331</SubjectID>
94   <SubjectName>Web Services Technology</SubjectName>
95   <Section>2</Section>
96 </Subject>
97 <Subject>
98   <SubjectID>09143491</SubjectID>
99   <SubjectName>Information technology for business</SubjectName>
100  <Section>2</Section>
101 </Subject>
102 <Subject>
103   <SubjectID>09144301</SubjectID>
104   <SubjectName>Preparation for Professional Experience in Information Technology</SubjectName>
105  <Section>1</Section>
106 </Subject>
107 </Subjects>
108 </Student>
109 <Student>
110   <StudentId>1163109060157</StudentId>
111   <StudentName>นายกรกช รั่รวงพานิช</StudentName>
112   <Status>Not_Enrolled</Status>
113   <Faculty>Faculty of Science and Technology</Faculty>
114   <Program>Information Technology</Program>
115   <Advisor>อนุชา ตุ้งคิชราณ</Advisor>
116   <AdvisorMobile>0814999435</AdvisorMobile>
117   <Year>2565</Year>
118   <Semmester>2</Semmester>
119 <Subjects>
120   <Subject>
121     <SubjectID>09142302</SubjectID>
122     <SubjectName>Embedded System and Internet of Things Technology</SubjectName>
123     <Section>2</Section>
124   </Subject>
125   <Subject>
126     <SubjectID>09142316</SubjectID>
127     <SubjectName>Information Technology Project 1</SubjectName>
128     <Section>2</Section>
129   </Subject>
130   <Subject>
131     <SubjectID>09143322</SubjectID>
132     <SubjectName>Database Application Development Software</SubjectName>
133     <Section>2</Section>
134   </Subject>
135   <Subject>
136     <SubjectID>09143323</SubjectID>
137     <SubjectName>Database Architecture and Administration</SubjectName>
138     <Section>2</Section>
```

# Register.xml

```
139 </Subject>
140 <Subject>
141   <SubjectID>09143324</SubjectID>
142   <SubjectName>Software Development Process</SubjectName>
143   <Section>2</Section>
144 </Subject>
145 <Subject>
146   <SubjectID>09143331</SubjectID>
147   <SubjectName>Web Services Technology</SubjectName>
148   <Section>2</Section>
149 </Subject>
150 <Subject>
151   <SubjectID>09143491</SubjectID>
152   <SubjectName>Information technology for business</SubjectName>
153   <Section>2</Section>
154 </Subject>
155 <Subject>
156   <SubjectID>09144301</SubjectID>
157   <SubjectName>Preparation for Professional Experience in Information Technology</SubjectName>
158   <Section>1</Section>
159 </Subject>
160 </Subjects>
161 </Student>
162 </Register>
```



# Register.aspx

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.UI;
6  using System.Web.UI.WebControls;
7  using System.Text;
8  using System.Xml;
9  using System.Xml.Linq;
10 
11 public partial class _3ReadXMLSpecificElements : System.Web.UI.Page
12 {
13     protected void Page_Load(object sender, EventArgs e)
14     {
15     }
16 
17     //access all Elements having a Specific Attribute using LINQ to XML
18     protected void btnDisplay_Click(object sender, EventArgs e)
19     {
20         XElement xelement = XElement.Load(Server.MapPath("~/App_Data/Register.xml"));
21         var name = from nm in xelement.Elements("Student")
22                    where (string)nm.Element("Status") == ddlStatus.Text
23                    select nm;
24 
25         Response.Write("Details of Enrolled or Not_Enrolled students:");
26         foreach (XElement xEle in name)
27             Response.Write("<br>" + xEle);
28     }
29 
30     protected void btnDisplay1_Click(object sender, EventArgs e)
31     {
32         XElement xelement = XElement.Load(Server.MapPath("~/App_Data/Register.xml"));
33         var subjects = from subject in xelement.Elements("Student")
34                        where (string)subject.Element("Subjects").Element("Subject").Element("SubjectID") == "0914230"
35                        select subject;
36 
37         Response.Write("Details of a specified subjectID : ");
38         foreach (XElement xEle in subjects)
39             Response.Write(xEle);
40     }
41 
42     protected void btnDisplay2_Click1(object sender, EventArgs e)
43     {
44         XElement xelement = XElement.Load(Server.MapPath("~/App_Data/Register.xml"));
45         Response.Write("List of all Subjects: <br>");
46         foreach (XElement xEle in xelement.Descendants("Subject"))
47         {
48             Response.Write((string)xEle + ", <br>");
49         }
50     }
51 }
```

```
25     foreach (XElement xEle in name)
26         Response.Write("<br>" + xEle);
27     }
28 
29     protected void btnDisplay1_Click(object sender, EventArgs e)
30     {
31         XElement xelement = XElement.Load(Server.MapPath("~/App_Data/Register.xml"));
32         var subjects = from subject in xelement.Elements("Student")
33                        where (string)subject.Element("Subjects").Element("Subject").Element("SubjectID") == "0914230"
34                        select subject;
35 
36         Response.Write("Details of a specified subjectID : ");
37         foreach (XElement xEle in subjects)
38             Response.Write(xEle);
39     }
40 
41     protected void btnDisplay2_Click1(object sender, EventArgs e)
42     {
43         XElement xelement = XElement.Load(Server.MapPath("~/App_Data/Register.xml"));
44         Response.Write("List of all Subjects: <br>");
45         foreach (XElement xEle in xelement.Descendants("Subject"))
46         {
47             Response.Write((string)xEle + ", <br>");
48         }
49     }
50 }
```



