

# Unit 5:- Client Side Technologies

Page No.

1

Date: / /

## \* Syllabus -

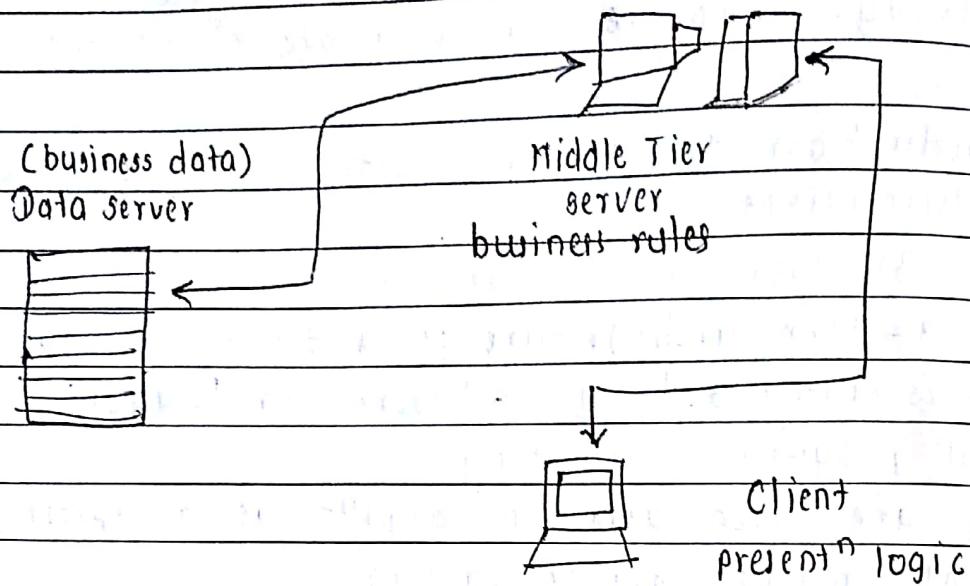
- Introduction to 3 tier & n-Tier Web Architectures,  
Need of Client side technology in multi-tier architectures  
XML, Client side technologies - HTML, DHTML, JAVA Applets, Active X controls, DOM, AJAX.
- Case study- Mobile or portable client side technologies.

## \* Introduction to 3-tier web & n-Tier Web Architectures :-

- In 3-Tier Architecture -
  - A 3-tier architecture is a type of web architecture which is composed of 3 "tiers" or 'layers' of logical program computing.
  - They are often used in apps as a specific type of client-server system.
  - 3 tier architectures provide many benefits for prod & development environments by modularizing the user interface, business logic & data storage layers.
  - It is a 3-way interactn in a client/server environment.
    - The user interface is stored in the client.
    - The business appn logic is stored in one or more servers.
    - The Data is stored in a Database server.
- 1) Presentation Tier - The top-most level of the appn is the user interface. The main fn of the interface is to translate tasks & results to something the user can understand.
- 2) Logic Tier - This layer coordinates the appn, processes commands, makes logical decisions & evaluations, & performs calculations. It also moves & processes data betw the 2 surrounding layers.
- 3) Data Tier - Here informatn is stored & retrieved from a db or file system. The info. is then passed

back to the logic tier for processing & then eventually back to the user.

### \* 3-Tier Architecture -



- These apps run on the Traditional client | server Model but from a appn server.
- Client only displays the GUI & data, but has no part in producing results.
- Dlb server serves to few connns.

### \* 3-Tier Advantages -

- Scalability -
  - The Appn servers can be deployed on many m/s.
  - The dlb no longer requires a conn from every client.
- Reusability -
  - If a standard object is employed, the specific lang. of implementn of middle tier can be made transparent.
- Data integrity -
  - The middle tier can ensure that only valid data is allowed to be updated in the dlb.

- Improved Security -

- Since the client doesn't have direct access to the db, Data layer is more secure.
- Business logic is generally more secure since it is placed on a secured central server.

- Reduced Distribution -

- Changes to business logic only need to be updated on app10 servers & need not to be distributed onto clients.

- Improved Availability -

- Mission Critical Apps can make use of redundant app10 servers & redundant app10 servers, so it can recover from nw of server failures.

- Increased Complexity / Effort -

- In general 3-tier architect. is more complex to build compared to 2-tier architecture.
- Point of Comm<sup>n</sup> are doubled.

