



**PPT ON
WEB TECHNOLOGIES
V SEM (IARE-R16)**



UNIT -I

INTRODUCTION TO HTML AND JAVASCRIPT

- HTML (Hypertext Markup Language) is a language for specifying how text and graphics appear on a web page.
- In 1960 Ted Nelson introduced Hypertext.
- HTML is a markup language which is used to create static web pages.
- If you are thinking of creating your own web pages, you need to know at least basic HTML.
- These HTML documents are plain text files; user can create these documents using text editor like Notepad, gedit, sublime text or amaya.

History of HTML

- HTML was originally developed by Tim Berners-Lee while at CERN (European Organization for Nuclear Research) , and popularized by the Mosaic browser developed at NCSA (National Center for Supercomputing Applications). HTML standards are created by group of interested organizations and individuals called **W3C (World Wide Web Consortium)**. There have now been three official HTML standards:
- HTML 2.0 was released in 1994 and remains the baseline for backwards compatibility and should be supported by all browsers.
- HTML 3.2 was released in 1996 with many useful additions
- HTML 4.0 was released in 1997 and slightly amended in 1999
- Now, The current version is HTML 5.0 was released in 2012

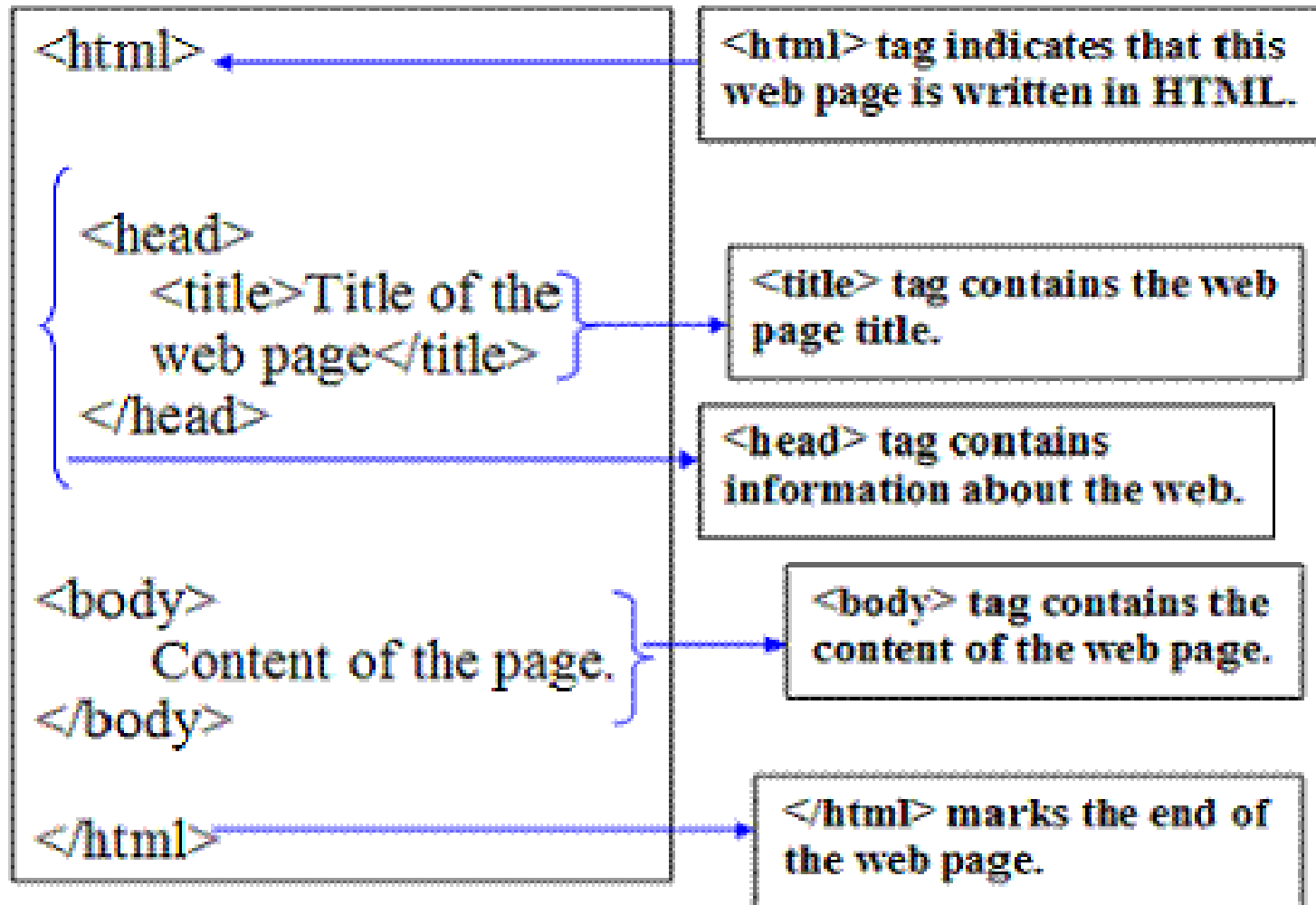
Advantages of HTML:

- It is widely used.
- Easy to learn and use.
- It is error-free language
- Every browser supports HTML language.
- It is by default in every **OS** so, you don't need to purchase extra software.

Disadvantages of HTML:

- It can create only static and plain pages so, if we need dynamic pages then HTML is not useful.
- Need to write lot of code for making simple webpage.
- Security features are not good in HTML.

BASIC STRUCTURE OF HTML



An HTML document has two* main parts:

- *head*: The head element contains title and meta data of a web document.
 - *body*: The body element contains the information that you want to display on a web page.
- HTML (Hypertext Markup Language) is a language for specifying how text and graphics appear on a web page.
 - In 1960 Ted Nelson introduced Hypertext.
 - HTML is a markup language which is used to create static web pages.
 - These HTML documents are plain text files; user can create these documents using text editor like Notepad, gedit, sublime text or amaya.

- The real power and flexibility of HTML is in Hyperlinks. Hyperlinks are created with anchor tag(<a>)
- You can use hyperlink to
 - create links with HTML pages
 - link different HTML pages
 - access services at other Internet sites

Anchor tag:

- The anchor tag is created by <a> . . . tags. The tag has three sections: the address of the referenced document, a piece of text displayed as link, and the closing tag.

Syntax:

 Text

- *href* means Hypertext references that can be used for giving the path of a file which you want to link.
- The “Text” between the `<a> . . . ` tags acted as a hyperlink. This text is called “*hypertext*”. When you click on this text, the linked page or file will be displayed. You can also replace this text with image. In that case, the image will act as a hyperlink.

href

- Used to specify the path and file name of the HTML page that you need to access by using a hyperlink

Name or id

The *name* attribute enable you to create anchor with in HTML page. This anchor tag is used to bookmark a location in an HTML page.

Title

The *title* attribute is used to provide information about a link. The value specified for this attribute appears as a small tip when you place the mouse pointer over the hyperlink.

Target

The *target* attribute is used to specify the target window where the linked HTML page is displayed. You can assign the name of a frame as the value for this attribute

Attribute Values

Value Description

- `_blank` Opens the linked document in a new window or tab
- `_self` Opens the linked document in the same frame as it was clicked (this is default)
- `_parent` Opens the linked document in the parent frame
- `_top` Opens the linked document in the full body of the window
- *framename* Opens the linked document in a named frame

Creating hyperlinks to HTML Pages

- To create hyperlinks to an HTML page, you specify the address of the HTML pages as the value of *href* attribute of the `<a>...` tags

first.html

```
<html>
```

```
<head> <title> Navigation </title> </head>
```

```
<body>
```

```
<p align="left">You are in fist page<a href="second.html">Next  
    Page</a> </p>
```

```
</body>
```

```
</html>
```

.

- One of the most effective ways of structuring a web site is to use lists. Lists provides straight forward **index in the website**. HTML provides three types of lists.
 - Unordered / Bulleted list
 - Ordered / Numbered list and
 - Definition/Description list
- Lists can be easily embedded within other lists to provide complex yet readable structures.
- `...`:
- The Ordered / Unordered lists are each made up of set of **list items**. These list items are added with `` tag. Elements of a list item may format with any of the usual formatting tags and may be images or hyperlinks.

Unordered Lists

- Unordered lists are also called Un-numbered lists. The Unordered list elements are used to represent a list of items, which are typically separated by white space and/or marked by bullets. Using `` tag is used to create unordered lists in HTML. The list of items is included in between `.....`. The *TYPE* attribute can also be added to the `` tag that indicates the displayed bullet along with list of item is square, disc or circle. By default it is disc.

Syntax:-

```
<UL [TYPE="square | disc | circle "] >
  <LI>item name1 </LI>
  <LI>item name2 </LI>
  -----
  <LI>item name n </LI>
</UL>
```

Ordered Lists:-

- Ordered lists are also called **Numbered** or **Sequenced lists**. In the ordered list the list of items have an order that is signified by numbers, hence it some times called as **number lists**. Elements used to present a list of items, which are typically separated by white space and/or marked by numbers or alphabets. An orders list should start with the element, which is immediately followed by a element which is same as
- in unordered list. End of ordered lists is specified with ending tag .

