



---

DYNAMIC HYPERTEXT MARK-UP LANGUAGE  
AND  
JAVASCRIPT

# Objective of the Chapter

---

When students have finished reading this chapter, they will be able to:

- ❖ Work with scripts.
- ❖ Use JavaScript libraries.
- ❖ Load an external script.
- ❖ Add an embedded script.
- ❖ Handle JavaScript events.
- ❖ Understand some important Keywords

# Introduction



---

While HTML defines your webpage's content and CSS defines its presentation, JavaScript defines special behavior. JavaScript is a huge subject.

**DHTML** stands for Dynamic Hypertext Markup language i.e., Dynamic HTML.

Introduced by Microsoft with the release of the 4th version of IE (Internet Explorer) in 1997.

A term describing a series of technologies. Not a stand-a-lone technology (such as HTML, Java)

Browser pages react without server assistance

**Dynamic HTML** is not a markup or programming language but it is a term that combines the features of various web development technologies for creating dynamic and interactive web pages.

# Introduction

---



**Def:** A *dynamic HTML document* is one whose tag attributes, tag contents, or element style properties can be changed after the document has been and is still being displayed by a browser

# DHTML components

---

- DHTML consists of the following four components or languages:
  1. HTML 4.0
  2. CSS
  3. JavaScript
  4. DOM:
    - DOM is the document object model. It is a w3c standard, which is a standard interface of programming for HTML.
    - It is mainly used for defining the objects and properties of all elements in HTML.
- DHTML editors include **Dreamweaver** and **Fusion**

## Difference between HTML and DHTML

**Following table describes the differences between HTML and DHTML:**

HTML (Hypertext Markup language)	DHTML (Dynamic Hypertext Markup language)
1. HTML is simply a markup language.	1. DHTML is not a language, but it is a set of technologies of web development.
2. It is used for developing and creating web pages.	2. It is used for creating and designing the animated and interactive web sites or pages.
3. This markup language creates static web pages.	3. This concept creates dynamic web pages.
4. It does not contain any server-side scripting code.	4. It may contain the code of server-side scripting.
5. The files of HTML are stored with the .html or .htm extension in a system.	5. The files of DHTML are stored with the .dhtm extension in a system.
6. A simple page which is created by a user without using the scripts or styles called as an HTML page.	6. A page which is created by a user using the HTML, CSS, DOM, and JavaScript technologies called a DHTML page.
7. This markup language does not need database connectivity.	7. This concept needs database connectivity because it interacts with users.

# Uses of DHTML



- It allows you to build rich client interfaces and to modify them dynamically
- Can be used to create animations, games, applications.
- Dynamic building of web pages is simple as no plug-in is required.
- Facilitates the usage of events, methods and properties and code reuse. □ It makes the Web experience faster and more interactive for end users.

# Keywords

---

**JavaScript:** Defines special behaviors for a webpage.

**JavaScript libraries:** Add simple interactivity and sophisticated behavior to pages, which help pages behave consistently across browsers.

**jQuery:** Most widespread among JavaScript libraries.

**External script:** Loads from an external file.

**Embedded scripts:** A script that exists in the HTML document.



# Keywords (Continue)

---

**Blocking behavior:** Browser neither downloads nor renders any content appearing after script element. Affects rendering speed of page.

**Minify:** Quick way to speed up script loading is to combine JavaScript into single file (or into as few as possible) and minify the code.

**JavaScript events:** Specific, predefined events triggered by visitor or browser.

**onblur:** JavaScript event where visitor leaves element that was previously in focus.

**onchange:** JavaScript event where visitor modifies value or contents of element.

# Keywords (Continue)

---

**Blocking behavior:** Browser neither downloads nor renders any content appearing after script element. Affects rendering speed of page.

**Minify:** Quick way to speed up script loading is to combine JavaScript into single file (or into as few as possible) and minify the code.

**JavaScript events:** Specific, predefined events triggered by visitor or browser.

**onblur:** JavaScript event where visitor leaves element that was previously in focus.

**onchange:** JavaScript event where visitor modifies value or contents of element.

# Keywords (Continue)

---

**onclick:** JavaScript event where visitor clicks specified area or hits Return or Enter key while focused on it (like on a link).

