

</> HTML

O. A. MARCUS

H

<>

—

—

—

—

—

—

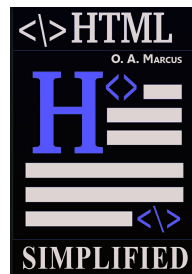
</>

SIMPLIFIED

# HTML SIMPLIFIED

Simplified And Turned To Fun

O. A. Marcus



Copyright © 2021 O. A. Marcus

All rights reserved

No part of this book may be reproduced, or stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without express written permission of the publisher.

ISBN: 9798594005655

Cover design by: Niflare

# CONTENTS

[Title Page](#)

[Copyright](#)

[Introduction to HTML](#)

[HTML tags](#)

[HTML attributes](#)

[The basic attributes](#)

[The style attribute](#)

[The Style element](#)

[Creating an input field](#)

[Creating a Form](#)

[Creating tables in HTML](#)

[Creating list using HTML](#)

[HTML 5](#)

[New element to HTML](#)

[Header element](#)

[Footer of a Webpage](#)

[The Generic Section](#)

[Streaming the content](#)

[The Side element](#)

[Multimedia element](#)

[The Embed element](#)

[New forms to form element](#)  
[Exciting feature in HTML5](#)

# INTRODUCTION TO HTML

Firstly what is the full meaning of HTML?

**F** **H** means **HYPER**

**T** means **TEXT**

**M** means **MARK-UP**

**L** means **LANGUAGE**

So therefore **HTML** simply means **Hyper Text Mark-up Language**

HTML is used to create ELECTRONIC DOCUMENTS which is also known as web pages, these are the things we see on WWW (WORLDWIDE WEB).

It is commonly used to create website.

HTML is not a programming language but a text file that is being written by a simple text editor, and it controls the presentation, structure and layout of the data on a webpage. HTML does not contain any functional or programming logic.

HTML consists of 2 important parts which are:

1. Document Creation: this is what the user will see.
2. Tags: this is how they see it.

HTML comprises of Tags which are:

**Tag <html>:** this tag <html> is the head of the family. It's from this tag that the browser recognizes, it's an HTML document. In this tag, <html> serves as the opening tag while </html> serves as the closing tag, so therefore for every opening tag <html> there is a closing tag </html>

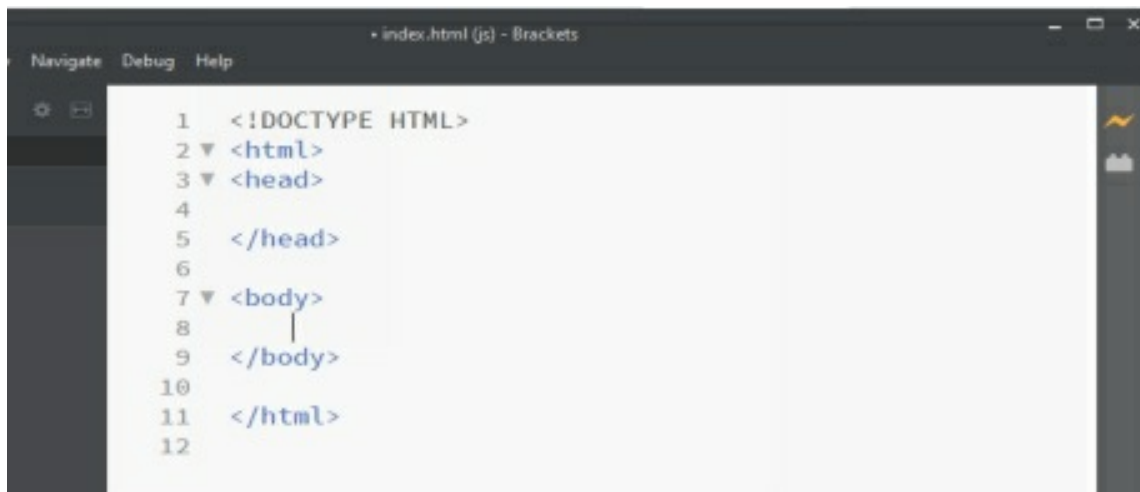


```
1 <!DOCTYPE HTML>
2 <html> </html>
3 |
```



```
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 |
5 </head>
6
7 </html>
8
```

**Tag <head>:** is a tag that comes after tag <html>. Tag <head> is a special tag, in such a way that the content written inside this tag will not display on the webpage. This tag comprises of the title (of the browser window) or any other instructions, which you want to provide to the web browser.



```
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4
5 </head>
6
7 <body>
8 |
9 </body>
10
11 </html>
12
```

**Tag <body>:** this tag also comes after tag <head>, this is where the actual content

that is visible on the webpage is written i.e. all the visual elements such as headings, paragraphs, images etc.

## What Is A Webpage?

Accessing information from the internet is being done by the use of any browser. The browser gets all the information asked for from the webserver, these information are stored as HTML document inside the webserver.

A webpage is the result of a team working together.

A **Web Server** can be hardware/ software that let you deliver the webpage to your browser.

A browser reads and decodes the instructions, display exciting content on the screen.

With all these above, we just realised, that's what a web page is.

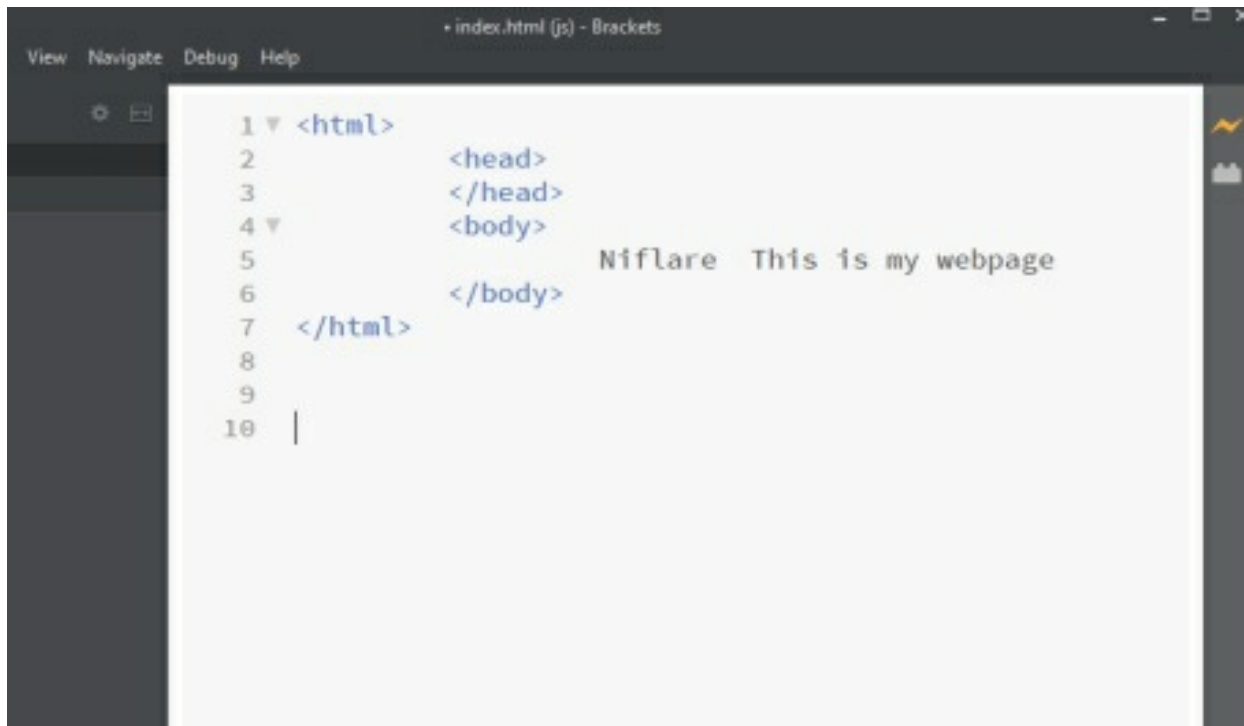
---

## To Create A Webpage

A simple text editor will be needed like Notepad++, TextEdit, Bracket etc.

After getting any text editor then try and revisit your mind palace concerning the basic html structure that you have learnt, the codes uses the same structure. Here is an example





```
1 <html>
2     <head>
3     </head>
4     <body>
5         Niflare This is my webpage
6     </body>
7 </html>
8
9
10 |
```

After writing the html codes then save it as .html or .htm file. Then try opening the file in your web browser then this will be the outcome. We made use of Bracket as our simple text editor with a browser.



So therefore the page at the left is our simple text editor which requires the html codes while the page at the right side is the outcome of the html codes that shows on the browser. Here is your first page above. In conclusion about writing of codes, the tags were written using lower case, so correct capitalization should be used throughout the code. All tags were closed in the proper order, if they were not closed properly the code won't

work.

**Indentation** is important also, it made the codes easy to understand.

**Indentation** can be defined an act of providing space from margin, to provide structure to the code.

# HTML TAGS

**W**e will be talking about the backbone of the HTML language which is the TAG, we have different types of tags that we have made mentioned when introducing HTML. Like we made mentioned we have:

`<html>...</html>` which is also known as ROOT ELEMENT. All Html web pages starts with this root element.

A screenshot of a code editor window titled "index.html (js) - Brackets". The editor shows three lines of code: line 1: `<!DOCTYPE HTML>`, line 2: `<html> </html>`, and line 3: a vertical cursor. The editor has a dark theme and includes a menu bar with "Navigate", "Debug", and "Help".

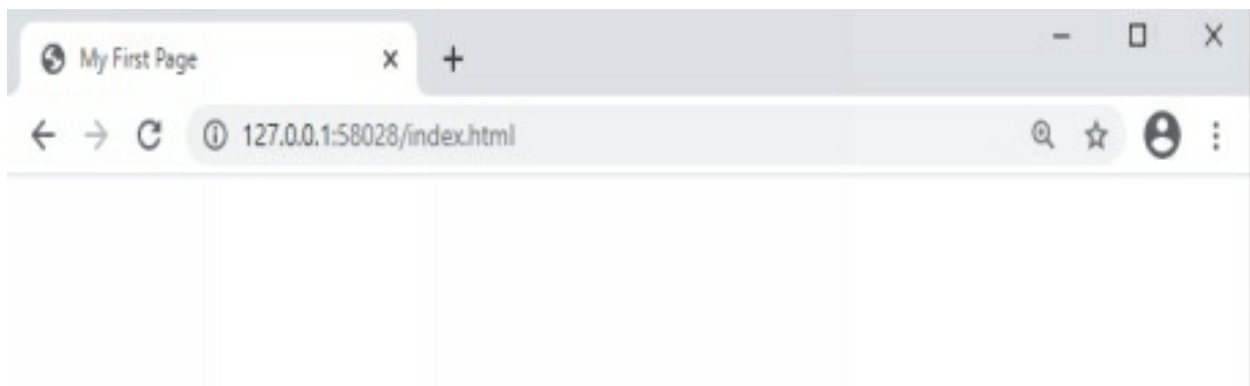
```
1 <!DOCTYPE HTML>
2 <html> </html>
3 |
```

`<title>...</title>` which is also known as the TITLE PAGE.



```
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4   <title>My First Page</title>
5 </head>
6
7 |
8 </html>
9
```

Result:



`<body>....</body>` which is also known as DOCUMENT HEAD. In this tag we also have some other tags in it such as `<p>` `<br>` etc.

---

## The Paragraph Tag

In the `<body>` tag, the tag paragraph is important when making a webpage.

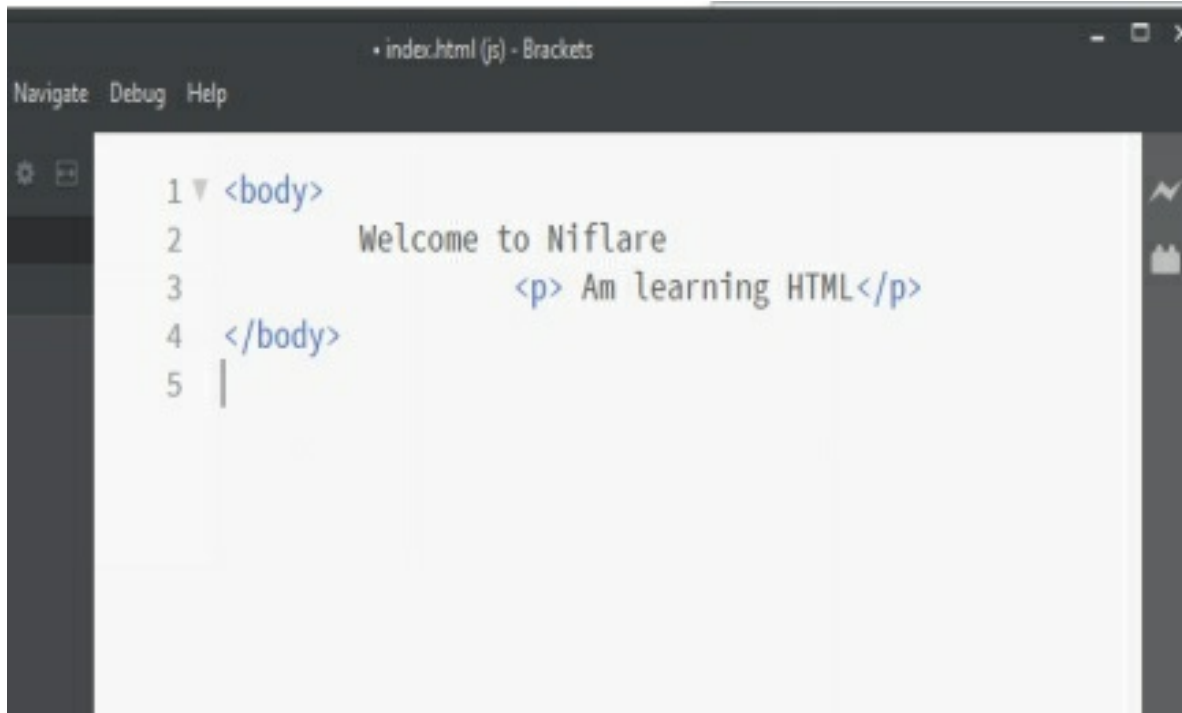
**Denotation** of paragraph is `<p>` and `</p>`.

**Location:** It's placed inside the body tag.

**Usage:** It lets you make new paragraphs.

**Visibility:** The text placed inside this tag will be visible on the main webpage.

Here is an example to take you through:



```
1 <body>
2     Welcome to Niflare
3         <p> Am learning HTML</p>
4 </body>
5 |
```

Note: Putting one tag inside another is termed as **Nesting**

Then when we open this above code in our web browser it looks like this:



---

## The Breaking Tag (<Br>)

Writing sentences in a new line provides symmetry to the writing. The **break** tag lets you start a sentence from a new line in HTML.

**Denotation:** <br>

**Location:** It is placed inside the body tag.

**Usage:** Lets you start from a new line.

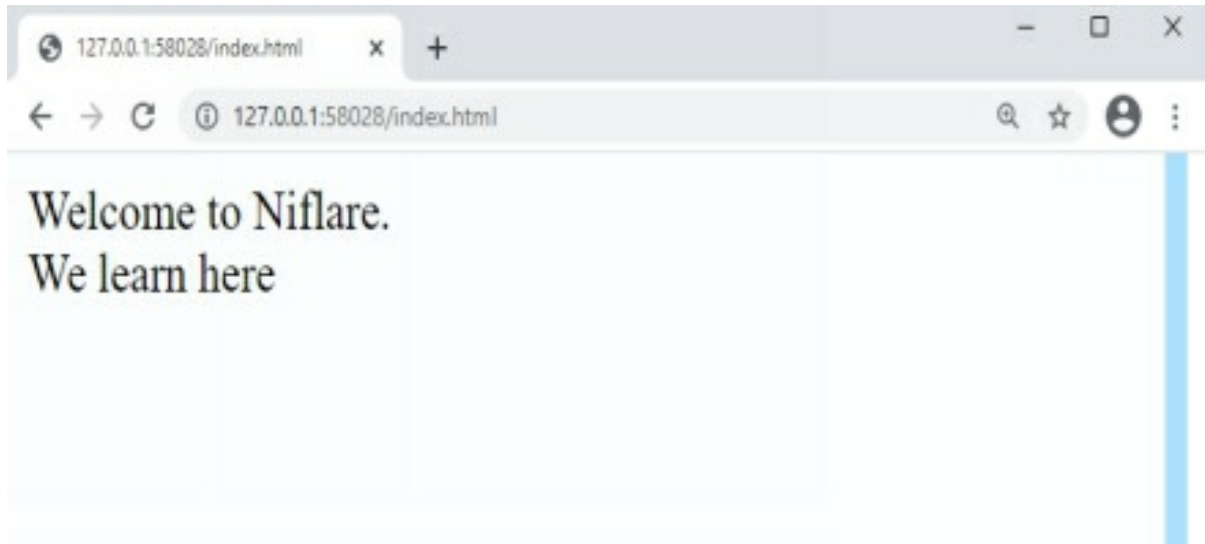
**Special Characteristic:** It's a **self-closing** HTML Tag.

Self-closing HTML Tag is when the closing and opening are included under same tag. Not all tags can be used in such a manner. It is identified by the forward slash towards the end.

Here is an example with the outcome on the browser:



```
index.html (js) - Brackets
Navigate Debug Help
1 <body>
2     <p> Welcome to Niflare.<br>We learn
   here</p>
3 </body>
4 |
```



As we can see the tag `<br>` breaks the sentence to another line and also the tag doesn't have closing tag. For better understanding with the above code, trying to display it without using `<br>`, it would all be in the same line.

---

## Bold, Italics Underline

There are different tags present to format the content on a web page. Let's learn about them.

### The Bold Tag

**Denotation:** `<b>` `</b>`

**Location:** It's placed inside the body tag.

**Usage:** Let's you mark the text **bold** to showcase importance.

### The Italic Tag

**Denotation:** `<i>` `</i>`

**Location:** It's placed inside the body tag.

**Usage:** Let's you mark the text in *italic* font.

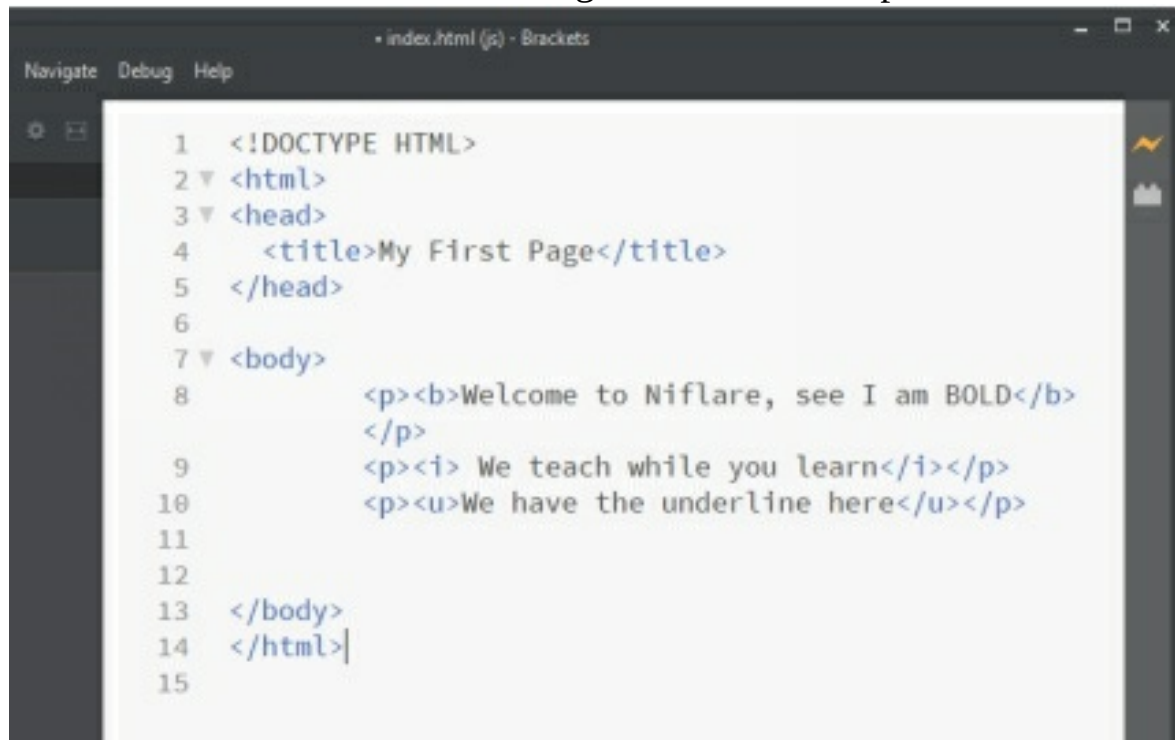
# The Underline Tag

**Denotation:** <u> </u>

**Location:** It's placed inside the body tag.

**Usage:** Just in case you want to **underline** a text.

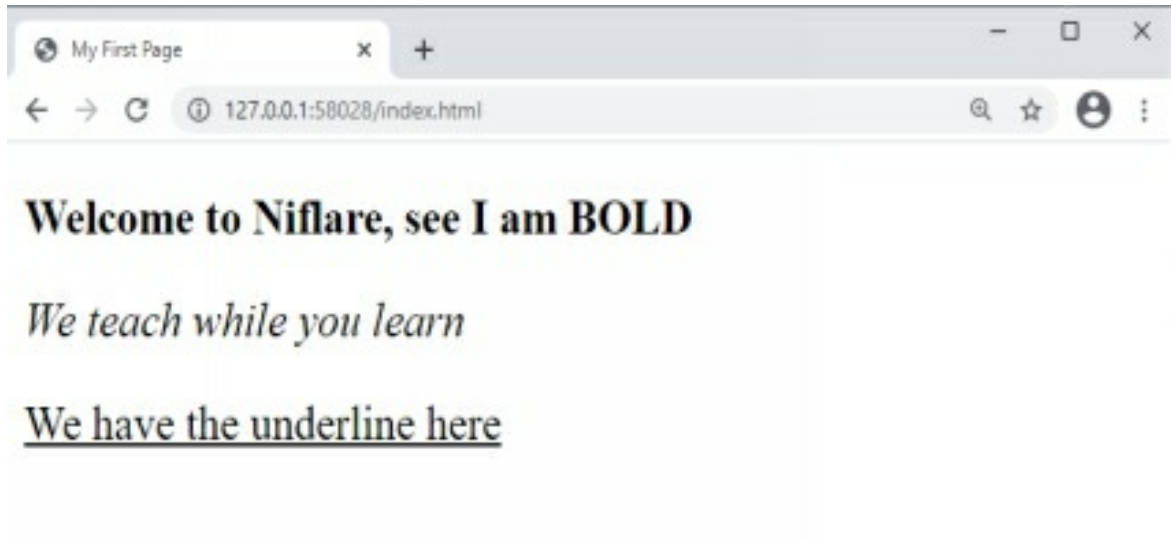
Now it time to make use of these tags, here is an example:

A screenshot of a code editor window titled "index.html (p) - Brackets". The editor shows the following HTML code:

```
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4   <title>My First Page</title>
5 </head>
6
7 <body>
8   <p><b>Welcome to Niflare, see I am BOLD</b>
9   </p>
10  <p><i> We teach while you learn</i></p>
11  <p><u>We have the underline here</u></p>
12
13 </body>
14 </html>|
15
```

In this above code, we have included everything in a separate paragraph. Once you try running the code, the content will automatically change, see below:





## Headings In HTML

We tend to prioritize the important stuff. In HTML we make use of headings.

**Denotation:** `<h1>....</h1>`, `<h2>.....</h2>`, `<h3>.....</h3>`, `<h4>.....</h4>`, `<h5>.....</h5>`, `<h6>.....</h6>`

**Location:** It's placed inside the body tag.

**Usage:** `<h1>` to `<h6>`- arranged on basis of priority (`<h1>`- highest)

**Visibility:** The text placed inside this tag will be visible on the main webpage.

Here is an example for pellucidity:

```
index.html (js) - Brackets
Navigate Debug Help
1 <body>
2     <h1>Welcome to Niflare</h1>
3     <h2>Welcome to Niflare</h2>
4     <h3>Welcome to Niflare</h3>
5     <h4>Welcome to Niflare</h4>
6     <h5>Welcome to Niflare</h5>
7     <h6>Welcome to Niflare</h6>
8 </body>
9
10 |
```

And here is the result of the above code:



Isn't it amazing how you can sort the important stuff from the content, just by providing a suitable heading structure?

## The Image Tag

Have you ever made a picture collage? It does look good right? So we can also make our webpages more interactive by adding images to it using `<img>` tag.

**Denotation:** `<img/>`

**Location:** It's placed inside the body tag.

**Usage:** Let's help you add an image to the page.

The tag is used like this:

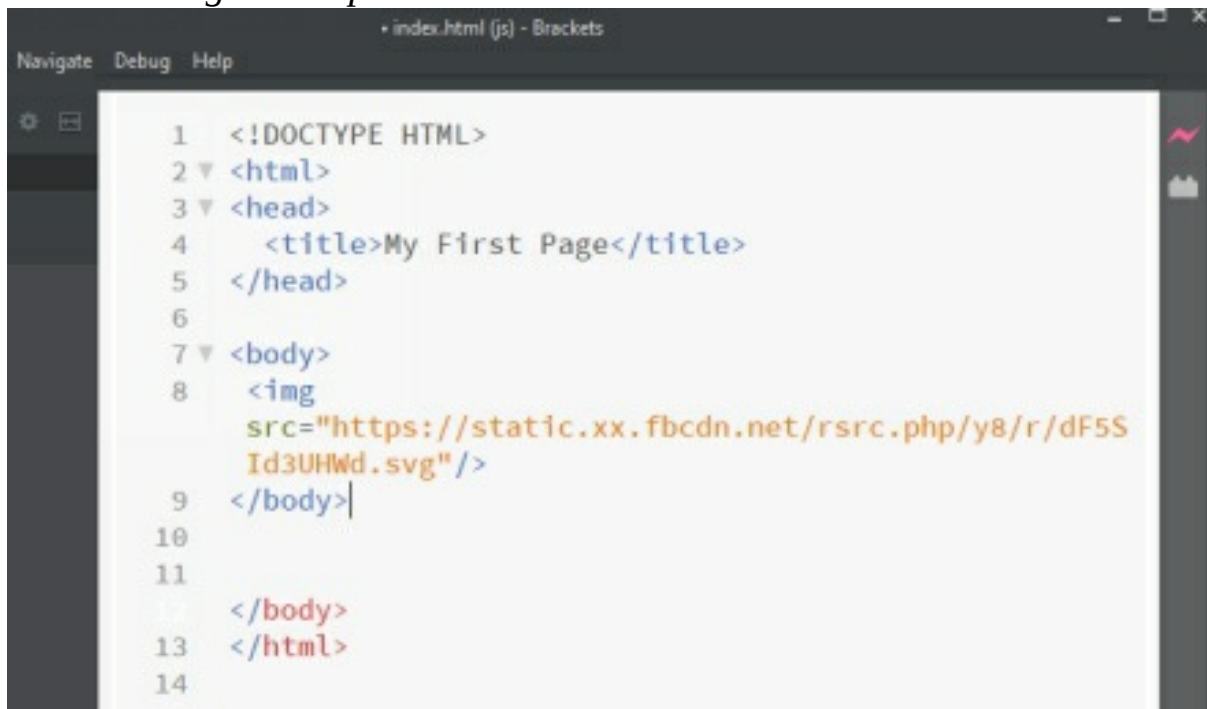
`<img src=""image's path'"/>`

**Src** is an attribute to include the image's path

An attribute Provides additional information about your html tag.

**Note:** the image's **path** can also be an **image URL**.

*Facebook logo url is passed to src attribute*



```
index.html (js) - Brackets
Navigate Debug Help
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4   <title>My First Page</title>
5 </head>
6
7 <body>
8   
11
12 </body>
13 </html>
14
```



---

## The Division Tag

A simple straight webpage, will it look good and be appealing? Of course not.

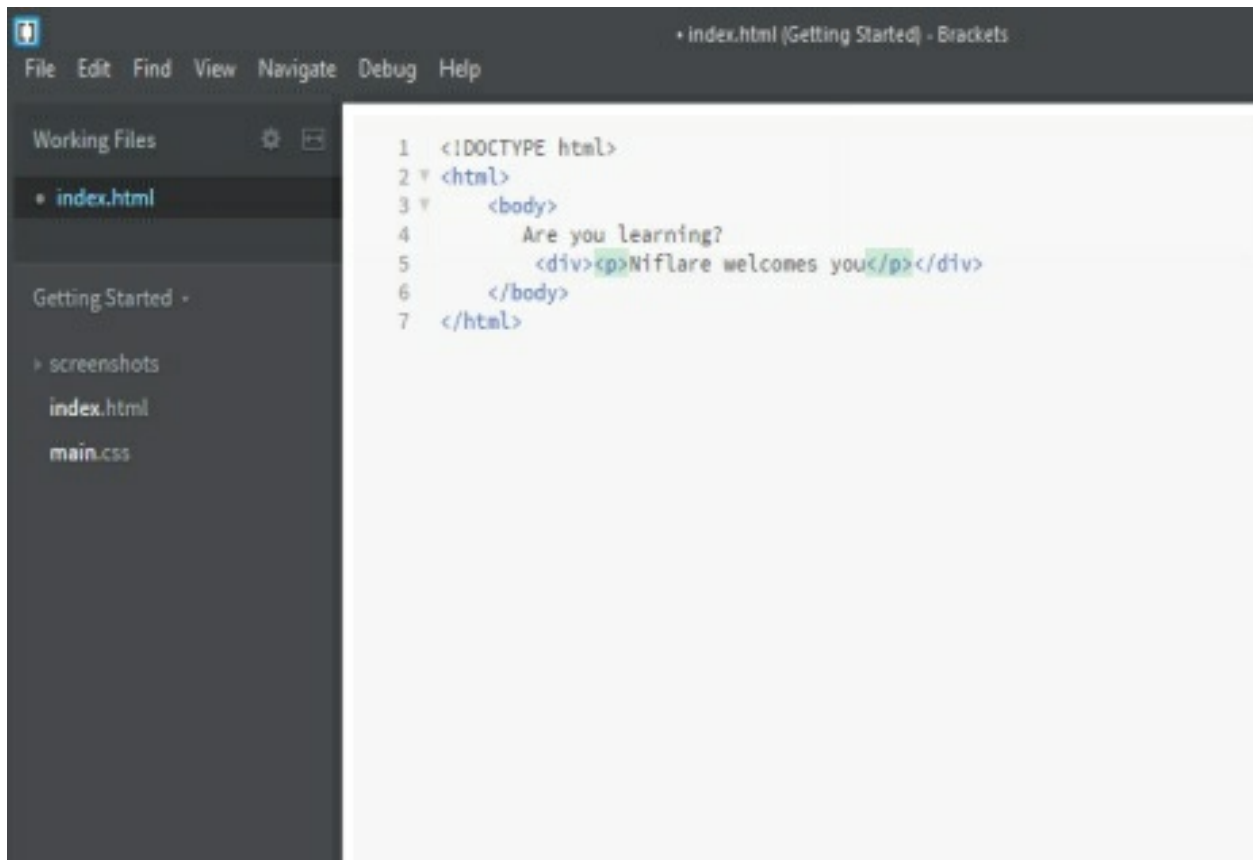
Hence, the div tag comes into the picture

**Denotation:** `<div> </div>`

**Location:** It's placed inside the body tag.

**Usage:** Let's get you to divide the page into sections, especially with CSS. It will not have any visible output now.

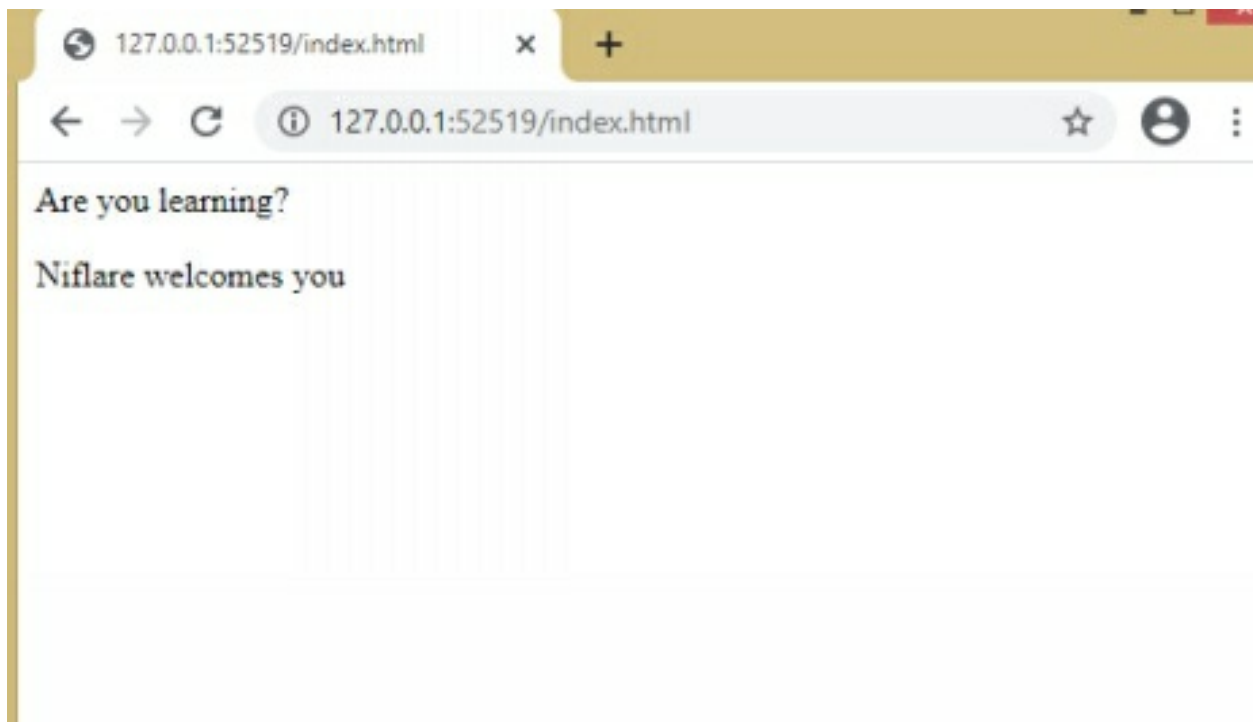
Let's give it a try:



```
1 <!DOCTYPE html>
2 <html>
3 <body>
4     Are you learning?
5     <div><p>Niflare welcomes you</p></div>
6 </body>
7 </html>
```

The screenshot shows the Brackets code editor interface. The top menu bar includes File, Edit, Find, View, Navigate, Debug, and Help. The left sidebar shows the 'Working Files' panel with 'index.html' selected. The main editor area displays the HTML code for 'index.html' with line numbers 1 through 7. The code includes a DOCTYPE declaration, an html root element, a body element containing the text 'Are you learning?' and a div containing a paragraph 'Niflare welcomes you'.

Then here is the output:



---

# The Title Tag

Just like the title of MR. or MRS. Adds meaning to your name, similarly you might have noticed the text in your browser window? That's the use of title tag.

**Denotation:** <title> </title>

**Location:** It's placed inside the head tag.

**Visibility:** Not on the main page.

**Usage:** Lets you provide special instructions to the browser.

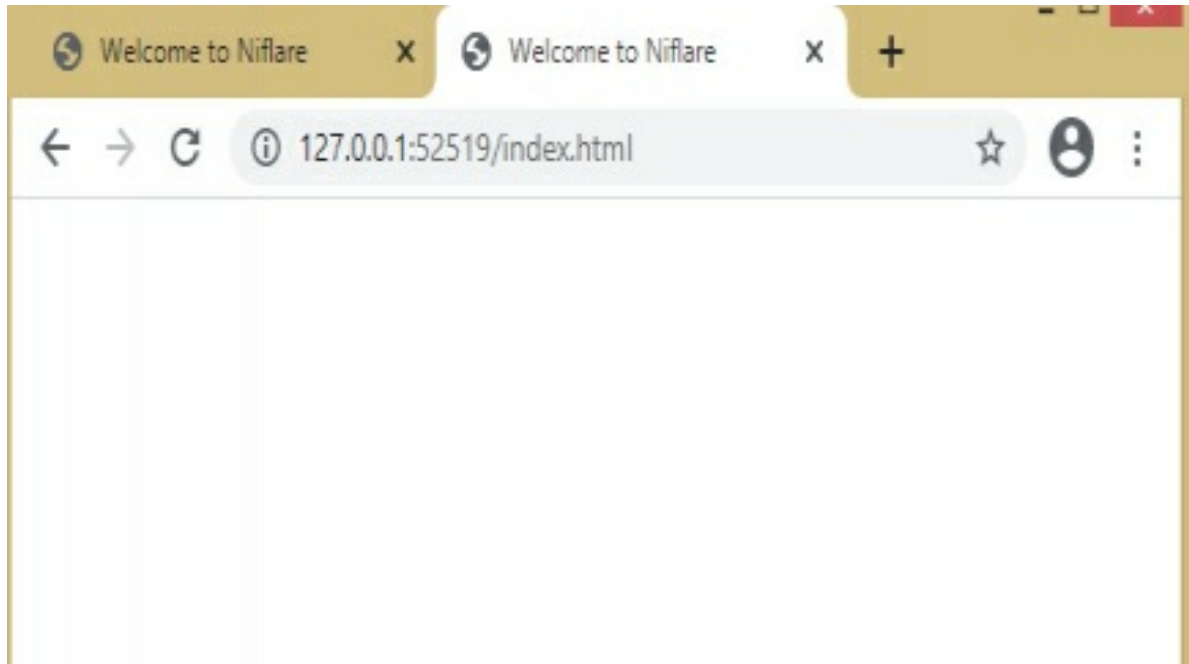
Want to see how it works? Try doing this on your simple text editor:

Here is the code:

A screenshot of a code editor window titled "index.html (Getting Started) - Brackets". The editor displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3   <head><title>Welcome to Wflare</title></head>
4   <body>
5     </body>
6 </html>
```

The left sidebar shows a "Working Files" panel with "index.html" selected, and a "Getting Started" panel with "screenshots", "index.html", and "main.css" listed.



Here is the result, check your browser window, you will be able to see the text written inside `<title>` tag. So therefore `<title>` tag is very helpful for search engines.

---

## The Anchor Tag

Imagine, if we had a connecting door to the kitchen, directly from our room, wouldn't that be fun? Similarly, to embed other web pages' links to your web page, an anchor tag is used.

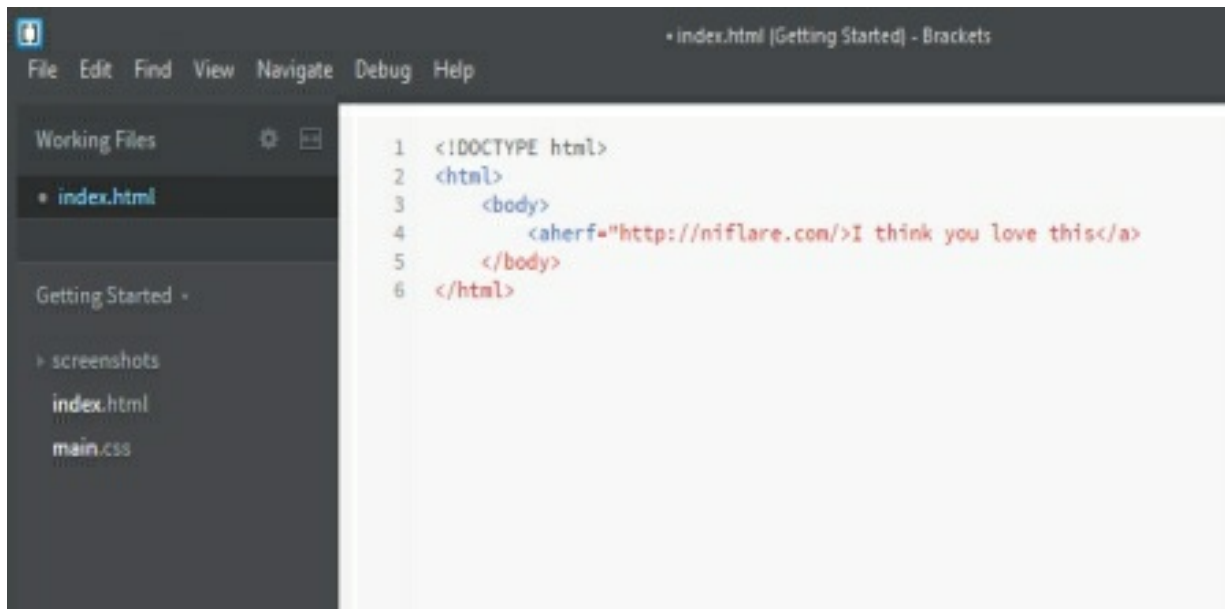
**Denotation:** `<a>` `</a>`

**Location:** It's placed inside the body tag.

**Usage:** Linking between web pages.

Here is how you can use it:

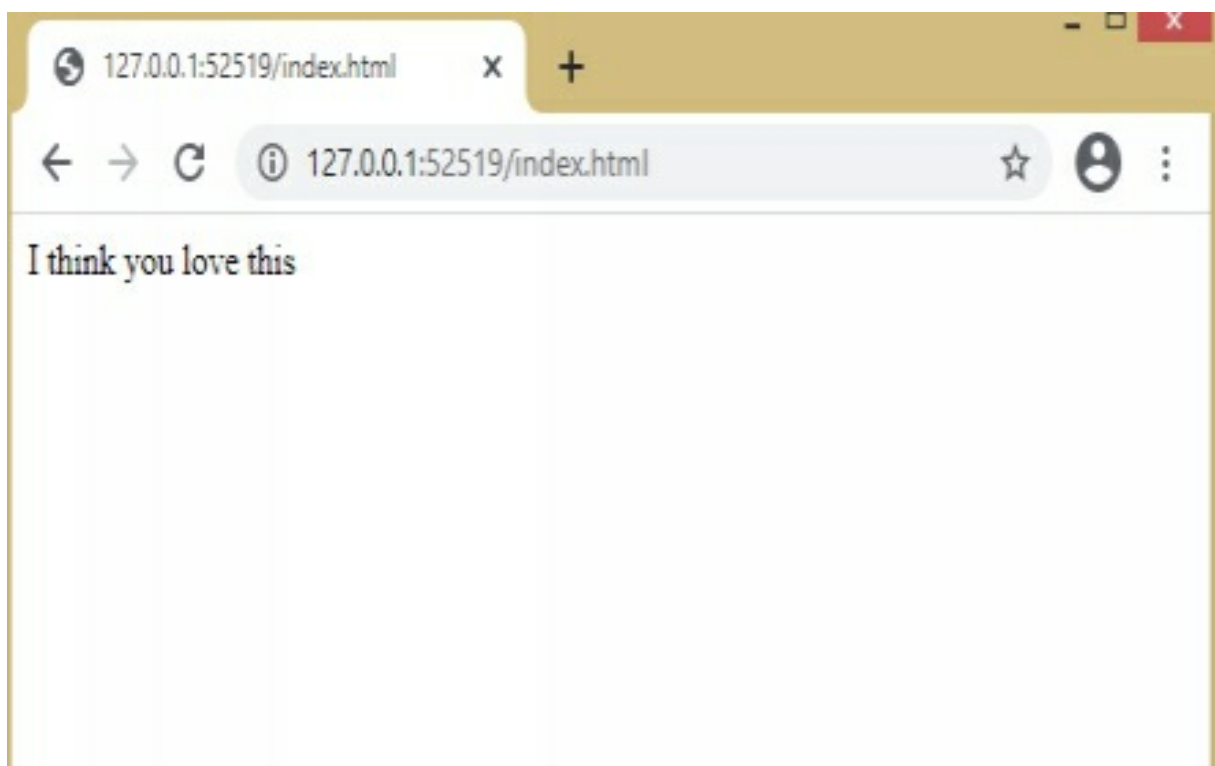
Herf- is another attribute used to define the link.



The screenshot shows the Brackets code editor interface. The title bar reads "index.html [Getting Started] - Brackets". The menu bar includes "File", "Edit", "Find", "View", "Navigate", "Debug", and "Help". On the left, the "Working Files" sidebar shows "index.html" selected. The main editor area contains the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3   <body>
4     <a href="http://niflare.com/">I think you love this</a>
5   </body>
6 </html>
```

Here is the result of the above code:





# HTML ATTRIBUTES

---

## **W**hat is an attributes?

An attributes gives some extra information about the html element to the browser. It can be a property related to width, height, or colour of an object.

The attribute is made up of 2 parts:

1. **Attribute Name**
2. **Attribute Value**

Note the following:

'=' is used to set the value.

“ ” the value is placed inside the double quotes.

It's like **attribute="value"**

---

# THE BASIC ATTRIBUTES

A secret passage to the hidden treasure, how exciting!

Here forms the same passage between the two web pages /documents. In layman terms, it lets you provide the link of the web page to allow the user to connect to other webpages.

```
<a href="https://www.niflare.com/index.html">Check out  
the link</a>
```

**Note:** The **src** attribute lets you provide the path for the image in a webpage. It instructs the browser where to look for the image.

```

```

You can define a width, height etc. for an image using these attributes:

**alt:** Defines text/name for an image. Just in case it can't be displayed.

**width:** An optional attribute to set the **width** of the image.

**height:** An optional attribute to set the **height** of the image.

Here is an example all being put to use.

```

```

```
File Edit Find View Navigate Debug Help
+ index.html (bracket) - Brackets

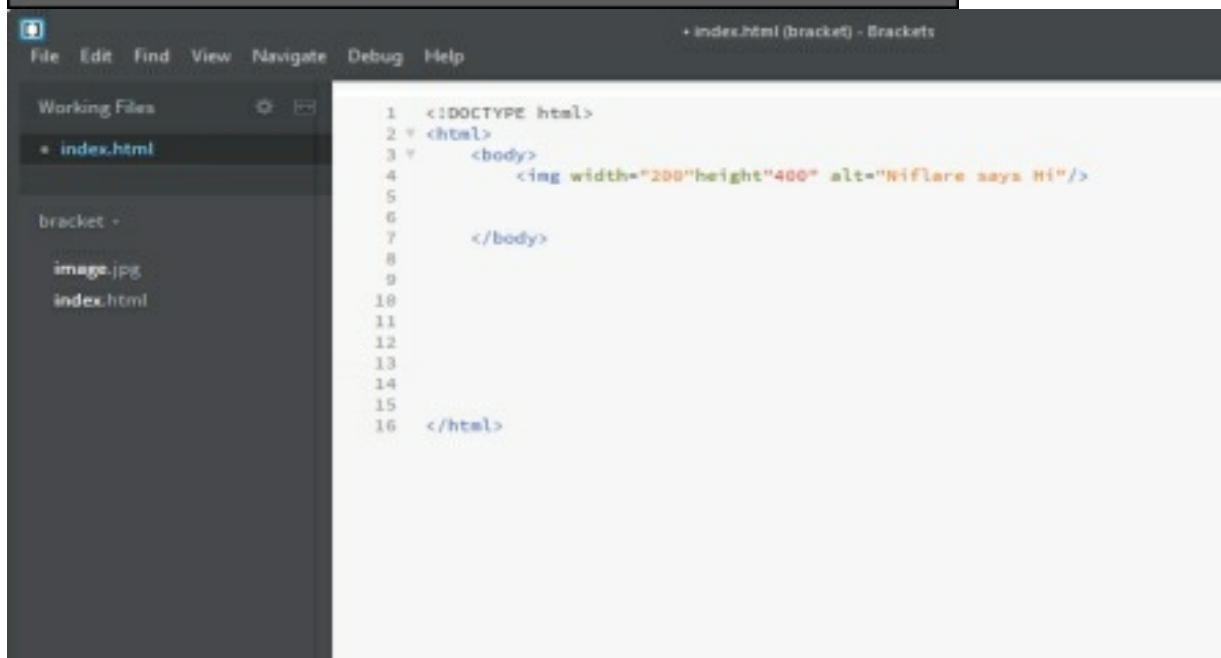
Working Files
  index.html
bracket -
  image.jpg
  index.html

1 <!DOCTYPE html>
2 <html>
3   <body>
4     
5
6   </body>
7
8
9
10
11
12
13
14
15 </html>
```



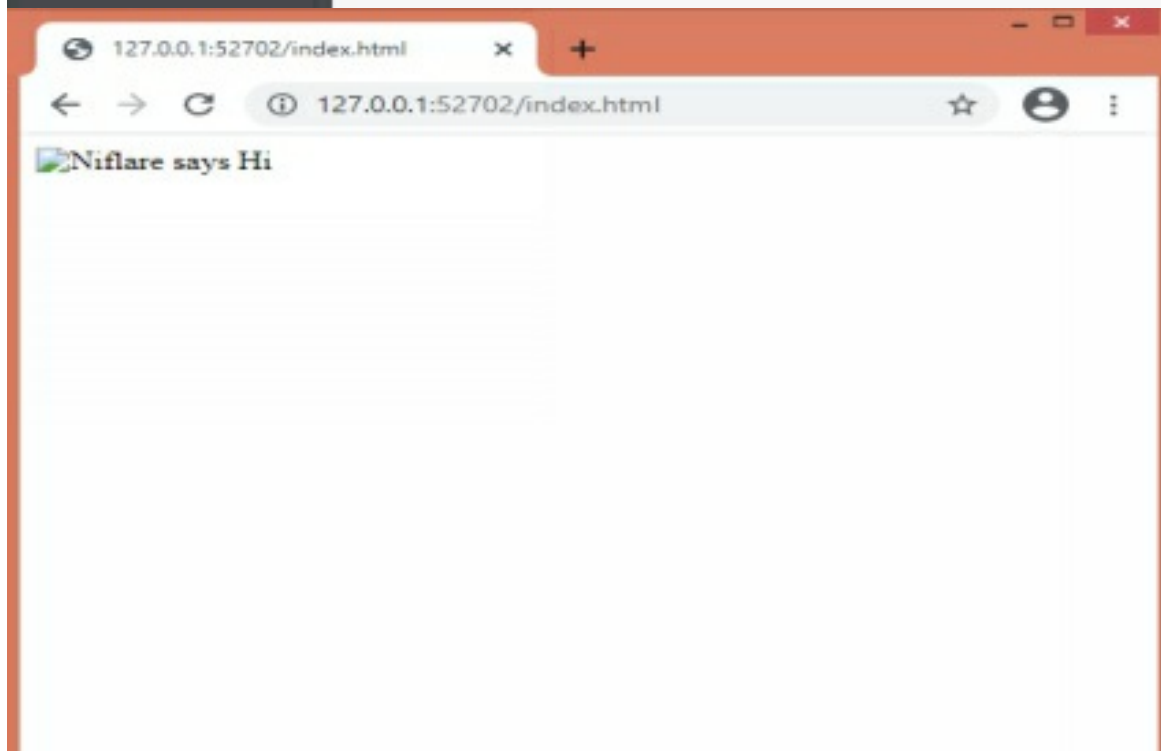
Try removing **src** to see how **alt** works.

```
<img width="200"height="400" alt="Niflare says Hi"/>
```



The screenshot shows the Brackets code editor interface. The top menu bar includes File, Edit, Find, View, Navigate, Debug, and Help. The left sidebar shows the 'Working Files' panel with 'index.html' selected. The main editor area displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4   <img width="200"height="400" alt="Niflare says Hi"/>
5
6
7 </body>
8
9
10
11
12
13
14
15
16 </html>
```



This might have provided you with a clearer picture.

---

# THE STYLE ATTRIBUTE

The style attribute does the final touch up of the HTML elements, it lets you change, or add different styles to it. It can be in form of background colour, size etc.

There is a simple syntax:

**style="property:value"**

The property implies basic colour, the font-size gives the property of an element

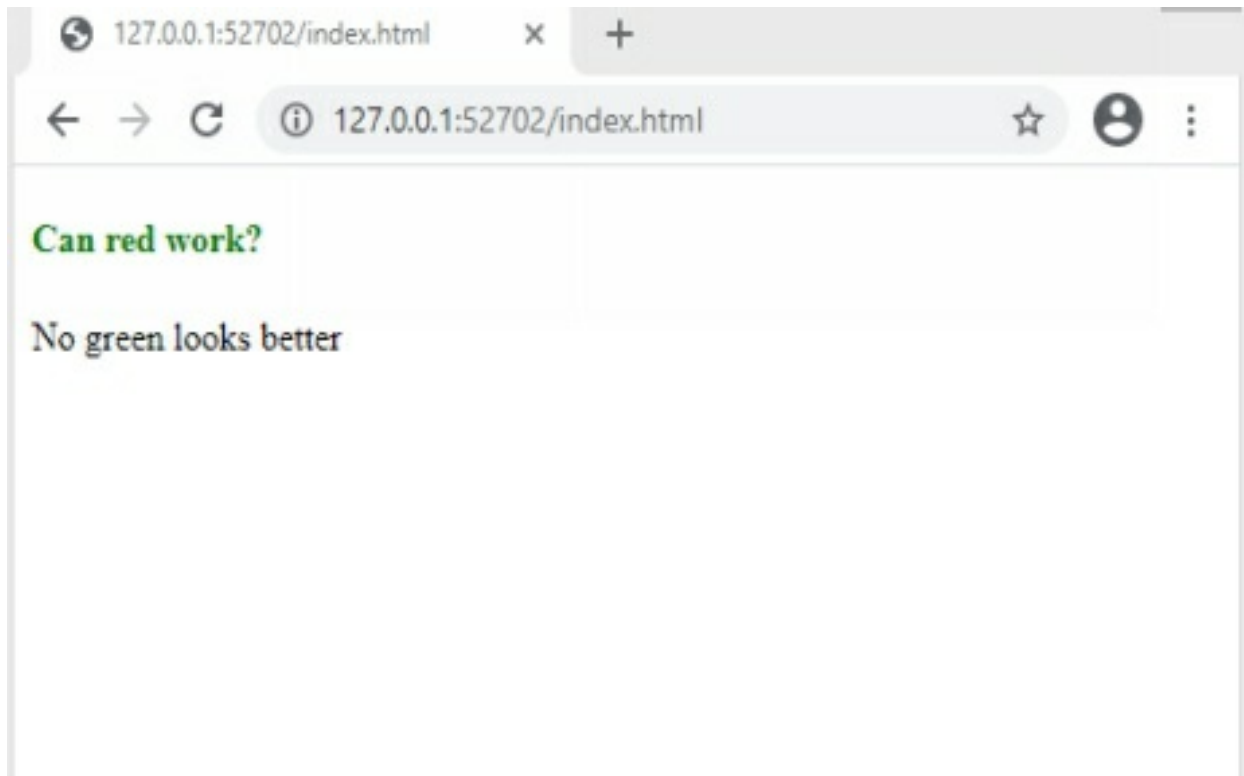
```
<h4 style="color:green">Can red work?</h4>  
<p style="font-size:200%">No green looks  
better</p>
```



The screenshot shows a code editor window titled "index.html (bracket) - Brackets". The editor displays the following HTML code:

```
1 <!DOCTYPE html>  
2 <html>  
3   <body>  
4     <h4 style="color: green">Can red work?</h4>  
5     <p style="font-size:200%">No green looks better</p>  
6  
7  
8  
9   </body>  
10  
11  
12  
13  
14  
15  
16  
17  
18 </html>
```

The code editor interface includes a menu bar (File, Edit, Find, View, Navigate, Debug, Help) and a sidebar on the left showing "Working Files" with "index.html" selected. The code is color-coded, with "color: green" and "font-size:200%" highlighted in orange.



This is how the changed text looks.

PS: it's a generic attribute and can be used with any of the html tags. Don't get confused, its similar to changing colours and size in a Word document, but by using HTML.

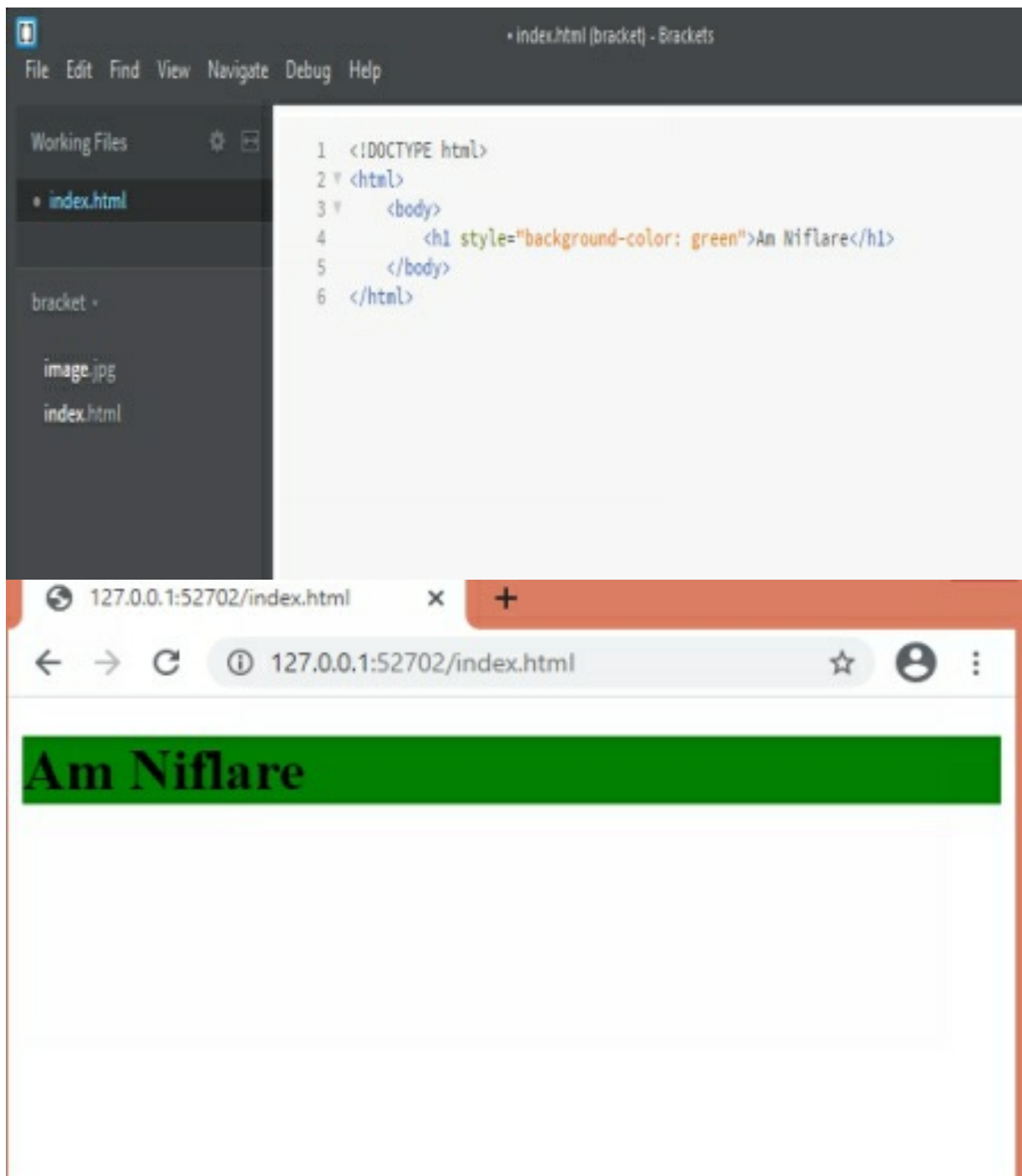
Try doing this.

Ready to change the background colour?

Hint: property: background-colour changes the background colour of the text

```
<h1 style="background-color:green">Am Niflare</h1>
```

And the output will be display like this:



---

# THE STYLE ELEMENT

Style can also be used as HTML tag.

We can build a new, table and even a list by using the <form>, <table>, <Li> tags.

Using style as an element is termed as embedded style.

**Denotation:** <style> & </style>

**Location:** it's placed inside the head tag

**Usage:** it lets you enhance the look and feel of the webpage.

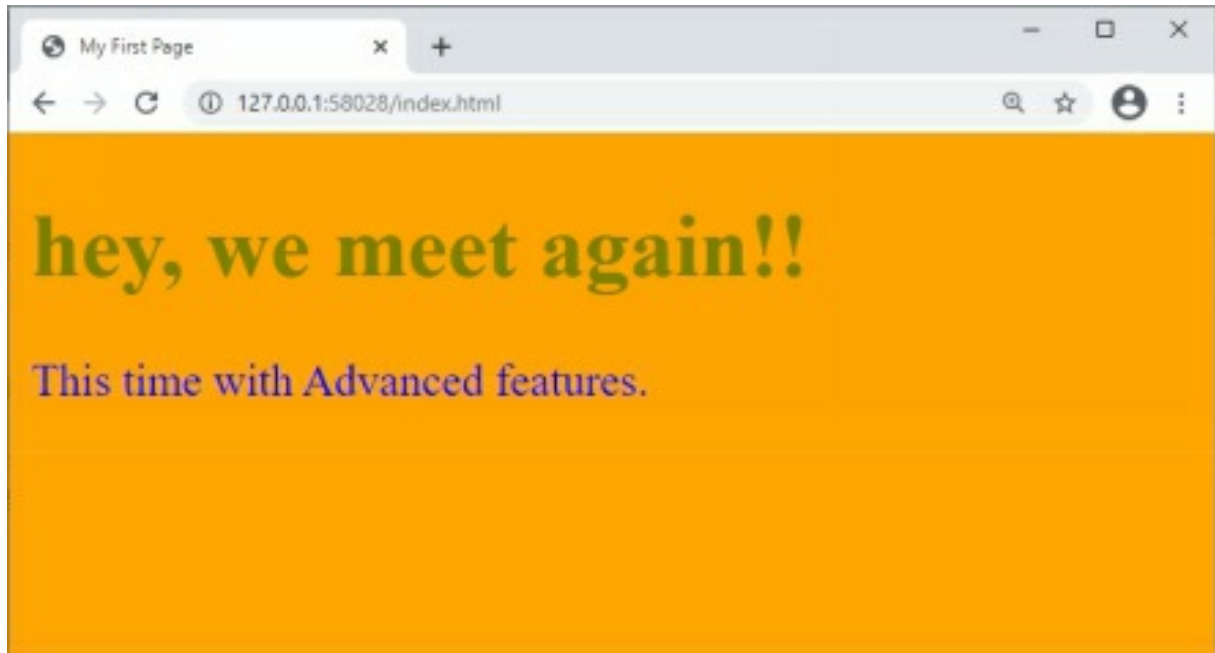
Let's see how it is done.



```
• index.html (a) - Brackets
View  Navigate  Debug  Help

1  <!DOCTYPE HTML>
2  <html>
3  <head>
4    <title>My First Page</title>
5    <style>
6      /* All headings visible with Purple text color*/
7      h1 { color:olive; }
8      /*All paragraphs text print in red color*/
9      p { color:blue; }
10     /*background color of the body should be powderblue*/
11     body{ background-color:orange;
12           }
13   </style>
14 </head>
15   |
16 <body>
17
18 <!-- This is a heading-->
19 <h1>hey, we meet again!!</h1>
20 <!--This prints a paragraph-->
21 <p> This time with Advanced features. </p>
22
23 </body>
24
25 </html>
26
```

Result:



---

# CREATING AN INPUT FIELD

Websites may want you to type information like your name, passwords etc. the simple solution to this is the input tag.

It comes as a package, with various attributes like id, name, type, placeholder etc.

The attributes are explained as follows:

**Denotation:** <input>

**Location:** place it as per your convenience

**Usage:** let's you take input from the user

**Special characteristic:** it's a self closing HTML Tag

**Common attributes:** id, type, name, placeholder, value

i.e

```
<- - This will print a paragraph with text color red and  
a textbook-->  
< p style="color:red,"  
Hey, yes you. can I have your number please:  
<input type="text" placeholder="number please">  
</p>
```

---

# CREATING A FORM

Filling and submitting an application form online, can be done by using the `<form>` element.

The simplest form comprises of **Text fields, Radio Buttons, Checkboxes** and a **'submit'** button.

Denotation : `<form> </form>`

Location: place it as per your convenience [`<body>`]

Usage: lets user enter data and submit it

Tags involved : `<input>`, `<button>`, `<label>`

Common attributes: action, type, id, name, value

Let's try this:

```
index.html [coding] - Brackets
File Edit Find View Navigate Debug Help
Working Files
index.html
coding -
index.html
1 <!DOCTYPE html>
2 <html>
3 <body>
4
5 <!--This will create a form-->
6 <form action="/my-form-page.php">
7
8 <!--A textbox shows with label'Name'-->
9 Name:<input type="text"><br/>
10
11 <!--A textbook shows with label'Email'-->
12 Email:<input type="text"><br/>
13
14 <!--Radio buttons created for selecting male or female -->
15 Gender:<input type="radio" name="gender" value="male">male
16
17 <input type="radio" name="gender" value="female">female<br/>
18
19 <!--This will show a Submit button-->
20
21 <button type="submit">submit</button>
22
23 </form>
24
25
26 </body>
```

Then this will be the result:

index.html × +

← → ↻ File | C:/Users/user/Documents/coding/index.htm

Name:

Email:

Gender:  male  female

**Action:** lets you define the location, where your data is submitted.

**Name, value:** let you specify name.

**Value:** it is use as reference element for input element, when the form is submitted.

---

# CREATING TABLES IN HTML

**TABLE:** A table is important for stored information for it to be orderly arranged. In HTML element the `<table>` element is use to create one table. `<tr>`,`<td>`,`<th>` are used to create **rows**, **cells** and **headers**.

**Denotation:** `<table></table>`

**Location:** inside the `<body>` element.

**Usage:** lets you arrange data in rows and columns.

**Tags involved:** `<tr>`, `<td>`,`<th>`.

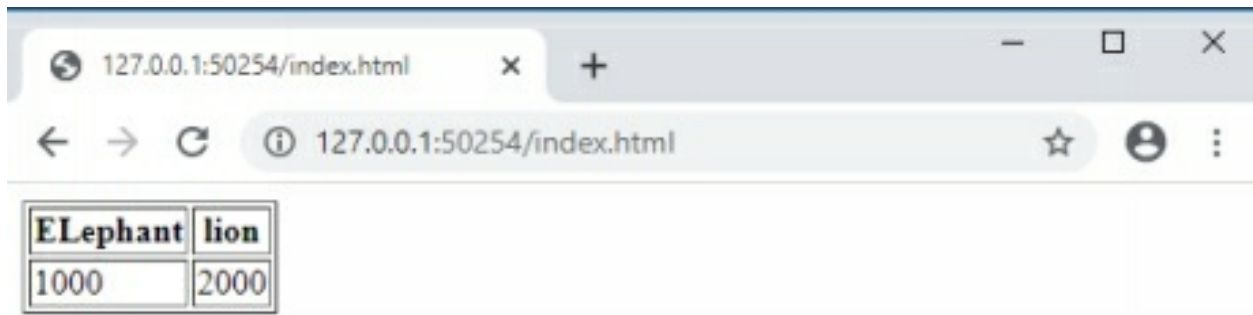
**Common attributes:** align, border, cellpadding , cellspacing.

Let's try this table:



```
1 |!DOCTYPE html>
2 |<html>
3 |<body>
4 |<!--This will create a table with border -->
5 |<table border="1">
6 |<!-- This will create a row -->
7 |<tr>
8 |<!--This prints a Heading-->
9 |<th>ELEphant</th>
10 |<!-- This prints another Heading-->
11 |<th>lion</th>
12 |</tr>
13 |<!-- This will create another row-->
14 |<tr>
15 |<!--This will create a column-->
16 |<td>1800</td>
17 |<!-- This will create another column-->
18 |<td>2800</td>
19 |</tr>
20 |</table>
21 |
22 |</body>
23 |
24 |
25 |</html>
```

Let's see the result :



The image shows a browser window with a single tab titled "127.0.0.1:50254/index.html". The address bar also displays "127.0.0.1:50254/index.html". Below the browser interface, a table is rendered with two columns and two rows. The first row contains the headers "ELEphant" and "lion", both in bold and centered. The second row contains the values "1000" and "2000".

<b>ELEphant</b>	<b>lion</b>
1000	2000

Headings are **bold, centre and aligned.**

---

# CREATING LIST USING HTML

Checklists are important part of our lives, especially when it comes to shopping.

So the following elements in HTML can be used to create a list.

<ul>- used for unordered lists.

<ol>- used for numbered lists.

<li>- defines an element of the list.

**Denotation:** <ul> </ul> and <ol> </ol>

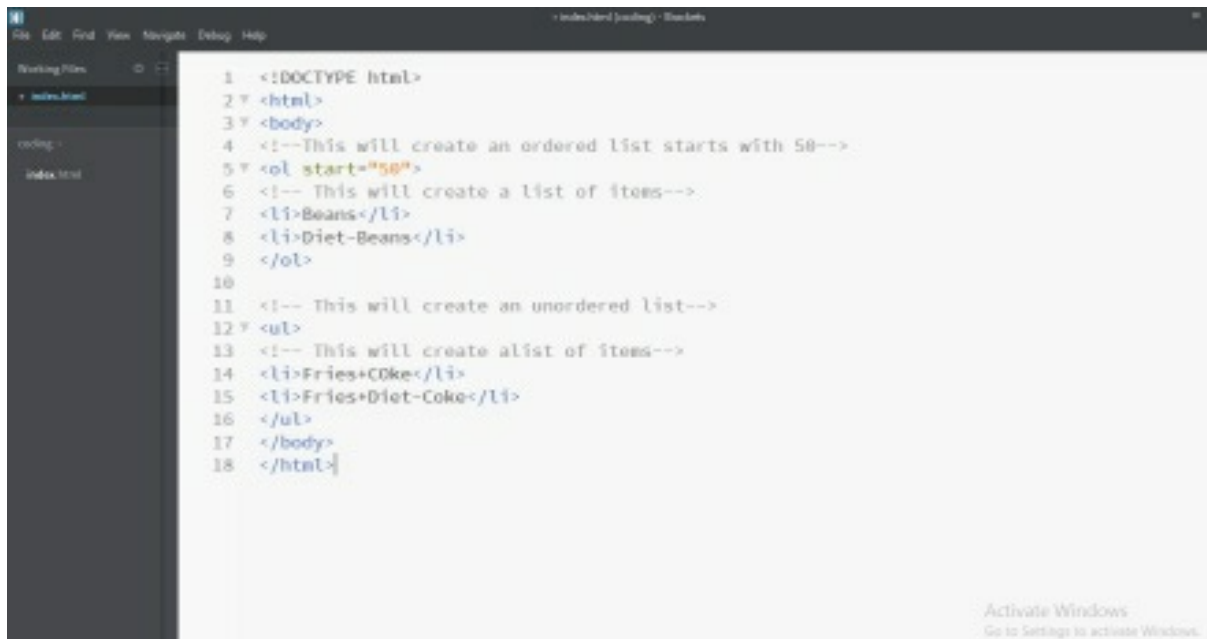
**Location:** inside the <body> element.

**Usage:** lets you arrange data in the form of a list.

**Tags involved :** <li>.

**Common Attributes:** type, start.

Let's create both **ordered** and **unordered** lists:

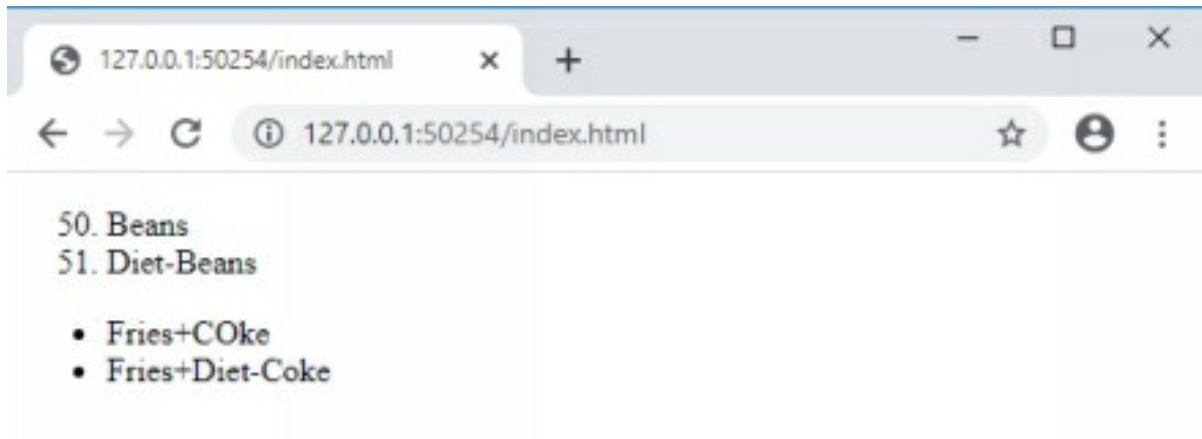


```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <!--This will create an ordered list starts with 50-->
5 <ol start="50">
6 <!-- This will create a list of items-->
7 <li>Beans</li>
8 <li>Diet-Beans</li>
9 </ol>
10
11 <!-- This will create an unordered list-->
12 <ul>
13 <!-- This will create a list of items-->
14 <li>Fries+Coke</li>
15 <li>Fries+Diet-Coke</li>
16 </ul>
17 </body>
18 </html>
```

Activate Windows  
Go to Settings to activate Windows.



And this should be our results:



# HTML 5

**HTML 5** is the 5<sup>th</sup> and latest version of HTML.

It helps you to include rich content like, **animation, graphics, videos** on your webpage and marks its presence with <DOCTYPE html>.

It's also help to build more interactive **forms** and **web applications**.

It's remove the need for additional browser plugins.

New elements like header, footer, article etc. are introduced for a better **meaningful structure**.

**User Experience and User Design** can be enhanced using HTML5.

Leads to a search engine friendly

**Geo-location** support is also provided by HTML5.

Character set used within the webpage can just be defined using <meta> tag and charset attribute in HTML5.

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

UTF-8, default character encoding used by HTML 5.

---

# NEW ELEMENT TO HTML

<!DOCTYPE html>- for all browsers to be on the same page.

For meaningful structure: <header>, <footer>, <article>,<section>,<nav>.

Media tags: <audio>, <video>, <canvas>.

Elements removed	Usage
<center>	Centered aligning
<Font>, <v>, <strike>	Font-color, size, underline and strikethrough text
<dr>	directory

Some deprecated attribute.

Elements removed	Usage
Width, height, rulers, summery	Attributes of <table> for studying rows and columns
Name	Specifies name of an input element Depreciated for <form> but not other elements
Type	Depreciated for <script> used for writing javascript in HTML

---

# HEADER ELEMENT

**HEADER:** are contents that goes on top of a webpage.

**Denotation:** <header> </header>

**Location:** inside the <body>

**Usage:** acts as a safeplace for your headings [<h1>..

######

Let's take a quick look.

A screenshot of a code editor window. The title bar shows 'index.html [coding] - Buckets'. The menu bar includes 'File', 'Edit', 'Find', 'View', 'Navigate', 'Debug', and 'Help'. The left sidebar shows 'Working Files' with 'index.html' selected. The main editor area displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <header>
5 <!-- This will print a Heading -->
6 <h1>Don't Study Me</h1>
7 <!-- This will print another Heading -->
8 <h2> Because You Won't Graduate</h2>
9
10 </header>
11
12 </body>
13 </html>
```

And here is our header:

127.0.0.1:50254/inde x

127.0.0.1:50254/inde x



127.0.0.1:50254/index.html

# **Don't Study Me**

## **Because You Won't Graduate**

---

# FOOTER OF A WEBPAGE

**FOOTER** is the last section, and that's where the name of author's goes in magazine blog, in HTML5, contact information, other web links, copyright of the webpage is segregated using `<footer>` element.

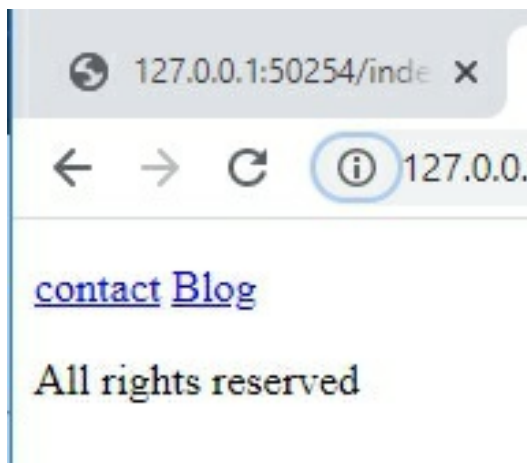
**Denotation:** `<footer> </footer>`.

**Location:** inside the `<body>`.

Let's have look at this small code:

```
1  <!DOCTYPE html>
2  <html>
3  <body>
4  <!-- This will create a footer of the webpage-->
5  <footer>
6  <!-- This will print a paragraph -->
7  <p>
8  <!-- This will create a hyperlink with Text Contact-->
9  <a href="contact.php">contact</a>
10 <!-- This will create a hyperlink with Text Blog-->
11 <a href="blog.php">Blog</a>
12 </p>
13 <!-- This will print another paragraph-->
14 <p>All rights reserved</p>
15     </footer>
16
17 </body>
18 </html>
```

You would have noticed, this is the footer content for most websites:



**NAVIGATION SECTION:** the major navigational links of a webpage are like a menu, About etc. are gathered together under the `<nav>` element.

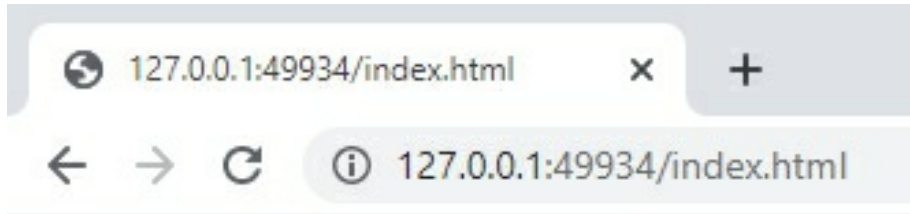
**Denotation:** `<nav> </nav>`.

Location: inside the `<body>`.

Let's create a neat horizontal menu for major navigational links:

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <!-- This will create a section in webpage where links are present-->
5 <nav>
6 <!-- This will print a paragraph -->
7 <p>
8 <!-- This will print the texts in bold-->
9 <b>major links of a webpage</b>
10 </p>
11 <!-- This will create a hyperlink with text About-->
12 <a href="About.html">About</a>
13 <!-- This will create a hyperlink with text Gallery-->
14 <a href="Gallery.html">Gallery</a>
15 <!-- This will create a hyperlink with Text Blog-->
16 <a href="Blog.html">Blog</a>
17 <!-- This will create a hyperlink with text Menu-->
18 <a href="Menu.html">Menu</a>
19 </nav>
20
21 </body>
22 </html>
```

Let's see how it looks:



## **major links of a webpage**

[About](#) [Gallery](#) [Blog](#) [Menu](#)



---

# THE GENERIC SECTION

The sections element in HTML5 comprises of the descriptive content like paragraphs of a webpage, which can be grouped by different headings.

**Denotations:** <section> </section>

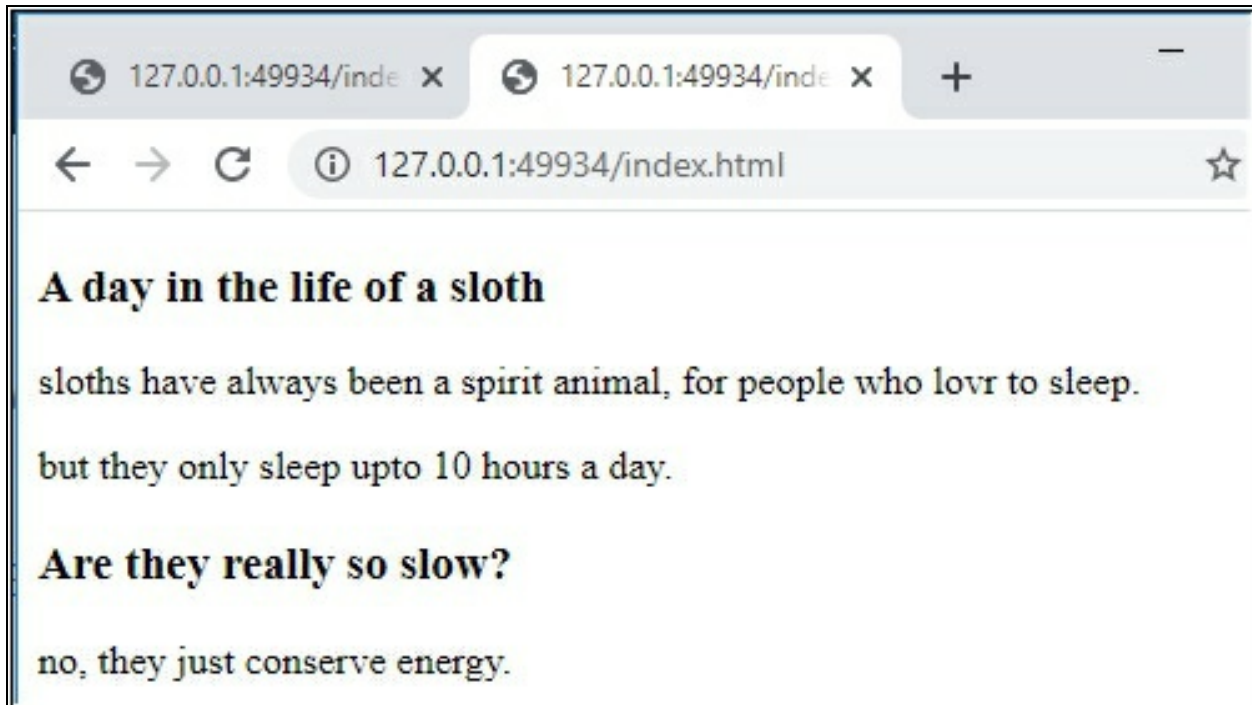
**Location:** inside the <body>

**Usage:** Marking up different sections of a webpage.

So check this out:

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <!-- This will create a section in webpage-->
5 <section>
6 <!-- This will print a Heading-->
7 <h3>A day in the life of a sloth</h3>
8 <!-- This will print a paragraph-->
9 <p>sloths have always been a spirit animal, for people who love to sleep.</p>
10 <!--This will print another paragraph-->
11 <p>but they only sleep upto 10 hours a day.</p>
12 </section>
13
14 <!-- This will create another section in webpage-->
15 <section>
16 <!-- This will print a Heading-->
17 <h3>Are they really so slow?</h3>
18 <!-- This will print a paragraph-->
19 <p>no, they just conserve energy.</p>
20 </section>
21 </body>
22 </html>
```

And here is the result:



Learning: The content inside `<div>` does not have a special meaning whereas in.

`<section>` our core intention is to divide on the basis of meaning.

---

# STREAMING THE CONTENT

The `<article>` contains content like **blog/forum** posts. The comment is also an example.

Denotation: `<article> </article>`.

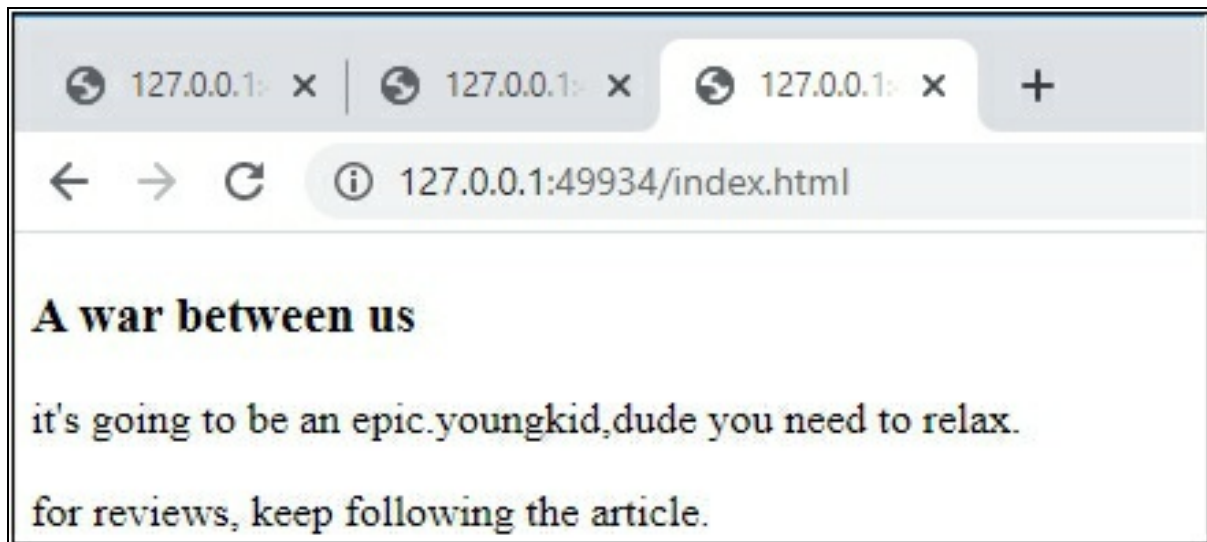
Location: inside the `<body>`.

Usage: reusable, meaningful, independent components of a webpage.

Let's explain this:

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <!-- This will create an article content-->
5 <article>
6 <!-- This will print a Heading-->
7 <h3>A war between us</h3>
8 <!-- This will print a paragraph-->
9 <p>
10 it's going to be an epic.youngkid,dude you need to relax.
11 </p>
12 <!--This will print another paragraph-->
13 <p>
14 for reviews, keep following the article.
15 </p>
16 </article>
17 </body>
18 </html>
```

And this the result:



<article> can be always used a child to the <section>

---

# THE SIDE ELEMENT

This are content which is indirectly related to the main content, but is still important, is clubbed together under the <aside> element, tangentially.

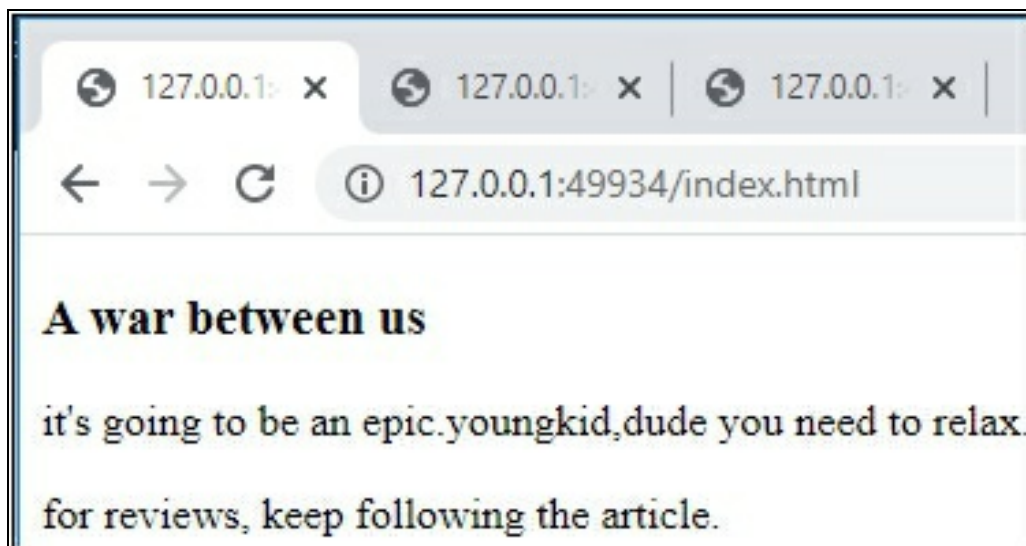
**Denotation:** <aside> </aside>

Location: inside the <body>

Usage: used as a sidebar; additional content which makes the main content look more useful can be placed inside <aside>.

```
1  <!DOCTYPE html>
2  <html>
3  <body>
4  <!-- This will print a paragraph-->
5  <p>
6  young kid is part of the people that rape the girl.
7  </p>
8  <!--This will create a sidebar-->
9  <aside>
10 <!-- This will print a Heading-->
11 <h4>stark paul</h4>
12 <!--This will print a paragraph-->
13 <p>He is the brave man!!</p>
14 </aside>
15 </body>
```

Result:



---

# MULTIMEDIA ELEMENT

**Audio support:** HTML5 let you embed music or audio file to your webpage. This can be achieved using the <audio>, <source> tags.

**Note:** <source> can be replaced with the **src** attribute.

**Denotation:** <audio> </audio>.

**Location:** inside the <body>.

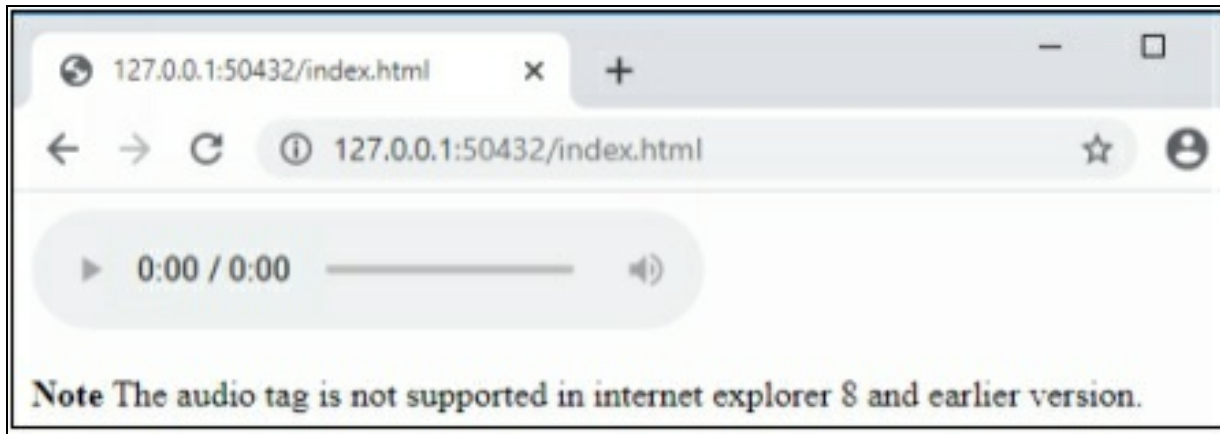
**Tags involved:** <source>.

**Major attributes involved:** src, type, controls.

Let's do some coding:

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <!-- This will load an audio-->
5 <audio controls="controls">
6 <!-- This refers the main source of the audio-->
7 <source src="music.mp3" type="audio/mpeg">
8 </audio>
9 <!-- This will print a paragraph-->
10 <p>
11 <!-- This will print the text bold-->
12 <b>Note</b>
13 The audio tag is not supported in internet explorer 8 and earlier version.</p>
14 </body>
15 </html>
```

Let's check this out!



Controls: provide you with the playback functionalities (play, pause, volume).

Src: lets you add url to embed. Can replace <source> if added with <audio>.

Type: specifies the supported format for audio.

**Video support:** as we all know the videos do easily catch a user's attention. so if you want to embed a video to your webpage, here is the <video> element.

**Denotation:** <video> </video>.

Location: inside the <body>.

Tags involved: <source>.

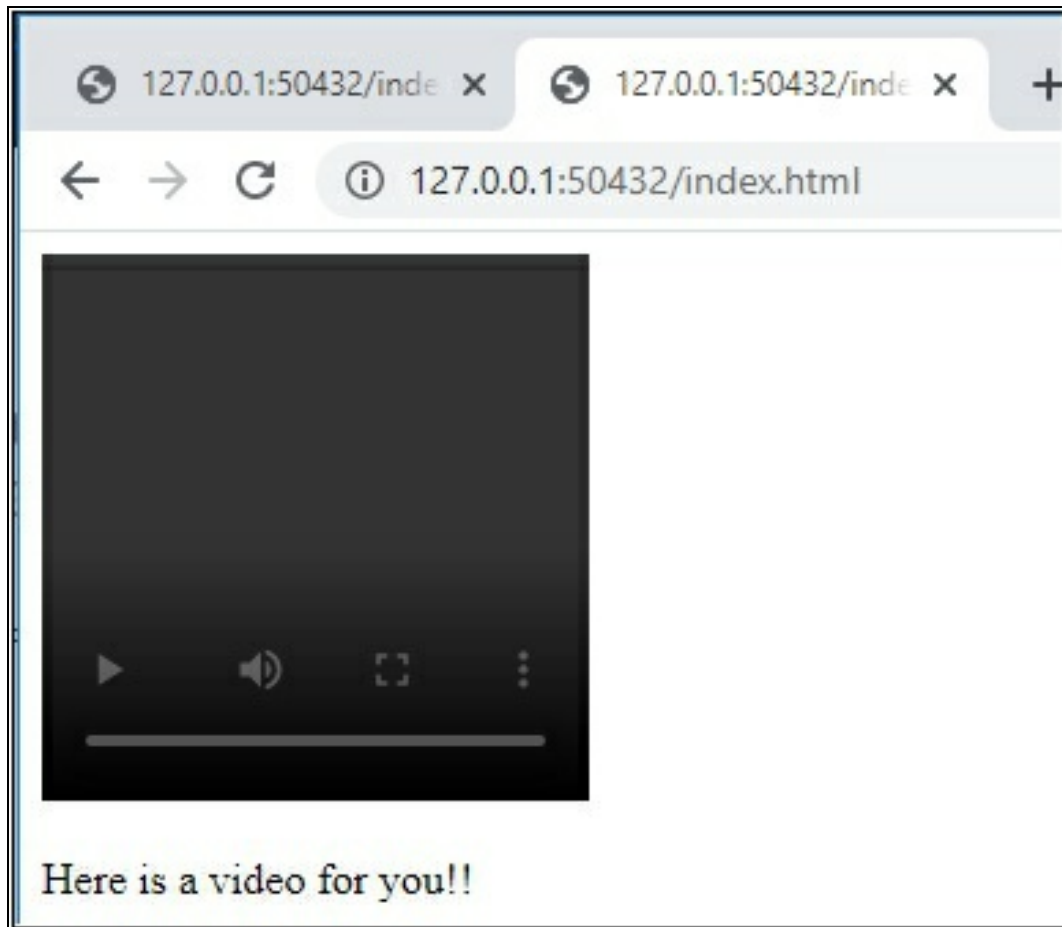
Major Attributes involved: src, type, controls, width, height, autoplay.

Let's try some coding:

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <!-- This will load a video which starts automatically-->
5 <video controls="controls" width="200" height="200" autoplay>
6 <!-- This refers the main source of the video-->
7 <source src="walking_video.mp4" type="video/mp4">
8 </video>
9 <!-- This will print a paragraph-->
10 <p> Here is a video for you!!</p>
11 </body>
12 </html>
```



And the result:



Adding **autoplay**, plays the video on webpage load. Can also be used for `<audio>`.

Try adding `poster = "URL.png"` attributes to the `<video>`.

NOTE: **poster** lets you **add a static frame** to the video before the user plays it.

---

# THE EMBED ELEMENT

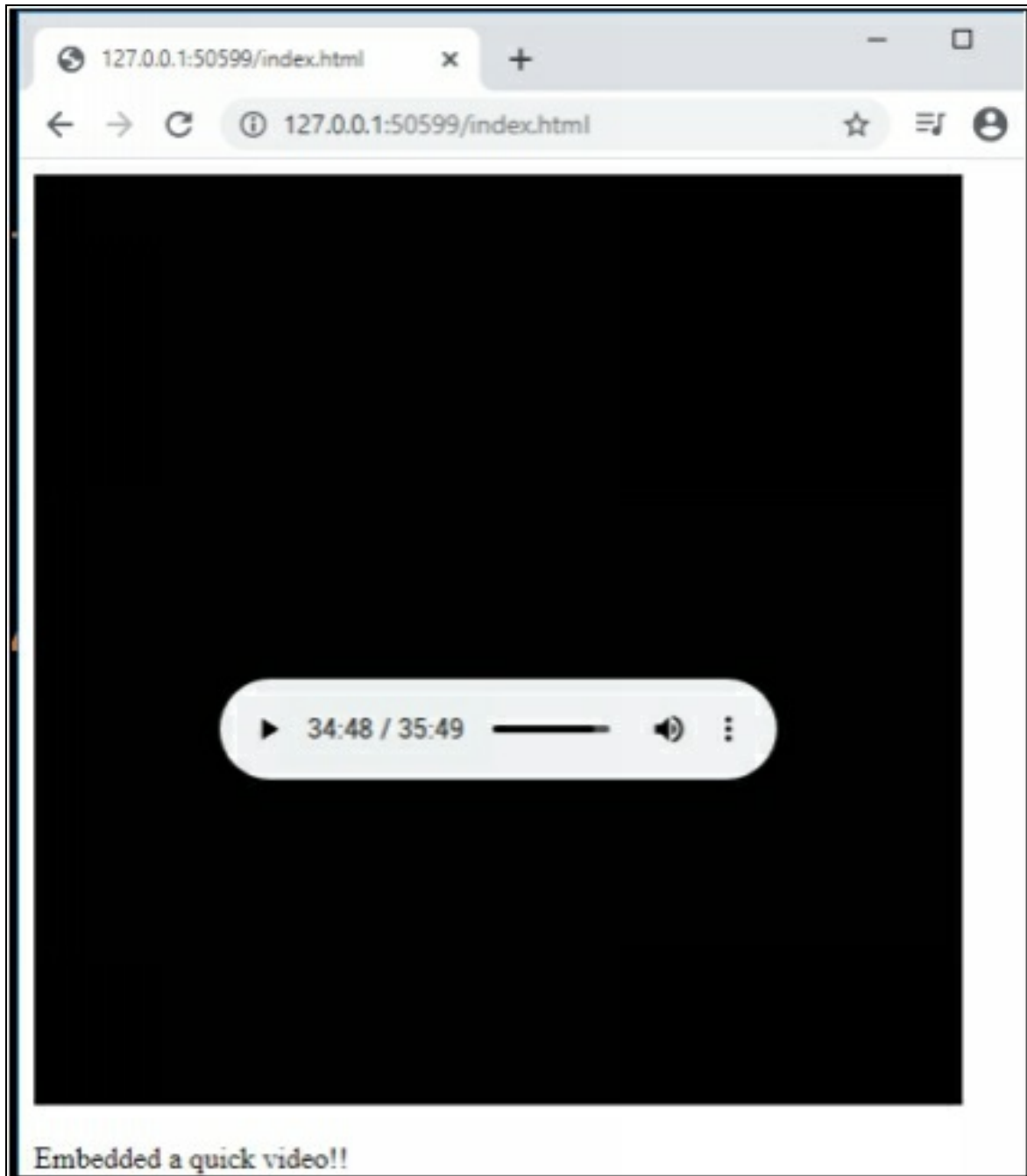
Adding third-party/external content to the webpage using <embed>

Denotation: <embed>.

Location: inside the <body>.

Major attributes involved: src, type, width, height.

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <!-- This will create a container for adding an external content-->
5 <embed src="The%20Cure%20(2020)%20(Naija.com).mp4" type="video/quicktime"
  width="500" height="500" >
6 <!-- This will print a paragraph-->
7 <p>Embedded a quick video!!</p>
8
9 </body>
10 </html>
```



**Additional Notes:**

Most of the attributes `<audio>`, `<video>` and `<embed>`, can be kept on mute initially, using the muted attribute.

For removing autoplay you have to remove the attribute rather than assigning it's value as false.

---

# NEW FORMS TO FORM ELEMENT

With html5, forms have become more much able. Let's learn about them:

`<form autocomplete="on/off">`

Provides an auto-completion option via the browser.

On submitting and reloading the form, it provides the suggestion of the values previously entered.

**<Field>**

Groups together the related elements.

Better view by distinguishable sections.

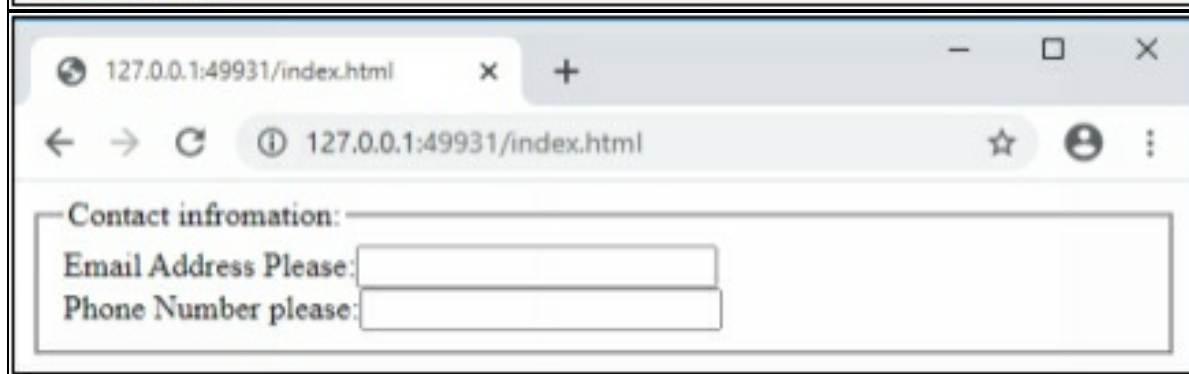
`<input>`, `<label>`, `<legend>` are nested in.

**<legend>**

To provide a name/title to the `<fieldset>`

Let's show how it is done:

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <!-- This will create a form-->
5 <form>
6 <!-- This will create a container for storing similar elements-->
7 <fieldset size="30">
8 <!-- This will provide a name for fieldset-->
9 <legend>Contact infromation:</legend>
10 <!-- This will create a textbox for Email-->
11 Email Address Please:<input type="text"><br>
12 <!-- This will create a textbox for Phone number-->
13 Phone Number please:<input type="text"><br>
14 </fieldset>
15 </form>
16 </body>
17 </html>
```



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:49931/index.html". The browser's address bar includes navigation buttons (back, forward, refresh) and a search icon. The main content area of the browser displays a contact form with the following elements:

- A legend titled "Contact infromation:" (note the typo).
- A text input field labeled "Email Address Please:".
- A text input field labeled "Phone Number please:".

---

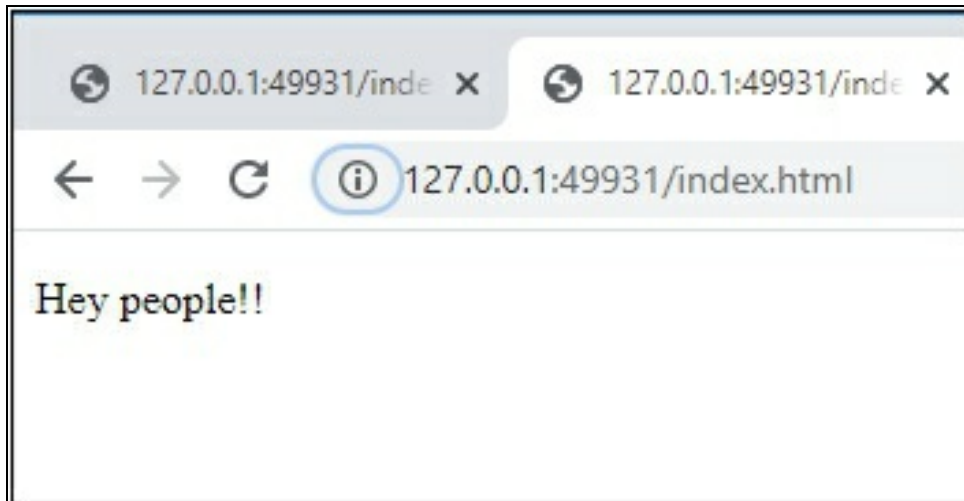
# EXCITING FEATURE IN HTML5

**Content is Editable:** your content is editable, like typing and deleting your texts.

It is achieved by using the attribute **contenteditable**

We just have to set it as true `<p contenteditable="true">`.

```
1  <!DOCTYPE html>
2  <html>
3  <body>
4  <p contenteditable="true">
5  Hey people!!
6      </p>
7  </body>
8  </html>|
```



Drag and drop: any element can be dragged and dropped within the HTML page.

It's can be achieved by using setting `draggable="true"`.

So we decide which elements are to be dragged and where.

**Ondragstart** an event to be defined using javascript on the element to be dragged.

Ondragover an event to specify where data can be dropped.