INN570 2013 Week 9 Prac Exercises

Birthday Book with XML files

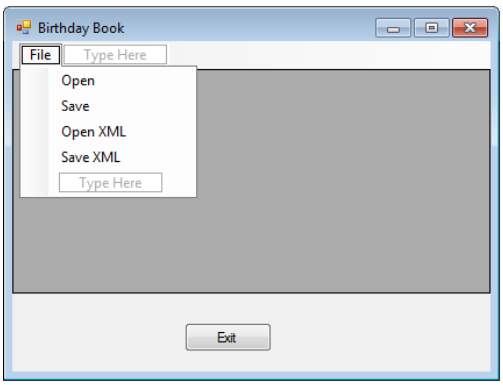
XML files can be easily transferred between devices and very often used to exchange data between different software systems. In this practical exercise, we will modify the Birthday Book to make it support XML file open and save.

Download the Birthday Book application as it is packaged for the exercise. Now we want to open and save XML files in this application.

Add the following line into code *BBook.cs*:

```
using System.Xml;
```

- **Add new items in the menuStrip container**
  - Add a new menu item called "Open XML" in the menuStrip.
  - Add another menu item called "Save XML" in the menuStirp.



- **Access to XML files using *DataTable* object**

  Microsoft .NET framework provides facilities for accessing XML files. One of the facilities is *DataTable* class which is included in *System.Data* namespace. In fact, the *System.Data* namespace contains the classes (*DataTable* is one of the classes) that represent the ADO.NET architecture. ADO.NET provides developers with facilities to build software components that efficiently manage data from multiple data sources including XML documents.

  In the simple Birthday Book application, the object used to store the birthday book data is an object of *DataTable* . Now, we are going to use methods `ReadXml()` and `WriteXml()` of *DataTable* to read from or save to XML files.

  - Double click on the "Open XML" item, Visual Studio will automatically generate "Click" event handler method `openXMLToolStripMenuItem_Click`. Add the following code into the method.

```
private void openXMLToolStripMenuItem_Click(object sender, EventArgs e)
 {
   OpenFileDialog openxml = new OpenFileDialog();
   openxml.Reset();
   openxml.Filter = " xml (*.xml)|*.xml";
   if (openxml.ShowDialog() == DialogResult.OK)
   {
       m_BBookTable.Clear();
      //using method ReadXml to read the XML file
      m_BBookTable.ReadXml(openxml.FileName);
   }
}
```

- Double click on the "Save XML" item, add the following code into method *saveXMLToolStripMenuItem_Click*.

```
private void saveXMLToolStripMenuItem_Click(object sender, EventArgs e)
{
     SaveFileDialog savetoxml = new SaveFileDialog();
     savetoxml.Reset();
     savetoxml.Filter =  "xml (*.xml)|*.xml";
       if (savetoxml.ShowDialog() == DialogResult.OK)
       {
           //using method WriteXml to save the XML file
           m_BBookTable.WriteXml(savetoxml.FileName);
       }
}
```

- Run the application, set Culture to *English* (*AU*), i.e., *en-AU*, and enter the following entries onto the birthday book table:

  Sam, June 12 1985, 200

  Janet, April 5 1990, 400

  Tony, November 5 1977, 1000

  Then, save it as an XML file. The content of the saved XML file should look like this.

```
<?xml version="1.0" standalone="true"?>
- <DocumentElement>
  - <Birthdays>
      <Name>Sam</Name>
      <Birthday>1985-06-12T00:00:00+10:00</Birthday>
      <Cost>200</Cost>
    </Birthdays>
  - <Birthdays>
      <Name>Janet</Name>
      <Birthday>1990-04-05T00:00:00+10:00</Birthday>
      <Cost>400</Cost>
    </Birthdays>
  - <Birthdays>
      <Name>Tony</Name>
      <Birthday>1977-11-05T00:00:00+10:00</Birthday>
      <Cost>1000</Cost>
    </Birthdays>
</DocumentElement>
```

Then select different cultures, for example *en-US* and *de*, and save the birthday table to different XML files. Open these XML files by double clicking on them, you will find that they are the same no matter what Culture that you have chosen.

Although the *Datatable* object of ADO.net allows you easily to read or write an XML file as a whole, but it does not allow you to customise the file, for example, it doesn't allow you to save the culture information into the XML file.

- **Storing culture information into XML files**

The *System.Xml* namespace in .NET Framework provides classes particularly for processing XML documents and these classes allow us to get into the details of XML documents. In the following code, we will use an object of class *XmlDocument* provided in *System.Xml* to read or write XML files at element level.