# Register.aspx

```
49 protected void btnDisplay3_Click(object sender, EventArgs e)
50 {
51     XElement xelement = XElement.Load(Server.MapPath("~/App_Data/Register.xml"));
52     IEnumerable<string> stuIds = from id in xelement.Elements("Student")
53         let StudentID = (string)id.Element("StudentId")
54         orderby StudentID
55         select StudentID;
56     Response.Write("List and Sort all StudentIds: ");
57     foreach (string sid in stuIds)
58         Response.Write(sid + ", ");
59 }
60
61 protected void btnDisplay4_Click(object sender, EventArgs e)
62 {
63     XElement xelement = XElement.Load(Server.MapPath("~/App_Data/Register.xml"));
64     IEnumerable<string> subIds = from id in xelement.Elements("Student")
65         let SubjectID = (string)id.Element("Subjects").Element("Subject").Element("SubjectID")
66         orderby SubjectID
67         select SubjectID;
68     Response.Write("List and Sort all SubjectIDs: ");
69     foreach (string subid in subIds)
70         Response.Write(subid + ", ");
71 }
72 }
```

List of all Subjects:

09142302Embedded System and Internet of Things Technology2,  
09142316Information Technology Project 12,  
09143322Database Application Development Software2,  
09143323Database Architecture and Administration2,  
09143324Software Development Process2,  
09143331Web Services Technology2,  
09143491Information technology for business2,  
09144301Preparation for Professional Experience in Information Technology1,  
09142302Embedded System and Internet of Things Technology2,  
09142316Information Technology Project 12,  
09143322Database Application Development Software2,  
09143323Database Architecture and Administration2,  
09143324Software Development Process2,  
09143331Web Services Technology2,  
09143491Information technology for business2,  
09144301Preparation for Professional Experience in Information Technology1,  
09142302Embedded System and Internet of Things Technology2,  
09142316Information Technology Project 12,  
09143322Database Application Development Software2,  
09143323Database Architecture and Administration2,  
09143324Software Development Process2,  
09143331Web Services Technology2,  
09143491Information technology for business2,  
09144301Preparation for Professional Experience in Information Technology1,  
1) และแสดงข้อมูลทั้งหมดของนักศึกษาที่ลงทะเบียน/ไม่ลงทะเบียน

2) และแสดงข้อมูลรายวิชาทั้งหมด หากนักศึกษาคนนั้นลงทะเบียน รหัสวิชา 09142302 Embedded System and Internet of Things Technology

3) และแสดงข้อมูลทุกรายวิชา โดยเรียงลำดับรหัสวิชาจากน้อยไปมาก

4) และแสดงข้อมูลเฉพาะรหัสนักศึกษา โดยเรียงลำดับรหัสวิชาจากน้อยไปมาก

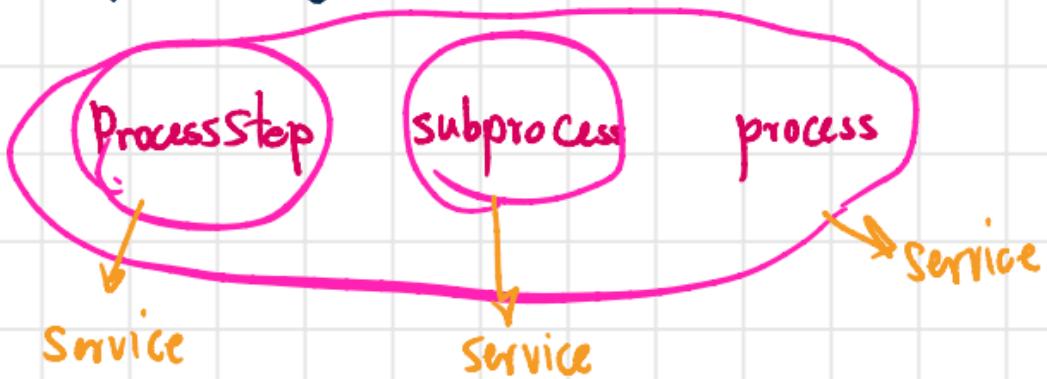


12

SOA

## 1. SOA & Web Service part 3-5

Service Encapsulates Logic = บริการ Service อยู่ด้วยกัน



- สิ่งที่สำคัญของ service 2 ส่วน คือ การเข้าใช้งานผ่าน interface, implementation เขียนโปรแกรม
- ข้อดีการพัฒนาแนวนี้คือถ้าเรามี logic ไฟฟ้ามีความสามารถเขียน implementation ข้างนอกแล้วนำมาทำได้โดยไม่กระทบ interface
- คอนเซป คือ write one (java เขียนครั้งเดียว), access from anywhere (run ที่ไหนก็ได้)
- web service เขียนที่เดียวเมื่อ deploy ไป dic. สามารถ access ที่ไหนก็ได้ผ่าน service interface
- สร้าง service ได้หลายแบบ ทั้งสามแบบเรียกว่า proxy
  - New Service เป็น service ที่พัฒนาใหม่ก่อนเดียว
  - Wrapper Legacy พัฒนา service มาอีกอันแล้วพัฒนาอีกอันมาเชื่อมกันด้วย logic
  - Composite Service มี service อุบัต้องการลิงก์เข้ากับอีก 3 service และนำมาทำเป็น service เดียวกัน



## สรุป Web Service

- ใช้หลักการ SOA ในเชิงบริการไม่ขึ้นอยู่กับภาษา
- เป็น connection ที่ได้รับความนิยม
- implementation ตามสถาปัตยกรรมเชิงบริการ
- เน้นให้บรรลุเป้าหมายเรื่องของ loose coupling
- เจ้าใจ protocol เดิม เช่น HTTP, XML
- เป็นแพลตฟอร์ม independent ไม่ต้องคำนึงถึงว่าจะเชื่อมต่อแพลตฟอร์มอะไร เพราะมีอิสระจากทุกอุปกรณ์ผ่าน service interface



1. ถ้าไม่ใช้ web service จะทำ SOA ได้ไหม?

- ตอบ- เทคโนโลยี เช่น corda, RMI อาศัยหลักการของ service และการ publish Standard interface สามารถทำ service aggregation ได้ เพราะมี technology coupling ตอบสนองหลักการขายของ SOA ได้ระดับนึงเท่านั้น

2. อะไรที่ต้องมีในแอปพลิเคชันจึงถือว่าเป็นไปตาม SOA ถ้า/มีจะไม่เป็น

- ตอบ- ต้องมี service เพราะเป็น building block การออกแบบ service ต้องคำนึงถึงหลักการต่างๆ ของ SOA เช่น การมี loose coupling การมี granularity ที่เหมาะสมกับการให้บริการ

3. ถ้าแอปพลิเคชันเริ่บกใช้งาน 1 service ถือว่าเป็น SOA ไหม?

- ตอบ- ถือว่าเป็นแต่การสร้างแอปและ service ต้องคำนึงถึงหลักการของ SOA

4. service จำเป็นต้อง reuse โดยผู้อื่นหรือไม่ ถ้าไม่สร้างไว้ใช่เองจะถือว่าเป็นไปตามหลักการของ SOA หรือไม่

- ตอบ- การกำหนด service สามารถใช้ reusability เป็นแนวทางได้ หากฟังก์ชันนั้นถูกใช้ซ้ำ หลายแอปจะกำหนดเป็น service แต่ reusability ไม่ใช่สาเหตุเดียวในการสร้าง service



## 5. service, object, software component เมื่อนำรือต่างกันอย่างไร

- ตอบ- หลักการทำงาน service คล้ายกันหลักการทำงานของ Obj และ soft ตรงการติดต่อล่วงปะกอบในระบบผ่าน API ที่แยก  
ออกมายกส่วน implementation และสนับสนุน reuse มีความแตกต่างกันในแง่มุม granularity, reuse

## 6. Service จะมาแทนที่ Obj และ soft ไหม?

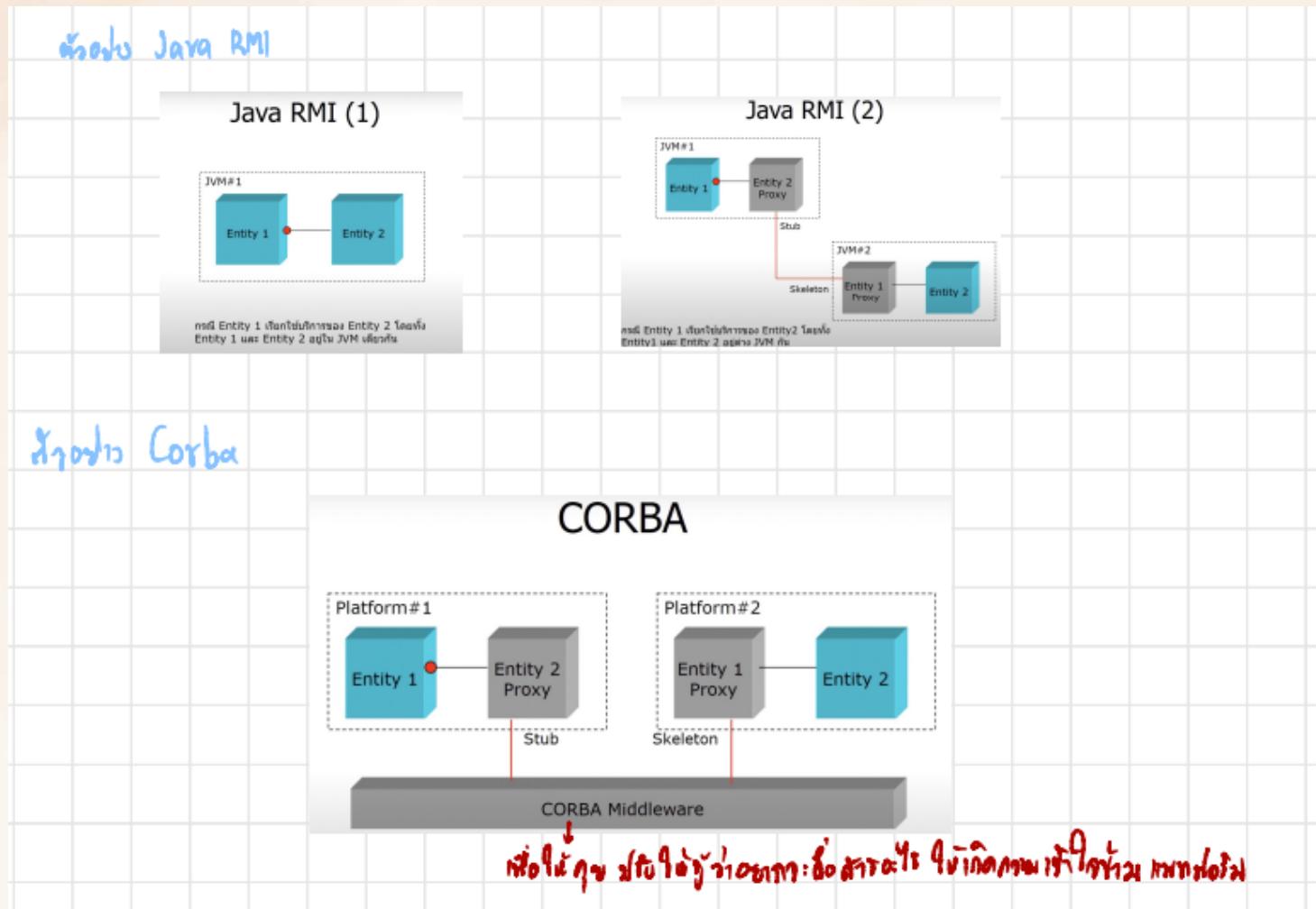
- ตอบ- Service ไม่ได้จะมาแทนที่ Obj และ สามารถใช้ Obj เป็นเทคโนโลยีในการพัฒนาส่วน implementation ของ Service ได้

## 7. ควรเริ่มทำ SOA when?

- ตอบ- องค์กรสามารถปรับตัวและเริ่มทำได้เลยจะทำให้องค์กรตื้มความได้เปรียบ
- เริ่มจากการทำ Pilot Project เล็กๆ โดยทำเป็น Intra system Service และขยายเป็น Internal Service และใหญ่ขึ้นเป็น External Service

## SOA : Related Technology ขยายความ 50A ต่อจากเดิม

- RPC-Remote Produced Call เป็นบริการจาก entity ที่อยู่นอกขอบเขตของระบบล้วนใหญ่ ว่าเป็น Proxy, servier สมัยก่อน จะเป็น JAVA, RMI, MS COM
- ทำงานอย่างไร



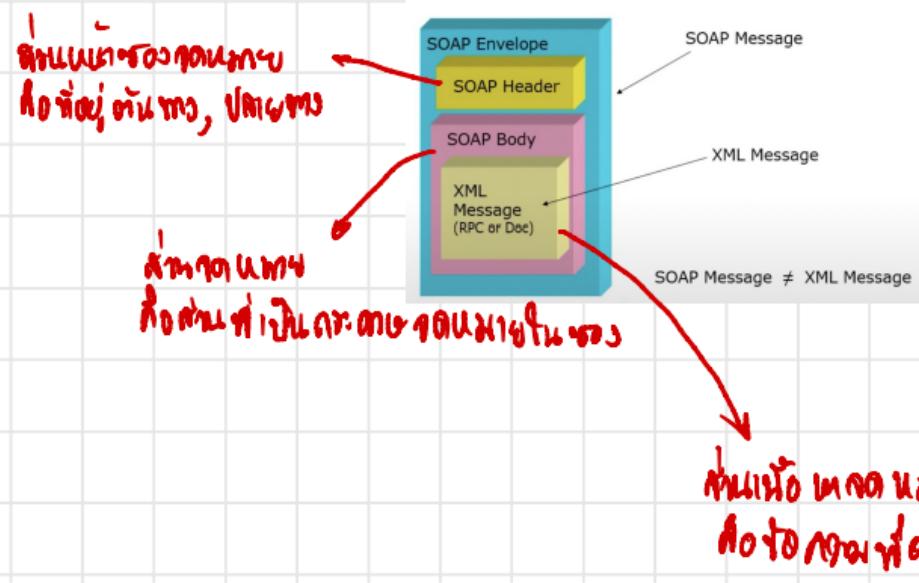
## XML គីឡូម៉ា

- เป็นมาจากการ Extensible Markup Language
  - เป็น Markup Language มี Tag คล้ายกับ HTML
  - ออกแบบมาให้ใช้อธิบายข้อมูลต่างๆ ผู้ใช้งานนั้นได้เอง
  - ประกอบด้วย XML Document และ XML schema
  - ใช้ไฟล์ Document Type Definition หรือ XML Schema อธิบายโครงสร้าง Tag ที่ใช้ใน XML Document
  - ได้รับการดูแลโดย W3C

