**DMU Institute of Technology**

**Information Technology academic program**

Integrative programming Laboratory

**LAB SESSION 1: Creating an XML document**

1. Create the following XML document using any text editor – using the XML editor installed on our Lab is recommended (editix2019), save the document as \*.xml (e.g. Breakfastmenu.xml) and view it with different web browsers

<?xml version="1.0" encoding="UTF-8"?>

<**breakfast\_menu**>

<**food**>

<**name**>Belgian Waffles </name>

<**price**>$5.95</price>

<**description**>Two of our famous Belgian Waffles with plenty of real maple syrup</**description**>

<**calories**>650</**calories**>

</**food**>

<**food**>

<**name**>Strawberry Belgian Waffles</**name**>

<**price**>$7.95</**price**>

<**description**>Light Belgian waffles covered with strawberries and whipped cream</**description**>

<**calories**>900</**calories**>

</**food**>

<**food**>

<**name**>Berry-Berry Belgian Waffles</**name**>

<**price**>$8.95</price>

<**description**>Light Belgian waffles covered with an assortment of fresh berries and whipped

cream</**description**>

<**calories**>900</**calories**>

</**food**>

<**food**>

<**name**>French Toast</**name**>

<**price**>$4.50</**price**>

<**description**>Thick slices made from our homemade sourdough

bread</**description**>

<**calories**>600</**calories**>

</**food**>

<**food**>

<**name**>Homestyle Breakfast</**name**>

<**price**>$6.95</**price**>

<**description**>Two eggs, bacon or sausage, toast, and our ever-popular hash

browns</**description**>

<calories>950</**calories**>

</**food**>

</**breakfast\_menu**>

1. Checking syntax and well-formedness of the above XML document.

Examples to check

1. Write the XML declaration as the following, missing the question mark at the end and view the update on the browser.

<?xml version="1.0" encoding="UTF-8">

1. Change the XML version number to 2.0 and see what will happen.
2. Remove the root element and see what will happen?
3. Remove the quotation mark of the **utf-8** encoding attribute value and see what will happen.
4. Remove the **name** and **price** tag of the first food element and see what will happen.
5. Rewrite the first name close as **</ name>**. Note the space between **</** and **name.**
6. Make the name closing tag as </Name>. Note the **N** capital letter.
7. Create a new food node with no any child elements as **<food/>** which is a sibling of the existing food node.
8. **Handling Special characters in XML:** observe what happen to the following file when you view on the browser and then if you encounter any inconveniency, fix the errors and rewrite the markup again to produce the correct document.

<?xml version=”1.0” encoding=”utf-8”>

<examples1>

<name fname=”Asnake” lname=”Worku” fname=”Alazar” lname=”Mingizem Tamiru”></name>

<compare:1>John said “I am the best XML programmer ”</comparison:1>

<compare:2> 6<7 & 8>1</compare:2>

</examples1>

<examples2>

<!-- This is an example of empty element -->

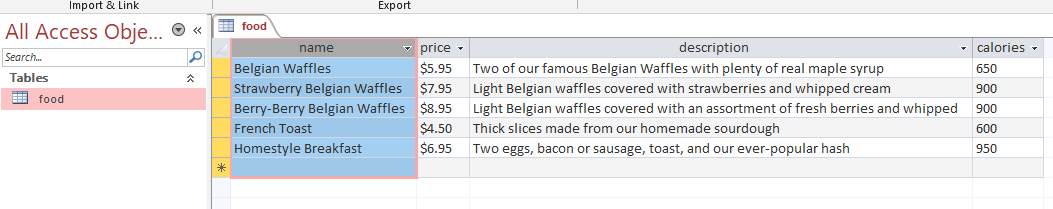
</examples2>

Use entity references as well as CDATA section.

1. Import and Export XML into and from excel and access files
2. Import the XML data in question no. 1 into access database.

Steps:

1. Open Ms. Access and create a new blank database.
2. **Import XML data:** click on ‘**external** **data’** tab
3. Click on New Data Source 🡪from file 🡪 XML file
4. Browse the XML file you want to import
5. Click OK 🡪 access wizard you the table structure it has created from file
6. Choose an import option (you can leave default) 🡪 click Ok.

We have created a food table in a blank database from Q#1 as follows