TABLES

- Table is one of the most useful HTML constructs. Tables are finding all over the web applications. The main uses of table are that they are used to structure the pieces of information and to structure the whole web page.
- Tables allow you to present data across rows and columns, making it easy for reader to correlate connected pieces of information.
- A Table is collection of rows and columns or collection of cells.
- The basic element in a table is *cell* or the *grid*. A cell is formed when a row crosses a column.
- You can specify the amount of space between cells in a table. This space is called *cell spacing*.
- You can specify the amount of space between the contents of a cell and the cell wall. This space is called *cell padding*.

- ◎ You can use various elements to specify the details of a table. Many table elements also take attributes, which allows you to further specify the look of the table.

TABLE

Table row

Table data

Table Heading

The TABLE Element

- ◎ align: used to specify the alignment of a table in a HTML page.
- ◎ border: used to specify the thickness of the table border in pixels.
- ◎ bgcolor: used to specify the background color for the table.
- ◎ frame: used to specify the which side of the outer border is visible in the browser. You must specify the border attribute before you specify the FRAME attribute.

Color and Images

- Color is essential to the Web experience; it brings life to pages and makes them exciting. Color can be used in number of places on a Webpage: The background can be colored, individual elements can be altered, and links which already colored can be changed
- To change the colors of links or of the page background hexadecimal values are placed in the <body> tag:

Syntax:

- `<body bgcolor = "#rrggbb" text = "#rrggbb" link= "#rrggbb" vlink= "#rrggbb" alink= "#rrggbb">`
- The *vlink* attribute sets the color of links visited recently, *alink* the color of a currently active link. The six figure hexadecimal values must be enclosed in double quotes and preceded by a hash(#).

Images

- Images are second aspect of pleasant Web experience. You can add images to an HTML page to either improve its appearance or present important information. To add images to an HTML page, you can use several image formats. These formats include „gif“, „jpg“ and „png“.
- Images can be added in two different ways:
 - By using “background” attribute of <body> tag
 - By using tag

The <BODY> Tag: background attribute

- You can use the background attribute of the <BODY> tag to add image as a background in an HTML page.

Syntax:

- <body background = “URL”>. . .</body>
- This tag will set a background image present in the URL.

The Image tag

- You can use image tag in an HTML page to add images along with text. An image added using the image tag occupies space within the HTML page. To add image to HTML page, use `` tag.

Syntax:

- ``

Attributes:

- `src` : used to specify the name of the file. When using `` tag, it is mandatory to specify a value for the **SRC** attribute
- `align` : used to specify the vertical alignment of an image
- `height` : used to specify the vertical area that an image will occupy in HTML page
- `Width` : used to specify the horizontal area that an image will occupy in HTML page

FRAMES

- Frames provide a pleasing interface which makes your web site easy to navigate. You can use frames in HTML page to manage the layout of the page. Frames divide the browser window into horizontal or vertical sections. Each section can be used to display a separate HTML page.
- Each frame within a webpage has the following properties:
- It has unique name
- It displays an HTML document independent of other frames
- Its size can be dynamically changed according to the size of the content in HTML page

Creating frames

- When we talk about frames actually we are referring to frameset, which is a special type of web page. Simply frameset is nothing but collection of frames. Web page that contains frame element is called framed page. Framed page begins with `<frameset>` tag and ends with `</frameset>`. Each individual frame is identified through `<frame>` tag. Creation of framed page is very simple. You can nest the framesets. First you decide how you want to divide your webpage and accordingly define frame elements.

Syntax:

`<FRAMESET cols="n,n" rows="n,n">`

`<FRAME>` Tags

`</FRAMESET>`

CASCADING STYLE SHEETS

- CSS stands for Cascading Style Sheets. A style sheet is document that allows you to control the rendering, such as fonts, colors, typefaces and other aspects of style, of a web document. You can define style sheet properties within HTML document or in an external file.
- A Style sheet is a collection of rules

Advantages:

- Enables you to separate content from formatting.
- Reduces download time by removing formatting information from the document.
- Allows you to ensure a consistent appearance across a site
- It supports reusability (Inheritance).

What is CSS?

- CSS is an abbreviation for Cascading Style Sheets.
- CSS works with HTML and other Markup Languages (such as XHTML and XML) to control the way the content is presented.
- Cascading Style Sheets is a means to separate the appearance of a webpage from the content of a webpage. CSS is a recommendation of the World Wide Web Consortium (the W3C).

Why Use CSS?

- Increases Download Speed
- Makes Pages More Accessible
- Reduces Time
- Multiple Style Sheets Cascade Into One

Javascript Basics

- Script means small piece of code. Scripting languages are two kinds one is client-side other one is servers-side scripting.
- In general client-side scripting is used for verifying simple validation at client side, server-side scripting is used for database verifications.
- VBScript, java script and J script are examples for client-side scripting and ASP, JSP, Servlets, PHP etc. are examples of server-side scripting.
- **JavaScript** (originally known as "LiveScript") is a scripting language that runs inside the browser to manipulate and enhance the contents of Web pages. Java Script is designed to add interactivity to HTML pages. Web pages are two types
 - Static web page
 - Dynamic webpage

Javascript Basics

- Static web page where there is no specific interaction with the client
- Dynamic web page which is having interactions with client and as well as validations can be added.
- Microsoft's version of JavaScript is Jscript.
 - Java script code as written between `<script>-----</script>` tags
 - All java script statements end with a semicolon
 - Java script ignores white space
 - Java script is case sensitive language
 - Script program can save as either. Js or. html

Benefits of JavaScript

- It is widely supported by web browsers;
- It gives easy access to the document objects and can manipulate most of them.
- Java Script gives interesting animations with long download times associated with many multimedia data types;
- Web surfers don't need a special plug-in to use your scripts
- Java Script relatively secure - you can't get a virus infection directly from Java Script.
- JavaScript code resembles the code of C Language; the syntax of both the language is very close to each other. The set of tokens and constructs are same in both the language.

String Manipulation

- While JavaScript has many applications, working with string values is one of the most popular.
- Strings are objects within the JavaScript language.
- They are not stored as character arrays, so built-in functions must be used to manipulate their values.
- The functions provide various ways to access the contents of a string variable. Let's take a closer look at these functions
- This ranges from extracting portions of a string to determining if a string contains a specific character.

String Manipulation

- The following JavaScript functions provide developers with everything they need:
- `concat()` - Combines the text of two or more strings and returns a new string.
- `indexOf()` – Returns the starting index of a substring within another string. A `-1` is returned if no match is found.
- `charAt()` – Returns the character at the specified location.
- `lastIndexOf()` - Returns the index within the string of the last occurrence of the specified value, or `-1` if not found.
- `match()` - Used to match a regular expression against a string.
- `substring()` – A portion of a string is returned. A starting and ending location are passed to this function.

String Manipulation

- `replace()` – Used to find a match between a regular expression and a string, and to replace the matched substring with a new substring.
- `search()` - Executes the search for a match of a regular expression. If successful, search returns the index of the match inside the string. Otherwise, it returns -1.
- `slice()` - Extracts a section of a string and returns a new string.
- `split()` - Splits a string into an array of strings by separating the string into substrings.
- `length()` – The length of the string is returned as the count of the number of characters it contains.
- `toLowerCase()` – Converts the entire string to lower case.
- `toUpperCase()` – Converts the entire string to upper case.

Functions

- Functions are one of the fundamental building blocks in JavaScript. A function is a JavaScript procedure -a set of statements that performs a specific task when called. A function definition has these basic parts:
 - The *function* keyword
 - A function name
 - A comma-separated list of arguments to the function in parentheses
 - The statements in the function in curly braces: { }

Functions

Defining a Function

Defining the function means, name the function and specifies what to do when the function is called. You define a function within the `<SCRIPT>...</SCRIPT>` tags within the `<HEAD> ... </HEAD>` tags..

Here's an example of *defining* a function:

```
function msg()
{
window.alert("This is an alert box.");
}
```

Here's an example of a function that takes a parameter: function
welcome(string)

```
{
window.alert("Hi"+string);
}
```

Calling a Function

- Calling the function actually performs the specified actions. When you call a
- function, this is usually within the BODY of the HTML page, and you usually pass a parameter into the function on which the function will act.
- Here's an example of calling the same function:
- *msg()*;
- For the other example, this is how you may call it:
- `<input type="button" name="welcome" onClick="msg1("iare")"/>`



UNIT -II

OBJECTS IN JAVASCRIPT AND XML

Regular Expressions

- A regular expression is an object that describes a pattern of characters.
- When you search in a text, you can use a pattern to describe what you are searching for.
- A simple pattern can be one single character.
- A more complicated pattern can consist of more characters, and can be used for parsing, format checking, substitution and more.
- Regular expressions are used to perform powerful pattern-matching and "search-and-replace" functions on text.