# SOAP គីឡូវ៉ា

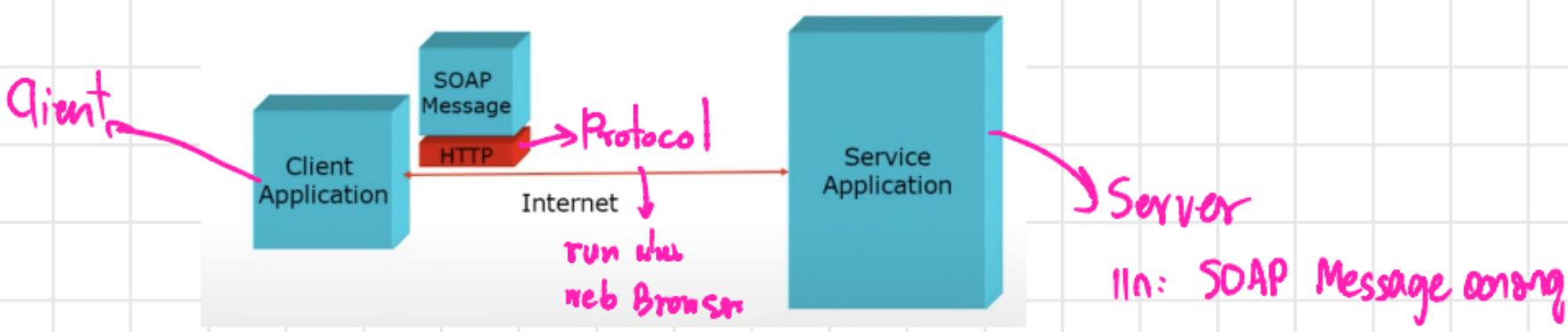
- เป็น Protocol ในการสื่อสารระหว่าง Application
  - กำหนดรูปแบบในการส่งข้อมูลความผ่าน Internet
  - ไม่ซึ้นกับแพลตฟอร์มและภาษาคอมพิวเตอร์ที่พัฒนา
  - มีโครงสร้าง W XML
  - SOAP Message ส่งข้อมูลความผ่านอินเทอร์เน็ตเมื่อใช้ร่วมกับ web ส่งผ่าน Firewall ได้
  - ได้รับการดูแลโดย W3C

## โครงสร้าง SOAP Message



\* SOAP Message = ห้องนักเรียน XML Message  
\* XML Message = ห้องนักเรียนของคุณ Message

## การส่ง SOAP Message

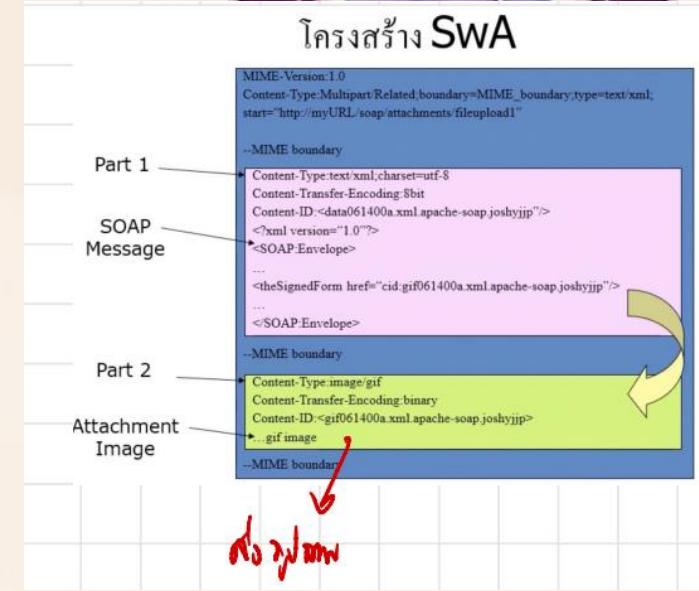
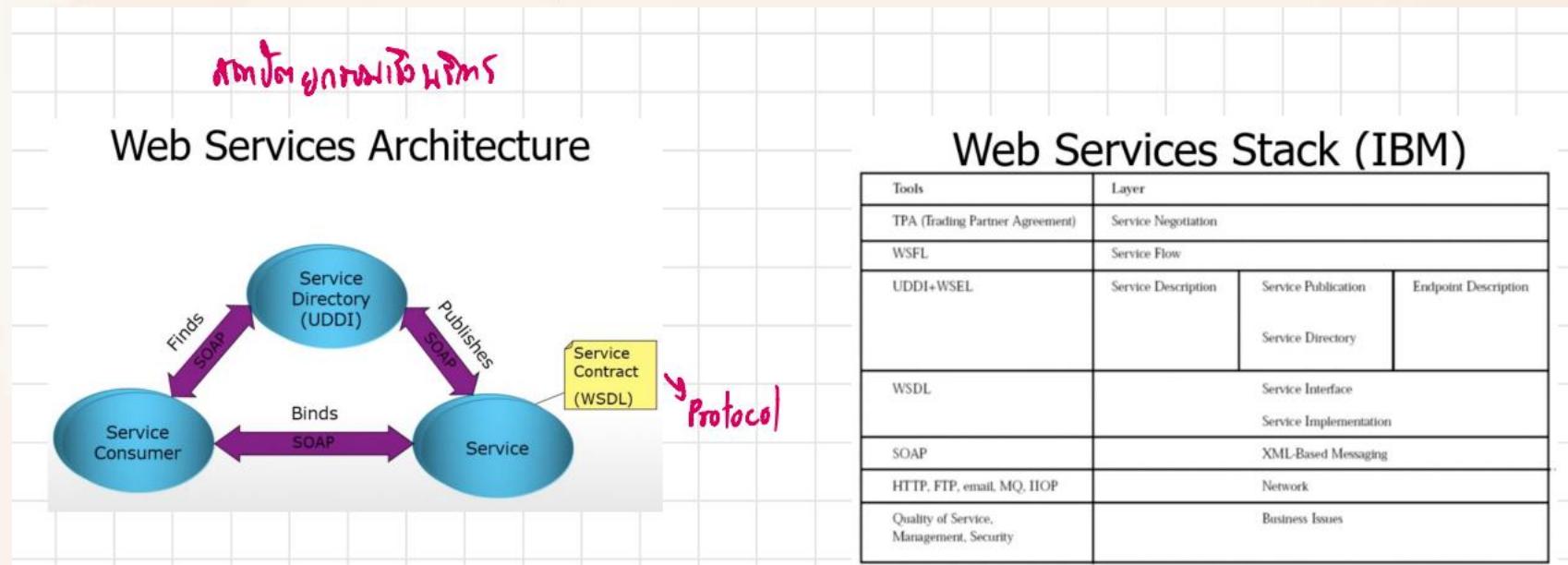


### 3. SOAP with Attachment (SWA)

- สามารถส่งรูปได้ไม่จำเป็นต้องเป็นข้อความ Image ,MP3
  - เป็นการส่งขนาดใหญ่
  - อาศัย URL เพื่อลิงก์ไปล้วนที่เอกสารแทบทุกที่ส่งไปด้วยกัน
  - ได้รับการดูแลโดย W3C

# Web Service គីវអ៊វីរ

- เป็นทางเลือกของ Connection Technology ในการ Implement SOA
  - ใช้มาตรฐานและเทคโนโลยีร่วมสมัยพื้นฐาน XML, XML Schema, SOAP, WSDL, UDDI, HTTP transport
  - 3. มีการกำหนดให้ Web Service Architecture และ Web service Stack เพื่อขยายความสามารถในอนาคต



## บทสรุป

- Web service และ ESB เป็นเทคโนโลยีที่ได้รับความนิยมในการ implement หลักการของ SOA ในองค์กร
- มาตรฐานและเทคโนโลยีที่เกี่ยวข้องในการพัฒนาแบบ SOA มีมากหลายเพิ่มขึ้นเรื่อยๆ

## Enterprise Service Bus คือ

- เป็นตัวสนับสนุน implement SOA
- มี architecture เอ็มบีแอปพลิเคชัน และ resource ทำงานร่วมกันแบบองค์รวมในระดับ องค์กรธุรกิจ
- มีช่องทางในการต่างๆ เชื่อมต่อได้แบบ Integrated Bus ทั้งที่เป็น SOA และ Non-SOA โดยใช้มาตรฐานกลางร่วมกัน
- มีระบบจัดการส่วนกลางให้เช่น Security, QOS
- มี ESB Basic Architecture เพื่อให้ผู้พัฒนาสามารถใช้อ้างอิงได้