XML in VB.net

XML is a general purpose tag based language and very easy to transfer and store data across applications. Like HTML , XML is a subset of SGML - Standard Generalized Markup Language. XML is a platform independent language, so the information formatted in XML can be used in any other platforms (Operating Systems). XML is a self-describing language and it gives the data as well as the rules to identify what information it contains.

XML files are made up of tags that contains data. Generally a start tag and end tag to hold the data. For example, if you want to create an XML tag name "Header" , the start tag is like ***< Header >***and the end tag is like ***< /Header >***. We can fill our information between these tags.

***< Header >***Header Content Here ***< /Header >***

While creating an XML file , some important points have to remember :

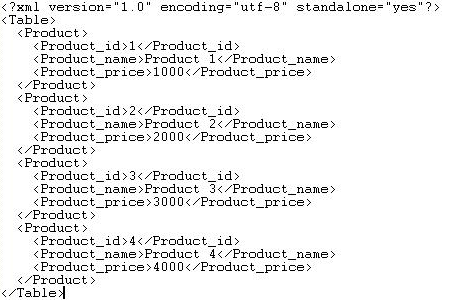
\* XML is case sensitive

ex:***< Header >***is not same as ***< HeadeR >***.

\* Tags must be closed in the reverse order that they were opened

ex : ***< first-tag >< second-tag >***Data here ***< /second-tag > < /first-tag >***

## Sample XML file

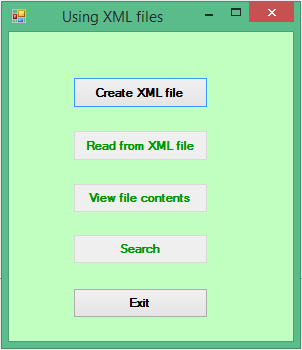


The .Net technology is widely supported XML file format. The .Net Framework provides the Classes for read, write, and other operations in XML formatted files . These classes are stored in the namespaces like System.Xml, System.Xml.Schema, System.Xml.Serialization, System.Xml.XPath, System.Xml.Xsl etc. The Dataset in ADO.NET uses XML as its internal storage format.

You can use any text editor to create an XML file . More over XML files are readable by humans as well as computers. From the following links you can see how to use XML in VB.NET.

# The interface

For this program I have created a simple interface of five buttons:

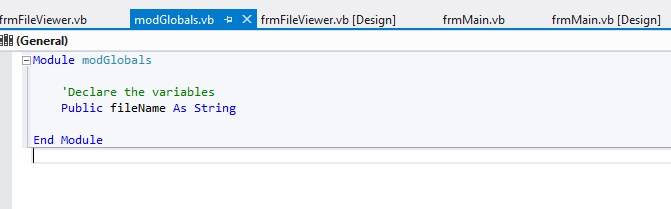


* btnCreate
* btnRead
* btnView
* btnSearch
* btnExit

Three of the buttons have had their *enabled* property set to false as we want the user to create their XML file before it can be read, viewed or searched. In the procedure associated with the button for creating the file I have set all three buttons’ *enabled* property to true.

# How to set global variables

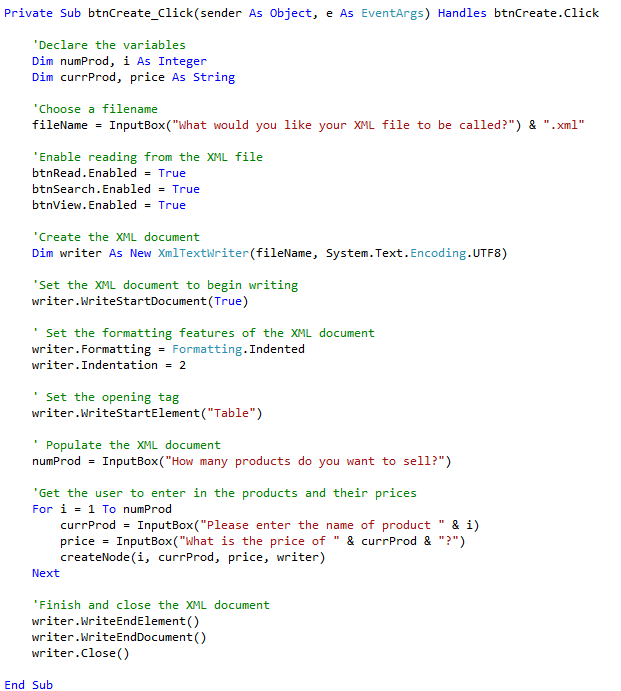
After creating my main form (frmMain.vb), I have also added a module by clicking on Project>Add module. I have called this modGlobals.vb and use it to store a global variable which allows me to have the string variable *fileName* recognised across all forms, allowing the user to set the XML files’ name and to also allow a form to read the contents of the file into a listbox.



