1. What is Web

* Web

→ A small entity (subset) of a bigger entity called Internet.
→ A system of interconnected resources which we can access via internet.

* Internet

→ Interconnected network of networks around the world.

* Development

→ building or creation process

website : static content (read only) (negligible user interaction)

webapp : dynamic content

* Network

→ system of machines which are connected

→ scale : LAN, MAN, WAN, etc.

* Ip Address

→ Unique Identification for a machine which is an address of that machine acrooss internet.

→ From Domain name to Ip address conversion is called Domain Name Resolution (DNR).
→ Ip address for Domain names are stored at Domain Name Server (DNS).
→ Our Ip is provided by ISP (Internet Service Provider)

* Browser

→ A tool (software) to access data resources on web
→ For the use to render (visual representation) something on the window

* Client - Server

→ client sends request and server sends response based on requests.

Requests
get → fetch
post → create
put → update
delete → remove

* Packets

→ Big data breaks into small chunks are called packets

2. HTML Introduction

* What is HTML ?

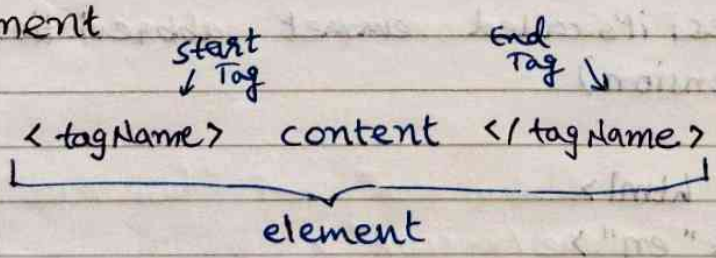
→ HTML stands for Hyper Text Markup Language.
→ HTML is the standard markup language for creating web pages.
→ HTML describes the structures of a web page.

* Example

```
<!DOCTYPE html>
<html>
<head>
<title> Page Title </title>
</head>

<body>
    <h1> First Heading </h1>
    <p> First Paragraph </p>
</body>
</html>
```

* Element



→ <!DOCTYPE html> declaration defines that this document is an HTML5 document.

→ <html> element is the root element of HTML page.

→ <head> element contains meta information about the HTML page

→ <title> element specifies a title for the HTML page (which is shown in the browser's title bar on in the page's tab)

→ <body> element defines the document's body, and is ~~sto~~ a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.

→ Some elements have no content and these empty elements do not have an end tag. (e.g.
 - defines a line break)

* Boilerplate Code.

→ Type ! and press enter, default auto complete HTML code will be filled on text editor.
→ Website : emmet cheat sheet - different auto complete codes, it's called emmet abbreviation (an extension)

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
                                   initial-scale=1.0">
```

```
    <title> website Title (tab Name) </title>
  </head>

  <body>

  </body>
</html>
```

→ here, charset is an attribute of meta tag, and is use for character encoding of the web page.

→ http-equiv attribute is for browser compatibility. It is now needed because IE browser is discontinued.

→ viewport attribute is for device screen view. suppose, for mobile device content width will be device-width and scale (zoom) will be 1.0.

3. Head Tag

* <head> ... </head>

→ head tag contents will be not shown for user interaction, it is for browser and SEO (search Engine Optimization). A container of meta-data.

→ title tag :
<title> content </title>
- content will be shown at tab of the browser.

→ <meta name = "description" content = "write description here" >

- When you search some website and you see after link there are some text below link, which describes about the website, it's comes from here.

→ <meta property = "og: image" content = "image link path here" >
<meta property = "og : title" content = " some website title >
<meta property = "og : description" content = "write description here" >

- When, you share some link, suppose on whatsapp, an image (or link) automatic unfurls and you can see a box; in that box you can see an image, a title and description, it's comes from og (open graph protocol)

→ <link rel = "~~icon~~ shortcut icon" href = "favicon.ico"
type = " image/x-icon" >

- On tab of browser, along with title you can also see an icon.
- here, icon file downloaded from favicon generator website and renamed to favicon.ico and put in with .html file. (or can be a path of icon).

4. Headings

→ <h1> heading content </h1>

- There are 6 (six) type of headings. <h1>, <h2> <h3>, <h4>, <h5>, <h6>.

- It is recommended to use h1 heading only one time in one web page.

- Font size decreases from h1 to h6 and all content will be bold type.

5. Paragraph

→ content

- content will be displayed on the page, but it will disregard extra spaces and line breaks within contents

- You want content to be in new line then write another p tag or type
 tag within content

→ <pre> Text in this pre element is displayed in a fixed-width font, and it preserves both spaces and line breaks
</pre>

→ write lorem10 and press enter, You will have 10 random words from lorem ipsum website. It's an emmet.

<hr> horizontal line (divider)

 a new line
 ... bold text
 ... bold emphasized text
 ... italic text
 ... italic emphasized text
<mark> .. </mark> highlighted text
<del> ... </del> strikethrough text
_{...} subscript text
^{...} superscript text

6. Lists

* Unordered List : <ul> ... ...

