

## Generally Every website has 3 ends,

1. Frontend part
2. Backend part
3. Database part

### 1. Frontend part:

- A person who knows Frontend work very well and who can do any part of frontend work is called Frontend Developer.
- Every Frontend Developer must know all UI Technologies.
- A good Frontend developer can easily solve user problems and he/she must be responsible for frontend designing.

### 2. Backend part:

- A person who knows Backend work very well and who can do any part of backend coding is called Backend Developer.
- Every Backend Developer must know any one or two backend technologies like Python or Java or .Net or PHP.....
- A good Backend Developer can easily write the logics which are required to handle the server or database... and he/she must be responsible for any kind of backend work.

### 3. Database part:

- A person who knows database logics and work on database side then call database developer.
- DB developers required to learn atleast any one database like MySQL, Mongo DB,...

### Full Stack Web Developer:

- A person who can do both frontend and backend and database works very well is called Full Stack Web Developer.
- Stack means layer, so full stack means all layers.
- A person who knows all layers of website is called Full Stack Web Developer.
- Full Stack Web Developer must know about database also along with frontend and backend technologies.

### About UI Technologies:

The Frontend Technologies are nothing but UI Technologies.

- UI stands for User Interface.
- An interface is a set of items or menus through which a user communicates with a Server.
- UI is a junction between a user and a computer program.
- UI Technologies are
  1. HTML
  2. CSS
  3. Bootstrap
  4. Javascript
  5. jQuery
  6. DOM

## Uses of UI Technologies,

1. HTML ( Hyper Text Markup Language )
  - Our web page related content we can design (Static Response)
2. CSS ( Cascade Style Sheet )
  - We can apply styles to our web page content
3. Bootstrap
4. JavaScript
  - Dynamic Response we can add to our web page
5. JQuery -
  - Dynamic Response we can add to our web page more effectively
6. DOM ( Document Object Model )

## Why Learn HTML?

- HTML is the foundation of all web pages. Without HTML, you wouldn't be able to organize text or add images or videos to your web pages.
- HTML is the beginning of everything you need to know to create our web pages!

## HTML ( Hyper Text Markup Language )

- It stands for Hyper Text Markup Language.
- HTML contains collection of tags or elements.
- HTML tags will decide the structure of Webpage.
- Whatever we want to display on the browser or webpage, the corresponding content should be written between the tags in HTML file.
- At runtime, It will ignore the tags and only content will go to browser from the html file and displays on the browser.
- All HTML tags are Predefined tags only.  
**For example:** <h1>....</h1>, <p>...</p>
- HTML tags are not Userdefined tags  
**For example:** <emp>....</emp>, <student> ...</student>
- HTML Tags are not case sensitive tags,  
**For example:** <h1> and <H1> is same.
- Every HTML tag has its own behaviour and this behavior applies to Content when html executing.

## Basic Syntax of HTML file:

<html>

<head>

---->> Meta Description about the project (titles, metadata, dates..)

</head>

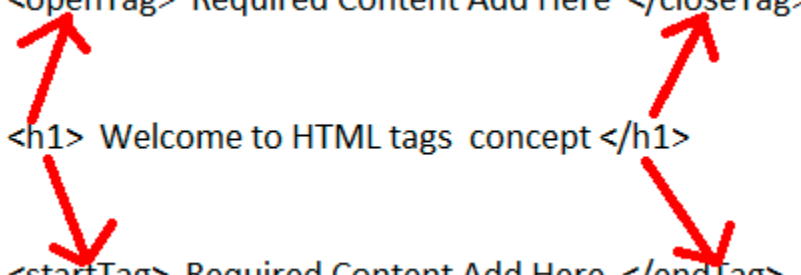
<body>

--->> Main content to display on the browser

</body>

</html>

## HTML tag structure / Syntax:

`<openTag> Required Content Add Here </closeTag>`  
  
`<startTag> Required Content Add Here </endTag>`

- Whenever we run the program or project then html will ignore all the tags and it will load only the content to the browser.

## Different Sources to write HTML program

- notepad
- notepad++
- editplus
- pycharm
- sublimetext
- VisualStudioCode (VCS tool)
- eclipse
- eric
- pyscripter, etc

## Basic step to write html program in NOTEPAD

Step1: Open Notepad file

Step2: Write the HTML program in the Notepad file

Step3: Save the Notepad file with extension **.html** like firstprogram.html (It will save in browser format)

Step4: To Run the HTML file, goto saved file location and double click on the saved browser file.

(or) right click on saved html file name and click on "open with" and click on required browser to open with html file output in a required browser.

Step6: Note: If we want to see the saved html file code then right click on browser file and click on open with "notepad" or any other required Editor|IDE.

## demo.html

```
<!DOCTYPE html>    --->> optional
<html>
  <head>
```

```

    <title> UI Technologies </title>
</head>

<body>
    <h1> Welcome to UI </h1>
    <p> Welcome to Paragraph.</p>
</body>
</html>

```

Note:

- The content inside the <body> section will be displayed in a browser.
- The content inside the <title> element will be shown in the browser's title bar or in the page's tab.

### Example Explained:

- The <!DOCTYPE html> declaration defines that this document is an HTML5 document
- The <html> element is the root element of an HTML page
- The <head> element contains meta information about the HTML page
- The <title> element specifies a title for the HTML page (which is shown in the browser's title bar or in the page's tab)
- The <body> element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.
- The <h1> element defines a large heading
- The <p> element defines a paragraph

### Web Browsers:

The purpose of a web browser (Chrome, Edge, Firefox, Safari) is to read HTML documents and display them correctly.

A browser does not display the HTML tags, but uses them to determine how to display the document:

### The HTML <head> Element

- The <head> element is a container for metadata. HTML metadata is data about the HTML document. Metadata is not displayed.
- The <head> element is placed between the <html> tag and the <body> tag.
- Note: Metadata typically define the document title, character set, styles, links, scripts, and other ---- meta information.

### Heading tags

=====

- Headings are defined with the <h1> to <h6> tags.
- <h1> defines the most important heading. <h6> defines the least important heading.
- Note: Browsers automatically add some white space (a margin) before and after a heading.

## Q) Write a Program to display all the Headers tag content on browser ?

```
<html>
  <head>
    <title> UI Technologies  </title>
  </head>
  <body>
    <h1> Welcome to UI </h1>
    <h2> Welcome to TCS </h2>
    <h3> Welcome to HTML </h3>
    <h4> Welcome to CSS</h4>
    <h5> Welcome to Bootstrap  </h5>
    <h6> Welcome to JavaScript</h6>
  </body>
</html>
```

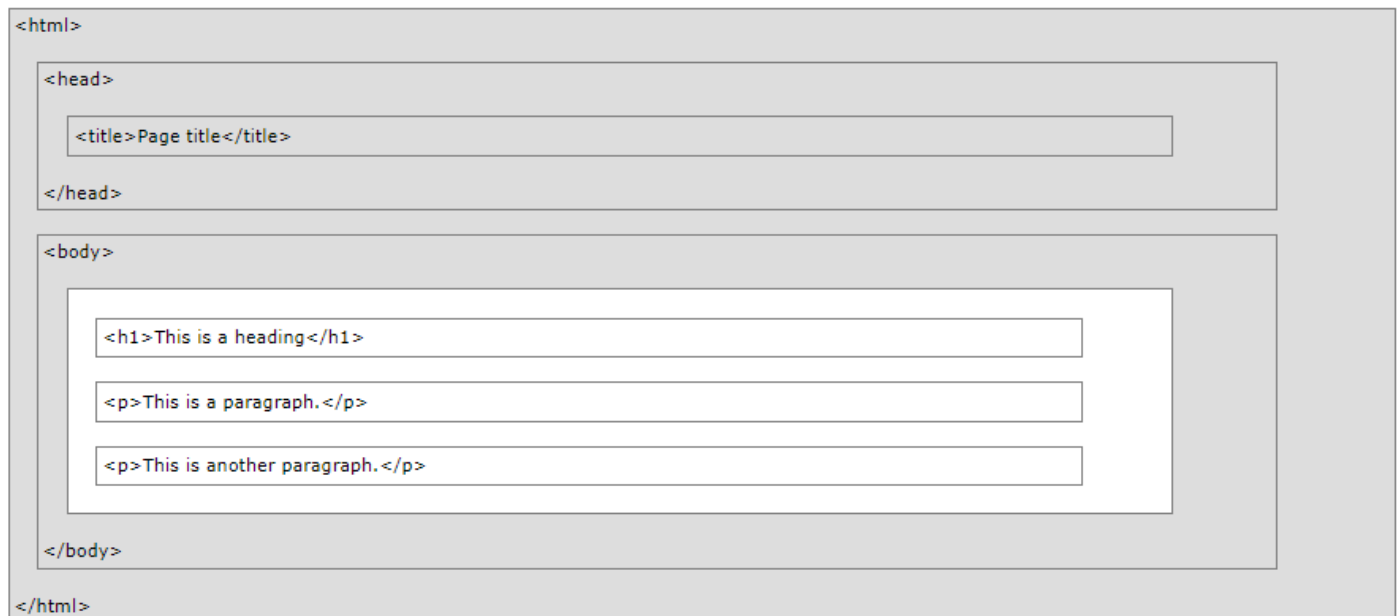
### HTML Paragraphs:

=====

- The HTML <p> element defines a paragraph.
- A paragraph always starts on a new line, and browsers automatically add some white space (a margin) before and after a paragraph
- For example : <p> Welcome to Paragraph.</p>

## HTML Page Structure

Below is a visualization of an HTML page structure:



**Note:** The content inside the <body> section (the white area above) will be displayed in a browser. The content inside the <title> element will be shown in the browser's title bar or in the page's tab.

## HTML contains so many tags, some tags are

Basic html structure tags ----->> <html> , <head> , <body> , <title> ,<meta>

Paragraph tags ----->> <p> , <pre>

List Items tags ----->> <ol> , <ul> , <li>

Table tags ----->> <table> <thead> , <tbody>, <tr> , <th> , <td>

Link and image tags ----->> <a> , <img> ,

Form tags ----->> <form>, <input>, <label> , <button>, <select> ,<option>,  
<datalist>, , <fieldset> , <legend>

Text Format Tags ----->> <b> , <i> , <u> , <ins> , <strong>, <del>, <sup> ,  
<sub>, <center> ,

Header and Footer tags ----->> <header>, <footer>

And so on...