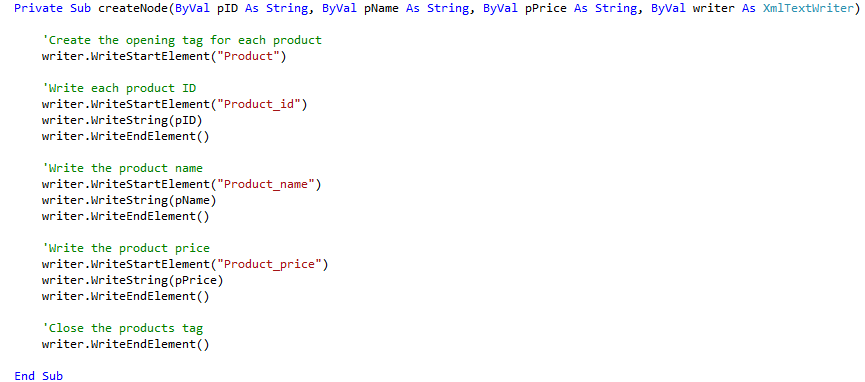
# How to create an XML file in VB.net

XML is a platform independent language, so the information formatted in XML can be used in any other platforms (Operating Systems). If we create an XML file in one platform it can be used in other platforms also.

For creating a new XML file in VB.NET, we are using ***XmlTextWriter*** class . The class takes FileName and Encoding as argument. Also we are here passing formatting details . The following source code creates an XML file and allows the user to name the file and how many products are to be added. filename is a global variable declared in a separate module.

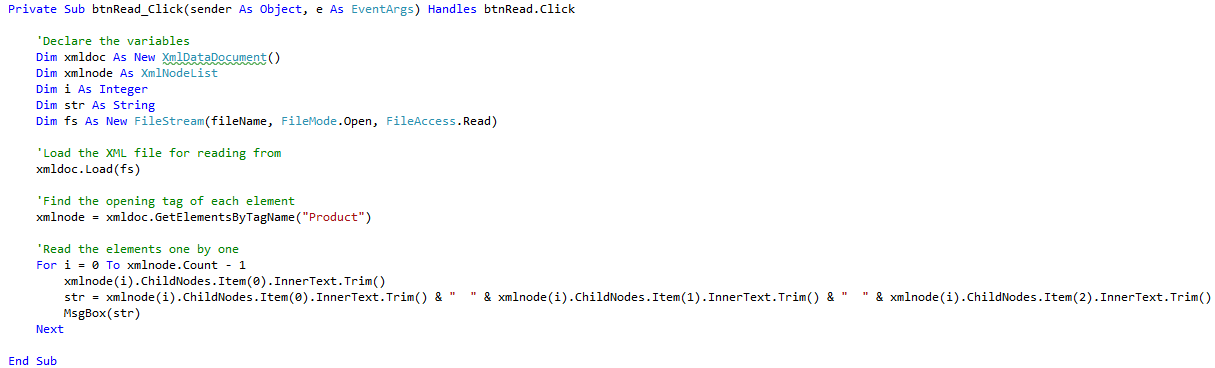


createNode is a separate procedure that is called once for each product that the user wishes to enter. It sends a product ID (Increments numerically starting at 1), the product’s name, its price and that it is being written to an XML file.



# Reading from an XML file

In the previous program we create an **XML** file and let the user name it. We saved it to a string variable called fileName. The following program read that file and extract the contents inside the XML tag to display to the user one item at a time. We can read an XML file in several ways depends on our requirement. This program read the content in Node wise. Here we are using XmlDataDocument class to read the **XML** file. In this program it search the Node < Product > and its child Nodes and extract the data in child nodes.



# Searching an XML file

XML files are made up of tags that contains information. The .Net technology is widely supported XML file format. Also the Dataset in ADO.NET uses XML format as its internal storage format.

The following source code shows how to search an item in an XML file using Dataset. Here Dataset using an XmlReader for read the content of the file. Locate the XML file using XmlReader and pass the XmlReader as argument of Dataset. By using the Dataset, search the product Product2 in the file Product.XML with the help of DataView.

