



WEB PROGRAMMING (203105353)

Prof. Tejal Patel , Assistant Professor,
DEPARTMENT OF INFORMATION TECHNOLOGY



CHAPTER-1

HTML and HTML5



Introduction

HTML is a language for describing web pages.

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





HTML Tags

HTML markup tags are usually called HTML tags

- HTML tags are keywords surrounded by **angle brackets** like `<html>`
- HTML tags normally **come in pairs** like `` and ``
- The first tag in a pair is the **start tag**, the second tag is the **end tag**
- Start and end tags are also called **opening tags** and **closing tags**





How to create web page?

Step 1: Open Notepad

Click **Start** (bottom left on your screen). Click **All Programs**. Click **Accessories**. Click **Notepad**.

Step 2: Write HTML Code

A screenshot of a Notepad window titled "Demo - Notepad". The window has a menu bar with "File", "Edit", "Format", "View", and "Help". The text area contains the following HTML code:

```
<html>
<head>
  <title>Parul University</title>
</head>
<body>
  <h1>INFORMATION TECHNOLOGY</h1>
  <p>PARUL INSTITUTE OF ENGINEERING & TECHNOLOGY</p>
</body>
</html>
```





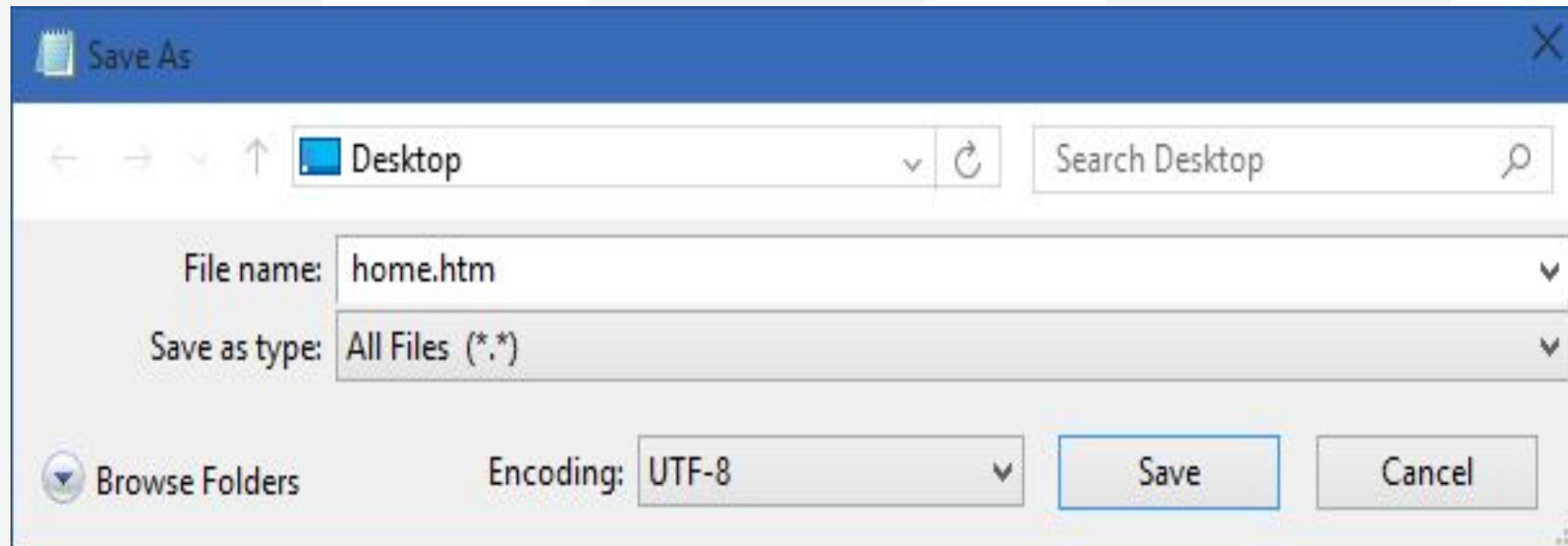
How to create web page?(Contd..)

Step 3: Save the HTML Page

Save the file on your computer.

Select **File > Save as** in the Notepad menu.

Name the file "home.htm" or any other name ending with htm.

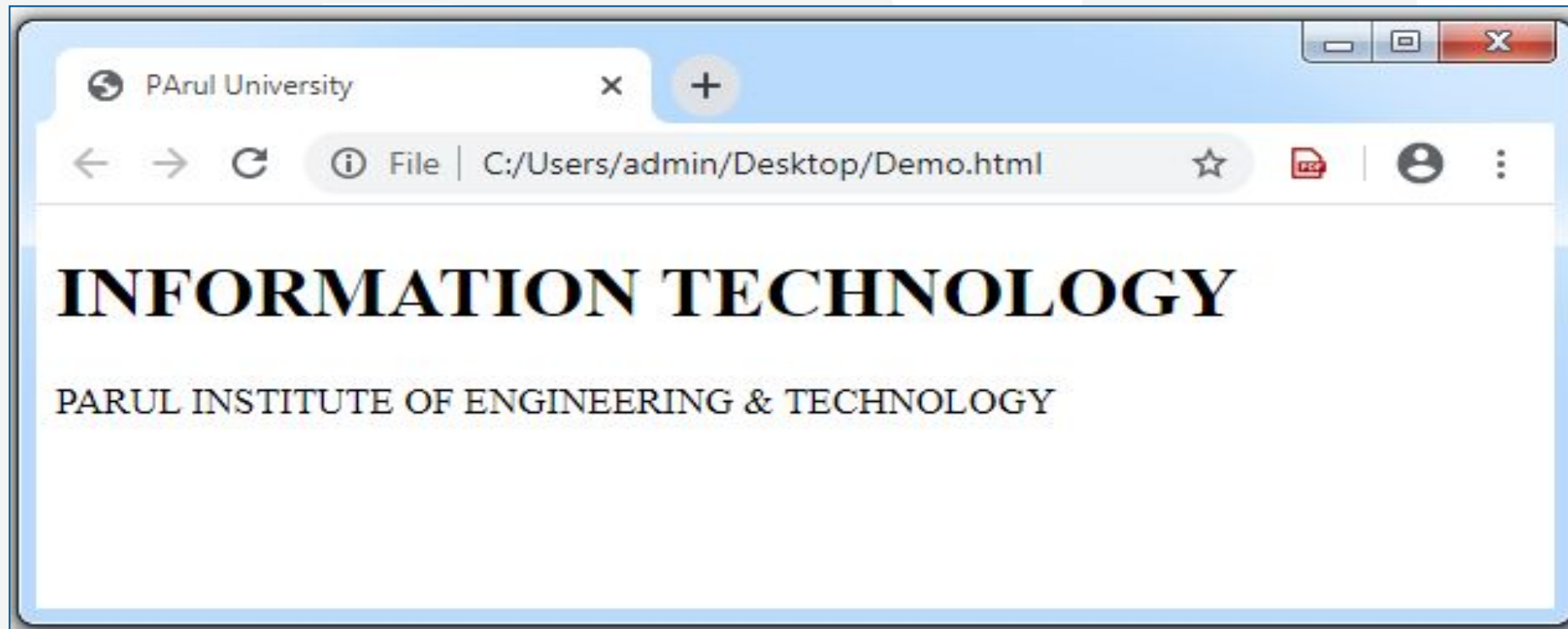




How to create web page?(Contd..)