- \* Introdn of N-Tier Architecture -

- N-tier arch. is also called multi-tier arch. because the s/w is engineered to have the processing, data mgmt, & present<sup>n</sup> fns physically & logically separated.
- That means that these diff. fns are hosted on several m/c's or clusters, ensuring that services are provided without resources being shared, & st as much, these services are delivered at top capacity.
- The "N" in the name n-tier architecture refers to any no. from 1.
- 3 three different tiers. (N-tier archit. would involve dividing an app10 into 3 diff. tiers.)

These would be the

1. Logic tier,
2. The presentat<sup>n</sup> tier &
3. the data tier.

### 1. Presentation tier -

The top-most level of the appln is the user interface. The main fn of the interface is to translate tasks & results to something the user can understand.

### 2. Logic tier -

This layer coordinates the appln, processes commands, makes logical decisions & evaluations, and performs calculations. It also moves & processes data betw the 2 surrounding layers.

### 3. Data tier -

Here information is stored & retrieved from a db or file system. The info. is then passed back to the logic tier for processing, & then eventually back to the user.

#### \* What are the Benefits of N-Tier Arch?

- There are several benefits to using N-tier architecture for ur slw. These are scalability, ease of management, flexibility & security.
- Secure - You can secure each of the 8 tiers separately using diff. methods.
- Easy to manage - You can manage each tier separately, adding or modifying each tier without affecting the other tiers.
- Scalable - If you need to add more resources, you can do it per tier, without affecting the other tiers.
- Flexible - Apart from isolated scalability, you can also expand each tier in any manner that your requirements dictate.

## \* Need of Client side technology in Multi-tier architectures.

- The appn development landscape has been changing continuously over the past few years, both on the client side(frontend) as well as on the server side(backend).
- On the client side, we have plenty of awesome new & updated Javascript [and other scripting] frameworks & on the server side, we have new architectural approaches such as single page apps, microservices, & serverless architectures.
- In the last decade, the web has dominated as the preferred platform for delivering content & services.
- As a result, every business wanted to be online, & to meet this explosion in demand, developers adopted an N-Tier architectural approach to quickly build & deploy reliable apps.
- An N-Tier architecture consists of multiple independent layers; each layer represents a diff. concern of the system. From a high level, most systems are generally divided into 3 main layers: client, server & storage.
- The client Layer is what the end user sees & interacts with & usually refers to either a thin client or a thick client .
- The storage layer retains important data over time, even when power is turned off, and is often a basic relational db system.
- The Server layer sits betw the client & storage layers & is where all the real action of the appn takes place. Since so much happens within this layer, the server Layer is usually further divided into multiple sub-layers : web, business & persistence.
- The server Web Layer is the entry point of the appn on the server side, & is responsible for handling user interactions, converting requests to models, generating & developing dynamic User interfaces, session mgmt, &

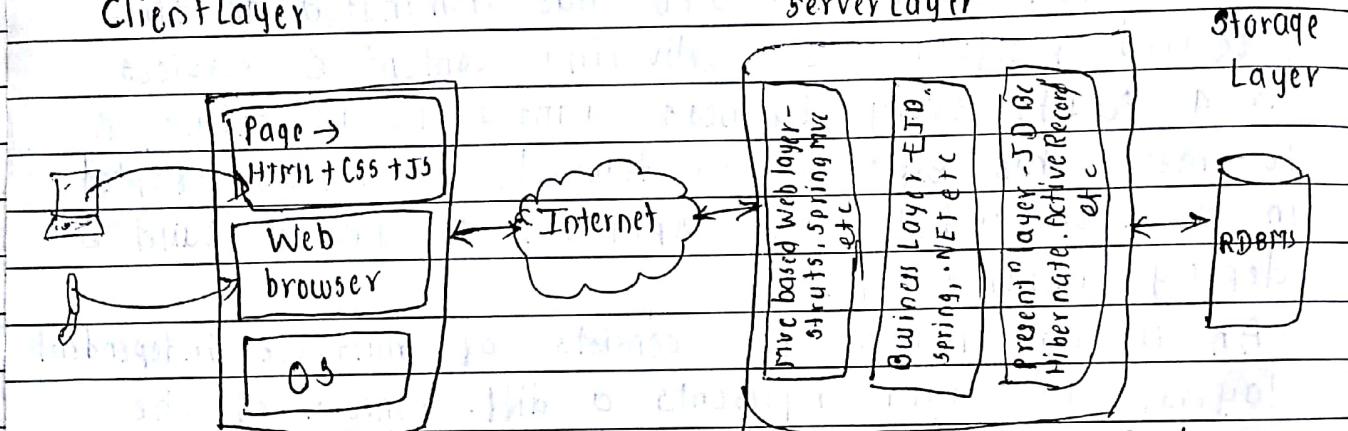
& other tasks.

