Table of Contents

1.0 Introduction to HTML 2
1.1 HTML Tags 2
1.2 HTML Element 2
1.3 Basic structure of HTML Document 3
1.4 First HTML Example 4
2.0 HTML Basics 6
2.1 HTML Headings 6
2.2 Html Paragraphs 6
2.3 HTML Line Breaks 7
2.4 Horizontal Rule 7
2.5 HTML Attributes 8
2.6 HTML Tip: Use Lowercase Tags 8
3.0 HTML Text Formatting 9
3.1 HTML Bold and Strong Formatting 9
3.1 HTML *Italic* and *Emphasized* Formatting 10
3.2 HTML Small Formatting 10
3.3 HTML Marked Formatting 11
3.5 HTML Subscript Formatting 12
3.7 HTML Styles 12
4.0 HTML Hyperlinks (Links) 13
5.0 HTML Images 14
5.1 Attributes of <img> tag 14
6.0 HTML Lists 15
6.1 HTML Unordered Lists 15
6.3 Nested list 17
6.4 Description list 18
7.0 HTML Tables 19
7.1 HTML Table Headers 20
7.2 Attributes of a <table> tag 20
a) Table Width 20
b)Cell padding 21
8.0 HTML Frames 23
9.0 HTML Forms 25
9.1 A simple syntax of using <form> is as follows: 25
9.2 Most frequently used form attributes are: 25
When to Use GET? 25
When to Use POST? 26
9.3 The <input> Element 26

1.0 Introduction to HTML

HTML is a markup language for telling a Web browser how to format and display a Web page. Markup language specifies the format in web browser such as opera mini, Mozilla Firefox, Internet explorer, google chrome etc.

1.1 HTML Tags

Predefined tags are employed to describe the format of a document. Most tags have a "**start tag**" and an "**end tag**" (also called container tags) and the content is embedded between the two tags.
Some tags are standalone only without any content

HTML markup tags are usually called HTML tags

- **HTML tags** are keywords surrounded by angle brackets like **<html>**
- HTML tags normally come in pairs like ** and **
- The first tag in a pair is the **start tag**; the second tag is the **end tag**
- Start and end tags are also called **opening tags** and **closing tags**

1.2 HTML Element

The element content is everything between the start and the end tag

- An HTML element starts with a **start tag / opening tag**
- An HTML element ends with an **end tag / closing tag**
- Some HTML elements have **empty content**
- Empty elements are **closed in the start tag**
- Most HTML elements can have **attributes**

1.3 Basic structure of HTML Document

The structure of an HTML document is defined

```
<html>

<head>

<title> </title>

</head>

<body>

</body>

</html>
```

1. **< ! DOCTYPE>** tag specifies the version of the HTML document and other related information.
2. **<HTML> and </HTML>** tags define the start and the end of an HTML document, respectively.
   Within the <HTML> tag pair, there are two main sections namely the **HEAD section** as included inside the <HEAD></HEAD> tag pair and the **BODY section** as included inside the <BODY></BODY> tag pair.

3. The **HEAD** section provides information (e.g. the document title) for the web browser to process the document but the information is not displayed.

4. The web browser only displays the information within the **<BODY></BODY>** tag pair,

NB: *In an HTML document, users can also insert comments inside the comment tag pair:* ***<!- -*** *and* ***-->.*** *The comments are not displayed by the web browser.*

1.4 First HTML Example

HTML can be edited by using a professional HTML editor like:

1. Adobe Dreamweaver
2. Microsoft Expression Web
3. CoffeeCup HTML Editor

However, for learning HTML we recommend a text editor like **Notepad**.
We believe using a simple text editor is a good way to learn HTML. When you save an HTML file, you can use either the **.htm** or the **.html** file extension.

```
<html>
<head>
<title>my first page</title>
</head>
<body>
welcome to my first page
</body>
</html>
```

Steps

Step 1: Start Notepad

To start Notepad go to:

Click Start

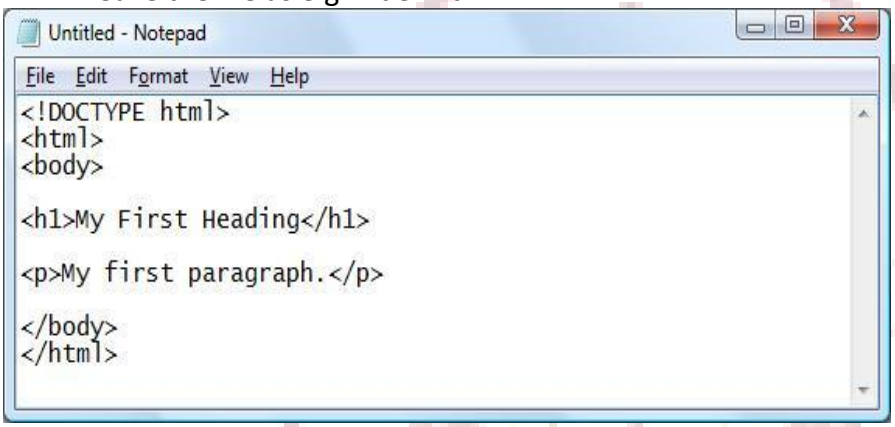
All Programs

Accessories

Notepad

Step 2: Edit Your HTML with Notepad

1. Type your HTML code into your Notepad:
2. Save the file as e.g. index.html

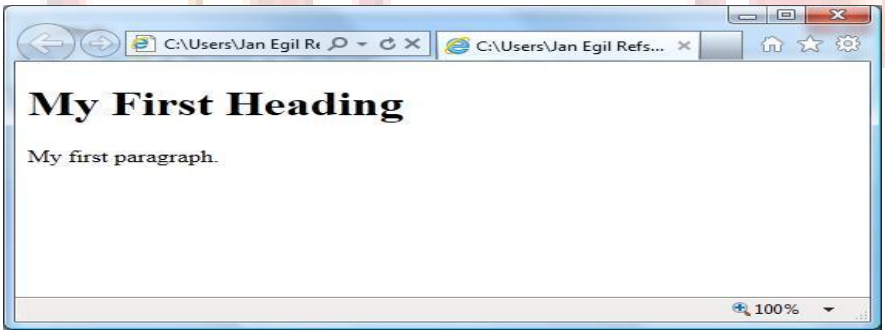


Step 3: Save Your HTML

1. Select **Save as**... In Notepad's file menu.
2. When you save an HTML file, you can use either the .**htm** or the **.html** file extension. There is no difference; it is entirely up to you.

Step 4: Run the HTML in Your Browser

- Just browse the folder and double-click your HTML file.
- The result should look much like this:



2.0 HTML Basics

2.1 HTML Headings

HTML provides six headers called header elements for specifying the relative importance of information.

Headings are defined with the **<h1>** to **<h6>** tags. Example

```
<html>
<head>
<title>headings Example<title>
</head>
<body>
<h1>this is a heading</h1>
<h2>this is a heading</h2>
<h3>this is a heading</h3>
<h4>this is a heading</h4>
<h5>this is a heading</h5>
<h6>this is a heading</h6>
</body>
</html>
```

Search engines use your headings to index the structure and content of your web pages.
H1 headings should be used as main headings, followed by H2 headings, then the less important H3 headings, and so on.

Note: Browsers automatically add some empty space (a margin) before and after each heading.

2.2 Html Paragraphs

HTML paragraphs are defined with the <p> tag. Example

```
<html>
<head>
<title>Paragraph Example</title>
</head>
<body>
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
</body>
</html>
```

Note: Browsers automatically add an empty line before and after a paragraph.

2.3 HTML Line Breaks

The **
** tag defines a line break.
**
** is an empty element without a closing tag.

HTML elements with no content are called empty elements. Empty elements can be closed in the start tag.

Example

```
<html>
<head>
<title>paragraph and line break </title>
</head>
<body>
<p> this is a para <br>graph with the world paragraph in breaking tag </p>
</body>
</html>
```

Note: Adding a slash to the start tag, like <BR />, is the proper way of closing empty elements, accepted by HTML, XHTML and XML. Example

2.4 Horizontal Rule

The **<hr>** tag creates a horizontal line in an HTML page. The **<hr>** element can be used to separate content:

```
<html>
<head>
<title>Horizontal Line </title>
</head>
<body>
<p> this is a paragraph with the world paragraph in breaking tag </p>
<hr />
<p>this is a paragraph</p>
<hr />
<p>this is a paragraph</p>
</body>
</html>
```

2.5 HTML Attributes

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

Attribute Example

- HTML links are defined with the <a> tag. The link address is specified in the **href attribute**:

  This is a link

  NOTE:

  1. Always Quote Attribute Values- Attribute values should always be enclosed in quotes. Double style quotes are the most common, but single style quotes are also allowed.

  2. **Use Lowercase Attributes** -Attribute names and attribute values are case-insensitive. However, the World Wide Web Consortium (W3C) recommends lowercase attributes/attribute values in their HTML 4 recommendation.

2.6 HTML Tip: Use Lowercase Tags

- HTML tags are not case sensitive: <P> means the same as <p>. Many web sites use uppercase HTML tags.
- W3Schools use lowercase tags because the World Wide Web Consortium (W3C) **recommends** lowercase in HTML 4, and **demands** lowercase tags in XHTML.

3.0 HTML Text Formatting

HTML also defines special **elements**, for defining text with a special **meaning**.
HTML uses elements like <b> and <i> for formatting output, like **bold** or *italic* text.
Formatting elements were designed to display special **types of text**:

1. Bold text
2. Important text
3. Italic text
4. Emphasized text
5. Marked text
6. Small text
7. Deleted text
8. Inserted text
9. Subscripts
10. Superscripts

3.1 HTML Bold and Strong Formatting

The HTML **<b>** element defines **bold** text, without any extra importance.

```
<html>
<head>
<title>Bold text example</title>
</head>
<body>
<p>This text is normal.</p>
<p><b>This text is bold.</b></p>
</body>
</html>
```

The HTML **** element defines **strong** text, with added semantic "strong" importance.

```
<html>
<head>
<title>Strong text</title>
</head>
<body>
<p>This text is normal.</p>
<p><strong>this text is strong.</strong></p>
</body>
</html>
```

3.1 HTML *Italic* and *Emphasized* Formatting

The HTML **<i>** element defines *italic* text, without any extra importance.

```
<html>
<head>
<title> Italic text</title>
</head>
<body>
<p>This text is normal.</p>
<p><i>This text is italic.</i></p>
</body>
</html>
```

The HTML **** element defines *emphasized* text, with added semantic importance.

```
<html>
<head>
<title>Emphasized text</title>
</head>
<body>
<p>This text is normal.</p>
<p><em>This text is emphasized.</em></p>
</body>
</html>
```

3.2HTML Small Formatting

The HTML **<small>** element defines **small** text:

```
<html>
<head>
<title>Small text example</title>
</head>
<body>
<h2>HTML <small>Small</small> Formatting</h2>
</body>
</html>
```

3.3HTML Marked Formatting

The HTML **<mark>** element defines **marked** or highlighted text:

```
<html>
<head>
<title>Marked text example</title>
</head>
<body>
This is a <mark> Marked </mark> text
</body>
</html>
```

3.4 HTML ~~Deleted~~ Formatting

The HTML **<del>** element defines **deleted** (removed) of text.

Example

```
<html>
<head>
<title>Deleted text example</title>
</head>
<body>
<p>My favorite color is <del>blue</del> red.</p>
</body>
</html>
```


petnjau@gmail.com

3.5 HTML Subscript Formatting

The HTML **<sub>** element defines **subscripted** text.

Example

```
<html>
<head>
<title>Subscript example</title>
</head>
<body>
<p>This is <sub>subscripted</sub> text.</p>
</body>
</html>
```

3.7 HTML Styles

The style attribute is used to style HTML elements.

```
<html>
<head>
<title>Styles and Colors</title>
</head>
<body style="background-color:PowderBlue;">
<h1>Look! Styles and colors</h1>
<p style="font-family:verdana;color:red">
This text is in Verdana and red</p>
<p style="font-family:times;color:green">
This text is in Times and green</p>
<p style="font-size:30px">This text is 30 pixels high</p>
</body>
</html>
```

4.0 HTML Hyperlinks (Links)

A hyperlink (or link) is a word, group of words, or image that you can click on to jump to a new document or a new section within the current document.

Linking is one of the most important XHTML features which references (or links) to other resources. Both text and image can act as links.

When you move the cursor over a link in a Web page, the arrow will turn into a little hand.

Links are specified in HTML using the <a> tag.

The <a> tag (anchor) can be used in three ways:

1. To create a link to another document, by using the **href** attribute
2. To create a bookmark inside a document, by using the **name** attribute
3 Anchors can link to email addresses using **mailto:** URL-when one clicks on this type of anchored link, most browsers launch the default e-mail program (e.g Outlook Express)

Link text

The href attribute specifies the destination of a link.

Example

Visit W3Schools

```
<html>
<head>
<title>Link Example </title>
</head>
<body>
<a href="index.html">Home</a>
<a href=”http://www.yahoo.com/”>yahoo.com</a>
<a href="tutorial.html">html tutorial</a> this is a link to a page on this website.
<a href=”mailto:petnjau@gmail.com”/>Email Us</a>
</body>
</html>
```

5.0 HTML Images

<img> tag is used to insert the image in the document. The image file's location is specified using the image element's **src** attribute
systax: <img src="mypic.jpg" alt="no image" height="70" width="60">

The <img> tag is empty, which means that it contains attributes only, and has no closing tag.

5.1 Attributes of <img> tag

1. **Src**- stands for "source". The value of the src attribute is the URL of the image you want to display.
2. **Alt**- **alt** attribute specifies an alternate text for an image, if the image cannot be displayed.
3. Optional attributes **width** and **height** specify the image's width and height respectively.

```
<html>
<head>
<title>images</title>
</head>
<body>
<img src="smiley.jpg" width="32" height="32" alt="no image" />
</body>
</html>
```

5.2 Make a hyperlink of an image

How to use an image as a link

```
<html>
<head>
<title>Image link</title>
</head>
<body>
<p>Create a link of an image:
<a href="tutorial.html">
<img src="smiley.gif" alt="HTML tutorial" width="32" height="32" />
</a></p>
<p>No border around the image, but still a link:
<a href=" tutorial.html ">
<img border="0" src="smiley.gif" alt="HTML tutorial" width="32" height="32" />
</a></p>
</body>
</html>
```

6.0 HTML Lists

The most common HTML lists are ordered and unordered lists:

An ordered list:

1. The first list item
2. The second list item
3. The third list item

An unordered list:

- List item
- List item
- List item

6.1 HTML Unordered Lists

An unordered list starts with the **<ul>** tag.
Each list item starts with the **<li>** tag.
Note: The list items are marked with bullets (typically small black circles).

```
<html>
<head>
<title>Unordered List</title>
</head>
<body>
<ul>
<li>Coffee</li>
<li>Milk</li>
</ul>
</body>
</html>
```

Type attribute specifies the type of bullet to use (circle, square, disk)

```
<html>
<head>
<title>List type</title>
</head>
<body>
<ul type="disc">
 <li>Apples</li>
 <li>Bananas</li>
</ul>
</body>
</html>
```

6.2 HTML Ordered Lists

An ordered list starts with the <ol> tag. Each list item starts with the <li> tag.
The list items are marked with numbers.

```
<html>
<head>
<title>Ordered list</title>
</head>
<body>
<ol>
<li>Coffee</li>
<li>Milk</li>
</ol>
</body>
</html>
```

Type attribute specifies the type of ordered list to use (A, I, i)

```
<h4>Roman numbers list:</h4>
<ol type="i">
 <li>Apples</li>
<li>Bananas</li>
 <li>Lemons</li>
 <li>Oranges</li>
 </ol>
```

To specify the starting value of the ordered list, we can use the **START** attribute of the **<OL>** tag. For example, **<OL TYPE=1 START=6>** means that the first item of the list starts at ***6***

6.3 Nested list

Demonstrates how you can nest lists

```
<html>
<body>
<h4>A nested List:</h4>
<ul>
 <li>Coffee</li>
 <li>Tea
  <ul>
  <li>Black tea</li>
  <li>Green tea</li>
  </ul>
 </li>
 <li>Milk</li>
</ul>
</body>
</html>
```

Nested list 2

demonstrates a more complicated nested list

```
<html>
<body>
<h4>A nested List:</h4>
<ul>
 <li>Coffee</li>
 <li>Tea
  <ul>
  <li>Black tea</li>
  <li>Green tea
   <ul>
   <li>China</li>
   <li>Africa</li>
   </ul>
  </ul>
 <li>Milk</li>
</ul>
</body>
</html>
```

A nested List:

- Coffee
- Tea
  - Black tea
  - Green tea
    - China
    - Africa
- Milk

6.4 Description list

- A description list is a list of terms/names, with a description of each term/name.
- The **<dl>** tag defines a description list.
- The **<dl>** tag is used in conjunction with **<dt>** (defines terms/names)
- and <**dd>** describes each term/name.

Example

```
<html>
<head>
<title>Definition list</title>
</head>
<body>
<h4>A Definition List:</h4>
<dl>
<dt>Coffee</dt>
 <dd>Black hot drink</dd>
 <dt>Milk</dt>
 <dd>White cold drink</dd>
</dl>
</body>
</html>
```

A Definition List:

Coffee
: Black hot drink

Milk
: White cold drink

7.0 HTML Tables

- Tables are defined with the **<table>** tag.
- A table is divided into rows with the **<tr>** tag,
- And each row is divided into data cells with the **<td>** tag, **td stands for "table data**," and holds the content of a data cell.
- A **<td>** tag can contain text, links, images, lists, forms, other tables, etc.

```
<html>
<head>
<title>Table example</title>
</head>
<body>
<table border="1">
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td>row 2, cell 2</td>
</tr>
</table>
</body>
</html>
```

row 1, cell 1	row 1, cell 2
row 2, cell 1	row 2, cell 2

7.1 HTML Table Headers

Header information in a table is defined with the **<th>** tag.
The text in a th element will be **bold and centered.**

```
<html>
<head>
<title>table with headers</title>
</head>
<body>
<table border="1">
<tr>
<th>Header 1</th>
<th>Header 2</th>
</tr>
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td>row 2, cell 2</td>
</tr>
</table>
</body>
</html>
```

Header 1	Header 2
row 1, cell 1	row 1, cell 2
row 2, cell 1	row 2, cell 2

7.2 Attributes of a <table> tag

a) Table Width

-This defines the width of the table

```
<html>
<body>
<table border="1" width=”50%”>
 <tr>
  <th>Month</th>
  <th>Savings</th>
 </tr>
 <tr>
  <td>January</td>
  <td>$100</td>
 </tr>
 <tr>
  <td>February</td>
  <td>$50</td>
 </tr>
</table>
</body>
</html>
```

b) Cell padding

-cellpadding creates more white space between the cell content and its borders.

```
<html>
<head><title>Cellpadding</title></head>
<body>
<h4>With cellpadding:</h4>
<table border="1" cellpadding="10">
<tr>
 <td>First</td>
 <td>Row</td>
</tr>
<tr>
 <td>Second</td>
 <td>Row</td>
</tr>
</table>
</body>
</html>
```

First	Row
Second	Row

Cell spacing

-cellspacing is used to increase the distance between the cells (controls the space between table cells)

```
<html>
<head><title>spacing</title></head>
<body>
<h4>With cellspacing:</h4>
<table border="1" cellspacing="10">
<tr>
 <td>First</td>
 <td>Row</td>
</tr>
<tr>
 <td>Second</td>
 <td>Row</td>
</tr>
</table>
</body>
</html>
```

With cellspacing:

First	Row
Second	Row

petnjau@gmail.com

7.3 Table cells that span more than one row/column.

How to define table cells that span more than one row or one column.

```
<html>
<head>
<title>Rowspan and colspan</title>
</head>
<body>
<h4>Cell that spans two columns :</h4>
<table border="1">
<tr>
 <th>Name</th>
 <th colspan="2">Telephone</th>
</tr>
<tr>
 <td>Bill Gates</td>
 <td>555 77 854</td>
 <td>555 77 855</td>
</tr>
</table>
<h4>Cell that spans two rows:</h4>
<table border="1">
<tr>
 <th>First Name:</th>
 <td>Bill Gates</td>
</tr>
<tr>
 <th rowspan="2">Telephone :</th>
 <td>555 77 854</td>
</tr>
<tr>
 <td>555 77 855</td>
</tr>
</table>
</body>
</html>
```

Cell that spans two columns using colspan attribute:

Name	Telephone	
Bill Gates	555 77 854	555 77 855

Cell that spans two rows using rowspan attribute:

First Name:	Bill Gates
Telephone:	555 77 854
	555 77 855

8.0 HTML Frames

With frames, you can display more than one HTML document in the same browser window. Each HTML document is called a frame, and each frame is independent of the others.
The disadvantages of using frames are:

- The web developer must keep track of more HTML documents
- It is difficult to print the entire page

The HTML frameset Element

The frameset element holds one or more frame elements. Each frame element can hold a separate document. The frameset element states HOW MANY columns or rows there will be in the frameset, and HOW MUCH percentage/pixels of space will occupy each of them.

The <frame> tag defines one particular window (frame) within a frameset.
In the example below we have a frameset with two columns.
The first column is set to 25% of the width of the browser window. The second column is set to 75% of the width of the browser window. The document "frame_a.htm" is put into the first column, and the document "frame_b.htm" is put into the second column:

```
<frameset cols="25%,75%">
  <frame src="frame_a.htm" />
  <frame src="frame_b.htm" />
</frameset>
```

Note: The frameset column size can also be set in pixels (cols="200,500"), and one of the columns can be set to use the remaining space, with an asterisk (cols="25%,*").
How to use the <noframes> tag
How to use the <noframes> tag (for browsers that do not support frames).

```
<html>
<frameset cols="25%,50%,25%">
  <frame src="frame_a.htm" />
  <frame src="frame_b.htm" />
  <frame src="frame_c.htm" />
<noframes>
<body>Your browser does not handle frames!</body>
</noframes>
</frameset>
</html>
```

Nested framesets

How to create a frameset with three documents, and how to mix them in rows and columns.

```
<html>
<frameset rows="50%,50%">
<frame src="frame_a.htm" />
 <frameset cols="25%,75%">
 <frame src="frame_b.htm" />
 <frame src="frame_c.htm" />
  </frameset>
</frameset>
</html>
```

Frameset with noresize="noresize"

How to use the noresize attribute. Move the mouse over the borders between the frames and notice that you cannot move the borders.

```
<html>
<frameset rows="50%,50%">
  <frame noresize="noresize" src="frame_a.htm" />
  <frame noresize="noresize" src="frame_b.htm" />
</frameset>
</html>
```

Navigation frame

How to make a navigation frame. The navigation frame contains a list of links with the second frame as the target. The file called "tryhtml_contents.htm" contains three links. The source code of the links:

```
<a href ="frame_a.htm" target ="showframe">Frame a</a><br>
<a href ="frame_b.htm" target ="showframe">Frame b</a><br>
<a href ="frame_c.htm" target ="showframe">Frame c</a>
```

The second frame will show the linked document.

```
<html>
<frameset cols="120,*">
<frame src="tryhtml_contents.htm" />
  <frame src="frame_a.htm" name="showframe" />
</frameset>
</html>
```

Inline frame

How to create an inline frame (a frame inside an HTML page)

```
<html>
<body>
<iframe src="default.asp"></iframe>
<p>Some older browsers don't support iframes.</p>
<p>If they don't, the iframe will not be visible.</p>
</body>
</html>
```

9.0HTML Forms

- ✓ HTML forms are used to collect user input.
- ✓ The **<form>** element defines an HTML form:
- ✓ A form can contain input elements like text fields, checkboxes, radio-buttons, submit buttons, select lists, textarea, fieldset, legend, and label elements.

```
<form>
.
input elements
.
</form>
```

<form name="input" action="html_form_action.asp" method="get">
The attribute method specifies how the form's data will be sent to web server.
using **method="post"** appends for data to the browser request which contains the protocol (HTTP) and the requested resource's URL.
Met**hod="get"** appends the form directly to the end of the URL.

9.1 A simple syntax of using <form> is as follows:

<form action="back-end script" name="form name" method="posting method">
Form elements like input, textarea etc.
</form>

9.2 Most frequently used form attributes are:

1. **Name:** This is the name of the form.
2. **Action:** Here you will specify any script URL which will receive uploaded data.
3. **Method:** Here you will specify method to be used to upload data. It can take various values but most frequently used are GET and POST.
4. **Target:** It specifies the target page where the result of the script will be displayed. It takes values like _blank, _self, _parent etc

When to Use GET?

- ✓ You can use GET (the default method): If the form submission is passive (like a search engine query), and without sensitive information.
- ✓ When you use GET, the form data will be visible in the page address:

action_page.php?firstname=Mickey&lastname=Mouse

GET is best suited to short amounts of data. Size limitations are set in your browser.

When to Use POST?

- ✓ You should use POST: If the form is updating data, or includes sensitive information (password).
- ✓ POST offers better security because the submitted data is not visible in the page address.

9.3 The <input> Element

- ✓ The **<input>** element is the most important **form element**.
- ✓ The <input> element has many variations, depending on the **type** attribute.

Examples of input elements:

1) Text Fields

<input type="text" />- defines a one-line input field that a user can enter text into:

Example:

```
<form action=”index.php” method=”post”>
First name: <input type="text" name="firstname" /><br>
Last name: <input type="text" name="lastname" />
</form>
```

2) Password Field

<input type="password" />- defines a password field:

```
<form action=”index.php” method=”post”>
Password: <input type="password" name="pwd" />
</form>
```

3) Radio Buttons

- ✓ **<input type="radio" />**- defines a radio button.
- ✓ Radio buttons let a user select ONLY ONE option in a number of choices:

```
<form action=”index.php” method=”post”>
<input type="radio" name="gender" value="male" /> Male<br>
<input type="radio" name="gender" value="female" /> Female
</form>
```

4) Checkboxes

- ✓ **<input type="checkbox" />** defines a checkbox.
- ✓ Checkboxes let a user select ONE or MORE options of a limited number of choices.

```
<form action=”index.php” method=”post”>
<input type="checkbox" name="vehicle" value="Bike" /> I have a bike<br />
<input type="checkbox" name="vehicle" value="Car" /> I have a car
</form>
```

5) Submit Button

- ✓ **<input type="submit" />** defines a submit button.
- ✓ A submit button is used to send form data to a server.
- ✓ The data is sent to the page specified in the form's **action attribute**.
- ✓ The file defined in the action attribute usually does something with the received input:

```
<form action=”index.php” method=”post”>
 Username: <input type="text" name="user" />
 <input type="submit" value="Submit" />
 </form>
```

6) The <select> Element (Drop-Down List)

- ✓ The **<option>** element defines the options to select.
- ✓ The list will normally show the first item as selected.
- ✓ You can add a selected attribute to define a predefined option.

```
<html>
<head>
<title>Dropdown</title>
</head>
<body>
<form action="">
<select name="cars">
<option value="volvo">Volvo</option>
<option value="saab">Saab</option>
<option value="fiat">Fiat</option>
<option value="audi">Audi</option>
</select>
</form>
</body>
</html>
```

7) <u>Textarea</u>

In a text-area the user can write an unlimited number of characters.
The tag **<textarea>** is used to define a text area
It has the following attributes

1. **Name**-used to identify the textarea
2. **Cols**- used to set number of characters that can be typed per line. i.e width of the textarea.
3. **Rows**- used to define the number of lines before an overflow. i.e the height of textarea

Example:
<textarea name=”requests” rows="10” cols="30"> </textarea>