Module-4A

What is HTML?

HTML is a language for describing web pages.

- HTML stands for **H**yper **T**ext **M**arkup **L**anguage
- HTML is not a programming language, it is a **markup language**
- A markup language is a set of **markup tags**
- HTML uses **markup tags** to describe web pages

HTML Tags

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**

HTML Documents = Web Pages

- HTML documents **describe web pages**
- HTML documents **contain HTML tags** and plain text
- HTML documents are also **called web pages**

The purpose of a web browser (like Internet Explorer or Firefox) is to read HTML documents and display them as web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page:

```
<html>
        <body>
                <h1>My First Heading</h1>
                <p>My first paragraph.</p>
        </body>
</html>
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

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

What You Need

You don't need any tools to learn HTML at W3Schools.

- You don't need an HTML editor
- You don't need a web server
- You don't need a web site

Editing HTML

HTML can be written and edited using many different editors like Dreamweaver and Visual Studio.
However, in this tutorial we use a plain text editor (like Notepad) to edit HTML. We believe using a plain text editor is the best way to learn HTML.

Create Your Own Test Web

********CREATE FILE**********

Use Your Test Web For Learning

We suggest you experiment with everything you learn at W3Schools by editing your web files with a text editor (like Notepad).
Note: If your test web contains HTML markup tags you have not learned, don't panic. You will learn all about it in the next chapters.

.HTM or .HTML File Extension?

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.

HTML Elements

An HTML element is everything from the start tag to the end tag:

Start tag *	Element content	End tag *
	This is a paragraph	
	This is a link	

* The start tag is often called the **opening tag**. The end tag is often called the **closing tag**.

HTML Element Syntax

- An HTML element starts with a **start tag / opening tag**
- An HTML element ends with an **end tag / closing tag**
- The **element content** is everything between the start and the end tag
- Some HTML elements have **empty content**
- Empty elements are **closed in the start tag**

Tip: You will learn about attributes in the next chapter of this tutorial.

Nested HTML Elements

Most HTML elements can be nested (can contain other HTML elements).
HTML documents consist of nested HTML elements.

HTML Document Example

```
<html>
        <body>
                <p>This is my first paragraph.</p>
```

```
    </body>
</html>
```

The example above contains 3 HTML elements.

HTML Example Explained

The element:

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

The element defines a paragraph in the HTML document.
The element has a start tag and an end tag .
The element content is: This is paragraph.

The <body> element:

```
<body>
    <p>This is paragraph.</p>
</body>
```

The <body> element defines the body of the HTML document.
The element has a start tag <body> and an end tag </body>.
The element content is another HTML element (a p element).

The <html> element:

```
<html>
    <body>
        <p>This is paragraph.</p>
    </body>
</html>
```

The <html> element defines the whole HTML document.
The element has a start tag <html> and an end tag </html>.
The element content is another HTML element (the body element).

Don't Forget the End Tag

Some HTML elements might display correctly even if you forget the end tag:

```
<p>This is a paragraph
<p>This is a paragraph
```

The example above works in most browsers, because the closing tag is considered optional.
Never rely on this. Many HTML elements will produce unexpected results and/or errors if you forget the end tag .

Empty HTML Elements

HTML elements with no content are called empty elements.

 is an empty element without a closing tag (the <br> tag defines a line break).
Tip: In XHTML, all elements must be closed. Adding a slash inside the start tag, like <br />, is the proper way of closing empty elements in XHTML (and XML).

HTML Tip: Use Lowercase Tags

HTML tags are not case sensitive: <P> means the same as . 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.

HTML Block and Inline Elements

Block-level Elements

A block-level element always starts on a new line, and the browsers automatically add some space (a margin) before and after the element.
A block-level element always takes up the full width available (stretches out to the left and right as far as it can).
Two commonly used block elements are: and .

- The element defines a paragraph in an HTML document.
- The element defines a division or a section in an HTML document.

Example
Hello World
Hello World

Here are the block-level elements in HTML:

<address> <article> <aside> <blockquote> <canvas> <dd> <dl> <dt>

Inline Elements

An inline element does not start on a new line.
An inline element only takes up as much width as necessary.
This is a element inside a paragraph.
Example
Hello World

Here are the inline elements in HTML:

<a> <abbr> <i> <img> <input> <kbd> <label> <map> <object> <output> <q>
<select> <small>   <sub> <sup> <textarea>

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**:
Example
This is a link

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.
Tip of the day: In some rare situations, when the attribute value itself contains quotes, it is necessary to use single quotes: name='John "ShotGun" Nelson'

HTML Tip: 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.
Newer versions of (X)HTML will demand lowercase attributes.

HTML Headings

HTML headings are defined with the <h1> to <h6> tags.

Example

```
<h1>This is a heading</h1>
<h2>This is a heading</h2>
<h3>This is a heading</h3>
```

HTML Links

HTML links are defined with the <a> tag.

Example

```
<a href="http://www.w3schools.com">This is a link</a>
```

Note: The link address is specified in the href attribute.
(You will learn about its attributes in later chapters).

HTML Images

HTML images are defined with the <img> tag.

Example

```
<img src="w3schools.jpg" width="104" height="142" />
```

Note: The name and the size of the image are provided as attributes.

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**:

Example

```
<a href="http://www.w3schools.com">This is a link</a>
```

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.
Tip of the day: In some rare situations, when the attribute value itself contains quotes, it is necessary to use single quotes: name='John "ShotGun" Nelson'

HTML Tip: 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.
Newer versions of (X)HTML will demand lowercase attributes.

HTML Attributes Reference

A complete list of legal attributes for each HTML element is listed in our:Below is a list of some attributes that are standard for most HTML elements:

Attribute	Value	Description
Class	*Classname*	Specifies a classname for an element
Id	*Id*	Specifies a unique id for an element

Style	*style_definition*	Specifies an inline style for an element
Title	*tooltip_text*	Specifies extra information about an element (displayed as a tool tip)

HTML Headings

Headings are defined with the <h1> to <h6> tags.
<h1> defines the most important heading. <h6> defines the least important heading.

Example

```
<h1>This is a heading</h1>
<h2>This is a heading</h2>
<h3>This is a heading</h3>
```

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

Headings Are Important

Use HTML headings for headings only. Don't use headings to make text **BIG** or **bold**.
Search engines use your headings to index the structure and content of your web pages.
Since users may skim your pages by its headings, it is important to use headings to show the document structure.
H1 headings should be used as main headings, followed by H2 headings, then the less important H3 headings, and so on.

HTML Lines

The <hr /> tag creates a horizontal line in an HTML page.

The hr element can be used to separate content:

Example

```
<p>This is a paragraph</p>
<hr />
<p>This is a paragraph</p>
<hr />
<p>This is a paragraph</p>
```

HTML Comments

Comments can be inserted into the HTML code to make it more readable and understandable.
Comments are ignored by the browser and are not displayed.
Comments are written like this:

Example

```
<!-- This is a comment -->
```

Note: There is an exclamation point after the opening bracket, but not before the closing bracket.

HTML Tip - How to View HTML Source

Have you ever seen a Web page and wondered "Hey! How did they do that?"
To find out, right-click in the page and select "View Source" (IE) or "View Page Source" (Firefox), or similar for other browsers. This will open a window containing the HTML code of the page.

HTML Tag Reference

W3Schools' tag reference contains additional information about these tags and their attributes.

You will learn more about HTML tags and attributes in the next chapters of this tutorial.

Tag	Description
<html>	Defines an HTML document
<body>	Defines the document's body
<h1> to <h6>	Defines HTML headings
<hr />	Defines a horizontal line
<!-->	Defines a comment

HTML Paragraphs

Paragraphs are defined with the <p> tag.

Example

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

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

Don't Forget the End Tag

Most browsers will display HTML correctly even if you forget the end tag:

Example

```
<p>This is a paragraph
<p>This is another paragraph
```

The example above will work in most browsers, but don't rely on it. Forgetting the end tag can produce unexpected results or errors.

Note: Future version of HTML will not allow you to skip end tags.

HTML Line Breaks

Use the <br /> tag if you want a line break (a new line) without starting a new paragraph:

Example

```
<p>This is<br />a para<br />graph with line breaks</p>
```

The <br /> element is an empty HTML element. It has no end tag.

 or

In XHTML, XML, elements with no end tag (closing tag) are not allowed.

Even if <br> works in all browsers, writing <br /> instead works better in XHTML and XML applications.

HTML Output - Useful Tips

You cannot be sure how HTML will be displayed. Large or small screens, and resized windows will create different results.

With HTML, you cannot change the output by adding extra spaces or extra lines in your HTML code.

The browser will remove extra spaces and extra lines when the page is displayed. Any number of lines count as one line, and any number of spaces count as one space.

HTML Tag Reference

W3Schools' tag reference contains additional information about HTML elements and their attributes.

Tag	Description
<p>	Defines a paragraph
 	Inserts a single line break

HTML Text Formatting

This text is bold

This text is big

This text is italic

```
This is computer output
```

This is $_{\text{subscript}}$ and $^{\text{superscript}}$

HTML Formatting Tags

HTML uses tags like <b> and <i> for formatting output, like **bold** or *italic* text.
These HTML tags are called formatting tags (look at the bottom of this page for a complete reference).

Often renders as <b>, and renders as <i>.

However, there is a difference in the meaning of these tags:

<b> or <i> defines bold or italic text only.

 or means that you want the text to be rendered in a way that the user understands as "important". Today, all major browsers render strong as bold and em as italics. However, if a browser one day wants to make a text highlighted with the strong feature, it might be cursive for example and not bold!

HTML Fonts

The HTML <font> Tag Should NOT be Used

The <font> tag is deprecated in HTML 4, and removed from HTML5.
The World Wide Web Consortium (W3C) has removed the <font> tag from its recommendations.
In HTML 4, style sheets (CSS) should be used to define the layout and display properties for many HTML elements.
The example below shows how the HTML could look by using the <font> tag:

Example

```
<p>
        <font size="5" face="arial" color="red">
                This paragraph is in Arial, size 5, and in red text color.
        </font>
</p>
<p>
        <font size="3" face="verdana" color="blue">
                This paragraph is in Verdana, size 3, and in blue text color.
        </font>
</p>
```

HTML Styles - CSS

CSS is used to style HTML elements.

Look! Styles and colors

This text is in Verdana and red

This text is in Times and blue

This text is 30 pixels high

Styling HTML with CSS

CSS was introduced together with HTML 4, to provide a better way to style HTML elements. CSS can be added to HTML in the following ways:

- in **Cascading Style Sheet files** (CSS files)
- in the **<style> element** in the HTML head section
- in the **style attribute** in single HTML elements

Using the HTML Style Attribute

It is time consuming and not very practical to style HTML elements using the style attribute. **The preferred way to add CSS to HTML, is to put CSS syntax in separate CSS files.** However, in this HTML tutorial we will introduce you to CSS using the style attribute. This is done to simplify the examples. It also makes it easier for you to edit the code and try it yourself.

HTML Style Example - Background Color

The background-color property defines the background color for an element:

Example

```
<html>
        <body style="background-color:yellow;">
                <h2 style="background-color:red;">This is a heading</h2>
                <p style="background-color:green;">This is a paragraph.</p>
        </body>
```

```
</html>
```

HTML Style Example - Font, Color and Size

The font-family, color, and font-size properties defines the font, color, and size of the text in an element:

Example

```
<html>
	<body>
		<h1 style="font-family:verdana;">A heading</h1>
		<p style="font-family:arial;color:red;font-size:20px;">A paragraph.</p>
	</body>
</html>
```

The font-family, color, and font-size properties make the old <font> tag obsolete.

HTML Style Example - Text Alignment

The text-align property specifies the horizontal alignment of text in an element:

Example

```
<html>
	<body>
		<h1 style="text-align:center;">Center-aligned heading</h1>
		<p>This is a paragraph.</p>
	</body>
</html>
```

The text-align property makes the old tag obsolete.

Deprecated Tags and Attributes

In HTML 4, several tags and attributes were deprecated. Deprecated means that they will not be supported in future versions of HTML.

The message is clear: Avoid using deprecated tags and attributes!

These tags and attributes should be avoided:

Tags	Description
	Deprecated. Defines centered content
<font> and <basefont>	Deprecated. Defines HTML fonts
<s> and <strike>	Deprecated. Defines strikethrough text
<u>	Deprecated. Defines underlined text
Attributes	**Description**
align	Deprecated. Defines the alignment of text
bgcolor	Deprecated. Defines the background color
color	Deprecated. Defines the text color

For all of the above: Use styles instead!

HTML Links

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.
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 can be used in two 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

HTML Link Syntax

The HTML code for a link is simple. It looks like this:

```
<a href="url">Link text</a>
```

The href attribute specifies the destination of a link.

Example

```
<a href="http://www.w3schools.com/">Visit W3Schools</a>
```

Clicking on this hyperlink will send the user to W3Schools' homepage.

Tip: The "*Link text*" doesn't have to be text. It can be an image or any other HTML element.

HTML Links - The target Attribute

The target attribute specifies where to open the linked document.The example below will open the linked document in a new browser window or a new tab:

Example

```
<a href="http://www.w3schools.com/" target="_blank">Visit W3Schools!</a>
```

HTML Links - The name Attribute

The name attribute specifies the name of an anchor.

The name attribute is used to create a bookmark inside an HTML document.

Note: The upcoming HTML5 standard suggests using the id attribute instead of the name attribute for specifying the name of an anchor. Using the id attribute actually works also for HTML4 in all modern browsers.

Bookmarks are not displayed in any special way. They are invisible to the reader.

Example

A named anchor inside an HTML document:

```
<a name="tips">Useful Tips Section</a>
```

Create a link to the "Useful Tips Section" inside the same document:

```
<a href="#tips">Visit the Useful Tips Section</a>
```

Or, create a link to the "Useful Tips Section" from another page:

```
<a href="http://www.w3schools.com/html_links.htm#tips">
Visit the Useful Tips Section</a>
```

Basic Notes - Useful Tips

Note: Always add a trailing slash to subfolder references. If you link like this: href="http://www.w3schools.com/html", you will generate two requests to the server, the

server will first add a slash to the address, and then create a new request like this: href="http://www.w3schools.com/html/".

Tip of the day: Named anchors are often used to create "table of contents" at the beginning of a large document. Each chapter within the document is given a named anchor, and links to each of these anchors are put at the top of the document.

HTML Link Tags

Tag	Description
<a>	Defines an anchor

HTML Images

HTML Images - The <img> Tag and the Src Attribute

In HTML, images are defined with the <img> tag.
The <img> tag is empty, which means that it contains attributes only, and has no closing tag.
To display an image on a page, you need to use the src attribute. Src stands for "source". The value of the src attribute is the URL of the image you want to display.

Syntax for defining an image:

```
<img src="url" alt="some_text"/>
```

The URL points to the location where the image is stored. An image named "boat.gif", located in the "images" directory on "www.w3schools.com" has the URL: http://www.w3schools.com/images/boat.gif.
The browser displays the image where the <img> tag occurs in the document. If you put an image tag between two paragraphs, the browser shows the first paragraph, then the image, and then the second paragraph.

HTML Images - The Alt Attribute

The required alt attribute specifies an alternate text for an image, if the image cannot be displayed.

The value of the alt attribute is an author-defined text:

```
<img src="boat.gif" alt="Big Boat" />
```

The alt attribute provides alternative information for an image if a user for some reason cannot view it (because of slow connection, an error in the src attribute, or if the user uses a screen reader).

HTML Images - Set Height and Width of an Image

The height and width attributes are used to specify the height and width of an image.

The attribute values are specified in pixels by default:

```
<img src="pulpit.jpg" alt="Pulpit rock" width="304" height="228" />
```

Tip: It is a good practice to specify both the height and width attributes for an image. If these attributes are set, the space required for the image is reserved when the page is loaded.

However, without these attributes, the browser does not know the size of the image. The effect will be that the page layout will change during loading (while the images load).

Basic Notes - Useful Tips

Note: If an HTML file contains ten images - eleven files are required to display the page right. Loading images takes time, so my best advice is: Use images carefully.

Note: When a web page is loaded, it is the browser, at that moment, that actually gets the image from a web server and inserts it into the page. Therefore, make sure that the images actually stay in the same spot in relation to the web page, otherwise your visitors will get a broken link icon. The broken link icon is shown if the browser cannot find the image.

HTML Image Tags

Tag	Description
<img />	Defines an image
<map>	Defines an image-map
<area />	Defines a clickable area inside an image-map

HTML Tables

HTML Tables

Apples	**44%**
Bananas	23%
Oranges	13%
Other	10%

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.

Table Example

```
<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>
```

How the HTML code above looks in a browser:

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

HTML Tables and the Border Attribute

If you do not specify a border attribute, the table will be displayed without borders. Sometimes this can be useful, but most of the time, we want the borders to show.

To display a table with borders, specify the border attribute:

```
<table border="1">
        <tr>
                <td>Row 1, cell 1</td>
                <td>Row 1, cell 2</td>
        </tr>
</table>
```

HTML Table Headers

Header information in a table are defined with the <th> tag.
All major browsers display the text in the <th> element as bold and centered.

```
<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>
```

How the HTML code above looks in your browser:

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

HTML Lists

The most common HTML lists are ordered and unordered lists:

HTML 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

HTML Unordered Lists

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

```
<ul>
        <li>Coffee</li>
        <li>Milk</li>
</ul>
```

How the HTML code above looks in a browser:

- Coffee
- Milk

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.

```
<ol>
        <li>Coffee</li>
        <li>Milk</li>
</ol>
```

How the HTML code above looks in a browser:

1. Coffee
2. Milk

HTML Definition Lists

A definition list is a list of items, with a description of each item.
The <dl> tag defines a definition list.
The <dl> tag is used in conjunction with <dt> (defines the item in the list) and <dd> (describes the item in the list):

```
<dl>
<dt>Coffee</dt>
<dd>- black hot drink</dd>
<dt>Milk</dt>
<dd>- white cold drink</dd>
</dl>
```

How the HTML code above looks in a browser:

Coffee

        - black hot drink

Milk

        - white cold drink

Basic Notes - Useful Tips

Tip: Inside a list item you can put text, line breaks, images, links, other lists, etc.

HTML List Tags

Tag	Description
	Defines an ordered list
	Defines an unordered list
	Defines a list item
<dl>	Defines a definition list
<dt>	Defines an item in a definition list
<dd>	Defines a description of an item in a definition list

HTML Forms and Input

HTML Forms are used to select different kinds of user input.

HTML Forms

HTML forms are used to pass data to a server.
A form can contain input elements like text fields, checkboxes, radio-buttons, submit buttons and more. A form can also contain select lists, textarea, fieldset, legend, and label elements.
The <form> tag is used to create an HTML form:

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

HTML Forms - The Input Element

The most important form element is the input element.
The input element is used to select user information.
An input element can vary in many ways, depending on the type attribute. An input element can be of type text field, checkbox, password, radio button, submit button, and more.
The most used input types are described below.

Text Fields

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

```
<form>
        First name: <input type="text" name="firstname" /><br />
        Last name: <input type="text" name="lastname" />
</form>
```

How the HTML code above looks in a browser:

First name:

Last name:

Note: The form itself is not visible. Also note that the default width of a text field is 20 characters.

Password Field

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

```
<form>
        Password: <input type="password" name="pwd" />
</form>
```

How the HTML code above looks in a browser:

Password:

Note: The characters in a password field are masked (shown as asterisks or circles).

Radio Buttons

<input type="radio" /> defines a radio button. Radio buttons let a user select ONLY ONE of a limited number of choices:

```
<form>
<input type="radio" name="sex" value="male" /> Male<br />
<input type="radio" name="sex" value="female" /> Female
</form>
```

How the HTML code above looks in a browser:

- Male
- Female

Checkboxes

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

```
<form>
<input type="checkbox" name="vehicle" value="Bike" /> I have a bike<br />
<input type="checkbox" name="vehicle" value="Car" /> I have a car
</form>
```

How the HTML code above looks in a browser:

- I have a bike
- I have a car

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 name="input" action="html_form_action.asp" method="get">
Username: <input type="text" name="user" />
<input type="submit" value="Submit" />
</form>
```

How the HTML code above looks in a browser:

Username:

If you type some characters in the text field above, and click the "Submit" button, the browser will send your input to a page called "html_form_action.asp". The page will show you the received input.

HTML Frames

With frames, several Web pages can be displayed in the same browser window.
ATTENTION. Do not expect frames to be supported in future versions of HTML.

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:

- Frames are not expected to be supported in future versions of HTML
- Frames are difficult to use. (Printing the entire page is difficult).
- The web developer must keep track of more HTML documents

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 HTML frame Element

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%,*").

Basic Notes - Useful Tips

Tip: If a frame has visible borders, the user can resize it by dragging the border. To prevent a user from doing this, you can add noresize="noresize" to the <frame> tag.

Note: Add the <noframes> tag for browsers that do not support frames.

Important: You cannot use the <body></body> tags together with the <frameset></frameset> tags! However, if you add a <noframes> tag containing some text for browsers that do not support frames, you will have to enclose the text in <body></body> tags! See how it is done in the first example below.

HTML Frame Tags

Tag	Description
<frameset>	Defines a set of frames
<frame />	Defines a sub window (a frame)
<noframes>	Defines a noframe section for browsers that do not handle frames

HTML Iframes

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

Syntax for adding an iframe:

<iframe src="*URL*"></iframe>

The URL points to the location of the separate page.

Iframe - Set Height and Width

The height and width attributes are used to specify the height and width of the iframe.
The attribute values are specified in pixels by default, but they can also be in percent (like "80%").

Example

```
<iframe src="demo_iframe.htm" width="200" height="200"></iframe>
```

Iframe - Remove the Border

The frameborder attribute specifies whether or not to display a border around the iframe.
Set the attribute value to "0" to remove the border:

Example

```
<iframe src="demo_iframe.htm" frameborder="0"></iframe>
```

Use iframe as a Target for a Link

An iframe can be used as the target frame for a link.
The target attribute of a link must refer to the name attribute of the iframe:

Example

```
<iframe src="demo_iframe.htm" name="iframe_a"></iframe>
<p><a href="http://www.w3schools.com" target="iframe_a">W3Schools.com</a></p>
```

HTML iframe Tag

Tag	Description
<iframe>	Defines an inline sub window (frame)

HTML Colors

Colors are displayed combining RED, GREEN, and BLUE light.

Color Values

HTML colors are defined using a hexadecimal notation (HEX) for the combination of Red, Green, and Blue color values (RGB).
The lowest value that can be given to one of the light sources is 0 (in HEX: 00). The highest value is 255 (in HEX: FF).
HEX values are specified as 3 pairs of two-digit numbers, starting with a # sign.

Color Values

Color	Color HEX	Color RGB
Black	#000000	rgb(0,0,0)
Red	#FF0000	rgb(255,0,0)
Lime	#00FF00	rgb(0,255,0)
blue	#0000FF	rgb(0,0,255)
yellow	#FFFF00	rgb(255,255,0)
Aqua	#00FFFF	rgb(0,255,255)
Megenta/fuchsia	#FF00FF	rgb(255,0,255)
silver	#C0C0C0	rgb(192,192,192)
white	#FFFFFF	rgb(255,255,255)

Advance HTML

HTML Layouts

Web page layout is very important to make your website look good.
Design your webpage layout very carefully

Website Layouts

Most websites have put their content in multiple columns (formatted like a magazine or newspaper).
Multiple columns are created by using <table> or tags. Some CSS are normally also added to position elements, or to create backgrounds or colorful look for the pages.

HTML Layouts - Using Tables

The simplest way of creating layouts is by using the HTML <table> tag.
The following example uses a table with 3 rows and 2 columns - the first and last row spans both columns using the colspan attribute:

Example

```
<html>
        <body>
                <table width="500" border="0">
                        <tr>
                                <td colspan="2" style="background-color:#FFA500;">
                                        <h1>Main Title of Web Page</h1>
```

```
                    </td>
                </tr>
                <tr  valign="top">
                    <td style="background-color:#FFD700;width:100px;text-align:top;">
                        <b>Menu</b><br />
                        HTML<br />
                        CSS<br />
                        JavaScript
                    </td>
                    <td style="background-
                    color:#EEEEEE;height:200px;width:400px;text-align:top;">
                    Content goes here</td>
                </tr>
                <tr>
                    <td colspan="2" style="background-color:#FFA500;text-
                    align:center;">
                    Copyright © 2011 W3Schools.com
                    </td>
                </tr>
            </table>

        </body>
</html>
```

The HTML code above will produce the following result:

Main Title of Web Page

Menu **HTML** **CSS** **JavaScript**	Content goes here

Note: Even though it is possible to create nice layouts with HTML tables, tables were designed for presenting tabular data - NOT as a layout tool!

HTML Layouts - Using Div Elements

The div element is a block level element used for grouping HTML elements.
The following example uses five div elements to create a multiple column layout, creating the same result as in the previous example:

Example

```
<html>
	<body>
		<div id="container" style="width:500px">
		<div id="header" style="background-color:#FFA500;">
			<h1 style="margin-bottom:0;">Main Title of Web Page</h1>
		</div>
		<div id="menu" style="background-
		color:#FFD700;height:200px;width:100px;float:left;">
			<b>Menu</b><br />
			HTML<br />
			CSS<br />
			JavaScript
		</div>
		<div id="content" style="background-
		color:#EEEEEE;height:200px;width:400px;float:left;">
			Content goes here
		</div>
		<div id="footer" style="background-color:#FFA500;clear:both;text-align:center;">
			Copyright © 2011 W3Schools.com
		</div>
		</div>
	</body>
</html>
```

The HTML code above will produce the following result:

Main Title of Web Page

Menu
HTML
CSS
JavaScript

Content goes here

Copyright © 2012 Gaurish.com

HTML Layout - Useful Tips

Tip: The biggest advantage of using CSS is that, if you place the CSS code in an external style sheet, your site becomes MUCH EASIER to maintain. You can change the layout of all your pages by editing one file. To learn more about CSS, study our CSS tutorial.

Tip: Because advanced layouts take time to create, a quicker option is to use a template. Search Google for free website templates (these are pre-built website layouts you can use and customize).

HTML Layout Tags

Tag	Description
<table>	Defines a table
	Defines a section in a document

HTML Doctypes

A doctype declaration refers to the rules for the markup language, so that the browsers render the content correctly.

Example

An HTML document with a doctype of HTML 4.01 Transitional:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
        <head>
                <title>Title of the document</title>
        </head>
        <body>
        The content of the document……
        </body>
</html>
```

HTML Different Doctypes

The doctype declaration is not an HTML tag; it is an instruction to the web browser about what version of the markup language the page is written in.
The doctype declaration refers to a Document Type Definition (DTD). The DTD specifies the rules for the markup language, so that the browsers render the content correctly.
The doctype declaration should be the very first thing in an HTML document, before the <html> tag.

Tip of the day: Always add a doctype to your pages. This helps the browsers to render the page correctly!

HTML 5

HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web.

HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

The new standard incorporates features like video playback and drag-and-drop that have been previously dependent on third-party browser plug-ins such as Adobe Flash, Microsoft Silverlight, and Google Gears.

Browser Support

The latest versions of Apple Safari, Google Chrome, Mozilla Firefox, and Opera all support many HTML5 features and Internet Explorer 9.0 will also have support for some HTML5 functionality.

The mobile web browsers that come pre-installed on iPhones, iPads, and Android phones all have excellent support for HTML5.

New Features

HTML5 introduces a number of new elements and attributes that can help you in building modern websites. Here is a set of some of the most prominent features introduced in HTML5.

- **New Semantic Elements** − These are like <header>, <footer>, and <section>.
- **Forms 2.0** − Improvements to HTML web forms where new attributes have been introduced for <input> tag.
- **Persistent Local Storage** − To achieve without resorting to third-party plugins.
- **WebSocket** − A next-generation bidirectional communication technology for web applications.
- **Server-Sent Events** − HTML5 introduces events which flow from web server to the web browsers and they are called Server-Sent Events (SSE).
- **Canvas** − This supports a two-dimensional drawing surface that you can program with JavaScript.
- **Audio & Video** − You can embed audio or video on your webpages without resorting to third-party plugins.
- **Geolocation** − Now visitors can choose to share their physical location with your web application.
- **Microdata** − This lets you create your own vocabularies beyond HTML5 and extend your web pages with custom semantics.
- **Drag and drop** − Drag and drop the items from one location to another location on the same webpage.

Backward Compatibility

HTML5 is designed, as much as possible, to be backward compatible with existing web browsers. Its new features have been built on existing features and allow you to provide fallback content for older browsers.

It is suggested to detect support for individual HTML5 features using a few lines of JavaScript.

If you are not familiar with any previous version of HTML, I would recommend that you go through our **HTML Tutorial** before exploring the features of HTML5.

The DOCTYPE

DOCTYPEs in older versions of HTML were longer because the HTML language was SGML based and therefore required a reference to a DTD.

HTML 5 authors would use simple syntax to specify DOCTYPE as follows −

```
<!DOCTYPE html>
```

The above syntax is case-insensitive.

How to Use Styles

When a browser reads a style sheet, it will format the document according to it.
There are three ways of inserting a style sheet:

- External style sheet
- Internal style sheet
- Inline styles

External Style Sheet

An external style sheet is ideal when the style is applied to many pages. With an external style sheet, you can change the look of an entire Web site by changing one file. Each page must link to the style sheet using the <link> tag. The <link> tag goes inside the <head> section:

```
<head>
        <link rel="stylesheet" type="text/css" href="mystyle.css" />
</head>
```

Internal Style Sheet

An internal style sheet can be used if one single document has a unique style. Internal styles are defined in the <head> section of an HTML page, by using the <style> tag, like this:

```
<head>
        <style type="text/css">
                body {background-color:yellow;}
                p {color:blue;}
        </style>
</head>
```

Inline Styles

An inline style can be used if a unique style is to be applied to one single occurrence of an element.
To use inline styles, use the style attribute in the relevant tag. The style attribute can contain any CSS property. The example below shows how to change the text color and the left margin of a paragraph:

```
<p style="color:blue;margin-left:20px;">This is a paragraph.</p>
```

HTML Style Tags

Tag	Description
<style>	Defines style information for a document
<link />	Defines the relationship between a document and an external resource

HTML head Elements

The HTML head Element

The head element is a container for all the head elements. Elements inside <head> can include scripts, instruct the browser where to find style sheets, provide meta information, and more.
The following tags can be added to the head section: <title>, <base>, <link>, <meta>, <script>, and <style>.

The HTML title Element

The <title> tag defines the title of the document.
The title element is required in all HTML/XHTML documents.
The title element:

- defines a title in the browser toolbar
- provides a title for the page when it is added to favorites
- displays a title for the page in search-engine results

A simplified HTML document:

```
<html>
<head>
<title>Title of the document</title>
</head>

<body>
The content of the document……
</body>

</html>
```

The HTML base Element

The <base> tag specifies a default address or a default target for all links on a page:

```
<head>
        <base href="http://www.w3schools.com/images/" />
        <base target="_blank" />
</head>
```

The HTML link Element

The <link> tag defines the relationship between a document and an external resource.
The <link> tag is most used to link to style sheets:

```
<head>
        <link rel="stylesheet" type="text/css" href="mystyle.css" />
</head>
```

The HTML style Element

The <style> tag is used to define style information for an HTML document.
Inside the style element you specify how HTML elements should render in a browser:

```
<head>
        <style type="text/css">
                body {background-color:yellow}
                p {color:blue}
        </style>
</head>
```

The HTML meta Element

The <meta> tag provides metadata about the HTML document.
The meta element will be explained in the next chapter.

The HTML script Element

The <script> tag is used to define a client-side script, such as a JavaScript.
The script element will be explained in a later chapter.

HTML head Elements

Tag	Description
<head>	Defines information about the document
<title>	Defines the title of a document
<base />	Defines a default address or a default target for all links on a page
<link />	Defines the relationship between a document and an external resource
<meta />	Defines metadata about an HTML document
<script>	Defines a client-side script
<style>	Defines style information for a document

HTML Meta

The HTML meta Element

Metadata is information about data.
The <meta> tag provides metadata about the HTML document. Metadata will not be displayed on the page, but will be machine parsable.
Meta elements are typically used to specify page description, keywords, author of the document, last modified, and other metadata.
The <meta> tag always goes inside the head element.
The metadata can be used by browsers (how to display content or reload page), search engines (keywords), or other web services.

Keywords for Search Engines

Some search engines will use the name and content attributes of the meta element to index your pages.
The following meta element defines a description of a page:

```
<meta name="description" content="Free Web tutorials on HTML, CSS, XML" />
```

The following meta element defines keywords for a page:

```
<meta name="keywords" content="HTML, CSS, XML" />
```

The intention of the name and content attributes is to describe the content of a page.

Note: A lot of webmasters have used <meta> tags for spamming, like repeating keywords (or using wrong keywords) for higher ranking. Therefore, most search engines have stopped using <meta> tags to index/rank pages.

HTML Scripts

The HTML script Element

The <script> tag is used to define a client-side script, such as a JavaScript.
The script element either contains scripting statements or it points to an external script file through the src attribute.
The required type attribute specifies the MIME type of the script.

Common uses for JavaScript are image manipulation, form validation, and dynamic changes of content.
The script below writes Hello World! to the HTML output:

Example

```
<script type="text/javascript">
document.write("Hello World!")
</script>
```

The HTML noscript Element

The <noscript> tag is used to provide an alternate content for users that have disabled scripts in their browser or have a browser that doesn't support client-side scripting.
The noscript element can contain all the elements that you can find inside the body element of a normal HTML page.
The content inside the noscript element will only be displayed if scripts are not supported, or are disabled in the user's browser:

Example

```
<script type="text/javascript">
document.write("Hello World!")
</script>
<noscript>Sorry, your browser does not support JavaScript!</noscript>
```

HTML Script Tags

Tag	Description
<script>	Defines a client-side script
<noscript>	Defines an alternate content for users that do not support client-side scripts

Examples

Define the character set used:

```
<meta charset="UTF-8">
```

Define keywords for search engines:

```
<meta name="keywords" content="HTML, CSS, JavaScript">
```

Define a description of your web page:

```
<meta name="description" content="Free Web tutorials">
```

Define the author of a page:

```
<meta name="author" content="John Doe">
```

Refresh document every 30 seconds:

```
<meta http-equiv="refresh" content="30">
```

Setting the viewport to make your website look good on all devices:

<meta name="viewport" content="width=device-width, initial-scale=1.0">

Example of <meta> tags:

<meta charset="UTF-8">
<meta name="description" content="Free Web tutorials">
<meta name="keywords" content="HTML, CSS, JavaScript">
<meta name="author" content="John Doe">

Setting the Viewport

The viewport is the user's visible area of a web page. It varies with the device - it will be smaller on a mobile phone than on a computer screen.

You should include the following <meta> element in all your web pages:

<meta name="viewport" content="width=device-width, initial-scale=1.0">

This gives the browser instructions on how to control the page's dimensions and scaling.

The width=device-width part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).

The initial-scale=1.0 part sets the initial zoom level when the page is first loaded by the browser.

Here is an example of a web page *without* the viewport meta tag, and the same web page *with* the viewport meta tag:

HTML Entities

Reserved characters in HTML must be replaced with character entities.

HTML Entities

Some characters are reserved in HTML.
It is not possible to use the less than (<) or greater than (>) signs in your text, because the browser will mix them with tags.
To actually display reserved characters, we must use character entities in the HTML source code.
A character entity looks like this:
&entity_name;
OR
&#entity_number;
To display a less than sign we must write: **<** or **<**

Tip of the day: The advantage of using an entity name, instead of a number, is that the name is easier to remember. However, the disadvantage is that browsers may not support all entity names (the support for entity numbers is very good).

Non-breaking Space

A common character entity used in HTML is the non-breaking space ( ).
Browsers will always truncate spaces in HTML pages. If you write 10 spaces in your text, the browser will remove 9 of them, before displaying the page. To add spaces to your text, you can use the   character entity.

HTML Useful Character Entities

Note: Entity names are case sensitive!

Result	Description	Entity Name	Entity Number
	non-breaking space		
<	less than	&lt;	&#60;
>	greater than	&gt;	&#62;
&	ampersand	&amp;	&#38;
¢	cent	&cent;	&#162;
£	pound	&pound;	&#163;
¥	yen	&yen;	&#165;
€	euro	&euro;	&#8364;
§	section	&sect;	&#167;
©	copyright	&copy;	&#169;
®	registered trademark	&reg;	&#174;
™	trademark	&trade;	&#8482;

HTML Uniform Resource Locators

A URL is another word for a web address.
A URL can be composed of words, such as "w3schools.com", or an Internet Protocol (IP) address: 192.68.20.50. Most people enter the name of the website when surfing, because names are easier to remember than numbers.

URL - Uniform Resource Locator

When you click on a link in an HTML page, an underlying <a> tag points to an address on the world wide web.
A Uniform Resource Locator (URL) is used to address a document (or other data) on the world wide web.
A web address, like this: http://www.w3schools.com/html/default.asp follows these syntax rules:
scheme : / / host.domain : port / path / filename

Explanation:

- **scheme** - defines the **type** of Internet service. The most common type is **http**
- **host** - defines the **domain host** (the default host for http is **www**)
- **domain** - defines the Internet **domain name**, like w3schools.com
- **:port** - defines the **port number** at the host (the default port number for http is **80**)
- **path** - defines a **path** at the server (If omitted, the document must be stored at the root directory of the web site)
- **filename** - defines the name of a document/resource

Common URL Schemes

The table below lists some common schemes:

Scheme	Short for....	Which pages will the scheme be used for...
http	HyperText Transfer Protocol	Common web pages starts with http://. Not encrypted
https	Secure HyperText Transfer Protocol	Secure web pages. All information exchanged are encrypted
ftp	File Transfer Protocol	For downloading or uploading files to a website. Useful for domain maintenance
file		A file on your computer

HTML URL Encoding

URL encoding converts characters into a format that can be transmitted over the Internet.

URL - Uniform Resource Locator

Web browsers request pages from web servers by using a URL.
The URL is the address of a web page, like: **http://www.w3schools.com.**

URL Encoding

URLs can only be sent over the Internet using the ASCII character-set.
Since URLs often contain characters outside the ASCII set, the URL has to be converted into a valid ASCII format.
URL encoding replaces non ASCII characters with a "%" followed by two hexadecimal digits.
URLs cannot contain spaces. URL encoding normally replaces a space with a + sign.

Try It Yourself

If you click the "Submit" button below, the browser will URL encode the input before it is sent to the server. A page at the server will display the received input.

Hello Günter

Try some other input and click Submit again.

URL Encoding Examples

Character	URL-encoding

€	%80
£	%A3
©	%A9
®	%AE
À	%C0
Á	%C1
Â	%C2
Ã	%C3
Ä	%C4
Å	%C5

HTML Form Elements

Tag	Description
<form>	Defines an HTML form for user input
<input>	Defines an input control
<textarea>	Defines a multiline input control (text area)
<label>	Defines a label for an <input> element
<fieldset>	Groups related elements in a form
<legend>	Defines a caption for a <fieldset> element
<select>	Defines a drop-down list
<optgroup>	Defines a group of related options in a drop-down list
<option>	Defines an option in a drop-down list
<button>	Defines a clickable button
<datalist>	Specifies a list of pre-defined options for input controls

```
<form action="/action_page.php">
  <fieldset>
    <legend>Personalia:</legend>
    <label for="fname">First name:</label>
    <input type="text" id="fname" name="fname"><br><br>
    <label for="lname">Last name:</label>
    <input type="text" id="lname" name="lname"><br><br>
    <label for="email">Email:</label>
    <input type="email" id="email" name="email"><br><br>
    <label for="birthday">Birthday:</label>
    <input type="date" id="birthday" name="birthday"><br><br>
     <label for="cars">Choose a car:</label>
        <select name="cars" id="cars">
            <optgroup label="Swedish Cars">
                 <option value="volvo">Volvo</option>
                <option value="saab">Saab</option>
            </optgroup>
            <optgroup label="German Cars">
                <option value="mercedes">Mercedes</option>
                <option value="audi">Audi</option>
          </optgroup>
         </select>
        <input type="submit" value="Submit">
      </fieldset>
    </form>
```

Tip: The <legend> tag is used to define a caption for the <fieldset> element.

Definition and Usage
The **<input>** tag specifies an input field where the user can enter data.
The **<input>** element is the most important form element.
The **<input>** element can be displayed in several ways, depending on the type attribute.

The different input types are as follows:

- <input type="button">
- <input type="checkbox">
- <input type="date">
- <input type="email">
- <input type="file">
- <input type="hidden">
- <input type="image">
- <input type="month">
- <input type="number">
- <input type="password">
- <input type="radio">
- <input type="range">
- <input type="reset">
- <input type="search">
- <input type="submit">

- <input type="text"> **(default value)**
- <input type="time">

HTML Multimedia

Multimedia comes in many different formats. It can be almost anything you can hear or see, like images, music, sound, videos, records, films, animations, and more.
Web pages often contain multimedia elements of different types and formats.

Multimedia Formats

Multimedia elements (like audio or video) are stored in media files.
The most common way to discover the type of a file, is to look at the file extension.
Multimedia files have formats and different extensions like: .wav, .mp3, .mp4, .mpg, .wmv, and .avi.
Note: Only MP4, WebM, and Ogg video are supported by the HTML standard.
Example 1:- With Attribute control

```
<!DOCTYPE html>
<html>
<body>
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
Your browser does not support the video tag.
</video>
</body>
</html>
```

Example 2: With attribute autoplay and muted

```
<!DOCTYPE html>
<html>
<body>
<video width="320" height="240" autoplay muted>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
</body>
</html>
```

Common Audio Formats

MP3 is the best format for compressed recorded music. The term MP3 has become synonymous with digital music.

If your website is about recorded music, MP3 is the choice.

Note: Only MP3, WAV, and Ogg audio are supported by the HTML standard.

Example : With attribute controls autoplay and muted

```
<!DOCTYPE html>
<html>
<body>

<audio controls autoplay muted>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
Your browser does not support the audio element.
</audio>
```

```
</body>
</html>
```

HTML Plug-ins

Plug-ins are computer programs that extend the standard functionality of the browser.

The <object> Element

The <object> element is supported by all browsers.
The <object> element defines an embedded object within an HTML document.
It was designed to embed plug-ins (like Java applets, PDF readers, and Flash Players) in web pages, but can also be used to include HTML in HTML:

```
<object width="100%" height="500px" data="snippet.html"></object>
<object data="audi.jpeg"></object>
```

The <embed> Element

The <embed> element is supported in all major browsers.
The <embed> element also defines an embedded object within an HTML document.
Web browsers have supported the <embed> element for a long time. However, it has not been a part of the HTML specification before HTML5.

```
<embed src="audi.jpeg">
<embed width="100%" height="500px" src="snippet.html">
```

Note : that the <embed> element does not have a closing tag. It can not contain alternative text.

How to validate your Code of HTML.

1. Open Your web Browser and type this Url : https://validator.w3.org/
2. There you will find 3 option you can select one of them and check your code.