- The Server Business Layer is where business logic is implemented as a carefully composed & well-defined API.
- The Server Persistence Layer is responsible for abstracting away the app's interaction with the storage layer's specific relational db via Object-Relational Mapping (ORM) tools like Hibernate, Eclipselink, Spring JDBC Template, or other ORM tools.

Client Layer

Server Layer

Storage Layer



- The goal of a microservice is to do one thing & do that one thing well, & can be implemented using almost any technical stack, not necessarily the same stack as other services.

\* XML

- XML stands for Extensible Markup Language.
- XML is a markup lang. similar to HTML.
- XML was designed to describe data.
- XML tags are not predefined in XML. You must define your own tags.
- XML is self describing.
- XML uses a DTD (Document Type Definition) to formally describe the data.

- XML is not a replacement for HTML. XML & HTML were designed with diff. goals.
  - It was designed to describe data & to focus on what data is. HTML was designed to display data & to focus on how data looks.
  - HTML is about displaying info., XML is about describing info.
  - It is extensible.
  - It has been amazing to see how quickly the XML standard has been developed, & how quickly a large no. of svw vendors have adopted the standard.
- \* How can XML be used?
- XML can keep data separated from your HTML
  - XML can be used to store data inside HTML documents.
  - XML can be used as a format to exchange info.
  - XML can be used to store data in files or in databases.
  - With XML this data can now be stored in a separate XML file.
  - This way you can concentrate on using HTML for formatting & display, & be sure that changes in the underlying data will not force changes to any of your HTML code.
  - XML can also store data inside HTML documents.
  - XML can be used to exchange data. XML can be used to store data. XML can also be used to store data in files or in databases. Apps can be written to store & retrieve info from the store, & generic apps can be used to display the data.

### \* XML Syntax -

- An example XML document :

```
<?xml version = "1.0"?>
<note>
  <to>Tove</to>
  <from>Janis</from>
```

<heading> Reminder </heading>  
<body> Don't forget me this weekend! </body>  
<note>

- The first line in the document: The XML declaration should always be included. It defines the XML version of the document.

#### XML Attributes -

- XML attributes are normally used to describe XML elements, or to provide additional information about elements. From HTML you can remember this construct : <IMG> SRC = "computer.gif", In this HTML example SRC is an attr. to the IMG element. The SRC attr. provides additional info. about the element.
- Attr. are always contained within the start tag of an element. Here are some examples:

#### XML examples :

<file type = "gif">  
<person id = "3344">

#### \* Need of XML -

1. Simplicity — XML can be easily understood. We can create our own tags & build the appn. We are free to develop the system as per our requirements & with our own conventions. This makes the thing very simple for us.
2. Organization — The design process can be segmented to build the platform. Data can be stored on one page while the formatting rules can be stored on another page.

#### 3. Accessibility —

- Data can be divided in XML. This makes the access of data easy & fast whenever there is need of making change in data.

4. Standardization - XML is an international standard. This means XML document can be viewed anywhere in the world.

#### 5. Multiple Apps -

"Write once, use anywhere, any no. of times" rule is applied to XML. For XML data we can create any no. of display pages as we want.

### \* XML Key Components -

- 1. Character - An XML document is a string of characters. Almost every legal Unicode character may appear in an XML document.

#### 2. Processor & Application -

- The processor analyzes the markup & passed structured information to an appn. The specification places requirements on what an XML processor must do & not do, the appn is outside its scope.

#### 3. Markup & Content -

- The characters making up an XML document are divided into markup & content, which may be distinguished by the appn of simple syntactic rules.

#### 4. Tag -

- A tag is a markup construct that begins with < and ends with >. Tags come in 3 flavors:

- Start-tag, such as <section>

- End-tag, such as </section>

- Empty-element tag, such as <line-break/>

#### 5. Element -

- An element is a logical document component that either begins with a start-tag & ends with a matching end-tag or consists only of an empty-element tag.

eg. <greeting> Hello, world! </greeting>.

Another is <line-break/>.

## 6. Attribute -

- An attribute is a markup construct consisting of a name-value pair that exists within a start-tag or empty element tag.

e.g. `<img src = "Rose.jpg" alt = "Rose" />`,

where the names of the attr. are "src" & "alt", & their values are "Rose.jpg" & "Rose".

rep.

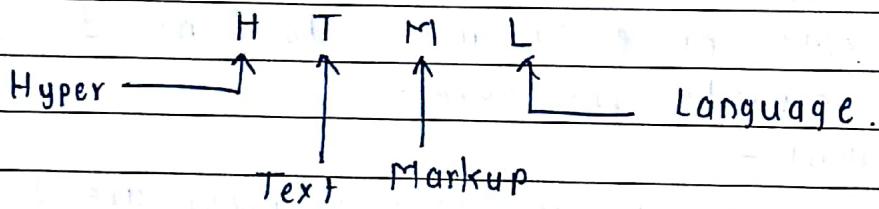
## 7. XML declaration -

XML documents may begin with an XML declaration that describes some information about themselves.

