Introduction to HTML:

HTML means Hypertext Markup Language. HTML is a markup language which is used to create static web pages. It was created by Berners-Lee by late 1991. If you are thinking of creating your own web pages, you need to know at least basic HTML.

In HTML ML(Markup Language) is used to apply layout and formatting conventions to a text document, and HT(Hypertext) means the way of linking web pages together. The latest version is HTML5, which was released in 2014

History of HTML

HTML was originally developed by Tim Berners-Lee while at CERN, and popularized by the Mosaic browser developed at NCSA. HTML standards are created by group of interested organizations and individuals called **W3C (World Wide Web Consortium).** There have now been three official HTML standards:

1. HTML 2.0 was released in 1994 and remains the baseline for backwards compatibility and should be supported by all browsers.
2. HTML 3.2 was released in 1996 with many useful additions
3. HTML 4.0 was released in 1997 and slightly amended in 1999
4. Now, The current version is HTML 5.0 was released in 2014

Build Blocks of HTML

HTML documents can be designed with the help of three statements, those are

1. tags
2. attributes
3. elements

1. Tags:

A tag is a format name surrounded by angular brackets which tells when to switch on a piece of formatting and when to switch it off.

- A tag is made up of left operator (<), a right operator (>) and a tag name between these two operators.
- If you forget to mention the right operator (>) or if you give any space between left operator and tag name browser will not consider it as tag.
- At the same time if browser not understands the tag name it just ignores it, browser won‟t generate any errors.
- HTML language is not case sensitive; hence user can write the code in either upper case or lower case. No difference between <HTML>, <html> and <hTmL>
- Syntax:

< tagname [parameter=value]>

- Ex:

  HR is a tag name that displays a horizontal ruler line.

- ---------- **Different types of Tags:**
  - Container/Paired tags
  - Empty/Singleton tags
- Singleton tag does not require an ending tag but they end with slash. (Ex:<HR/>)
- Container tag required an ending tag, which is similar to opening tag except backslash before the tag name (Ex: <HTML> is opening tag, then ending tag is </HTML>

2. Attributes:

It specify additional information about tag. It is always defined in starting tag

It is made up two parts :

a. name -- it is the property you want to set

b. value -- it to be set and always put in the quotations

Attribute names and values are case sensitive, and should only be written in lower case letters.

You can add multiple attributes in one HTML element, but need to give space between two attributes.

Syntax:

<**tag** attribute_name="value">content**</tag>**

Example:

** hai **

Attribute Name **Attribute Value**

3. Elements:

- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- Most HTML elements are written with a start tag and an end tag with content in between.
- Syntax:

  <tagname>Content goes here...</tagname>

- Ex: <h1>My First Heading</h1>

Element

Structure of an HTML document:

All HTML documents follow the same basic structure. They have the root tag as <html>, which contains <*head*> tag and <*body*> tag. The <head> tag is used to place control information used by the browser and server. The <body> tag contains the actual user information that is to be displayed on the computer screen. The basic document is shown below:

```
<html>
    <head>
        <title> home page </title>
    </head>
    <body>
        Content
    </body>
</html>
```

Besides <head> and <body> tag, there are some other tags like title, which is a sub tag of head, that displays the information in the title bar of the browser. <h1> is used to display the line in its own format i.e., bold with some big font size. <p> is used to write the content in the form of paragraph.

Comments in HTML

An HTML comment begins with "<!--" and ends with "-->". There should not be a space between angular bracket and exclamation mark. Each comment can contain as many lines of text as you like. If comment is having more lines, then each line must start and end with -- and must not contain -- within its body.

<! -- this is a single line comment line - ->

<! -- this is a multiline comment --

-- spawned over --

-- three line -->

The Document Head

The only tag that most authors insert in their head sections is the title:

<title>. . . </title>

All HTML documents have just one title which is displayed at the top of the browser window. The title is used as the name in the bookmark fields and on search engines

The Document Body

In HTML, Documents are structured as blocks of text, each of which can be formatted independently. The two major block of text in HTML documents are the paragraph and heading.

In HTML paragraphs are created by <p> tag and headings are created by <hn> tag where n starts from 1 to 6

Creating HTML Page

The Following steps are needed to create a HTML page

Step 1: Open any text editor like Notepad, gedit, Wordpad etc.

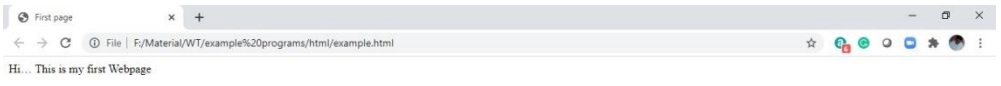
Step 2: Use the File menu to create a new document (File New) and type the HTML code

```
<HTML>
  <HEAD>
    <TITLE>First page </TITLE>
  <HEAD>
  <BODY>
    Hi… This is my first Webpage
  </BODY>
  </HTML>
```

Step 3: Go to the file menu and choose *"Save as"* option (File->Save as) and give the name of the file as "example.html" under root directory(C:) or any location for easy to identify the file

Step 4: After saving, an Default browser icon will be displayed at specified path or location whare it is placed in our system

Step 5: Double click to execute it. The output displayed as follows



Basic HTML tags

1. Character-formatting tags:

You can use character formatting tags to format a text block that is as small as a single character or as large as an entire document. Some of the most frequently used Character-formatting tags are :

1. Boldface tag

This tag is used for implement bold effect on the text

 …….

2. Italic tag

This implements italic effects on the text.

…….

3. Underline tag

This is used to specify that the selected text be displayed with underline.

<u>. . . </u>

4. Big tag

This is to specify that the selected text be displayed as bigger font size as compared to the font for the rest of the text.

<big>. . .</big>

5. Small tag

This is to specify that the selected text be displayed as smaller font size as compared to the font for the rest of the text.

<small>. . .</small>

6. strong tag

This tag is used to always emphasized the text

. . .

7. sub and sup tag

These tags are used for subscript and superscript effects on the text.

_{. ..}

^{. ..}

8. **<emp> tag:** which is used to display content in italic.

Syntax: <emp> ---- </emp>

9. **<strike> tag :** This tag is used to draw a strikethrough on a section of text

Syntax: <strike> ---- </strike>

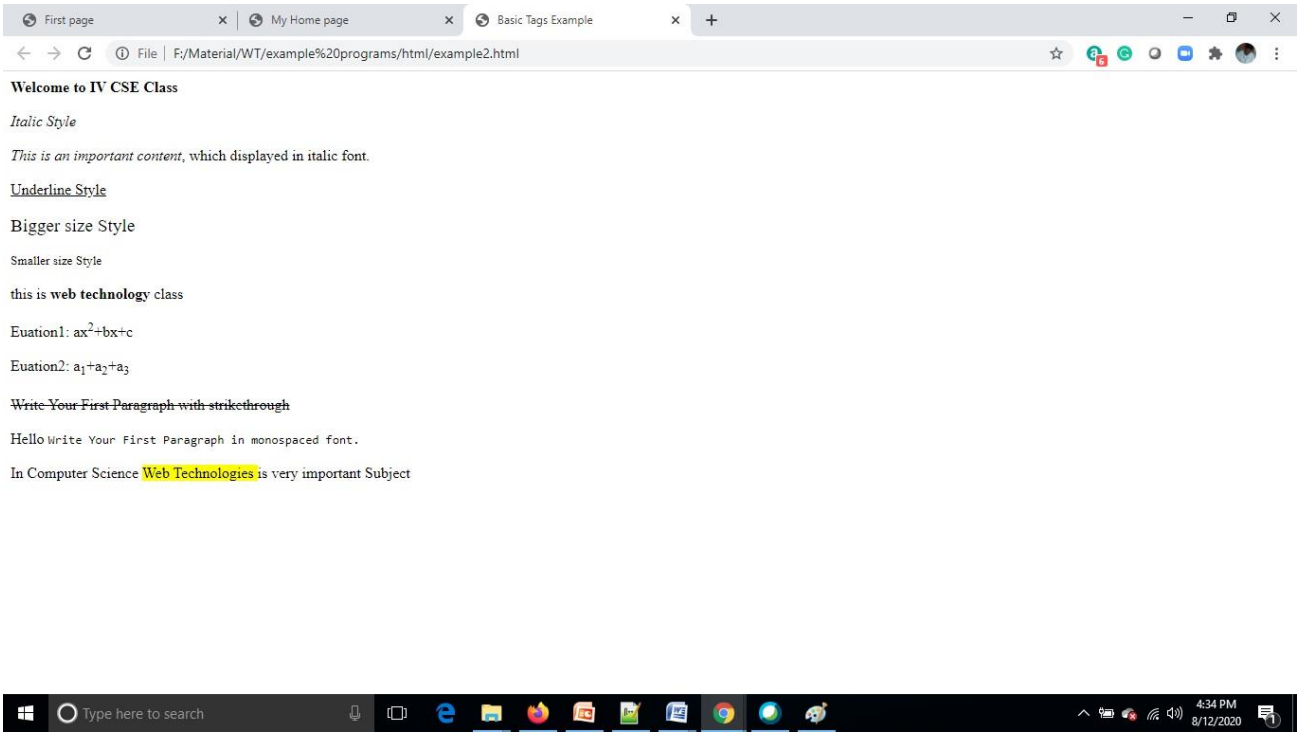
10. **<mark> tag :** This tag is used to highlight text.

Syntax: <mark> ---- </mark>

Example Program:

```
<html>
<head>
    <title>Basic Tags Example</title>
</head>
<body>
<b> Welcome to IV CSE Class</b>
<br><br>
<i> Italic Style </i>
<br><br>
<em>This is an important content</em>, which displayed in italic font.
<br><br>
<u> Underline Style </u>
<br><br>
<big> Bigger size Style </big>
<br><br>
<small> Smaller size Style </small>
<br><br>
this is<strong> web technology </strong>class
<br><br>
Euation1: ax<sup>2</sup>+bx+c
<br><br>
Euation2: a<sub>1</sub>+a<sub>2</sub>+a<sub>3</sub>
<br><br>
<strike>Write Your First Paragraph with strikethrough</strike>
<br><br>
Hello <tt>Write Your First Paragraph in monospaced font.</tt>
<br><br>
 In Computer Science <mark> Web Technologies </mark> is very important Subject
 </body>
</html>
```

Output:



2. Block-formatting tags:

You can use following tags to format blocks of text within HTML document. Some of the most frequently used Block-formatting tags are :

1. Body tag (<body>)

Body tag contains some attributes such as bgcolor, background etc. *bgcolor* is used for setting the background color of a webpage which takes background color name or hexadecimal number such as #000000 to #FFFFFF and background attribute used for setting mage as a background for webpage and it will take the path of the image which you can place as the background image in the browser.

Syntax:

<body bgcolor=”name/#rrggbb” background=”image name”> . . . </body>

Example:

<body bgcolor=”#F2F3F4” background= “c:\vijay\image.jpg”>...</boby>

2. Paragraph tag ( )

Most text is part of a paragraph of information. Each paragraph is aligned to the left, right or center of the page by using an attribute called as align.

Syntax:

<p [align=”left” | “right” | “center”] > . . .

3. Heading tag(<Hn>)

HTML is having six levels of heading that are commonly used. The largest heading tag is <h1>. The different levels of heading tag besides <h1> are <h2>, <h3>, <h4>, <h5> and <h6>. Each heading tag has an attribute called as align which can be set

to left, center, or right. By default all headings align left.

Syntax:

<h1 [align=”left” | “right” | “center”]> . . . </h1>
<h2 [align=”left” | “right” | “center”]> . . . </h2>
.
<h6 [align=”left” | “right” | “center”]> . . . </h6>

4. <hr> tag

This tag places a horizontal line across the screen. These lines are used to break up the page. This tag does not require an end tag. This tag also contains attributes which determines how the rule will be displayed. It can be aligned but by default is centered on the screen. The *size* attribute specifies the thickness of the rule in pixels. *noshade* draws the rule as a single thick line rather than giving it‟s default 3D appearance.

Syntax:

<hr align=”left” | “right” | “center” width=”nn%” size=”n” [noshade]/>.

5. base font

This specify the minimum text size for the entire webpage but not the headings. Syntax:

<basefont size=”n”>

6. font tag

This sets font type, size, color and relative values for a particular text. Absolute font sizes are can be set from 1 to 7. Relative font sizes are set by using +/- 1 to 7. The color of the text is set by *color* attribute. This takes hex value which represents the amounts of red, green and blue in a chosen color.

Syntax:

<font face=”font name ”size=”[+|-]n” color=”#rrggbb”>

7. tt tag

This tag is used to give teletype effect on the text

<tt>. . .</tt>

8. Line break tag

This tag is used to the break the line and start from the next line. It is an empty tag.

**
**

9. Pre-formatted text tag

It Considers spaces, new lines etc. and as it is prints the information.

<PRE>. . .</PRE>

10. MARQUEE Tag:

This is to create scrolling effect for the selected text in an HTML Page. When you use this tag, the selected text is highlighted and scrolls from right to left and vice versa. Syntax:

<marquee> . . .</marquee>

Marquee tag attributes:

a) align: sets the alignment of the text relative to the marquee. Set to top(default), middle or bottom.
b) behavior: Sets how the text in the marquee should move, It can be scroll(default), slide(text enters from one side and stops at the other end), or alternate(text seems to bounce from one side to other)
c) bgcolor: sets the background color for the marquee box
d) direction: sets the direction of the text for scrolling. It can be left(default), right, down or up.
e) height and width: to specify the height and width of a marquee text either in pixels or in % of the screenheight.
f) Loop: to specify the number of times a marquee will loop. -1 for infinite times.
g) scrolldelay: to control the speed at which marquee text is drawn on the screen. This attribute specifies the number of milliseconds between each successive draw of the marquee text.
h) scrollamount: to specify the number of pixels that marquee text moves in every movement.

11. Blinking text Tag

<blink>……</blink> displays enclosed text as blinking on and off approximately once a second. This tag only works in Netscape navigator

12. Escape sequences

These are character escape sequence which are required if you want to display characters that HTML uses as control sequences. All of these characters starts with an ampersand **(&)** and are terminated with a semicolon **(;)**

& <	>		" ©
& <	>		‘ ©

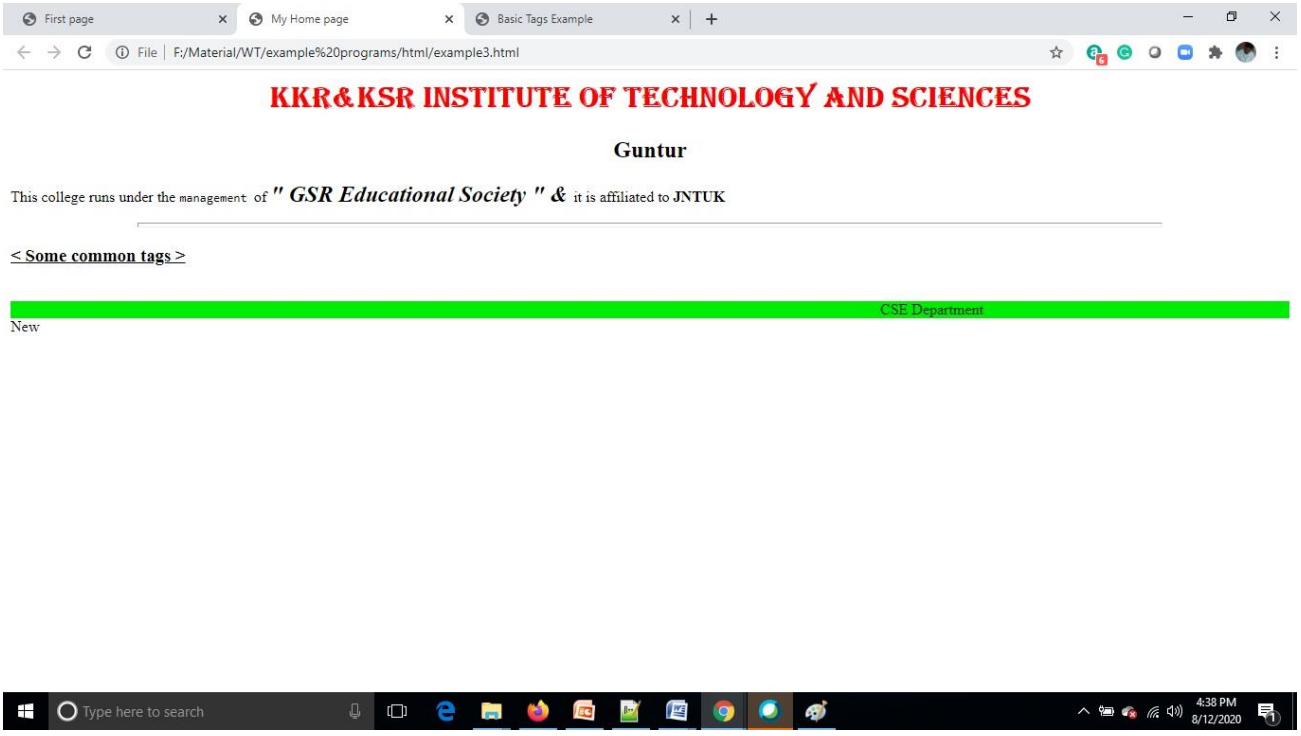
Example:

HTML code to implement common tags.

```
<html>
<head> <! -- This page implements Formatting tags -->
<title> My Home page </title>
```

```
</head>
<body >
<h1 align="center"> <font face="Algerian" color="red">KKR&KSR INSTITUTE OF
TECHNOLOGY AND SCIENCES</font></h1>
<h2 align="center"> Guntur</h2>
<basefont size=4>
<p> This college runs under the <tt> management </tt> of <font size=5> <b><i>&quot GSR
Educational Society &quot &amp </i></b></font>
it is affiliated to <strong> JNTUK
</strong>
<hr size=5 width=80%>
<h3> <u>&lt Some common tags &gt</u> </h3><br>
<marquee align=”middle” behavior=”slide” bgcolor=”red” direction=”down”> CSE Department
</marquee>
<blink>New</blink>
</body>
</html>
```

Output:



HYPERLINKS

The real power and flexibility of HTML is in Hyperlinks. Hyperlinks are created with anchor tag**(<a>).** It is used to create link with another web page.

Anchor tag:

The anchor tag is created by <a> . . . tags.

The tag has three sections:

a. the address of the referenced document
b. a piece of text displayed as link
c. closing tag.

The "Text" between the <a> . . . tags acted as a hyperlink. This text is called *"hypertext".* When you click on this text, the linked page or file will be displayed.

A link can be an image or any other HTML element such as E-Mail, Buttons, etc..

Syntax:

 Text

1. **Href:** Used to specify the path and file name of the HTML page that you need to access by using a hyperlink

By default, links will appear as follows in all browsers:

- An unvisited link is underlined and blue
- A visited link is underlined and purple
- An active link is underlined and red

2. **Target**: This attribute is used to specify the location where linked document is opened.

The target attribute can have one of the following values:

- _self - Default. Opens the document in the same window/tab as it was clicked
- _blank - Opens the document in a new window or tab
- _parent - Opens the document in the parent frame
- _top - Opens the document in the full body of the window

3. **Title:** It specifies extra information about an element. The information is most often shown as a tooltip text when the mouse moves over the element.
4. **Name:** It is used to create a link to a particular section of a given webpage by

using **name** attribute

Color and Images

Color is essential to the Web experience; it brings life to pages and makes them exciting. Color can be used in number of places on a Webpage: The background can be colored, individual elements can be altered, and links which already colored can be changed

To change the colors of links or of the page background hexadecimal values are placed in the <body> tag:

Syntax:

<body bgcolor = "#rrggbb" text = "#rrggbb" link= "#rrggbb" vlink= "#rrggbb" alink = "#rrggbb">

The *vlink* attribute sets the color of links visited recently, *alink* the color of a currently active link. The six figure hexadecimal values must be enclosed in double quotes and preceded by a hash(#).

Images

Images are second aspect of pleasant Web experience. You can add images to an HTML page to either improve its appearance or present important information. To add images to an HTML page, you can use several image formats. These formats include *„gif", „jpg" and „png".*

Images can be added in two different ways:

1. By using "*background"* attribute of <body> tag
2. By using <img/> tag

The <BODY> Tag: background attribute

You can use the background attribute of the <BODY> tag to add image as a background in an HTML page.

Syntax:

<body background = "URL">. …</body>

This tag will set a background image present in the URL.

The Image tag

Images can be used to easily understand the most complex aspects in web page. An image added using the image (<img>) tag and it occupies space within the HTML page.

Syntax:

<img src="URL" height="n" width="n" align = "left" | "right" | "top"|"middle" alt = " string ">

Attributes:

a) src : used to specify the name of the file. When using <img> tag, it ismandatory to

specify a value for the **SRC** attribute

b) align : used to specify the vertical alignment of an image

c) height : used to specify the vertical area that an image will occupy in HTML page

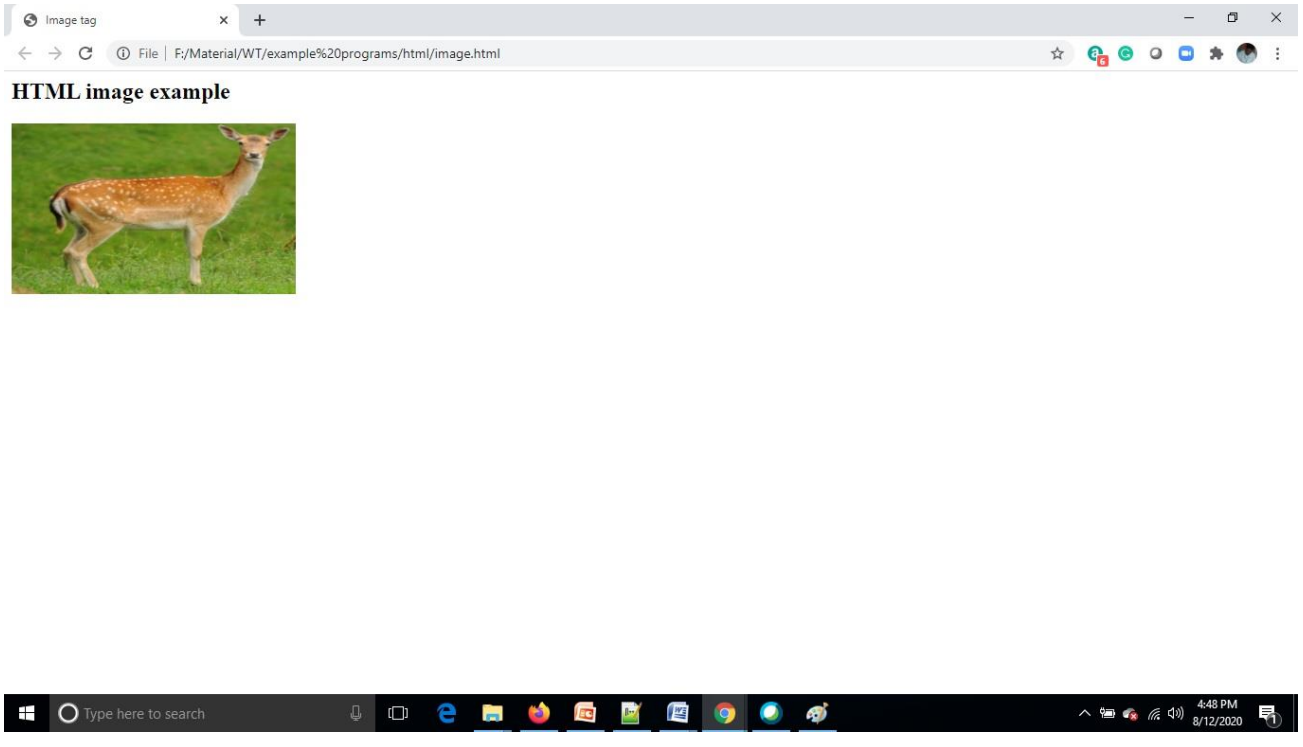
d) Width : used to specify the horizontal area that an image will occupy in HTML page

e) alt : used to specify the text when browser unable to display the image orimage not available.

Example : HTML code that implements color and image

```
<html>
<head>
    <title>Image tag</title>
</head>
<body>
<h2>HTML image example </h2>
<img src="F:\Material\WT\example programs\animal.jpg" height="180" width="300" alt="animal image">
</body>
</html>
```

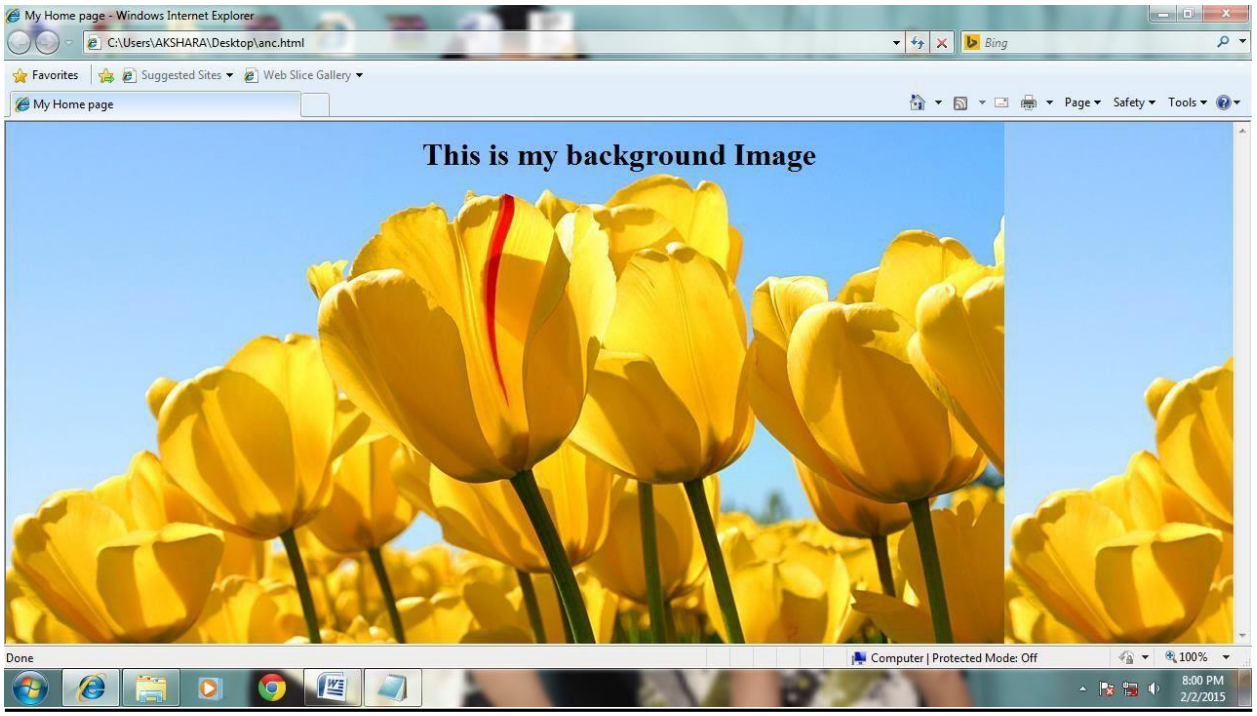
Output:



Example : HTML code that implements background image

```
<html>
```

```
<head> <title> My Home page </title> </head>
<body background="C:\Users\Public\Pictures\Sample Pictures\Tulips.jpg">
<h1 align="center"> This is my background Image</h1>
</body> </html>
```



Output:

Lists

One of the most effective ways of structuring a web site is to use lists. Lists provides straight forward index in the website. HTML provides three types of lists.

1. Unordered / Bulleted list
2. Ordered / Numbered list and
3. Definition/Description list

Lists can be easily embedded within other lists to provide complex yet readable structures.

**
. . .:**

The Ordered / Unordered lists are each made up of set of list items. These list items are added with ***<li>*** tag. Elements of a list item may format with any of the usual formatting tags and may be images or hyperlinks.

Unordered Lists

Unordered lists are also called Un-numbered lists. The Unordered list elements are used to represent a list of items. Using <UL> tag is used to create unordered lists in HTML. Which is paired tag, so it requires ending tag that is </UL>. The list of items is included in

between <UL>…..</UL>. The *TYPE* attribute can also be added to the <UL> tag that indicates the displayed bullet along with list of item is **square, disc** or **circle**. By default it is disc.

Syntax:-

```
<ul type=disc|circle|square >
    <li> item1</li>
    <li> item1</li>
        ----
        ----
</ul>
```

Ordered Lists:-

Ordered lists are also called **Numbered** or **Sequenced lists**. In the ordered list the list of items have an order that is signified by numbers, hence it some times called as number lists. Elements used to present a list of items.

An orders list should start with the <OL> element, which is immediately followed by a <LI> element which is same as <LI> in unordered list. End of ordered lists is specified with ending tag </OL>.

Different Ordered list types like roman numeral list, alphabet list etc. can be specified with ***TYPE*** attribute.

Another optional parameter with <OL> tag is ***START*** attribute, which indicates the starting number or alphabet of the ordered list. For example TYPE=”A” and START=5 will give list start with letter E.

Different Ordered list types

Type=”1” (default) e.g.1,2,3,4…..

Type=”A” Capital letters e.g.A,B,C…

Type=”a” Small letters e.g. a,b,c……

Type=”I” Large roman letters e.g. I, II, III,…

Syntax:-

```
<ol type=1|I|i|A|a start='starting' reversed >
    <li> item1</li>
    <li> item1</li>
        ----
        ----
</ol>
```

Definition/Description List:

Definition List displays elements in definition form like in dictionary. The <dl>, <dt> and <dd> tags are used to define description list.

The 3 HTML description list tags are given below:

<dl> tag defines the description list.

<dt> tag defines data term.

<dd> tag defines data definition (description).

Nested Lists: Lists can be nested that is Nested Lists is list with in another list.

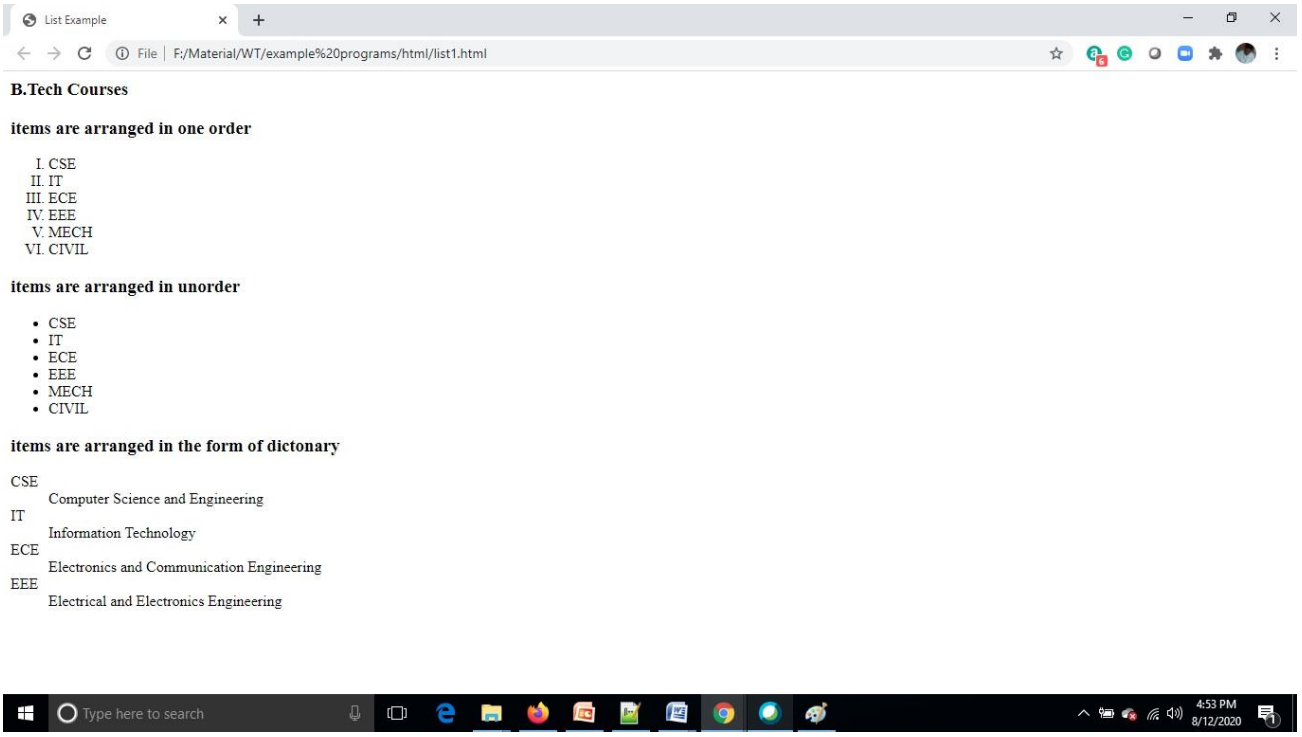
Example:

Write a HTML program for displaying names of B.Tech Courses displayed in with different styles of list.

```
</head>
<body>
<h3>B.Tech Courses </h3>
<h3> items are arranged in one order</h3>
<ol type="I">
<li>CSE </li>
<li>IT </li>
<li>ECE</li>
<li>EEE</li>
<li>MECH</li>
<li>CIVIL</li>
</ol>
<h3> items are arranged in unorder</h3>
<ul type="disc">
<li>CSE </li>
<li>IT </li>
<li>ECE</li>
<li>EEE</li>
<li>MECH</li>
<li>CIVIL</li>
</ul>
<h3> items are arranged in the form of dictonary</h3>
<dl>
<dt>CSE</dt>
```

```
        <dd>Computer Science and Engineering</dd>
    <dt>IT</dt>
        <dd>Information Technology</dd>
     <dt>ECE</dt>
        <dd>Electronics and Communication Engineering</dd>
      <dt>EEE</dt>
        <dd>Electrical and Electronics Engineering</dd>
      </dl>
    </body>
    </html>
```

Output:



Tables

table tag is used to display data in tabular form (row * column). There can be many columns in a row.

We can create a table to display data in tabular form, using <table> element, with the help of <tr> , <td>, and <th> elements.

In Each table, table row is defined by <tr> tag, table header is defined by <th>, and table data is defined by <td> tags.

You can use various elements to specify the details of a table. Many table elements also take attributes, which allows you to further specify the look of the table.

- TABLE
- Table row
- Table data
- Table Heading

The TABLE Element

The TABLE element is the container element for table and uses the <TABLE>... </TABLE> tags to enclose all the other table tags. If the <TABLE> tag is omitted or not closed, the browser ignores all the other tags that you specify for the table. Everything that we write between these two tags will be within a table. The attributes of the table will control in formatting of the table.

The <TABLE> tag has the following attributes:

a. align: used to specify the alignment of a table in a HTML page.
b. border: used to specify the thickness of the table border in pixels.
c. bgcolor: used to specify the background color for the table.
d. frame: used to specify the which side of the outer border is visible in the browser. You must specify the border attribute before you specify the FRAME attribute.

Values for the FRAME Attribute

Value	Description
VOID	Removes all External borders
BOX	Displays a box around the table i.e. all four sides of the table
HSIDES	Displays an external border at the top and bottom of the table
VSIDES	Displays an external border at the right and left of the table
LHS	Displays an external border only on the left hand side of the table
RHS	Displays an external border only on the right hand side of the table
ABOVE	Displays an external border only on the top of the table
BELOW	Displays an external border only on the bottom of the table

e. rules: used to specify the borders between cells. You must specify the border attribute before you specify the RULES attribute.

Values for the RULES Attribute

Value	Description
NONE	Removes all internal rules from the table
ROWS	Displays rules between rows of the table
COLS	Displays rules between columns of the table
ALL	Displays rules between all rows and columns

f. cellspacing: used to specify the spacing between cells in pixels.
g. cellpadding: used to specify the spacing between cell content and cell wall in pixels.
h. height: used to specify the height of table in pixels or %.
i. width: used to specify the width of table in pixels or %.

Syntax:

<TABLE align="left | right | center" border="n" bgcolor="#rrggbb" cellspacing="n" cellpadding="n" frame="value" rules="value" height="n | %" width="n | %">

. . .

</TABLE>

The Table row element

You use table row element to create rows in a table. The Table row element uses the <TR> tag to create a row. You can use <TR> tag within <TABLE> tags. <TR> tag used as container for the row.

The<TR> tag has the following attributes:

a. align: used to specify the horizontal alignment of the contents for cells of a row.
b. valign: used to specify the vertical alignment of the cell content for all cells ofthe row.
c. bgcolor: used to specify the background color of the row.

Syntax:

<TR *align="left | right | center" align="top | bottom |middle" bgcolor="#rrggbb"* > ….. </TR>

The Table data element

You use the Table data element to create data cells. The Table data element uses <TD> tag within the <TABLE> and <TR> tags to create data cells. A data cell only appears with in a table row.

The <TD> tag has the following ***attributes:***

a. colspan: used to specify the number of columns the cell can span.
b. rowspan: used to specify the number of rows the cell can span
c. align: used to specify the horizontal alignment of the data within a cell.
d. valign: used to specify the vertical alignment of data within the cell.
e. bgcolor: used to specify the background color of the cell.

Syntax:

<TD *align="left | right | center" align="top | bottom |middle" bgcolor="#rrggbb" colspan="n" rowspan="n"*> ….. </TD>

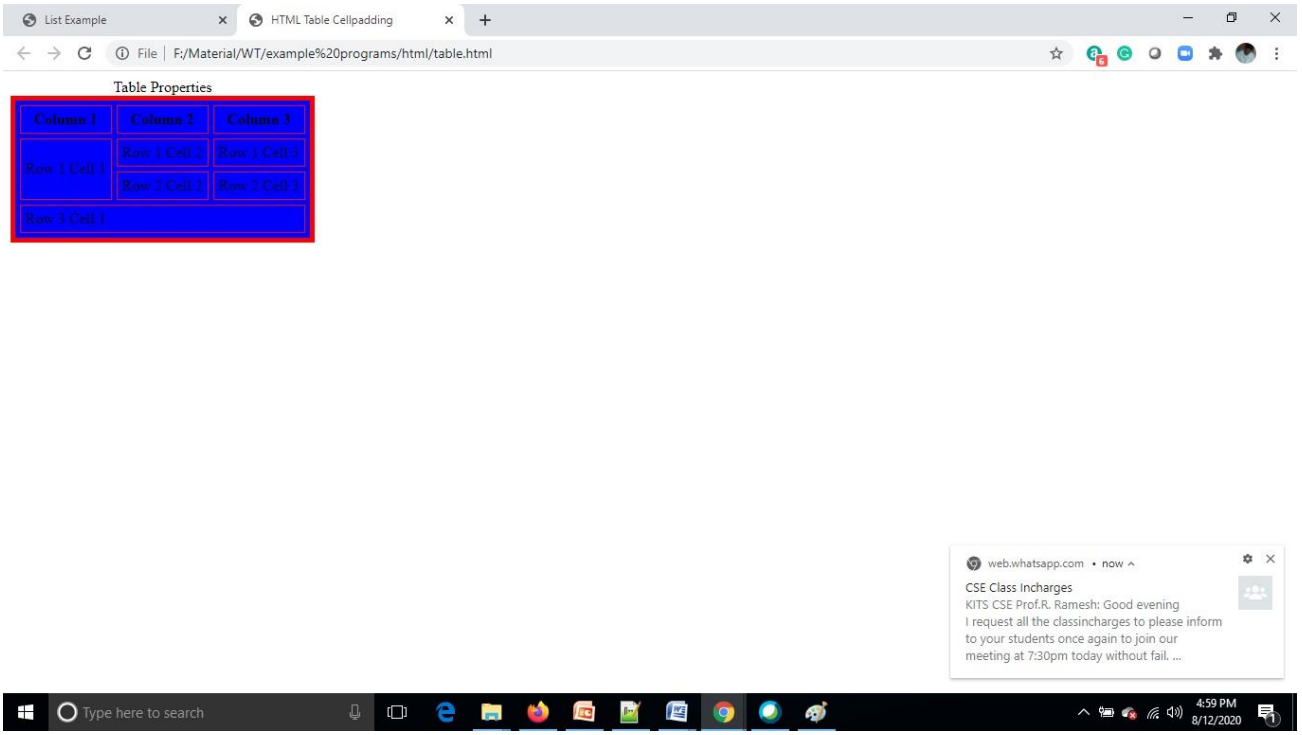
The Table heading element

You use the table heading element to create heading cells for table rows, columns, or both. The table heading element uses <TH> tag to create heading cells. The <TH> tag is exactly same as <TD> tag, except that the text displayed in bold. <TH> tag and <TD> tag share the same properties.

Example: HTML Program that printstable

```
<html>
<head>
<title>HTML Table Cellpadding
</title>
</head>
<body>
<table border = "5" cellpadding="5" cellspacing="5" bordercolor="red" bgcolor="blue">
<caption> Table Properties </caption>
 <tr>
          <th>Column 1</th>
          <th>Column 2</th>
          <th>Column 3</th>
        </tr>
        <tr>
          <td rowspan = "2">Row 1 Cell 1</td>
          <td>Row 1 Cell 2</td>
          <td>Row 1 Cell 3</td>
        </tr>
        <tr>
          <td>Row 2 Cell 2</td>
          <td>Row 2 Cell 3</td>
        </tr>
        <tr>
          <td colspan = "3">Row 3 Cell 1</td>
        </tr>
</table>
  </body>
</html>
```

Output:



Frames:

HTML frames are used to divide your browser window into multiple sections where each section can load a separate HTML document. A collection of frames in the browser window is known as a frameset.

Each frame within a webpage has the following properties:

- It has unique name
- It displays an HTML document independent of other frames
- Its size can be dynamically changed according to the size of the content in HTML page

Creating Frames

To use frames on a page we use <frameset> tag instead of <body> tag. The <frameset> tag defines, how to divide the window into frames. The **rows** attribute of <frameset> tag defines horizontal frames and **cols** attribute defines vertical frames. Each frame is indicated by <frame> tag and it defines which HTML document shall open into the frame.

The <FRAMESET> . . . </FRAMESET> Tags

The <FRAMESET> . . . </FRAMESET> tags are the container tags for all other tags that are used to create frames. These tags are used to specify the size and number of the frames to be created in an HTML Page. The frames within these tags are jointly referred to as

a *frameset*.

Syntax:

```
<FRAMESET cols=”n,n” rows=”n,n”>
    <FRAME> Tags
</FRAMESET>
```

Attributes

COLS: The COLS attribute is used to specify the number of vertical frames within HTML Page.

ROWS: The ROWS attribute is used to specify the number of horizontal frames within HTML Page. To specify the values of the attributes, use a comma-separated list of values. The values can be of the following two types:

- Value: This numeric value is used to specify number of pixels that a frame will occupy in the browser window.
- Value%: This value is used to specify percentage of space that each frame will occupy in the browser window. The total of all the values specified in the comma-separated list is 100%.

In order to divide into two columns we can use the following syntax

<FRAMESET COLS=”25%,75%>

<frame> tags

</FRAMESET>

In the second diagram we have three rows so by using rows parameter of frameset, we can divide logically the window into three rows.

<FRAMESET ROWS=”20%,70%,10%>

<frame> tags

</FRAMESET>

According to above code, first row occupies 20% of the window, third row occupies 10% of the window, second row occupies 70% of the window.

<FRAME> Tag

You use the <FRAME> Tag to create frame with in *frameset*. You can use the following attributes to specify the name of the frame and the HTML page that you need to display in the frame.

- *SRC*: SRC attribute to specify the HTML page that you want to display in a frame.
- *NAME*: NAME attribute to specify the name of the frame. This *NAME* is used by the anchor element to display the linked HTML Pages within frame.
- *SCROLLING*: attribute used to add scrollbars to a frame. The SCROLLING attribute takes three value: YES,NO, AUTO.

- The value *YES* specifies that a scrollbar should always be visible on the frame
- The value *NO* specifies that the scrollbar should never be visible on the frame
- The value *AUTO* specifies the browser to automatically display or remove scrollbars from a frame

- *FRAMEBORDER:* attribute to specify whether a frame should have a border. The value 0(zero) specifies no border. If you specify any other numeric value, the border is displayed.
- *NORESIZE:* By default, You can resize a frame by dragging its borders. You can restrict this by specifying the NORESIZE attribute.

Syntax:

<FRAME SRC = ”URL” NAME =” myframe” SCROLLING = ”yes | no | auto”
FRAMEBORDER = ”0|1” [NORESIZE]/>

Nested Framesets

Some times it is required to divide your window into rows and columns, and then there is requirement of nested framesets. *Frameset with in another frameset is known as nested frameset.*

Example:

home.html

```
<frameset rows="20%,*">
    <frame name="fr1" src="example.html">
    <frameset cols="25%,*">
        <frame name="fr2" src="list1.html">
        <frame name="fr3" src="image.html">
    </frameset>
</frameset>
```

image.html

```
<html>
<head>
    <title>Image tag</title>
</head>
<body>
<h2>HTML image example </h2>
<img src="F:\Material\WT\example programs\animal.jpg" height="180" width="300" alt="animal image">
</body>
</html>
```

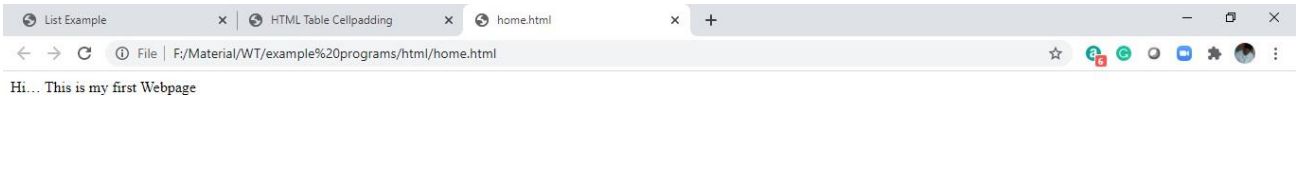
List1.html

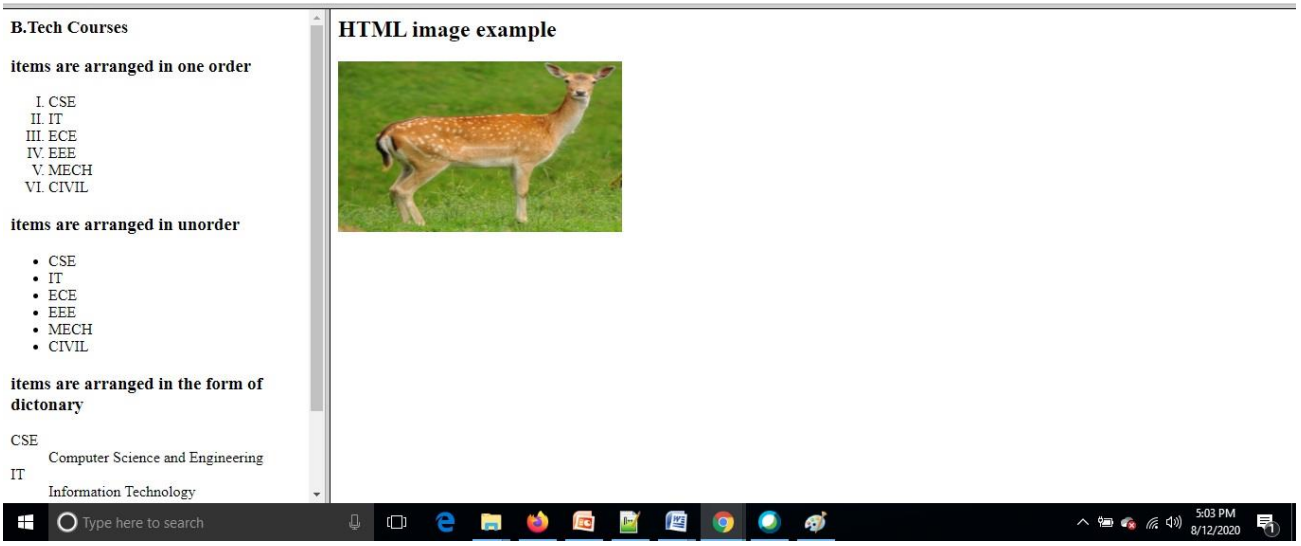
```
</head>
<body>
<h3>B.Tech Courses </h3>
<h3> items are arranged in one order</h3>
<ol type="I">
<li>CSE </li>
<li>IT </li>
<li>ECE</li>
<li>EEE</li>
<li>MECH</li>
<li>CIVIL</li>
</ol>
<h3> items are arranged in unorder</h3>
<ul type="disc">
<li>CSE </li>
<li>IT </li>
<li>ECE</li>
<li>EEE</li>
<li>MECH</li>
<li>CIVIL</li>
</ul>
<h3> items are arranged in the form of dictonary</h3>
<dl>
                        <dt>CSE</dt>
<dd>Computer Science and Engineering</dd>
                      <dt>IT</dt>
<dd>Information Technology</dd>
  <dt>ECE</dt>
<dd>Electronics and Communication Engineering</dd>
    <dt>EEE</dt>
<dd>Electrical and Electronics Engineering</dd>
    </dl>
</body>
</html>
```

example.html

```
<HTML>
    <HEAD>
        <TITLE>First page </TITLE>
    <HEAD>
    <BODY>
        Hi… This is my first Webpage
    </BODY>
</HTML>
```

Output:





Forms

Forms are the best way of adding interactivity to a web page. A form is a collection of fields that you can use to gather information from people visiting your site. Forms acts as a means of user interactions on the Web.

Working of a Form

A Form contains certain text fields, radio buttons, check boxes and buttons for entering data. When data entry is complete, the user submits the form for processing using SUBMIT button. This communicates to the browser that the user has completed data entry. The browser sends data to the server. The server process the details and sends response to browser, which displayed on the browser.

The FORM Tag

A form is defined using the <FORM>. . . </FORM> tags. The FORM tag has four attributes:

1. NAME
2. ACTION
3. METHOD
4. ENCTYP

E Syntax:

<FORM action=”URL” method = “POST” | “GET” enctype=”encoding”> . . . </FORM>

NAME: used to specify the name of the form

ACTION: The ACTION attribute of the form tag is used to specify the name, and location of a script that will be used to process the data.

METHOD: The METHOD attribute of the form tag is used to specify the method by which browser sends the data to the server for processing. The type method can be either GET or POST. When *get* is used, the data is included as part of the URL. The *post* method encodes the data within the body of the message. Post can be used to send large amount of data, and it is more secure than *get*.

ENCTYPE: The ENCTYPE attribute is used to specify the content type for encoding the data. There are two content types:

1. application/x-www-form-urlencoded(default)
2. multipart/form-data.

ENCTYPE is used only when you need to upload a file.

The INPUT Element

The INPUT Element is used to specify input fields, such as single line text fields, passwords fields, check boxes, radio buttons etc.,. The INPUT element is defined by using <INPUT> tag.

You can create the following controls using the INPUT tag:

- Text
- Password
- Radio
- Checkbox
- Button
- Submit
- Reset
-

Syntax:

<INPUT type = “text” | “password” | “checkbox” | “radio” | “submit” name=”string” value=”string” size=”n” maxlength=”n” />

The tags used inside the form tag are:

In the above tag, the attribute *type* is used to implement text, password, checkbox, radio, submit, reset button and button.

Text: It is used to input the characters of the size n and if the value is given than it is used as a default value. It uses single line of text. Each component can be given a separate name using the nameattribute.

Password: It works exactly as text, but the content is not displayed to the screen, instead * is displayed.

Radio: This creates a radio button that accepts single value from a set of alternatives. They are always grouped together with a same name but different values.

Checkbox: This creates a checkbox that enables you select multiple values from a set of alternatives. They are always grouped together with a same name but different values.

Submit: This creates a Submit button which displays the value attribute as its text. It is used to send the data to the server.

Reset: This creates a Reset button which displays the value attribute as its text. It is used to clear the form data.

Button: This creates a user-defined button which displays the value attribute as its text. It works according to user requirement.

The SELECT Element

The SELECT Element is a container element. This element is used to create a list box or drop-down list box.

Syntax:

<SELECT NAME=”string” SIZE=”n” MULTIPLE> . . .</SELECT>

Where

MULTIPLE: Indicates that the user is allowed to make several selections from list box

*NAME: repre*sents name of the list box

SIZE: Specifies the number of visible items in a list box.

The OPTION Element

It represents a choice in the list box and only occurs in SELECT element.

Syntax:

<OPTION VALUE=”String” SELECTED>……</OPTION>

SELECTED: indicates the option is initially selelcted

VALUE: value that indicates a particular option selected.

The TEXTAREA Element

This creates a multi-line plain text field into which the user can enter anything they like. The area will be sized at rows by cols but supports automatic scrolling.

Syntax:

<TEXTAREA NAME=”string” ROWS=”n” COLS=”n”> . . . </TEXTAREA>

Where

NAME: Specify the name of the text field

ROWS: Set the number of rows of text that will be visible

COLS: Set the number of columns of text that will be visible

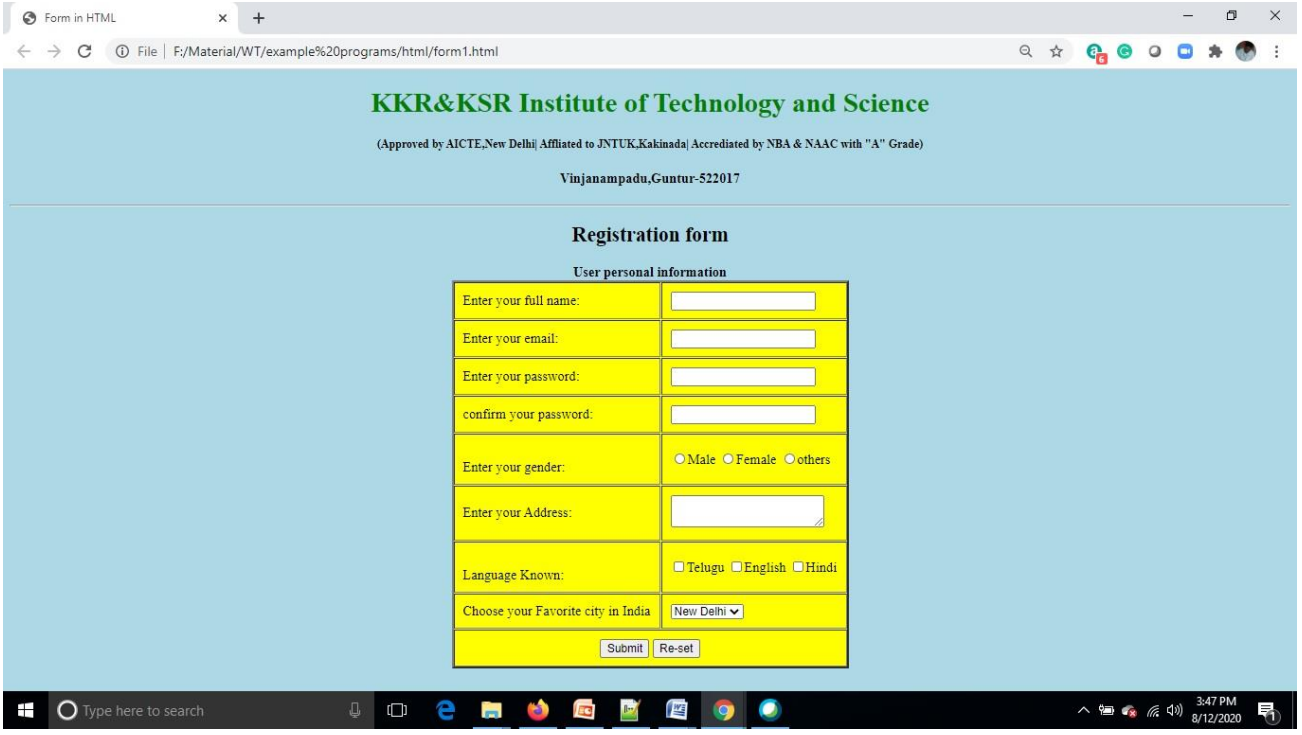
Example:

```
<html>
 <head>
  <title>Form in HTML</title>
</head>
 <body bgcolor="lightblue">
      <center>
            <h1><font color="green"> KKR&KSR Institute of Technology and
Science</font></h1>
            <h5>(Approved by AICTE,New Delhi| Affliated to JNTUK,Kakinada| Accrediated
by NBA & NAAC with "A" Grade)</h4>
            <h4> Vinjanampadu,Guntur-522017</h4>
            <hr>
    <h2>Registration form</h2>
   <form>
          <legend><strong>User personal information</strong></legend>
            <table border="2" bgcolor="yellow" cellpadding="10" cellspacing="2">
```

```
<tr><td>
<label>Enter your full name:</label> </td><td>
<input type="text" name="name"></td></tr>
<tr><td>
<label>Enter your email:</label> </td><td>
<input type="email" name="email"></td></tr>
<tr><td>
<label>Enter your password:</label> </td><td>
<input type="password" name="pass"></td></tr><tr><td>
<label>confirm your password:</label> </td><td>
<input type="password" name="pass"></td></tr><tr><td>
<br><label>Enter your gender:</label> </td><td>
<input type="radio" id="gender" name="gender" value="male"/>Male
<input type="radio" id="gender" name="gender" value="female"/>Female
<input type="radio" id="gender" name="gender" value="others"/>others
</td></tr><tr><td>
<label>Enter your Address:</label> </td><td>
<textarea></textarea></td></tr><tr><td>
<br><label>Language Known:</label> </td><td>
<input type="checkbox" id="telugu" name="Telugu" value="Telugu"/>Telugu
<input type="checkbox" id="english" name="English" value="English"/>English
<input type="checkbox" id="hindi" name="hindi" value="hindi"/>Hindi
</td></tr><tr><td>
<label>Choose your Favorite city in India</label> </td><td>
<select>
<option>New Delhi</option>
<option>Indore</option>
<option>Jaipur</option>
<option>Jodhpur</option>
<option>Hydrabad</option>
<option>Mumbai</option>
<option>Bengaluru</option>
<option>Lucknow</option>
</select> </td></tr>
<tr><td colspan="2" align="center">
<input type="submit" value="Submit"> <input type="submit" value="Re-set"> </td></tr>
```

```
                    </table>
        </center>
    </form>
  </body>
</html>
```

Output:



Cascading Style Sheets

CSS stands for Cascading Style Sheets. CSS is used to apply the style in the web page which is made up of HTML elements. CSS is easy to learn and understood but it provides powerful control over the presentation of an HTML document. A Style sheet is a collection of rules

Advantages:

1. Solves a big problem
2. CSS saves time
3. Provide more attributes
4. It supports reusability (Inheritance).

Defining a style:

A rule is a statement that defines a style.

A rule consists of two parts:

1. Selector
2. Declaration

Syntax:

Selector{Property1: value1; Property2: value2; .....;}

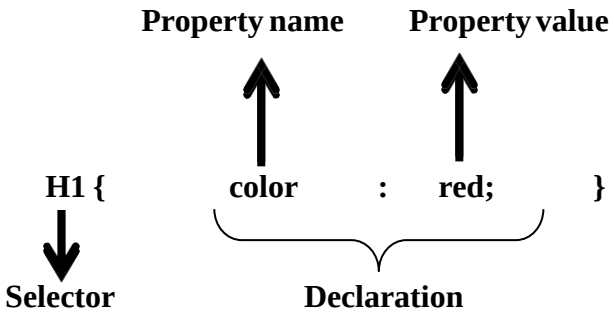
A selector is an HTML statement that is linked to a specific style.

A declaration defines the style for selector. A declaration again has two parts:

1. Property name
2. Property value

Property name is name of the property for which you need to define a style. Property value is the value assigned to property name.

Example:



Types of style sheets

You can define three types of style sheets:

1. Inline style sheets
2. Internal/Embedded style sheets

3. External style sheets

An inline sheet applies style to a particular element in a webpage. Embedding a style sheet is for defining styles to a single webpage. An external style sheet can apply styles to multiple web pages.

Inline style sheets

The style included within the tag is known as *inline style sheet*. To include style definition within tag, you use "STYLE" property.

Syntax:

<TAG STYLE="propertyname: property value;">. . . </TAG>

An Inline style affects only the element for which the style is defined. The style does not affect any other element even if the element is of the same type.

Internal style sheets

In this, the style definitions are enclosed in <STYLE> . . . </STYLE> tags. In turn, this STYLE tag pairs must be placed in between <HEAD> . . . </HEAD> tags. The style defined in <HEAD> section is applied to whole document. The styles defined for the tags within <STYLE> tag will reflect for every instance of the tag in the entire document.

Syntax:

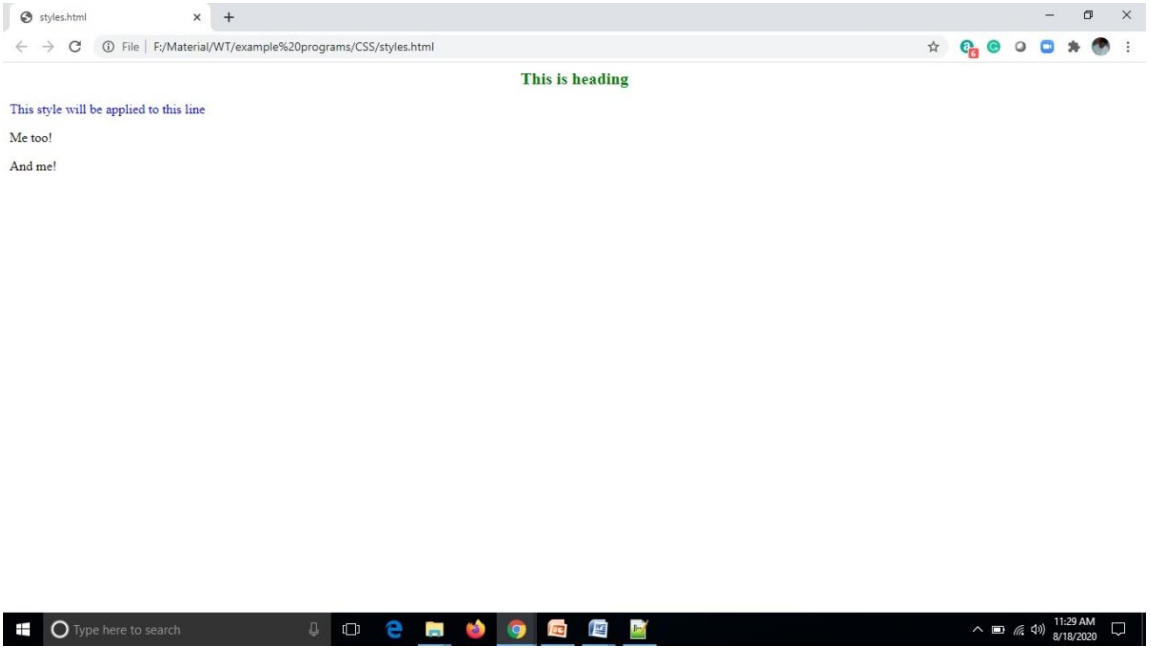
```
<HTML>
<HEAD>
    <STYLE TYPE="TEXT/CSS">
        <!-- STYLE DEFINITIONS -->
    </STYLE>
</HEAD>
<BODY> . . . </BODY>
</HTML>
```

Example:

```
<html>
  <head>
    <style>
      h1{
        color: green;
        font-size: 20px;
```

```
                }
            </style>
        </head>
        <body>
            <center>
                <h1>This is heading</h1></center>
                <p style="color:blue">This style will be applied to this line</p>
                <p id="para1">Me too!</p>
                <p>And me!</p>
        </body>
</html>
```

Output:



External Style Sheets

External CSS is used to apply CSS on multiple pages or all pages. Here, we write all the CSS code in a css file. Its extension must be **".css"** It uses the **<link>** tag on every pages and the **<link>** tag should be put inside the head section.

<LINK> Tag

External style sheets are called using the <*link*> tag which should be placed in the head section of an HTML document. This tag takes three attributes.

Attributes of the <link> tag:

- **rel** - When using an external stylesheet on a webpage, this attribute takes the value "stylesheet"
- **type** - When using an external stylesheet on a webpage, this attribute takes the value "text/css"

- **href** - Denotes the name and location of the external style sheet to be used.

Syntax:

<link rel ="stylesheet" type="text/css" href="filename.css">

Example:

External Style Sheet(ext.css)

```
body {
        background-color: lightblue;
    }
h1 {
        color: navy;
        margin-left: 20px;
   }
```

1st page(dept.html):

```
<html>
    <head>
        <link rel="stylesheet" type="text/css" href="ext.css">
    </head>
    <body>
        <center>
            <h1>
                <br/>CSE<br/>
                <br/>IT<br/>
                <br/>ECE<br/>
                <br/>EEE<br/>
                <br/>MECH<br/>
                <br/>CIVIL<br/>
            </h1>
        <center>
    </body></html>
```

2nd page(desc.html):

```
<html>
    <head>
        <link rel="stylesheet" type="text/css" href="ext.css">
    </head>
    <body>
        <center>
```

```
<br/> <br/>
<h1>Description of College</h1>
</center>
</body></html>
```

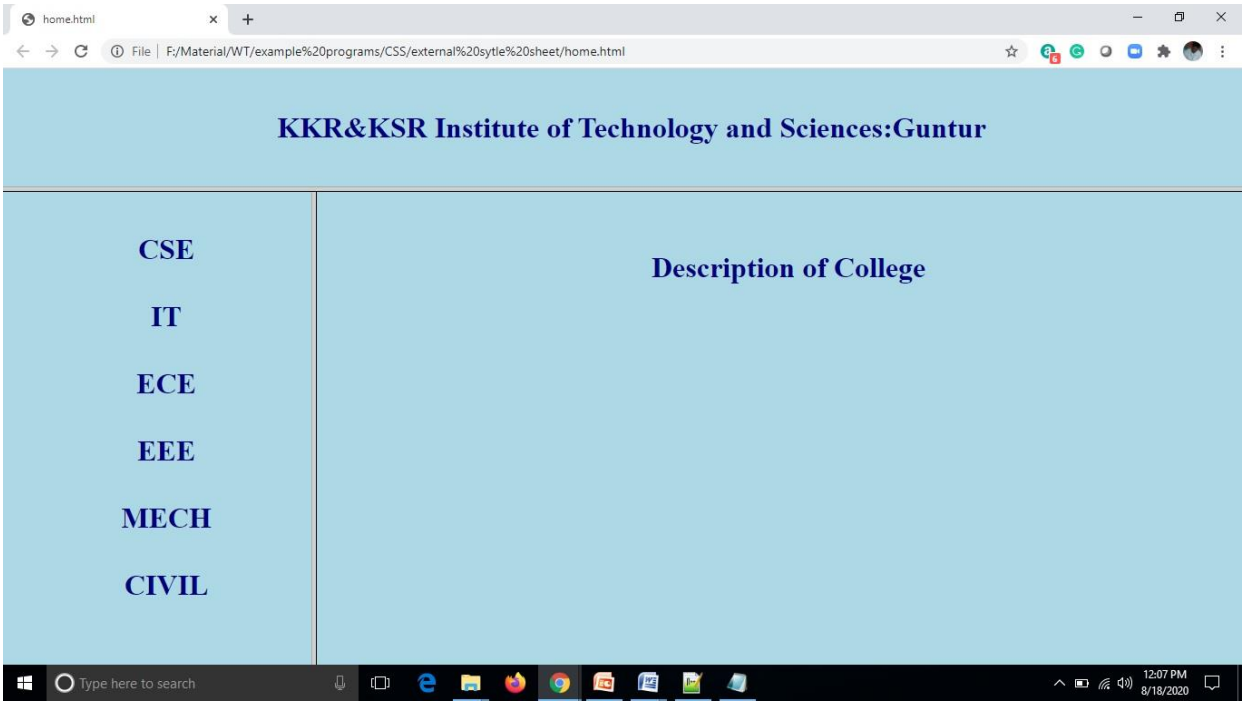
3rd Page(Home.html)

```
<frameset rows="20%,*">
<frame name="fr1" src="top.html">
<frameset cols="25%,*">
<frame name="fr2" src="dept.html">
<frame name="fr3" src="desc.html">
</frameset>
</frameset>
```

4th Page(top.html):

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="ext.css">
</head>
<body>
<center>
<br/><h1> KKR&KSR Institute of Technology and
Sciences:Guntur</h1>
</center>
</body></html>
```

Output:


home.html
File | F:/Material/WT/example%20programs/CSS/external%20sytle%20sheet/home.html
KKR&KSR Institute of Technology and Sciences:Guntur
CSE
IT
ECE
EEE
MECH
CIVIL
Description of College
Type here to search
12:07 PM
8/18/2020

CSS Selector:

These are used *to select the content you want to style*. Selectors are the part of CSS rule set.

There are five types of selectors

1. Element Selector
2. Id Selector
3. Class Selector
4. Universal Selector
5. Group Selector

1. Element Selector

The element selector selects the HTML element by name.

Syntax:

```
<style>
    tagname{CSS Style definitions}
</style>
<tagname>Content </tagname>
```

Example:

```
<style>
    center{color:red;font-size:30px}
</style>
<center>KKR&KSR INSTITUTE OF TECHNOLOGY AND SCIENCES</center>
```

In above example we apply color and font styles to content placed in between tag by using “element selector”

2. Id Selector

The id selector selects the id attribute of an HTML element to select a specific element. It is written with the hash character (#), followed by the id of the element.

Syntax:

```
<style>
    #idvalue
    {
        Style definitions
    }
</style>
<tagname id=”idvalue”>Content </tagname>
```

Example:

```
<style>
        #size1{color:blue}
</style>
<h6 id="size1">(Affliated to JNTUK,Kakinada|Accrediated by NBA &
NAAC with "A" Grade)</h6></center>
```

In above example we apply **blue color** as a style **to <h6>** tag by using id selector

3. Class selectors

Class selectors are used, when you want different elements to share the same format. A class selector definition starts with a period **(.)** followed by user defined and then style definition.

Syntax:

```
<style>
   .classvalue
   {
      Style definitions
   }
</style>
<tagname class=”classvalue”> Content </tagname>
```

Example:

```
<style>
        .center {
                        text-align: center;
                       color: green;
                }
</style>
<h3 class="center">Vinjanampadu, Guntur</h3>
```

In above example we apply color and text alignment styles to content placed in between <h3> tag by using “class selector”.

4. Group Selector

The grouping selector is used to select all the elements with the same style definitions. Grouping selector is used to minimize the code. Commas are used to separate each selector in grouping.

Syntax:

```
<style>
   Tagname1, tagname2, …..
   {
```

```
        Style definitions
    }
</style>
<tagname1> Content </tagname1>
<tagname2> Content </tagname2>
```

Example:

```
<style>
    h1, h2, p {
            text-align: center;
            color: green;
        }
</style>
    <h1>Computer Science and Engineering</h1>
    <h2>Web Technologies</h2>
    <p>This is a paragraph.</p>
```

In above example to apply the common text alignment and color of the text in between the content of <h1>, <h2> and <p> tags, this is nothing but group tag

Complete Example:

```
<html>
    <head>
        <style>
            center{color:red;font-size:30px}
            #size1{color:blue}
            .center {
                    text-align: center;
                    color: green;
                }
            h1, h2, p {
                    text-align: center;
                    color: green;
                }
        </style>
    </head>
    <body>
```

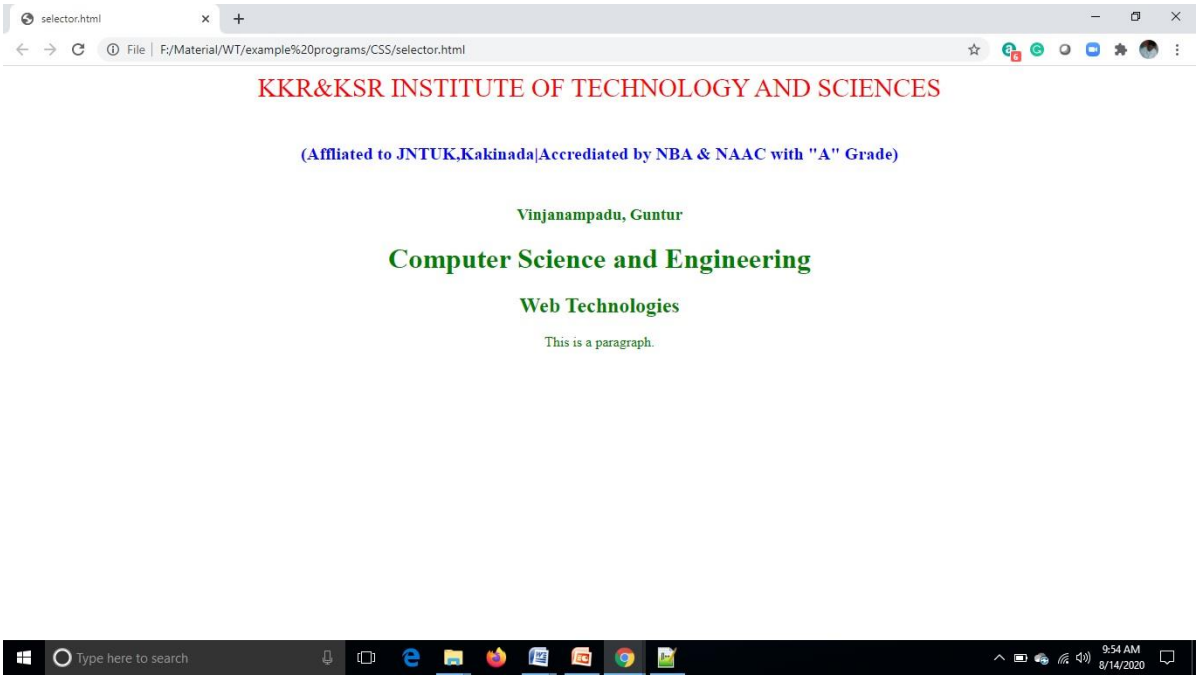
```
<!-- element selector -->
    <center>KKR&KSR INSTITUTE OF TECHNOLOGY AND SCIENCES</center>

<!-- id selector -->
<center>
    <h6 id="size1">(Affliated to JNTUK,Kakinada|Accrediated by NBA & NAAC with "A" Grade)</h6></center>

<!-- class selector -->
    <h3 class="center">Vinjanampadu, Guntur</h3>

<!-- group selector -->
    <h1>Computer Science and Engineering</h1>
    <h2>Web Technologies</h2>
    <p>This is a paragraph.</p>
</body>
</html>
```

Output:



5. Universal Selector

The universal selector is used as a wildcard character (*). It selects all the elements on the pages.

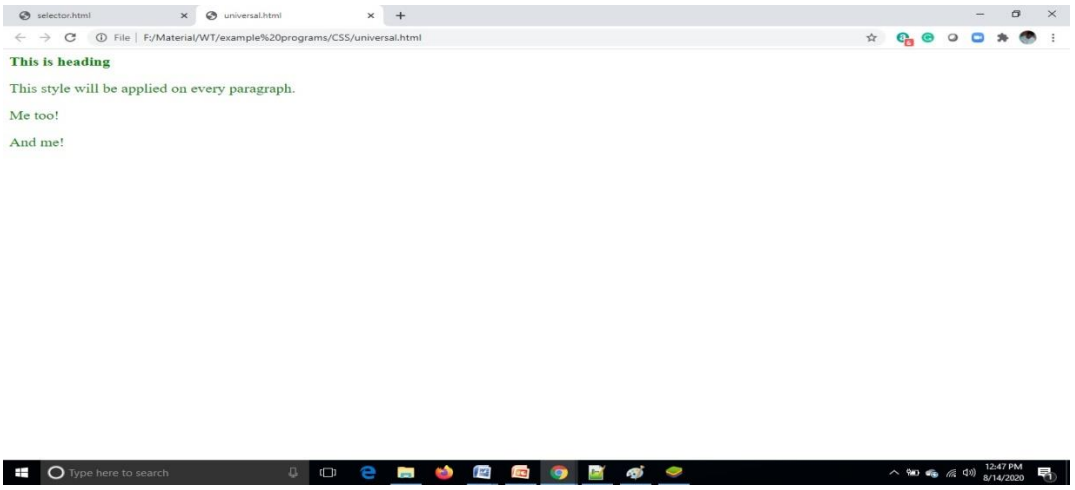
Syntax:

```
<style>
   *
   {
      Style definitions
   }
</style>
<tagname1> Content </tagname1>
<tagname2> Content </tagname2>
```

Example:

```
<html>
   <head>
      <style>
         * {
            color: green;
            font-size: 20px;
         }
      </style>
   </head>
   <body>
      <h2>This is heading</h2>
      <p>This style will be applied on every paragraph.</p>
      <p id="para1">Me too!</p>
      <p>And me!</p>
   </body>
</html>
```

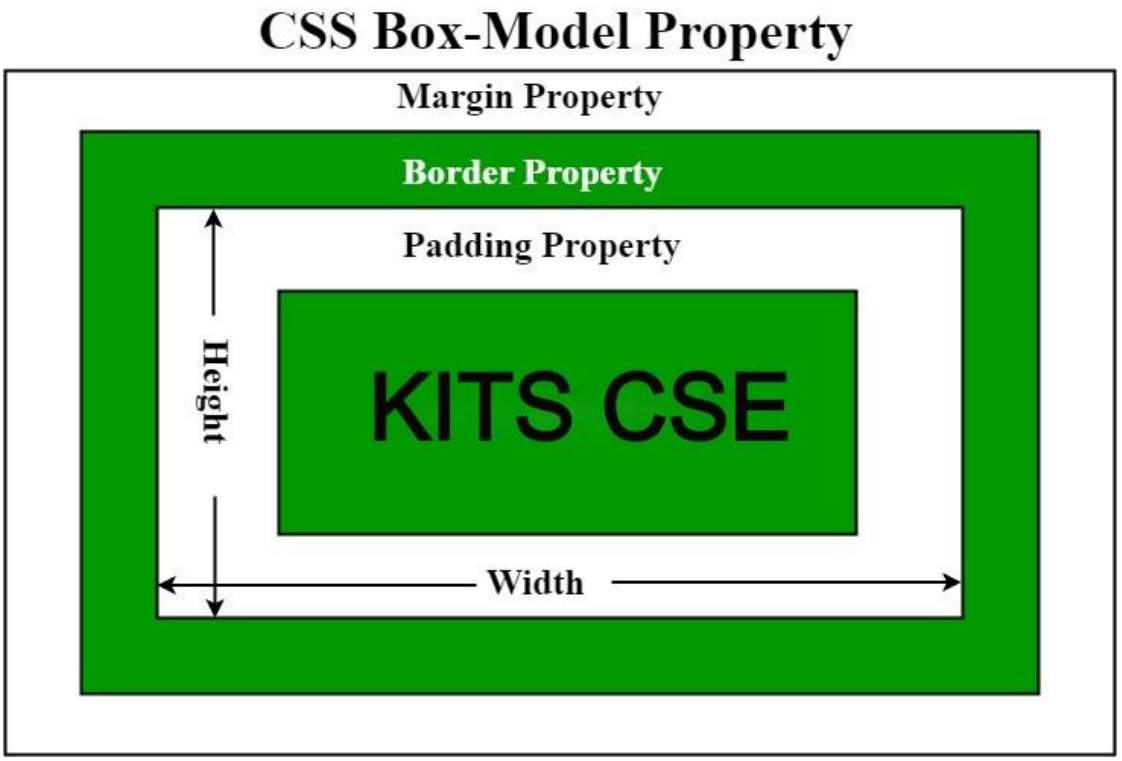
Output:



Box model

CSS box model is a container which contains multiple. It is used to create the design and layout of web pages. It can be used as a toolkit for customizing the layout of different elements. The web browser renders every element as a rectangular box according to the CSS box model. Box-Model has multiple properties in CSS.

1. Borders
2. margins
3. padding
4. Content



Margin

Margin is a CSS property that defines the space of outside of an element to its next outside element. Margin affects elements that both have or do not have borders. If an element has a border, margin defines the space from this border to the next outer element. If an element does not have a border, then margin defines the space from the element content to the next outer element.

Border Area:

It is the area between the box's padding and margin. Its dimensions are given by the width and height of border.

Padding

Padding is a CSS property that defines the space between an element content and its border (if it has a border). If an element has a border around it, padding will give space from that border to the element content which appears in that border. If an element does not have a border around it, then adding padding has no effect at all on that element, because there is no border to give space from.

Difference Between Padding and Margin

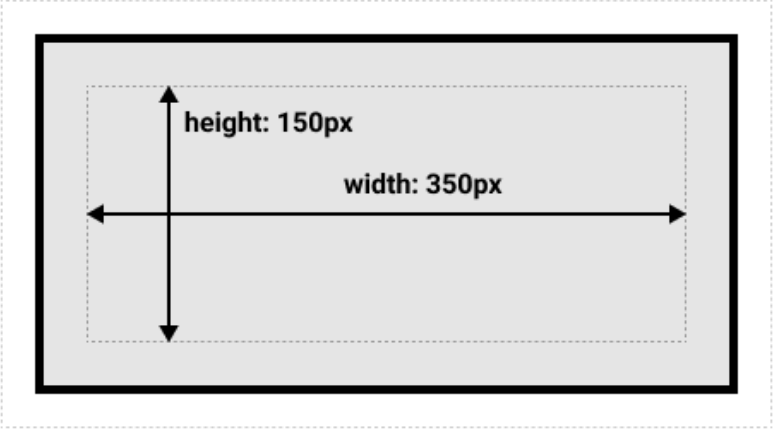
So the difference between margin and padding is that while padding deals with the inner space, margin deals with the outer space to the next outer element.

Content Area:

This area consists of content like text, image, or other media content. It is bounded by the content edge and its dimensions are given by content box width and height.

Example:

```
.box    {
            width: 350px;
            height: 150px;
            margin: 10px;
            padding: 25px;
            border: 5px solid black;
        }
```



The space taken up by our box using the standard box model will actually be 410px (350 + 25 + 25 + 5 + 5), and the height 210px (150 + 25 + 25 + 5 + 5), as the padding and border are added to the width used for the content box.

Conflict Resolution:

In some times to apply one or more declarations to elements then it causes conflicts. The browser faces the problem which type of dimension is applied to the element. CSS conflict resolution done by following two ways those are

1. The direct control of the author of the style sheet who, in turn, can change his/her style rules to resolve these conflicts.
2. The browser resolves the conflicts using the rules for cascading and specificity

Specificity:

When two different selectors are used but both select the same element, the precedence is decided by specificity rules.

The specificity to determines which style declarations are ultimately applied to an element.In that the universal selector (*) has low specificity, while ID selectors are highly specific!

Specificity Hierarchy

Every selector has its place in the specificity hierarchy. There are four categories which define the specificity level of a selector:

1. **Inline styles** - An inline style is attached directly to the element to be styled.
   Example: <h1 style="color: #ffffff;">.
2. **IDs** - An ID is a unique identifier for the page elements, such as #navbar.
3. **Classes, attributes and pseudo-classes** - This category includes .classes, [attributes] and pseudo-classes such as :hover, :focus etc.
4. **Elements and pseudo-elements** - This category includes element names and pseudo-elements, such as h1, div, :before and :after.

Specificity Rules

1. **Equal specificity: the latest rule counts** - If the same rule is written twice into the external style sheet, then the lower rule in the style sheet is closer to the element to be styled, and therefore will be applied:

2. **ID selectors have a higher specificity than attribute selectors**

3. **Contextual selectors are more specific than a single element selector -** The embedded style sheet is closer to the element to be styled.

4. **A class selector beats any number of element selectors** - a class selector

The specificity of an element as a 4-digit number, and the greater number wins. The number is found in the following way:-

1. The thousands digit denotes the presence of inline styles.
2. The hundreds digit denotes the number of IDs specified in the declaration.
3. The tens digit denotes the number of classes specified in the declaration.
4. The one's digit denotes the number of HTML tags specified in the declaration.

Example:

```
<head>
        <style>
                .parent{
                                color:blue;
                                background-color:green;
                                width: 500px;
                                height:500px;
                        }
                .box4.box{
                                color:yellow;
                                font-size:1em;
                         }
                #b{
                        color:pink;
                  }
                div > .box{
                                font-size:3em;
                                background-color:red;
                                border: 1px solid black;
                                margin:10px;
                        }
        </style>
</head>
<body>
        <div class="parent">
                <div class="box" style="color:purple">Iam a box</div>
```

```
        <div class="box" id="a">Iam a box</div>
        <div class="box">Iam a box</div>
        <div class="box box4" id="b">Iam a box</div>
    </div>
</body>
```

Calculation Process:

1. ** Iam a box **

There are two selectors for above line. One is using “div >.box”, and the other is the inline style. Let's compare their scores:

Using the rules above, the selector using inline styles gets a score of 1000 because it is relevant to inner style sheet.

selector using “div > .box” gets a score of 0011.

So, selector using inline style wins (since 1000>11), and the conflicting style, i.e., font-colour, should be purple. The other styles should simply merge.

2. ** Iam a box **

Again there are two selectors here, namely “#b” and “div > .box”.

“#b” gets a score of 0100 because it is **“id attribute**” , and “div > .box” has score of 0011.

Again, as 100>11, the ID selector wins, but here there are no conflicting declarations here.

3. ** Iam a box **

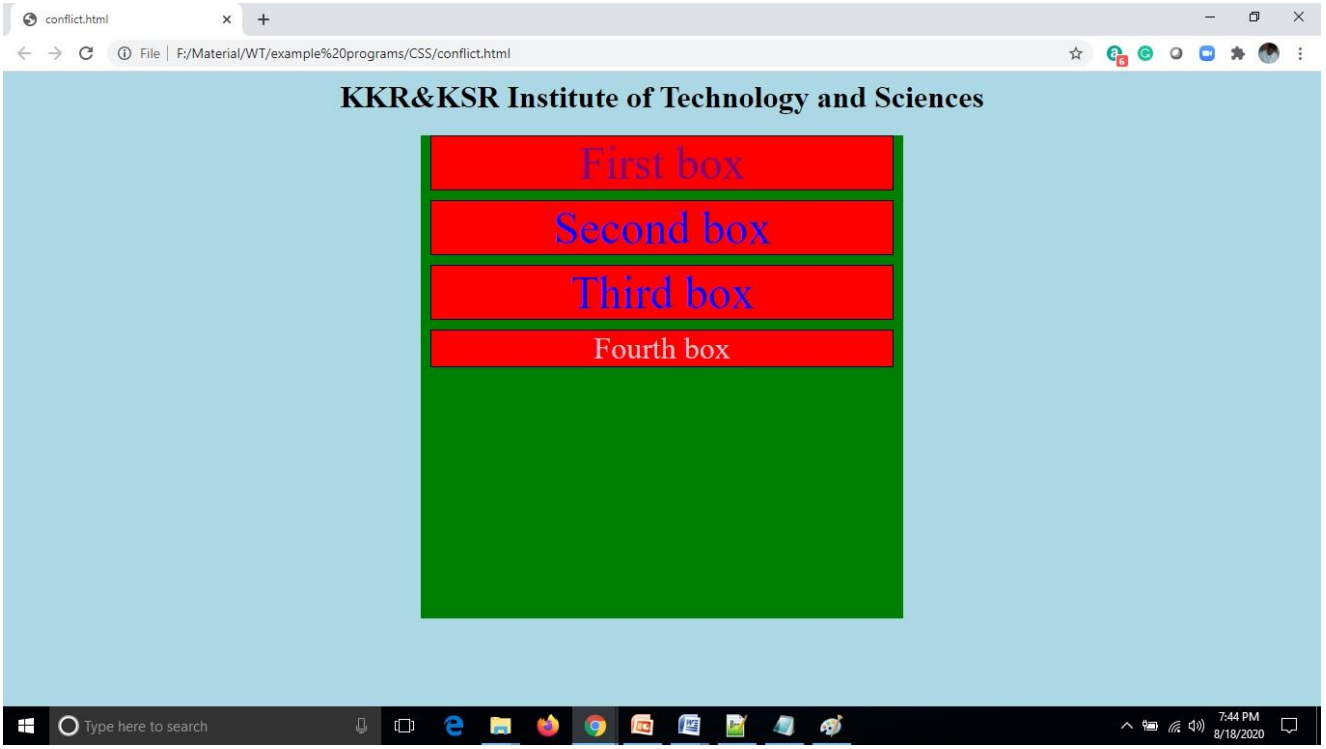
There is only one selector “div > .box”.

4. ** Iam a box **

The “div > .box” is common for this as well. The other one has two classes, and therefore has a score of 0020.

Again, as 20>11, “.box4.box” wins.

Output:


conflict.html
File | F:/Material/WT/example%20programs/CSS/conflict.html
KKR&KSR Institute of Technology and Sciences
First box
Second box
Third box
Fourth box
Type here to search
7:44 PM
8/18/2020