Assignment No. : 3

TITLE

Title: XML and CSS/XSL

PROBLEM STATEMENT

Design the XML document to store the information of the employees of any business organization and demonstrate the use of:
a) DTD
b) XML Schema
And display the content in (e.g., tabular format) by using CSS/XSL.

OUTCOMES

Students will be able to,

1. Design static webpage using XML.
2. Apply CSS/XSL to XML pages.

SOFTWARE & HARDWARE REQUIREMENTS

Software: Notepad, Any Browser

THEORY-CONCEPT

XML stands for Extensible Markup Language. It is nothing but the text-based markup language which is derived from Standard Generalized Markup Language(SGML).

XML tags identify the data and are used to store and organize the data, rather than specifying how to display it like HTML tags, which are used to display the data. XML is not going to replace HTML in the near future, but it introduces new possibilities by adopting many successful features of HTML.

There are three important characteristics of XML that make it useful in a variety of systems and **solutions −**

XML is extensible − **XML allows you to create your own self-**descriptive tags, or language, that suits your application.

XML carries the data, does not present it − **XML allows you to store the data irrespective** of how it will be presented.

XML is a public standard − **XML was developed by an organization called the World Wide** Web Consortium (W3C) and is available as an open standard.

TECHNOLOGY/TOOL

The XML document have an XML declaration, but it is optional, and it is written as−

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

Where version is nothing but the version of an XML document and UTF specifies the character-encoding used in the document.

Each XML-**element needs to be closed either with start or with end elements as shown below −**

<element>**……**</element>

An XML document can have only one root element.

<root>

<x>...</x>

<y>...</y>

</root>

XML Attributes:

Using a name/value pair, an attribute specifies a single property for an element. An XML-element can have one or more attributes. For example −

XMLTutorial

Here href is the attribute name and http://www.google.com/ is attribute value.

XML DTD:

DTD stands for Document Type Definition.

A DTD defines the structure and the legal elements and attributes of an XML document.

The purpose of a DTD is to define the structure and the legal elements and attributes of an XML document:

.xml	.dtd
<?xml version="1.0" encoding="UTF-8"?> <!DOCTYPE note SYSTEM "Note.dtd"> <note> <to>Tove</to> <from>Jani</from> <heading>Reminder</heading> <body>Don't forget me this weekend!</body> </note>	<!DOCTYPE note [<!ELEMENT note (to,from,heading,body)> <!ELEMENT to (#PCDATA)> <!ELEMENT from (#PCDATA)> <!ELEMENT heading (#PCDATA)> <!ELEMENT body (#PCDATA)>]>

The DTD above is interpreted like this:

!DOCTYPE note - Defines that the root element of the document is note

!ELEMENT note - Defines that the note element must contain the elements: "to, from, heading, body"

!ELEMENT to - Defines the to element to be of type "#PCDATA"

!ELEMENT from - Defines the from element to be of type "#PCDATA"

!ELEMENT heading - Defines the heading element to be of type "#PCDATA"

!ELEMENT body - Defines the body element to be of type "#PCDATA"

XML Schema:

An XML Schema describes the structure of an XML document, just like a DTD.
An XML document with correct syntax is called "Well Formed".
An XML document validated against an XML Schema is both "Well Formed" and "Valid".

```
<xs:element name="note">

<xs:complexType>
  <xs:sequence>
    <xs:element name="to" type="xs:string"/>
    <xs:element name="from" type="xs:string"/>
    <xs:element name="heading" type="xs:string"/>
    <xs:element name="body" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

</xs:element>
```

The Schema above is interpreted like this:
<xs:element name="note"> defines the element called "note"
<xs:complexType> the "note" element is a complex type
<xs:sequence> the complex type is a sequence of elements
<xs:element name="to" type="xs:string"> the element "to" is of type string (text)
<xs:element name="from" type="xs:string"> the element "from" is of type string
<xs:element name="heading" type="xs:string"> the element "heading" is of type string
<xs:element name="body" type="xs:string"> the element "body" is of type string

DESIGN/EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Write the XML code in notepad and save with .xml extension.
2. Write the CSS /XSL code in notepad and save with .css/.xsl extension.
3. Import CSS/XSL file in XML page.
4. Open XML page in the browser.

TEST CASES

Manual testing is used to check whether CSS gets applied or not.

CONCLUSION/ANALYSIS

Hence, we have designed static web page using XML and CSS

SAMPLE PROGRAM CODE & OUTPUT

Book.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/css" href="book_catalog.css"?>
<CATALOG>
<BOOK>
<TITLE>Database Management System</TITLE>
<AUTHOR>Korth</AUTHOR>
<PRICE>500</PRICE>
<YEAR>1985</YEAR>
</BOOK>
<BOOK>
<TITLE>Computer Network</TITLE>
<AUTHOR>Tenenbaum</AUTHOR>
<PRICE>600</PRICE>
<YEAR>1985</YEAR>
</BOOK>
<BOOK>
<TITLE>Software Engineering and project Management</TITLE>
<AUTHOR>Roger Pressman</AUTHOR>
<PRICE>600</PRICE>
<YEAR>1985</YEAR>
</BOOK>
```

book_catalog.css