An e.g. is `<?xml version = "1.0" encoding = "UTF-8"?>`.

## \* Client-Side Technologies :-

### 1] HTML :-



- HTML stands for Hyper Text Markup Language.
- Hypertext is nothing but the way in which the diff. web pages are linked with each other. Such links are called as Hypertext.
- As its name indicates, HTML is a Markup Language, that means we can use HTML to simply "mark-up" a text document with tags & that instructs the web browser like chrome how to structure it to display.
- Now a day HTML is widely used for the purpose of formatting web pages by using various tags.

## \* Structure of HTML Document:-

- All the elements are included in the main opening & closing `<html>` tags.

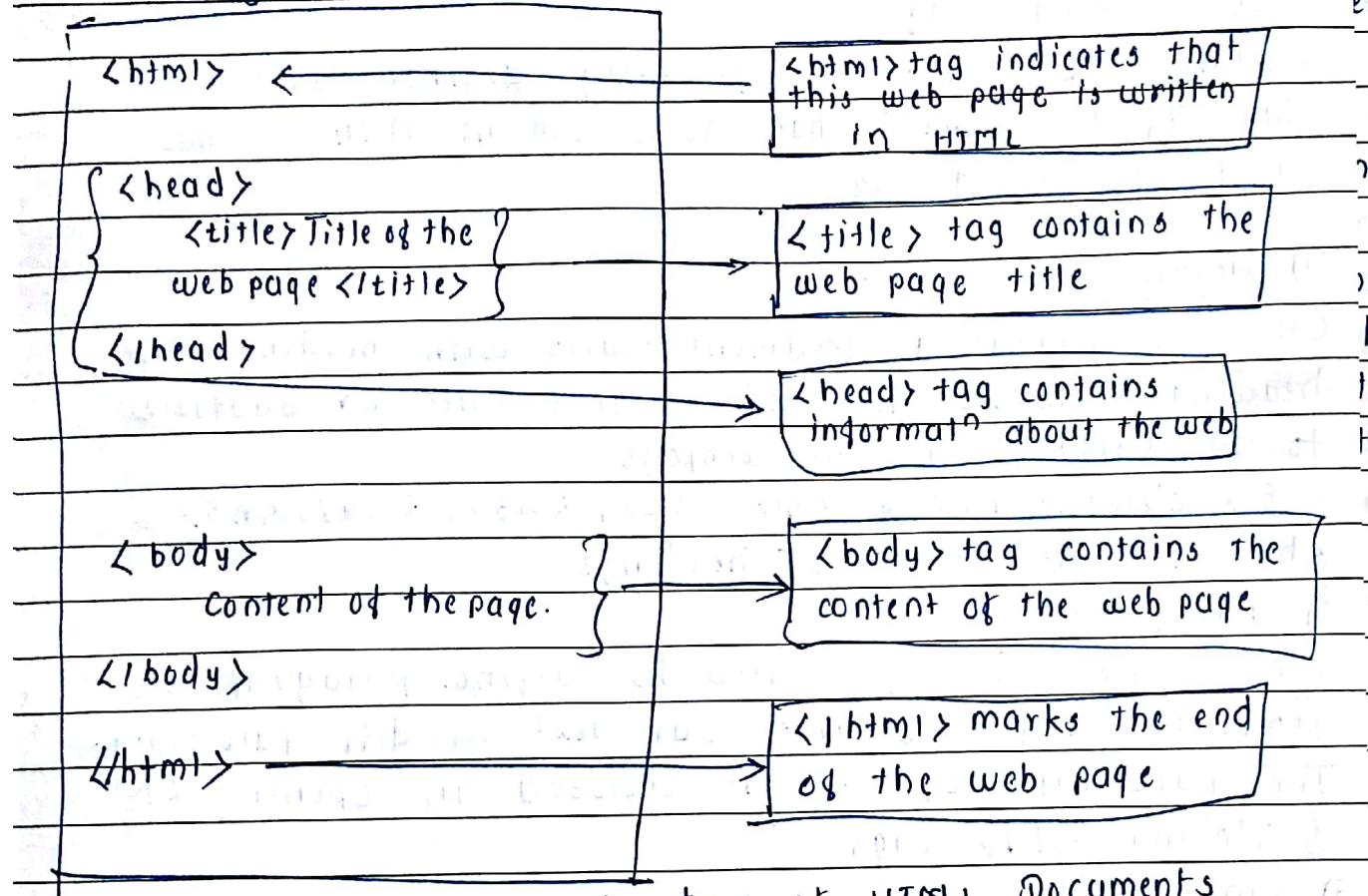


Fig. 1.3.1 : structure of HTML Documents

- An HTML document contains 2 main parts:

1. Head : The head element contains title & metadata of a web document. This section is used to declare variables & dns in scripting lang.
2. Body : The body element contains the information which we like to display on a web page. All text to be displayed & control creatn is done in this section.

