3

HTML Basics

एचटीएमएल बेसिक्स

Web Designing and Publishing

3.1 Introduction of HTML - एचटीएमएल एक परिचय

HTML stands for Hyper Text Markup Language. HTML is the most widely used language on Web to develop web pages. In Late 1991 **Berners-Lee** created HTML but "HTML 2.0(1995)" was the first standard HTML. Web browsers can read HTML files and render them into visible or audible web pages. Browsers do not display the HTML tags and scripts, but use them to interpret the content of the page.

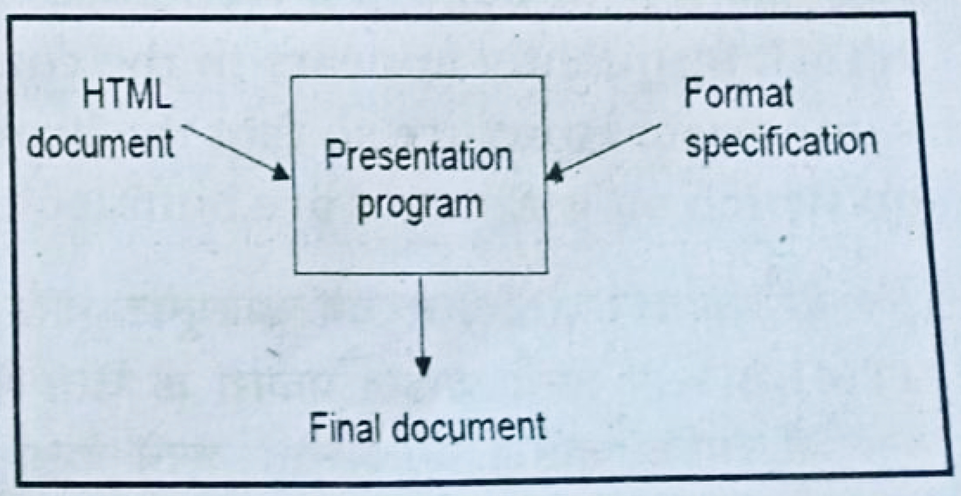
HTML का तात्पर्य हाइपर टेक्स्ट मार्कअप लैंग्वेज। वेब पेज के निर्माण के लिए ये लैंग्वेज सबसे अधिक लोकप्रिय है। 1991 में बर्नस ली ने इसका अविष्कार किया था परन्तु 1995 में HTML 2.0 को पहला मानक HTML बनाया गया था। ब्राउजर वेब पेज को आसानी से पढ़ कर उसे हमारे लिए देखने के योग्य बनाता है। ब्राउज़र पर स्क्रिप्ट नहीं दिखती बल्कि उस पेज का डेटा दिखता है। ब्राउजर HTML टैग और स्क्रिप्ट प्रदर्शित नहीं करते हैं, लेकिन पृष्ठ की सामग्री की व्याख्या करने के लिए उनका उपयोग करते हैं।

In another word: **Hyper Text:** Hyper Text simply means "Text within Text". A text has a link within it, is a hypertext. Every time when you click on a word which brings you to a new webpage, you have clicked on a hypertext.

अन्य शब्दों में, हाइपर टेक्स्टः हाइपर टेक्स्ट का अर्थ है "टेक्स्ट के भीतर टेक्स्ट"। एक टेक्स्ट के भीतर एक लिंक है, जो हाइपरटेक्स्ट है। हर बार जब आप एक शब्द पर क्लिक करते हैं जो आपको एक नए वेबपेज पर लाता है, तो आपने एक हाइपरटेक्स्ट पर क्लिक किया है।

Markup language: A markup language is a programming language that is used make text more interactive and dynamic. It can turn a text into images, tables, links etc. HTML documents are ASCII text files that can be edited using simple editors.

मार्कअप लैंग्वेजः एक मार्कअप भाषा एक प्रोग्रामिंग भाषा है जिसका उपयोग टेक्स्ट को अधिक इंटरैक्टिव और डॉयनॉमिक बनाने के लिए किया जाता है। यह एक टेक्स्ट को इमेज, टेबल, लिंक आदि में बदल सकता है। HTML में बनी फाइल ASCII टेक्स्ट फाइल होती है, जिसे किसी भी साधारण एडिटर से एडिट कर सकते है।



HTML allows you to specify the document specification separate from the formatting rules.

- HTML is used for the document specification, that is, it is used to specify the logical structure of the document. For example, we can mark some text as being a section header without any specification of what the header looks like.

HTML का प्रयोग डॉक्यूमेंट बनाने के लिए होता और वह किसी भी डॉक्यूमेंट की तार्किक संरचना करता है। उदाहरण के लिए, किसी भी टेक्स्ट को हम हेडर बना सकते हैं बिना यह जाने कि उस हेडर का रूप कैसा होगा।

- Format information is provided separately. For example, we can provide a format specification for a section header, specifying that it should appear as a 14 pt. font in boldface, left-justified.
  फॉर्मेट की जानकारी अलग से होती है। जैसे कि सेक्शन हेडर के लिए हम ये 14 पॉइंट, बोल्ड रूप में बायीं ओर जस्टीफाइड रूप में दिखना चाहिए।
- The basic idea is that a presentation program (Netscape is an example of such a program) takes the HTML specification and a format specification and creates the actual document.
  आधारभूत बात यह है कि एक प्रजेंटेशन प्रोग्राम जिसका उदाहरण नेटस्केप नेविगेटर है को HTML के फॉर्मेट को लेकर एक वास्तविक डॉक्यूमेंट का निर्माण करता है।

HTML Versions: Since the early days of the web, there have been many versions of HTML

Version	Year
HTML	1991
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2014

HTML Tags: HTML tags are keywords (tag names) surrounded by angular brackets.
एचटीएमएल टैग्स कीवर्ड (टैग नेम) है, जो एंगुलर ब्रैकेट के अन्दर होते है।

<tag_name> content </tag_name>

- HTML tags normally come **in pairs** like <HTML> and </HTML>
  एचटीएमएल टैग आमतौर पर जोड़े में आतें हैं <HTML> and </HTML>
- The first tag in a pair is the **start tag,** the second tag is the **end tag.**
  एक जोड़ी का पहला टैग स्टार्ट टैग, तथा दूसरा टैग एण्ड टैग होता है।
- The end tag is written like the start tag, but with a slash (/) before the tag name.
  अंतिम टैग को स्टार्ट टैग की तरह ही लिखा जाता है, लेकिन टैग नेम से पहले ब्रेकेट के अन्दर स्लैश (/) का चिन्ह लगाया जाता है।

Note: The start tag is often called the **opening tag**. The end tag is often called the **closing tag**.
स्टार्टिंग टैग को अक्सर ओपनिंग टैग कहा जाता है। एंड टैग को अक्सर क्लोजिंग टैग भी कहा जाता है।

3.2 Basic Structure of HTML - एचटीएमएल की मूल संरचना

A basic HTML page is a web document that normally has the file extension **.html** or **.htm**, though HTML frequently appears in the content of other file types as well. All HTML documents follow the same basic structure so that the browser that renders the file knows what to do. The basic structure on which all webpages are built looks like this:

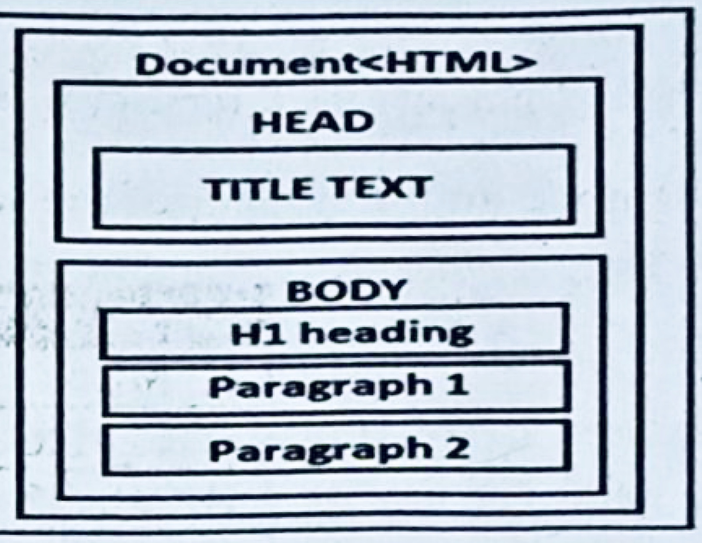
एक बेसिक HTML पेज, वेब डाक्यूमेंट होता है जिसमें आम तौर पर फाइल एक्सटेंशन .html या .htm होता है, हालांकि HTML अक्सर अन्य फाइल प्रकारों के डेटा में भी दिखाई देता है। सभी HTML डाक्यूमेंट एक ही बेसिक स्ट्रक्चर को फॉलो करते हैं ताकि फाइल को प्रस्तुत करने वाला ब्राउजर समान रूप से यह जाने कि उसे HTML निर्देशों के साथ क्या करना है। बेसिक स्ट्रक्चर, जिस पर सभी वेब पेज बनाए जाते हैं, कुछ इस तरह दिखता है:

An HTML document has two main parts : एचटीएमएल डॉक्यूमेंट के दो मुख्य भाग होते हैं :

1. **Head:** The head element contains title and Meta data of a web document.
   हेड: हेड एलिमेंट में किसी भी वेब डॉक्यूमेंट के हेड में टाइटल और मेटा डेटा होता है।
2. **Body:** The body element contains the information that you want to display on a web page.
   बॉडी: बॉडी एलिमेंट में वह जानकारी होती है जिसे आप वेब पेज पर प्रदर्शित करना चाहते हैं।

Below figure shows the basic structure of HTML ...
नीचे का फीगर एचटीएमएल के बेसिक स्ट्रक्चर को दर्शाता है–

```
<!DOCTYPE HTML>
<HTML>
  <HEAD>
     <TITLE> TITLE TEXT </TITLE>
  </HEAD>
  <BODY>
     <H1>Heading</H1>
     <P>Paragraph 1</P>
     <P>Paragraph 2</P>
  </BODY>
</HTML>
```



Besides this, HTML page has also some other elements which are given below:
इसके अलावा एचटीएमएल पेज में कुछ अन्य एलिमेन्ट भी हैं, जो नीचे दिए गए हैं–

- The first line of code, **<!DOCTYPE html>**, is called a doctype declaration and tells the browser which version of HTML the page is written in. In this case, we're using the doctype that corresponds to HTML5, the most up-to-date version of the HTML language. There are a number of different doctype declarations that correspond to various versions of HTML.

  कोड की पहली लाईन, <!DOCTYPE html>, एक doctype डिक्लेरेशन कहलाती है और ब्राउजर को बताती है कि पेज का HTML code किस वर्जन में लिखा गया है। इस मामले में, हम doctype का उपयोग कर रहे हैं जो HTML5 से मेल खाता है, और एचटीएमएल लैंग्वेज का सबसे अधिक अपडेटेड वर्जन है। HTML के विभिन्न वर्जनो के अनुरूप कई अलग–अलग डॉक्टाइप डिक्लेरेशन बनाए गए हैं।

- The **<html> tag** tells the browser that this is an HTML document. The <html> tag represents the root of an HTML document. The <html> tag is the container for all other HTML elements (except for the <!DOCTYPE> tag).

  <html> टैग ब्राउजर को बताता है कि यह एक HTML डॉक्यूमेंट है। <html> टैग एक HTML डॉक्यूमेंट के रूट को रिप्रेजेन्ट करता है। <html> टैग अन्य सभी HTML एलिमेन्टो (<! DOCTYPE>टैग को छोड़कर) के लिए कंटेनर की तरह कार्य करता है।

- The**<head> tag** is another tag used in the HTML document used at the initial and primary level to put a heading to be displayed in the browser window. We put the information between the opening and closing of this tag which gives the information about the page. <Head> टैग एक अन्य टैग है जिसका उपयोग HTML डॉक्यूमेंट में प्रारंभिक और प्राथमिक स्तर पर उपयोग किया जाता है ताकि ब्राउजर विंडो में प्रदर्शित होने के लिए शीर्षक डाला जा सके। हम अपने इन्फार्मेशन को ओपनिंग और क्लोजिंग टैग के बीच रखते हैं ताकि यह पेज के बारे मे कुछ इन्फार्मेशन बताये।

- **<title> tag** is a primary level or entry level HTML tag which is used to assign title information at the title bar to be displayed at the title bar of a web browser you will be using.

<title> टैग एक प्राथमिक स्तर या प्रवेश स्तर का HTML टैग है, जिसका उपयोग टाइटल बार पर टाइटल की जानकारी प्रदान करने के लिए किया जाता है जिसे आप उपयोग किए जा रहे वेब ब्राउजर के टाईटल बार में प्रदर्शित करेंगे।

- The text between **<body>.....</body>** tag describes the visible page content. This is another HTML tag which is considered the main part of the HTML document which contains the entire information related to a web page tried to be shown in the web page by web designer. As you ended every tag you used in the HTML document earlier above, in the same way you will be putting the closing tag of <body> tag following with </body>.

  <body>.....</body> टैग में लिखा गया टेक्स्ट किसी पेज के कन्टेट जो ब्राउजर पर दिखाई देंगे का वर्णन करता है। यह एक और HTML टैग है जिसे HTML डाक्यूमेंट का मुख्य भाग माना जाता है जिसमें वेब पेज से संबंधित पूरी जानकारी होती है जिसे वेब डिजाइनर द्वारा वेब पेज में दिखाने की कोशिश की जाती है। जैसा कि आपने ऊपर HTML डॉक्यूमेंट में उपयोग किए गए प्रत्येक टैग को बंद कर दिया था, उसी तरह आप <body> टैग को भी क्लोसिंग टैग </body> के साथ लगायेंगे और बन्द कर देंगे।

- The text between <h1> and </h1> describes a heading.
  <h1> और </ h1> के बीच का टेक्स्ट एक शीर्षक का वर्णन करता है।

- The text between and describes a paragraph.
   और </ p> के बीच का टेक्स्ट पैराग्राफ का वर्णन करता है।

3.2.1 Writing a Program Using HTML in Notepad - नोटपैड पर एचटीएमएल का उपयोग करके एक प्रोग्राम लिखना

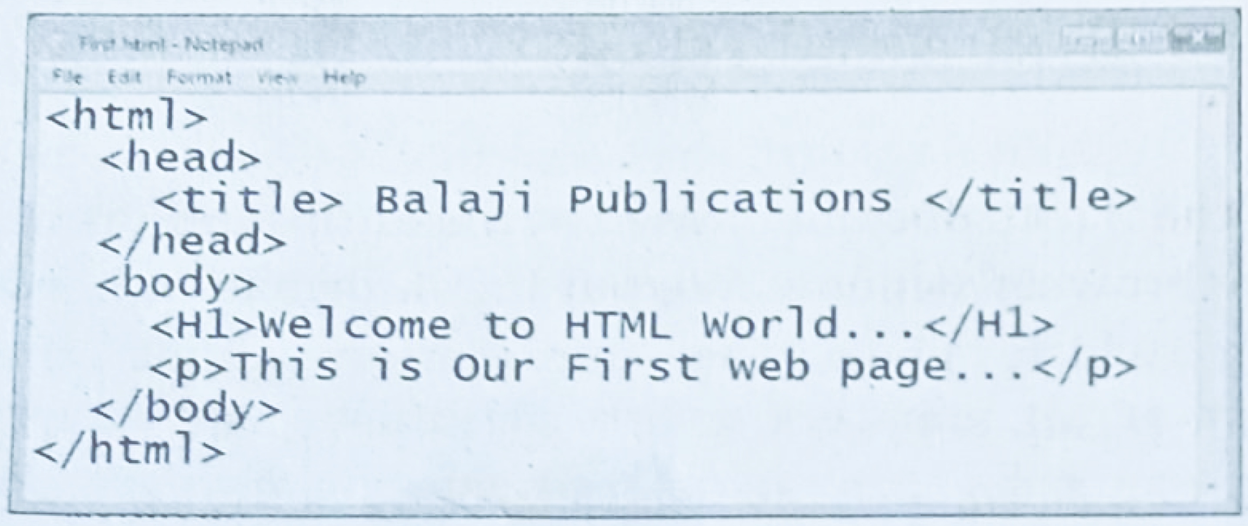
However, for learning HTML we recommend a text editor like Notepad (PC) or TextEdit (Mac). We believe using a simple text editor is a good way to learn HTML. Follow the steps given below to create your first web page with Notepad.

हालांकि, HTML सीखने के लिए हम नोटपैड (पीसी) या टेक्स्ट एडिट (मैक) जैसे टेक्स्ट एडिटर पर कार्य करने की सलाह देते हैं। हम मानते हैं कि एक साधारण टेक्स्ट एडिटर का उपयोग करना HTML सीखने का एक अच्छा तरीका है। नोटपैड के साथ अपना पहला वेब पेज बनाने के लिए नीचे दिए गए चरणों का पालन करें।

Step 1 : Click on Start button-->Select All Programs-->Accessories-->Select Notepad and Click it.

स्टार्ट बटन पर क्लिक करें –>सेलेक्ट आल प्रोग्राम–> उसके बाद एसेसरीज पर क्लिक करें, और अंत में नोटपैड सेलेक्ट करके क्लिक करें।

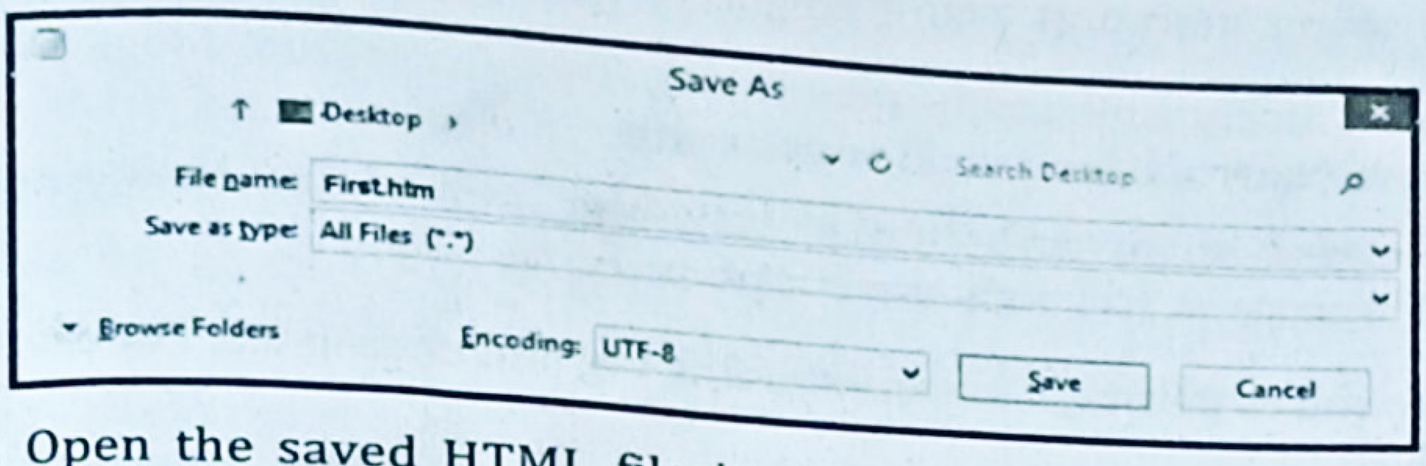
Step 2 : Write HTML coding. HTML कोडिंग को लिखें



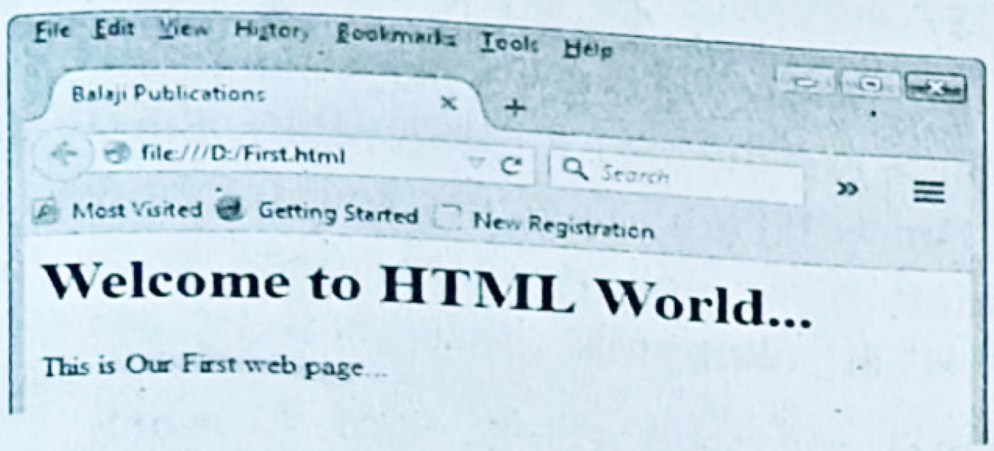
Step 3: Select **File Menu--> Save as** in the Notepad Application.
File Menu--> Save as को सेलेक्ट करे नोटपैड एप्लिकेशन में।

Step 4 : Name the file **"First.html"** or any other name ending with **.html** or **.htm**.

फाइल का नाम "First.html " या .html या .htm के साथ समाप्त होने वाला कोई अन्य नाम लिख कर सेव करें।



Step 4 : Open the saved HTML file in your favorite browser. The result will look much like in browser window. अपने पसंदीदा ब्राउजर में सेव की गई HTML फाइल खोलें। आपका परिणाम ब्राउजर विंडो में दिखाई देगा।



3.2.2 What is an Element? - एलीमेंट क्या है?

An HTML element is an individual component of an HTML document or web page, once this has been parsed into the Document Object Model. HTML is composed of a tree of HTML elements and other nodes, such as text nodes. Each element can have HTML attributes specified. Elements can also have content, including other elements and text. Many HTML elements represent semantics, or meaning. For example, the title element represents the title of the document.

एक HTML एलिमेन्ट पूरे HTML डॉक्यूमेंट का या किसी वेब पेज का एक घटक या कॉम्पोनेन्ट होता है, एक बार यह डॉक्यूमेन्ट आब्जेक्ट मॉडल मे पार्स किया जाता है। एचटीएमएल ने एचटीएमएल एलिमेन्ट्स के ट्री और अन्य नोड्स जैसे टेक्स्ट नोड्स को कम्पोज किया है। प्रत्येक एलिमेन्ट के HTML एट्रिब्यूट निर्धारित एवं सुनिश्चित होते हैं। एलिमेन्ट्स के रूप में डेटा की सामग्री भी हो सकती है, जिसमें कोई और एलिमेन्ट भी सम्मिलित हो सकता है। ये एलिमेन्ट अपने नाम से अपने कार्य का प्रदर्शन करते हैं जैसे टाइटल एलिमेन्ट, वेब पेज के टाइटल यानी शीर्षक को बताता है।

HTML element can be categorized as .

1. Container element 2. Empty element.

HTML एलिमेंट को कंटेनर एलिमेंट एवं एम्प्टी एलिमेंट के रूप में वर्गीकृत किया गया है :

1. **Container Element:** Container Elements can hold one or more Elements, positioned vertically or horizontally, to provide sophisticated control over layout and rendering. Because they enable extremely flexible positioning, Container Elements can function like HTML tags, but with the following additional advantages.

   कंटेनर एलिमेंट के पास एक से अधिक एलिमेंट का प्रतिनिधित्व जो वर्टिकली या हारिजेन्टली रूप से पोजीशन होता है जिससे कि किसी भी वेब पेज के लेआउट पर पूरा नियंत्रण बना रहता है। कंटेनर एलिमेंट्स भी HTML के <DIV> की भांति कार्य करता है, और विभिन्न निम्नलिखित लाभ प्रदान करता है।

   a. **HTML5 formatting options** provide the benefits of the latest features, including outline sectioning and multimedia, while remaining compatible with existing markup.

HTML 5 के फॉर्मेटिंग ऑप्शन नवीनतम सुविधाओं का लाभ प्रदान करते हैं, जिसमें आउटलाइन सेक्शनिंग और मल्टीमीडिया शामिल हैं, जबकि मौजूदा मार्कअप के साथ संगत है।

b. **Render handler support** quickly and easily generating Tab, Accordion, Mega menu, and Carousel controls, or custom controls designed for your site.

रेंडर हैंडलर सपोर्ट जल्दी और आसानी से टैब जनरेट होते है, एकॉरडीयन, मेगा मेनू और कैराउजल कण्ट्रोल या कस्टम कन्ट्रोल डिजाइन आपकी वेबसाइट के लिए आदि बनाना बहुत आसान हो जाता है।

This type of HTML elements requires pair tags, means starting tag and ending tag. For example:

 ...

इस प्रकार के एचटीएमएल एलिमेन्टो के लिए पेयर टैग अर्थात टैग शुरू करने और टैग समाप्त करने की आवश्यकता होती है। उदाहरण के लिए– ...

2. **Empty element:** HTML elements with no content are called empty elements.
 is an empty element without a closing tag (the <br> tag defines a line break). Empty elements can be "closed" in the opening tag like this: </br >. HTML5 does not require empty elements to be closed. But if you want stricter validation, or you need to make your document readable by XML parsers, you should close all HTML elements.

एम्प्टी एलिमेंट : ऐसे एलिमेंट जिसमें कोई डेटा नहीं होते हैं उन्हें एम्प्टी एलिमेंट कहते हैं। **
** एलिमेंट एक एम्प्टी एलिमेंट हैं, जिसमें इस टैग को बंद नहीं किया जाता है। HTML5 में किसी भी एम्प्टी एलिमेंट को बंद टैग की आवश्यकता नहीं होती है। परन्तु यदि आप सख्त सत्यापन करना चाहते हैं, या अपने डॉक्यूमेंट को XML पर्सेर्स द्वारा पठनीय बनाने हेतु सभी HTML टैग को बंद करना चाहिए।

3.3 Head Section & Elements of Head Section
हेड सेक्शन और हेड सेक्शन के एलिमेन्ट

The head section is the container of a set of elements that provide metadata for the document. This metadata can specify the title (title element), character set, author, description or keywords of the web page (meta element), a style sheet or relational information (link element) or embedded style declarations (style element), among others. The presence of the title element is mandatory inside the head section of an HTML document. Most metadata are not displayed in the browser (although the title usually appears in the browser's title bar) but it can be useful for the functionality of the page.

