Recent

Topics

Tutorials

Highlights

Settings

Feedback (http://community.

Sign Out

Loading and creating XML documents (XMLUtil)

The `XMLUtil` object is a utility that provides methods to do the follow

- Create an XML wrapper
- Load an existing XML file to an XML wrapper

Similar to other wrapper classes provided by UFT (such as those for Test Objects), it is convenient to simplify the usage of XML data via code. It provides an array of methods that cover the most frequently used functions of `XMLDOM`. Of course, if one needs specific functions not provided by the wrapper, it is always possible to recur to the `XMLI` or `LINQ` object model (which is not covered in this book).

For example, to get a specific node by ID, we shall use the native method `GetElementByID`.

Getting ready

From the **File** menu, navigate to **New** | **Test**, or use the *Ctrl + N* shortcut.

How to do it...

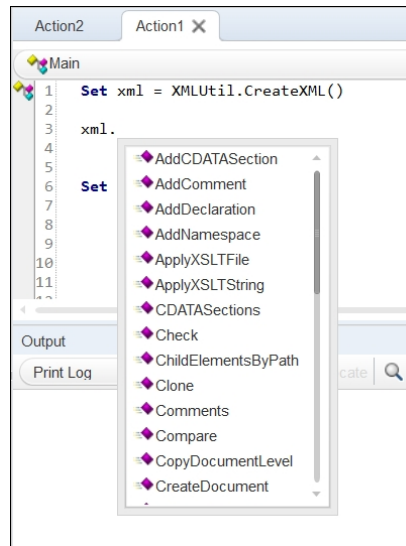In the `Action` object of the test, write the following code:

```
Set xml = XMLUtil.CreateXML()
```

The `xml` variable is set to a new instance of the `MicXmlData` class (see a more detailed description in the following screenshot). The instance will be empty with regard to data. To create an instance with data from an XML file, use code such as the following:

```
Set xml = XMLUtil.CreateXMLFromFile(XMLFilePathname)
```

Where `XMLFilePathname` is a variable containing the full path to an XML file.

The following screenshot shows a partial list of the methods of the `MicXmlData` object that can be used to load and manipulate XML documents:

For example, it is possible to use a `MicXmlData` object to create an XML document. During runtime, add data on the go, as it is retrieved from the AUT, and later, load an XML file storing the expected results data and perform a comparison. It is true that in general, an Excel file serving as a `DataTable` would just do. However, if the data needs to be structured in a more complex fashion, then an XML document is more suitable.