```
BOOK {
    Display: block;
     margin-left:0; magin-
bottom: 30pt;
 } CATALOG {
   Width:100%;
   background-color: #ffffff;
  } TITLE {
   Color:  ff0000;  display:
   block;font-size: 20pt;
}
AUTHOR {
   display:   block;   color:
   #0000ff;font-size: 20pt;
}
YEAR, PRICE {
    Color:#00000; Display:
    block; Margin-left: 20pt;
}
```

Output:

Database Management System
Korth
500
1985

Computer Network
Tenenbaum
600
1985

Software Engineering and project Management
Roger Pressman
600
1985

EXPECTED OUTPUT

1	Department	Employee ID	Employee	Title	Manger ID	Type
2	Finance	ID4	Gary Miller	Analyst	ID6	Staff
3	Finance	ID12	Richard Elliot	Assistant	ID20	Assistant
4	Finance	ID20	Roger Mun	Division Lead	ID1	Manager
5	Finance	ID28	Daniel Garrett	Division Lead	ID1	Manager
6	Finance	ID25	Paul Hill	Analyst	ID6	Staff
7	Finance	ID1	Crystal Doyle	Finance Director		Manager
8	Finance	ID15	Betina Bauer	Analyst	ID2	Staff
9	Finance	ID2	Daniela Schreiber	Department Lead	ID28	Manager
10	Finance	ID11	Dan Ziegler	Reporting	ID2	Staff
11	Finance	ID26	Robert Richardson	Reporting	ID2	Staff
12	Finance	ID6	Robert Blume	Department Lead	ID20	Manager
13	Finance	ID31	Lukas Hofer	Assistant	ID6	Assistant
14	Finance	ID3	Natalie Porter	Projects	ID2	Staff
15	Finance	ID29	Andre Cooper	Projects	ID2	Staff

ORAL QUESTIONS

1. Explain difference between HTML and XML?
2. What is XML DOM?
3. Explain difference between CDATA and PCDATA?
4. What is mean by simple element and complex element?
5. What is XPATH?
6. Explain XSL and XSLT?

Assignment No. - 4

TITLE

HTML, CSS,Java Script

OBJECTIVES

1. Understand about basic concepts of JavaScript.
2. Use JavaScript for validation of data.

PROBLEM STATEMENT

Implement an application in Java Script using following:
a) Design UI of application using HTML, CSS etc.
b) Include Java script validation
c) Use of prompt and alert window using Java Script

e.g., Design and implement a simple calculator using Java Script for operations like addition, multiplication, subtraction, division, square of number etc.
a) Design calculator interface like text field for input and output, buttons for numbers and operators etc.
b) Validate input values
c) Prompt/alerts for invalid values etc.

OUTCOMES

Students will be able to,

1. Design static webpage using HTML,CSS
2. Apply JavaScript to HTML pages for validation of data.

SOFTWARE & HARDWARE REQUIREMENTS

Software's: Text Editor, Any Browser

JavaScript is a programming language of HTML as well web. It is preferred for creating network-centric applications.It is integrated and complimentary with Java. As JavaScript is integrated with HTML it is very easy to implement. It is open as well as cross-platform.

Advantages:

The advantages **of using JavaScript are −**

It requires less server interaction

Immediate feedback to the visitors

Increased interactivity

Richer interfaces

Validation:

When client enters the all necessary data and press the submit button form validation is done at server side If data entered by a client is incorrect or missing,the server needs to send all data back to the client and request for resubmission of form with correct information. This is really a lengthy process which puts a lot of load(burden) on the server.

So, JavaScript provides a way to validate form's data on the client's side itself before sending it to the web server. Form validation performs two functions-

Basic Validation –First of all the form must be checked to make sure all the mandatory fields are filled in. It would require just a loop through each field in the form and check for the data.

Data Format Validation − **Secondly, the data that is entered must be checked for correct** format and its value. The code must include appropriate logic to test correctness of data.

TECHNOLOGY/TOOL

JavaScript can be implemented using JavaScript statements that are placed within the <script>.

You can place the <script> tags, containing your JavaScript, anywhere within your web page, but it is normally recommended that you should keep it within the <head> tags.

The script tag takes two important attributes:

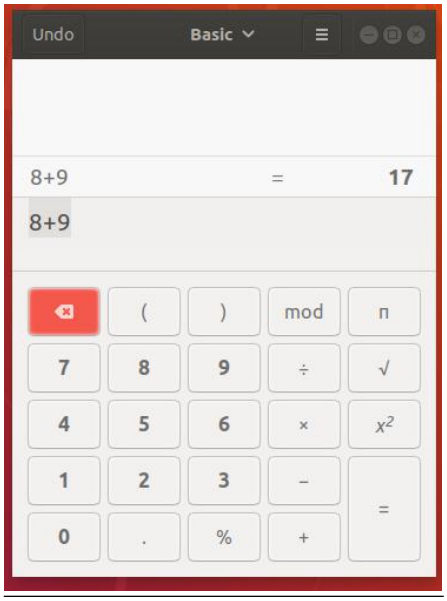
Language − This attribute specifies what scripting language you are using. Typically, its value will be JavaScript. Although recent versions of HTML (and XHTML, its successor) have phased out the use of this attribute.

Type − This attribute is what is now recommended to indicate the scripting languagein use and its value should be set to "text/javascript".

DESIGN/EXECUTION STEPS

1. In this step, you write the head tag's initial code and include the .js and .css files.
2. creates buttons for digits from 0 to 9 and operators like +,-,*,/, and =
3. Add styling in .css file
4. Handle the input provided by the user
   A. Add necessary function in .js file to handle the input provided by the user
   B. Add validation for blank input
   C. Add validation for divide by zero error

EXPECTED OUTPUT (CALCULATOR APP)



CONCLUSION/ANALYSIS

Hence, we applied validate the data using JavaScript.