```
<ul>
  <li> item1 </li>
  <li> item2 </li>
  <li> item3 </li>
</ul>
```

Output :
- item1
- item2
- item3

* Ordered List : <ol> ... ...

```
<ol>
  <li> item1 </li>
  <li> item2 </li>
  <li> item3 </li>
</ol>
```

Output :
1. item1
2. item2
3. item3

→ You can also use nested list.
e.g. another three items under item1, also set item1 inside heading.

→ In Ordered List, you can change type of series.
default type = 1, 2, 3, ...

```
<ol type = "A" >
<ol type = "a" >
<ol type = "i" >
```

Output : A, B, C, ...
a, b, C, ...
i, ii, iii, ...

7. Attributes, Anchor

* Attributes attributes name

→ Attributes are used to add more information to the tag.

* Anchor <a> ...

→ Used to add links to your page

 Text for Clicking
↑ hypertext reference

e.g. Google
 Netflix
Output: Google Netflix

→ Absolute Link - url from Internet
→ Relative Link - path from File Folder (for multiple page link)

→ Instead of content (text for clicking), you can also write <img> tag, then it will become clickable pic.

→ href = " mailto : myemail@example.com " ⇒ for mail

→ href = " tel : +919876543210 " ⇒ for telephone

→ href = " #myID " ⇒ for navigate to "myID" element in same page
→ href = " #top " ⇒ for back to top ("#top" is default)

8. Image

* < img >

→ Used to add image to your page

<img src="image.png" alt="Image Name Text" >

→ We have here given only name of image because this is relative url and image is in same folder as html file.

→ As source, we can also use absolute link (url from internet)

→ alt attribute used, when image did not display for some reason then this text will be displayed.

→ height = "100px" width = "100px" these attributes can be used to resize images (but do it in css).

9. Br, Bold, Italic, Underline, Comments

* Breaking Line

→ Used to add next line (line breaks) to your page.

* Bold Text ...

→ write text here

* Italic Text ...

→ write text here

* Underline Text <u> ... </u>

→ <u> write text here </u>

* Comments

→ This is part of code that should not be parsed.

<!-- write comments here -->

* Inline Elements

→ Takes up only necessary width
→ Doesn't start from new line
→ Anchor, Image, ...

* Block Elements

→ Takes up the full-width available (whole block)
→ starts from new line
→ Heading, paragraph, ...

e.g. <h1> heading text1 </h1>
<h1> heading text2 </h1>

Output:
heading text1
heading text2

11. Div and Span Element

* Div (Content Division) Element ...

→ Div is a container used to hold other HTML elements or group elements together

→ Div is Block element

e.g.
```
<div>
    <a href="www.google.com">Google</a>
    <a href="www.youtube.com">Youtube</a>
</div>
    <a href="www.facebook.com">Facebook</a>
```

Output: Google Youtube
Facebook

* Span Element ...

→ Span is also a generic container used to hold other HTML elements or group elements together.

→ Span is Inline element.

e.g. In above example replace div with span.
Output: Google Youtube Facebook

12. Semantic Markup

→ It is the markup that relates to the meaning of content (meaning is already in its name).

→ Benefits of using semantic markups:
- meaningful code and structured layout
- SEO (Search Engine Optimization) friendly
- readable and can use screen reader (better UX - User experience)

→ Semantic Tags:

header - Top content
footer - Bottom content
main - Main content
nav - navigation (different pages navigation (links) (mostly we find at in the header tag)
section - group of related content
aside - related links (indirect content - not the main conte
article -

13. Entities

→ An HTML entity is a piece of text("string") that begins with an ampersand (&) and ends with a semicolon (;).

→ Used to display reserved characters (which would otherwise be interpreted as HTML code) and invisible characters (like non-breaking spaces).

→ Can also use in place of characters that are difficult to type with a standard keyboard.

→ Browser interprets them and renders correct character

e.g.   = non-breaking space
< = <
> = >
♥ - ♥ - ♥
★ - ★ - ★
& - &
" - "
(MDN has list of entities - for decimal convert unicode to decimal)
&#8377 - ₹

14. Emmet

→ A toolkit for web-developers (shortcuts for codes)

→ visit emmet.io → cheat sheet (left-bottom in nav)

e.g. parent-child ⇒ nav > ul > li ⇒ <nav>

...

</nav>

Siblings ⇒ p > div + img ⇒
...
<img>

multiplication ⇒ ul > li *3 ⇒
...
...
...

15. Table

* Table <table> ... </table>

→ <tr> ... </tr> - To display table row
→ <td> ... </td> - To display table data
→ <th> ... </th> - To display table header (which has default formatting of bold and center)

→ e.g.