Regular Expressions

Syntax:

- `var patt=new RegExp(pattern,modifiers);` `var patt=/pattern/modifiers;`
 - pattern specifies the pattern of an expression
- modifiers specify if a search should be global, case-sensitive, etc.

A script language may take name data from a user and have to search through the string one character at a time.

- The usual approach in scripting language is to create a pattern called a regular expression
- Reg Exp describes a set of characters that may be present in a string.
Example: `var pattern = "target";` `var string = "can you find the target";` `string. match(pattern);`

Regular Expressions

- Brackets

Brackets are used to find a range of characters: • [abc] --- Find any character between the brackets

- [^abc] --- Find any character not between the brackets • [0-9] --- Find any digit from 0 to 9
- [A-Z] --- Find any character from uppercase A to uppercase Z
- [a-z] --- Find any character from lowercase a to lowercase z
- [A-z] --- Find any character from uppercase A to lowercase z
- [adgk] --- Find any character in the given set

Exception Handling

Exception:

- Exception is an abnormal condition.
- In JavaScript , exception is an event that disrupts the normal flow of the program.
- It is an object which is thrown at runtime.

Exception Handling:

- Exception Handling is a mechanism to handle runtime errors.
- The core advantage of exception handling is to maintain the normal flow of the application.
- Exception normally disrupts the normal flow of the application that is why we use exception handling

Exception Handling

JavaScript Errors:

There are three types of errors in programming:

- Syntax Errors,
- Runtime Errors
- Logical Errors.

Exception Handling

The try...catch...finally Statement

- JavaScript implements the try...catch...finally construct as well as the throw operator to handle exceptions.
- You can catch programmer-generated and runtime exceptions, but you cannot catch JavaScript syntax errors.
- The try block must be followed by either exactly one catch block or one finally block (or one of both).
- When an exception occurs in the try block, the exception is placed in e and the catch block is executed.
- The optional finally block executes unconditionally after try/catch.

Built –in Objects, Events

Built-in objects in java script are:

- Document object
- Form object
- Browser object
- Date Object

The form object:

- Two aspects of the form can be manipulated through JavaScript.
- First, most commonly and probably most usefully, the data that is entered onto your form can be checked at submission.
- Second you can actually build forms through JavaScript.

Example: `var t1=document.forms[0].elements;`

Built –in Objects, Events

The Date object:

- JavaScript provides functions to perform many different date manipulation. Some of the functions are:
- **Date()**
 - Date(year, month, day [,hour, minute, second])
 - getDate()
 - getDay()
 - getFullYear()
 - getHours()

The Date object:

- getMinutes()
- getSeconds()
- setDate(day)
- setFullYear(year [,month, day])
- toString()

Events

- JavaScript is a event-driven system.
- Nothing happens unless it is initiated by an event outside the script.

Event Handler Description:

- Blur (onBlur)
- Change (onChange)
- Click (onClick)
- Double click (onDbClick)
- Focus (onFocus)
- Keydown (onKeyDown)
- Keypress (onKeyPress)
- Keyup (onKeyUp)
- Load (onLoad)

Event Handler Description:

- Mousedown (onMouseDown)
- Mousemove (onMouseMove)
- Mouseout (onMouseOut)
- Mouseover (onMouseOver)
- Submit (onSubmit)
- Unload (onUnload)

Dynamic HTML with Javascript

Dynamic HyperText Markup Language (DHTML) is a combination of Web development technologies used to create dynamically changing websites.

- Web pages may include animation, dynamic menus and text effects.
- The technologies used include a combination of HTML, JavaScript or VB Script, CSS and the document object model (DOM).

Dynamic HTML or DHTML, is a collection of technologies used together to create interactive and animated web sites by using a combination of a HTML and CSS or JavaScript.

- DHTML allows authors to add effects to their pages that are otherwise difficult to achieve.
- Use a form to capture user input, and then process and respond to that data without having to send data back to the server.

Difference between Static and dynamic web pages: Static page:

- In static web pages the theme and content of web pages remained fixed
- Browsing and loading of static web pages are more quicker than dynamic web pages
- Changing the content in static web pages is a difficult.
- If file extension of a URL is in .htm or .html then it is a static web pages.
- Static web pages are created through HTML language
- Static web pages plan is an easy and cheap method if you wants to create a static and non updating web pages.

Dynamic page:

- In dynamic web pages they changed according to run time.
- Browsing and loading of dynamic web pages are slow
- Changing the content in these web pages is a easy task because in dynamic web pages server application do this automatically.
- If file extension of a URL is in .php, .asp and .jsp then it is the example of dynamic web pages.
- Creating dynamic web pages is very tough but this method is advisable if you have plan to update the content and material frequently

Dynamic HTML with Javascript

JavaScript can also be used to change the inner content and attributes of HTML elements dynamically.

- To change the content of an HTML element use:
`document.getElementById(id).innerHTML=new HTML`
- To change the attribute of an HTML element use:
`document.getElementById(id).attribute=new value.`

Dynamic Style:

This concept will dynamically update the styling applied to your elements by manipulating your CSS at runtime using JavaScript.

- To apply dynamic style(color) to the page use the following syntax:
- `document.body.style.backgroundColor=colorname;`

Data validation is the common process that takes place in the web sites. One common request is for a way of validating the username and password. Following program shows the validation of data which uses two frames, in one frame user is going to enter the data and in the other frame equivalent result is going to be displayed.

Example: JavaScript code for data validation

- **Mypage.html**

```
<html><head><title>framepage</title></head>  
<frameset rows="20%,*"> <frame name="f1" src=""> <frameset  
  cols="20%,*"><frame name="f2" src=""><frameset cols="50%,*">  
  <frame name="f3" src="D:\Documents and  
  Settings\Amer\Desktop\amer\p6\reg.html">  
  <frame name="f4" src="D:\Documents and  
  Settings\Amer\Desktop\amer\p6\profile.html">  
</frameset></frameset></frameset></html>
```

Myform.html

```
<html><head>  
<script language = "javascript" src = "D:\ Documents and Settings \ nari\  
    Desktop\ amer\ p6\ validate.js">  
</script> </head>  
<body> <form>  
Background    Color:    <input    type="text"    size=16    name="bg"  
    value="white">  
Candidate's name:<input type="text" size=16 name="c">  
<input type="button" value="showit" onClick="validate()">  
</form> </body> </html>
```

Data Validation

Validate.js

```
function validate() {  
var t1=document.forms[0].elements;  
var t2=parent.frames['f4'].document;  
var bg1=t1.bg.value;  
var c1=t1.c.value;  
t2.open();  
t2.write("<body bgcolor="+bg1+">");  
t2.write("Candidate name is : "+c1);  
t2.write("</body>");  
t2.close();  
}
```

XML stands for Extensible Markup Language. It is a text-based markup language 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 a subset of SGML. XML is used to describe the structure of a document not the way that is presented. XML is the recommendation of World Wide Consortium (W3C). 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 selfdescriptive 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.

XML is extensible: XML allows you to create your own selfdescriptive 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.

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

Difference Between XML and HTML:

- XML is not a replacement for HTML.
- XML and HTML were designed with different goals
- XML was designed to transport and store data, with focus on what data is HTML was designed to display data, with focus on how data looks
- XML was created to structure, store, and transport information.

Tags and Elements: XML is not a replacement for HTML. An XML file is structured by several XML-elements, also called XMLnodes or XML-tags. XML-elements' names are enclosed by triangular brackets < > as shown below: Syntax Rules for Tags and Elements:

Element Syntax: Each XML-element needs to be closed either with start or with end elements as shown below:

- <element>....</element>

Sample XML program:

```
<?xml version="1.0"?>  
<college>  
  <studdetail>  
    <name>  
      <firstname>Teja</firstname>  
      <lastname>Santosh</lastname>  
    </name>  
  </studdetail>  
</college>
```

DTD

- Document type definition have been successfully used in SGML applications for many year.
- XML Document Type Declaration, commonly known as DTD, is a way to describe XML language precisely.
- DTDs check vocabulary and validity of the structure of XML documents against grammatical rules of appropriate XML language
- An XML DTD can be either specified inside the document, or it can be kept in a separate document and then linked separately.

Types of DTD:

DTD are of two types:

- Internal DTD
 - External DTD
- The Building Blocks of XML Documents:
- Elements
 - Attributes
 - Entities
 - PCDATA
 - CDATA

Internal DTD:

```
<?xml version="1.0"?>
<!DOCTYPE note[<!ELEMENTnote(to,from,heading,body)>
<!ELEMENT to(#PCDATA)>
<!ELEMENT from(#PCDATA)>
<!ELEMENT heading(#PCDATA)>
<!ELEMENT body(#PCDATA)> ]>
<note><to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don'tforgetmethisweekend</body>
</note>
```

External DTD:

Note.xml:

```
<?xml version="1.0"?>  
<!DOCTYPE note SYSTEM "Note.dtd">  
<note>  
<to>Tove</to>  
<from>Jani</from>  
<heading>Reminder</heading>  
<body>Don't forget me this weekend!</body>  
</note>
```

External DTD:

Note.dtd:

```
<!ELEMENT note(to,from,heading,body)>
```

```
<!ELEMENT to(#PCDATA)>
```

```
<!ELEMENT from(#PCDATA)>
```

```
<!ELEMENT heading(#PCDATA)>
```

```
<!ELEMENT body (#PCDATA)>
```

Xml Schema:

- XML Schema is commonly known as XML Schema Definition (XSD).
- It is used to describe and validate the structure and the content of XML data.
- XML schema defines the elements, attributes and data types. Schema element supports Namespaces.
- It is similar to a database schema that describes the data in a database.

Definition Types:

Types are two types:

- Simple Type
- Complex Type

Document Object Model:

- DOM is an application program interface (API) for XML documents.
- The DOM API specifies the logical structure of XML documents and the ways in which they can be accessed and manipulated.
- The DOM API is just a specification.
- They can build static documents, navigate and search them.
- They can add new elements, delete elements, and modify the content of existing elements.

Difference between DOM and SAX:

DOM:

- It is an official recommendation of the World Wide Web Consortium (W3C).
- It provides interfaces on components of a tree which is a DOM document.
- It creates a tree structure in memory from the input document and then waits for requests from the client.
- It always serves the client application with the entire document no matter how much is actually needed by the client.
- DOM is an open standard.
- The XML file is arranged in a tree fashion.
- DOM supports random access to the data of XML file.

SAX:

- It is an event-driven online algorithm for parsing XML documents.
- SAX stands for Simple API for XML Parsing.
- It provides interfaces on handlers.
- It does not create any internal structure.
- SAX is developed especially for java programs.
- It implements a model that is memory resident.
- To access data from XML file, SAX follows top to bottom approach.

Presenting XML:

- One fundamental characteristic of XML is the separation of the structure, content and presentation of data.
- This course covers creating output presentations using XSL (eXtensible Stylesheet Language).
- With XSL stylesheets, it is possible to transform an XML document into an HTML presentation
- It can transform to any other text presentation, or even another XML document with a different structure.
- One fundamental characteristic of XML is the separation of the structure, content and presentation of data.
- • This course covers creating output presentations using XSL (eXtensible Stylesheet Language).



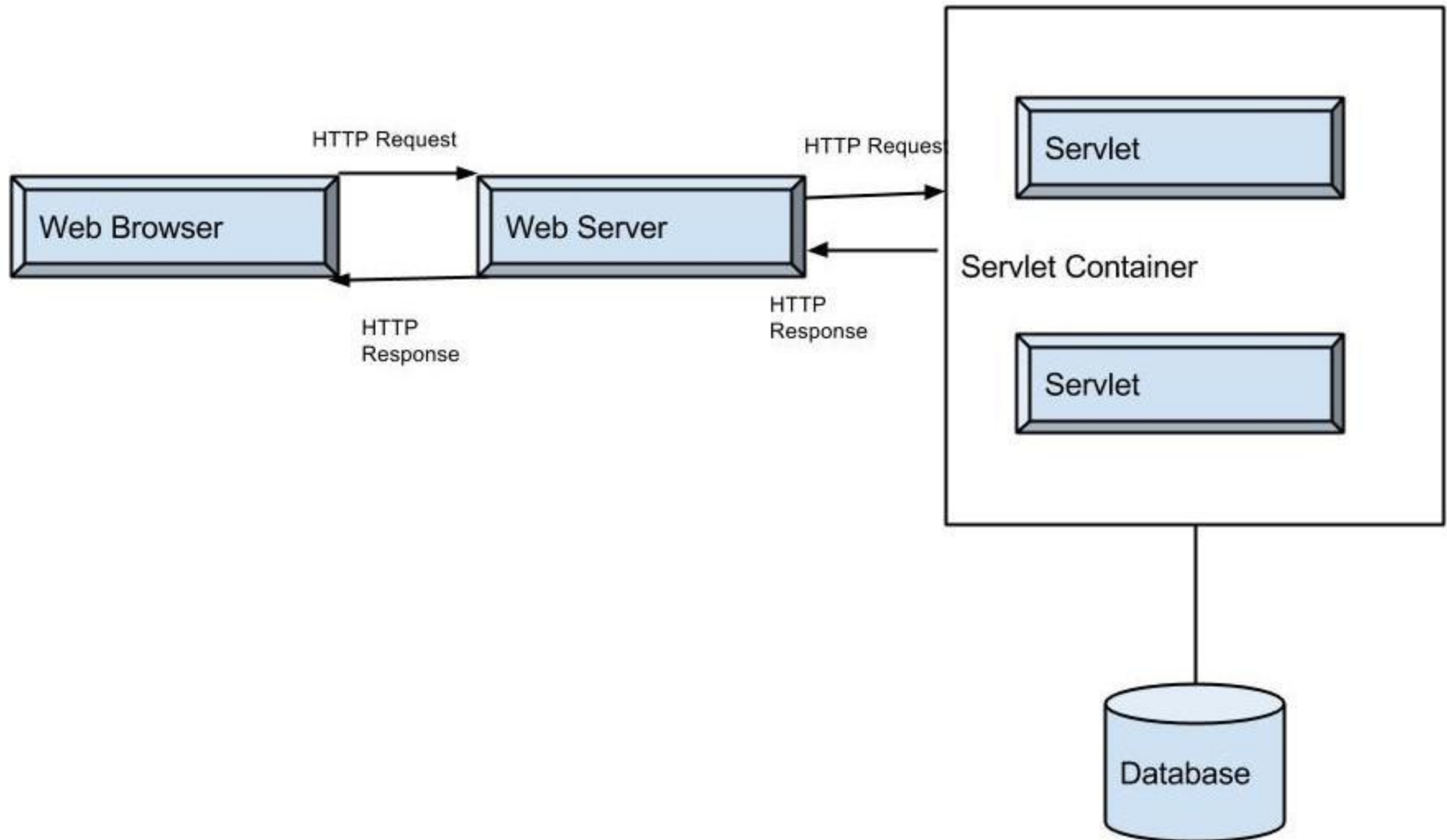
UNIT-III

SERVLETS AND JSP

What is SERVLET

- Servlets are an important component of a J2EE application. Servlets along with JavaServer Pages (JSP) and EJB modules can be termed as server-side J2EE component types.
- Servlet is a Java Programming Language.
- Servlets are used to create web applications.
- Servlets are used to extend the applications hosted by web servers. Servlet runs in a J2EE application server
- Common Gateway Interface (CGI) technology was used for dynamic content prior to introduction of Servlets.

SERVLET ARCHITECTURE



Advantages and Disadvantages of SERVLET

Pro's:

- Servlets provide component based and platform-independent methods for building Web based applications.
- Each Request is run in a separate thread ,so servlet request processing is faster than CGI.
- Servlets overcomes the limitations of CGI program.
- Servlets run on Java Virtual Machine and in any platform and it is simple to write.
- Servlet are more powerful and the performance is better.

Con's:

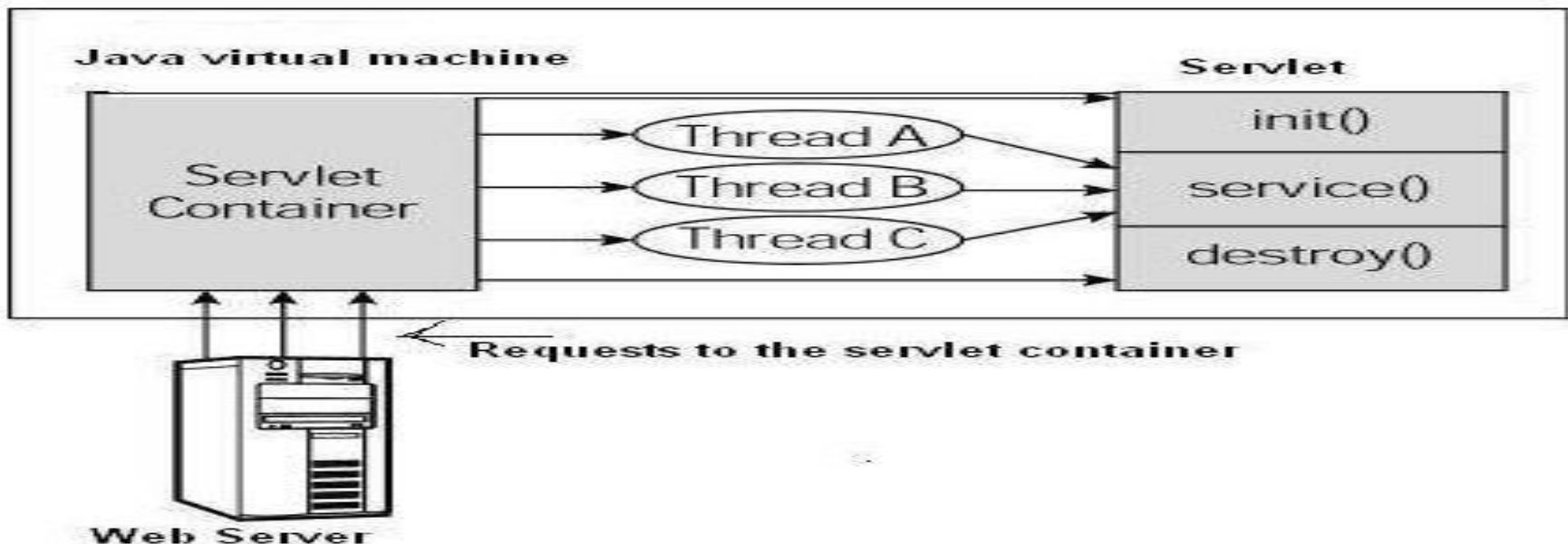
- Servlets often contain both business logic and presentation logic so it makes application difficult to understand.
- You would need JRE to be installed to run a servlet program.

Lifecycle of a SERVLET

- A servlet life cycle can be defined as the entire process from its creation till the destruction.

Servlet Life Cycle Methods:

- The following are the life cycle methods of a servlet instance:
 - `init()`
 - `service()`
 - `destroy()`



init()

- This method is called once for a servlet instance. When first time servlet is called servlet container creates instance of that servlet and loaded into the memory. Future requests will be served by the same instance without creating the new instance. Servlet by default multithreaded application. `init()` method is used for initializing servlet variables which are required to be passed from the deployment descriptor `web.xml`. `ServletConfig` is passed as the parameter to `init()` method which stores all the values configured in the `web.xml`. It is more convenient way to initialize the servlet.

service()

- This method is called for the each request. This is the entry point for the every servlet request and here we have to write our business logic or any other processes. This method takes `HttpServletRequest` and `HttpServletResponse` as the parameters. It is not mandatory to write this method, normally developers are interested in writing `doGet()` or `doPost()` methods which is by default called from the `service()` method. If you override `service()`, it is your responsibility to call the appropriate methods. If you are not overridden the `service()` method, based on the types of the request the methods will be called.

What is SERVLET

destroy()

- This method will be called once for an instance. It is used for releasing any resources used by the servlet instance. Most of the times it could be database connections, File IO operations, etc. `destroy()` is called by the container when it is removing the instance from the servlet container. Servlet instance is deleted or garbage collected by the container only when the web server issues shut down or the instance is not used for a long time.

A Simple SERVLET

- Servlets are Java classes which service HTTP requests and implement the `javax.servlet.Servletinterface`.
- Web application developers typically write servlets that extend `javax.servlet.http.HttpServlet`, an abstract class that implements the Servlet interface and is specially designed to handle HTTP requests.

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
// Extend HttpServlet class
public class HelloWorld extends HttpServlet {
    private String message;
    public void init() throws ServletException {
        // Do required initialization
        message = "Hello World";
    }
    public void doGet(HttpServletRequest request, HttpServletResponse response)throws
    ServletException, IOException {
```

```
response.setContentType("text/html"); // Set response content type
// Actual logic goes here.
PrintWriter out = response.getWriter();
out.println("<h1>" + message + "</h1>");}
public void destroy() {// do nothing.}}
```

Compiling a Servlet:

- set classpath for servlet-api.jar file, Provide command in command prompt as javac HelloWorld.java

Servlet Deployment

- Place the HelloWorld.class file, by default, a servlet application is located at the path <Tomcat-installationdirectory>/webapps/ROOT and the class file would reside in <Tomcat-installationdirectory>/webapps/ROOT/WEB-INF/classes.
- Generate web.xml file for HelloWorld.java application.

Web.xml file:

```
<web-app>  
<servlet>  
<servlet-name>HelloWorld</servlet-name>  
<servlet-class>HelloWorld</servlet-class>  
</servlet>  
<servlet-mapping>  
<servlet-name>HelloWorld</servlet-name>  
<url-pattern>/HelloWorld</url-pattern>  
</servlet-mapping>  
</web-app>
```

- Now, start tomcat server using <Tomcat-installationdirectory> \bin \startup.bat (on Windows) or <Tomcat-installationdirectory> /bin /startup.sh (on Linux/Solaris etc.) and finally type <http://localhost:8080/HelloWorld> in the browser's address box. If everything goes fine, you would get the following result.



THE SERVLET API

The Servlet API is used in order to create servlets. It provides Classes and Interfaces to develop web based applications. It consists of two java packages are used to developed the servlet programs, they are,

- javax.servlet
- javax.servlet.http

1. The javax.servlet package contains a number of classes and interfaces that describe and define the contracts between a servlet class and the runtime environment provided for an instance of such a class by a conforming servlet container.

Interfaces in javax.servlet package: •Servlet

- ServletRequest
- ServletResponse
- ServletConfig
- ServletContext

Classes in javax.servlet package

- GenericServlet
- ServletInputStream
- ServletOutputStream
- ServletException

2. The javax.servlet.http package contains a number of classes and interfaces that describe and define the contracts between a servlet class running under the HTTP protocol and the runtime environment provided for an instance of such a class by a conforming servlet container.

Interfaces in javax.servlet.http package:

- HttpSession
- HttpServletRequest
- HttpServletResponse
- HttpSessionBindingListener

Classes in javax.servlet.http package

- HttpServlet
- Cookie
- HttpSessionEvent
- HttpSessionBindingEvent

The javax.Servlet Package

The javax.servlet package contains a number of interfaces and classes that establish the framework in which servlets operate.

- The following table summarizes the core interfaces that are provided in this package. The most significant of these is Servlet.
- All servlets must implement this interface or extend a class that implements the interface. The Servlet Request and Servlet Response interfaces are also very important.

The Servlet Interface

- All servlets must implement the Servlet interface. It declares the `init()`, `service()`, and `destroy()` methods that are called by the server during the lifecycle of a servlet. A method is also provided that allows a servlet to obtain any initialization parameters. The `init()`, `service()`, and `destroy()` methods are the life cycle methods of the servlet.
- These are invoked by the server. The `getServletConfig()` method is called by the servlet to obtain initialization parameters.
- A servlet developer overrides the `getServletInfo()` method to provide a string with useful information. This method is also invoked by the server.

The javax.Servlet Package

The ServletConfig Interface

- The Servlet Config interface allows a servlet to obtain configuration data when it is loaded.

The ServletContext Interface

- The ServletContext interface enables servlets to obtain information about their environment.

The ServletRequest Interface

- The ServletRequest interface enables servlet to obtain information about a client request.

The ServletResponse Interface

- The ServletResponse interface enables servlet to formulate a response for a client.

Reading Servlet Parameters

The `ServletRequest` interface includes methods that allow you to read names and values of parameters that are included in a client request. We will develop a servlet that illustrates their use.

- A webpage is defined in `PostParameters.htm`, and servlet is defined in `PostParametersServlet.java`
- The HTML source code for `PostParameters.htm` is shown in the following listing. It defines a table that contains two labels and two text fields. One of the labels is Employee and the other is Phone. There is also a submit button. Notice that the action parameter of the form tag specifies a URL. The URL identifies the servlet to process the HTTP POST request.

Reading Servlet Parameters

```
<html><body><center><formname="Form1" method="post"
action="http://localhost:8080/servlets-examples/servlet
/PostParametersServlet">
<table><tr><td><B>Employee</td>
<td><inputtype=textBoxname="e"size="25"value=""></td>
</tr><tr>
<td><B>Phone</td>
<td><inputtype=textBoxname="p"size="25"value=""> </td></tr>
</table>
<inputtype=submitvalue="Submit">
</body>
</html>
```

Reading Servlet Parameters

- ◎ The source code for PostParametersServlet.java is shown in the following listing. The service() method is overridden to process client requests. The getParameterNames() method returns an enumeration of the parameter names. These are processed in a loop. You can see that the parameter name and value are output to the client. The parameter value is obtained via the getParameter() method.

PostParametersServlet.java

```
import java.io.*; import java.util.*; import javax.servlet.*;

public class PostParametersServlet
    extends GenericServlet{

    public void service(ServletRequest request,
        ServletResponse response)
        throws ServletException, IOException{
```

Reading Servlet Parameters

```
PrintWriter pw = response.getWriter();  
//Get enumeration of parameter names.  
Enumeration e = request.getParameterNames();  
//Display parameter names and values.  
while(e.hasMoreElements()){  
    String pname = (String)e.nextElement();  
    pw.print(pname + " = ");  
    String pvalue = request.getParameter(pname);  
    pw.println(pvalue);  
}  
pw.close();  
}  
}
```

Reading Servlet Parameters

- Compile the servlet. Next, copy it to the appropriate directory, and update the web.xml file, as previously described. Then, perform these steps to test this example:
 1. Start Tomcat(if it is not already running).
 2. Display the web page in a browser.
 3. Enter an employee name and phone number in the text fields.
 4. Submit the web page.

The Javax. Servlet. HTTP Package

The Javax. Servlet. HTTP Package

- The javax.servlet.http package contains a number of interfaces and classes that are commonly used by servlet developers. You will see that its functionality makes it easy to build servlets that work with HTTP requests and responses.

Interfaces

HttpServletRequest Enables servlets to read data from an HTTPrequest.

HttpServletResponse EnableservletstowritedataatoanHTTPresponse.

HttpSession Allows session data to be read and written.

HttpSessionBindingListener Informs an object that it is bound too run bound from a session

The Javax. Servlet. HTTP Package

Class

- Cookie Allows state information to be stored on a client machine.
- HttpServlet Provides methods to handle HTTP requests and responses.
- HttpSessionEvent Encapsulates a session-changed event.
- HttpSessionBindingEvent Indicates when a listener is bound too run bound from a session value, or that a session attribute changed.

The HttpServletRequest Interface

- TheHttpServletRequestinterfaceenablesaservlettoobtaininformationab outaclientrequest.

The HttpServletResponse Interface

- The HttpServletResponse interface enables a servlet to formulate an HTTPresponse to a client. Several constants are defined. These correspond to the different status codes that can be assigned to an HTTPresponse.

Handling HTTP Requests and Responses

- The `HttpServlet` class provides specialized methods that handle the various types of HTTP requests. A servlet developer typically overrides one of these methods. These methods are `doDelete()`, `doGet()`, `doHead()`, `doOptions()`, `doPost()`, `doPut()`, and `doTrace()`. A complete description of the different types of HTTP requests is beyond the scope of this topic. However, the GET and POST requests are commonly used when handling form input.

Using Cookies:

Cookies are text files stored on the client computer and they are kept for various information tracking purpose. Java Servlets transparently supports HTTP cookies. The servlet is invoked when a form on a web page is submitted.

Sessions:

HTTP is a state less protocol. Each request is independent of the previous one. However, in some applications, it is necessary to save state information so that information can be collected from several interactions between a browser and a server. Sessions provide such a mechanism.

- A session can be created via the `getSession()` method of `HttpServletRequest`. An `HttpSession` object is returned. This object can store a set of bindings that associate names with objects. The `setAttribute()`, `getAttribute()`, `getAttributeNames()`, and `removeAttribute()` methods of `HttpSession` manage these bindings. It

- A session can be created via the `getSession()` method of `HttpServletRequest`. An `HttpSession` object is returned. This object can store a set of bindings that associate names with objects. The `setAttribute()`, `getAttribute()`, `getAttributeNames()`, and `removeAttribute()` methods of `HttpSession` manage these bindings. It is important to note that session state is shared among all the servlets that are associated with a particular client.

Anatomy of JSP

A JSP page is a web page with JSP elements for generating the parts that differ for each request.

<code><%@ page language="java" contentType="text/html" %></code>	JSP element
<code><html></code>	template text
<code><body bgcolor="white"></code>	
<code><jsp:useBean id="userInfo" class="com.ora.jsp.beans.userInfo.UserInfoBean"> <jsp:setProperty name="userInfo" property="*" /> </jsp:useBean></code>	JSP element
<code>The following information was saved: User Name:</code>	template text
<code><jsp:getProperty name="userInfo" property="userName" /></code>	JSP element
<code>Email Address:</code>	template text
<code><jsp:getProperty name="userInfo" property="emailAddr" /></code>	JSP element
<code> </body> </html></code>	template text

Anatomy of JSP

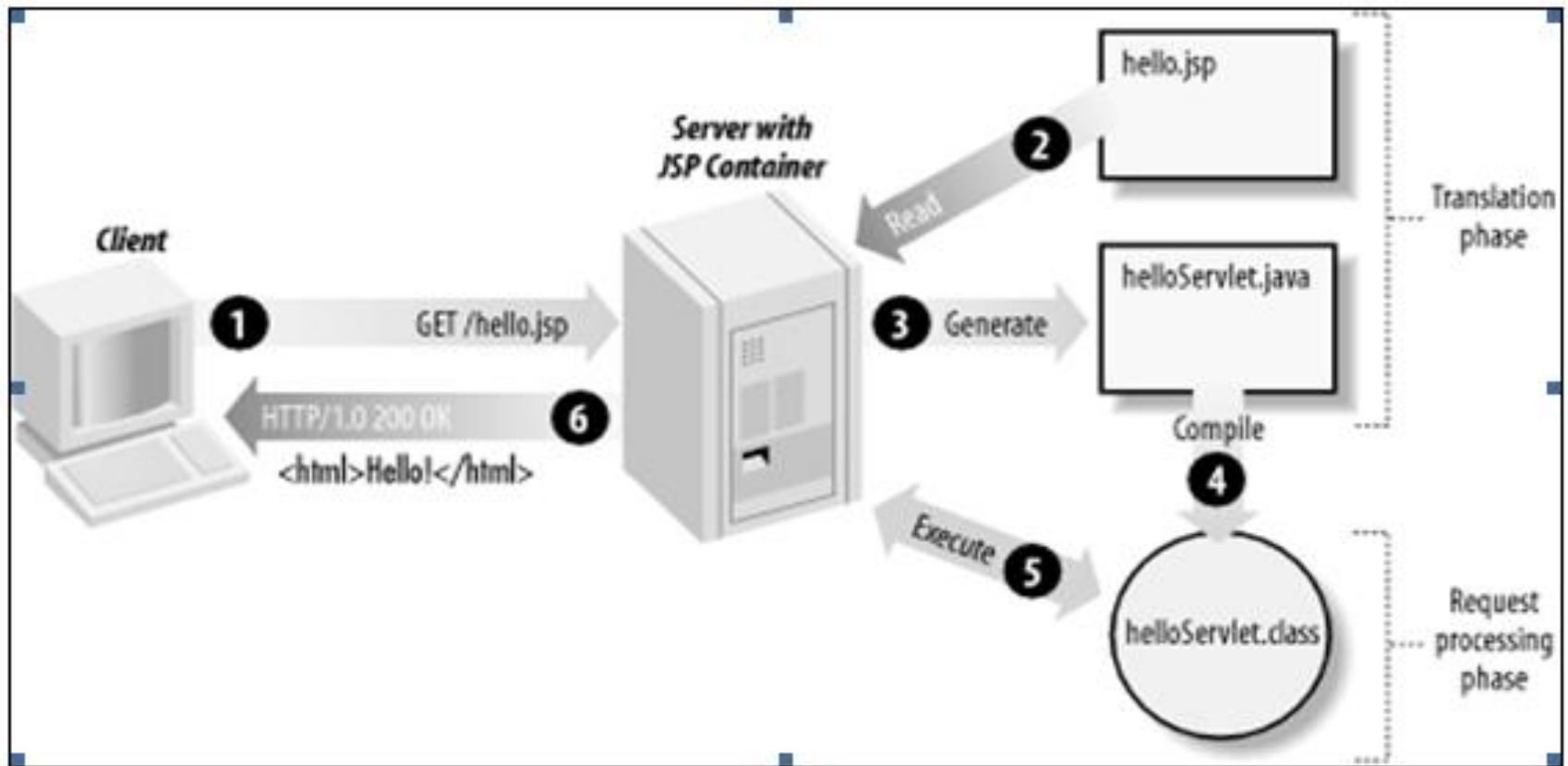
- JSP page is a combination of JSP element and template text.
- Template text can be any text which includes HTML, WML, XML, or even plain text. it can be used with any markup language.
- Template text is always passed directly to the browser.
- When a JSP page request is processed, the template text and dynamic content generated by the JSP elements are merged, and the result is sent as the response to the browser.

JSP Processing

- A web server needs a servlet container to provide an interface to servlets, the server needs a JSP container to process JSP pages.
- The JSP container is responsible for intercepting requests for JSP pages.
- To process all JSP elements in the page, the container first turns the JSP page into a servlet (known as the JSP page implementation class).
- The template text is converted to `println()` statements similar to the ones in the hand-coded servlet and all JSP elements are converted to Java code that implements the corresponding dynamic behavior. The container then compiles the servlet class.
- Converting the JSP page to a servlet and compiling the servlet form the translation phase.

- The JSP container initiates the translation phase for a page automatically when it receives the first request for the page. Since the translation phase takes a bit of time, the first user to request a JSP page notices a slight delay.
- The translation phase can also be initiated explicitly and is referred to as pre-compilation of a JSP page.
- Pre-compiling a JSP page is a way to avoid hitting the first user with this delay.
- The JSP container is also responsible for invoking the JSP page implementation class (the generated servlet) to process each request and generate the response. This is called the request processing phase.

The two phases are illustrated in the figure



Anatomy of JSP

- JSP page is a combination of JSP element and template text.
- Template text can be any text which includes HTML, WML, XML, or even plain text. it can be used with any markup language.
- Template text is always passed directly to the browser.
- When a JSP page request is processed, the template text and dynamic content generated by the JSP elements are merged, and the result is sent as the response to the browser.

- As long as the JSP page remains unchanged, any subsequent request goes straight to the request processing phase (i.e., the container simply executes the class file).
- When the JSP page is modified, it goes through the translation phase again before entering the request processing phase.
- The JSP container is also responsible for invoking the JSP page implementation class (the generated servlet) to process each request and generate the response. This is called the request processing phase.

JSP Declarations

- A JSP declaration declares one or more variables or methods that one can use in Java code later in the JSP file.
- The JSP declaration tag is a piece of java code mainly used to declare fields and methods

SYNTAX

