# Assignment :- 1

1. Explain characteristics, syntax and usage of XML.

Ans:- Characteristics :-

- 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 and is available as an open standard.

XML Usage :-

- XML can work behind the scene to simplify the creation of HTML documents for large web sites.
- XML can be used to exchange the information between organizations and systems.
- XML can be used for offloading and reloading of databases
- XML can be used to store and arrang the data, which can customize your data handling needs.
- XML can easily be merged with style sheets to create almost any desired output.
- virtually, any type of data can be expressed as an XML document.

XML Syntax :-

1. Root :- XML documents must contain one

root element that is the parent of all other elements. Ex; <root>

```
        <child>
              <subchild>......</subchild>
        </child>
    </root>
```

2. Prolog :- <?XML version ="1.0' encoding="UTF-8"

→ XML prolog is optional. If it exists, it must come first in the document.

→ XML documents can contain international characters, like Norwegian or French.

→ To avoid this errors, you should specify the encoding used, or save your XML files as UTF-8.

→ UTF-8 is the default character encoding for XML documents.

3. All XML elements must have closing tag unlike HTML. Ex; <P> Hello world </P>
<br/>

4. XML tags are case sensitive. That means opening and closing tags must be return with the same case.

5. XML attributes values must always be quoted ("

6. Entity references:- some characters have special meaning in XML. That is if you place a character like "<" in XML element between generate error. because XML parser interprits it has a start of new element. Ex; <message> salary < 1000 </message>

7. Now to avoid this error, we use entity reference (&lt;). there are 5 predefined Entity reference.

2. Explain about XML document and rules of declaratio

Ans:- An XML document is a basic unit of XML information composed of elements and other markup in an orderly package. An XML document can contains wide variety of data.

→ Document Elements are the building blocks of XML. These divide the document into a hierarchy of sections, each serving a specific Purpose.

→ You can separate a document into multiple sections so that they can be rendered differently or used by a search engine.

→ The elements can be containers, with a combination of text and other elements.

- Rules of declaration :-

→ If the XML declaration is present in the XML, it must be placed as the first line in the XML document

→ If the XML declaration is included, it must contain version number attributes.

→ The parameter names and values are case sensitive

→ The names are always in lower case.

→ The order of placing the parameters is important. The correct order is : version, encoding and standalone.

→ Either single or double quotes may be used.

→ The XML declaration has no closing tag.

# Assignment :- 2

1. Explain about various Events and Effects of jquery.

Ans:- Jquery is tailor-made to respond to events in an HTML page.

→ All the different visitor's actions that a web page can respond to are called events.

→ An event represents the precise moment when something happens.

Ex; • moving a mouse over an element
- selecting a radio button
- clicking on an element

→ The term 'fires / fired" is often used with events. Ex; "The keypress event is fired, the moment you press a key.'

→ Here are some common DOM events:

| Event Mouse | keyboard Events | Form Event | Document/window even |
|---|---|---|---|
| click | key press | Submit | load |
| dbclick | keydown | change | resize |
| mouseenter | key up | focus | scroll |
| mouseleave | - | blur | unload |

• JQuery Effects:-

■ Jquery hide(), show() & toggle()

→ You can hide and show HTML elements with the hide() and show ()methods.

→ You can toggle between the hide () and show() methods with the toggle() method.

→ Syntax :- $(selector).hide(speed, callback);
$ (selector).show (speed, callback);
$ (selector).toggle (speed, callback);

→ The optional speed parameter specifies the speed of the hiding / showing, and can ~~take~~ take the following value : "slow", "fast" or milliseconds.

→ The optional callback parameter is a function to be executed after the hide () or show () or toggle () method completes.

2. Explain about jquery manipulation methods.

Ans:- Jquery provides various methods to add, edit or delete DOM element in HTML page.