## \* HTML Elements | Tags -

- Defn- HTML tags are standard keywords in the web page which define how the web browser should format & display the content.

- Usually most of the tags have 2 parts, an opening & a closing part.

for eg., `<html>` is the opening tag while `</html>` is the closing tag.

- An HTML element is generally defined with starting tag. If the element has some content then it ends with a closing tag.

### 1) Headings -

Use - In general a document starts with heading. The heading elements are used to give title or subtitles to be displayed on the webpage.

- Six elements namely `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` & `<h6>` are used to give headings.

### 2) Paragraph -

Use :- The `<p>` tag is used to define paragraph in HTML. This tag structures our text in diff. paragraphs. The paragraph of text is enclosed in opening `<p>` & closing `</p>` tags.

### 3) Line Breaks -

Use - The `<br>` tag is used to give the line break. It works just like '`\n`' of C programming lang. This tag is called an empty element as it does not need any closing tag.

### 4) Colors & Fonts -

#### a) Colors -

- The look & feel of a website is dependent upon the colors used in a webpage for text & backgrounds. The colors can be specified at page level with the help of `<body>` tag or can be set for individual tags.

The <body> tag has no. of attributes which are used to set colors to diff. entities.

1. bgcolor - Used to set color to the background of the page.
2. text - Used to set color to the body text.
3. alink - Used to set color to the active links or selected links.
4. link - Used to set color to the linked text.
5. vlink - Used to set color to the visited links - that is, for linked text that you have already clicked on.

**B) Font** - User friendliness is the basic need of any website. This can be achieved by making it readable with the help of <font> tag. The text in a webpage can be formatted by setting the <font> --- </font> tag & various font attributes.

- The <font> tag has foll. attributes:

#### Attributes of <font> tag

- 1. Face
- 2. size
- 3. color

#### 1. Face

**Use:-** The face attribute is used to specify the name of font for the text.

<font face = "Arial"> Arial Font </font>

- To the face attr., we can give multiple font names separated by a comma.

#### 2. size:-

**Use:-** The size of a font can be set using the size attribute.

The allowed range of values for font size is from 1 (smallest) to 7 (largest). 3 is the default font size.

<font size = 5> size is 5 </font>

- It is also possible to set the relative font size. That means how many sizes greater or smaller than the current font size.

<font size="2"> Less by 2 </font>  
<font size="3"> Greater by 2 </font>

### 3. Color -

Use - Used to set the color of font. This attribute can accept value as standard name of a color or color code.

## 5) Links - Hyperlinks - In a website there are multiple web pages.

Use -

- Hyperlinks are used to navigate in the website. That means to move from one webpage to another. A webpage can have various hyperlinks which can take us directly to another pages & even specific parts of the current page.

- The anchor tag `<a>` is used to specify the hyperlink. The text written in opening `<a>` & closing `</a>` tag is known as hypertext.

## 6) Frames -

Use - If we wish to display multiple pages at the same time on the browser window, then we can use the frames to divide the browser window into diff. sections & in each section we can display a separate webpage.

## 7) Lists -

Use - As the name suggests, these elements are used to give list of items. Lists are used everywhere on the websites. There are no. of things like articles, website navigation menus, and product features on e-commerce website.

### - Types of HTML list -

HTML provides 3 ways to give the lists -

A. Unordered

B. Ordered

C. Definition

## \* AJAX

### \* Introduction -

AJAX stands for Asynchronous Javascript and XML.

### Use -

- AJAX is an advanced technique used to create web apps which are better, interactive, and faster with the help of diff. scripting languages like HTML, CSS, JavaScript & XML.
- AJAX uses XHTML for content, CSS for present., & Document Object Model plus Javascript for creatn of dynamic contents.
- AJAX allows the web pages to be modified asynchronously by exchanging small bunch of informat? with the server.
- The conventional web pages reload the whole page if there is need to make changes in the content.  
eg. - Eg. of appln using AJAX -
- Google Maps, Gmail, Youtube, & Facebook tabs.

### \* Working of AJAX -

- AJAX uses the XMLHttpRequest object to communicate with the server.
- The fig. 4.3.1 shows the flow of AJAX. Here you will get exact idea about the working of AJAX.

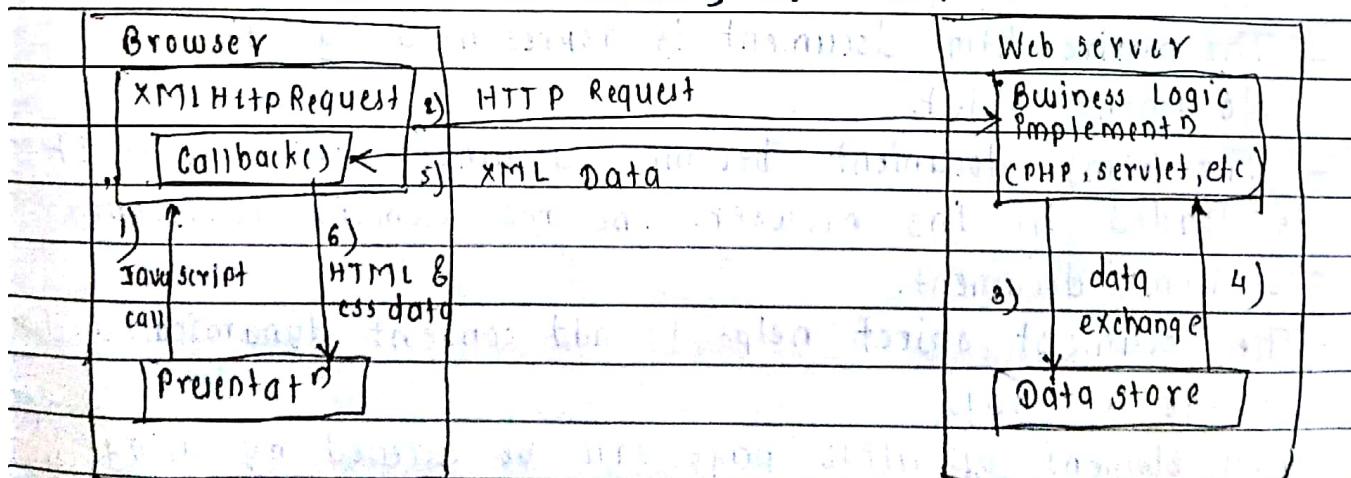


Fig. 4.3.1

- An important role is played by the XMLHttpRequest Object in AJAX processing.

- Page No. \_\_\_\_\_  
Date. \_\_\_\_\_
- Request is sent by the user thru UI (User Interface) & a JavaScript call goes to XMLHttpRequest object.
  - Using the XMLHttpRequest object, the HTTP request sent to the server.

#### \* AJAX Processing Steps and AJAX Script -

Step 1 : An event by the client is fired.

Step 2 : An XMLHttpRequest object is created.

Step 3 : The XMLHttpRequest object is configured.

Step 4 : An asynchronous request is made by the XMLHttpRequest to the Webserver.

Step 5 : The Webserver returns the result in XML format.

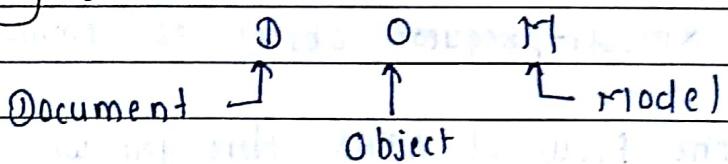
Step 6 : The callback(s) fn is called by the XMLHttpRequest object & processes the result.

Step 7 : The HTML DOM is updated.

Step 1 : An event by the client is fired.

When an event is fired, a Javascript method gets called.

#### \* DOM :-



- DOM stands for Document Object Model.

- The entire html document is represented by the document object.

- The html document becomes document object when it is loaded in the browser. The root element represents the html document.

- The document object helps to add content dynamically in the web page.

- Any element of HTML page can be accessed by using the document object.

\* Hierarchy of objects in web document -

- DOM is the method by which the content of document is accessed & modified.
- In a web document, the organization of objects is implemented in a hierarchical structure.
  1. Window object
  2. Document —
  3. Form —
  4. Form control elements.

1. Window Object -

It resides at top of the hierarchy. It is the topmost element of the object hierarchy.

2. Document Object -

All the HTML documents which get loaded into browser are considered as document objects.

The contents of the page are stored in the document object.

3. Form object -

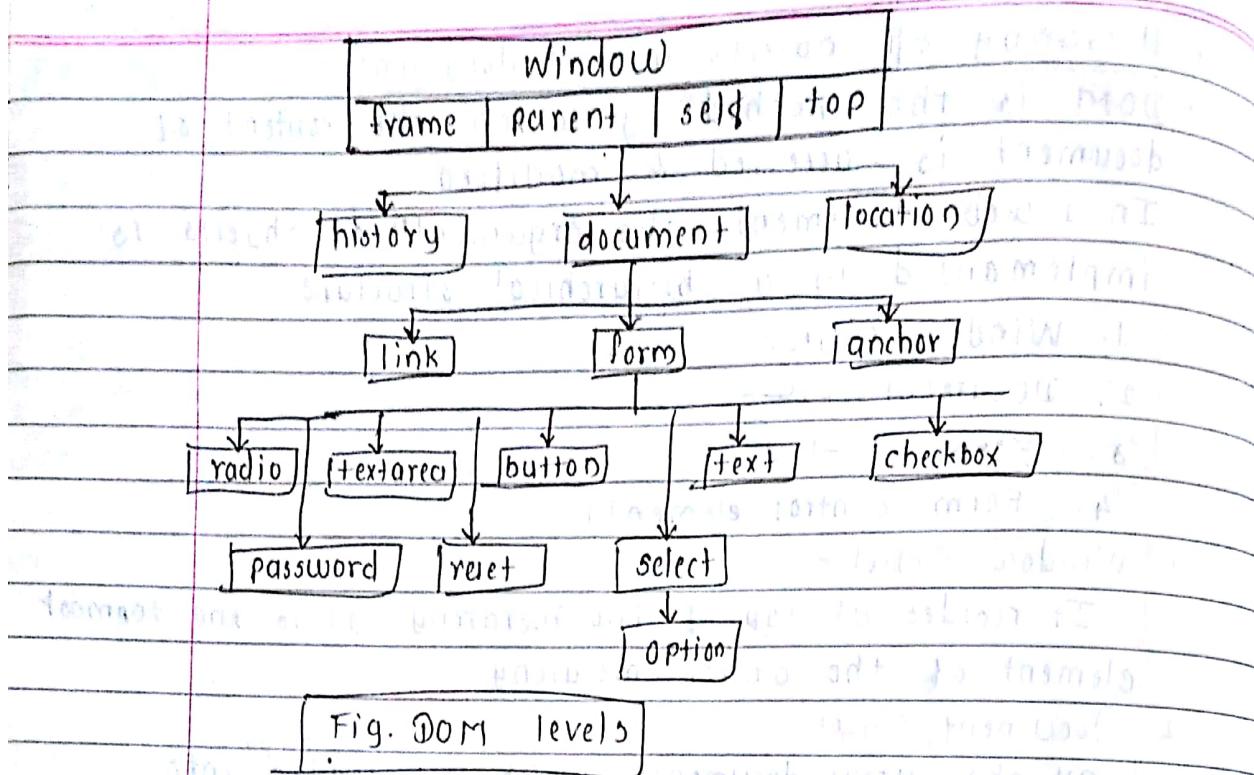
Everything which is contained in the opening `<form>` & closing `</form>` tag gets the form object.

4. Form control elements -

The form object has all the elements which are defined for the objects like text fields, buttons, checkboxes, select box, radio buttons etc.

\* DOM Levels -

- The DOM provides all the features to JavaScript to create dynamic HTML.
- changes can be made in all HTML elements.
- ————— in attributes of HTML elements.
- ————— all CSS styles in the page.
- Existing HTML elements & attributes can be deleted.
- New HTML elements & attr. can be added.
- Response can be given to HTML events.
- New HTML events can be created in HTML page.



\* Manipulating DOM :-

- Using Javascript we can add or remove nodes in the document.

1. Creating new HTML elements(Nodes) :-

- Initially we have to create the element which can be then appended to the existing element.

2. Creating new HTML elements - insertBefore()

- In the previous program, we have seen the appendChild() method which appends the element at the end in parent.

- the insertBefore() method is used to add the new element at the beginning.

3. Removing existing HTML elements - removeChild()

- To remove an HTML element, we should have the information about the parent element. The removeChild() method is used to remove the HTML element.

4. Replacing HTML elements - replaceChild()

- It is also possible to replace an existing element by new one. The replaceChild() method is used for this purpose.

## \* DHTML :-

- Defn -

- DHTML stands for "Dynamic hypertext transfer markup lang."

- DHTML is not a lang.

- DHTML is a term describing the art of making dynamic & interactive WebPages.

- DHTML is a combinat' of web development technologies used to create dynamically changing websites.

- WebPages may include animat'n, dynamic menus & text effects

- The technologies used include a combinat' of HTML, Javascript or VBScript, CSS & the document Object Model (DOM).

- Designed to enhance a web user's experience.

- DHTML includes the foll. features:

- Dynamic content, which allows the user to dynamically change web page content.

- Dynamic positioning of web page elements.

- Dynamic style which allows the user to change the web page's color, font, size or content.

- Advantages of DHTML -

1. Small file size - DHTML files are small compared to other interactive media such as flash or shockwave. Therefore they have a shorter download time & take up less bandwidth.

2. Supported by both major browser manufacturers - both Microsoft & Netscape currently support DHTML in some shape or form.

3. DHTML will be a standard - the WWW consortium or the W3C is the currently implementing standards for DHTML technologies. It has already released preliminary specifications for DOM & CSS.

4. No plug-ins necessary - DHTML uses most of the features already present in the browsers, so there is no need to download any sort of plug-ins.

5. Doesn't require a Java virtual machine.

#### \* Disadvantages of DHTML

1. Costly editing tools - DHTML provides great functionality but the editors available for that in market are pretty expensive.

2. Only new browser support DHTML - DHTML is only supported by Netscape 4.0 or higher & Internet Explorer (IE) 4.0 or higher version.

3. Long & complex coding - DHTML coding is long & complex only the expert Javascript & HTML programmers can write them & edit them with good degree of functionality.

4. Browser support problem - DHTML suffers from browser support problems.

5. Unprotected source code.

#### Uses of DHTML

- Animate text & images in their document independently moving each element from any starting point to any ending point, following a predetermined path or one chosen by the user.

- Use a form to capture user input, and then process & respond to that data without having to send data back to the server.

- Include rollover buttons or drop-down menus.

- \* Difference b/w HTML & DHTML -
- HTML stands for Hyper Text Mark-up Lang. where DHTML stands for Dynamic Hyper Text Mark-up Lang.
- HTML is a lang. where DHTML is a technology.
- HTML is a collect? of tags whereas in DHTML, It is a coll? of technology.
- DHTML sites are dynamic in nature & HTML is static.
- A plain page without any styles & scripts called as HTML whereas A page with HTML, CSS, DOM & Scripts called as DHTML.
- HTML sites will be slow upon client-side technologies while DHTML sites will be fast enough upon client-side technologies.

### \* Java Applets :-

- An applet is a Java program that runs in a Web browser. An applet can be a fully functional Java appl? because it has the entire Java API at its disposal.
- There are some important differences betn an applet & a standalone Java appl?, including the foll-
  - An applet is a Java class that extends the `java.applet.Applet` class.
  - A `main()` method is \*not\* invoked on an applet, and an applet class will not define `main()`.
  - Applets are designed to be embedded within an HTML page.
  - When a user views an HTML page that contains an applet, the code for the applet is downloaded to the user's machine.
  - A JVM is required to view an applet. The JVM can be either a plug-in of the Web browser or a separate runtime environment.

- The JVM on the user's machine creates an instance of applet class & invokes various methods during the applet's lifetime.

#### \* Life Cycle of an Applet -

- 4 methods in the Applet class gives you the framework on which you build any serious applet -

1) init - This method is intended for whatever initialization is needed for your applet. It is called after the param tags inside the applet tag have been processed.

2) start - This method is automatically called after the browser calls the init method.

3) stop - This method is automatically called when the user moves off the page on which the applet sits. It can, ∴ be called repeatedly in the same applet.

4) destroy - This method is only called when the browser shuts down normally.

5) paint - Invoked immediately after the start() method, and also anytime the applet needs to repaint itself in the browser. The paint() method is actually inherited from the java.awt.

e.g. A "Hello, World" Applet.

On the basis of a simple applet named HelloWorldApplet.

```
import java.applet.*; import java.awt.*;  
import java.awt.Graphics;  
public class HelloWorldApplet extends Applet {  
    public void paint (Graphics g) {  
        g.drawString ("HelloWorld", 25, 50);  
    }  
}
```

## \* Active X Controls :-

- Active X is a framework for defining reusable SW components in a programming lang. independent way.

- Apps -

- 1) Calculator for Addition

- 2) System clock

- It is designed to increase the dynamic designs of a website. The controls are components that can easily be inserted into a Web page or other app to reuse packaged functionality someone else has programmed.

- Benefits of ActiveX -

- Active web content which attracts the user & retain their interest in the web page.

- Open cross platform support on Windows & Unix OS.

- The tools in which programmer wants to build ActiveX controls are very common & familiar like VB, Visual C++, & Java.