**ondblclick:** JavaScript event where visitor double-clicks specified area.

**onfocus:** JavaScript event where visitor selects, clicks, or tabs to specified element.

**onkeydown:** JavaScript event where visitor presses down on a key while in specified element.

**onkeypress:** JavaScript event where visitor presses down and lets go of a key while in specified element.

# Keywords (Continue)

---

**onkeyup:** JavaScript event where visitor lets go of key after typing in specified element.

**onload:** JavaScript event where browser finishes loading page, including all external files (images, style sheets, JavaScript, and so on).

**onmousedown:** JavaScript event where visitor presses mouse button down over specified element.

**onmousemove:** JavaScript event where visitor moves mouse cursor.

**onmouseout:** JavaScript event where visitor moves mouse away from specified element after having been over it.

# Keywords (Continue)

---

**onmouseover:** JavaScript event where visitor points mouse at element.

**onmouseup:** JavaScript event where visitor lets mouse button go after having clicked the element (the opposite of onmousedown).

**onreset:** JavaScript event where visitor clicks form's reset button or presses Return or Enter key while focused on the button.

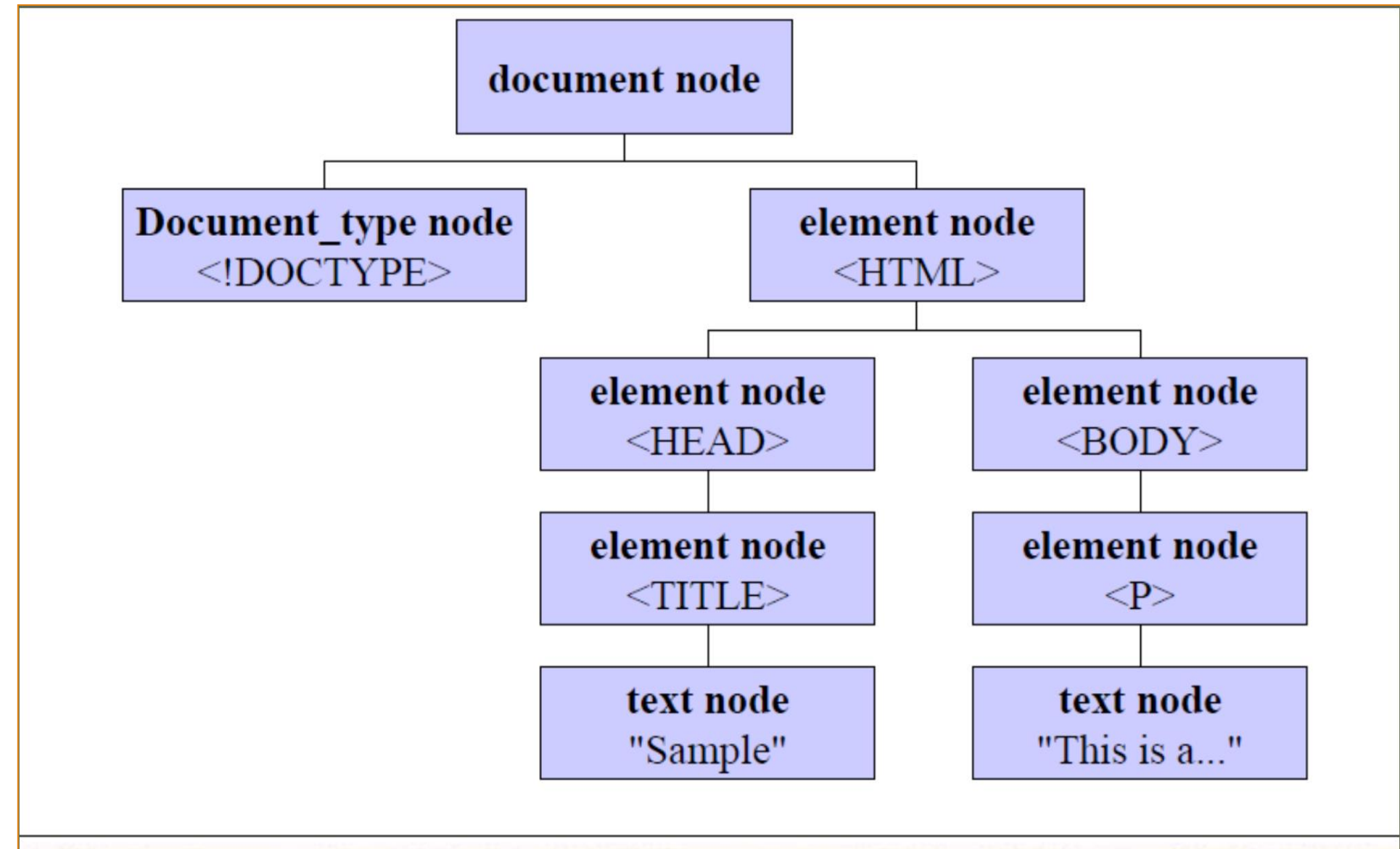
**onselect:** JavaScript event where visitor selects one or more characters or words in the element.