Step 4: View HTML Page in any Browser

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



Text-HTML Formatting Elements

Tag	Format
<code><i>Parul University</i></code>	Italic text
<code>..< code=""><></code>	Bold text
<code><u>..<u>< code=""></u><></code>	Underline
<code>..< code=""><></code>	important text
<code><small>..<small>< code=""></small><></code>	smaller text
<code><center>..<center>< code=""></center><></code>	Text in center
<code><sub>..<sub>< code=""></sub><></code>	subscripted text.
<code><sup>..<sup>< code=""></sup><></code>	superscripted text
<code><ins>..<ins>< code=""></ins><></code>	inserted text
<code>..< code=""><></code>	deleted text
<code><mark>..<mark>< code=""></mark><></code>	marked/highlighted text
<code>..< code=""><></code>	emphasized text





Text-HTML Formatting Elements-Example

```
<html>
<head> <title>Welcome</title> </head>
<body>
  <p style="color:red">Parul Institute of Engineering &Technology</font></p>
  <p><i>Parul Institute of Engineering &Technology</i></p>
  <p><b>Parul Institute of Engineering &Technology</b></p>
  <p><u>Parul Institute of Engineering &Technology</u></p>
  <p><strong>Parul Institute of Engineering &Technology</strong></p>
  <p><small>Parul Institute of Engineering &Technology</small></p>
  <p>College<sub>Parul Institute of Engineering & Technology</sub>Name</p>
  <p>College<sup>Parul Institute of Engineering &Technology</sup>Name</p>
  <p>Parul Institute of Engineering &Technology,<ins>Waghodia</ins></p>
  <p>Parul Institute of <del>Engineering</del> Technology</p>
  <p><em>Parul Institute of Engineering & Technology</em></p>
</body>
</html>
```





Text-HTML Formatting Elements-Example

OUTPUT:





Hyperlink

- A hyperlink (or link) is a word, group of words, or image that you can click on to jump to a new document or a new section within the current document.
- When you move the cursor over a link in a Web page, the arrow will turn into a little hand.
- Links are specified in HTML using the `<a>` tag.
- The `<a>` tag can be used in two ways:
 1. To create a link to another document, by using the href attribute
 2. To create a bookmark inside a document, by using the name attribute





HTML Link Syntax

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

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

The href attribute specifies the destination of a link.

Example:

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

which will display like this: [Visit W3Schools](http://www.w3schools.com/)

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

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





HTML The Tag and the Src Attribute

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





HTML The Tag and the Src Attribute

- The URL points to the location where the image is stored. An image named "boat.gif", located in the "images" directory on "www.w3schools.com" has the URL: <http://www.w3schools.com/images/boat.gif>.
- The browser displays the image where the 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 The Alt Attribute

- The required alt attribute specifies an alternate text for an image, if the image cannot be displayed.
- The value of the alt attribute is an author-defined text:
``
- The alt attribute provides alternative information for an image if a user for some reason cannot view it (because of slow connection, an error in the src attribute, or if the user uses a screen reader).





HTML Tables

- Tables are defined with the `<table>` tag.
- A table is divided into rows (with the `<tr>` tag), and each row is divided into data cells (with the `<td>` tag).
- `td` stands for "table data," and holds the content of a data cell. A `<td>` tag can contain text, links, images, lists, forms, other tables, etc.





Table Example

Tables are defined with the <table> tag.
Table Example

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





HTML Tables

- How the HTML code above looks in a browser:

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





HTML Tables and the Border Attribute

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

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





HTML Table Headers

- Header information in a table are defined with the <th> tag.
- The text in a th element will be bold and centered.

```
<table border="1">
<tr>
<th>Header 1</th>
<th>Header 2</th>
</tr>
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td>row 2, cell 2</td>
</tr>
</table>
```



HTML Table Headers

How the HTML code above looks in a browser:

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





HTML Table Tags

Tag	Description
<u><table></u>	Defines a table
<u><th></u>	Defines a table header
<u><tr></u>	Defines a table row
<u><td></u>	Defines a table cell
<u><caption></u>	Defines a table caption
<u><colgroup></u>	Defines a group of columns in a table, for formatting
<u><col /></u>	Defines attribute values for one or more columns in a table
<u><thead></u>	Groups the header content in a table
<u><tbody></u>	Groups the body content in a table
<u><tfoot></u>	Groups the footer content in a table





HTML Lists

An ordered list:

- 1.The first list item
- 2.The second list item
- 3.The third list item

An unordered list:

- List item
- List item
- List item



HTML Unordered Lists

- An unordered list starts with the `` tag. Each list item starts with the `` tag.
- The list items are marked with bullets (typically small black circles).

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

- How the HTML code above looks in a browser:
 - Coffee
 - Milk





HTML Ordered Lists

- An ordered list starts with the `` tag. Each list item starts with the `` tag.
- The list items are marked with numbers.

```
<ol>
```

```
<li>Coffee</li>
```

```
<li>Milk</li>
```

```
</ol>
```

- How the HTML code above looks in a browser:
 1. Coffee
 2. Milk





HTML Definition Lists

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

```
<dl>
```

```
<dt>Coffee</dt>
```

```
<dd>- black hot drink</dd>
```

```
<dt>Milk</dt>
```

```
<dd>- white cold drink</dd>
```

```
</dl>
```





HTML Definition Lists

- How the HTML code above looks in a browser:

Coffee

- black hot drink

Milk

- white cold drink





Forms

HTML forms are used to collect user data/input.

HTML forms contain **form elements**.

It is defines as a:

```
<form>
```

...

form elements

...

```
</form>
```

Form elements are different types of input elements, checkboxes, radio buttons, submit buttons, and more.





Forms(Contd..)

A.<input> Element:

The <input> element is used for collecting information from the user. It has many variations, depending on the **type attribute**.

Type	Description
text	Defines normal text input(accept char. & no. into text box)
Password	Its similar to above text box but anything that is typed cannot be seen.
radio	Defines radio button input (for selecting one of many choices)
submit	Defines a submit button (for submitting the form)
checkbox	This gives a check box that can be toggled between checked & unchecked
Button	This makes a button available





Forms(Contd..)

