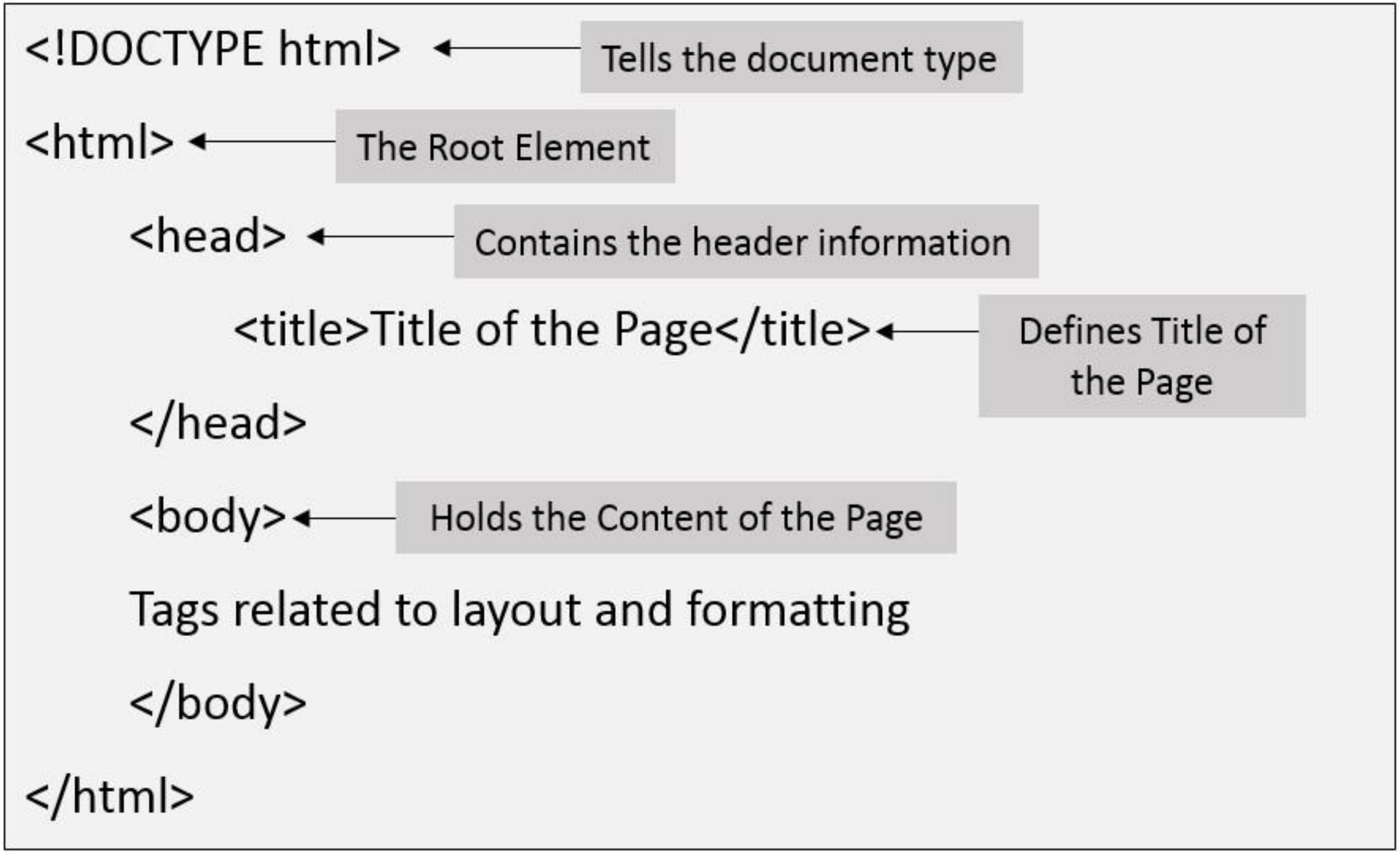
# HTML

# What is HTML?

- ✓HTML stands for Hyper Text Markup Language
- ✓HTML is the standard markup language for creating Web pages.
- ✓H → Hyper →reference/link
- ✓T →Text → Data/information
- ✓M →Markup → predefined/marked
- ✓L →language→ communication
- ✓Markup language is a set of markup tags
- ✓Tags describe the document content
- ✓HTML Document also Called Web pages
- ✓HTML is Not Case Sensitive

# Structure of HTML



```
<!DOCTYPE html>
<html>
      <head>
            <title>Title of the Page</title>
      </head>
      <body>
      Tags related to layout and formatting
      </body>
</html>
```

Tells the document type

The Root Element

Contains the header information

Defines Title of the Page

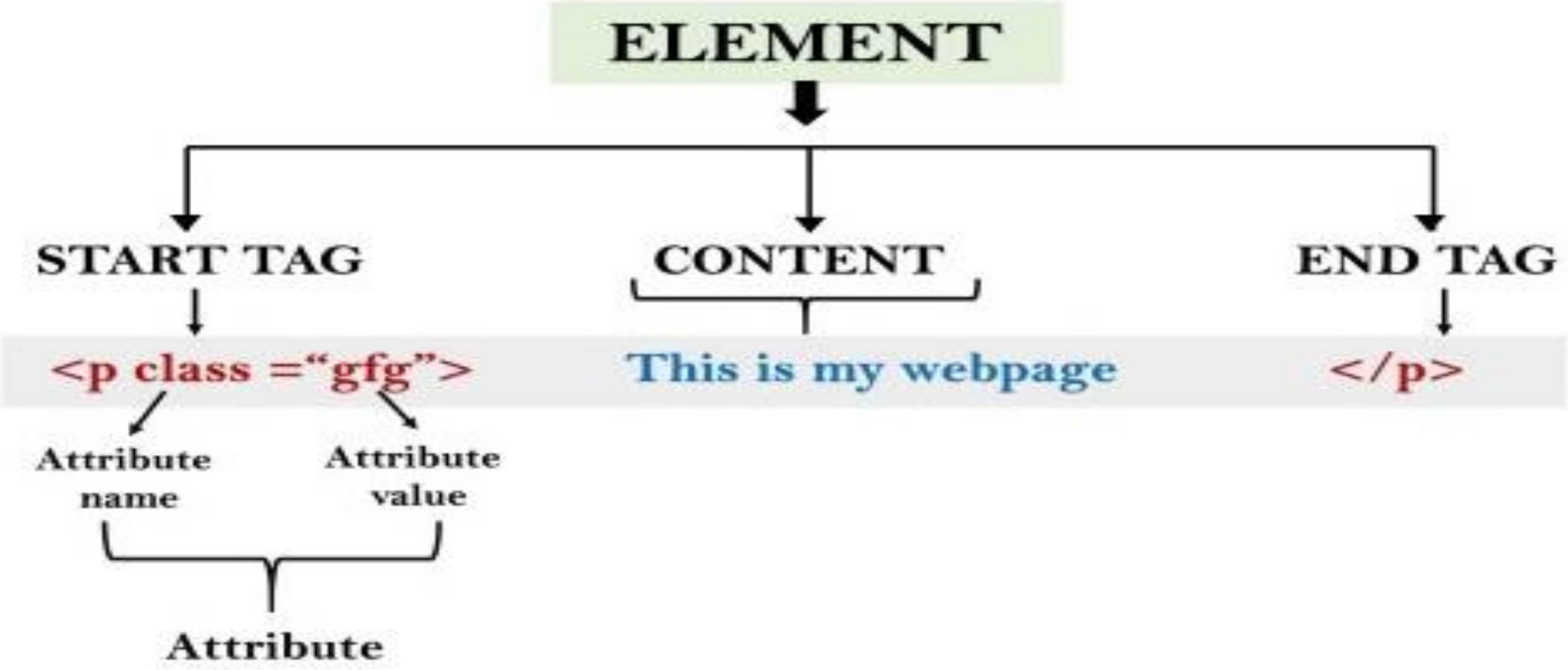
Holds the Content of the Page

- ✓ The <!DOCTYPE html> declaration defines that this document is an HTML5 document
- ✓ The <html> element is the root element of an HTML page
- ✓ The <head> element contains meta information about the HTML page
- ✓ The <title> element specifies a title for the HTML page (which is shown in the browser's title bar or in the page's tab)
- ✓ The <body> element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.

# HTML Elements

- ✓ An HTML element is defined by a start tag, some content, and an end tag.

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



# HTML Elements

## Nested  Html Elements

- ✓ HTML element can be nested (this means that elements can contain other elements).
- ✓ All HTML documents consist of nested HTML elements.

Example:

```
<!DOCTYPE html>
<html>
	<body>
			<h1>My First Heading</h1>
			<p>My first paragraph.</p>
	</body>
</html>
```

# HTML Elements

## Never Skip the End Tag

✓ Some HTML elements will display correctly, even if you forget the end tag:

✓ Example:

```
<!DOCTYPE html>
<html>
    <body>
            <p>This is a paragraph.
            <h1>
            <p>This is a paragraph.
    </body>
</html>
```

# HTML Elements

## Empty HTML Elements

- ✓ HTML elements with no content are called empty elements.
- ✓ The <br> tag defines a line break, and is an empty element without a closing tag.
- ✓ The <hr> tag defines a thematic break in an HTML page, and is most often displayed as a horizontal rule and is an empty element without a closing tag.

### Example:

```
<!DOCTYPE html>
<html>
    <body>
            <p>This is a <br> paragraph with a line break.</p>
            <hr>
    </body>
</html>
```

# HTML Elements

## List of Empty tags

- ✓ <br>: Inserts a line break in a webpage wherever needed.
- ✓ <hr>: Inserts a horizontal line wherever needed in the webpage.
- ✓ <img>: This tag is used to display the images on the webpage which were given in the src attribute of the tag.
- ✓ <input>: This is mainly used with forms to take the input from the user and we can also define the type of the input.
- ✓ <link>: When we store our CSS in an external file this can be used to link external files and documents to the webpage and it is mainly used to link CSS files.
- ✓ <meta>: Contains all metadata of the webpage. Metadata is the data about data and is described in the head tag.
- ✓ <source>: When an external media source is needed to be included in the webpage. source tag is used to insert any media source like audio, video etc… in our webpage.

# HTML attributes

- ✓ HTML attributes provide additional information about HTML elements.
- ✓ All HTML elements can have attributes
- ✓ Attributes are always specified in the start tag
- ✓ Attributes usually come in name/value pairs like: name="value“

Syntax:

```
<element  attribute_name="value">content</element>
```

# HTML Headings

- ✓HTML headings are titles or subtitles that you want to display on a webpage.
- ✓HTML headings are defined with the <h1> to <h6> tags.
- ✓<h1> defines the most important heading.
- ✓<h6> defines the least important heading.

Example:

```
<h1 style="font-size:60px;">Heading 1</h1>
```

# HTML Paragraphs

- ✓ A paragraph always starts on a new line, and is usually a block of text.
- ✓ The HTML <p> element defines a paragraph.
- ✓ A paragraph always starts on a new line, and browsers automatically add some white space (a margin) before and after a paragraph.

Example:

```
<p>
  This paragraphcontains a lot of lines
  in the source code. ,but the
  browser  ignores it.
  </p>
```

# HTML Paragraphs

## HTML <pre>

✓ The text inside a <pre> element is displayed in a fixed-width font (usually Courier), and it preserves both spaces and line breaks.

✓ <pre> define preformatted text.

Example:

```
<pre>
        This paragraphcontains a
        lot of lines in the source
        code.
</pre>
```

# HTML Styles

✓ The HTML style attribute is used to add styles to an element, such as color, font, size, and more.

Syntax:

```
<tagname style="property:value;">
```

The *property* is a CSS property.

The *value* is a CSS value.

Example:

```
<h1 style="text-align:center;">Centered Heading</h1>
<p style="text-align:center;">Centered paragraph.</p>
```

# HTML Text Formatting

- ✓ HTML contains several elements for defining text with a special meaning.
- ✓ <b> - Bold text
- ✓ <strong> - Important text
- ✓ <i> - Italic text
- ✓ <em> - Emphasized text
- ✓ <mark> - Marked text
- ✓ <small> - Smaller text
- ✓ <del> - Deleted text
- ✓ <ins> - Inserted text
- ✓ <sub> - Subscript text
- ✓ <sup> - Superscript text

# HTML Text Formatting

List of Text Formatting

<u> - Underlined text

<strike> - Strikethrough text

<big> - Larger text

<font> - Font size, color, and face

<h1>, <h2>, <h3>, <h4>, <h5>, <h6> - Headings of different levels

<p> - Paragraph

<blockquote> - Blockquote

# HTML Text Formatting

List of Text Formatting

<pre> - Preformatted text

<code> - Code text

<abbr> - Abbreviation OR <acronym> - Acronym

<address> - Address information

<cite> - Citation information

<dfn> - Definition information

# HTML Quotation

## <blockquote>

✓ The HTML <blockquote>element defines a section that is quoted from another source.

Example:

```
<blockquote cite="http://www.worldwildlife.org/who/index.html">
      For 60 years, WWF has worked to help people
      and nature thrive. As the world's leading
      conservation organization, WWF works in
      nearly 100 countries.
</blockquote>
```

# HTML Quotation

<q>

Example: <p>WWF's goal is to: <q>Build a future where people live in harmony with nature.</q></p>

<abbr>

Example: <p>The <abbr title="World Health Organization">WHO</abbr> was founded in 1948.</p>

<address>

Example:

```
<address>
Written by John Doe.<br>
Visit us at:<br>
Example.com<br>
Box 564, Disneyland<br>
USA
</address>
```

# HTML Quotation

<cite>

Example:

```
<p><cite>The Scream</cite> by Edvard Munch.
Painted in 1893.</p>
```

<bdo>

Example:

```
<bdo dir="rtl">This text will be written from right to
left</bdo>
```

# HTML Comment

## Comment Tag

- ✓ you can add comments to your HTML source by using the following syntax:

## Syntax:

```
<!-- Write your comments here -->
```

# HTML Styles - CSS

✓ CSS stands for Cascading Style Sheets.

✓ CSS saves a lot of work. It can control the layout of multiple web pages all at once.

CSS can be added to HTML documents in 3 ways:

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

# HTML Styles - CSS

## Inline CSS

- ✓ An inline CSS uses the style attribute of an HTML element.

Example:

```
<h1 style="color:blue;">A Blue Heading</h1>

<p style="color:red;">A red paragraph.</p>
```

# HTML Styles - CSS

## Internal CSS

- ✓ An internal CSS is defined in the <head> section of an HTML page, within a <style> element.

Example:

```
<style>
    body {background-color: powderblue;}
    h1   {color: blue;}
    p    {color: red;}
</style>
```

# HTML Styles - CSS

## External CSS

- ✓ An external style sheet, add a link to it in the <head> section of each HTML page

Example:

```
<head>
    <link rel="stylesheet" href="styles.css">
</head>
```

# HTML Styles - CSS

## External CSS

- ✓ The external style sheet can be written in any text editor. The file must not contain any HTML code, and must be saved with a .css extension.

Example:

```
body {background-color: powderblue;}
h1 {color: blue;}
p {color: red;}
```

# CSS Colors,Fonts and Sizes

- ✓ The CSS color property defines the text color to be used.
- ✓ The CSS font-family property defines the font to be used.
- ✓ The CSS font-size property defines the text size to be used.

Example:

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

# CSS Colors

- ✓ The CSS color property defines the text color to be used.

Example:

Text-color

```
h1 {color: blue;}
```

Backgrund-color

```
h1{background-color:Tomato;}
```

Border-color

```
h1{border:2px solid violet;}
```

# CSS RGB Colors

- ✓ An RGB color value represents RED, GREEN, and BLUE light sources.
- ✓ In CSS, a color can be specified as an RGB value, using this formula: rgb(red, green, blue)
- ✓ Each parameter (red, green, and blue) defines the intensity of the color between 0 and 255.
- ✓ For example, rgb(255, 0, 0) is displayed as red, because red is set to its highest value (255) and the others are set to 0.
- ✓ To display black, set all color parameters to 0, and white as 255, like this : rgb(0,0,0) and rgb(255,255,255)

Example:

```
        h1{background-color:rgb(23,45,77)}
```

# CSS RGBa Colors

- ✓ RGBA color values are an extension of RGB color values with an alpha channel - which specifies the opacity for a color.
- ✓ An RGBA color value is specified with:

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

Example:

```
h1{ background-color:rgba(23,45,77,0.5) }
```

# CSS Hex Colors

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

Example:

```
        h1{background-color:#3c3c3c;}
```

# CSS HSL Colors

- ✓ HSL stands for hue, saturation, and lightness.
- ✓ In CSS, a color can be specified using hue, saturation, and lightness (HSL) in the form: hsl(hue, saturation, lightness)
- ✓ Hue is a degree on the color wheel from 0 to 360. 0 is red, 120 is green, and 240 is blue.
- ✓ Saturation is a percentage value. 0% means a shade of gray, and 100% is the full color.
- ✓ Lightness is also a percentage. 0% is black, 50% is neither light or dark, 100% is white

Example:

```
h1{background-color:hsl(147, 50%, 47%);}
```

# CSS  HSLa Colors

- ✓ HSLA color values are an extension of HSL color values with an alpha channel - which specifies the opacity for a color.
- ✓ An HSLA color value is specified with:

  hsla(hue, saturation, lightness, alpha)
- ✓ The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (not transparent at all):

Example:

```
h1{background-color:hsla(9, 100%, 64%, 0.4);}
```

# CSS Backgrounds

- ✓ background-color
- ✓ background-image
- ✓ background-repeat
- ✓ background-attachment
- ✓ background-position
- ✓ background (shorthand property)

## CSS background-color

## CSS background-image

The background-image property specifies an image to use as the background of an element.

Example:

```
body { background-image: url(“name.gif"); }
```

# CSS Backgrounds

## CSS background-repeat

- ✓ By default, the background-image property repeats an image both horizontally and vertically.
- ✓ Some images should be repeated only horizontally or vertically, or they will look strange
- ✓ If the image above is repeated only horizontally (background-repeat: repeat-x;), the background will look better.

Example:

```
body {
    background-image: url(“name.png");
    background-repeat: repeat-x;
    }
```

# CSS Backgrounds

## CSS background-repeat: no-repeat

✓ Showing the background image only once is also specified by the background-repeat property.

Example:

```
body {
    background-image: url(“name.png");
    background-repeat: no-repeat;
    }
```

# CSS Backgrounds

## CSS background-position

✓ The background-position property is used to specify the position of the background image.

Example:

```
body {
  background-image: url(“name.png");
  background-repeat: no-repeat;
  background-position: right top;
}
```

# CSS Backgrounds

## CSS background-attachment

- ✓ The background-attachment property specifies whether the background image should scroll or be fixed.

Example:

```
body {
  background-image: url("img_tree.png");
  background-repeat: no-repeat;
  background-position: right top;
  background-attachment: fixed; // background-attachment: scroll;
}
```

# CSS Backgrounds

## CSS background - Shorthand property

- ✓ To shorten the code, it is also possible to specify all the background properties in one single property. This is called a shorthand property.

Example:

```
body {
  background-color: #ffffff;
  background-image: url("img_tree.png");
  background-repeat: no-repeat;
  background-position: right top;
}
```

or

```
body { background: #ffffff url("img_tree.png") no-repeat right top; }
```

# CSS Borders

- ✓ border - Sets all the border properties in one declaration
- ✓ border-bottom - Sets all the bottom border properties in one declaration
- ✓ border-bottom-color - Sets the color of the bottom border
- ✓ border-bottom-style - Sets the style of the bottom border
- ✓ border-bottom-width - Sets the width of the bottom border
- ✓ border-color - Sets the color of the four borders
- ✓ border-left - Sets all the left border properties in one declaration
- ✓ border-left-color - Sets the color of the left border
- ✓ border-left-style - Sets the style of the left border
- ✓ border-left-width - Sets the width of the left border

# CSS Borders

- ✓ border-radius-Sets all the four border-*-radius properties for rounded corners
- ✓ border-right - Sets all the right border properties in one declaration
- ✓ border-right-color - Sets the color of the right border
- ✓ border-right-style - Sets the style of the right border
- ✓ border-right-width - Sets the width of the right border
- ✓ border-style - Sets the style of the four borders
- ✓ border-top - Sets all the top border properties in one declaration
- ✓ border-top-color - Sets the color of the top border
- ✓ border-top-style - Sets the style of the top border
- ✓ border-top-width - Sets the width of the top border
- ✓ border-width - Sets the width of the four borders

## CSS Border

- ✓ The CSS border property defines border around an HTML element.

Example:

```
h1 {   border: 2px solid powderblue;}
```

## CSS Padding

- ✓ The CSS padding property defines a padding (space) between the text and the border.

Example:

```
h1 {   border: 2px solid powderblue; padding: 30px;}
```

## CSS Margin

- ✓ The CSS margin property defines a margin (space) outside the border.

Example:

```
h1 {
            border: 2px solid powderblue;
            margin: 50px;
    }
```

# CSS Outline Style

The outline-style property specifies the style of the outline, and can have one of the following values:

- ✓ dotted - Defines a dotted outline
- ✓ dashed - Defines a dashed outline
- ✓ solid - Defines a solid outline
- ✓ double - Defines a double outline
- ✓ groove - Defines a 3D grooved outline
- ✓ ridge - Defines a 3D ridged outline
- ✓ inset - Defines a 3D inset outline
- ✓ outset - Defines a 3D outset outline
- ✓ none - Defines no outline
- ✓ hidden - Defines a hidden outline

# CSS Outline Style

Example:

```
p {outline-style: dotted; }
p {outline-style: dashed; }
p {outline-style: solid; }
p {outline-style: double; }
p {outline-style: groove; }
p {outline-style: ridge; }
p {outline-style: inset; }
p {outline-style: outset; }
```

# CSS Outline Width

The outline-width property specifies the width of the outline, and can have one of the following values:

- ✓ thin (typically 1px)
- ✓ medium (typically 3px)
- ✓ thick (typically 5px)
- ✓ A specific size (in px, pt, cm, em, etc)

Example:

```
p {
  border: 1px solid black;
  outline-style: solid;
  outline-color: red;
  outline-width: thin;
}
```

# CSS Outline Color

The outline-color property is used to set the color of the outline.

Example:

```
p {
  border: 2px solid black;
  outline-style: solid;
  outline-color: red;
}
```

# CSS Outline Shorthand

The outline property is a shorthand property for setting the following individual outline properties:

- ✓ outline-width
- ✓ outline-style (required)
- ✓ outline-color

Example:

```
p {outline: dashed;}

p {outline: dotted red;}

p {outline: 5px solid yellow;}

p {outline: thick ridge pink;}
```

# CSS Outline Offset

The outline-offset property adds space between an outline and the edge/border of an element. The space between an element and its outline is transparent.

Example:

```
p {
  margin: 30px;
  background: yellow;
  border: 1px solid black;
  outline: 1px solid red;
  outline-offset: 15px;
}
```

# CSS Text

## Text Color and Background Color

- ✓ The color property is used to set the color of the text.
- ✓ we define both the background-color property and the color property.

Example:

```
body {
  background-color: lightgrey;
  color: blue;
}
```

# CSS Text

## Text Alignment

The text-align property is used to set the horizontal alignment of a text.

A text can be left or right aligned, centered, or justified.

When the text-align property is set to "justify", each line is stretched so that every line has equal width, and the left and right margins are straight.

- ✓ text-align
- ✓ text-align-last
- ✓ vertical-align

Example:

```
h1 { text-align: center; }
h2 { text-align: left; }
h3 { text-align: right; }
div { text-align: justify;}
```

# CSS Text

## Text Align Last

The text-align-last property specifies how to align the last line of a text.

### Example:

```
p { text-align-last: right; }
p { text-align-last: center; }
p { text-align-last: justify; }
```

## Text Direction

The direction and unicode-bidi properties can be used to change the text direction of an element.

### Example:

```
p {
  direction: rtl;
  unicode-bidi: bidi-override;
}
```

# CSS Text

## Vertical Alignment

The vertical-align property sets the vertical alignment of an element.

Example:

```
img { vertical-align: baseline; }

img { vertical-align: text-top; }

img { vertical-align: text-bottom; }

img { vertical-align: sub; }

img { vertical-align: super; }
```

# CSS Text Decoration

## Text Decoration

- ✓ text-decoration-line
- ✓ text-decoration-color
- ✓ text-decoration-style
- ✓ text-decoration-thickness
- ✓ text-decoration

## Example:

```
p {
  text-decoration-line: underline;
  text-decoration-color: red;
  text-decoration-style: wavy;

  text-decoration-thickness: auto;
}
```

# CSS Text Decoration

## Text Transformation

The text-transform property is used to specify uppercase and lowercase letters in a text.

Example:

```
p { text-transform: uppercase; }

p { text-transform: lowercase; }

p { text-transform: capitalize; }
```

# CSS Text Decoration

## Text Spacing

- ✓ text-indent
- ✓ letter-spacing
- ✓ line-height
- ✓ word-spacing
- ✓ white-space

## Example:

```
p { text-indent: 50px; }
h1 { letter-spacing: 5px; }
p { line-height: 0.8; }
p { word-spacing: 10px; }
p { white-space: nowrap; }
```

# CSS Text Decoration

## Text Shadow

The text-shadow property adds shadow to text.

Example:

```
h1 { text-shadow: 2px 2px; }
```

# CSS Fonts

In CSS, we use the font-family property to specify the font of a text.

## Example:

p1 { font-family: "Times New Roman", Times, serif; }

## CSS Font Fallbacks

Below are some commonly used font fallbacks, organized by the 5 generic font families:

- ✓ Serif
- ✓ Sans-serif
- ✓ Monospace
- ✓ Cursive
- ✓ Fantasy

# CSS Fonts

## Font Style

The font-style property is mostly used to specify italic text.

This property has three values:

✓ normal - The text is shown normally

✓ italic - The text is shown in italics

✓ oblique - The text is "leaning"

## Example:

```
p { font-style: normal; }
```

# CSS Fonts

## Font Weight

The font-weight property specifies the weight of a font.

Example:

```
p { font-weight: bold; }
```

## Font Variant

The font-variant property specifies whether or not a text should be displayed in a small-caps font.

Example:

```
p { font-variant: small-caps; }
```

# CSS Fonts

## Font Size

- ✓ The font-size property sets the size of the text.
- ✓ Being able to manage the text size is important in web design. However, you should not use font size adjustments to make paragraphs look like headings, or headings look like paragraphs.

## Example:

```
h1 { font-size: 40px; } // It is also in em, px, %
```

# HTML Links

- ✓ HTML links are hyperlinks.
- ✓ we can click on a link and jump to another document.
- ✓ When you move the mouse over a link, the mouse arrow will turn into a little hand.

Syntax

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

Example

<a href=“https://www.google.com”>Google</a>

# HTML Images

- ✓The HTML <img> tag is used to embed an image in a web page.
- ✓The <img> tag is empty, it contains attributes only, and does not have a closing tag.

The <img> tag has two required attributes:

- ✓ src - Specifies the path to the image
- ✓alt - Specifies an alternate text for the image

Syntax

<img src="*url*" alt="*alternatetext*">

# CSS Links

Links can be styled differently depending on what state they are in. The four links states are

- ✓ a:link - a normal, unvisited link
- ✓ a:visited - a link the user has visited
- ✓ a:hover - a link when the user mouses over it
- ✓ a:active - a link the moment it is clicked

Example 1:

```
/* unvisited link */
a:link { color: red; }
/* visited link */
a:visited { color: green; }
/* mouse over link */
a:hover { color: hotpink; }
/* selected link */
a:active { color: blue; }
```

# CSS Links

## Example2:

```
a:link, a:visited {
  background-color: white;
  color: black;
  border: 2px solid green;
  padding: 10px 20px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
}

a:hover, a:active {
  background-color: green;
  color: white;
}
```

# HTML Tables

✓ HTML tables allow web developers to arrange data into rows and columns.

<table> → Defines a table

<th> →Defines a header cell in a table

<tr> →Defines a row in a table

<td> →Defines a cell in a table

<caption> →Defines a table caption

# HTML Tables

Example:

```
<table>
        <tr>
               <th>Company</th>
               <th>Contact</th>
               <th>Country</th>
       </tr>
       <tr>
               <td>CT</td>
               <td>HYD</td>
               <td>INDIA</td>
       </tr>
       <tr>
               <td>Collegs</td>
               <td>Traning</td>
               <td>US</td>
       </tr>
</table>
```

## Table Border

- ✓ When you add a border to a table, you also add borders around each table cell.
- ✓ To add a border, use the CSS border property on table, th and td elements.

## Example:

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

## Table Collapsed

- ✓ To avoid having double borders like in the example above, set the CSS border-collapse property to collapse. This will make the borders collapse into a single border.

Example:

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

## Round Table Borders

- ✓ With the border-radius  property, the borders get rounded corners.

Example:

```
table, th, td {border: 1px solid black;
               border-radius: 10px;}
```

# Dotted Table Borders

- ✓ With the border-style property, you can set the appearance of the border.
- ➢ Dotted
- ➢ Dashed
- ➢ Solid
- ➢ Double
- ➢ Groove
- ➢ Ridge
- ➢ Inset
- ➢ Outset
- ➢ None
- ➢ hidden

Example

```
th,td{
        border-style:dotted;
}
```

## Colspan

✓ To make a cell span over multiple columns, use the colspan attribute.

Example:

```
<table>
     <tr> <th colspan="2">Name</th>
               <th>Age</th>    </tr>
          <tr> <td>CT</td>
               <td> Ramana </td>
               <td>47</td>     </tr>
          <tr> <td>DEV</td>
               <td>RAVI</td>
               <td>46</td>     </tr>
</table>
```

## Rowspan

- ✓ To make a cell span over multiple rows, use the rowspan attribute.

Example:

```
<table>
    <tr>
        <th>Name</th>
        <td>RAMANA</td>    </tr>
    <tr>
        <th rowspan="2">Phone</th>
        <td>9376-8729-7992</td>  </tr>
    <tr>
        <td>6692-2979-2876</td>  </tr>
    </table>
```

## Table Width and Height

The width and height of a table are defined by the width and height properties.

## Example:

```
table { width: 100%; }
th { height: 70px; }
```

## Table Alignment

- ✓ The text-align property sets the horizontal alignment (like left, right, or center) of the content in <th> or <td>.
- ✓ By default, the content of <th> elements are center-aligned and the content of <td> elements are left-aligned.

- ✓ To center-align the content of <td> elements as well, use text-align: center.
- ✓ The vertical-align property sets the vertical alignment (like top, bottom, or middle) of the content in <th> or <td>.
- ✓ By default, the vertical alignment of the content in a table is middle (for both <th> and <td> elements).
- ✓ The following example sets the vertical text alignment to bottom for <td> elements.

Example:

```
td { text-align: center; }

td {
  height: 50px;
  vertical-align: bottom;
}
```

# HTML Lists CSS list Properties

In HTML, there are two main types of lists:

- ✓ unordered lists (<ul>) - the list items are marked with bullets
- ✓ ordered lists (<ol>) - the list items are marked with numbers or letters

The CSS list properties allow you to:

- ✓ Set different list item markers for ordered lists
- ✓ Set different list item markers for unordered lists
- ✓ Set an image as the list item marker
- ✓ Add background colors to lists and list items

# HTML Lists CSS list  Properties

Example:

```
ul { list-style-type: circle; }

ul { list-style-type: square; }

ol { list-style-type: upper-roman; }

ol { list-style-type: lower-alpha; }

ul { list-style-image: url('sqpurple.gif'); }
```

# HTML Lists

✓ HTML lists allow web developers to group a set of related items in lists.

<ul> →Defines an unordered list

<ol> →Defines an ordered list

<li> →Defines a list item

<dl> →Defines a description list

<dt> →Defines a term in a description list

<dd> →Describes the term in a description list

# Block-level Elements

- ✓ A block-level element always starts on a new line, and the browsers automatically add some space (a margin) before and after the element.
- ✓ The <p> element defines a paragraph in an HTML document.
- ✓ The <div> element defines a division or a section in an HTML document.

Example

```
<p>Hello World</p>

<div>Hello World</div>
```

# Inline Elements

✓ An inline element does not start on a new line.

✓ An inline element only takes up as much width as necessary.

Example

```
<p>This is an inline span
<span style="border: 1px solid black">
Hello World
</span>
element inside a paragraph.
</p>
```

# HTML CLASSES

- ✓ The class attribute is often used to point to a class name in a style sheet. It can also be used by a JavaScript to access and manipulate elements with the specific class name.
- ✓ To create a class; write a period (.) character, followed by a class name. Then, define the CSS properties within curly braces {}.

Example:

In head tag --->

```
.city {background-color: tomato;color: white;padding: 10px;}
```

In body tag

```
<h2 class="city">London</h2>
<p>London is the capital of England.</p>
```

# HTML Id

- ✓ The id attribute is used to point to a specific style declaration in a style sheet. It is also used by JavaScript to access and manipulate the element with the specific id.

Example:

In head tag --->

```
#city {background-color: tomato;color: white;padding: 10px;}
```

In body tag

```
<h2 id="city">London</h2>
<p>London is the capital of England.</p>
```

# HTML Iframes

✓ An HTML iframe is used to display a web page within a web page.

Syntax:

```
    <iframe src="url" title="description"></iframe>
```

Example:

```
<iframe src="demo_iframe.html" height="200"
w     idth="300" title="Iframe Example">
</iframe>
```

# HTML Iframes

- ✓ The target attribute specifies where to open the linked document.
- ✓ The target attribute can have one of the following values.

_self - Default. Opens the document in the same window/tab as it was clicked

Example:

```
    <a href="https://www.google.com/">Visit
Google.com!</a>
```

# HTML Iframes

_blank - Opens the document in a new window or tab

Example:

<a href=https://www.google.com/ target ="_blank"> Visit Google.com!</a>

_parent - Opens the document in the parent frame

Example:

<a href=https://www.google.com/ target ="_parent"> Visit Google.com!</a>

# HTML Iframes

_top - Opens the document in the full body of the window

Example:

<a href=https://www.google.com/ target =“_top”> Visit Google.com!</a>

# HTML Forms

- ✓ An HTML form is used to collect user input.
- ✓ The user input is most often sent to a server for processing.

Syntax:

```
<form>
   .
   form elements
   .
</form>
```

# HTML <form> Elements

The HTML <form> element can contain one or more of the following form elements:

1. <form> - Defines an HTML form for user input
2. <input> - Defines an input control
3. <label> - Defines a label for an <input> element
4. <select> - Defines a drop-down list
5. <textarea> - Defines a multiline input control
6. <button> - Defines a clickable button
7. <fieldset> - Groups related elements in a form
8. <legend> - Defines a caption for a <fieldset> element
9. <datalist> - Specifies a list of pre-defined options for input controls
10. <option> - Defines an option in a drop-down list
11. <optgroup> - Defines a group of related options in a drop-down list

✓ The <form> element is a container for different types of input elements, such as: text fields, checkboxes, radio buttons, submit buttons, etc.

## The <input> Element

<input type="text"> →Displays a single-line text input field

<input type="radio"> →Displays a radio button

<input type="checkbox"> →Displays a checkbox

<input type="submit"> →Displays a submit button

<input type="button"> →Displays a clickable button

## HTML Input Types

Here are the different input types you can use in HTML:

- ✓ <input type="button">
- ✓ <input type="checkbox">
- ✓ <input type="color">
- ✓ <input type="date">
- ✓ <input type="datetime-local">
- ✓ <input type="email">
- ✓ <input type="file">
- ✓ <input type="hidden">
- ✓ <input type="image">

- ✓ <input type="month">
- ✓ <input type="number">
- ✓ <input type="password">
- ✓ <input type="radio">
- ✓ <input type="range">
- ✓ <input type="reset">
- ✓ <input type="search">
- ✓ <input type="submit">
- ✓ <input type="tel">
- ✓ <input type="text">
- ✓ <input type="time">
- ✓ <input type="url">
- ✓ <input type="week">

# CSS Navigation Bar

## Navigation Bar

- ✓ Having easy-to-use navigation is important for any web site.
- ✓ Navigation Bar = List of Links
- ✓ A navigation bar needs standard HTML as a base.
- ✓ A navigation bar is basically a list of links, so using the <ul> and <li> elements makes perfect sense.

# CSS Navigation Bar

## Vertical Navigation Bar

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  margin: 0;
}

ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  width: 25%;
  background-color: #f1f1f1;
  position: fixed;
  height: 100%;
  overflow: auto;
}
```

# CSS Navigation Bar

```
li a {
  display: block;
  color: #000;
  padding: 8px 16px;
  text-decoration: none;
}
li a.active {
  background-color: #04AA6D;
  color: white;
}
li a:hover:not(.active) {
  background-color: #555;
  color: white;
}
</style>
</head>
<body>
<ul>
  <li><a class="active" href="#home">Home</a></li>
  <li><a href="#news">News</a></li>
```

# CSS Navigation Bar

```
<li><a href="#contact">Contact</a></li>
  <li><a href="#about">About</a></li>
</ul>
<div style="margin-left:25%;padding:1px 16px;height:1000px;">
  <h2>Fixed Full-height Side Nav</h2>
  <h3>Try to scroll this area, and see how the sidenav sticks to the page</h3>
  <p>Notice that this div element has a left margin of 25%. This is because the side navigation is set to 25% width. If you remove the margin, the sidenav will overlay/sit on top of this div.</p>
  <p>Also notice that we have set overflow:auto to sidenav. This will add a scrollbar when the sidenav is too long (for example if it has over 50 links inside of it).</p>
  <p>Some text..</p>
  <p>Some text..</p>
  <p>Some text..</p>
  <p>Some text..</p>
  <p>Some text..</p>
  <p>Some text..</p>
  <p>Some text..</p>
</div>
</body>
</html>
```

# CSS Navigation Bar

## CSS Horizontal Navigation Bar

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  font-size: 28px;
}
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #333;
  position: -webkit-sticky; /* Safari */
  position: sticky;
  top: 0;
}
li {
  float: left;
}
```

# CSS Navigation Bar

```
li a {
  display: block;
  color: white;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
}
li a:hover {
  background-color: #111;
}
.active {
  background-color: #4CAF50;
}
</style>
</head>
<body>
<div class="header">
  <h2>Scroll Down</h2>
  <p>Scroll down to see the sticky effect.</p>
</div>
```

# CSS Navigation Bar

```
<ul>
  <li><a class="active" href="#home">Home</a></li>
  <li><a href="#news">News</a></li>
  <li><a href="#contact">Contact</a></li>
</ul>
<h3>Sticky Navigation Bar Example</h3>
<p>The navbar will <strong>stick</strong> to the top when you reach its scroll position.</p>
</body>
</html>
```