```
<table> border = "black"  ← not recommended, do it in css
  <caption> Table Caption (Title) </caption>
    <tr>
      <th> Header Text 1 </th>
      <th> Header Text 2 </th>
    </tr>

    <tr>
      <td> data 1 </td>
      <td> data 2 </td>
    </tr>

    <tr>
      <td> data 3 </td>
      <td> data 4 </td>
    </tr>
</table>
```

→ Output:

Table Caption (Title)	
Header Text 1	Header Text 2
data 1	data 2
data 3	data 4

← no border for caption and no bold, only center
← bold + center

* Semantics in Table

→ <thead> ... </thead> - To wrap table header
→ <tbody> ... </tbody> - To wrap table body
→ <tfoot> ... </tfoot> - To wrap table footer

→ e.g.
```
<thead>
    <tr>
        <th> Heading 1 </th>
        <th> Heading 2 </th>
    </tr>
</thead>
```

* Colspan & Rowspan Attributes

→ Used to create cells which spans over multiple rows & columns. (default cell has 1 rowspan & 1 colspan.)

→ e.g.
```
<th rowspan = "2" > Item </th>
<th colspan = "2" > Price </th>
<th> INR </th>
<th> USD </th>
```

Output:

Item	Price	
	INR	USD

16. Form

* Form <form> ... </form>

→ Form are used to collect data from the user

➔ Action Attribute

→ Used to define what action needs to be performed when a form is submitted or where the form data should be sent

eg. <form action = "/action.php"> ... </form>
<form action = "https://youtube.com/"> ... </form>

* Input (Form Element) <input>

→ Used to create multiple form controls.
→ There are multiple types of inputs that can be created using type attribute.

<input type = "text">

→ There are various types : text, password, email, number, decimal, time, color, etc.

➔ Placeholder Attribute (Input Element)

→ hint text (in grey color)
→ <input type = "text" placeholder = "enter text here">

output: enter text here

* Label <label> ... </label>

→ Label element represents a caption for an item in a user interface.

```
<label>
    Enter Your Name:
    <input type="text" placeholder="Enter Name here">
</label>
```

Output: Enter Your Name : (Enter Name here)

→ classical approach:
this way, when mouse hover over label it will remain arrow and when click on label, cursor will be shown in input box.

```
<label for="nameID">Enter Your Name:</label>
<input type="text" placeholder="Enter Name Here"
    id="nameID">
```

* Button <button> ... </button>

→ If button inside "form", then on click event of "submit" button will be performed as per "action" attribute of "form" element.

```
<button type="submit">Submit</button>
<button type="reset">Reset</button>
<button type="button">text inside button</button>
```

no need to write this if a simple button of html

→ button can also be in input element as "type" attribute will be "submit" but this is not recommended.

```
<input type = "submit" >
<input type = "submit" value = "click me" >
```

(submit) (click me)

⇒ Name Attribute (Input Element)

→ Name of the form control, submitted with the form as part of a name/value pair

→ meaning, if you have given "name" attribute as "username" in "input" element then whatever text inside of that "input" will become "username = sometext" (name/value pair) and will be shown in url after click on "submit" button and "action" attribute performes.

→ e.g.
```
<form action = "https://youtube.com/results" >
    <label for = "searchTextID" > Search on YouTube: </label>
    <input type = "text"
        placeholder = "Enter text here"
        id = "searchTextID"
        name = "search_query" >
    <br> <br>
    <button type = "submit" > Submit </button>
</form>
```

youtube search url has this keyword, so we have to do same then full url will become same as youtube search result

Output: Search On Youtube : [Enter text here] ← here, type apple then URL will be

(Submit). → Youtube.com/results?search_query=apple

* Checkbox (Input Element)(type value)

```
<input type = "checkbox" name = "age" id = "ageID" >
<label for = "ageID" > I am 18+ </label>
```

Output: ☐ I am 18+

→ After submit, passed data will have name/value pair as ⇒ age=on

→ If you want already checked box then write "checked" in "input" element <input type="checkbox" checked>

→ Now, If we want group of checkboxes, then data will not be passed as "on" but we will give specific "value" as attribute value and all "input" will have same "name".

```
<label for = "subject" > subject selection : </label>
<input type = "checkbox"
       name = "subject"
       id = "physicsID"
       value = "physics" >
<label for = "physicsID" > Physics </label>

<input type = "checkbox"
       name = "subject"
       id = "MathsID"
       value = "maths" >
<label for = "mathsID" > Maths </label>
```

Output: Subject Selection : ☐ Physics ☐ Maths

* Radio (Input Element) (type value)

→ Difference between checkbox and radio is that checkbox can pass multiple value but radio can only pass one value.

```
<label for="gender"> Gender: </label>

<input type="radio"
       name="gender"
       id="maleID"
       value="male" >
<label for="maleID"> Male </label>

<input type="radio"
       name="gender"
       id="femaleID"
       value="female
<label for="femaleID"> Female </label>
```

Output: Gender : O Male O Female

* Select: <select> ... </select>

→ a dropdown selection (menu)

```
<label for="countryID"> Country : </label>

<select name="country" id="countryID">
  <option> -- select an option -- </option>
  <option value="india"> India </option>
  <option value="canada"> Canada </option>
</select>
```

→ If you want to show selected default then write "selected" in open tag of "option"

* Range (Input Element) (type value)

→ a progress bar

```
<label for="volumeID"> Volume </label>
<input type="range" min="0" max="100" name="volume"
       id="volumeID">
```

→ Output: Volume 

→ step = "10" - for increasing range on one swipe
→ value = "70" - for value (initial) when screen first opens or "reset" is clicked.

* TextArea <textarea>...</textarea>

<label for="feedbackID">Provide Your Feedback:</label>
<textarea id="feedbackID"></textarea>

→ Default: textarea will be of 2 rows & 10 columns

→ rows="5" - for more rows
→ cols="20" - for more columns
→ placeholder="enter text here" - for hint text

Output: Provide Your Feedback: