HTML :

HTML stands for Hyper Text Markup Language it was created by Berners-Lee in late 1991 , which is the most widely used language on Web to develop web pages.

Web browsers can read HTML files and compose them into visible or audible web pages. Browsers do not display the HTML tags and scripts, but use them to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language, rather than a programming language .

HTML VERSIONS

VERSION	YEAR
Html	1991
Html 2.0	1995
Html 3.2	1997
Html 4.01	1999
XHtml	2000
Html 5	2012

GETTING STARTED :

HTML files are just simple text files, so to start writing in HTML, you need a simple text editor ,for learning HTML a simple text editor like Notepad (PC) or TextEdit (Mac) will be easy . HTML can be edited by using a professional HTML editor like Adobe Dreamweaver , Microsoft Expression Web,CoffeeCup HTML Editor .

To create a simple Webpage with notepad :

Open Notepad (PC) or TextEdit (Mac)

Write some HTML into Text Editor :

Following is an example of a simple HTML document with Heading and Paragraph :

The following code can be copy/pasted in the "Try yourself" Section.

```
 <!DOCTYPE html>
<html>

<body>

  <h1>This is Heading</h1>

  <p> This is paragraph</p>

</body>

</html>
```

Save the HTML page :

The finished page should be saved in .Html extension .

UTF-8 is the preferred encoding for HTML files .

To view the HTML page :

Open the saved HTML file in your browser. The page will look like this:

HTML BASICS :

A simple HTML Document :

```
<!DOCTYPE html>
<html>

<head>
  <title>Title of the page</title>
</head>
```

```
<body>
  <h1>This is Heading</h1>
  <p>This is paragraph</p>
</body>
</html>
```

Explanation for the above HTML document :

1.The DOCTYPE declaration defines the document type to be HTML.

2.The text between <html> and </html> describes an HTML document.

3.The text between <head> and </head> provides info about the document.

4.The text between <title> and </title> provides a title for the document.

5.The text between <body> and </body> describes the visible page content.

6.The text between <h1> and </h1> describes a heading.

7.The text between <p> and </p> describes a paragraph.

HEADING IN HTML :

Every document will have a heading. HTML has different sizes for headings. which use the following elements <h1>, <h2>, <h3>, <h4>, <h5>, and <h6>. While displaying any heading, browser adds one line before and one line after that heading.

Example for Heading :

```
 <!DOCTYPE html>

<html>

<head>

<title>Heading Example</title>

</head>

<body>
<h1>Example for heading h1</h1>
<h2>Example for heading h2</h2>
```

```
<h3>Example for heading h3</h3>

<h4>Example for heading h4</h4>

<h5>Example for heading h5</h5>

<h6>Example for heading h6</h6>

</body>


</html>
```

Result for the above Example :

PARAGRAPH IN HTML:

In HTML you can structure your text into different paragraphs by using the tag <p>.Each paragraph of text should go in between an opening <p> and a closing </p> tag .

Example for Paragraph :

```
 <!DOCTYPE html>

<html>


<head>


<title>Paragraph Example</title>
```

```
</head>

<body>
<p>This is First Paragraph.</p>
<p>This is Second Paragraph.</p>
<p>This is Third Paragraph.</p>
</body>

</html>

</html>
```

Result for the above example :

LINE BREAK IN HTML :

If you use the <br> tag , anything following it starts from the next line . This tag is an example of an empty element, there is no need for opening and closing tags, as there is nothing to go in between them.

Example for Line break :

```
<!DOCTYPE html>
<html>

<head>

<title>Example for Line Break
</title>

</head>

<body>
<p>This <br>
is an example for<br>
Line Break</p>
</body>

</html>
```

Result for the above example :

CENTERING CONTENT :

To center any content in HTML You can use <center> tag this will to put any content inside the tag in the center of the page or any table cell.

Example for Centering content :

```
<!DOCTYPE html>
<html>

<head>

<title>Centering content</title>

</head>

<body>
<p>This is not in the center .</p>
<center>
<p>This is in the center.</p>
</center>

</body>
</html>
```

Result for the above example :

PRESERVE FORMATTING :

If want your text to be exact format of how it is written in the HTML document you can use the preformatted tag <pre>. Any text between the opening <pre> tag and the closing </pre> tag will preserve the formatting of the source document.

Example for Preserve formatting :

```
<!DOCTYPE html>

<html>

<head>

<title>Preserve Formatting</title>
</head>



<body>
```

```
<pre>

This
  is
   an
    Example for preserve
</pre>
</body>

</html>
```

Result for the above example :

HORIZONTAL LINE :

To create a horizontal line between a section of document the <hr> tag can be used it creates a line from the current position in the document to the right margin and breaks the line accordingly .

Example for horizontal line :

```
<!DOCTYPE html>

<html>

<head>

<title>Horizontal Line </title>

</head>

<body>
<p>This is above the line</p>
<hr />
<p>This is below the line</p>
</body>

</html>
```

HTML BASICS :

A simple HTML Document :

```
<!DOCTYPE html>
<html>

<head>
  <title>Title of the page</title>
</head>

<body>

  <h1>This is Heading</h1>

  <p>This is paragraph</p>

</body>

</html>
```

Explanation for the above HTML document :

1.The DOCTYPE declaration defines the document type to be HTML.

2.The text between <html> and </html> describes an HTML document.

3.The text between <head> and </head> provides info about the document.

4.The text between <title> and </title> provides a title for the document.

5.The text between <body> and </body> describes the visible page content.

6.The text between <h1> and </h1> describes a heading.

7.The text between <p> and </p> describes a paragraph.

HEADING IN HTML :

Every document will have a heading. HTML has different sizes for headings. which use the following elements <h1>, <h2>, <h3>, <h4>, <h5>, and <h6>. While displaying any heading, browser adds one line before and one line after that heading.

Example for Heading :

```
 <!DOCTYPE html>

<html>
```

```
<head>

<title>Heading Example</title>

</head>

<body>
<h1>Example for heading h1</h1>
<h2>Example for heading h2</h2>
<h3>Example for heading h3</h3>
<h4>Example for heading h4</h4>
<h5>Example for heading h5</h5>
<h6>Example for heading h6</h6>
</body>

</html>
```

Result for the above Example :

PARAGRAPH IN HTML:

In HTML you can structure your text into different paragraphs by using the tag <p>.Each paragraph of text should go in between an opening <p> and a closing </p> tag .

Example for Paragraph :

```
 <!DOCTYPE html>

<html>


<head>


<title>Paragraph Example</title>


</head>


<body>

<p>This is First Paragraph.</p>

<p>This is Second Paragraph.</p>

<p>This is Third Paragraph.</p>

</body>


</html>


</html>
```

Result for the above example :

LINE BREAK IN HTML :

If you use the <br> tag , anything following it starts from the next line . This tag is an example of an empty element, there is no need for opening and closing tags, as there is nothing to go in between them.

Example for Line break :

```
<!DOCTYPE html>
<html>


<head>


<title>Example for Line Break
</title>


</head>


<body>
<p>This <br>
is an example for<br>
```

```
Line Break</p>

</body>


</html>
```

Result for the above example :

CENTERING CONTENT :

To center any content in HTML You can use <center> tag this will to put any content inside the tag in the center of the page or any table cell.

Example for Centering content :

```
<!DOCTYPE html>

<html>


<head>


<title>Centering content</title>


</head>
```

```
<body>

<p>This is not in the center .</p>

<center>

<p>This is in the center.</p>

</center>


</body>

</html>
```

Result for the above example :

PRESERVE FORMATTING :

If want your text to be exact format of how it is written in the HTML document you can use the preformatted tag <pre>. Any text between the opening <pre> tag and the closing </pre> tag will preserve the formatting of the source document.

Example for Preserve formatting :

```
<!DOCTYPE html>
```

```
<html>

<head>

<title>Preserve Formatting</title>
</head>


<body>
<pre>

This
    is
      an
         Example for preserve
</pre>
</body>

</html>
```

Result for the above example :

HORIZONTAL LINE :

To create a horizontal line between a section of document the <hr> tag can be used it creates a line from the current position in the document to the right margin and breaks the line accordingly .

Example for horizontal line :

```
<!DOCTYPE html>

<html>

<head>

<title>Horizontal Line </title>

</head>

<body>
```

```
<p>This is above the line</p>

<hr/>

<p>This is below the line</p>

</body>


</html>
```

TEXT COLOR :

Text color property changes the text color for an HTML element .

Example for Text color :

```
<!DOCTYPE html>


<html>


<body>

<h1 style="color:RoyalBlue">


This is a heading</h1>


<p style="color:SteelBlue">
```

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

</html>
```

Result for the above Example :TEXT FONTS :

The font-family property defines the font to be used for an HTML element .

Example for Text fonts :

```
<!DOCTYPE html>

<html>

<body>

<h1 style="font-family:courier">

Heading in courier font</h1>
```

```
<p style="font-family:PT Serif">

paragraph in PT Serif font.</p>

</body>

</html>
```

TEXT SIZE :

The font-size property defines the text size to be used for an HTML element .

Example for Text size :

```
<!DOCTYPE html>

<html>

<body>
<h1 style="font-size:180%">

This is a heading</h1>

<p style="font-size:120%">
```

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

</html>
```

TEXT ALIGNMENT :

The text-align property defines the horizontal text alignment for an HTML element

Example for Text alignment :

```
<!DOCTYPE html>

<html>

<body>
<h1 style="text-align:center">

Heading is Centered</h1>

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

```
</html>
```

HTML FORMATTING :

In HTML the text can be formatted in many ways with the ability to make text bold, italicized, or underlined , these are few of the option available to indicate how text can appear in HTML .

BOLD FORMATTING :

Bold formatting property changes the element to bold format,anything that appears within <b>……..</b> element,is displayed in bold .

Example for Bold formatting :

```
<!DOCTYPE html>

<html>

<head>

<title>Bold Formatting</title>

</head>
```

```
<body>

<p>The following word uses a


<b>bold</b> typeface.</p>

</body>


</html>
```

Result for the above Example

ITALIC FORMATTING :

Italic formatting property changes the element to bold format anything that appears within <i>…</i> element is displayed in italicized

Example for Italic Formatting :

```
<!DOCTYPE html>


<html>


<head>
```

```
<title>Italic Text Example</title>

</head>

<body>
<p>The following word uses a

<i>italicized</i> typeface.</p>
</body>

</html>
```

Result for the Above Example :

UNDERLINED & STRIKE :

Underlined and Strike format property changes the element to underlined and strikethrough,anything that appears within <u>……..</u> element is displayed with underline and Anything that appears within <strike>..</strike> element is displayed with strikethrough.

Example for Underlined and strike :

```
<!DOCTYPE html>
```

```
<html>

<head>

<title>Underlined and Strike

Text Example</title>

</head>

<body>

<p>The following word uses a
<u>underlined</u> typeface.</p>

<p>The following word uses a
<strike>strikethrough</strike>
typeface.</p>


</body>

</html>
```

Result for the above Example :

MARKED AND DELETED :

Marked and Deleted format property changes the element to Marked and deleted , anything that appears within <mark>…..</mark> element is displayed as marked text and Anything that appears within <del>…..</del> element is displayed as deleted text.

Example for Marked and Deleted :

```
<!DOCTYPE html>

<html>

<head>

<title> Marked & Deleted </title>

</head>

<body>

<p> This is <mark>Marked</mark></p>
```

```
<p> This is <del>Deleted</del> </p>

</body>

</html>
```

Result for the above Example :

LARGE AND SMALL TEXT:

Large and Small format property changes the element to Larger and small text . The content of the <big>…</big> element is displayed one font size larger than the rest of the text surrounding it . The content of the <small>…</small> element is displayed one font size smaller than the rest of the text surrounding it .

The Example for Large and Small Text :

```
<!DOCTYPE html>

<html>

<head>
```

```
<title>Larger Text Example</title>

</head>

<body>

<p>The following word is in
<big>LARGE</big> typeface.</p>

<p>The following word is in
<small>SMALL</small> typeface.</p>

</body>

</html>
```

Result for the above Example :

GROUPING CONTENT :

The <div> and <span> elements allow you to group together several elements to create sections or subsections of a page.The <div> tag defines a division or a section in an HTML document.The <span> element can be used to group inline elements only . So , if you have a part of a sentence or paragraph which you want to group together, you could use the <span> element .

Example for Grouping content :

```
<!DOCTYPE html>

<html>

<body>

<div style="color:steelblue">

<h3>This is a heading in a
div element</h3>

<p>This is some text in a
div element.</p>

</div>

<p>This is the example of
<span style="color:steelblue">
```

```
span tag</span>

</body>
</html>

</body>

</html>
```

Result for the above Example :

SUPER & SUBSCRIPT :

Superscript and Subscript format property changes the element to Superscript and Subscript . The content of the <sup> ……. </sup> element is displayed in superscript format . The content of the <sub>…</sub> element is displayed in subscript format .

Example of Super & Subscript :

```
<!DOCTYPE html>

<html>

<head>

<title>Super & Subscript Example
</title>

</head>

<body>

<p>The following word uses a
<sup>superscript</sup> typeface</p>

<p>The following word uses a
<sub>subscript</sub> typeface</p>

</body>

</html>
```

Result for the above Example :

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

Css

HTML CSS :

CSS stands for Cascading Style Sheets , it defines how HTML elements are to be displayed.

It can be added to HTML elements in 3 ways:

1.Inline - using a style attribute in HTML elements

2.Internal - using a <style> element in the HTML <head> section

3.External - using one or more external CSS files

The syntax for CSS Styling is :

element { property:value ; property:value }

INLINE STYLING :

Inline styling is useful for applying a unique style to a single HTML element .

Example for Inline Styling :

```
<!DOCTYPE html>

<html>

<body>

<h1 style="color:steelblue">

This is inline styling example</h1>

</body>

</html>
```

Result for the above Example :

INTERNAL STYLING :

An internal style sheet can be used to define a common style for all HTML elements on a page.

Internal styling is defined in the <head> section of an HTML page, using a <style> tag

Example for Internal styling :

```
<!DOCTYPE html>
<html>

<head>

<style>
body {background-color:

MediumTurquoise  }

h1   {
color:steelblue;
font-family:verdana;
font-size:300%;
}

p   {

color:DarkOrchid;
font-family:verdana;
font-size:200%;

  }

</style>
</head>

<body>
  <h1>This is a heading</h1>

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

</html>
```

Result for the above Example :

EXTERNAL STYLING :

In External styling the same styles can be used across multiple pages of the site .

External styles are defined in the <head> section of an HTML page, in the <link> tag .

Example for External Styling :

```
<!DOCTYPE html>

<html>

<head>

<link rel="External"
href="style1.css">

</head>

<body>
<h1>This is a heading</h1>
```

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

</html>
```

Result for the above example :

CSS BOX :

This creates a box a around an HTML element , CSS border property defines a visible border around an element and padding property defines a padding space inside the border , margin property defines a margin space outside the border .

Example for CSS Box :

```
<!DOCTYPE html>
<html>

<head>
<style>
```

```
p {
    border:5px solid steelblue;
    padding:10px;
    margin:30px;
}

div {
    width: 320px;
    padding: 10px;
    border: 5px solid Turquoise;
    margin: 30px;
}

</style>
</head>
<body>

<p>CSS BOX EXAMPLE</p>

<div>

<p> This a Box </p>

</div>
```

```
</body>

</html>
```

Result for the above Example

NAVIGATION BAR

A navigation bar is a section of a graphical user interface intended to aid visitors in accessing information. Navigation bars are implemented in file browsers, web browsers and as a design element of some web sites

Vertival Navigation Bar

In the vertical navigation bar all the content will be in vertical format.

Example For Vertical Navigation Bar

```
<!DOCTYPE html>

<html>


<head>
```

```
<style>

ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
    width: 200px;
    background-color: #f1f1f1;
}

li a {
    display: block;
    color: #000;
    padding: 8px 0 8px 16px;
    text-decoration: none;
}

li a.active {
    background-color: steelblue;
    color: white;
}

li a:hover:not(.active) {
    background-color: turquoise;
    color: white;
}

</style>

</head>

<body>

<h2>Vertical Navigation Bar</h2>

<p>In this example, Active class is
in a Steelblue background color and
a white text.</p>

<ul>

<li><a class="active" href="#home">
Home</a></li>
<li><a href="#Content">Content</a>
</li>
<li><a href="#contact">Contact</a>
</li>
<li><a href="#about">About</a>
</li>
```

```
</ul>

</body>
</html>
```

Result for the above Example

Fixed Vertical Navigation Bar

In fixed Vertical Nave Bar, the nav bar will be fixed in the same position even if the content of the page is scrolled down.

Example for Fixed Vertical Nav Bar

```
<!DOCTYPE html>
<html>

<head>

<style>

body {
    margin: 0;
}

ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
    width: 26%;
    background-color: #f1f1f1;
    position: fixed;
    height: 100%;
    overflow: auto;
}

li a {
    display: block;
    color: #000;
    padding: 8px 0 8px 16px;
    text-decoration: none;
}

li a.active {
    background-color: steelblue;
    color: white;
}
```

```
li a:hover:not(.active) {

    background-color: turquoise;

    color: white;

}


</style>


</head>


<body>


<ul>

<li><a class="active" href="#home">

Home</a></li>

<li><a href="#news">News</a>

</li>

<li><a href="#contact">Contact</a>

</li>

<li><a href="#about">About</a>

</li>

</ul>


<div style="margin-left:26%;

padding:2px 16px;height:1000px;">
```

```
<h2>Fixed Side Nav Bar</h2>

<h3>Try to scroll this area, and
see how the sidenav sticks
to the page</h3>

<p>Notice that this div element
has a left margin of 26%. This
is because the side navigation
is set to 26% width. If you
remove the margin, the sidenav
will overlay/sit on top of
this div.</p>


<p>Also notice that we have set
overflow:auto to sidenav. This
will add a scrollbar when the
sidenav is too long (for
example if it has over 50 links
inside of it).</p>

  <p>Text 1</p>
  <p>Text 2</p>
  <p>Text 3</p>
  <p>Text 4</p>

  <p>Text 5</p>

  <p>Text 6</p>

  <p>Text 7</p>

  <p>Text 8</p>

  <p>Text 9</p>


</div>


</body>

</html>
```

Result for the above Example

Horizontal Navigation Bar

In the horizontal navigation bar all the content will be in horizontal format.

```
<!DOCTYPE html>

<html>

<head>

<style>

ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
    overflow: hidden;
    background-color: #333;
}

li {
    float: left;
}

li a {
    display: block;
    color: white;
    text-align: center;
    padding: 14px 16px;
    text-decoration: none;
}

li a:hover:not(.active) {
    background-color: turquoise;
}

.active {
    background-color: steelblue;
```

```
}
</style>
</head>
<body>
<h3> Horizontal Navigation Bar </h3>

<p>In this example, Active class is
in a Steelblue background color and
a white text.</p>

<ul>

<li><a class="active" href="#home">
Home</a></li>
<li><a href="#news">Content</a>
</li>
<li><a href="#contact">Contact</a>
</li>
<li><a href="#about">About</a>
</li>

</ul>
```

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

Result for the above Example

Fixed Horizontal Nav Bar

```
<!DOCTYPE html>
<html>

<head>

<style>

body {margin:0;}

ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
    overflow: hidden;
    background-color: #333;
    position: fixed;
    top: 0;
    width: 100%;
}

li {
    float: left;
}

li a {
    display: block;
    color: white;
    text-align: center;
    padding: 14px 16px;
    text-decoration: none;
}

li a:hover:not(.active) {
    background-color: turquoise;
}

.active {
    background-color: steelblue;
}
```

```
</style>
</head>
<body>

<ul>

<li><a class="active" href="#home">
Home</a></li>
<li><a href="#news">News</a>
</li>
<li><a href="#contact">Contact</a>
</li>
<li><a href="#about">About</a>
</li>

</ul>

<div style="padding:20px;margin-top:
30px;background-color:lightgrey;
height:1500px;">

<h1>Fixed Top Navigation Bar</h1>

<h2>Scroll this page to see the
effect</h2>
```

```
<h2> Nav bar will stay at
the top of the page while
scrolling</h2>

<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
<p>Some text </p>
```

```
</div>


</body>
</html>
```

Result for the above Example

DROPDOWN

A drop-down menu is a graphical control element, similar to a list box, that allows the user to choose one value from a list. When a drop-down list is inactive, it displays a single value. When activated, it displays (drops down) a list of values, from which the user may select one.

Example For Dropdown

```
<!DOCTYPE html>
<html>
<head>
<style>

.dropdown1 {
   position: relative;
   display: inline-block;
```

```
}

.dropdown1-content {
    display: none;
    position: absolute;
    background-color: lightgrey;
    min-width: 160px;
    box-shadow: 0px 8px 16px 0px
    rgba(0,0,0,0.2);
    padding: 12px 16px;
}

.dropdown1:hover .dropdown1-content {
    display: block;
}

.dropbtn {
    background-color: steelblue;
    color: white;
    padding: 16px;
    font-size: 16px;
    border: none;
    cursor: pointer;
}



.dropdown {
    position: relative;
    display: inline-block;
}

.dropdown-content {
    display: none;
    position: absolute;
    background-color: #f9f9f9;
    min-width: 160px;
    box-shadow: 0px 8px 16px 0px
    rgba(0,0,0,0.2);
}

.dropdown-content a {
    color: black;
    padding: 12px 16px;
    text-decoration: none;
    display: block;
```

```
}

.dropdown-content a:hover
{background-color: #f1f1f1}

.dropdown:hover .dropdown-content {
    display: block;
}

.dropdown:hover .dropbtn {
    background-color: turquoise;
}
</style>
</head>
<body>

<h2>Dropdown Menu</h2>
<p>Move the mouse over the button to
open the dropdown menu.</p>

<div class="dropdown">

<button class="dropbtn">Dropdown
</button>
```

```
<div class="dropdown-content">

<a href="#">Link 1</a>
<a href="#">Link 2</a>
<a href="#">Link 3</a>

</div>
</div>
```

```
<h2>Hoverable Dropdown</h2>
<p>Move the mouse over the text
below to open the dropdown
content.</p>

<div class="dropdown1">
  <span>Mouse over me</span>
  <div class="dropdown1-content">
    <p>Hoverable Dropdown</p>
  </div>
  </div>



</body>
</html>
```

Result for the above Example

IMAGE GALLERY

Example for Image Gallery

```
<!DOCTYPE html>
<html>

<head>

<style>
```

```
div.img {
    margin: 10px;
    border: 1px solid #ccc;
    float: left;
    width: 120px;
    height : 200px;
}

div.img:hover {
    border: 3px solid steelblue;
}

div.img img {
    width: 100%;
    height: 100px;
}

div.desc {
    padding: 15px;
    text-align: center;
}

</style>

</head>
```

```
<body>

<div class="img">

<a target="_blank" href=

"img_tiger.jpg">
<img src="img_tiger.jpg" alt="tiger"
width="300" height="200">

</a>

<div class="desc">This is a Tiger

</div>
</div>

<div class="img">

<a target="_blank" href=
"img_olm.jpg">
<img src="img_olm.jpg" alt="olm"
width="300" height="200">
</a>
```

```
<div class="desc">This is a Olm

</div>
</div>

<div class="img">

<a target="_blank" href=
"img_pygmy.jpg">
<img src="img_pygmy.jpg" alt="pygmy"
width="300" height="200">
</a>

<div class="desc">This is a Pygmy
Marmoset

</div>
</div>

<div class="img">

<a target="_blank" href=
"img_tarsier.jpg">
<img src="img_tarsier.jpg" alt=
"tarsier" width="300" height="200">
</a>

<div class="desc">This is Tarsier

</div>
</div>

</body>
</html>
```

Result for the above Example

FORMS

A form on a web page allows a user to enter data that is sent to a server for processing.

```
<!DOCTYPE html>
<html>

<head>
```

```
<style>

<!– This is for Padded field –>

input[type=text] {
    width: 100%;
    padding: 12px 20px;
    margin: 8px 0;
    box-sizing: border-box;
}

<!– This is for Bordered Field –>

input[type=text1] {
width: 100%;
padding: 12px 20px;
margin: 8px 0;
box-sizing: border-box;
border: 4px solid steelblue;
border-radius: 4px;
}


<!– This is for image Field –>
```

```
input[type=text2] {
width: 100%;
box-sizing: border-box;
border: 2px solid #ccc;
border-radius: 4px;
font-size: 16px;
background-color: white;
background-image: url
('sicon.png');
background-position: 10px 10px;
background-repeat: no-repeat;
padding: 12px 20px 12px 40px;
}


<!– This is for Animated Field –>


input[type=text3] {

width: 130px;
box-sizing: border-box;
border: 3px solid steelblue;
border-radius: 4px;
font-size: 16px;
background-color: white;
background-image: url('sicon.png');
background-position: 10px 10px;
background-repeat: no-repeat;
padding: 12px 20px 12px 40px;
-webkit-transition: width 0.4s
ease-in-out;
transition: width 0.4s ease-in-out;
}

input[type=text3]:focus {
    width: 60%;
}

</style>

</head>

<body>

<h3>Padded Text fields:</h3>

<form>
```

```
<label for="fname">First Name
</label>
<input type="text" id="fname"
name="fname">
<label for="fname">Last Name</label>
<input type="text" id="lname"
name="lname">

</form>




<h3> Bordered Text Fields</h3>




<form>

<label for="fname">First Name</label>
<input type="text1" id="fname"
name="fname">
<label for="lname">Last Name</label>
<input type="text1" id="lname"
```

```
name="lname">

</form>



<h3>Text Fields with icon/image </h3>

<form>

<input type="text2" name="search"
placeholder="Search..">

</form>




<h3> Animated Text Field </h3>


<form>

  <input type="text3" name="search"
placeholder="Search..">
```

```
</form>
```

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

Result for the above Example

Padded Text fields:

First Name Last Name

Bordered Text Fields

First Name Last Name

Text Fields with icon/image

Animated Text Field

STYLING TEXT AREA & SELECT MENU

```
<!DOCTYPE html>
<html>
<head>

<style>


select {
    width: 60%;
    padding: 16px 20px;
    border: 3px solid steelblue;
    background-color: turquoise;
}



textarea {
    width: 60%;
    height: 150px;
    padding: 12px 20px;
    box-sizing: border-box;
    border: 3px solid steelblue;
    border-radius: 10px;
    background-color: #f8f8f8;
    font-size: 16px;
    resize: none;
}

</style>

</head>
<body>

<h3>Styled Select Menu</h3>

<form>

<select id="country" name="country">
<option value="ind">India</option>
<option value="ind">China</option>
<option value="ind">Italy</option>
</select>

</form>
```

```
<h3> Styling Text Area </h3>

<form>

<textarea>

some text ..
some text ..
some text ..
some text ..
some text ..
some text ..
some text ..
some text ..
some text ..
some text ..
some text ..
some text ..


</textarea>
</form>
```

```
</body>

</html>
```

Result for the above Example

Styled Select Menu

Styling Text Area

ANIMATION&TRANSITION

Animation

An animation lets an element gradually change from one style to another, we can change as many CSS properties and many times as we want.

To use CSS3 animation, we must first specify some keyframes for the animation.Keyframes hold what styles the element will have at certain times.

When we specify CSS styles inside the @keyframes rule, the animation will gradually change from the current style to the new style at certain times.

To get an animation to work, we must bind the animation to an element.

```
<!DOCTYPE html>
```

```
<html>
<head>
<style>

div {
width: 200px;
height: 150px;
background-color: steelblue;
position: relative;
-webkit-animation-name: example;
-webkit-animation-duration: 4s;
-webkit-animation-iteration-count:3;
animation-name: example;
animation-duration: 4s;
animation-iteration-count: 3;
}


@-webkit-keyframes example {

0%  {background-color:steelblue;
left:0px; top:0px;}
25% {background-color:yellow;
left:200px; top:0px;}
50% {background-color:blue;
left:200px; top:200px;}
75% {background-color:green;
left:0px; top:200px;}
100%{background-color:orange;
left:0px; top:0px;}

}



</style>
</head>
<body>

<p><h3>ANIMATION</h3> </p>

<div></div>

</body>
</html>
```

Result for the above Example

Transition

Transitions allows you to change property values smoothly from one value to another, over a given duration. To create a transition effect, you must specify two things, the CSS property you want to add an effect to and the duration of the effect.If the duration part is not specified, the transition will have no effect, because the default value is 0.

```
<!DOCTYPE html>

<html>

<head>

<style>


#a {


    width: 100px;

    height: 100px;

    background: steelblue;

    -webkit-transition: width 2s;

    transition: width 2s;


}


#a:hover {

    width: 400px;

}
```

```
#b {

width: 100px;
height: 100px;
background: turquoise;
-webkit-transition: width 2s,
height 2s, -webkit-transform 2s;
transition: width 2s, height 2s,
transform 2s;

}

#b:hover {

width: 300px;
height: 300px;
-webkit-transform: rotate(180deg);
transform: rotate(180deg);

}


</style>
```

```
</head>

<body>

<h3> Transition  </h3>

<div id ="a">
</div>

<p>Hover over the div element above,
to see the transition effect.</p>

<div id ="b">
</div>

</body>
</html>
```

Result For Above Example

Transition

Hover over the div element above, to see the transition effect.

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

HTML LINKS :

A webpage can contain various links that take you directly to other pages and even specific parts of a given page , these links are known as hyperlinks . Hyperlinks allow visitors to navigate between Web sites by clicking on words, phrases, and images .

HTML link Syntax :

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

Example for HTML links :

```
<!DOCTYPE html>

<html>
```

```
<head>

<title>Hyperlink Example</title>

</head>

<body>

<p>Click following link</p>

<a href="http://www.html12app.com"
target="_self">HTML LEARN</a>

</body>

</html>
```

Result for the above Example

BASE PATH LINKS :

If you link HTML documents related to the same website, it is not required to give a complete URL for every link. You can get rid of it if you use tag in your HTML document header. This tag is used to give a base path for all the links.

Example for Base path links :

```
<!DOCTYPE html>

<html>

<head>

<title>Base Link Example</title>

<base href="http://www.html12.com/">

</head>

<body>

<p>Click following link</p>
<a href="/html/index.htm" target=
"_blank">LEARN HTML</a>

</body>
```

```
</html>
```

Result for the above Example

TARGET ATTRIBUTE :

This attribute is used to specify the location where linked document is opened

Attributes	Description
_blank	Opens the linked document in a new window or tab.
_self	Opens the linked document in the same frame.
_parent	Opens the linked document in the parent frame.
_top	Opens the linked document in the full body of the window.
targetframe	Opens the linked document in a named frame.

Example for Attributes :

```
<!DOCTYPE html>

<html>
```

```
<head>

<title>Attributes Example</title>

<base href=
"http://www.HTML12app.com/">

</head>

<body>

<p>Click the following links</p>

<a href="/html/index.htm" target=
"_blank">Opens in New tab</a>

<a href="/html/index.htm" target=
"_self">Opens in Self frame</a>

<a href="/html/index.htm" target=
"_parent">Opens in Parent frame</a>

<a href="/html/index.htm" target=
"_top">Opens in full Body</a>
```

```
</body>

</html>
```

Result for the above Example :

HTML IMAGE LINK :

In HTML to use an image as hyperlink the image should be used inside hyperlink .

Example for Image link :

```
<!DOCTYPE html>

<html>

<head>

<title>Image link Example</title>
```

```
</head>

<body>

<p>Click following link</p>

<a href="http://www.html12app.com"
target="_self">
<img src="/images/logo.png"
alt="LEARN HTML" border="0"/>
</a>

</body>

</html>
```

Result for the above Example :

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

HTML IMAGES :

There are images in every website now days it increases the visual look of the website , we will see how to use images in HTML .

Image Syntax :

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

1. In HTML, images are defined with the <img> tag.

2. The src attribute defines the url (location) of the image.

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

HTML image Example :

```
<!DOCTYPE html>

<html>

<body>
```

```
<h2>PICTURES OF PENGUINS</h2>

<img src="penguins.jpg"
alt="Penguins">

</body>

</html>
```

Result for above Example :

IMAGE HEIGHT & WIDTH :

In HTML , image width and height can be modified based on the requirement using width and height attributes.

Example for Image Height & Width :

```
<!DOCTYPE html>

<html>
```

```
<head>

<title>Set Image Width and Height
</title>

</head>

<body>

<p>Setting image width & height</p>
<img src="test.png" alt="TestImage"
width="200" height="100">

</body>

</html>
```

Result of the above Example :

SET IMAGE LOCATION :

By default, the browser expects to find the image in the same folder as the web page , if images is stored in a sub-folder it should be referred to that particular folder .

Example for Image location :

Assuming the image location is in "pic/penguins.png"

```
<!DOCTYPE html>

<html>

<head>

<title>IMAGE LOCATION</title>

</head>

<body>

<p>IMAGE LOCATION</p>
<img src="pic/penguins.png"
alt="PENGUINS" >

</body>

</html>
```

Result for the above Example :

IMAGE ALIGNMENT :

By default image will align at the left side of the page, but by using align attribute it can be set in the center or right.

Example for Image Alignment :

```
<!DOCTYPE html>

<html>

<head>

<title>Image Alignment</title>

</head>

<body>
```

```
<p>Setting image Alignment</p>

<img src="testimage.png"
alt="ImageAlignment" align="right">

</body>

</html>
```

Result for the above Example :

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

HTML TABLES :

The HTML tables allows to arrange data like text, images, links, other tables, etc. into rows and columns of cells.

1. Tables are defined with the <table> tag.

2. Tables are divided into table rows with the <tr> tag.

3. Table rows are divided into table data with the <td> tag.

4. A table row can also be divided into table headings with the <th>.

Example for HTML Tables :

```
<!DOCTYPE html>
<html>

<head>
<style>
table, th, td {
   border: 1px solid black;
   border-collapse: collapse;
}

</style>
</head>

<body>

<table style="width:50%">
```

```
  <tr>
    <th>Country</th>
    <th>capital</th>

  </tr>

  <tr>
    <td>India</td>
    <td>New delhi</td>

  </tr>

  <tr>
    <td>Germany</td>
    <td>Berlin</td>

  </tr>

  <tr>
    <td>New zealand</td>
    <td>Wellington</td>

  </tr>
</table>
```

```
</body>

</html>
```

Result for the above Example :

Table with Cell padding :

Cell padding specifies the space between the cell content and its borders, If not specified a padding, the table cells will be displayed without padding.

Example for Cell Padding :

```
<!DOCTYPE html>

<html>


<head>

<style>

table, th, td {

    border: 1px solid black;

    border-collapse: collapse;

}

th, td {
    padding: 15px;
}
</style>
</head>

<body>



<table style="width:50%">
  <tr>
    <th>Country</th>
    <th>capital</th>

  </tr>

  <tr>
    <td>India</td>
    <td>New delhi</td>

  </tr>

  <tr>
    <td>Germany</td>
    <td>Berlin</td>
  </tr>

  <tr>
    <td>New zealand</td>
    <td>Wellington</td>

  </tr>
</table>

<p>The cell padding is set
to 15px.</p>

</body>
</html>
```

Result for the above Example :

TABLE WITH BORDER SPACING :

Border spacing specifies the space between the cells.

Example for Border Spacing :

```
<!DOCTYPE html>
<html>

<head>
<style>
table, th, td {
    border: 1px solid black;
    padding: 5px;
}
table {
    border-spacing: 15px;
}
</style>
</head>

<body>

<table style="width:50%">
  <tr>
    <th>Country</th>
    <th>capital</th>
  </tr>

  <tr>
    <td>India</td>
    <td>New delhi</td>

  </tr>

  <tr>
    <td>Germany</td>
    <td>Berlin</td>

  </tr>

  <tr>
    <td>New zealand</td>
    <td>Wellington</td>

  </tr>
</table>



<p> border-spacing is set
to 15px.</p>
```

```
</body>

</html>
```

Result for the above Example :

CELL THAT SPAN MANY COLUMNS :

To make a cell span more than one column, colspan attribute should be used .

Example for Cell that span many columns :

```
<!DOCTYPE html>

<html>


<head>

<style>

table, th, td {

    border: 1px solid black;

    border-collapse: collapse;

}
```

```
th, td {
    padding: 5px;
    text-align: left;
}
</style>
</head>

<body>

<h2>Cell that spans two columns:</h2>
<table style="width:100%">
  <tr>
    <th>Name</th>
    <th colspan="2">Telephone</th>
  </tr>
  <tr>
    <td>xyz</td>
    <td>044 67 154</td>
    <td>91 72 455</td>
  </tr>
</table>

</body>
</html>
```

Result for the above Example :

CELL THAT SPAN MANY ROWS:

To make a cell span more than one row, use the rowspan attribute.

Example for cell that span many rows :

```
<!DOCTYPE html>
<html>

<head>
<style>
table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
}
th, td {
    padding: 5px;
    text-align: left;
}
</style>
</head>

<body>

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

<table style="width:100%">
  <tr>
    <th>Name:</th>
    <td>xyz</td>
  </tr>
  <tr>
    <th rowspan="2">Telephone:</th>
    <td>044 17 674</td>
  </tr>
  <tr>
    <td>91 12 6785</td>
  </tr>
</table>

</body>
</html>
```

Result for the above Example :

TABLE WITH CAPTION :

To add a caption to a table, use the <caption> tag.

Example for Table with caption :

```
<!DOCTYPE html>
<html>

<head>
<style>
table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
}
th, td {
    padding: 5px;
    text-align: left;
}
</style>
</head>
```

```
<body>

<table style="width:50%">

 <caption> CAPITALS </caption>

  <tr>
    <th>Country</th>
    <th>capital</th>

  </tr>

  <tr>
    <td>India</td>
    <td>New delhi</td>

  </tr>

  <tr>
    <td>Germany</td>
    <td>Berlin</td>

  </tr>
  <tr>
    <td>New zealand</td>
    <td>Wellington</td>

  </tr>
</table>

</body>
</html>
```

Result for the above Example :

DIFFERENT STYLES FOR DIFFERENT TABLES :

To define a different style for a different table, add an id attribute to the table .

Example for Different Styles :

```
<!DOCTYPE html>
<html>

<head>
<style>
table {
    width:50%;
}
table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
}
th, td {
    padding: 5px;
    text-align: left;
}
table#t01 tr:nth-child(even) {
    background-color: #6B9BC3;
}
table#t01 tr:nth-child(odd) {
   background-color:#6DDAD6 ;
}
table#t01 th {
    background-color: #6DDAD6 ;
    color: black;
}
</style>
</head>

<body>


<table id="t01">
  <tr>
    <th>First Name</th>
    <th>Last Name</th>

  </tr>
  <tr>
    <td>jack</td>
    <td>jck</td>

  </tr>
  <tr>
    <td>Eve</td>
    <td>Adam</td>

  </tr>
  <tr>
    <td>John</td>
    <td>wee</td>

  </tr>
</table>

</body>
</html>
```

Result for the above Example :

TABLE COLOR

To define color for the borders, and the text and background color of elements

Example for Table color :

```
<!DOCTYPE html>
<html>
<head>
<style>
table, td, th {
    border: 1px solid steelblue;
}
```

```
th {
    background-color: #6DDAD6;
    color: black;
}
</style>
</head>

<body>

<table>
  <tr>
    <th>First Name</th>
    <th>Last Name</th>

  </tr>
  <tr>
    <td>jack</td>
    <td>jck</td>

  </tr>
  <tr>
    <td>Eve</td>
    <td>Adam</td>
  </tr>
  <tr>
    <td>John</td>
    <td>wee</td>

  </tr>
</table>

</body>
</html>
```

Result for the above Example :

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

HTML TABLES :

The HTML tables allows to arrange data like text, images, links, other tables, etc. into rows and columns of cells.

1. Tables are defined with the <table> tag.

2. Tables are divided into table rows with the <tr> tag.

3. Table rows are divided into table data with the <td> tag.

4. A table row can also be divided into table headings with the <th>.

Example for HTML Tables :

```
<!DOCTYPE html>
<html>

<head>
<style>
table, th, td {
   border: 1px solid black;
   border-collapse: collapse;
}

</style>
</head>
```

```
<body>

<table style="width:50%">
  <tr>
    <th>Country</th>
    <th>capital</th>

  </tr>

  <tr>
    <td>India</td>
    <td>New delhi</td>

  </tr>

  <tr>
    <td>Germany</td>
    <td>Berlin</td>

  </tr>

  <tr>
    <td>New zealand</td>
    <td>Wellington</td>
  </tr>
</table>

</body>
</html>
```

Result for the above Example :

Table with Cell padding :

Cell padding specifies the space between the cell content and its borders, If not specified a padding, the table cells will be displayed without padding.

Example for Cell Padding :

```
<!DOCTYPE html>
<html>

<head>
<style>
table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
}
th, td {
    padding: 15px;
}
</style>
</head>

<body>



<table style="width:50%">
  <tr>
    <th>Country</th>
    <th>capital</th>

  </tr>

  <tr>
    <td>India</td>
    <td>New delhi</td>

  </tr>

  <tr>
```

```
    <td>Germany</td>
    <td>Berlin</td>

  </tr>

  <tr>
    <td>New zealand</td>
    <td>Wellington</td>

  </tr>
</table>

<p>The cell padding is set
to 15px.</p>

</body>
</html>
```

Result for the above Example :

TABLE WITH BORDER SPACING :

Border spacing specifies the space between the cells.

Example for Border Spacing :

```
<!DOCTYPE html>
<html>

<head>
<style>
table, th, td {
    border: 1px solid black;
    padding: 5px;
}
table {
    border-spacing: 15px;
}
</style>
</head>

<body>

<table style="width:50%">
  <tr>
```

```
    <th>Country</th>
    <th>capital</th>

  </tr>

  <tr>
    <td>India</td>
    <td>New delhi</td>

  </tr>

  <tr>
    <td>Germany</td>
    <td>Berlin</td>

  </tr>

  <tr>
    <td>New zealand</td>
    <td>Wellington</td>

  </tr>
</table>
```

```
<p> border-spacing is set
to 15px.</p>

</body>
</html>
```

Result for the above Example :

CELL THAT SPAN MANY COLUMNS :

To make a cell span more than one column, colspan attribute should be used .

Example for Cell that span many columns :

```
<!DOCTYPE html>
<html>

<head>
<style>
table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
}
th, td {
    padding: 5px;
    text-align: left;
}
</style>
</head>

<body>

<h2>Cell that spans two columns:</h2>
<table style="width:100%">
  <tr>
    <th>Name</th>
    <th colspan="2">Telephone</th>
  </tr>
  <tr>
    <td>xyz</td>
    <td>044 67 154</td>
    <td>91 72 455</td>
  </tr>
</table>
```

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

Result for the above Example :

CELL THAT SPAN MANY ROWS:

To make a cell span more than one row, use the rowspan attribute.

Example for cell that span many rows :

```
<!DOCTYPE html>
<html>

<head>
<style>
table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
}
```

```
th, td {
    padding: 5px;
    text-align: left;
}
</style>
</head>

<body>

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

<table style="width:100%">
  <tr>
    <th>Name:</th>
    <td>xyz</td>
  </tr>
  <tr>
    <th rowspan="2">Telephone:</th>
    <td>044 17 674</td>
  </tr>
  <tr>
    <td>91 12 6785</td>
  </tr>
</table>
```

```
</body>

</html>
```

Result for the above Example :

TABLE WITH CAPTION :

To add a caption to a table, use the <caption> tag.

Example for Table with caption :

```
<!DOCTYPE html>

<html>


<head>

<style>

table, th, td {

    border: 1px solid black;

    border-collapse: collapse;

}

th, td {

    padding: 5px;

    text-align: left;
}

</style>

</head>


<body>


<table style="width:50%">


 <caption> CAPITALS </caption>


  <tr>

    <th>Country</th>

    <th>capital</th>


  </tr>


  <tr>

    <td>India</td>

    <td>New delhi</td>


  </tr>


  <tr>

    <td>Germany</td>

    <td>Berlin</td>
  </tr>

  <tr>
    <td>New zealand</td>
    <td>Wellington</td>

  </tr>
</table>






</body>
</html>
```

Result for the above Example :

DIFFERENT STYLES FOR DIFFERENT TABLES :

To define a different style for a different table, add an id attribute to the table .

```
}
```

Example for Different Styles :

```
<!DOCTYPE html>
<html>

<head>
<style>
table {
    width:50%;
}
table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
}
th, td {
    padding: 5px;
    text-align: left;
}
table#t01 tr:nth-child(even) {
    background-color: #6B9BC3;
}
table#t01 tr:nth-child(odd) {
   background-color:#6DDAD6 ;
}
```

```
table#t01 th {
    background-color: #6DDAD6 ;
    color: black;
}
</style>
</head>

<body>


<table id="t01">
  <tr>
    <th>First Name</th>
    <th>Last Name</th>

  </tr>
  <tr>
    <td>jack</td>
    <td>jck</td>

  </tr>
  <tr>
    <td>Eve</td>
    <td>Adam</td>
  </tr>
  <tr>
    <td>John</td>
    <td>wee</td>

  </tr>
</table>

</body>
</html>
```

Result for the above Example :

TABLE COLOR

To define color for the borders, and the text and background color of elements

Example for Table color :

```
<!DOCTYPE html>
<html>
<head>
<style>
```

```
table, td, th {
    border: 1px solid steelblue;
}

th {
    background-color: #6DDAD6;
    color: black;
}
</style>
</head>

<body>

<table>
  <tr>
    <th>First Name</th>
    <th>Last Name</th>

  </tr>
  <tr>
    <td>jack</td>
    <td>jck</td>

  </tr>
  <tr>
```

```
    <td>Eve</td>
    <td>Adam</td>

  </tr>
  <tr>
    <td>John</td>
    <td>wee</td>

  </tr>
</table>

</body>
</html>
```

Result for the above Example :

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

HTML LIST :

HTML offers three types of lists they are :

1.Unordered list: This can be created using <ul> tag,this will list items using plain bullets.

2.Ordered list: This can be created usingn<ol> tag,this will use different schemes of numbers to list items.

3.Definition list: This can be created using <dl> tag,this arranges your items in the same way as they are arranged.

UNORDERED LIST :

An unordered list is a collection of items that have no special order or sequence.This list is created by using HTML<ul> tag and each list starts with <li> tag , each item in the list is marked with a bullet .

Example for Unordered List :

```
<!DOCTYPE html>

<html>

<head>
```

```
<title>Unordered List</title>

</head>

<body>
<ul>
<li>Tiger</li>
<li>cheetah</li>
<li>jaguar</li>
<li>Lion</li>
</ul>
</body>

</html>
```

Result for the above Example :

UNORDERED TYPE ATTRIBUTE :

You can use type attribute for <ul> tag to specify the type of bullet you like,by default it is a disc the options are circle , square , none .

1. <ul type="square">

2. <ul type="disc">

3. <ul type="circle">

4. <ul type="none">

Example for Type Attribute :

```
<!DOCTYPE html>
<html>
<head>
<title>TYPE ATTRIBUTE</title>
</head>
<body>

<h4>Square attribute example </h4>

  <ul type="square">
  <li>Tiger</li>
  <li>Lion</li>

  </ul>
```

```
<h4>Disc attribute example </h4>

<ul type="disc">
  <li>Tiger</li>
  <li>Lion</li>

  </ul>

<h4>circle attribute example </h4>

<ul type="circle">
  <li>Tiger</li>
  <li>Lion</li>

  </ul>

<h4>none attribute example </h4>

<ul type="none">
  <li>Tiger</li>
  <li>Lion</li>

  </ul>
```

```
</body>

</html>
```

Result for above Example :

ORDERED LIST :

An Ordered list is a list of items in a numbered list.This list is created by using <ol> tag. The numbering starts at one and is incremented by one for each successive ordered list element tagged with <li>.

Example for Ordered List :

```
<!DOCTYPE html>

<html>

<head>

<title>Ordered List</title>

</head>
```

```
<body>

<ol>


<li>Tiger</li>

<li>cheetah</li>

<li>jaguar</li>

<li>Lion</li>


</ol>

</body>


</html>
```

Result for above Example :

ORDERED TYPE ATTRIBUTE :

Type attribute can be used for <ol> tag to specify the type of numbering by default it is a number .

1. <ol type="1"> - Default Numerals.

2. <ol type="T"> - Upper-Case Numerals.

3. <ol type="i"> - Lower-Case Numerals.

4. <ol type="a"> - Lower-Case Letters.

5. <ol type="A"> - Upper-Case Letters.

Example for Ordered type Attribute :

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>

<h4> Default Numerals </h4>

  <ol type="1">
  <li>Tiger</li>
  <li>Lion</li>
  </ol>

<h4>Upper-Case Numerals</h4>
```

```
<ol type="T">
   <li>Tiger</li>
   <li>Lion</li>
   </ol>

<h4> Lower-Case Numerals </h4>

<ol type="i">
   <li>Tiger</li>
   <li>Lion</li>
   </ol>

<h4>Lower-Case Letters </h4>

<ol type="a">
   <li>Tiger</li>
   <li>Lion</li>
   </ol>

<h4>Upper-Case Letters </h4>

<ol type="A">
   <li>Tiger</li>
   <li>Lion</li>
   </ol>
```

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

Result for above Example :

DISCRIPTION LIST :

A description list, is a list of items, with a description of each item.

1. The <dl> tag defines a list.

2. The <dt> tag defines the Item name.

3. The <dd> tag defines the description.

Example for Description List :

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

```
<h2> Description List</h2>

<dl>
<dt>Tiger</dt>
<dd>Tiger is Endangered</dd>
<dt>Lion</dt>
<dd>Lion is the king of jungle</dd>
</dl>

</body>
</html>
```

Result for above Example :

HORIZONTAL LIST :

HTML lists can be styled in many different types , horizontal list is one of the type .

Example for Horizontal list :

```
<!DOCTYPE html>
```

```
<html>

<head>

<style>

ul#list li {
    display:inline;
}

</style>

</head>

<body>

<h2>Horizontal List</h2>

<ul id="list">
<li>Tiger</li>
<li>cheetah</li>
<li>jaguar</li>
<li>Lion</li>
</ul>
```

```
</body>

</html>
```

Result for above Example :

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

HTML LAYOUT :

A webpage layout gives a structure to the website , layout is very important to give better look to the website .

WEB SITE LAYOUT :

Header : Header Defines a header for a document or a section .

Navigation : Navigation Defines a container for navigation links .

Content : Defines a section of the page with document or article and other information .

Sidebar : Defines content aside from the main content of the page .

Footer : Defines a footer for a document or a section .

LAYOUT USING div ELEMENT :

The div element is used in this layout
it is a block level element used for
grouping HTML elements.
Example for div Element :

```
<!DOCTYPE html>
<html>

<head>
<style>
#header {
   background-color:steelblue;
   color:white;
```

```
    text-align:center;
    padding:5px;
}
#nav {
    line-height:30px;
    background-color:#48D1CC;
    color:white;
    height:350px;
    width:110px;
    float:left;
    padding:5px;
}
#section {
    width:350px;
    float:left;
    padding:10px;
}
#footer {
    background-color:steelblue;
    color:white;
    clear:both;
    text-align:center;
   padding:5px;
}
</style>
```

```
</head>

<body>

<div id="header">
<h1>BIG CATS</h1>
</div>

<div id="nav">
Tiger<br>
Lion<br>
Cheetah<br>
</div>

<div id="section">

<h2>Tiger</h2>

<p>
The tiger (Panthera tigris)
is the largest cat species,
reaching a total body length
of up to 3.38 m (11.1 ft) over
curves and weighing up to 388.7
kg (857 lb) in the wild.
```

```
<p>
The tiger populations
occurring in small pockets
isolated from each other, of
which about 2,000 exist on the
Indian subcontinent.
</p>

</div>

<div id="footer">
Copyright  2015
</div>

</body>
</html>
```

Result for above Example :

LAYOUT USING TABLES :

The layout using Tables are the simplest and most popular way of creating layouts . These tables are arranged in columns and rows, so it can utilized in many ways according to the need

.

Example for Layout using Tables :

```
<!DOCTYPE html>
<html>
<head>
<title>Layout using Tables</title>
</head>
<body>



<table width="100%" border="0">
<tr>
<td colspan="2"bgcolor="steelblue">
<center>
<h1>BIG CATS</h1>
</center>
</td>
</tr>


<tr valign="top">
```

```
<td bgcolor="#48D1CC" width="50">

<b>Main Menu</b><br/>

Tiger<br/>

Lion<br/>

Cheetah

</td>


<td bgcolor="#eee" width="100"

height="200">

The tiger (Panthera tigris)

is the largest cat species

</td>


</tr>




<tr>

<td colspan="2"bgcolor="steelblue">

<center>

Copyright  2015

</center>

</td>

</tr>
```

```
</table>

</body>

</html>
```

Result for above Example :

LAYOUT USING HTML5 :

HTML5 offers new semantic and dynamic elements that define different parts of a web page with a modern look and feel .

Example for Layout using HTML5 :

```
<!DOCTYPE html>

<html>


<head>

<style>

header {

   background-color:steelblue;

   color:white;

   text-align:center;

   padding:5px;

}
```

```
nav {
   line-height:30px;
   background-color:#48D1CC;
   height:300px;
   width:100px;
   float:left;
   padding:5px;
}
section {
   width:350px;
   float:left;
   padding:10px;
}
footer {
   background-color:steelblue;
   color:white;
   clear:both;
   text-align:center;
   padding:5px;
}
</style>
</head>

<body>
```

```
<header>

<h1>BIG CATS</h1>

</header>

<nav>

Tiger<br>
Lion<br>
Cheetah<br>

</nav>

<section>

<h1>Tiger</h1>

<p>
The tiger (Panthera tigris) is the
largest cat species, reaching a
total body length of up to 3.38 m
(11.1 ft) over curves and weighing
up to 388.7 kg (857 lb) in the wild.
</p>
```

```
</section>

<footer>

Copyright  2015

</footer>

</body>
</html>
```

Result for above Example :

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

HTML LAYOUT :

A webpage layout gives a structure to the website , layout is very important to give better look to the website .

WEB SITE LAYOUT :

Header : Header Defines a header for a document or a section .

Navigation : Navigation Defines a container for navigation links .

Content : Defines a section of the page with document or article and other information .

Sidebar : Defines content aside from the main content of the page .

Footer : Defines a footer for a document or a section .

LAYOUT USING div ELEMENT :

The div element is used in this layout
  it is a block level element used for
  grouping HTML elements.

Example for div Element :

```
<!DOCTYPE html>
```

```
<html>


<head>

<style>

#header {

   background-color:steelblue;

   color:white;

   text-align:center;

   padding:5px;

}

#nav {

   line-height:30px;

   background-color:#48D1CC;

   color:white;

   height:350px;

   width:110px;

   float:left;

   padding:5px;

}

#section {

   width:350px;

   float:left;

   padding:10px;

}

#footer {
    background-color:steelblue;

    color:white;

    clear:both;

    text-align:center;

   padding:5px;

}

</style>

</head>


<body>


<div id="header">

<h1>BIG CATS</h1>

</div>


<div id="nav">

Tiger<br>

Lion<br>

Cheetah<br>

</div>


<div id="section">


<h2>Tiger</h2>
```

```
<p>
The tiger (Panthera tigris)
is the largest cat species,
reaching a total body length
of up to 3.38 m (11.1 ft) over
curves and weighing up to 388.7
kg (857 lb) in the wild.

<p>
The tiger populations
occurring in small pockets
isolated from each other, of
which about 2,000 exist on the
Indian subcontinent.
</p>

</div>

<div id="footer">
Copyright  2015
</div>

</body>
</html>
```

Result for above Example :

LAYOUT USING TABLES :

The layout using Tables are the simplest and most popular way of creating layouts . These tables are arranged in columns and rows, so it can utilized in many ways according to the need .

Example for Layout using Tables :

```
<!DOCTYPE html>

<html>

<head>

<title>Layout using Tables</title>

</head>

<body>




<table width="100%" border="0">

<tr>

<td colspan="2"bgcolor="steelblue">

<center>
```

```
<h1>BIG CATS</h1>
</center>
</td>
</tr>


<tr valign="top">

<td bgcolor="#48D1CC" width="50">
<b>Main Menu</b><br/>
Tiger<br/>
Lion<br/>
Cheetah
</td>

<td bgcolor="#eee" width="100"
height="200">
The tiger (Panthera tigris)
is the largest cat species
</td>

</tr>


<tr>
```

```
<td colspan="2"bgcolor="steelblue">

<center>

Copyright  2015

</center>

</td>

</tr>


</table>

</body>

</html>
```

Result for above Example :

LAYOUT USING HTML5 :

HTML5 offers new semantic and dynamic elements that define different parts of a web page with a modern look and feel .

Example for Layout using HTML5 :

```
<!DOCTYPE html>

<html>


<head>
```

```
<style>
header {
    background-color:steelblue;
    color:white;
    text-align:center;
    padding:5px;
}
nav {
    line-height:30px;
    background-color:#48D1CC;
    height:300px;
    width:100px;
    float:left;
    padding:5px;
}
section {
    width:350px;
    float:left;
    padding:10px;
}
footer {
    background-color:steelblue;
    color:white;
    clear:both;
    text-align:center;
    padding:5px;
}
</style>
</head>

<body>

<header>

<h1>BIG CATS</h1>

</header>

<nav>

Tiger<br>
Lion<br>
Cheetah<br>

</nav>

<section>

<h1>Tiger</h1>
```

```
<p>
The tiger (Panthera tigris) is the
largest cat species, reaching a
total body length of up to 3.38 m
(11.1 ft) over curves and weighing
up to 388.7 kg (857 lb) in the wild.
</p>

</section>

<footer>

Copyright  2015

</footer>

</body>
</html>
```

Result for above Example :

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

HTML LAYOUT :

A webpage layout gives a structure to the website , layout is very important to give better look to the website .

WEB SITE LAYOUT :

Header : Header Defines a header for a document or a section .

Navigation : Navigation Defines a container for navigation links .

Content : Defines a section of the page with document or article and other information .

Sidebar : Defines content aside from the main content of the page .

Footer : Defines a footer for a document or a section .

LAYOUT USING div ELEMENT :

The div element is used in this layout
it is a block level element used for
grouping HTML elements.

Example for div Element :

```
<!DOCTYPE html>
<html>

<head>
<style>
#header {
   background-color:steelblue;
   color:white;
   text-align:center;
   padding:5px;
}
#nav {
   line-height:30px;
   background-color:#48D1CC;
   color:white;
   height:350px;
   width:110px;
   float:left;
   padding:5px;
}
#section {
    width:350px;
    float:left;
    padding:10px;
}
#footer {
    background-color:steelblue;
    color:white;
    clear:both;
    text-align:center;
   padding:5px;
}
</style>
</head>

<body>

<div id="header">
<h1>BIG CATS</h1>
</div>

<div id="nav">
Tiger<br>
Lion<br>
```

```
Cheetah<br>
</div>

<div id="section">

<h2>Tiger</h2>

<p>
The tiger (Panthera tigris)
is the largest cat species,
reaching a total body length
of up to 3.38 m (11.1 ft) over
curves and weighing up to 388.7
kg (857 lb) in the wild.

<p>
The tiger populations
occurring in small pockets
isolated from each other, of
which about 2,000 exist on the
Indian subcontinent.
</p>

</div>
```

```
<div id="footer">
Copyright  2015
</div>

</body>
</html>
```

Result for above Example :

LAYOUT USING TABLES :

The layout using Tables are the simplest and most popular way of creating layouts . These tables are arranged in columns and rows, so it can utilized in many ways according to the need .

Example for Layout using Tables :

```
<!DOCTYPE html>
<html>
<head>
<title>Layout using Tables</title>
</head>
<body>
```

```
<table width="100%" border="0">
<tr>
<td colspan="2"bgcolor="steelblue">
<center>
<h1>BIG CATS</h1>
</center>
</td>
</tr>

<tr valign="top">

<td bgcolor="#48D1CC" width="50">
<b>Main Menu</b><br />
Tiger<br />
Lion<br />
Cheetah
</td>

<td bgcolor="#eee" width="100"
height="200">
The tiger (Panthera tigris)
```

```
is the largest cat species

</td>


</tr>




<tr>

<td colspan="2"bgcolor="steelblue">

<center>

Copyright  2015

</center>

</td>

</tr>


</table>

</body>

</html>
```

Result for above Example :

LAYOUT USING HTML5 :

HTML5 offers new semantic and dynamic elements that define different parts of a web page with a modern look and feel .

Example for Layout using HTML5 :

```
<!DOCTYPE html>
<html>

<head>
<style>
header {
   background-color:steelblue;
   color:white;
   text-align:center;
   padding:5px;
}
nav {
   line-height:30px;
   background-color:#48D1CC;
   height:300px;
   width:100px;
   float:left;
   padding:5px;
}
section {
   width:350px;
   float:left;
```

```
    padding:10px;
}
footer {
    background-color:steelblue;
    color:white;
    clear:both;
    text-align:center;
    padding:5px;
}
</style>
</head>

<body>

<header>

<h1>BIG CATS</h1>

</header>

<nav>

Tiger<br>
Lion<br>
Cheetah<br>
```

```
</nav>

<section>

<h1>Tiger</h1>

<p>
The tiger (Panthera tigris) is the
largest cat species, reaching a
total body length of up to 3.38 m
(11.1 ft) over curves and weighing
up to 388.7 kg (857 lb) in the wild.
</p>

</section>

<footer>

Copyright  2015

</footer>

</body>
</html>
```

Result for above Example :

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

HTML IFRAMES :

An IFrame (Inline Frame) is an HTML document embedded inside another HTML document on a website , which is displaying a web page within a web page. The IFrame HTML element is often used to insert content from another source ,it can be configured with its own scrollbar independent of the surrounding page's scrollbar.

Syntax :

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

IFRAME HEIGHT & WIDTH :

Use the height and width attributes to specify the size of the IFRAMES ,the attribute values are specified in pixels by default, but they can also be in percent.

Example for HEIGHT & WIDTH :

```
<!DOCTYPE html>

<html>

<body>

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

</body>

</html>
```

Result for the above Example :

IFRAME EDITING THE BORDER :

The border of the Iframe can be edited and changed according to the needs , By default, an iframe has a black border around it . The Iframe border size , color and style can be changed with css .

Example for IFRAME BORDER EDITING :

```
<!DOCTYPE html>

<html>

<body>

<iframe src="edit_exiframe.htm"
style="border:4px outset red">
</iframe>

</body>

</html>
```

Result for the above Example :

USING IFRAME AS TARGET AS LINK :

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

Example for IFRAME as Target Link :

```
<!DOCTYPE html>

<html>

<body>

<iframe width="100%" height="300px"
src="d_iframe.htm" name="iframe_ex">
</iframe>

<a href="http://www.html12app.com
/demo12" target="iframe_ex">
IFRAME-TARGET</a>

<p>When the target of a link matches
the name of an iframe, the link will
open in the iframe.</p>

</body>
```

```
</html>
```

Result for the above Example :

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

HTML JAVASCRIPTS :

A script is a small piece of program that can add interactivity to the website. For example, a script could generate a pop-up alert box message, or provide a dropdown menu. This script could be written using Javascript or VBScript , various small functions, called event handlers can be written using any of the scripting language and then you can trigger those functions using HTML attributes.

The <script> tag is used to define a clientside script, such as a JavaScript.

EXTERNAL & INTERNAL SCRIPTS :

External Javascripts :

If the functionality to be defined is used in various HTML documents then it's better to keep that functionality in a separate Javascript file and then include that file in your HTML documents. A Javascript file will have extension as .js and it will be included in HTML files using script tag.

Example for External Javascript :

```
<!DOCTYPE html>

<html>

<head>

<title>External Script</title>

<script src="/html/External.js"
type="text/javascript"/>

</script>

</head>

<body>
<h2>Example for External Script</h2>

<input type="button"
```

```
onclick="External();

" value="Click here"  />


</body>


</html>
```

Result for the above Example :

Internal Javascripts :

The script code can be written directly into the HTML document. script code is placed in header of the document using script tag .

Example for Internal JavaScript :

```
<!DOCTYPE html>


<html>


<head>


<title>Internal Script</title>
```

```
<script type="text/javascript">
function internal(){
alert("Internal Script");
}
</script>

</head>

<body>

<h2>Example for Internal script</h2>

<input type="button"
onclick="internal();
"value="Click here" />

</body>

</html>
```

Result for the above Example :

EVENT HANDLER :

Event handlers are simple defined functions which can be called against any mouse or keyboard event.

Example for Event handler

```
<!DOCTYPE html>
<html>

<head>

<title>Event Handlers </title>

<script type="text/javascript">
function EventHandler(){
 alert("Event Handler Example");
}
</script>

</head>

<body>
```

```
<p onmouseover="EventHandler();">

Bring your mouse here to see

the alert message</p>


</body>


</html>
```

Result for the above Example :

THE NOSCRIPT TAG :

The Noscript tag provide alternative info to the users whose browsers don't support scripts and for those users who have disabled script option their browsers. This can be done using the Noscript tag

Example for Noscript tag :

```
<!DOCTYPE html>
```

```
<html>

<body>

<p id="demo"></p>

<script>
document.getElementById("demo")
.innerHTML = "NOSCRIPT TAG!";
</script>

<noscript> Sorry, your browser
does not support JavaScript!
</noscript>

<p>A browser without support for
JavaScript will show the text
written inside the noscript
element.</p>

</body>

</html>
```

Result for the above Example :

JAVASCRIPT VARIABLES

JavaScript variables are containers for storing data values. Variable should be assigned some values The (=) equal sign is called assignment operator . In JavaScript, the equal sign (=) is an "assignment" operator, not an "equal to" operator, you assign the value of what is on the right side of the = sign to whatever is on the left side of the = sign, you cannot perform operations with empty variables.

JavaScript variables can hold numbers like 200 , and text values like "Tiger".

In programming, text values are called text strings.

Strings are written inside double or single quotes. Numbers are written without quotes.

If you put quotes around a number, it will be treated as a text string.

JAVASCRIPT IDENTIFIERS

All JavaScript variables must be identified with unique names.

These unique names are called identifiers. Identifiers can be short names (like x and y), or more descriptive names (age, sum, totalVolume). The general rules for constructing names for variables (unique identifiers) are:

Names can contain letters, digits, underscores, and dollar signs.

Names must begin with a letter.

Names can also begin with $ and @.

Names are case sensitive (L and l are different variables) Reserved words (like JavaScript keywords) cannot be used as names.

JAVASCRIPT OPERATORS

OPERATOR	DISCRIPTION
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus
++	Increment
--	Decrement
==	equal to
===	equal value and equal type
!=	not equal
!==	not equal value
>	greater than
<	less than
>=	greater than or equal to
<=	less than or equal to
?	ternary operator

Example for Javascript Variables

```
<!DOCTYPE html>


<html>
```

```
<body>

<h1>JavaScript Variables</h1>



<p>In this example, L, M, N, O are
variables</p>

<p id="sample"></p>

<script>

var L = 4;
var M = 6;
var N = L + M;
var O = L * M;
document.getElementById("sample")
.innerHTML = O;

</script>



<p>You can declare many variables in
one statement, Start the statement
```

```
with var and separate the variables
by comma.</p>


<p id="sample 1"></p>

<script>

var Animal = "tiger", weight = 200 ;
document.getElementById("sample 1")
.innerHTML = Animal + " weight is "
+ weight + " kg ";

</script>

</body>

</html>
```

Result for the above Example :

JAVASCRIPT FUNCTION

A JavaScript function is a block of code designed to perform a particular task ,a JavaScript function is executed when "something" invokes it (calls it).

When JavaScript reaches a return statement, the function will stop executing, if the function was invoked from a statement, JavaScript will "return" to execute the code after the invoking statement.

Javascript Function Syntax

```
function name(parameter1, parameter2){

code to be executed }
```

A JavaScript function is defined with the function keyword, followed by a name, followed by parentheses ().

Function names can contain letters, digits, underscores, and dollar signs .

The parentheses may include parameter names separated by commas: (parameter1, parameter2, parameter3 ...)

The code to be executed, by the function, is placed inside curly brackets {}

Example For JavaScript Function

```
<!DOCTYPE html>

<html>

<body>


<p>This example calls a function

which performs a calculation,
and returns the result:</p>

<p id="sample"></p>

<script>

function myFunction(L, M) {
   return L * M;
}
document.getElementById("sample")
.innerHTML = myFunction(6, 3);

</script>



<p>This example calls a function to
convert from Seconds to Minutes:</p>
<p id="sample1"></p>

<script>

function tominutes(f) {
   return f/60+" minutes ";
}
document.getElementById("sample1")
```

```
.innerHTML = tominutes(120);

</script>


</body>
</html>
```

Result for the above Example :

Boolean

The Boolean object represents two values, either "true" or "false", intended to represent the truth values of logic, very often in programming, you will need a data type that can only have one of two values, like YES / NO , ON / OFF , TRUE / FALSE.

Everything with a "Real" value is True

Everything without a "Real" value is False

The Boolean value of 0 (zero) is false

The Boolean value of -0 (minus zero) is false

The Boolean value of "" (empty string) is false

The Boolean value of undefined is false

The Boolean value of null is false

The Boolean value of NaN is false

Example for Boolean

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

<p>Display the value of Boolean
(8 > 3):</p>

<button onclick="myBoolean()">Try it
</button>

<p id="sample"></p>

<script>
function myBoolean() {
document.getElementById("sample")
.innerHTML = Boolean(8 > 3);
}
</script>
```

```
<p>Display the Boolean value of 0</p>

<button onclick="myboolean()">Try it
</button>

<p id="sample1"></p>

<script>
function myboolean() {
  var x = 0;

document.getElementById("sample1")
.innerHTML = Boolean(x);
}
</script>

</body>
</html>
```

Result for above Example :

Display the value of Boolean (8 > 3):

Try it

Display the Boolean value of 0

Try it

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

JAVASCRIPT CONDITIONS

Conditional statements are used to perform different actions based on different conditions , you need to use conditional statements that allow your program to make correct decisions and perform right actions.

In JavaScript we have the following conditional statements:

Use if to specify a block of code to be executed, if a specified condition is true

Use else to specify a block of code to be executed, if the same condition is false

Use else if to specify a new condition to test, if the first condition is false

Use switch to specify many alternative blocks of code to be executed

if Statement

if statement is used to specify a JavaScript code to be executed if a condition is true , here a JavaScript expression is evaluated , If the resulting value is true then the given statements are executed. If the expression is false, then no statement would be not executed.

if Statement Syntax

```
if (condition) {

code to be executed if the condition is true }
```

else Statement

The else statement is used to specify a Javascript code to be executed if the condition is false.

else Statement Syntax

```
if (condition) {

code to be executed if the condition is true } else {

code to be executed if the condition is false }
```

else if Statement

The else if statement is used to specify a new condition , if the first condition is false , the Else If statement is an extension to the If Statement that allows you to create as many conditional statements as you want .

else if Syntax

```
if (condition1) {

code to be executed if condition1 is true } else if (condition2) {

code to be executed if the condition1 is false and condition2 is true }

else {

code to be executed if the condition1 is false and condition2 is false }
```

Exampe for javaScript Conditions

```
<!DOCTYPE html>

<html>

<body>



<h3>This is the example of if

Statement </h3>


<p>Click the button to know your

driving eligibility</p>
```

```
<button onclick="myFunction()">
Try it</button>

<p id="demo"></p>

<script>

function myFunction() {

var message;
var age = 23 ;

if (age > 19) {
message="you are eligible to drive";
}

document.getElementById("demo")
.innerHTML = message;

}

</script>
```

```
<h3>This is the example of else

Statement </h3>


<p>Click the button to know your

driving eligibility</p>


<button onclick="myFunction1()">

Try it</button>


<p id="demo1"></p>


<script>


function myFunction1() {


var message1;

var age = 1;


if (age > 19) {

message1="you are eligible to drive";


} else {

message1= "not eligible to drive";
}

document.getElementById("demo1")
.innerHTML = message1;
}

</script>

<h3>This is the example of else if
Statement </h3>

<p>Click the button to know your
driving eligibility</p>

<button onclick="myFunction2()">
Try it</button>

<p id="demo2"></p>

<script>

function myFunction2() {
```

```
var message2;

var age = 19;

if (age < 19) {
message2 = "not eligible to drive";

} else if (age > 19) {
message2="you are eliglibe to drive";

} else {
message2="you have just turned 19!";
}

document.getElementById("demo2")
.innerHTML = message2;
}

</script>

</body>
</html>
```

Result for the above Example :

This is the example of if Statement

Click the button to know your driving eligibility

Try it

This is the example of else Statement

Click the button to know your driving eligibility

Try it

This is the example of else if Statement

Click the button to know your driving eligibility

Try it

JAVASCRIPT SWITCH

The switch statement is used to give an expression to evaluate several different statements to execute based on the value of the expression. The interpreter checks each case against the value of the expression until a match is found. If nothing matches, a default condition will be used.

JAVASCRIPT SWITCH SYNTAX

```
switch(expression) {
case n: Code Statment
break;
case n: Code Statment
break;
default: Default Code Statment
}
```

The switch expression is evaluated once.

The value of the expression is compared with the values of each case.

If there is a match, the associated block of code is executed.

The break statements indicate the end of a particular case , this will stop the execution of more code and case testing.

The default keyword specifies the code to run if there is no case match .

Example For JavaScript Switch

```
<!DOCTYPE html>
<html>
  <body>
<h2>Example For JavaScript Switch
</h2>
```

```
<h3>Enter your grade to know
your marks</h3>


<script>

var grade='U';

document.write("GradeMarks Range :");

switch (grade)

{

case 'S': document.write
("91 to 100 Marks");
break;

case 'A': document.write
("81 to 90 Marks");
break;

case 'B': document.write
("71 to 80 Marks");
break;

case 'C': document.write
(" 61 to 70 Marks");
break;

case 'U': document.write(" Failed");
break;

default:  document.write
("Unknown Grade<br />")

}

</script>

</body>

</html>
```

Result for the above Example :

JAVASCRIPT LOOP

While writing a HTML program, you may encounter a situation where you need to perform an action over and over again. In such situations, you would need to write loop statements to reduce the number of lines.

Different Kinds of Loops

JavaScript supports different kinds of loops they are :

for - loops through a block of code a number of times.

for/in - loops through the properties of an object.

while - loops through a block of code while a specified condition is true.

do/while - also loops through a block of code while a specified condition is true.

Syntax For JavaScript loop

For Loop :

```
for (statement 1; statement 2; statement 3) {
code to be executed
}
```

While Loop :

```
while (condition)
{
```

```
code to be executed
}
```

Example for For Loop

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

<h3>  Example for For Loop </h3>

<p>Click the button to loop through
a block of code five times.</p>

<button onclick="Function()">
Try it</button>

<p id="Sample"></p>

<script>

function Function() {
var text = "";
var i;
for (i = 1; i < 6; i++) {

text += "The number is " +i+ "<br>";

}


document.getElementById("Sample")

.innerHTML = text;

}


</script>


</body>

</html>
```

Result for the above Example :

```
Example for For Loop


Click the button to loop through a block of code five times.


Try it
```

JavaScript For/In Loop

The for/in loop is used to loop through an object's properties.In each iteration, one property from object is assigned to variablename and this loop continues till all the properties of the object are exhausted.

Example for For/In Loop

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

<h3>Example for For/In Loop </h3>

<p id="sample"></p>

<script>

var txt = "";
var person = {name:"John",
gender:"Male", age:28};
var x;

for (x in person) {

   txt += person[x] + " ";
}
```

```
document.getElementById("sample")

.innerHTML = txt;


</script>


</body>

</html>
```

Result for the above Example :

While Loop

The purpose of a while loop is to execute a statement or code block repeatedly as long as an expression is true, once the expression becomes false, the loop terminates.

Example For While Loop

```
<!DOCTYPE html>

<html>

<body>


<h3> Example For While Loop </h3>
```

```
<p>Click the button to loop through
a block of code as long as i is
less than 12.</p>

<button onclick="Function()">
Try it</button>

<p id="sample"></p>

<script>

function Function() {
var text = "";
var i = 1;

while (i < 12) {
text += "<br>The number is " + i;
i++;
   }

document.getElementById("sample")
.innerHTML = text;

}
```

```
</script>

</body>
</html>
```

Result for the above Example :

Example For While Loop

Click the button to loop through a block of code as long as i is less than 12.

Try it

Do/While Loops

The do/while loop is similar to the while loop except that the condition check happens at the end of the loop. This means that the loop will always be executed at least once, even if the condition is false. Then it will repeat the loop as long as the condition is true.

Example For Do/While Loop

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

```
<h3>Example For Do/While Loop </h3>

<p>Click the button to loop through
a block of code as long as i is
less than 12.</p>

<button onclick="Function()">
Try it</button>

<p id="sample"></p>

<script>

function Function() {
var text = ""
var i = 1;

do {
text += "<br>The number is " + i;
i++;
}

while (i < 12)
document.getElementById("sample")
.innerHTML = text;
}

</script>

</body>
</html>
```

Result for the above Example :

Example For Do/While Loop

Click the button to loop through a block of code as long as i is less than 12.

Try it

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.

RESPONSIVE WEB DESIGN

Responsive web design provides an optimal experience, easy reading and easy navigation with a minimum of resizing on different devices such as desktops, mobiles and tabs, Responsive web design uses only HTML and CSS , it is not a program or a JavaScript. When you use CSS and HTML to resize, hide, shrink, enlarge, or move the content to make it look good on any screen it is called Responsive Web Design.

Web pages can be viewed using many different devices like desktops, tablets, and phones. Your web page should look good, and be easy to use, regardless of the device Web pages should not leave out information to fit smaller devices, but rather adapt its content to fit any device.

VIEWPORT

The viewport is the user's visible area of a web page, the viewport varies with the device, and will be smaller on a mobile phone than on a computer screen. Before tablets and mobile phones, web pages were designed only for computer screens, and it was common for web pages to have a static design and a fixed size.

When we started surfing the internet using tablets and mobile phones, fixed size web pages were too large to fit the viewport. To fix this, browsers on those devices scaled down the entire web page to fit the screen.

Rules For Viewport

Do Not use large fixed width elements, For example, if an image is displayed at a width wider than the viewport it can cause the viewport to scroll horizontally. Remember to adjust this content to fit within the width of the viewport.

Do NOT let the content rely on a particular viewport width to render well, since screen dimensions and width in CSS pixels vary widely between devices, content should not rely on a particular viewport width to render well.

Use CSS media queries to apply different styling for small and large screens, Setting large absolute CSS widths for page elements, will cause the element to be too wide for the viewport on a smaller device. Instead, consider using relative width values, such as width: 100%. Also, be careful of using large absolute positioning values. It may cause the element to fall outside the viewport on small devices.

Example With Viewport

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content=
"width=device-width, initial-scale=
1.0">
<style>

img {
max-width:100%;
height:auto;
}


div {
```

```
background-color: #F9FFFF;

width: 272px;

padding: 10px;

border: 5px solid steelblue;

margin: 1px;

}


</style>

</head>

<body>


<h3> Tiger </h3>





<img src="Img_tiger.jpg"

alt="Tiger" style="width:500px;

height:450px">


<p> The tiger (Panthera tigris)

is the largest cat species,

reaching a total body length

of up to 3.38 m (11.1 ft) over

curves and weighing up to 388.7

kg (857 lb) in the wild. with
```

```
  most remaining populations

  occurring in small pockets

  isolated from each other, of

  which about 2,000 exist on the

  Indian subcontinent. </p>



  </body>


  </html>
```

when viewed in different screen size the web content will be adjusted automatically according to the size.

Result for above Example :

Example Without Viewport

```
<!DOCTYPE html>

<html>

<body>
```

```
<h3> Tiger </h3>



<img src="Img_tiger.jpg"

alt="Basic Html" style="width:500px;

height:450px">



<p> The tiger (Panthera tigris)

is the largest cat species,

reaching a total body length

of up to 3.38 m (11.1 ft) over

curves and weighing up to 388.7

kg (857 lb) in the wild. with

most remaining populations

occurring in small pockets

isolated from each other, of

which about 2,000 exist on the

Indian subcontinent. </p>



</body>

</html>
```

Result for above Example :

GRIDVIEW

Many web pages are based on a grid-view, which means that the page is divided into columns, using a grid-view is very helpful when designing web pages. It makes it easier to place elements on the page.A responsive grid-view often has 12 columns, and has a total width of 100%, and will shrink and expand as you resize the browser window.

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content=
"width=device-width, initial-scale=
1.0">

<style>

* {
  box-sizing: border-box;
}

.row:after {
    content: "";

    clear: both;

    display: block;

}


[class*="col-"] {

    float: left;

    padding: 15px;


}

.col-1 {width: 8.33%;}

.col-2 {width: 17.16%;}

.col-3 {width: 25%;}

.col-4 {width: 33.33%;}

.col-5 {width: 41.66%;}

.col-6 {width: 50%;}

.col-7 {width: 58.33%;}

.col-8 {width: 66.66%;}

.col-9 {width: 75%;}

.col-10 {width: 82.33%;}

.col-11 {width: 91.55%;}

.col-12 {width: 100%;}

html {

font-family:"Lucida Sans",sans-serif;

}
```

```
.header {
    background-color: steelblue;
    color: white;
    padding: 15px;
}

.menu ul {
    list-style-type: none;
    margin: 0;
    padding: 0;

}

.menu li {

padding: 8px;
margin-bottom: 7px;
background-color :turquoise;
color: #ffffff;
box-shadow: 0 1px 3px rgba
(0,0,0,0.12), 0 1px 2px rgba
(0,0,0,0.24);

}
```

```
.menu li:hover {
  background-color: #0099cc;
}

</style>

</head>


<body>

<div class="header">
<h1>Cats</h1>
</div>



<div class="row">

<div class="col-3 menu">
<ul>
<li>Tiger</li>
<li>Cheetah</li>
<li>Puma</li>
```

```
<li>panther</li>
</ul>
</div>



<div class="col-9">

<h1>Tiger</h1>

<p>The tiger (Panthera tigris)
is the largest cat species,
reaching a total body length
of up to 3.38 m (11.1 ft) over
curves and weighing up to 388.7
kg (857 lb) in the wild. with
most remaining populations
occurring in small pockets
isolated from each other, of
which about 2,000 exist on the
Indian subcontinent.</p>

</div>

</div>
```

```
</body>

</html>
```

Result for above Example :

This is how the webpage looks in smallscreen devices

MEDIA QUERIES

Media query is a CSS technique introduced in CSS3.It uses the @media rule to include a block of CSS properties only if a certain condition is true. We can add a breakpoint where certain parts of the design will behave differently on each side of the breakpoint.

```
<!DOCTYPE html>

<html>

<head>


<meta name="viewport" content=

"width=device-width, initial-scale=

1.0">
```

```
<style>

* {
    box-sizing: border-box;
}



.row:after {
    content: "";
    clear: both;
    display: block;
}



[class*="col-"] {
    float: left;
    padding: 15px;
}



html {
font-family:"Lucida Sans",sans-serif;
}
```

```
}


.header {
    background-color: #9933cc;
    color: #ffffff;
    padding: 15px;
}


.menu ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
}


.menu li {

padding: 8px;
margin-bottom: 7px;
background-color :#33b5e5;
color: #ffffff;
box-shadow: 0 1px 3px rgba
(0,0,0,0.12), 0 1px 2px
rgba(0,0,0,0.24);

}
```

```
.menu li:hover {
 background-color: #0099cc;
}


.aside {
   background-color: #33b5e5;
padding: 15px;
color: #ffffff;
text-align: center;
font-size: 14px;
box-shadow: 0 1px 3px rgba
(0,0,0,0.12), 0 1px 2px
rgba(0,0,0,0.24);
}


.footer {
   background-color: #0099cc;
   color: #ffffff;
   text-align: center;
   font-size: 12px;
   padding: 15px;
}
```

```
/* For mobile phones: */

[class*="col-"] {

    width: 100%;

}



@media only screen and

(min-width:600px) {

    /* For tablets: */

    .col-m-1 {width: 8.33%;}

    .col-m-2 {width: 16.66%;}

    .col-m-3 {width: 25%;}

    .col-m-4 {width: 33.33%;}

    .col-m-5 {width: 41.66%;}

    .col-m-6 {width: 50%;}

    .col-m-7 {width: 58.33%;}

    .col-m-8 {width: 66.66%;}

    .col-m-9 {width: 75%;}

    .col-m-10 {width: 83.33%;}

    .col-m-11 {width: 91.66%;}

    .col-m-12 {width: 100%;}

}
```

```
@media only screen and
(min-width: 768px) {
  /* For desktop: */
  .col-1 {width: 8.33%;}
  .col-2 {width: 16.66%;}
  .col-3 {width: 25%;}
  .col-4 {width: 33.33%;}
  .col-5 {width: 41.66%;}
  .col-6 {width: 50%;}
  .col-7 {width: 58.33%;}
  .col-8 {width: 66.66%;}
  .col-9 {width: 75%;}
  .col-10 {width: 83.33%;}
  .col-11 {width: 91.66%;}
  .col-12 {width: 100%;}
}


</style>
</head>
<body>

<div class="header">
<h1>Cats</h1>
```

```
</div>

<div class="row">
<div class="col-3 col-m-3 menu">
<ul>
<li>Tiger</li>
<li>Cheetah</li>
<li>Panther</li>
<li>Lion</li>
</ul>
</div>

<div class="col-6 col-m-9">
<h1>Tiger</h1>
<p>The tiger (Panthera tigris)
is the largest cat species,
reaching a total body length
of up to 3.38 m (11.1 ft) over
curves and weighing up to 388.7
kg (857 lb) in the wild. with
most remaining populations
occurring in small pockets
isolated from each other, of
which about 2,000 exist on the
Indian subcontinent.</p>
```

```
</div>

<div class="col-3 col-m-12">
<div class="aside">
<h2>Tiger</h2>
<p>Tiger is the largest cat</p>
<h2>Cheetah</h2>
<p>Cheetah is the fastest cat</p>
<h2>Lion</h2>
<p>Lion is the King </p>
</div>
</div>

</div>

<div class="footer">
<p>Resize the browser window to see
how the content respond to the
resizing.</p>
</div>

</body>
</html>
```

Result for above Example :

This is how the webpage looks in smallscreen devices

RESPONSIVE VIDEOS

In Responsive Webpage the video in the webpage will be Scaled and adjusted according to Size of the Screen

Using The Width Property

If the width property is set to 100%, the video player will be responsive and scale up and down, the video player can be scaled up to be larger than its original size.

```
<!DOCTYPE html>

<html>

<head>


<meta name="viewport" content=

"width=device-width, initial-scale=

1.0">
```

```
<style>

video {
   width: 100%;
   height: auto;
}

</style>

</head>

<body>

<video width="400" controls>

<source src="mo.mp4"
type="video/mp4">
<source src="mo.ogg"
type="video/ogg">
Your browser does not support HTML5
video.

</video>
```

```
<p>Resize the browser window to see

how the size of the video player

will scale.</p>




</body>

</html>
```

Result for above Example :

This is how the webpage looks in smallscreen devices

Using The Max-Width Property

If the max-width property is set to 100%, the video player will scale down if it has to, but never scale up to be larger than its original size.

```
<!DOCTYPE html>

<html>

<head>
```

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


<style>

video {
  max-width: 100%;
  height: auto;
}

</style>
</head>
<body>

<video width="400" controls>
<source src="mo.mp4"
type="video/mp4">
<source src="mo.ogg"
type="video/ogg">
Your browser does not support
HTML5 video.
</video>
```

```
<h3>Max-Width Property</h3>


<p>The video player will scale down
if it has to,but never scale up to
be larger than its original size</p>

</body>
</html>
```

Result for above Example :

This is how the webpage looks in smallscreen devices

FRAMEWORKS

There are many existing CSS Frameworks that offer Responsive Design.Many are free and easy to use.

Bootstarp

Bootstrap is most popular web design framework based on HTML,CSS and Java script and it helps you to design web pages in responsive way for all devices

```
<html>
<head>


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



<link rel="stylesheet" href="http:
//maxcdn.bootstrapcdn.com/bootstrap
/3.2.0/css/bootstrap.min.css">

<style>

body{
color:slateblue;
}

.jumbotron{
```

```
background-color:turquoise;

}




</style>


</head>

<body>


<div class="container">


<div class="jumbotron">


<h1>Cats</h1>


</div>




<div class="row">

<div class="col-md-4">


<h2>Tiger</h2>

<p> The tiger (Panthera tigris)

is the largest cat species,
reaching a total body length
of up to 3.38 m (11.1 ft) over
curves and weighing up to 388.7
kg (857 lb) in the wild. with
most remaining populations
occurring in small pockets
isolated from each other, of
which about 2,000 exist on the
Indian subcontinent. </p>
</div>




<div class="col-md-4">


<h2>Lion</h2>
<p>The lion (Panthera leo) is one
of the five big cats in the genus
Panthera and a member of the family
Felidae , it is the second-largest
living cat after the tiger . lions
live for 10ï¿½14 years in the wild,
although in captivity they can live
more than 20 years.</p>
```

```
</div>



<div class="col-md-4">

<h2>Cheetah</h2>
<p>The cheetah (Acinonyx jubatus) is
a big cat in the subfamily Felinae
that inhabits most of Africa and
parts of Iran. It is the only extant
member of the genus Acinonyx. The
cheetah can run as fast as 109.4 to
120.7 km/h (68.0 to 75.0 mph),faster
than any other land animal. </p>

</div>



</body>
</html>
```

Result for above Example :

This is how the webpage looks in smallscreen devices

Skeleton

Another popular framework is Skeleton, it uses only CSS to make responsive web pages.

```
<!DOCTYPE html>
<html>
<head>

<title>Skeleton Example</title>
<meta charset="utf-8">

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


<link rel="stylesheet"
```

```
href="skeleton.css">

<link rel="stylesheet"
href="normalize.css">

</head>
<body>

<div class="container">

<h1>THE CATS</h1>


<div class="row">

<div class="one-half column">Tiger
<p>The tiger (Panthera tigris)
is the largest cat species,
reaching a total body length
of up to 3.38 m (11.1 ft) over
curves and weighing up to 388.7
kg (857 lb) in the wild. with
most remaining populations
occurring in small pockets
isolated from each other, of
```

```
which about 2,000 exist on the
Indian subcontinent. </p>

</div>


<div class="one-half column">Lion
<p>The lion (Panthera leo) is one
of the five big cats in the genus
Panthera and a member of the family
Felidae , it is the second-largest
living cat after the tiger . lions
live for 10ï¿½14 years in the wild,
although in captivity they can live
more than 20 years.</p>
</div>

</div>

</div>

</body>
</html>
```

Result for above Example :

This is how the webpage looks in smallscreen devices

TRY YOURSELF :

Type the code or copy/paste below and press show result to see the result.