हेड सेक्शन उन एलिमेन्टो के समूह का एक कंटेनर है जो डॉक्यूमेन्ट के लिए मेटाडेटा प्रदान करते हैं। यह मेटाडेटा टाईटिल (टाईटिल एलिमेन्ट), कैरेक्टर सेट, ऑथर, डिस्क्रिप्शन या वेब पेज (मेटा एलिमेन्ट) के कीवर्ड, एक स्टाईल शीट या रिलेशनल इनफार्मेशन (लिंक एलिमेन्ट) या एम्बेडेड स्टाईल डेक्लेरेशन (स्टाईल एलिमेन्ट), आदि को स्पेशिफाई कर सकता है। किसी HTML डाक्यूमेन्ट के हेड सेक्शन के अंदर टाईटिल एलिमेन्ट की उपस्थिति अनिवार्य है। अधिकांश मेटाडेटा ब्राउजर में प्रदर्शित नहीं होते हैं (हालांकि टाईटिल आमतौर पर ब्राउजर के टाईटिल बार में दिखाई देता है) लेकिन यह पेज की कार्यक्षमता के लिए उपयोगी हो सकता है।

Note: TITLE element is always required. In fact, the minimal HTML 3.0 document consists of the TITLE element alone. The HEAD element has no attributes.

नोटः TITLE एलिमेन्ट की हमेशा आवश्यकता होती है। वास्तव में, न्यूनतम HTML 3.0 में अकेले TITLE एलिमेन्ट होते हैं। HEAD एलिमेन्ट में कोई अलग से ऐट्रीब्यूट नहीं होते है।

Example:

Syntax: The <head> tag is written as <head></head> with the metadata content enclosed between the start and end tags. The <head> tag is placed between the opening and closing <html> tags. Most HTML pages must have a <title> tag within the <head> tag.

सिंटैक्स: <head> टैग को <head>.....</ head> के रूप में लिखा जाता है, जिसमें मेटाडाटा कन्टेट स्टार्ट और एंड टैग के बीच संलग्न होता है। <Head> टैग को <html> टैग के ओपनिंग और क्लोजिंग करने के बीच में रखा गया है अधिकांश HTML पेजों में <head> टैग के भीतर <title> टैग होता है।

```
<!doctype html>
<html>
<head>
        <title>Page title...</title>
</head>
<body>
        Page body...
</body>
</html>
```

Within the HEAD element, only certain elements are allowed. Information in the HEAD element may include the following elements (arranged alphabetically):

HEAD के भीतर, केवल कुछ एलिमेन्टो को लिखने की अनुमति है। HEAD एलिमेन्ट की जानकारी में निम्नलिखित एलिमेन्ट शामिल हो सकते हैं (अल्फाबेटिकल आर्डर में व्यवस्थित):

1. **BASE element :** The BASE element allows the URL of the document itself to be recorded in situations in which the document may be read out of context. URLs within the page may be in a "partial" form relative to this base address. The default base address is the URL used to retrieve the web page.

   बेस एलिमेन्ट : बेस एलिमेन्ट डाक्यूमेन्ट के URL को उन स्थितियों में रिकार्ड करने की सुविधा देता है जिनमें डॉक्यूमेन्ट कंटेक्स्ट से बाहर पढ़ा जा सकता है। पेज के भीतर के URL इस बेस एड्रेस के सापेक्ष "आंशिक" रूप में हो सकते हैं। डिफॉल्ट बेस एड्रेस URL का उपयोग वेब पेज को पुनः प्राप्त करने के लिए किया जाता है।

   For example: उदाहरण के लिए–

   ```
   <base href="http://google.com/docs/mydoc.html">
   ...
   <img src="images/about.gif">
   ```
   Which resolves to "http://google.com/docs/images/about.gif".

2. **ISINDEX:** The ISINDEX element informs the HTML user agent that the document is an index document. As well as reading it, the reader may use a keyword search. The document can be queried with a keyword search by adding a question mark to the end of the document address, followed by a list of keywords separated by plus signs.

   ISINDEX: ISINDEX एलिमेन्ट HTML उपयोगकर्ता एजेंट को सूचित करता है कि डॉक्यूमेन्ट एक इनडेक्स डॉक्यूमेन्ट है। इसे पढ़ने के साथ–साथ पाठक एक कीवर्ड सर्च का उपयोग कर सकते हैं। डॉक्यूमेन्ट के पते के अंत में एक प्रश्न चिह्न जोड़कर डॉक्यूमेन्ट को खोजा जा सकता है, इसके बाद प्लस चिह्नों द्वारा अलग किए गए खोज शब्दों की एक सूची दी जाएगी।

   Note: A server normally generates the ISINDEX tag automatically. If added by hand to an HTML page, the browser assumes that the server can handle a search on the web page. Obviously the

server must have this capability for it to work: simply adding <ISINDEX> in the web page is not enough to make searches happen if the server does not have a search engine!

नोट– एक सर्वर सामान्य रूप से ISINDEX टैग स्वचालित रूप से उत्पन्न करता है। यदि HTML पेज पर हाथ से जोड़ा जाता है, तो ब्राउजर मानता है कि सर्वर वेब पेज पर सर्च को संभाल सकता है। जाहिर है कि सर्वर के पास काम करने के लिए यह क्षमता होनी चाहिए: बस वेब पेज में <ISINDEX> जोड़ने के लिए सर्च करने के लिए पर्याप्त नहीं है अगर सर्वर में सर्च इंजन नहीं है।

Example : <ISINDEX>

The URL used for processing queries can be overridden with the HREF attribute. You can also use the PROMPT attribute to change the default prompt supplied by the browser, e.g.

प्रोसेसिंग क्वेरीज के लिए उपयोग किए जाने वाले URL को HREF विशेषता के साथ ओवररिडेन किया जा सकता है। आप ब्राउजर द्वारा दिए गए डिफॉल्ट प्रॉम्प्ट को बदलने के लिए PROMPT ऐट्रीब्यूट का भी उपयोग कर सकते हैं, जैसे

```
<ISINDEX HREF="Personal.db" PROMPT="Enter Surname:">
```

LINK: The LINK element indicates a relationship between the web page and some other object. A web page may have any number of LINK elements. The LINK element is empty (does not have a closing tag), but takes the same attributes as the anchor element.

लिंक: लिंक एलिमेन्ट वेब पेज और कुछ अन्य ऑब्जेक्ट के बीच एक संबंध को इंगित करता है। वेब पेज में किसी भी प्रकार के लिंक एलिमेन्ट हो सकते हैं। लिंक एलिमेन्ट एम्प्टी है (इसमें क्लोजिंग टैग नहीं है), लेकिन एंकर एलिमेन्ट के समान ऐट्रीब्यूट रखता है।

3.3.1 The important attributes are: इसके महत्वपूर्ण ऐट्रीब्यूट हैं:

- **REL:** This defines the relationship defined by the link. -
  यह लिंक द्वारा परिभाषित संबंध को परिभाषित करता है।

- **REV:** This defines a reverse relationship. A link from document A to document B with REV=relation expresses the same relationship as a link from B to A with REL=relation. REV=made is sometimes used to identify the document author, either the author's email address with a mail to URI, or a link to the author's home page.

  यह एक रिवर्स रिलेशनशिप को परिभाषित करता है। डॉक्यूमेन्ट ए से डॉक्यूमेन्ट बी से REV=relation का एक लिंक उसी संबंध को व्यक्त करता है जो बी से ए के बीच REV=relation से लिंक के रूप में है। REV= made डॉक्यूमेंट कभी–कभी लेखक की पहचान करने के लिए उपयोग किया जाता है, या तो लेखक का ईमेल पता एक मेल यूआरआई के साथ, या लेखक के होम पेज के लिए एक लिंक।

- **HREF:** This names an object using the URI notation.
  यह URI नोटेशन का उपयोग करके किसी आब्जेक्ट का नाम देता है।

Using LINK element to define document specific toolbars: An important use of the LINK element is to define a toolbar of navigation buttons or an equivalent mechanism such as menu items.

डॉक्यूमेंट स्पेसिफिक टूलबार को परिभाषित करने के लिए लिंक एलिमेन्ट का उपयोग करना: लिंक एलिमेन्ट का एक महत्वपूर्ण उपयोग नेविगेशन बटन के एक टूलबार या मेनू आइटम जैसे समकक्ष मैकेनिज्म को परिभाषित करना है।

LINK relationship values reserved for toolbars are: टूलबार के लिए लिंक रिलेशनशिप वैल्यू में रिजर्व हैं:

REL=Home	**The link references a home page or the top of some hierarchy.**	**होम लिंक एक होम पेज या कुछ पदानुक्रम के शीर्ष का संदर्भ देता है।**
REL=ToC	The link references a document serving as a table of contents.	टीओसी लिंक टेबिल ऑफ कन्टेन्ट्स के रूप में एक डॉक्यूमेन्ट को संदर्भित करता है।
REL=Index	The link references a document providing an index for the current document.	इंडेक्स लिंक एक डॉक्यूमेन्ट को वर्तमान डॉक्यूमेन्ट के लिए एक इंडेक्स प्रदान करता है।
REL= Glossary	The link references a document providing a glossary of terms that pertain to the current document.	शब्दावली एक डॉक्यूमेन्ट है जो मौजूदा डॉक्यूमेन्ट से संबंधित शर्तों की शब्दावली प्रदान करता है।
REL= Copyright	The link references a copyright statement for the current document.	कॉपीराइट लिंक वर्तमान डॉक्यूमेन्ट के लिए कॉपीराइट कथन का संदर्भ देता है।
REL=Up	When the document forms part of a hierarchy, this link references the immediate parent of the current document.	जब डॉक्यूमेन्ट एक पदानुक्रम का हिस्सा बनता है, तो यह लिंक वर्तमान डॉक्यूमेन्ट के immediate parent को संदर्भित करता है।
REL=Next	The link references the next document to visit in a guided tour.	अगला लिंक एक निर्देशित दौरे में जाने के लिए अगले डॉक्यूमेंट का संदर्भ देता है।
REL= Previous	The link references the previous document in a guided tour.	पिछला लिंक एक निर्देशित दौरे में पिछले डॉक्यूमेन्ट का संदर्भ देता है।
REL=Help	The link references a document offering help	सहायता लिंक एक डॉक्यूमेन्ट की पेशकश करने में मदद करता है।
REL= Bookmark	Bookmarks are used to provide direct links to key entry points into an extended document.	बुकमार्क का उपयोग एक विस्तृत डॉक्यूमेन्ट में मुख्य प्रविष्टि बिंदुओं के लिए सीधे लिंक प्रदान करने के लिए किया जाता है।

An example of toolbar LINK elements: टूलबार लिंक एलिमेन्टों का एक उदाहरण:

```
<LINK REL="Previous" HREF="mainDoc1.html">
<LINK REL="Next" HREF=mainDoc2.html>
<LINK REL="Bookmark" TITLE="Order Form" HREF="mainDoc3.html">
```

3.3.2 Using LINK to include a Document Banner
डॉक्यूमेन्ट बैनर को शामिल करने के लिए लिंक का उपयोग करना

The LINK element can be used with REL=Banner to reference another document to be used as banner for this document. This is typically used for corporate logos, navigation aids, and other information which shouldn't be scrolled with the rest of the document.