→ The following table lists some important methods to add/remove new DOM elements.

| Method | Description |
|---|---|
| append() | Inserts content to the end of element which is specified by a selector. |
| before() | Inserts content before an element which is specified by a selector. |
| after() | Insert content after an element which is specified by a selector. |
| prepend() | Insert content at the beginning of an element specified by a selector. |
| remove() | Removes element from DOM which is specified by selector. |
| replace All() | Replace target element with specified element. |
| Wrap() | Wrap an HTML structure around each element which is specified by selector. |

## Assignment :- 3

1. Compare JSON and XML.

Ans:-

| JSON | XML |
|---|---|
| - It is Javascript object Notation | - It is Exctensible markup language. |
| - It is based on Javascript | - It is derived from SGML. |
| - It is a way of representing objects. | - It is markup Language and uses tag structure to represent data items. |
| - It does not provides any support for namespace. | - It supports namespaces. |
| - It supports array. | - It doesn't supports array. |
| - Its files are very easy to read as compared to XML. | - Its documents are comparatively difficult to read and interpret. |
| - It doesn't use end tag. | - It has start and end tags. |
| - It is less secured. | - It is more secured than JSON. |
| - It doesn't supports comments. | - It supports comments. |
| - It supports only UTF-8 encoding. | - It supports various encoding. |

2. Explain about JSON arrays with example.

Ans:- JSON array represents ordered list of values.

→ JSON array can store multiple values.

→ It can store string, number, boolean or object in JSON array.

→ In JSON array, values must be separated by comma.

→ The [ (square bracket) represents JSON array.

- **JSON Array of strings:-**

→ Let's see an example of JSON arrays storing string values.

ex; ["sunday", "Monday", "Tuesday", "wednesday", "Thursday", "Friday", "saturday"]

- **JSON Array of Numbers:-**

→ Let's see an example of JSON arrays storing number values.

ex; [12, 34, 56, 43, 95]

- **JSON Array of Booleans:-**

→ Let's seen an example of JSON arrays storing boolean values.

ex; [true, true, false, false, true]

- **JSON Array of objects:-**

→ Let's see a simple JSON array example having 4 objects.

ex;
```
{"employees": [
    {"name": "Ram", "email": "ram@gmail.com", "age": 23},
    {"name": "shyam", "email": "shyam23@gmail.com", "age": 2x},
    {"name": "John", "email": "john@gmail.com", "age": 33},
    {"name": "Bob", "email": "bob32@gmail.com", "age": 41}
]}
```

- **JSON Multidimensional Array:-**

→ We can store array inside JSON array, it is known as array of arrays or multidimensional array.

Ex;
```
[
  ["a", "b", "c"]
  ["m", "n", "o"]
  ["x", "y", "z"]
]
```

# Assignment :- 4

1. Explain AJAX and its architecture.

Ans: AJAX stands for Asynchronous JavaScript And XML. In a nutshell, it is the use of the XML Http Request object to communicate with servers.

→ It can send and receive information in various formats, including JSON, XML, HTML, and text files.

AJAX Architecture :-

→ The application will be a single web page developed with Javaserver pages technology. The user will be able to invoke the web page using a web browser and enter the subscription ID which the application validates in real time.

→ As the ID is validated asynchronously, the user can input more information.