```
<%! declaration; [ declaration; ]+ ... %>
```

JSP Directives

- The JSP directives are messages that tells the web container how to translate a JSP page into the corresponding servlet. A JSP directive affects the overall structure of the servlet class.

Syntax:

```
<%@ directive attribute="value" %>
```

There are three types of directives:

1. page directive
2. include directive
3. tag lib directive

1. The page Directive:

The page directive is used to provide instructions to the container. These instructions pertain to the current JSP page. It defines page-dependent attributes, such as scripting language, error page, and buffering requirements.

Syntax: `<%@ page attribute = "value" %>`

2. The Include Directive

The include directive is used to include the contents of any resource it may be jsp file, html file or text file. The include directive includes the original content of the included resource at page translation time.

Syntax: `<%@ include file="resourceName" %>`

3. The Taglib directive

Syntax:

```
<%@ taglib uri="uriofthetaglibrary" prefix="prefixoftaglibrary" %>
```

Example:

```
<html>
```

```
<body>
```

```
<%@ taglib uri="http://www.javatpoint.com/tags" prefix="mytag"  
" %>
```

```
<mytag:currentDate/>
```

```
</body>
```

```
</html>
```




UNIT-IV

INTRODUCTION TO PHP

PHP INSTALLATION

To start using PHP, you can:

Find a web host with PHP and MySQL support

- Install a web server on your own PC, and then install PHP and MySQL
- Use a Web Host With PHP Support

If your server has activated support for PHP you do not need to do anything.

Just create some .php files, place them in your web directory, and the server will automatically parse them for you.

You do not need to compile anything or install any extra tools.

Because PHP is free, most web hosts offer PHP support.

Set Up PHP on Your Own PC, However, if your server does not support PHP, you must: install a web server, install PHP, install a database, such as MYSQL.

Configuring PHP

Configuring PHP Development Environment in Windows

This tutorial shows you two ways of configuring your PHP development environment on the Windows operating system. The first and most convenient way is to install and configure an AMP (Apache, MySQL, PHP) package. This tutorial shows you how to install the XAMPP package. The second way is to install and configure each component separately.

Contents

- Using an AMP Package
- Installing and Configuring the XAMPP Package
- Checking XAMPP Installation
- Installing and Enabling the XDebug Debugger
- Installing the Components Separately
- Apache HTTP Server

Programming in a Web Environment:

In order to develop and run PHP Web pages, three vital components need to be installed on your computer system.

Web Server - PHP will work with virtually all Web Server software, including Microsoft's Internet Information Server (IIS) but then most often used is freely available Apache Server. Download Apache for free here: <http://httpd.apache.org/download.cgi>

Database - PHP will work with virtually all database software, including Oracle and Sybase but most commonly used is freely available MySQL database. Download MySQL for free here:

<http://www.mysql.com/downloads/index.html>

PHP Parser - In order to process PHP script instructions, a parser must be installed to generate HTML output that can be sent to the Web Browser. This tutorial will guide you how to install PHP parser on your computer.

The Anatomy of a PHP Page

The basic outline of a PHP script consists of:

1. PHP Opening tag
2. Comments
3. Statements
4. PHP Closing tag
5. File name ends .php

hello.php

```
<?php
```

```
//The infamous "Hello, World!"
```

```
echo("Hello, World! \n") ;
```

```
?>
```

Overview of PHP Data Types and Concepts

PHP tag, informs interpreter to execute PHP code `<?php?>`

Short codes `<? ?>` (without the php) can be used but are considered bad form

Comments

Use `#` or `//`

Statements

In this case the `echo()` function outputs a quoted string (as its argument) within its parentheses.

NOTE: Each statement must end with a semi-colon ;

Forgetting the semi-colon, is one of the most common problems!!

Whitespace, to make code more readable

PHP Data Types

Variables can store data of different types, and different data types can do different things.

PHP supports the following data types:

String

Integer

Float (floating point numbers - also called double)

Boolean

Array

Object

NULL

Resource

Operators in PHP

- An operator is a symbol that specifies a particular action in an expression. Many operators may be familiar to you.
- The precedence and associativity of operators are significant characteristics of a programming language. Operator Precedence, associativity, and Purpose.

Operators are classified into following types:

- AssignmentOperators
- ArithmeticOperators
- StringOperators
- IncrementandDecrementOperators
- LogicalOperators
- EqualityOperators
- ComparisonOperators

Expressions and Statement in PHP

Expressions:

- An expression is a phrase representing a particular action in a program. All expressions consist of at least one operand and one or more operators.

A few examples as follow:

- `$a = 5; // assign integer value 5 to the variable $a`
- `$a = "5"; // assign string value "5" to the variable $a`
- `$sum = 50 + $some_int; // assign sum of 50 + $some_int to $sum`
- `$wine = "Zinfandel"; // assign "Zinfandel" to the variable $wine`
- `$inventory++; // increment the variable $inventory by 1`

Conditional Statements:

- Conditional statements make it possible for your computer program to respond accordingly to a wide variety of inputs, using logic to discern between various conditions based on input value. You can use conditional statements in your code to make your decisions. PHP supports following three decision making statements.
- if...else statement – use this statement if you want to execute a set of code when a condition is true and another if the condition is not true
- else if statement – is used with the if...else statement to execute a set of code if one of the several condition is true.
- switch statement – is used if you want to selection of many blocks of code to be executed, use the Switch statement. The switch statement is used to avoid long blocks of if..elseif..elsecode.

Example for each statement:

- <html>
- <body>
- <?php
- \$array = array(1, 2, 3, 4, 5);
- foreach(\$array as \$value)
- {
- echo "Value is \$value
";
- }
- ?>
- </body>
- </html>

Strings:

- A string is a sequence of characters, like "Hello world!".
- PHP provides various string functions to access and manipulate strings.

String Functions:

- `strtolower()`:
 - The `strtolower()` function returns string in lowercase letter.
 - Syntax: `stringstrtolower(string$string)`
- `strtoupper()`:
 - The `strtoupper()` function returns string in uppercase letter.
 - Syntax: `stringstrtoupper(string$string)`
- `ucfirst()`:
 - The `ucfirst()` function returns string converting first character into uppercase. It doesn't change the case of other characters.
 - Syntax: `stringucfirst(string$str)`

Arrays:

- PHP array is an ordered map (contains value on the basis of key). It is used to hold multiple values of similar type in a single variable.

Advantage of PHP Array

- Less Code: We don't need to define multiple variables.
- Easy to traverse: By the help of single loop, we can traverse all the elements of an array.
- Sorting: We can sort the elements of array.

PHP Array Types

There are 3 types of array in PHP.

- Indexed Array
- Associative Array
- Multidimensional Array

PHP Functions:

- PHP function is a piece of code that can be reused many times.
- It can take input as argument list and return value.
- There are thousands of built-in functions in PHP.
- In PHP, we can define Conditional function, Function within Function and Recursive function also.

Advantage of PHP Functions:

- Code Reusability: PHP functions are defined only once and can be invoked many times.
- Less Code: It saves a lot of code because you don't need to write the logic many times. By the use of function, you can write the logic only once and reuse it.
- Easy to understand: PHP functions separate the programming logic. So it is easier to understand the flow of the application because every logic is divided in the form of functions.

PHP User-defined Functions:

- We can declare and call user-defined functions easily

Syntax:

```
function functionname(){//code to be executed}
```

Example:

function1.php

```
<?php
```

```
functionsayHello(){
```

```
echo"HelloPHPFunction";}
```

```
sayHello();//callingfunction
```

```
?>
```

Output:

Hello PHP Function

PHP Function Arguments:

- We can pass the information in PHP function through arguments which is separated by comma.
- PHP supports Call by Value(default), Call by Reference, Default argument values and Variable-length argument list.

Example:

```
functionarg.php
<?php
functionsayHello($name){
echo"Hello$name<br/>";}
sayHello("Sonoo");
sayHello("Vimal");
sayHello("John");
?>
```


PHP Function: Returning Value:

functiondefaultarg.php

```
<?php
```

```
functioncube($n)
```

```
{
```

```
return$n*$n*$n;
```

```
}
```

```
echo"Cubeof3is:".cube(3);
```

```
?>
```

Output:

Cube of 3 is: 27



UNIT-V

PHP AND DATABASE ACCESS

Basic Database Concepts

Databases are sets of data organized in such a way that the information can be easily accessed, managed, and updated.

Databases terms:

- **Table:** it contains data in structured format. In form of rows and columns.
- **Key:** It is a unique value that is sufficient to identify any record in a table.
- **Join:** When we want to access information from more than one table, we should join two tables using join operation.
- **Query:** In SQL this is a statement that is written to gather information from a database.
- **Schema:** Schema is the way to relate or map multiple tables.

Basic Database Concepts

Requirements to connect PHP and MySql

- If we want to use PHP with MySQL, then first of all we need to install a webserver (Mostly Apache) in case of LAMP. then we should install PHP and MySQL both.
- We can check that PHP and MySQL both are installed on our web server by using the following code:

```
<?php  
phpinfo();  
?>
```

Basic Database Concepts

- Save above code with the filename `phpinfo.php` and open it in a browser. If it will display complete details of PHP installation, and there is a section about MySQL, then it is confirmed that MySQL is installed.
- When we want to connect with MySQL database, we need some credentials to complete the process like:
 - Database name
 - Host name
 - User name
 - Password

Basic Database Concepts

- We can use the following code to make a connection with MySQL database. In the following code `mysql_connect()` function is used to make a connection, which receives three parameters (hostname, username, password). After creation of connection successfully, we use `mysql_select()` function to connect with database.
- `<?php`
- `mysql_connect("localhost","username","password") or die(mysql_error());`
- `mysql_select_db("database_name") or die(mysql_error());?>`
- We can also define these credentials in variables and then we can use those variables to make a connection such as:
- `$host="localhost";`
- `$username="root";`
- `$password="guest@123";`
- `$db_name="product";`

Connecting to a MySQL database

Requirements to connect PHP and MySql

- If we want to use PHP with MySQL, then first of all we need to install a webserver (Mostly Apache) in case of XAMPP. Then we should install PHP and MySQL both.
- We can check that PHP and MySQL both are installed on our webserver by using the following code:

```
<?php  
Phpinfo();  
?>
```

- Save above code with the filename phpinfo.php and open it in a browser. If it will display complete details of PHP installation, and there is a section about MySQL, then it is confirmed that MySQL is installed

Connecting to a MySQL database

- `<?php`
- `$db_connect=mysql_connect($host,$username,$password)or die(mysql_error());`
- `$db_found=mysql_select_db($db_name)or die(mysql_error());`
- `?>`
- After completing the task, we can close MySQL connection using the following statement:
- `mysql_close($db_connect);`

Retrieving and Displaying results

Fetching data set in PHP

- When we want to fetch data from MySQL database through PHP query, there would be an integer value that represents whether the query was successful or not? In PHP we can use following functions to fetch the data from database:
- `mysql_fetch_row`: It returns row as an enumerator array
- `mysql_fetch_object`: It returns row as an object
- `mysql_fetch_array`: It returns row as an associative array
- `mysql_result`: It returns one cell of data

Basic Database Concepts

Creating MySQL database using PHP

- We can create MySQL database directly using PHPscript. If we want to create database in mysql, we need to have full privileges on CREATE and DROP statements ofmysql. Following function scan be used for the same:
- `mysql_create_db()`:It is used to create a database with the name specified.
- `mysql_drop_db()`:this function is used to delete the specified database.
- `mysql_query()`:It is used to pass table definition and for drop also.

Modifying,Updating ,Deleting Data

delete.php

- <?php
- include('connect-db.php');
- if (isset(\$_GET['id']) && is_numeric(\$_GET['id']))
- {\$id = \$_GET['id'];
- // delete the entry
- \$result = mysql_query("DELETE FROM players WHERE id=\$id")or
die(mysql_error());header("Location: view.php");
- }else
- {header("Location: view.php");}
- ?>

MVC Architecture: PHP and AJAX

The Model-View-Controller concept involved in software development evolved in the late 1980s.

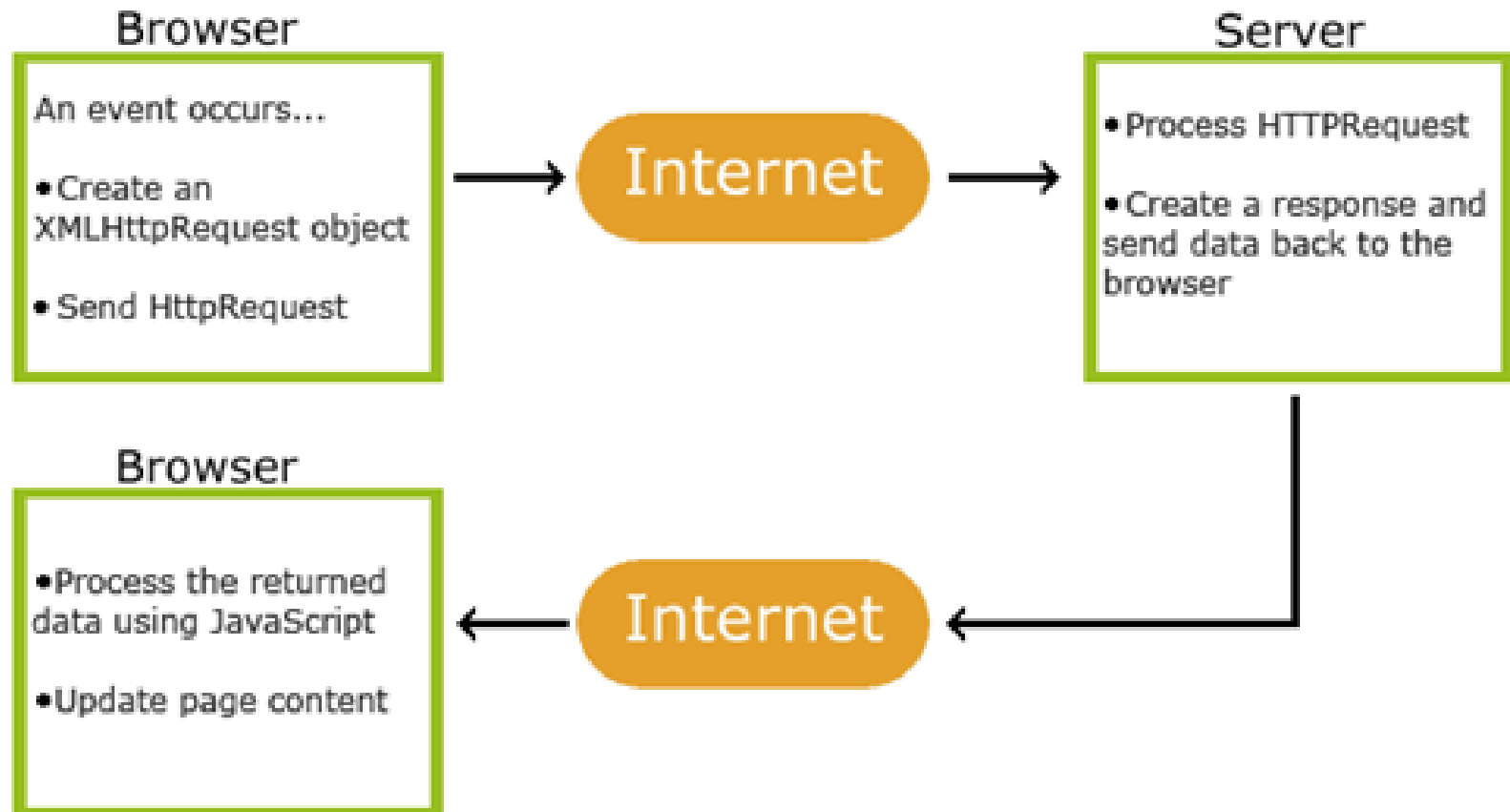
- It's a software architecture built on the idea that the logic of an application should be separated from its presentation.
- A system developed on the MVC architecture should allow a front-end developer and a back-end developer to work on the same system without interfering with each other.

MVC Architecture: PHP and AJAX

AJAX AND PHP:

- AJAX is an acronym for Asynchronous JavaScriptandXML. It is a group of inter-related technologies like JavaScript, DOM, XML, HTML, CSS etc.
- AJAX allows you to send and received at a asynchronously without reloading the webpage. So it is fast.
- AJAX allows you to send only important information to the server not the entire page. Soonly valuable data from the clientside is routed to the serverside. It makes your application interactive and faster.
- Ajax technology isused in web applications running on weblikegmail, facebook, twitter, googlemap, youtube etc.

MVC Architecture: PHP and AJAX



MVC Architecture: PHP and AJAX

AJAX is Based on Internet Standards

- AJAX is based on internet standards, and uses a combination of:
- XMLHttpRequest object (to exchanged at a asynchronously with a server)
- JavaScript/DOM(to display/interact with the information)
- CSS(to style the data)
- XML(often used as the format for transferring data)

- XML is a markup language to share the data across the web, XML is for both human readable and machine readable.

Example of shareable xmls are RSSFeeds. XML parsers are useful to read and update the data by using web browsers.

- Types of XML
 - Tree based
 - Event based

XML Parse Extensions

- XML parse Extensions are works based on libxml. The following xml parsers are available in the php core.
 - Simple XML parser
 - DO XML parser
 - XML parser
 - XML Reader