इस डॉक्यूमेन्ट के बैनर के रूप में उपयोग किए जाने वाले किसी अन्य डॉक्यूमेन्ट को संदर्भित करने के लिए लिंक एलिमेन्ट का उपयोग REL=Banner के साथ किया जा सकता है। यह आमतौर पर कारपोरेट लोगोंस, नेविगेशन एड्स और अन्य जानकारी के लिए उपयोग किया जाता है, जिसे बाकी डॉक्यूमेन्ट के साथ स्क्रॉल नहीं किया जाना चाहिए।

For example: <LINK REL=Banner HREF=banner.html>

The use of a LINK element in this way, allows a banner to be shared between several documents, with the benefit of being able to separately cache the banner. Rather than using a linked banner, you can also include the banner in the document itself, using the BANNER element.

इस तरह से एक लिंक एलिमेन्ट का उपयोग, बैनर को कई डॉक्यूमेन्टों के बीच साझा करने की अनुमति देता है, जिसके लाभ के साथ बैनर को अलग से कैश करने में सक्षम होता है। लिंक किए गए बैनर का उपयोग करने के बजाय, आप बैनर एलिमेन्ट का उपयोग करते हुए डॉक्यूमेन्ट में ही शामिल कर सकते हैं।

Link to an associated Style Sheet: The LINK element can be used with **REL=StyleSheet** to reference a style sheet to be used to control the way the current document is rendered.

एक संबद्ध स्टाइल शीट से लिंक: लिंक एलिमेन्ट का उपयोग **REL=StyleSheet** के साथ किया जा सकता है ताकि मौजूदा डॉक्यूमेन्ट को प्रस्तुत करने के तरीके को नियंत्रित करने के लिए style sheet का उपयोग किया जा सके।

For example: <LINK REL=StyleSheet HREF=housestyle.dsssl>

Other uses of the LINK element: Additional relationship names have been proposed, but do not form part of this specification. Servers may also allow links to be added by those who do not have the right to alter the body of a document.

लिंक एलिमेन्ट के अन्य उपयोग: अतिरिक्त संबंध नाम प्रस्तावित किए गए हैं, लेकिन इस एस्पेसिफीकेशन का हिस्सा नहीं हैं। सर्वर उन लिंक को भी जोड़ने की अनुमति दे सकते हैं जिनके पास डॉक्यूमेन्ट की संरचना को बदलने का अधिकार नहीं है।

4. **META:** The META element is used within the HEAD element to embed webpage meta-information not defined by other HTML elements. Such information can be extracted by servers/clients for use in identifying, indexing and cataloging specialized webpage meta-information.Although it is generally preferable to used named elements that have well defined semantics for each type of meta-information, such as title, this element is provided for situations where strict SGML parsing is necessary and the local DTD is not extensible.

   META एलीमेंट का उपयोग अन्य HTML एलिमेन्टो द्वारा परिभाषित वेबपेज मेटा–जानकारी को एम्बेड करने के लिए HEAD एलीमेंट के भीतर किया जाता है। इस तरह की जानकारी सर्वर/क्लाइंट द्वारा विशिष्ट वेबपेज मेटा–सूचना की पहचान, इंडेक्सिंग और कैटालागिंग में उपयोग के लिए निकाली जा सकती है। यद्यपि यह आमतौर पर उपयोग किए गए नाम एलिमेन्टो के लिए बेहतर है, जिन्होंने प्रत्येक प्रकार की मेटा–जानकारी के लिए अच्छी तरह से शब्दार्थ को परिभाषित किया है, जैसे शीर्षक, यह एलिमेन्ट उन परिस्थितियों के लिए प्रदान किया जाता है जहां सख्त एसजीएमएल पार्सिंग आवश्यक है और लोकल डीटीडी एक्सटेंसिबल नहीं है।

In addition, HTTP servers can read the contents of the web page head to generate response headers corresponding to any elements defining a value for the attribute HTTP-EQUIV. This provides document authors with a mechanism for identifying data that should be included in the response headers of an HTTP request.

इसके अलावा, HTTP सर्वर वेब पेज हेड की सामग्री को HTTP–EQUIV के लिए वैल्यू को परिभाषित करने वाले किसी भी एलिमेन्ट के अनुरूप प्रतिक्रिया हेडर उत्पन्न करने के लिए पढ़ सकते हैं। यह डेटा की पहचान करने के लिए एक मैकेनिज्म के साथ डॉक्यूमेन्ट ऑथर प्रदान करता है जिसे एक HTTP अनुरोध के प्रतिक्रिया हेडर में शामिल किया जाना चाहिए।

The META element has three attributes: मेटा एलिमेन्ट की तीन विशेषताऐं हैं–

i. **NAME:** Used to name a property such as author, publications date etc. If absent, the name can be assumed to be the same as the value of HTTP-EQUIV. किसी संपत्ति का नाम जैसे लेखक, प्रकाशन दिनांक आदि के लिए उपयोग किया जाता है। यदि अनुपस्थित है, तो नाम HTTP-EQUIV के वैल्यू के समान माना जा सकता है।

ii. **CONTENT:** Used to supply a value for a named property. एक नामित संपत्ति के मूल्य के लिए प्रयोग किया जाता है।