**onsubmit:** JavaScript event where visitor clicks form's submit button or presses Return or Enter key while focused on the button.

# DOM: The Document object Model

- DOM: Defines all items on a Web page as objects that can be manipulated.
- Otherwise, those elements are set, defined solely by the browser.

*Example, the DOM makes it possible to identify individual letters on a page as separate objects, and then to assign specific qualities, such as color or size, to each letter. In a sense, every letter becomes a tiny Web page.*



# DHTML

## Document Object Model (DOM)

Connects an object

Element defined with an *ID* or *name* attribute (CSS) to a JavaScript function

Allows any property to be changed

Connects the object with a JavaScript function

## Identifying an object

### Netscape

*Name* to identify an image

*ID* to identify elements that are NOT images

*ID* to identify an image

### MS IE

*Name* to identify an image

*ID* to identify elements that are NOT images

```

<div id="layer1"></div>
```

**Layers: objects using an ID attribute**

## Events

Occur when something happens in the browser window

Action samples

Loading of a new document

Leaving a web page

Mouse movement

Key is pressed

Browser window is moved

## Event Handlers

Programs built into the browser

**An internal program in the browser (Event Handler) is activated upon an occurrence of an event.**

Specific program agents handle specific events

Connects and action in the window to a JavaScript function which causes some reaction.



# Bootstrap

---

- ❖ It is the popular HTML, CSS and JavaScript framework for developing a responsive and mobile friendly website.

- ❖ <https://www.javatpoint.com/bootstrap-tutorial>

- ❖ Bootstrap was developed by **Mark Otto and Jacob Thornton at Twitter**.

- ❖ Released as an open source product in August 2011 on GitHub.

- ❖ In June 2014 Bootstrap was the No.1 project on GitHub.

## Advantage of Bootstrap:

It is very easy to use. Anybody having basic knowledge of HTML and CSS can use Bootstrap.

It facilitates users to develop a responsive website.

It is compatible on most of browsers like Chrome, Firefox, Internet Explorer, Safari and Opera etc



# jQuery

jQuery is a small and lightweight JavaScript library.

jQuery is cross-platform.

jQuery means "write less do more".

jQuery simplifies AJAX call and DOM manipulation.

<https://www.javatpoint.com/jquery-tutorial>

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>First jQuery Example</title>
```

```
<script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">
```

```
</script>
```

```
<script type="text/javascript" language="javascript">
```

```
$(document).ready(function() {
```

```
  $("p").css("background-color", "pink");
```

```
});
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<p>This is first paragraph.</p>
```

```
<p>This is second paragraph.</p>
```

```
<p>This is third paragraph.</p>
```

```
</body>
```

# JSON

---

JSON stands for JavaScript Object Notation.

JSON is lightweight data-interchange format.

JSON is easy to read and write than XML.

JSON is language independent.

JSON supports array, object, string, number and values.

<https://www.javatpoint.com/json-tutorial>

# AJAX

---

**AJAX** is an acronym for **Asynchronous JavaScript and XML**.

It is a group of inter-related technologies like JavaScript, DOM, XML, HTML/XHTML, CSS, XMLHttpRequest etc.

AJAX allows you to send and receive data asynchronously without reloading the web page. So it is fast.

AJAX allows you to send only important information to the server not the entire page.

It makes your application interactive and faster.

<https://www.javatpoint.com/ajax-example>

# Node.js

---

**Node.js** = Runtime Environment + JavaScript Library

**Node.js** is a cross-platform runtime environment and library for running JavaScript applications outside the browser. It is used for creating server-side and networking web applications.

It is open source and free to use.

It can be downloaded from this link <https://nodejs.org/en/>

<https://www.javatpoint.com/nodejs-tutorial>

# Advantages DHTML

Supported by most browsers

Small file sizes (text files)

No plug-ins required

Easy to learn

HTML

JavaScript

Fast Development

HTML

JavaScript

Faster Web Experience

Manipulate browser objects on the client side

Java not required

- DHTML can make your browser dynamic and interactive.
- Validation of input's given by the user can be done at the client side, without connection to the server.
- Content and design can be separated using Style sheets & uniformity of the site can be maintained using DHTML
- DHTML is fairly fast when loading depending on your PC,
- It enables the web-page to look up-to-date & interactive,
- It enables the web-page to look more professional by having the roll-over buttons, the drop-down bars, etc.
- XML DHTML is more efficient for content management purposes, By using more dynamic XML to create the pages, you will be able to import high volumes of text and other information (with the templates use) without spending the time coding pages individually.



# DISADVANTAGES OF DHTML

- **Editors' available in market are very expensive.**
- **Creating and editing your website involves lots of detailed coding**
- **When writing the code, if you don't check what is compatible with different browsers, It will not work and you will waste your time.**
- **Browser and OS incompatibilities**



Variations in browser vendors (Netscape vs IE)

Variations in browser versions (same vendor)

Variations of OS

Variations of OS Versions

## Browser Specific Technologies

### Netscape

JavaScript Style Sheets

Netscape layers (CSS prototypes)

### Microsoft

ActiveX (GUI objects)

Visual Filters (visual effects on graphics and text)

## Cross Browser Technologies

Cascading style sheets (CSS)

JavaScript

Document Object Model (DOM)

(connect window objects to scripts)

# DHTML

## Exercise 1:

```
<html>
<HEAD>
<script type="text/JavaScript">
function square(number)
{
return number * number
}
</script>
</head>
<body>
<script>
document.write("The function returned ", square(5), ".")
</script>
</body>
</html>
```



## **Exercise 2: (1)**

```
<html>
```

```
<script type="text/javascript">
```

```
function Time( )
```

```
{
```

```
var currentTime = new Date( )
```

```
var hours = currentTime.getHours( )
```

```
var minutes = currentTime.getMinutes( )
```

```
var seconds = currentTime.getSeconds( )
```

```
// call the AddZero function to add a zero in front of numbers that are less than 10
```

```
hours = AddZero(hours)
```

```
minutes = AddZero(minutes)
```

```
seconds = AddZero(seconds)
```

## Exercise 2 Continues: (2)

```
// write out the time
```

```
document.write("<b>" + hours + "hours" + minutes + "minutes" + seconds
```

```
+"seconds
```

```
</b>")
```

```
}
```

```
function AddZero(i)
```

```
{
```

```
if (i<10)
```

```
{i="0"+i}
```

```
Return i
```

```
}
```

```
</script>
```

```
</head>
```

```
// call the Time function
```

```
<body onload="Time( )">
```

```
</body> </html>
```

### Exercise 3: Receiving User Input

**Step 1:** Create a new HTML file, insert the script below into the head.

```
<script language="javascript" type="text/javascript">
<!-- var selected="1";
function changeBGColor( )
{ switch(selected)
{
  case ("1"): document.body.style.backgroundColor="#99ffff";
break;
case ("2"):
document.body.style.backgroundColor="#ff99ff";
break;
case ("3") : document.body.style.backgroundColor="#ffff99";
break;
default:
document.body.style.backgroundColor="white";
break; } } / -->
</script>
```

**Step 2:** creates a button that when clicked calls the function changeBGColor( ). Insert the following code into the body of your HTML file.

```
<form name="userinput" id="userinput"> <input type="button" onClick = "changeBGColor()" value="Click here" /> </form>
```

**Step 3:** insert the following code **immediately before the switch(selected) line** in the JavaScript function.

```
selected = window.prompt("Select a background color:\n" + "0 (white), 1 (teal), 2 (lavender), or 3 (yellow)", "1" );
```

# Group Assignment 3

---

Using a combination of all DHTML technologies, create and tracks the movements of an object.

Or

Creates new content dynamically by using the DOM manipulation to render animation and live data, etc.