- Replace the `openXMLToolStripMenuItem_Click` method with the following code:

```
private void openXMLToolStripMenuItem_Click(object sender, EventArgs e)
 {
     OpenFileDialog openxml = new OpenFileDialog();
     openxml.Reset();
     openxml.Filter = " xml (*.xml)|*.xml";
     if (openxml.ShowDialog() == DialogResult.OK)
     {
         m_BBookTable.Clear();

          //doc is an object of XmlDocument
         XmlDocument doc = new XmlDocument();
         doc.Load(openxml.FileName);

         if (doc.SelectSingleNode("//DocumentElement//Culture") != null)
         {
             //get 'Culture' from the XML file
             string CultureName =
             doc.SelectSingleNode("//DocumentElement//Culture").InnerText;

             //set the CurrentCulture with the culture from the XML file
             Thread.CurrentThread.CurrentCulture = new CultureInfo(CultureName);
         }
          //nodelist contains the data from the XML file
         XmlNodeList nodelist = doc.SelectNodes("//DocumentElement//Birthdays");

         //add the data to the birthday table m_BBookTable row by row
         for (int i = 0; i < nodelist.Count; i++)
         {
             string name = nodelist[i].SelectSingleNode("Name").InnerText;
             DataRow newrow = m_BBookTable.NewRow();
             if (nodelist[i].SelectSingleNode("Name") != null)
             {
                 newrow["Name"] = name;
             }
             if (nodelist[i].SelectSingleNode("Birthday") != null)
             {
                 newrow["Birthday"] =
Convert.ToDateTime(nodelist[i].SelectSingleNode("Birthday").InnerText);
             }
             if (nodelist[i].SelectSingleNode("Cost") != null)
             {
                 newrow["Cost"] =
Convert.ToDecimal(nodelist[i].SelectSingleNode("Cost").InnerText);
             }
             m_BBookTable.Rows.Add(newrow);
         }
     }
 }
```

- Replace the `saveXMLToolStripMenuItem_Click` method with the following code:

```
 private void saveXMLToolStripMenuItem_Click(object sender, EventArgs e)
```

```
{
    SaveFileDialog savetoxml = new SaveFileDialog();
    savetoxml.Reset();
    savetoxml.Filter = " xml (*.xml)|*.xml";
    if (savetoxml.ShowDialog() == DialogResult.OK)
    {
        //doc is an XmlDocument object
        XmlDocument doc = new XmlDocument();
        // docelements is an XmlElement object
        XmlElement docelements = doc.CreateElement("DocumentElement");

        //CultureElement is an XmlElement object used to store the current culture
        XmlElement CultureElement = doc.CreateElement("Culture");
        //get the current culture and assign it to CultureElement
        CultureElement.InnerText = Thread.CurrentThread.CurrentCulture.Name;
        //add the culture element to docelements
        docelements.AppendChild(CultureElement);


       //get data from the birthday table and add the data the XmlDocument object
        foreach (DataRow dr in m_BBookTable.Rows)
        {
            XmlElement birthdays = doc.CreateElement("Birthdays");
            XmlElement NAME = doc.CreateElement("Name");
            XmlElement BIRTHDAY = doc.CreateElement("Birthday");
            XmlElement Cost = doc.CreateElement("Cost");
            NAME.InnerText = dr[0].ToString();
            BIRTHDAY.InnerText = string.Format("{0:s}", dr[1]);
            Cost.InnerText = dr[2].ToString();
            if (dr[0] != DBNull.Value)
            {
                birthdays.AppendChild(NAME);
            }
            if (dr[1] != DBNull.Value)
            {
                birthdays.AppendChild(BIRTHDAY);
            }
            if (dr[2] != DBNull.Value)
            {
                birthdays.AppendChild(Cost);
            }
            docelements.AppendChild(birthdays);

        }
        doc.AppendChild(docelements);
        doc.Save(savetoxml.FileName);
    }
}
```

- Run the application and repeat the previous test, i.e., set Culture to *English* (*AU*) and enter the following entries onto the birthday book table:

  Sam, June 12 1985, 200

  Janet, April 5 1990, 400

  Tony, November 5 1977, 1000

Then select different cultures, for example *en-US* and *de*, and save the birthday table to different XML files. Open these XML files and you will find that they contain a Culture element with different culture settings in different files.

For example, with the culture set to *de*, the XML file would look like the following:

```
<?xml version="1.0"?>
- <DocumentElement>
    <Culture>de</Culture>
  - <Birthdays>
      <Name>Sam</Name>
      <Birthday>1985-06-12T00:00:00</Birthday>
      <Cost>200</Cost>
    </Birthdays>
  - <Birthdays>
      <Name>Janet</Name>
      <Birthday>1990-04-05T00:00:00</Birthday>
      <Cost>400</Cost>
    </Birthdays>
  - <Birthdays>
      <Name>Tony</Name>
      <Birthday>1977-11-05T00:00:00</Birthday>
      <Cost>1000</Cost>
    </Birthdays>
</DocumentElement>
```

When you open this XML file, the birthday book form would look like: