MEAN STACK DEVELOPMENT

UNIT - 01

HTML 5: Introduction to Web, Overview of Web Technologies, HTML - Introduction, HTML - Need, Case-insensitivity, Platform-independency, DOCTYPE Declaration, Types of Elements, HTML Elements - Attributes, Metadata Element, Sectioning Elements, Paragraph Element, Division and Span Elements, List Element, Link Element, Character Entities, HTML5 Global Attributes, Creating Table Elements, Table Elements : Colspan/ Rowspan Attributes, border, cellspacing and cellpadding attributes, Creating Form Elements, Input Elements - Attributes, Color and Date Pickers, Select and Datalist Elements, Editing Elements, Media, Iframe, Why HTML Security, HTML Injection, Clickjacking, HTML5 Attributes & Events Vulnerabilities, Local Storage Vulnerabilities, HTML5 - Cross-browser support, Best Practices For HTML Web Pages.

Introduction to Web:

The web, short for World Wide Web, is a system of interconnected documents & resources that are accessed over the internet. It is a vast network of information that has revolutionized the way we share and access information.

The web is based on a client- server architecture, where a user's web browser acts as the client & communicates with a server over the internet. When a user types in a web address, also known as a **Uniform Resources Locator – (URL),** browser sends a request to the sever, which then sends back the requested web page.

Web pages are typically written in **Hyper Text Markup Language – (HTML)**, which a markup language that allows developers to create structured content with links to other pages, images, and multimedia. Web pages can also include styling and layout information using **Cascading Style Sheets – (CSS)** and interactive behavior using JavaScript.

The web has evolved significantly since it's inception in the 1990's. It has gone from being a collection of static web pages to a dynamic & interactive platform that allows users to create, share & consume content in a variety of forms, including text, images, audio & video. The web has also given rise to various web-based technologies such as E-Commerce, social media & online education, which have fundamentally transformed the way we work, learn and communicate.

Overview of Web Technologies :-

Web technologies refer to the software and tools used to create and maintain websites and web applications. These technologies include programming languages, web development frameworks, content management systems, and web servers.

Here are some of the most common web technologies used today:

HTML: Hyper Text Markup Language (HTML) is the standard markup language used to create web pages.

CSS: Cascading Style Sheets (CSS) is a style sheet language used to describe the presentation of a document written in HTML.

JavaScript: A high-level programming language used to create interactive and dynamic web pages.

PHP: Hypertext Preprocessor (PHP) is a server-side scripting language used for web development to create dynamic web pages.

Python: A high-level programming language used for web development to create scalable web applications.

Ruby on Rails: A web development framework that uses the Ruby programming language to create web applications.

React: A JavaScript library used for building user interfaces.

Angular: A TypeScript-based open-source web application framework used for building dynamic web applications.

WordPress: A **Content Management System (CMS)** used to create blogs, websites, and online stores.

Apache: A popular open-source web server software used to serve web pages.

Nginx: A lightweight and high-performance web server and reverse proxy server.

MySQL: A popular open-source **relational database management system (RDBMS)** used for storing and retrieving data.

MongoDB: A NoSQL database program used for storing and retrieving data in a document-oriented database format.

These web technologies are continually evolving and being updated to provide better performance, security, and user experience.

HTML – Introduction:-

HTML (Hyper Text Markup Language) is a standard markup language used to create web pages and other online content. It provides a structure for web content by using a system of tags and attributes to describe the different elements of a page. It is an **Scripting Language** can be used to create web pages.

At a basic level, HTML consists of a series of tags that are used to define the various elements of a web page, such as headings, paragraphs, links, images, and tables. These tags are enclosed in angle brackets, and may include attributes that specify additional information about the element.

HTML is the foundation of the web and is used in conjunction with other technologies like CSS and JavaScript to create dynamic, interactive web pages. It is a language that is constantly evolving, with new versions being released periodically.

Learning HTML is a fundamental skill for anyone interested in web development, as it provides the foundation for creating web pages and understanding how the web works.

HTML – Need:-

HTML components form the building blocks of all websites. HTML **allows images and objects to be imersed and can be used to create interactive forms**. It provides a means to create structured documents by indicating structural explanation for text such as headings, paragraphs, lists, links, quotes, and other items.

HTML, or Hypertext Markup Language, is the standard language used to create web pages. It provides a way to structure content and create semantic meaning in web documents. Here are some of the things you will need to know to get started with HTML:

Basic syntax: HTML uses tags to structure content. Tags are written using angle brackets and usually come in pairs, with an opening tag and a closing tag. For example, a paragraph is written as <p>...</p>.

Elements: HTML elements are made up of tags and the content they contain. Elements can include headings, paragraphs, images, links, and more.

Attributes: HTML attributes provide additional information about elements, such as the source of an image or the destination of a link.

Document structure: HTML documents have a standard structure, with a head section that contains information about the document and a body section that contains the visible content.

Semantic markup: HTML provides a way to create semantic meaning in web documents, which can improve accessibility and search engine optimization. For example, using the <nav> tag to indicate a navigation menu, or the <article> tag to indicate a standalone piece of content.

Learning HTML is a great way to get started with web development. There are many resources available online, including tutorials, reference guides, and interactive courses.

Case-Insensitivity:-

In HTML, tag and attribute names are case-insensitive. This means that you can use uppercase or lowercase letters when writing HTML code and the browser will treat them the same way.

For example, the following two lines of code are equivalent:

<p>This is a paragraph.</p>

<P>This is a paragraph.</P>

Similarly, attribute values are generally case-insensitive, although there are some exceptions. For example, the 'type' attribute in an input element is case-sensitive, so' type="text"' and' type="TEXT"' are not equivalent.

It's good practice to be consistent in your use of case when writing HTML code, as it can make your code more readable and easier to understand. Many developers choose to use lowercase letters for HTML tags and attributes, as this is the convention used in most examples and tutorials

HTML is not a Case Sensitive Language because, during parsing, all HTML elements are converted to lowercase first. HTML5, the latest version of HTML, which is a new web standard, is also case insensitive. XHTML, an older version of HTML, was case-sensitive for lowercase letters.

All the HTML elements/tags are case-insensitive. Case-insensitive means **the tags/elements which are used in the code are understandable by the browser irrespective of the letters being the upper case or lower case classes**.

HTML is case-insensitive, meaning that tags and attributes can be written in any combination of uppercase and lowercase letters. For example, the <p> tag can be written as <P>, <p>, or even <P>.

Case-Insensitivity in HTML 5 is the ability of the browser to recognize HTML tags, attributes, and attribute values regardless of their capitalization. This means that the browser will treat "p" and "P" as the same tag, and "align" and "ALIGN" as the same attribute. This allows developers to write HTML code in a more flexible way, without having to worry about capitalization.

Platform-independency:-

Platform independency means it can run on any kind of platforms or OS. HTML refers to the ability of HTML to be used and rendered on different operating systems and hardware platforms without the need for modification. HTML is a markup language used to create web pages, and it is designed to be platform-independent.

HTML code can be interpreted by any device that has a web browser installed, regardless of the operating system or hardware platform it is running on. This means that HTML pages can be viewed on different types of devices, such as desktop computers, laptops, tablets, smartphones, and even smart TVs, as long as they have a compatible web browser.

Overall, the platform independency of HTML is an essential feature of the language that has made it possible for web pages to be accessed and viewed on a wide range of devices and platforms, without the need for developers to create separate versions of the content for each platform.

HTML (Hypertext Markup Language) is a markup language used to create web pages and is designed to be platform-independent. This means that HTML code can be interpreted by any device, regardless of the operating system or hardware platform it is running on, as long as there is a web browser installed that can read and display HTML content.

In addition, the separation of content and presentation in HTML helps to ensure that web pages are accessible to users with different types of devices, such as mobile devices with smaller screens or assistive technologies for users with disabilities.

Overall, HTML is a platform-independent technology that has helped to make the World Wide Web accessible to users on a wide range of devices and platforms, and has contributed to the growth and evolution of the internet as we know it today.

DOCTYPE Declaration:-

The doctype declaration in HTML (Hypertext Markup Language) is a declaration that specifies the version of HTML or XHTML that the web page is using. It is placed at the very beginning of an HTML document, before the <html> tag, and informs the browser how to interpret the markup in the page.

The doctype declaration starts with **<!DOCTYPE>** followed by the document type, such as html, html5, xhtml, or xhtml5, and an optional public identifier and system identifier. Here is an example of a doctype declaration for HTML5:

<!DOCTYPE html>

This doctype declaration is used for HTML5 documents, which is the current version of HTML. It tells the browser that the page is an HTML5 document and to use the appropriate parsing rules for that version.

In summary, the doctype declaration is an important part of an HTML document that specifies the version of HTML being used and ensures that the document is displayed correctly in different web browsers.

Types of Elements:-

An HTML element is defined by a start tag, some content, and an end tag.

In HTML, there are several types of elements that can be used to structure and display content on a web page. Here are some of the most common types of HTML elements:

Headings: <h1> to <h6> There are six levels of headings (h1 to h6) that can be used to define the importance of a section or sub-section of content.

Paragraphs: <p> The "p" element is used to define a paragraph of text.

**Lists: <ul> and ** HTML supports ordered lists (ol) and unordered lists (ul) for displaying lists of items. Each item in the list is defined using the "li" element.

Links: <a> The "a" element is used to create links to other pages or resources.

Images: <img> The "img" element is used to display images on a web page.

Tables: <table> The "table" element is used to create tables with rows and columns. Table cells are defined using the "td" and "th" elements.

Forms: <form> The "form" element is used to create forms for user input. Form controls, such as text inputs, radio buttons, and checkboxes, are defined using various input element types.

Divisions: <div> The "div" element is used to group and organize content within a web page.

Span: <span> The "Span" is used to group and apply styles to inline elements.

Head: <head> The "head" element contains metadata about the web page, such as the page title and links to stylesheets and scripts.

Body: <body> The "body" element contains the main content of the web page.

HTML Elements – Attributes:-

In HTML, elements can have attributes that provide additional information about the element or modify its behaviour. In HTML, elements are the building blocks used to structure and display content on a web page. Each HTML element can have attributes, which are used to provide additional information about the element or modify its behaviour. Attributes are added to an element's opening tag and are typically written as name-value pairs, separated by an equal sign.

Here are some common attributes that can be used with HTML elements:

id: Specifies a unique identifier for an element. This attribute provides a unique identifier for an HTML element on a web page. It is often used in conjunction with CSS and JavaScript to manipulate the element.

class: This attribute specifies one or more class names for an HTML element. Classes are used to group elements together and apply the same styles to all elements in the group. Specifies one or more class names for an element, which can be used to apply styles to multiple elements at once.

style: Specifies inline styles for an element, which can be used to apply CSS styles directly to an element. This attribute allows you to specify inline styles for an HTML element. Inline styles override any styles defined in an external stylesheet or internal stylesheet.

src: This attribute specifies the URL of an external resource, such as an image or a script, that is used by the element. Specifies the URL of an external resource, such as an image or script.

href: Specifies the URL of the destination page or resource for a link. This attribute specifies the URL of the destination page or resource for a link.

alt: This attribute specifies alternative text for an image, which is displayed if the image cannot be loaded. Specifies alternative text for an image, which is displayed if the image cannot be loaded.

title: Specifies additional information about an element, which is displayed as a tooltip when the user hovers over the element. This attribute specifies additional information about an element, which is displayed as a tooltip when the user hovers over the element.

target: This attribute specifies where to open the link when it is clicked, such as in a new window or tab. Specifies additional information about an element, which is displayed as a tooltip when the user hovers over the element.

type: Specifies the type of input control for form elements, such as "text" for a text input or "submit" for a submit button. This attribute specifies the type of input control for form elements, such as "text" for a text input or "submit" for a submit button.

disabled: This attribute disables an input control or button so that it cannot be used. Disables an input control or button so that it cannot be used.

- All HTML elements can have **attributes**
- Attributes provide **additional information** about elements
- Attributes are always specified in **the start tag**
- Attributes usually come in name/value pairs like: **name="value"**

These are just a few of the most commonly used HTML attributes. There are many more attributes available that can be used to modify the behaviour or appearance of HTML elements.

Metadata Element:-

Metadata is **data (information) about data**. <meta> tags always go inside the <head> element, and are typically used to specify character set, page description, keywords, author of the document, and viewport settings. Metadata will not be displayed on the page, but is machine parsable.

456

Some common metadata elements in HTML include:

<title>: This element is used to define the title of the document, which appears in the browser's title bar and is often used by search engines to help identify the content of the page.

<meta charset="">: This element specifies the character encoding used by the document, which is important for displaying non-ASCII characters correctly.

<meta name="description" content="">: This element provides a brief description of the document, which can be used by search engines to help identify the content of the page.

<meta name="keywords" content="">: This element provides a comma-separated list of keywords that are relevant to the document, which can also be used by search engines to help identify the content of the page.

Metadata elements are included in the **<head>** section of an HTML document, and do not appear on the page itself.

Sectioning Elements:-

What is a section in HTML? In HTML, a section is **a semantic element for creating standalone sections in a web page**. These sections should be made up of related content, like contact information. The section element should only be used if there isn't a more specific element to represent the related content.