1.Text Input: It Define a one-line input.(default width of a text field is 20 characters)

```
<html>
<body>
<h1>PARUL UNIVERSITY</h1>
<form>First name:<br>
<input type="text" name="firstname">
<br>
Last name:<br>
<input type="text" name="lastname">
</form></body>
</html>
```

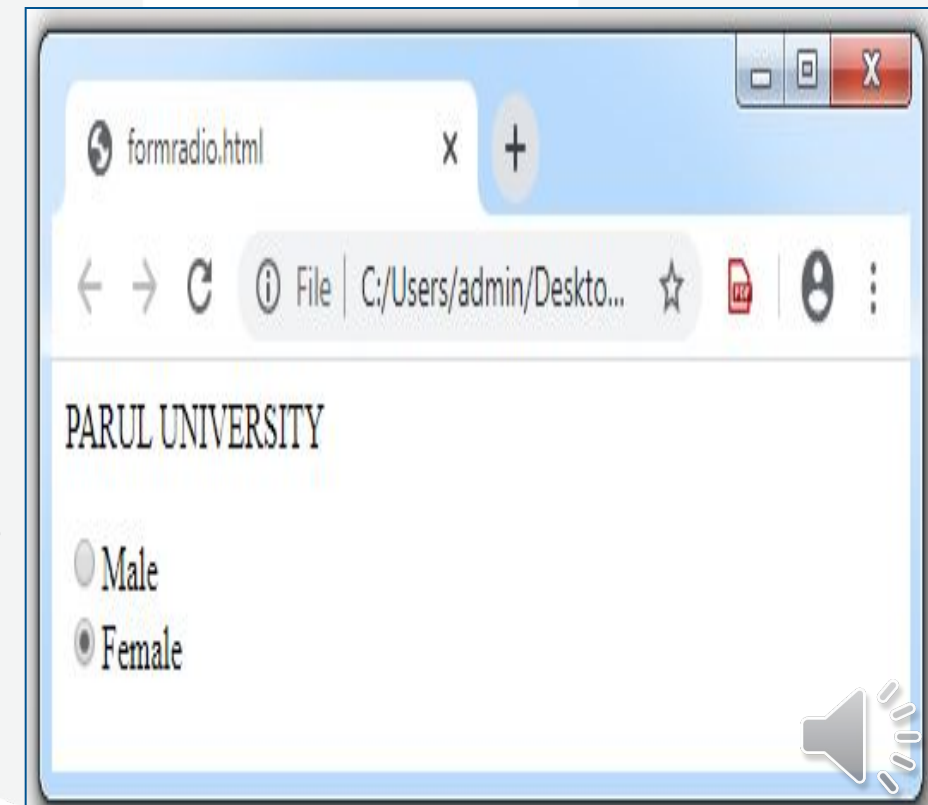
A screenshot of a web browser window titled 'formtext.html'. The browser's address bar shows the file path 'C:/Users/admin/Desktop...'. The page content displays 'PARUL UNIVERSITY' in a large, bold, serif font. Below the title, there are two text input fields. The first field is preceded by the label 'First name:' and the second field is preceded by the label 'Last name:'. The browser's interface includes standard navigation buttons (back, forward, refresh) and window controls (minimize, maximize, close).



Forms(Contd..)

2. Radio Button Input :It is used to select one of many given choices. you can select only one at same time.

```
<html>
<body>
  <p>PARUL UNIVERSITY</p>
  <form>
    <input type="radio" name="sex" value="male" >Male
    <br>
    <input type="radio" name="sex" value="female">Female
  </form>
</body>
</html>
```





Forms(Contd..)

3. The Submit Button: It is used for submitting a form data to a form-handler.

```
<html><body>
<form action="action_page.php">
<h1>PARUL UNIVERSITY</h1>
First name:<br>
<input type="text" name="firstname" value="Mickey"><br>
Last name:<br>
<input type="text" name="lastname" value="Mouse"><br><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

A screenshot of a web browser window showing the rendered HTML form. The browser's address bar shows 'formsubmit.html' and the file path 'C:/Users/admin/Desktop...'. The form content includes the heading 'PARUL UNIVERSITY', two text input fields labeled 'First name:' and 'Last name:', and a 'Submit' button. The first name field contains the text 'PIET' and the last name field contains 'IT'. A speaker icon is visible in the bottom right corner of the browser window.



Forms(Contd..)

4. **checkbox** : It is used for select ZERO or MORE options of a limited number of choices.

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

```
<form>
```

```
<h1> PARUL UNIVERSITY </h1>
```

```
<input type="checkbox" name="vehicle" value="Pen">I have a Pen <br>
```

```
<input type="checkbox" name="vehicle" value="Pencil">I have a Pencil<br>
```

```
<input type="checkbox" name="vehicle" value="Eraser">I have a Eraser
```

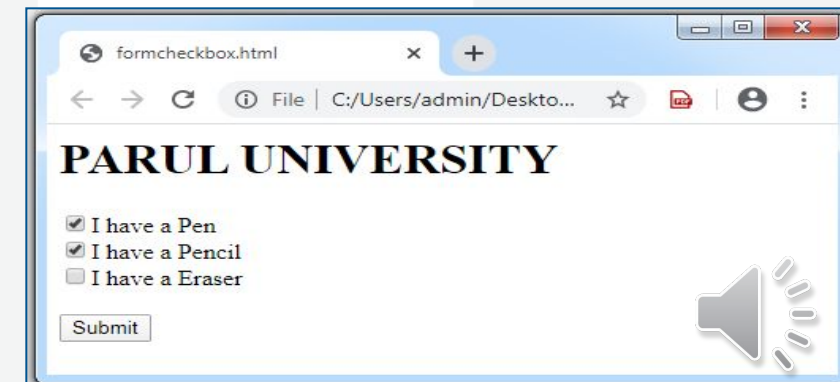
```
<br><br>
```

```
<input type="submit">
```

```
</form>
```

```
</body>
```

```
</html>
```

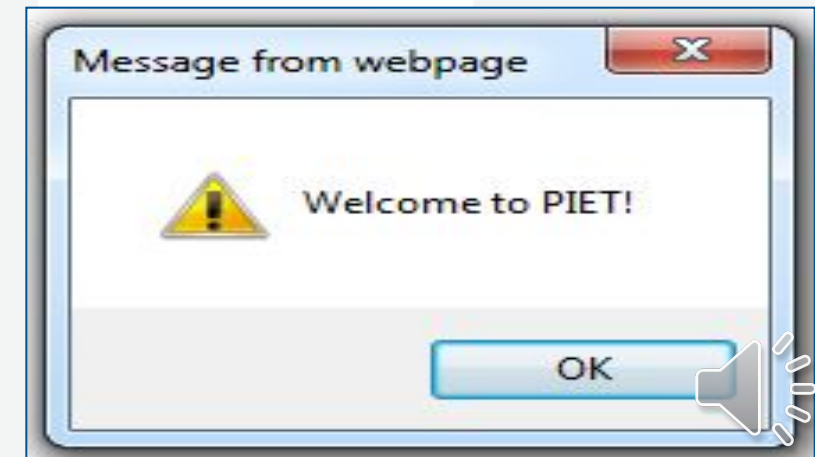
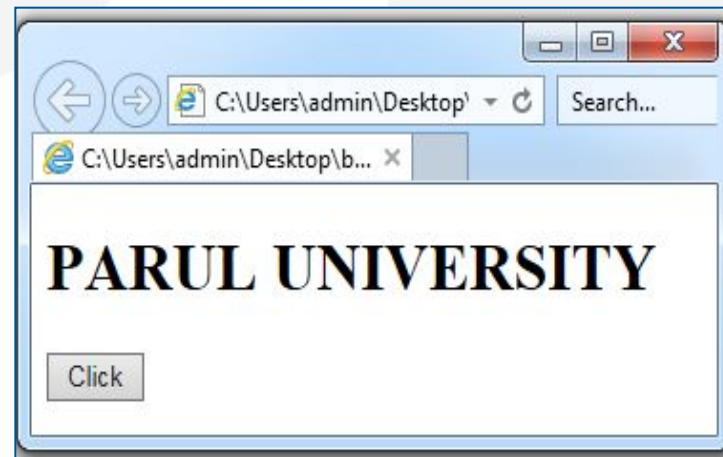




Forms(Contd..)

B. Button: The <button> element represents a clickable button. It is used to submit forms.

```
<html>
<body>
<form>
<h1>PARUL UNIVERSITY</h1>
<button type="button" onclick="alert('Welcome to PIET!')">Click</button>
</form>
</body>
</html>
```

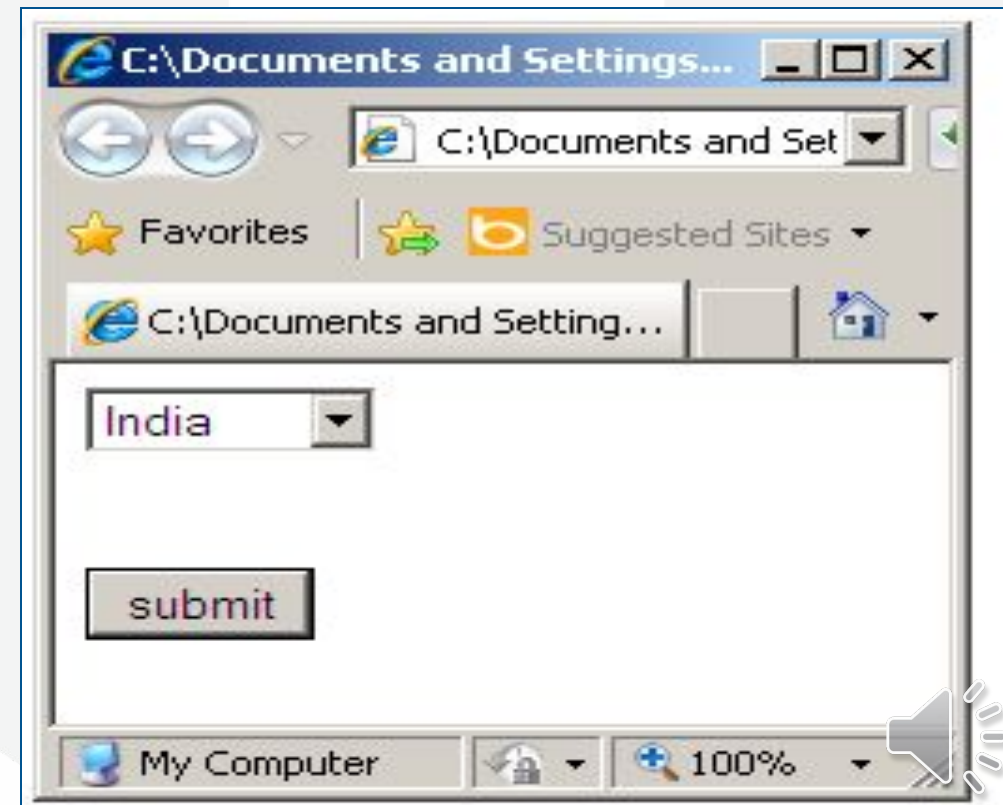




Forms(Contd..)

C.<select>Element(Drop-Down List): HTML <select> tag is used to create drop down list of options

```
<html><body>  
<form>  
<select name="country">  
<option value="India">India</option>  
<option value="canada">canada</option>  
<option value="Landon">Landon</option>  
<option value="Australia">Australia</option>  
</select>  
<br><br> <br>  
<input type="submit" value="submit">  
</form></body>  
</html>
```





Forms(Contd..)

D.<textarea>Element: It is defines a multi-line input field.

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

```
<form>
```

```
<textarea name="message" rows="25" cols="40">
```

Web Designing and development is used to create dynamic website.

```
</textarea>
```

```
<br>
```

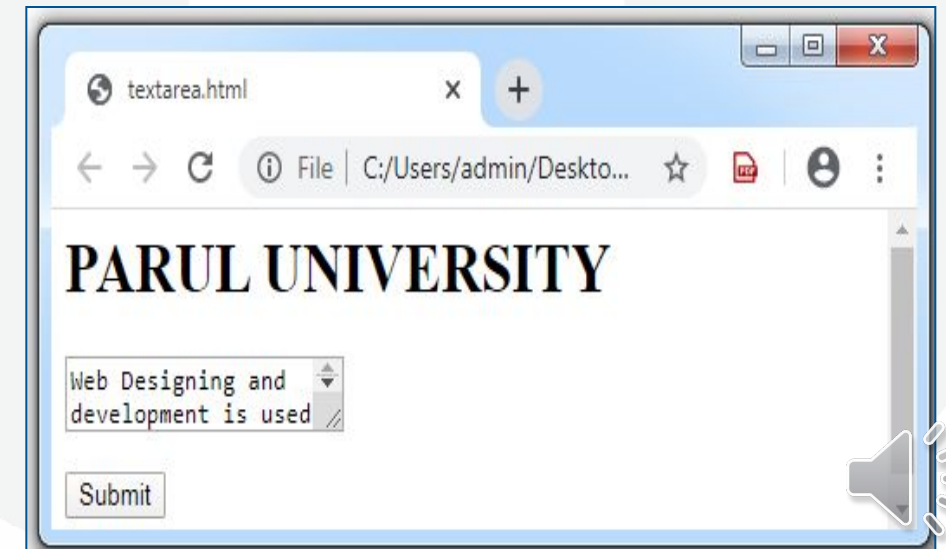
```
<br>
```

```
<input type="submit">
```

```
</form>
```

```
</body>
```

```
</html>
```





HTML Frames

- With frames, you can display more than one HTML document in the same browser window. Each HTML document is called a frame, and each frame is independent of the others.
- The disadvantages of using frames are:
 1. The web developer must keep track of more HTML documents
 2. It is difficult to print the entire page





The HTML frameset Element

- The frameset element holds one or more frame elements. Each frame element can hold a separate document.
- The frameset element states HOW MANY columns or rows there will be in the frameset, and HOW MUCH percentage/pixels of space will occupy each of them.





The HTML frame Element

- The <frame> tag defines one particular window (frame) within a frameset.
- In the example below we have a frameset with two columns.
- The first column is set to 25% of the width of the browser window. The second column is set to 75% of the width of the browser window. The document "frame_a.htm" is put into the first column, and the document "frame_b.htm" is put into the second column:

```
<frameset cols="25%,75%">  
  <frame src="frame_a.htm" />  
  <frame src="frame_b.htm" />  
</frameset>
```





HTML Colors

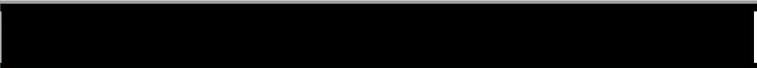








Colors are displayed combining RED, GREEN, and BLUE light

•Color Values:

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



Color Values

Color	Color HEX	Color RGB
	#000000	rgb(0,0,0)
	#FF0000	rgb(255,0,0)
	#00FF00	rgb(0,255,0)
	#0000FF	rgb(0,0,255)
	#FFFF00	rgb(255,255,0)
	#00FFFF	rgb(0,255,255)
	#FF00FF	rgb(255,0,255)
	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)





HTML5

- HTML5 is the latest and most enhanced version of HTML
- HTML5 is a standard for structuring and presenting content on the World Wide Web
- HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG)
- The new standard incorporates features like video playback and drag-and-drop that have been previously dependent on third-party browser plug-ins such as Adobe Flash, Microsoft Silverlight, and Google Gears



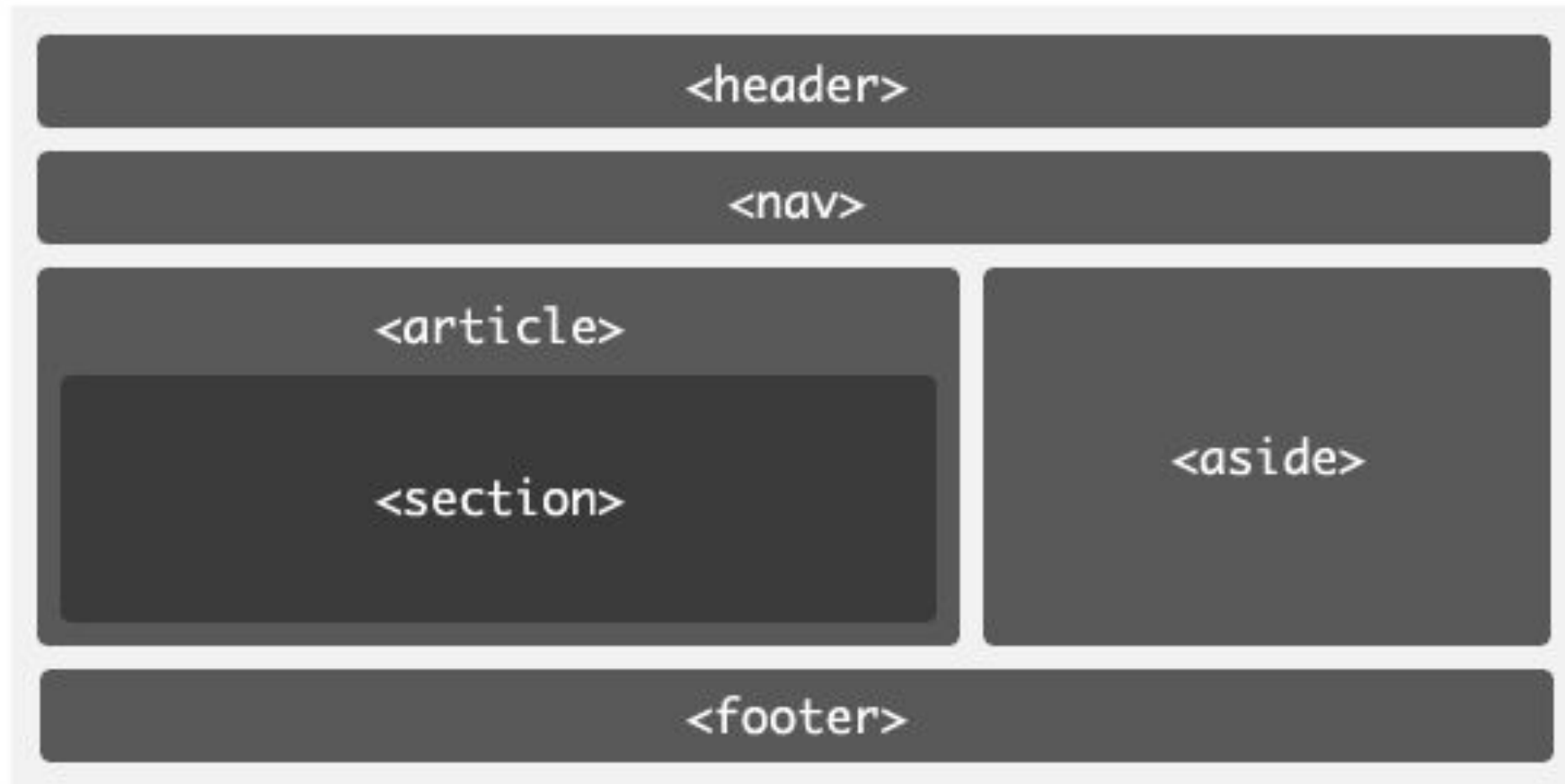


HTML4 V/s HTML5

HTML4	HTML5
1. DOCTYPE declaration is not easy.	1. DOCTYPE declaration is very simple.
2. Character encoding declaration is :<meta http-equiv="Content-Type" content="text/html; charset=utf-8">	2. Character encoding declaration is <meta charset="UTF-8">
3. It doesn't support Multimedia Tag.	3.It support Multimedia Tag. e.g<audio>, <video>
4. it was difficult task to get the geographical locations of the visitors visiting the site.	4.HTML5 is extremely easy to get the user location.
5. HTML4 is compatible with almost all web-browsers	5. HTML4 being newer version is not compatible with all the browsers.
6. HTML4 contained an <applet> tag	6. a new <object> tag has been introduced in HTML5.
7. It doesn't support graphics Tag.	7. It support graphics Tag. e.g. <svg>,<canvas>
8.It isn't contain calendar ,date ,time form controls.	8. It support New form control attributes (like number, date, time, calendar, and range)



HTML5 Structure





Example:

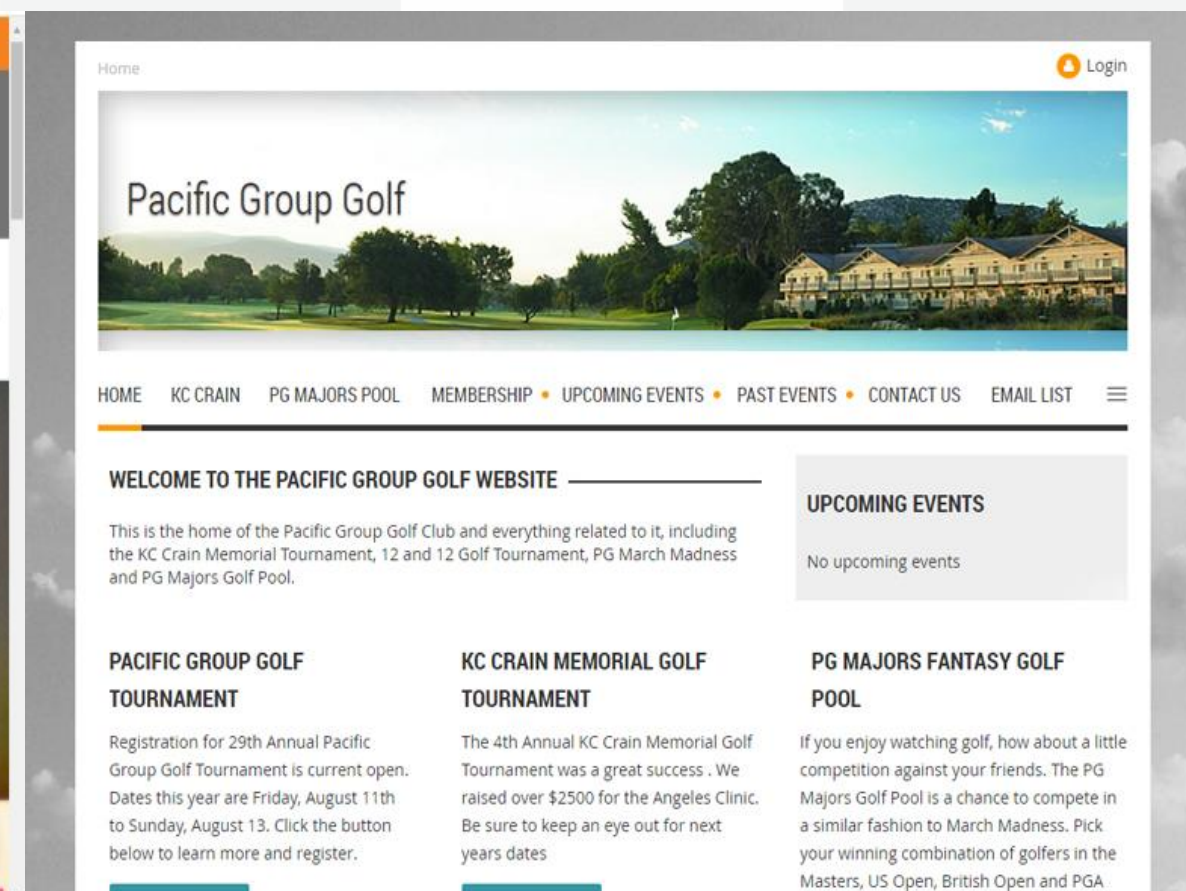


Figure 2.41: Website Layout Example

HTML5 Header

- The <header> tag in HTML is used to define the header for a document or a section
- The header tag contains information related to the title and heading of the related content
- A <header> element typically contains:
 1. one or more heading elements (<h1> - <h6>)
 2. logo or icon
 3. authorship information





HTML5 Header

```
<!DOCTYPE html>
<html>
  <head>
    <title>Header Tag</title>
  </head>
  <body>

    <header>
      <h1>This is the heading.</h1>
      <h4>This is the sub-heading.</h4>
      <p>This is the metadata.</p>
    </header>

  </body>
</html>
```

This is the heading.

This is the sub-heading.

This is the metadata.





HTML5 Footer

- The <footer> element defines a footer for a document or section.
- A <footer> element typically contains:
 1. authorship information
 2. copyright information
 3. contact information
 4. sitemap
 5. back to top links
 6. related documents
- You can have several <footer> elements in one document.





HTML5 Footer

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

    <footer>
      <p>Author: Hege Refsnes</p>
      <p><a href="mailto:hege@example.com">hege@example.com</a></p>
    </footer>

  </body>
</html>
```

Author: Hege Refsnes

hege@example.com





HTML5 <nav>

- The <nav> tag defines a set of navigation links.
- NOT all links of a document should be inside a <nav> element.
- The <nav> element is intended only for major block of navigation links.
- Browsers, such as screen readers for disabled users, can use this element to determine whether to omit the initial rendering of this content.





HTML5 <nav>

<nav>

```
<a href="/html/">HTML</a> |  
<a href="/css/">CSS</a> |  
<a href="/js/">JavaScript</a> |  
<a href="/python/">Python</a>
```

</nav>

The nav element

The nav element defines a set of navigation links:

[HTML](#) [CSS](#) | [JavaScript](#) | [Python](#)



HTML5 article

- The **<article>** tag is one of the new sectioning element in HTML5
- The HTML **<article>** tag is used to represent an article
- More specifically, the content within the **<article>** tag is independent from the other content of the site (even though it can be related)
- **Potential source for Article Element are :**
 1. A blog entry
 2. A magazine/newspaper article
 3. A forum post
 4. A user submitted comment





HTML5 article

```
<!DOCTYPE html>
<html>
<body>
<h1>The article element</h1>

<article>
  <h2>Google Chrome</h2>
  <p>Google Chrome is a web browser developed by Google</p>
</article>
<article>
  <h2>Mozilla Firefox</h2>
  <p>Mozilla Firefox is an open-source web browser developed by Mozilla</p>
</article>

</body>
</html>
```

The article element

Google Chrome

Google Chrome is a web browser developed by Google

Mozilla Firefox

Mozilla Firefox is an open-source web browser developed by Mozilla





HTML5 section

- The `<section>` element defines a section in a document.
- According to W3C's HTML documentation: "A section is a thematic grouping of content, typically with a heading."
- A web page could normally be split into sections for introduction, content, and contact information.





HTML5 section

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<section>
```

```
<h1>WWF</h1>
```

<p>The World Wide Fund for Nature (WWF) is an international organization working on issues regarding the conservation, research and restoration of the environment, formerly named the World Wildlife Fund. WWF was founded in 1961.</p>

```
</section>
```

```
<section>
```

```
<h1>WWF's Panda symbol</h1>
```

<p>The Panda has become the symbol of WWF. The well-known panda logo of WWF originated from a panda named Chi Chi that was transferred from the Beijing Zoo to the London Zoo in the same year of the establishment of WWF.</p>

```
</section>
```

```
</body>
```





HTML5 section

WWF

The World Wide Fund for Nature (WWF) is an international organization working on issues regarding the conservation, research and restoration of the environment, formerly named the World Wildlife Fund. WWF was founded in 1961.

WWF's Panda symbol

The Panda has become the symbol of WWF. The well-known panda logo of WWF originated from a panda named Chi Chi that was transferred from the Beijing Zoo to the London Zoo in the same year of the establishment of WWF.



Audio & Video Tag

HTML 5 support multimedia elements : `<audio>` and `<video>`.

We can add sound, music, videos, movies and animations on the web/website.

Multimedia elements Like audio and video are stored in media files.

It is used different extension like: .mpg, .mp3, .mp4, .ogg, .rm, .ram, .webm, .wav etc

HTML5 support Only MP3, Wav, ogg format for audio.

HTML5 support Only MP4, WebM, ogg format for video





Audio & Video Tag(Contd..)

Audio: Use src attribute to identify the media source and add a controls attribute so the user can play and pause the media.

```
<html>
<body>
  <p>PARUL UNIVERSITY </p>
  <audio width="100%" height="100%" controls="controls">
    <source src="horse.mp3" type=audio/mp3>
    <source src="audio.ogg" type="audio/ogg">
  </audio>
</body>
</html>
```





Audio & Video Tag(Contd..)

Video:

```
<html>
  <body>
    <video width="300" height="200" controls>
      <source src="movie.mp4" type="video/mp4">
      <source src="movie.ogg" type="video/ogg">
    </video>
  </body>
</html>
```





Audio & Video Tag(Contd..)

Attributes	Description
autoplay	The video/audio will automatically start without stopping to finish loading the data.
controls	User can control audio/video like pause/resume, volume, seeking etc.
autobuffer	The audio/video will automatically begin buffering even don't set to automatically play.
height	It is specifies the height of the audio/video's display area.
src	Use src attribute to identify the media source. This is optional. we can use <source> within the video block .
width	It is specifies the width of the audio/video's display area.(in CSS pixels)
loop	It will allow audio/video automatically seek back to the start after reaching at the end.
poster	This is a URL of an image to display until the user plays or seeks.
preload	It is specifies the audio/video will be loaded at page load, and ready to run. if autoplay is present then ignore this.





Canvas

HTML 5 support graphics elements : `<canvas>` and `<svg>`.

The use of `<canvas>` element is to draw graphics on a web.

The `<canvas>` element is only a container for graphics. You must use JavaScript to actually draw the graphics.

By default, a canvas has no border and no content. A canvas is a rectangular area on an HTML page.

Ex. `<canvas id="parul" width="90" height="80"></canvas>`

Canvas has no. of methods for drawing paths, boxes, circles, text, and adding images.

We can find `<canvas>` element in the DOM using `getElementById()` method like:

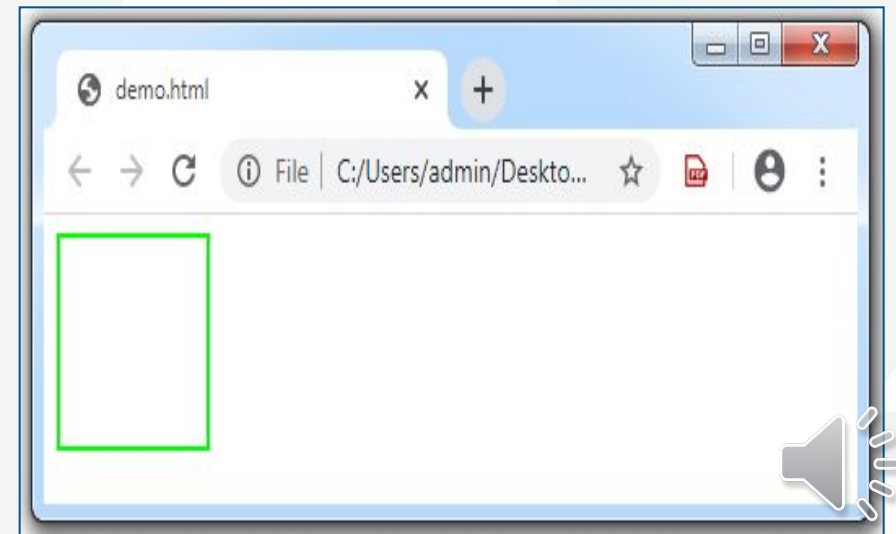
```
var canvas = document.getElementById("mycanvas");
```





Canvas(Contd..)

Example: <!DOCTYPE html>
<html> <head>
 <style>
 #parul
 { border:2px solid green; }
 </style>
</head>
<body>
 <canvas id="parul" width="90" height="90">
</canvas>
</body>
</html>



Geolocation

It is use to share your location with your web sites.

Most of the browsers and mobile devices support Geolocation API.

A JavaScript can capture your latitude and longitude and can be sent to backend web server and display your location on a map.

It is work with a new property of the global navigator object.

we can be create Geolocation object as below:

```
var geolocation = navigator.geolocation;
```





Geolocation(Contd..)

Navigator Object Properties:

Property	Description
appName	Return the name of the browser
appVersion	Return the version information of the browser
geolocation	Return a geolocation object that can be used to locate the user's position
appCodeName	Return the code name of the browser
language	Return the language of the browser
platform	Return for which platform the browser is compiled
onLine	Verify whether the browser is online
cookieEnabled	Determines whether cookies are enabled in the browser
product	Return the engine name of the browser





Geolocation(Contd..)

Geolocation Object also provides below Methods:

Methods	Description
watchPosition()	Retrieves periodic updates about the current geographic location of the device.
clearWatch()	Delete an ongoing watchPosition Call.
getCurrentPosition()	Retrieves the current geographic location of the user.



Geolocation(Contd..)

sample code to use any of the above method –

```
function getLoc()  
{  
  var geolocation = navigator.geolocation;  
  geolocation.getCurrentPosition(showLocation, errorHandler);  
}
```

showLocation and errorHandler are callback methods which is used to get actual position and to handle errors if there is any.

The `getCurrentPosition()` is used to return the User's Position.





Geolocation(Contd..)

Location Properties:

The **Position** object which stores the complete location information. It specifies the current geographic location of the device. The following are the properties of the Position object:

Property	Type	Description
coords	objects	Specifies the geolocation of the device.
coords.longitude	Number	Specifies the longitude estimate in decimal degrees. Range is[-180.00, +180.00]
coords.latitude	Number	Specifies the latitude estimate in decimal degrees. Range is[-90.00, +90.00]
coords.altitude	Number	Specifies the altitude estimate in meters. It's Optional.
coords.speed	Number	Specifies the device's current ground speed in meters per second. It's Optional.
coords.heading	Number	Specifies the device's current direction of movement in degrees counting clockwise relative to true north. It's Optional.
coords.accuracy	Number	Specifies the accuracy of the latitude and longitude estimate in meters. It's Optional.
timestamp	date	Specifies the time when the location information was retrieved and position object created. It's Optional.
coords.altitudeAccuracy	Number	Specifies the accuracy of the altitude estimate in meters. It's Optional.



Geolocation(Contd..)

This is a sample code of Position object. Here showLocation method is a callback method

```
function showLocation( position )  
{  
  var lati = position.coords.latitude;  
  var long = position.coords.longitude;  
  ...  
}
```





Geolocation(Contd..)

Handling Error

Geolocation is complicated, so its required to catch any error and handle it.

The geolocations methods `getCurrentPosition()` and `watchPosition()` make use of an error handler callback method which gives **PositionError** object.

There are two Handling Error properties :

Property	Type	Description
Message	String	Contains a human-readable description of the error
Code	Number	Contains a numeric code for error





Geolocation(Contd..)

Below table describe the Possible error code return in the PositionError Object:

Constant	Code	Description
UNKNOWN_ERROR	0	The method failed to retrieve the location of the device due to an unknown error.
PERMISSION_DENIED	1	The method failed to retrieve the location of the device because the application does not have permission to use the location services.
POSITION_UNAVAILABLE	2	The location of the device could not be determined.
TIMEOUT	3	The method was unable to retrieve the location information within the specified maximum timeout interval.





Geolocation(Contd..)

This is a sample code of PositionError object. Here errorHandler method is a callback method :

```
function errorHandler( err )  
{  
    if (err.code == 0)  
    {  
        // Unkonwn_Error  
    }  
}
```





Geolocation(Contd..)

Position Options:

The actual syntax of `getCurrentPosition()` method is:

`getCurrentPosition(callback, errorCallback, options)`

Here 3rd argument is `PositionOptions` object which specifies a set of options for retrieving the geographic location of the device.

Below table describe the option for third argument:

Property	Type	Description
timeout	Number	It is the number of milliseconds your web application is willing to wait for a position.
maximumAge	Number	Specifies the expiry time in milliseconds for caches location information.
enableHighAccuracy	Boolean	Specifies whether the widget wants to receive the most accurate location estimate possible. By default it is false.



Geolocation(Contd..)

This is a sample code which define how to use above mentioned methods –

```
function getLocation()  
{  
  
    var geolocation = navigator.geolocation;  
    geolocation.getCurrentPosition(showLocation,  
  
                                errorHandler, {maximumAge: 75000});  
  
}
```





Geolocation(Contd..)

Example: <html><body>

```
<p>PARUL UNIVERSITY </p>
```

```
<p>PARUL INSTITUTE OF ENGINEERING & TECHNOLOGY</p>
```

```
<button onclick="getLoc()">Get Location</button>
```

```
<p id="parul">
```

```
</p>
```

```
<script>
```

```
var a = document.getElementById("parul");
```

```
function getLoc() {
```

```
    if (navigator.geolocation) {
```

```
        navigator.geolocation.getCurrentPosition(showPos, DisplayError) }
```

```
    else {
```

```
        a.innerHTML = "Opps,Geolocation is not supported."; }}
```

```
function showPos(position)
```

```
{
```

```
    a.innerHTML = "Latitude: " + position.coords.latitude + "<br>Longitude: " + position.coords.longitude;
```

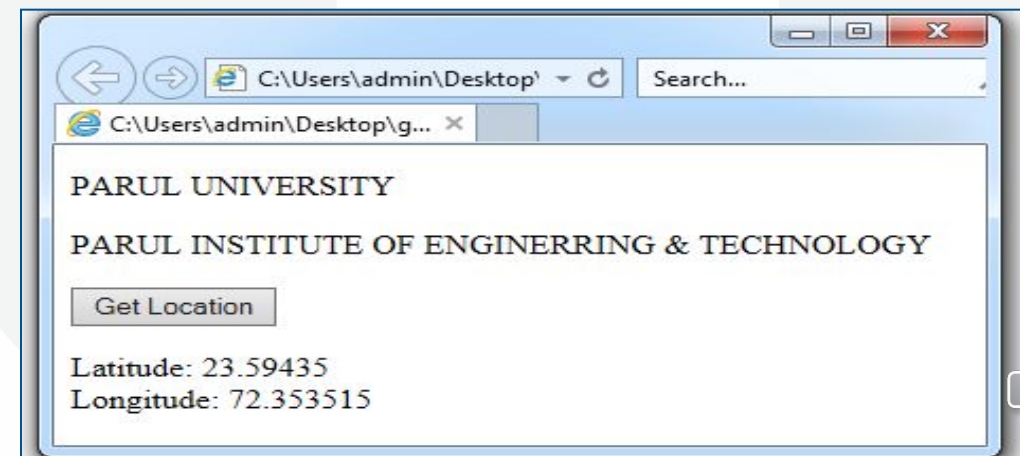
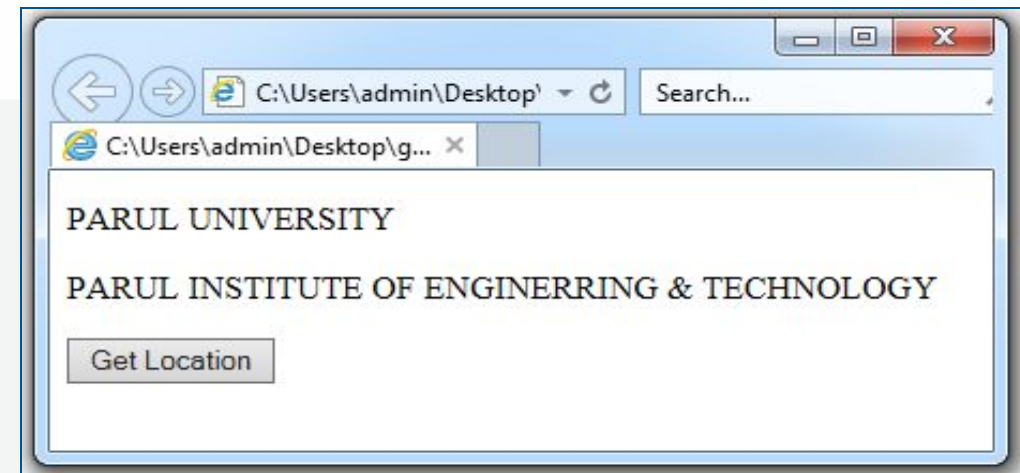
```
}
```





Geolocation(Contd..)

```
function DisplayError(error)
{
    switch(error.code)
    {
        case error.PERMISSION_DENIED:
            a.innerHTML = "denied the request."
            break;
        case error.POSITION_UNAVAILABLE:
            a.innerHTML = "Location information is not available."
            break;
        case error.TIMEOUT:
            a.innerHTML = "get user location timed out."
            break;
        case error.UNKNOWN_ERROR:
            a.innerHTML = "An unknown error occurred."
            break; } }</script></body></html>
```





Drag & Drop

In HTML5, any element can be draggable.

It is when you "grab" an object and drag it to a different location.

Drag and Drop (DnD) allows the user to click and hold the mouse button down over an element, drag it to another location, and release the mouse button to drop the element there.

To make an element draggable, set the draggable attribute to true:

```
<img draggable="true">
```



Drag & Drop (Contd..)

ondragstart & setData(): (What to Drag)
what happen when the element is dragged.

The ondragstart attribute calls a function, drag(event), that specifies what data to be dragged.

The dataTransfer.setData() method sets the data type and the value of the dragged data:

```
function drag(e)
{
    e.dataTransfer.setData("text", e.target.id);
}
```

Here, the data type is "text" and the value is the id of the draggable element ("drag1").





Drag & Drop (Contd..)

ondrop: (Do the Drop)

When the dragged data is dropped, a drop event occurs.

The ondrop attribute calls a function, drop(event):

```
function drop(e)
{
    e.preventDefault();
    var data = e.dataTransfer.getData("text");
    e.target.appendChild(document.getElementById(data));
}
```





Drag & Drop (Contd..)

Example : <html><head><style>

```
#parul {  
  width: 190px;  
  height: 80px;  
  padding: 10px;  
  border: 1px solid #aaaaaa; }  
</style><script>  
function allowDrop(e) {  
  e.preventDefault(); }  
function drag(e) {  
  e.dataTransfer.setData("text", e.target.id); }  
function drop(e) {  
  e.preventDefault();  
  var data = e.dataTransfer.getData("text");  
  e.target.appendChild(document.getElementById(data));  
}</script></head>
```





Drag & Drop (Contd..)

```
<body><p> PARUL LOGO</p>
```

```
<p>Drag the image into the rectangle:</p>
```

```
<div id="parul" ondrop="drop(event)" ondragover="allowDrop(event)"></div><br>
```

```

```

```
</body>
```

```
</html>
```



Drag & Drop (Contd..)

There are no. of Drag & Drop events, which are listed below:

Events	Description
dragenter	Fired when the mouse is first moved over the target element while a drag is occurring.
dragleave	Fired when the mouse leaves an element while a drag is occurring.
dragstart	Fired when the user start dragging of the object.
dragover	Fired as the mouse is moves over an element when a drag is occurring
drag	Fires every time the mouse is moved while the object is being dragged.
dragend	Fires when the user releases the mouse button while dragging an object.
drop	Fired on the element where the drop was occurred at the end of the drag operation