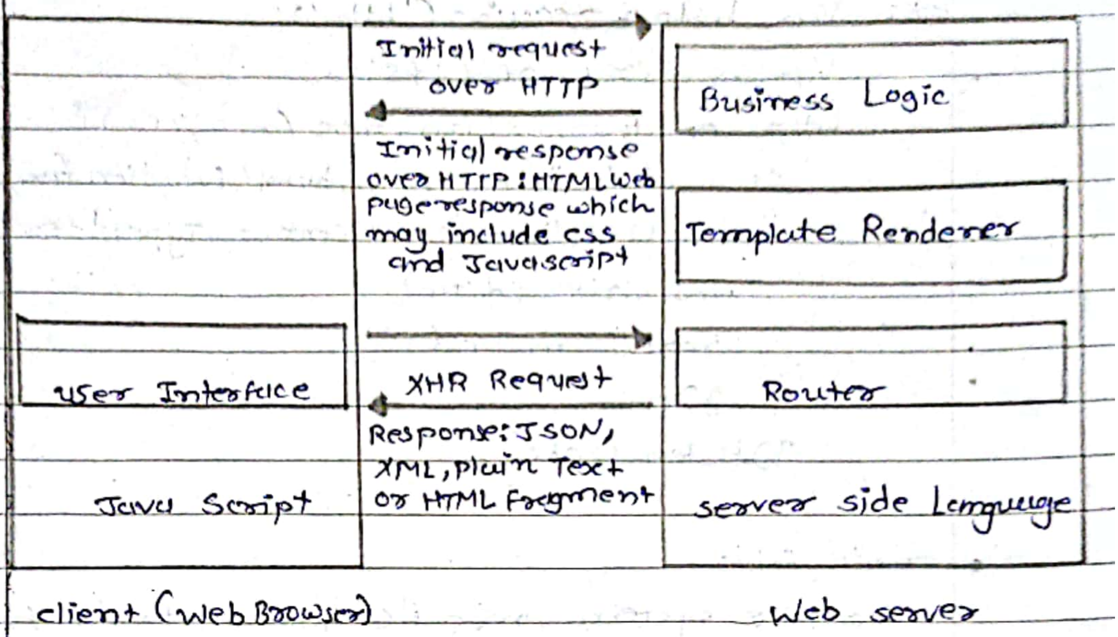
→ The user can view the book titles either by Author or publisher. The screen will populate the Authors or publishers list based on user choice.

→ Based on the selection, the Titles list is populated.

→ All those lists will populate in real time -- in other words, the page is not refreshed, but still the data comes from the backend tier.

→ we call this phenomenon real time refreshes

→ This is what makes Ajax powerful. The user does not wait for page reload to complete because there is no page reload.


Initial request over HTTP
Initial response over HTTP: HTML Web page response which may include css and Javascript
Business Logic
Template Renderer
XHR Request
Response: JSON, XML, plain Text or HTML Fragment
User Interface
Router
Java Script
server side Language
client (Web Browser)
Web server

2. Explain file system Module in Node.js.

Ans:- Node.Js is a Javascript runtime built on chrome's V8 Javascript engine. Node.Js helps developers to write Javascript code to run on the server-side, to generate dynamic content and deliver to the web climents.

→ To handle file operations like creating, reading, deleting etc.

→ To use this file system module, use the require() method:

```
Var fs = require('fs');
```

* Common use for the file system module:

- Read files
- Create files
- update files
- Delete files
- Rename files

- Read files

- The fs.readFile() method is used to read files on your computer.
- Assume we have the following HTML file (located in the same folder as Node.Js):

Ex;
```
Var http = require('http');
Var fs = require('fs');
http.createserver(function (req,res){
    fs.readfile('demofile1.html', function (err,data){
      res.writeHead (200,{'content-Type':'text'});
      res.write (data);
      return res.end ();
    });
  }).listen (8080);
```

- create files:-

- The file system module has methods for creating new files:
  - ★ fs.appendfile()
  - ★ fs.open()
  - ★ fs.writefile()

- The fs.appendfile() method appends specified content to a file. If the file does not exist, the file will be created:

Ex;
```
Var fs = require('fs');
  fs.appendfile('mynewfile1.txt','Hello content!',
  function (err) {
      if (err) throw err;
      console.log ('saved!');
  });
```

- update files :-

- The file system module has methods for updating files:
  - ★ fs.appendfile()
  - ★ fs.writefile()

- The fs.appendfile() method appends the specified content at the end of the specified file:

Ex;
```
Var fs = require ('fs');
  fs.appendfile('mynewfile1.txt','This is my text',
  function (err) {
        if (err) throw err;
        console.log ('updated!');
    });
```

- **Delete files:**

- To delete a file with the file system module, use the fs.unlink() method.
- The fs.unlink() method deletes the specified file:

Ex:
```
var fs = require('fs');
  fs.unlink ('my new file2.txt', function (err){
        if (err) throw err;
        console.log ('File deleted!');
    });
```

- **Rename files:**

- To rename a file with file system module, use the fs.rename () method.
- The fs.rename() method renames the specified files:

Ex;
```
Var fs = require ('fs');
    fs.rename('my new file1.txt', 'my renamed file.txt',
    function (err){
        if (err) throw err;
        console.log ('File Renamed!');
    });
```