In HTML, sectioning elements are used to define the structure and outline of a document. They are used to create distinct sections of content, and help to provide context and hierarchy to the information on a page. Some common sectioning elements in HTML include:

<header>: This element is used to define the header of a document or section. It often contains branding, a navigation menu, or other introductory content.

<nav>: This element is used to define a section of a document that contains navigation links.

<section>: This element is used to define a section of a document that is logically grouped together, such as a chapter of a book, an article, or a section of a webpage.

<article>: This element is used to define a self-contained piece of content, such as a blog post, news article, or product review.

<aside>: This element is used to define content that is related to the main content of the page, but is not an essential part of it, such as a sidebar or a callout box.

<footer>: This element is used to define the footer of a document or section. It often contains copyright information, contact details, or other closing content.

By using sectioning elements in HTML, you can create a clear and structured hierarchy of content, which can improve the accessibility, usability, and search engine optimization of your website.

Paragraph Element:-

It is one of the most commonly used HTML tags and is used to structure content on a webpage. 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.

<p>This is a paragraph.</p>
<p>This is another paragraph.</p>

Division and Span Elements:-

Span and div are both **generic HTML elements that group together related parts of a web page**. However, they serve different functions. A div element is used for block-level organization and styling of page elements, whereas a span element is used for inline organization and styling.

<div>: The Content Division element. The <div> HTML element is **the generic container for flow content**. It has no effect on the content or layout until styled in some way using CSS (e.g. styling is directly applied to it, or some kind of layout model like Flexbox is applied to its parent element).

****: HTML element is **a generic inline container for phrasing content, which does not inherently represent anything**. It can be used to group elements for styling purposes (using the class or id attributes), or because they share attribute values, such as lang .

A element which is used to color a part of a text:

<p>My mother has blue eyes.</p>

A <div> section in a document that is styled with CSS:

```
<html>
<head>
<style>
.myDiv {
  border: 5px outset red;
  background-color: lightblue;
  text-align: center;
}
</style>
```

```
            </head>
            <body>

            <div class="myDiv">
             <h2>This is a heading in a div element</h2>
             <p>This is some text in a div element.</p>
            </div>

            </body>
            </html>
```

List Element:-

<li>: The List Item element. **The <li> HTML element is used to represent an item in a list**. It must be contained in a parent element: an ordered list ( ), an unordered list ( ), or a menu (<menu>). In menus and unordered lists, list items are usually displayed using bullet points.

HTML lists **allow web developers to group a set of related items in lists**.

Link Element:-

The <link> tag **defines the relationship between the current document and an external resource**. The <link> tag is most often used to link to external style sheets or to add a favicon to your website. The <link> element is an empty element, it contains attributes only.

Link to an external style sheet:

```
            <head>
             <link rel="stylesheet" href="styles.css">
            </head>
```

Character Entities:-

Character entities are **used to display reserved characters in HTML**. &#entity_number; To display a less than sign (<) we must write: < or < Advantage of using an entity name: An entity name is easy to remember.

A character entity looks like this:

&*entity_name*;

OR

&#*entity_number*;

HTML5 Global Attributes:-

The global attributes are attributes that can be used with all HTML elements.

Attribute	Description
accesskey	Specifies a shortcut key to activate/focus an element
class	Specifies one or more classnames for an element (refers to a class in a style sheet)
contenteditable	Specifies whether the content of an element is editable or not
data-*	Used to store custom data private to the page or application
dir	Specifies the text direction for the content in an element
draggable	Specifies whether an element is draggable or not
hidden	Specifies that an element is not yet, or is no longer, relevant
id	Specifies a unique id for an element
lang	Specifies the language of the element's content
spellcheck	Specifies whether the element is to have its spelling and grammar checked or not
style	Specifies an inline CSS style for an element
tabindex	Specifies the tabbing order of an element
title	Specifies extra information about an element
translate	Specifies whether the content of an element should be translated or not

Creating Table Elements:-

```
<!DOCTYPE html>

<html>

<head>

<style>

table {

  font-family: arial, sans-serif;

  border-collapse: collapse;

  width: 100%;

}


td, th {

  border: 1px solid #dddddd;

  text-align: left;
```

```
    padding: 8px;

}


tr:nth-child(even) {

  background-color: #dddddd;

}

</style>

</head>

<body>


<h2>HTML Table</h2>


<table>

 <tr>

  <th>Company</th>

  <th>Contact</th>

  <th>Country</th>

 </tr>

 <tr>

  <td>Alfreds Futterkiste</td>

  <td>Maria Anders</td>

  <td>Germany</td>

 </tr>

 <tr>

  <td>Centro comercial Moctezuma</td>

  <td>Francisco Chang</td>
```

```
                    <td>Mexico</td>

                  </tr>

                  <tr>

                    <td>Ernst Handel</td>

                    <td>Roland Mendel</td>

                    <td>Austria</td>

                  </tr>

                  <tr>

                    <td>Island Trading</td>

                    <td>Helen Bennett</td>

                    <td>UK</td>

                  </tr>

                  <tr>

                    <td>Laughing Bacchus Winecellars</td>

                    <td>Yoshi Tannamuri</td>

                    <td>Canada</td>

                  </tr>

                  <tr>

                    <td>Magazzini Alimentari Riuniti</td>

                    <td>Giovanni Rovelli</td>

                    <td>Italy</td>

                  </tr>

                </table>

                </body>

                </html>
```

Output of Creating Table Elements:-

HTML tables allow web developers to arrange data into rows and columns.

Example

Company	Contact	Country
Alfreds Futterkiste	Maria Anders	Germany
Centro comercial Moctezuma	Francisco Chang	Mexico
Ernst Handel	Roland Mendel	Austria
Island Trading	Helen Bennett	UK
Laughing Bacchus Winecellars	Yoshi Tannamuri	Canada
Magazzini Alimentari Riuniti	Giovanni Rovelli	Italy

Table Elements : Colspan / Rowspan Attributes:-

The rowspan and colspan are the attributes of <td> tag. These are used to specify the number of rows or columns a cell should merge. **The rowspan attribute is for merging rows and the colspan attribute is for merging columns of the table in HTML**.

To make a cell span over multiple columns, use the colspan attribute:

```
<!DOCTYPE html>

<html>

<head>

<style>

table, th, td {

  border: 1px solid black;

  border-collapse: collapse;

}

</style>

</head>

<body>

<h2>Cell that spans two columns</h2>
```

```html
<p>To make a cell span more than one column, use the colspan attribute.</p>

<table style="width:100%">

 <tr>

   <th colspan="2">Name</th>

   <th>Age</th>

 </tr>

 <tr>

   <td>Jill</td>

   <td>Smith</td>

   <td>43</td>

 </tr>

 <tr>

   <td>Eve</td>

   <td>Jackson</td>

   <td>57</td>

 </tr>

</table>

</body>

</html>
```

To make a cell span over multiple rows, use the rowspan attribute:

```html
<!DOCTYPE html>

<html>

<head>

<style>

table, th, td {

  border: 1px solid black;
```

```
   border-collapse: collapse;

}

</style>

</head>

<body>

<h2>Cell that spans two rows</h2>

<p>To make a cell span more than one row, use the rowspan
attribute.</p>

<table style="width:100%">

 <tr>

  <th>Name</th>

  <td>Jill</td>

 </tr>

 <tr>

  <th rowspan="2">Phone</th>

  <td>555-1234</td>

 </tr>

 <tr>

  <td>555-8745</td>

 </tr>

</table>

</body>

</html>
```

Border:-

The HTML border attribute is **used to set visible border width to most HTML elements within the body**.

The CSS border properties allow you to specify the style, width, and color of an element's border.

I have borders on all sides.

I have a red bottom border.

I have rounded borders.

I have a blue left border.

CSS Border Style

The `border-style` property specifies what kind of border to display.

The following values are allowed:

- `dotted` - Defines a dotted border
- `dashed` - Defines a dashed border
- `solid` - Defines a solid border
- `double` - Defines a double border
- `groove` - Defines a 3D grooved border. The effect depends on the border-color value
- `ridge` - Defines a 3D ridged border. The effect depends on the border-color value
- `inset` - Defines a 3D inset border. The effect depends on the border-color value
- `outset` - Defines a 3D outset border. The effect depends on the border-color value
- `none` - Defines no border
- `hidden` - Defines a hidden border

The `border-style` property can have from one to four values (for the top border, right border, bottom border, and the left border).

Example

Demonstration of the different border styles:

```
p.dotted {border-style: dotted;}
p.dashed {border-style: dashed;}
p.solid {border-style: solid;}
p.double {border-style: double;}
p.groove {border-style: groove;}
p.ridge {border-style: ridge;}
p.inset {border-style: inset;}
p.outset {border-style: outset;}
p.none {border-style: none;}
p.hidden {border-style: hidden;}
p.mix {border-style: dotted dashed solid double;}
```

Result:

A dotted border.

A dashed border.

A solid border.

A double border.

A groove border. The effect depends on the border-color value.

A ridge border. The effect depends on the border-color value.

An inset border. The effect depends on the border-color value.

An outset border. The effect depends on the border-color value.

No border.

A hidden border.

A mixed border.

cellspacing and cellpadding attributes:-

Cell spacing refers to **the amount of space that is held between the cells in a table**. It is specified with the cellspacing attribute in the table element. Values are specified in number of pixels. Increasing the cell spacing results in wider shaded borders between cells.

The cell padding attribute places spacing around data within each cell. The cell spacing attribute places space around each cell in the table. The second and third examples below show the use of cell padding or spacing. You can apply both padding and spacing, as in the fourth and fifth examples.

HTML tables can adjust the padding inside the cells, and also the space between the cells.

HTML Table - Cell Padding

Cell padding is the space between the cell edges and the cell content.

By default the padding is set to 0.

To add padding on table cells, use the CSS padding property:

```
<!DOCTYPE html>

<html>

<head>

<style>

table, th, td {

  border: 1px solid black;

  border-collapse: collapse;

}

th, td {

  padding: 15px;
```

```html
      }

</style>

</head>

<body>

<h2>Cellpadding</h2>

<p>Cell padding specifies the space between the cell content and its
borders.</p>

<table style="width:100%">

 <tr>

  <th>Firstname</th>

  <th>Lastname</th>

  <th>Age</th>

 </tr>

 <tr>

  <td>Jill</td>

  <td>Smith</td>

  <td>50</td>

 </tr>

 <tr>

  <td>Eve</td>

  <td>Jackson</td>

  <td>94</td>

 </tr>

 <tr>

  <td>John</td>
```

```
    <td>Doe</td>

    <td>80</td>

   </tr>

</table>

<p><strong>Tip:</strong> Try to change the padding to 5px.</p>

</body>

</html>
```

HTML Table - Cell Spacing

Cell spacing is the space between each cell.

By default the space is set to 2 pixels.

To change the space between table cells, use the CSS border-spacing property on the table element:

```
<!DOCTYPE html>

<html>

<head>

<style>

table, th, td {

  border: 1px solid black;

}

table {

  border-spacing: 30px;

}

</style>

</head>

<body>
```

```html
<h2>Cellspacing</h2>

<p>Change the space between the cells with the border-spacing
property.</p>

<table style="width:100%">

 <tr>

  <th>Firstname</th>

  <th>Lastname</th>

  <th>Age</th>

 </tr>

 <tr>

  <td>Jill</td>

  <td>Smith</td>

  <td>50</td>

 </tr>

 <tr>

  <td>Eve</td>

  <td>Jackson</td>

  <td>94</td>

 </tr>

 <tr>

  <td>John</td>

  <td>Doe</td>

  <td>80</td>

 </tr>

</table>
```

```
                              </body>

                              </html>
```

Creating Form Elements:-

To create an HTML form, we will **use the HTML <form> element**. It starts with the <form> tag and ends with the </form> tag. We can add the input elements within the form tags for taking user input. form elements, such as text box, textarea, etc.

An HTML form is used to collect user input. The user input is most often sent to a server for processing.

```
                    <!DOCTYPE html>

                    <html>

                    <body>

                    <h2>HTML Forms</h2>

                    <form action="/action_page.php">

                      <label for="fname">First name:</label><br>

                      <input type="text" id="fname" name="fname" value="John"><br>

                      <label for="lname">Last name:</label><br>

                      <input type="text" id="lname" name="lname" value="Doe"><br><br>

                      <input type="submit" value="Submit">

                    </form>

                    <p>If you click the "Submit" button, the form-data will be sent to a page
                    called "/action_page.php".</p>

                    </body>

                    </html>
```

HTML Forms

First name:
```
John
```
Last name:
```
Doe
```
```
Submit
```

If you click the "Submit" button, the form-data will be sent to a page called "/action_page.php".

Input Elements – Attributes:-

The input required attribute **specifies that an input field must be filled out before submitting the form**. The required attribute works with the following input types: text, search, url, tel, email, password, date pickers, number, checkbox, radio, and file.

```
<!DOCTYPE html>

<html>

<body>

<h1>The input value attribute</h1>

<p>The value attribute specifies an initial value for an input field:</p>

<form action="/action_page.php">

  <label for="fname">First name:</label><br>

  <input type="text" id="fname" name="fname" value="John"><br>

  <label for="lname">Last name:</label><br>

  <input type="text" id="lname" name="lname" value="Doe"><br><br>

  <input type="submit" value="Submit">

</form>

</body>

</html>
```

The input value attribute

The value attribute specifies an initial value for an input field:

First name:
John
Last name:
Doe

Submit

NOTE:- Please Refer Remaining Attributes from W3schools.com

Color and Date Pickers:-

Color Pickers in HTML **allow the user to select and input a color**. Once submitted, the browser converts the input color into a string that represents it.

Refer this link for color picker: https://www.w3schools.com/colors/colors_picker.asp

The **<input type="date"> defines a date picker**. The resulting value includes the year, month, and day.

Tip: Always add the <label> tag for best accessibility practices!

```
<!DOCTYPE html>

<html>

<body>

<h1>Show a Date Control</h1>

<form action="/action_page.php">

  <label for="birthday">Birthday:</label>

  <input type="date" id="birthday" name="birthday">

  <input type="submit">

</form>

<p><strong>Note:</strong> type="date" is not supported in Internet Explorer 11 or prior Safari 14.1.</p>

</body>

</html>
```

Show a Date Control

Birthday: dd - mm - yyyy 🗓 Submit

Note: type="date" is not supported in Internet Explorer 11 or prior Safari 14.1.

Select and Datalist Elements:-

The <select> element is **used to create a drop-down list**. The <select> element is most often used in a form, to collect user input. The name attribute is needed to reference the form data after the form is submitted (if you omit the name attribute, no data from the drop-down list will be submitted).

```
<!DOCTYPE html>

<html>

<body>

<h1>The select element</h1>

<p>The select element is used to create a drop-down list.</p>

<form action="/action_page.php">

  <label for="cars">Choose a car:</label>

  <select name="cars" id="cars">

    <option value="volvo">Volvo</option>

    <option value="saab">Saab</option>

    <option value="opel">Opel</option>

    <option value="audi">Audi</option>

  </select>

  <br><br>

  <input type="submit" value="Submit">

</form>

<p>Click the "Submit" button and the form-data will be sent to a page on the

server called "action_page.php".</p>

</body>

</html>
```

The select element

The select element is used to create a drop-down list.

Choose a car: Volvo ⌄

Submit

Click the "Submit" button and the form-data will be sent to a page on the server called "action_page.php".

The <datalist> tag **specifies a list of pre-defined options for an <input> element**. The <datalist> tag is used to provide an "autocomplete" feature for <input> elements. Users will see a drop-down list of pre-defined options as they input data.

```html
<!DOCTYPE html>

<html>

<body>

<h1>The datalist element</h1>

<form action="/action_page.php" method="get">

  <label for="browser">Choose your browser from the list:</label>

  <input list="browsers" name="browser" id="browser">

  <datalist id="browsers">

    <option value="Edge">

    <option value="Firefox">

    <option value="Chrome">

    <option value="Opera">

    <option value="Safari">

  </datalist>

  <input type="submit">

</form>

<p><strong>Note:</strong> The datalist tag is not supported in Safari 12.0 (or earlier).</p>

</body>

</html>
```

The datalist element

Choose your browser from the list: [ ] Submit

Note: The datalist tag is not supported in Safari 12.0 (or earlier).

Editing Elements:-

You can edit the HTML — **tags, attributes, and content** — directly in the HTML pane: double-click the text you want to edit, change it, and press Enter to see the changes reflected immediately. You can add any HTML in here: changing the element's tag, changing existing elements, or adding new ones.

Kindly Refer This for More: https://www.w3schools.com/html/html_editors.asp

Media:-

The media attribute **specifies what media or device the linked document is optimized for**. This attribute is used to specify that the target URL is designed for special devices (like iPhone), speech or print media. This attribute can accept several values. Only used if the href attribute is present.

```
<!DOCTYPE html>

<html>

<body>

<h1>The a media attribute</h1>

<p>

<a href="att_a_media.asp?output=print" media="print and
(resolution:300dpi)">Open media attribute page for print</a>

</p>

</body>

</html>
```

The a media attribute

Open media attribute page for print

Iframe:-

An HTML iframe is used **to display a web page within a web page**.

Kindly Refer These To Understand More:

https://www.w3schools.com/html/html_iframe.asp

https://www.w3schools.com/tags/tag_iframe.ASP

Why HTML Security:-

HTML security consists of three different security measures: HTML encryption **to ensure web content cannot be accessed by unauthorized users**. The use of digital certificates to validate a domain and ensure content is coming from a trusted location (the URL in the browser address bar).

For More Refer More on Internet.

HTML Injection:-

HTML Injection also known as Cross Site Scripting. It is **a security vulnerability that allows an attacker to inject HTML code into web pages that are viewed by other users**.
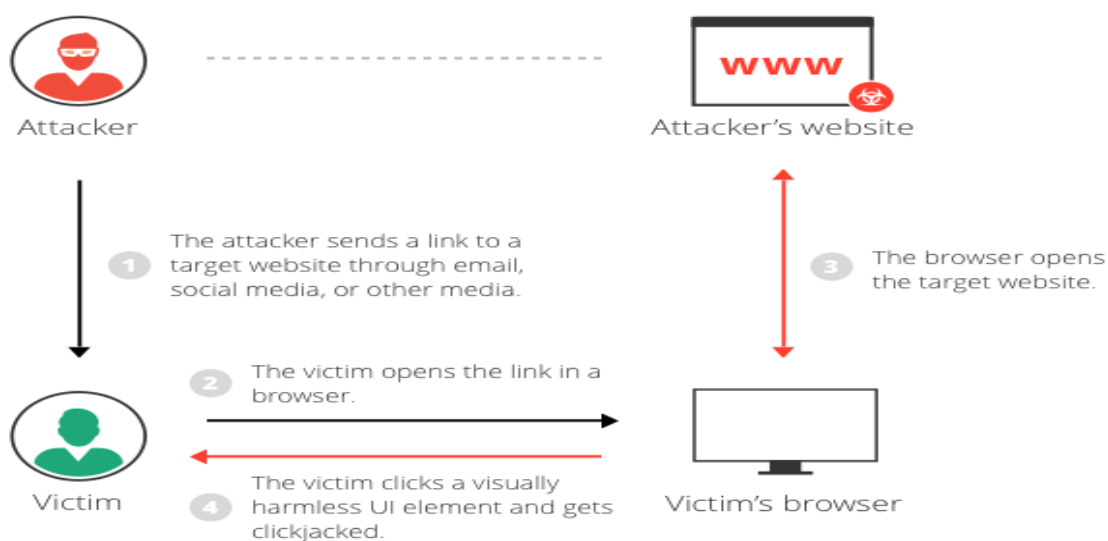
Example Of Html Injection

- First the attacker finds out a site which is vulnerable to HTML injection
- Then the attacker sends the URL with malicious code injected in the URL to the victim user either through email or some other mechanism.
- If the victim user click this malicious URL, it will run the JavaScript or VBScript code with the privileges of the victim user.
- Depending on the code being executed it can give out sensitive information of the user or even compromise the victim's computer.

Clickjacking:-

Clickjacking is **an attack that fools users into thinking they are clicking on one thing when they are actually clicking on another**. Its other name, user interface (UI) redressing, better describes what is going on.

To Know More Refer This: https://www.kaspersky.com/resource-center/definitions/clickjacking

Example:-

Attacker — — — — — — — — — — — — — — — Attacker's website

1. The attacker sends a link to a target website through email, social media, or other media.

2. The victim opens the link in a browser.

3. The browser opens the target website.

4. The victim clicks a visually harmless UI element and gets clickjacked.

Victim — Victim's browser

HTML5 Attributes & Events Vulnerabilities:-

Attributes **provide additional information about elements**. Attributes are always specified in the start tag. Attributes usually come in name/value pairs like: name="value".

Refer here: https://www.w3schools.com/html/html_attributes.asp

Refer this for HTML Events: https://www.w3schools.com/tags/ref_eventattributes.asp

It is **a security vulnerability that allows an attacker to inject HTML code into web pages that are viewed by other users**. Attackers often inject malicious JavaScript, VBScript, ActiveX, and/or HTML into vulnerable applications to deceive the user in order to gather data from them. **HTML Injection** also known as Cross Site Scripting. It is a security vulnerability that allows an attacker to inject HTML code into web pages that are viewed by other users.

Local Storage Vulnerabilities:-

As local storage was never intended to be secure, **there is no data protection and any JavaScript on the website can access it**. Hackers can exploit the existing XSS vulnerability on the website like the following screenshot when the user browses to https://set-localstorage.herokuapp.com/xss-injected-page.html.

HTML5 - Cross-browser support:-

Handling cross browser compatibility in HTML5

HTML5 is now compatible with all popular browsers (Chrome, Firefox, Safari, IE9, and Opera) and with the introduction of DOCTYPE, it is even possible to have a few HTML features in older versions of Internet Explorer too.

Best Practices For HTML Web Pages:-

Refer This for More: https://www.freecodecamp.org/news/html-best-practices/

THANK YOU

HAPPY LEARNING