

IWT module-2 part-2(BKPanda)

Intro to CSS style

CSS can be added to HTML documents in 3 ways:

- **Inline** - by using the **style** attribute inside HTML elements
- **Internal** - by using a **<style>** element in the **<head>** section
- **External** - by using a **<link>** element to link to an external CSS file

Inline CSS

- An inline CSS is used to apply a unique style to a single HTML element.

Syntax

<tagname style="property:value;">

e.g **<h1 style="color:blue;">**A Blue Heading**</h1>**

<p style="background-color:tomato;">This is a paragraph.**</p>**

<h1 style="font-family:verdana;">This is a heading**</h1>**

<h1 style="font-size:300%;">This is a heading**</h1>**

<h1 style="text-align:center;">Centered Heading**</h1>**

- Use **background-color** for background color
- Use **color** for text colors
- Use **font-family** for text fonts
- Use **font-size** for text sizes
- Use **text-align** for text alignment
- **border** property defines a border around an HTML element
- **margin** property defines a margin (space) outside the border.
- **padding** property defines a padding (space) between the text and the border

ex

```
<!doctype>
<html>
<head>
  <title>Internal style demo</title>
</head>
<body>
  <h1 style="color:blue;font-family:verdana;font-size:200%;border:2px solid powderblue;
padding: 30px; margin: 50px; background-color:rgb(25,25,71);">This is a heading</h1>
  <p>This is a paragraph.</p>
</body>
</html>
```

HTML colors are specified with predefined color names, or with RGB, HEX, HSL, RGBA, or HSLA values.

Common Color names: Tomato, Orange, DodgerBlue, MediumSeaGreen, Gray, SlateBlue, Violet, LightGray

Internal CSS

- An internal CSS is used to define a style for a single HTML page.
- An internal CSS is defined in the **<head>** section of an HTML page, within a **<style>** element.

e.g

```
<!DOCTYPE html>
<html>
<head>
<style>
  h1 {
    color: blue;
    font-family: verdana;
    font-size: 300%;
    border: 2px solid powderblue;
    padding: 30px;
    margin: 50px;
    background-color:rgb(25, 25, 71);
  }
</style>
```

```

h2 {
  color: blue;
  font-family: verdana;
  font-size: 300%;
  border: 2px solid powderblue;
  padding: 30px;
  margin: 50px;
  background-color: rgba(255, 99, 71, 0.2);; note a in rgba represents transparency
}
p {
  color: white;
  font-family: courier;
  font-size: 160%;
  border: 5px dashed red;
  padding: 30px;
  margin: 50px;
  background-color: DodgerBlue;
}
</style>
</head>
<body>

<h1>This is a heading</h1>
<h2>This is a heading2</h2>
<p>This is a paragraph.</p>

</body>
</html>

```

note about colors: rgb(red, green, blue). Each parameter (red, green, and blue) defines the intensity of the color with a value between 0 and 255. This means that there are $256 \times 256 \times 256 = 16777216$ possible colors! To display black, set all color parameters to 0, like this: `rgb(0, 0, 0)`. To display white, set all color parameters to 255, like this: `rgb(255, 255, 255)`

rgba(red, green, blue, alpha). The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (not transparent at all):

In HTML, a color can be specified using a hexadecimal value in the form: **#rrggbb** : Where rr (red), gg (green) and bb (blue) are hexadecimal values between 00 and ff (same as decimal 0-255). example, `#ff0000` is displayed as red, because red is set to its highest value (ff), and the other two (green and blue) are set to 00. To display black, set all color parameters to 00, like this: `#000000`. To display white, set all color parameters to ff, like this: `#ffffff`

In HTML, a color can be specified using hue, saturation, and lightness (HSL) in the form:

hsl(hue, saturation, lightness). Hue is a degree on the color wheel from 0 to 360. 0 is red, 120 is green, and 240 is blue. Saturation is a percentage value. 0% means a shade of gray, and 100% is the full color.

Lightness is also a percentage value. 0% is black, and 100% is white. E.g `hsl(240, 100%, 50%)`

Block and inline elements

Every HTML element has a default display value, depending on what type of element it is. There are two display values: **block and inline.**

Block-level Elements

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

Two commonly used block elements are: <p> and <div>.

The <p> element defines a paragraph in an HTML document.

The <div> element defines a division or a section in an HTML document.

Here are the block-level elements in HTML:

<u><address></u>	<u><figcaption></u>	<u><noscript></u>
<u><article></u>	<u><figure></u>	<u></u>
<u><aside></u>	<u><footer></u>	<u><p></u>
<u><blockquote></u>	<u><form></u>	<u><pre></u>
<u><canvas></u>	<u><h1>-<h6></u>	<u><section></u>
<u><dd></u>	<u><header></u>	<u><table></u>
<u><div></u>	<u><hr></u>	<u><tfoot></u>
<u><dl></u>	<u></u>	<u></u>
<u><dt></u>	<u><main></u>	<u><video></u>
<u><fieldset></u>	<u><nav></u>	

Inline Elements

An inline element does not start on a new line. An inline element only takes up as much width as necessary. An inline element cannot contain a block-level element!

Inline elements:

<u><a></u>	<u></u>	<u><small></u>
<u><abbr></u>	<u><input></u>	<u></u>
<u><acronym></u>	<u><kbd></u>	<u></u>
<u></u>	<u><label></u>	<u><sub></u>
<u><bdo><big>
</u>	<u><map></u>	<u><sup></u>
<u><button></u>	<u><object></u>	<u><textarea></u>
<u><cite></u>	<u><output></u>	<u><time></u>
<u><code></u>	<u><q></u>	<u><tt></u>
<u><dfn></u>	<u><samp></u>	<u><var></u>
<u></u>	<u><script></u>	
<u><i></u>	<u><select></u>	

The <div> Element: The <div> element is often used as a container for other HTML elements. The <div> element has no required attributes, but style, class and id are common. When used together with CSS, the <div> element can be used to style blocks of content:

```
<!DOCTYPE html>
<html>
<body>
<div style="background-color:black;color:white;padding:20px;">
  <h2>London</h2>
  <p>London is the capital city of England. It is the most populous city in the United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>
  <p>Standing on the River Thames, London has been a major settlement for two millennia, its history going back to its founding by the Romans, who named it Londinium.</p>
</div>
</body>
</html>
```

London

London is the capital city of England. It is the most populous city in the United Kingdom, with a metropolitan area of over 13 million inhabitants.

Standing on the River Thames, London has been a major settlement for two millennia, its history going back to its founding by the Romans, who named it Londinium.

The `` Element: The `` element is an inline container used to mark up a part of a text, or a part of a document. The `` element has no required attributes, but `style`, `class` and `id` are common.

When used together with CSS, the `` element can be used to style parts of the text:

```
<!DOCTYPE html>
<html>
<body>
<p>This is an inline span <span style="border: 1px solid black">Hello World</span> element inside a
paragraph.</p>
<p>The SPAN element is an inline element, and will not start on a new line and only takes up as much width
as necessary.</p>
</body>
</html>
```

o/p

This is an inline span Hello World element inside a paragraph.

The SPAN element is an inline element, and will not start on a new line and only takes up as much width as necessary.

Lists

Types:

- i. Unordered
- ii. Ordered
- iii. Definition
- iv. nested

Unordered Lists(bulleted)

e.g

- abc
- def
- geh
- ijk

The list items are marked with bullets. An unordered list starts with the tag. Each list item starts with the tag.

```
<html>
<body>
<h4>An Unordered List:</h4>
<ul>
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
</ul>
</body>
</html>
```

Inside a list item, you can put paragraphs, line breaks, images, links, other lists, and so on. You can display different kinds of bullets in an unordered list by using the **type attribute**.

```
<html>
<body>
<h4>Disc bullets list:</h4>
<ul type="disc">
    <li>Apples</li>
    <li>Bananas</li>
    <li>Lemons</li>
</ul>
<h4>Circle bullets list:</h4>
<ul type="circle">
    <li>Apples</li>
    <li>Bananas</li>
    <li>Lemons</li>
</ul>
<h4>Square bullets list:</h4>
<ul type="square">
    <li>Apples</li>
    <li>Bananas</li>
    <li>Lemons</li>
</ul>
</body>
</html>
```

Ordered Lists

1. abc
2. def
3. geh
4. ijk

An ordered list is also a list of items; the list items are numbered sequentially rather than bulleted. An ordered list starts with the tag. Each list item starts with the tag.

```

<html>
<body>
<h4>An Ordered List:</h4>
<ol>
    <li>Coffee</li>
    <li>Tea</li>
    <li>Milk</li>
</ol>
</body>
</html>

```

You can display different kinds of ordered lists by using the **type attribute**

```

<html>
<body>
<h4>Letters list:</h4>
<ol type="A">
    <li>Apples</li>
    <li>Bananas</li>
    <li>Lemons</li>
</ol>
<h4>Lowercase letters list:</h4>
<ol type="a">
    <li>Apples</li>
    <li>Bananas</li>
    <li>Lemons</li>
</ol>
<h4>Roman numbers list:</h4>
<ol type="I">
    <li>Apples</li>
    <li>Bananas</li>
    <li>Lemons</li>
</ol>
<h4>Lowercase Roman numbers list:</h4>
<ol type="i">
    <li>Apples</li>
    <li>Bananas</li>
    <li>Lemons</li>
</ol>
</body>
</html>

```

Definition Lists

It is a list of items (terms), together with a description of each item (term).

A definition list starts with a <dl> tag (definition list). Each term starts with a <dt> tag (definition term).

Each description starts with a <dd> tag (definition description).

```

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

```

Output

A Definition List:

Coffee
Black hot drink
Milk
White cold drink

Inside the <dd> tag you can put paragraphs, line breaks, images, links, other lists, and so on.

Nested Lists(List within a list)

A nested list is a list within another list. Usually the second list is indented another level and the item markers will appear differently than the original list,

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

Output

A nested List:

- Coffee
- Tea
 - Black tea
 - Green tea
- Milk

Nested lists can be several levels deep,

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

A nested List:

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

Summary

TAG	DESCRIPTION
	Defines an ordered list
	Defines an unordered list
	Defines a list item
<dl>	Defines a definition list
<dt>	Defines a term (an item) in a definition list
<dd>	Defines a description of a term in a definition list
<dir>	Deprecated. Use instead
<menu>	Deprecated. Use instead

Links

A link is the “address” to a document (or a resource) located on the World Wide Web or elsewhere within your own Web server. **i.e hyperlink** is a reference (an address) to a resource on the Web. Hyperlinks can point to any resource on the Web: an HTML page, an image, a sound file, a movie, and so on.

An HTML **anchor** is a term used to define a hyperlink destination inside a document.

The anchor element `<a>` defines both hyperlinks and anchors.

Syntax

The start tag contains attributes about the link.

`Link text`

The element content (Link text) defines the part to be displayed. The element content doesn't have to be text. You can link from an image or any other HTML element.

href Attribute

The href attribute defines the link “address”.

Example.

` `

``

``

Link can be

- To external document
E.g ``
- To a html document kept in a specific folder
e.g ` `
- To some other part of same document
``

Note: The target attribute is used to control how the browser responds when you click on the link. The options for target are:

OPTION	DESCRIPTION
<code>_blank</code>	Opens the linked document in a new window
<code>_self</code>	Opens the linked document in the same frame as it was clicked (this is default)
<code>_parent</code>	Opens the linked document in the parent frameset
<code>_top</code>	Opens the linked document in the full body of the window
<code>framename</code>	Opens the linked document in a named frame

Name Attribute(to link internal navigation in the same page or...)

syntax

``

e.g ``

href link to this portion may look like this

``

Named anchors are not displayed in any special way by the browser because they are invisible to the reader. Named anchors are sometimes used to create a table of contents at the beginning of a large document.

If a browser cannot find a named anchor that has been specified, it goes to the top of the document. No error occurs.

Note: Always add a trailing slash to subfolder references. If you link like this:

`href="http://www.abc.com/html"`, you will generate two HTTP requests to the server because the server will add a slash to the address and create a new request like this: `href="http://www.abc.com/html/"`.

Example

```
<html>
<body>
<h1>Link demo</h1><br>
<a name="index"><h2>Index</h2></a><br>
<a href="http://www.gita.edu.in/">My College</a><br>
<a href="myprofile.html">My Profile</a><br>
<a href="#C1"> Chapter 1.</a><br>
<a href="#C2"> Chapter 2.</a><br>
<a href="#C3"> Chapter 3.</a><br>
```

```

<a name = "C1"><h2>Chapter 1</h2></a><br>
<p>This chapter explains ba bla bla
    ba bla bla
    ba bla bla
    ba bla bla
    ba bla bla
</p>
    <a href="#index"> Go to Index page</a>

<a name = "C2"><h2>Chapter 2</h2></a><br>
<p>This chapter explains ba bla bla
    ba bla bla
    ba bla bla
    ba bla bla
    ba bla bla
</p>
    <a href="#index"> Go to Index page</a>
<a name = "C3"><h2>Chapter 3</h2></a><br>
<p>This chapter explains ba bla bla
    ba bla bla
    ba bla bla
    ba bla bla
    ba bla bla
</p>
    <a href="#index"> Go to Index page</a>
</body>
</html>

```

Note: for navigating within the same page you can also use id attribute

E,g

```

<h1 id = "features">My 3 Favorite Bugs</h1>  destination (internal)
<a href = "#features">Go to <em>Favorite Features</em></a> for link to go to destination

```

A hyperlink can also reference an internal link in *another* document by specifying the document name followed by a pound sign and the id value, as in: href = "*filename.html#id*"

For example, to link to a tag with the id attribute booklist in books.html, href is assigned "books.html#booklist". You can send the browser to an internal link on another website by appending the pound sign and id value of an element to any URL, as in: href = "*URL/filename.html#id*"

mailto: Link

The following example demonstrates how to link to an e-mail address and generate a new e-mail message in your default e-mail application (this works only if you have mail installed).

```

<html>
<body>
<p>
This is a mail link:
<a href="mailto:someone@microsoft.com?subject=Hello%20again">Send Mail</a>
</p>
<p>
<b>Note:</b> Spaces between words should be replaced by %20 to <b>ensure</b>
that the browser will display your text properly.
</p>
</body>
</html>

```

More example with cc bcc etc.

```
<html>
<body>
<p>
This is another mailto link:
<a
href="mailto:someone@microsoft.com?cc=someoneelse@microsoft.com&bcc=andsomeoneel
se2@microsoft.com&subject=Summer%20Party&body=You%20are%20invited%20
to%20a%20big%20summer%20party!">Send mail!</a>
</p>
<p>
<b>Note:</b> Spaces between words should be replaced by %20
to <b>ensure</b> that the browser will display your text
properly.
</p>
</body>
</html>
```

An Image Link: you can use image as a link also. E.g

```
<html>
<body>
<p>Create a link attached to an image:
<a href="default.htm"> </a></p>

<p>No border around the image, but still a link:
<a href="default.htm"> </a></p>

</body>
</html>
```

A file path describes the location of a file in a web site's folder structure.

File Path

A file path describes the location of a file in a web site's folder structure.

File paths are used when linking to external files, like:

- Web pages
- Images
- Style sheets
- JavaScripts

Path	Description
	The "picture.jpg" file is located in the same folder as the current page
	The "picture.jpg" file is located in the images folder in the current folder
	The "picture.jpg" file is located in the images folder at the root of the current web
	The "picture.jpg" file is located in the folder one level up from the current folder

HTML Images

img Tag and the src Attribute

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

The syntax of defining an image:

The URL points to the location or address where the image is stored. An image file named "boat.gif" located in the directory "images" on "www.anc.com" has the URL: http://www.abc.com/images/boat.gif

The browser puts the image where the image tag occurs in the document. If you put an image tag between two paragraphs, the browser shows the first paragraph, then the image, and then the second paragraph

```
<html>
<body>
<p>
An image:

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

Ex-2

```
<html>
<body>
<p>An image from another folder:</p>


<p>An image from ABC.com:</p>

</body>
</html>
```

Background Images

```
<html>
  <body background="background.jpg">
    <h3>Look: A background image!</h3>
    <p>Both gif and jpg files can be used as HTML backgrounds.</p>
    <p>If the image is smaller than the page, the image will repeat
    itself.</p>
  </body>
</html>
```

you can align images within the text.

```
<html>
<body>
<p>The text is aligned with the image at the bottom.</p>
<p>The text is aligned with the imagein the middle.</p>
<p>The text is aligned with the image at the top.</p>
<p><b>Note:</b> The bottom alignment is the default!</p>
</html>
</body>
```

o/p:

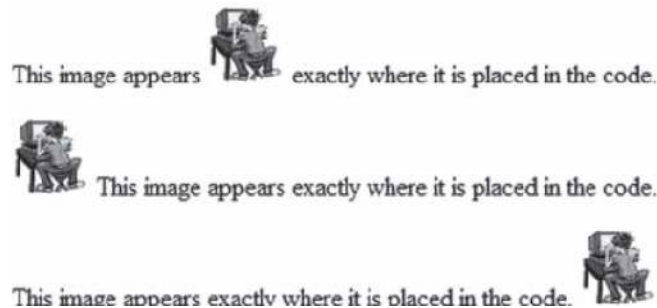


Note: The bottom alignment is the default!

Ex 2

```
<html>
<body>
<p>This image appears

exactly where it is placed in the code.</p>
<p>
This image appears exactly where it is placed in the code.</
p>
<p>This image appears exactly where it is placed in the
code.
</p>
</body>
</html>
```

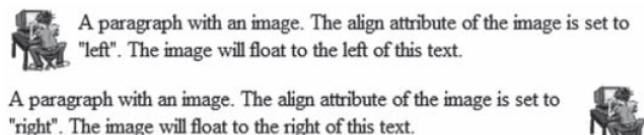


Floating Images

```
<html>
<body>
<p>

A paragraph with an image. The align attribute of the image is set to "left". The image will
float to the left of this text.
</p>
<p>

A paragraph with an image. The align attribute of the image is set to "right". The image will
float to the right of this text.
</p>
</body>
</html>
```



The width and height attributes allow the page to render properly and more efficiently before the image is downloaded. Without them, the page will render once, then re-render when each image is loaded. The image will be scaled to fit the stated height and width

```
<html>
<body>
<p>

</p>
<p>

</p>
<p>

</p>
<p>You can make an image smaller or larger by changing the
values of the height and width attributes.</p>
</body>
</html>
```



alt Attribute

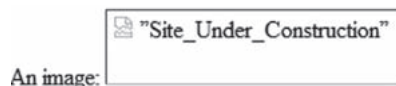
The alt attribute is used to define an alternate text for an image. The alt attribute tells the reader what he or she is missing on a page if the browser can't load images. The browser will then display the alternate text instead of the image.

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

It is a good practice to include alternate text for every image on a page to improve the display and usefulness of your document for people who have text-only browsers.

```
<html>
<body>
<p>
An image:

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



Creating an Image Map

The following example demonstrates how to create an image map with clickable regions. Each of the regions is a hyperlink

```
<html> <body>
<p>Click on the sun or on one of the planets to watch it closer:</p>

<map name="planetmap">
<area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm" />
<area shape="circle" coords="90,58,3" alt="Mercury" href="mercur.htm" />
<area shape="circle" coords="124,58,8" alt="Venus" href="venus.htm" />
</map>
</body> </html>
```

HTML Tables

Tables are an excellent way to organize and display information on a page. Tables are defined using the `<table>...</table>` tag.

A table is divided into rows with the `<tr> ..</tr>` tag, and each row is divided into data cells or columns using the `<td>` tag. `td` stand for “table data,” which is the content of a data cell. A data cell can contain text, images, lists, paragraphs, forms, horizontal rules, tables, and so on. Table rows can be header rows or data rows. Header rows are in tags `<th>...</th>`

A basic table includes the following tags:

<table> attributes are: `align`→left/right/center, `valign`→top/bottom/middle, `width`→no of pixel/percentage, `border`→border property of table, `cellpadding`→spacing within a cell, `cellspacing`, `colspan`→how many cols the cell will occupy, `rowspan`→ how many row the cell will occupy, `bgcolor`→background color of cell/row/table etc, `background`→background image of cell or/table etc.

Additional tags : caption with attribute `align=bottom/top` to show the place of table caption

- ▶▶ starts with a `table` tag.
- ▶▶ row starts with a `tr` tag.
- ▶▶ table data (cell) starts with a `td` tag or `<th>` for heading.

Lets we have to display the following 2 tables

COL1	COL2	COL3
111	2222	333
AAAA	22	22
	44	55
BBBB	def	

Vertical headers:

First Name:	Bill Gates
Telephone:	555 777 1854
Telephone:	555 777 1855

```
<!DOCTYPE html>
<html>
  <head>
    <title>Table</title>
  </head>
  <body>
    <table align=center border=5 width=50% bgcolor=gray>
      <caption align=top> MY TABLE</caption>
      <tr>
        <th>COL1</th>
        <th>COL2</th>
        <th>COL3</th>
      </tr>
      <tr>
        <td>111</td>
        <td>2222</td>
        <td>333</td>
      </tr>
      <tr>
        <td rowspan =2>AAAA</td>
        <td>22</td>
        <td>22</td>
      </tr>
      <tr>
        <td>44</td>
        <td>55</td>
      </tr>
```

```

        <tr>
            <td>BBBB</td>
            <td colspan =2 align=center>def</td>
        </tr>

    </table>

    <h4>Vertical headers:</h4>
    <table border="1">
        <tr>
            <th>First Name:</th>
            <td>Bill Gates</td>
        </tr>
        <tr>
            <th>Telephone:</th>
            <td>555 777 1854</td>
        </tr>
        <tr>
            <th>Telephone:</th>
            <td>555 777 1855</td>
        </tr>
    </table>

</body>
</html>

```

The border attribute controls the appearance of the table's borders or lines. The default border is 0, so if you do not specify a border attribute, the table is displayed without any borders

```

<html>
<body>
<h4>With a normal border:</h4>
<table border="1">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>With a thick border:</h4>
<table border="8">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>With a very thick border:</h4>
<table border="15">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
</body>
</html>

```


With a normal border:

First	Row
Second	Row

With a thick border:

First	Row
Second	Row

With a very thick border:

First	Row
Second	Row

Table cells with no content do not display very well in most browsers. To avoid this, add a nonbreaking space () to empty data cells to ensure the borders are visible

```
<table border="1">
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td>&nbsp;</td>
</tr>
</table>
```

How to display elements inside other elements

This is a paragraph This is another paragraph	This cell contains a table: <table><tr><td>A</td><td>B</td></tr><tr><td>C</td><td>D</td></tr></table>	A	B	C	D
A	B				
C	D				
This cell contains a list <ul style="list-style-type: none">• apples• bananas• pineapples	HELLO				

```
<html> <body>
<table border="1">
<tr>
<td>
<p>This is a paragraph</p>
<p>This is another paragraph</p>
</td>
<td>This cell contains a table:
<table border="1">
<tr>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>C</td>
<td>D</td>
</tr>
</table>
</td>
</tr>
<tr>
<td>This cell contains a list
<ul>
<li>apples</li>
<li>bananas</li>
<li>pineapples</li>
</ul>
</td>
<td>HELLO</td>
</tr>
</table>
</body> </html>
```

create more white space between the cell content and its borders(cell padding)

```
<html>
<body>
<h4>Without cellpadding:</h4>
<table border="1">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>With cellpadding:</h4>
<table border="1" cellpadding="10">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
</body>
</html>
```

Without cellpadding:

First	Row
Second	Row

With cellpadding:

First	Row
Second	Row

Increase the distance between the cells(cell spacing)

```
<html>
<body>
<h4>Without cellspacing:</h4>
<table border="1">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>With cellspacing:</h4>
<table border="1" cellspacing="10">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
</body>
</html>
```

Without cellpadding:

First	Row
Second	Row

With cellpadding:

First	Row
Second	Row

Add a background to a table(bgcolor,background)

```
<html>
<body>
<h4>A background color:</h4>
<table border="1" bgcolor="gray">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>A background image:</h4>
<table border="1" background="bgdesert.jpg">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
</body>
</html>
```

Add background to one or more table cells.

```
<html>
<body>
<h4>Cell backgrounds:</h4>
<table border="1">
<tr>
<td bgcolor="gray">First</td>
<td>Row</td>
</tr>
<tr>
<td background="bgdesert.jpg">
Second</td>
<td>Row</td>
</tr>
</table>
</body>
</html>
```

Use the frame attribute to control the borders around the table.

```
<html>
<body>
<h4>With frame="border":</h4>
<table frame="border">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>With frame="box":</h4>
<table frame="box">
<tr>
```

With frame="border":

With frame="box":

With frame="void":

```
<html>
<body>
<h4>With frame="above":</h4>
<table frame="above">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>With frame="below":</h4>
<table frame="below"><td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>With frame="hsides":</h4>
<table frame="hsides">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
</body>
</html>
```

With frame="above":

First	Row
Second Row	

With frame="below":

First	Row
Second Row	

With frame="hsides":

First	Row
Second Row	

```
<html>
<body>
<h4>With frame="vsides":</h4>
<table frame="vsides">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>With frame="lhs":</h4>
<table frame="lhs">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
<h4>With frame="rhs":</h4>
<table frame="rhs">
<tr>
<td>First</td>
<td>Row</td>
</tr>
<tr>
<td>Second</td>
<td>Row</td>
</tr>
</table>
</body>
</html>
```

With frame="vsides":

First	Row
Second Row	

With frame="lhs":

First	Row
Second Row	

With frame="rhs":

First	Row
Second Row	

Use the **frame** and **border** attributes to control the borders around the table.

```
<html>
<body>
<table frame="hsides" border="3">
<tr>
<td>First row</td>
</tr>
</table>
<br />
<table frame="vsides" border="3">
<tr>
<td>First row</td>
</tr>
</table>
</body>
</html>
```

First row

First row

TAG	DESCRIPTION
<table>	Defines a table
<th>	Defines a table header
<tr>	Defines a table row
<td>	Defines a table cell
<caption>	Defines a table caption
<colgroup>	Defines groups of table columns
<col>	Defines the attribute values for one or more columns in a table
<thead>	Defines a table head
<tbody>	Defines a table body
<tfoot>	Defines a table footer