**UNIT-3**

***CASCADING STYLE SHEET***

- INTRODUCTION TO CSS & TYPES OF CSS
- CLASS & ID SELECTOR
- CSS FONT PROPERTY
- CSS TEXT PROPERTY
- CSS BACKGROUND PROPERTY
- CSS MARGIN PROPERTY
- CSS LIST PROPERTY
- INTRODUCTION TO CSS 3
- CSS 3 BORDER PROPERTY
- CSS 3 GRADIENT PROPERTY
- CSS 3 DROP SHADOW PROPERTY
- CSS 3 2D & 3D TRANSFORM PROPERTY
- CSS 3 TRANSITION PROPERTY
- CSS 3 BOX SIZING PROPERTY
- CSS 3 POSITION PROPERTY
- CSS 3 MEDIA QUERY

- **WHAT IS CSS? EXPLAIN TYPES OF CSS (3 OR 5 M) (MIMP)**

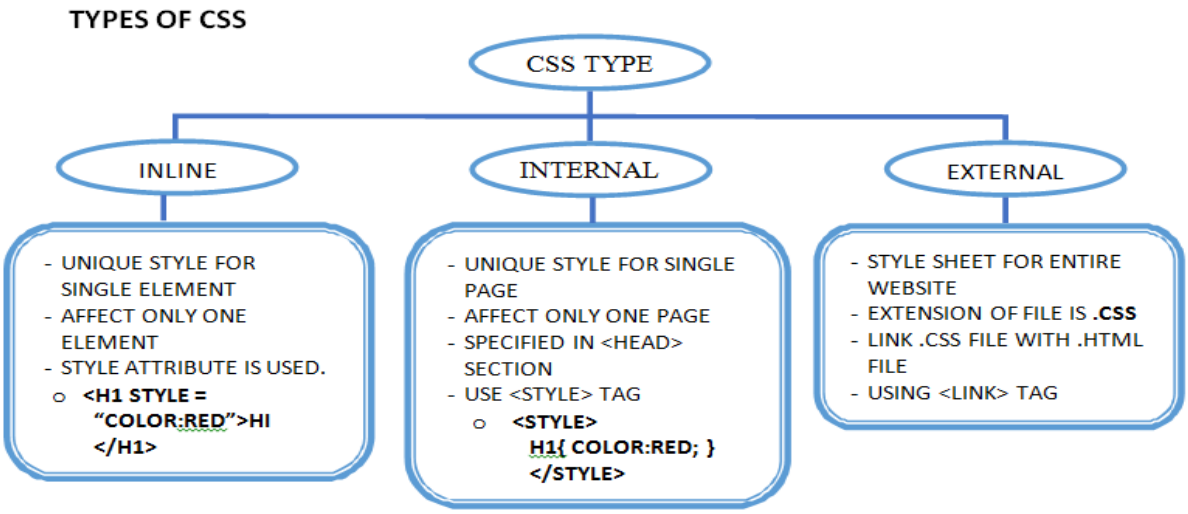
**DETAILS**

**CSS INTORDUCTION**

- **CSS** stands for **C**ascading **S**tyle **S**heets
- Styles define how to display HTML elements
- Styles are normally stored in Style Sheets
- CSS describes **how HTML elements are to be displayed on screen, paper, or in othermedia**
- CSS **saves a lot of work**. It can control the layout of multiple web pages all at once
- External style sheets are stored in CSS files
- CSS Solved a Big Problem
  - HTML was NEVER intended to contain tags for formatting a web page!
  - HTML was created to describe the content of a web page, like:
- <h1>This is a heading</h1>
- To solve this problem, the World Wide Web Consortium (W3C) created CSS.
- CSS removed the style formatting from the HTML page!
- Css Use with style tag. Or script tag using type attribute. Like script type=css"
- Css use in head section.
-

**Advantages of CSS**

- **CSS saves time** − You can write CSS once and then reuse same sheet in multiple HTML pages.
- **Pages load faster** − If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag.So less code means faster download times.
- **Easy maintena**nce − To make a global change, simply change the style, and all elements
  in all the web pages will be updated automatically.




**How to Insert a Style Sheet**

- When a browser reads a style sheet, it will format the document according to it. There arethree ways of inserting a style sheet:

1. External Style Sheet
2. Internal Style Sheet
3. Inline Style

**1. External Style Sheet**

- An external style sheet is ideal when the style is applied to many pages.
- With an external style sheet, you can change the look of an entire Web site bychanging one file.
- Each page must link to the style sheet using the <link> tag.
- The <link> tag goes inside the head section:

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css" />
</head>
```

- The browser will read the style definitions from the file mystyle.css, and format thedocument according to it.
- An external style sheet can be written in any text editor. The file should not containany html tags. Your style sheet should be saved with a .css extension.
- An example of a style sheet file is shown below:

```
Hr {color: sienna}
p {margin-left: 20px}
body {background-image: url("images/back40.gif")}
```

2. **Internal Style Sheet**

- An internal style sheet should be used when a single document has a unique style. You define internal styles in the head section by using the <style> tag, like this:

```
<head>
<style type="text/css">
hr {color: RED}
p {margin-left: 20px}
body {background-image: url("images/back40.gif")}
</style>
</head>
```

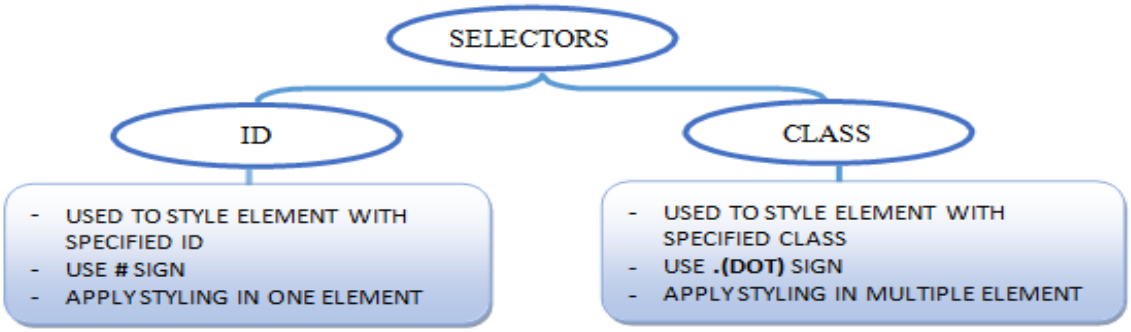
- The browser will now read the style definitions, and format the document according to it.

3. **Inline Style Sheet**

- An inline style loses many of the advantages of style sheets by mixing content with presentation. Use this method sparingly, such as when a style is to be applied to a single occurrence of an element.
- To use inline styles you use the style attribute in the relevant tag. The style attribute can contain any CSS property.
- The example shows how to change the color and the left margin of a paragraph:

```
<p style="color: RED; margin-left:
 20px">This is a paragraph</p>
```

➢ **EXPLAIN CLASS SELECTOR AND ID SELECTOR (3 M) (MIMP)**




**DETAIL**

**Class Selector**

- With the class selector you can define different styles for the same type of HTML element.
- Say that you would like to have two types of paragraphs in your document: one right-aligned paragraph, and one center-aligned paragraph. Here is how you can do it with styles:
  p.right {text-align: right}
  p.center {text-align: center}
- You have to use the class attribute in your HTML document:

  This paragraph will be right-aligned.

**The id Selector**

- The id selector uses the id attribute of an HTML element to select a specific element.
- The id of an element is unique within a page, so the id selector is used to select one unique element!
- You can also define styles for HTML elements with the id selector. The id selector is definedas a #.
  The style rule below will match the element that has an id attribute with a value of "green":#green {color: green}

➢ **EXPLAIN FONT PROPERTY OF CSS (3 OR 5 M)**

**DETAIL**

- The CSS font properties allow you to change the font family, boldness, size, and the style ofa text.

**1. font-family Property**

- The font-family property is a list of font family names and/or generic family names for an element.
- The browser will use the first value it recognizes.
- Separate each value with a comma, and always offer a generic-family name as the lastalternative.

**2. font-style Property**

- The font-style property sets the style of a font.

| Value | Description |
|---|---|

| | |
|---|---|
| Normal | Default. The browser displays a normal font |
| Italic | The browser displays an italic font |
| Oblique | The browser displays an oblique font |

**Example**

```
Body { font-style: italic }
```

**3. font-weight Property**

- The font-weight property sets how thick or thin characters in text should bedisplayed.

| Value | Description |
|---|---|
| Normal | Default. Defines normal characters |
| Bold | Defines thick characters |
| Bolder | Defines thicker characters |
| Lighter | Defines lighter characters |
| 100to900 | Defines from thin to thick characters. 400 is the same as normal, and 700 is the sameas bold |

**Example**

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

**4. font-size Property**

- The font-size property sets the size of a font.

**5. font-variant property**

- The font-variant property is used to display text in a small-caps font, which meansthat all the lower case letters are converted to uppercase letters, but all the letters in the small- caps font have a smaller font-size compared to the rest of the text.

| Value | Description |
|---|---|
| Normal | Default. The browser displays a normal font |
| small-caps | The browser displays a small-caps font |

**Example**

```
P{font-variant: small-caps}
```

- **EXPLAIN TEXT PROPERTY OF CSS (3 OR 5 M)**

**DETAIL**

- The CSS text properties allow you to control the appearance of text.
- It is possible to change the color of a text, increase or decrease the space between characters in a text, align a text, decorate a text, indent the first line in a text, and more

**1. color Property**

- The color property sets the color of a text.

**Examples**

```
Body { color: rgb(255,255,0)}
```

**2. text-align Property**

- The text-align property aligns the text in an element.

**Examples**

```
P{ text-align: center}
```

**3. word-spacing Property**

- The word-spacing property increases or decreases the white space between words.

| Value | Description |
|---|---|
| normal | Default. Defines normal space between words |
| *length* | Defines a fixed space between words |

**1. letter-spacing Property**

- The letter-spacing property increases or decreases the white space between characters.

| Value | Description |
|---|---|
| Normal | Default. Defines normal space between characters |
| *Length* | Defines a fixed space between characters |

**2. TEXT DECORATION**

Text decoration is used to add or remove decorations from the text.

**Text decoration can be underline, overline, line-through or none.**

Syntax:

```
body{
```

text-decoration:decoration type;

}

- **EXPLAIN BACKGROUND PROPERTY OF CSS (3 OR 5 M)(WITH EXAMPLE)(IMP)**

**DETAIL**

o The CSS background properties allow you to control the background color of an element, set an image as the background, repeat a background image vertically or horizontally, and position an image on a page.

1. **background-attachment**

- The background-attachment CSS property defines whether the background image scrollswith the document, or remains fixed to the viewing area.
- **Syntax**

The syntax of the property is given with:

- **Example**

body {

background-image: url("images/recycle.jpg");
background-attachment: fixed;

}

2. **background-color**

- The background-color CSS property sets the background color of an element.

**Example**

body {

background-color: yellow;}

3. **background-image**

- The background-image CSS property sets the background image for an element. It is often more convenient to use the shorthand background property.
- **Syntax**
- **Example**

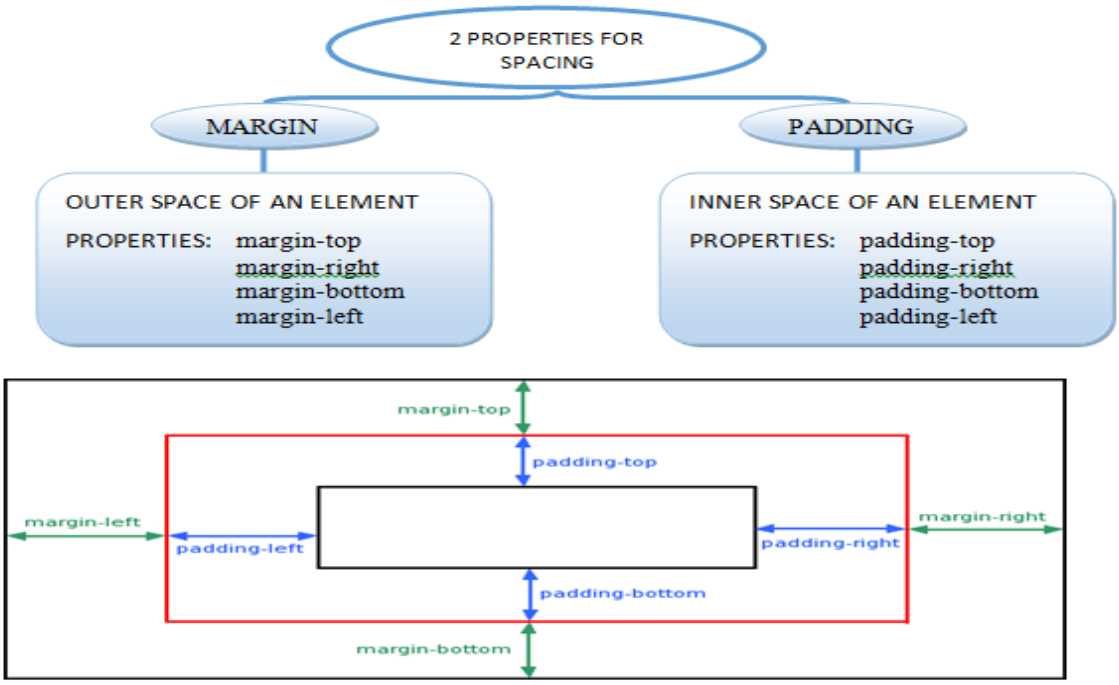
body {

background-image: url("images/pattern.jpg");

}

- **EXPLAIN MARGIN AND PADDING PROPERTIES OF CSS (2 OR 3 M)(5**

**M WITH EXAMPLE)(IMP)**




**DETAIL**

1. **MARGIN PROPERTY**

- The margin CSS property sets the margin on all four sides of the element. It is a shorthand property for margin-top, margin-right, margin-bottom, and margin- left property.
- **Syntax**

margin: length | percentage | auto ]| initial | inherit

- **Example**

```
h1 {
margin-left: 25px;
}
p {
margin: 50px 100px;
}
```

- If one value is set, this margin applies to all 4 sides.
- If two values are set, the first value applies to top and bottom, the second valueapplies to the right and left side.
- Three values apply to the top, horizontal (i.e. right and left) and bottom side.

- Four values apply to the top, right, bottom, left side in that order.

---

**2. PADDING PROPERTY**

- The padding CSS property sets the padding on all four sides of the element. It is a shorthand property for padding-top, padding-right, padding-bottom, and padding- left property.

- Example

```
p.one {
padding-right: 20px;
}
p.two {
padding: 35px 15px;
}
```

- This shorthand notation can take one, two, three, or four whitespace separated values.
  - If one value is set, this padding applies to all 4 sides.
  - If two values are set, the first value applies to top and bottom, the second value applies to the right and left side.
  - Three values apply to the top, horizontal (i.e. right and left) and bottom side.
  - Four values apply to the top, right, bottom, left side in that order.

➢ **EXPLAIN LIST PROPERTIES OF CSS (2 OR 3 M)**

**Detail**

**Types of HTML Lists**

- There are three different types of list in HTML:
  - Unordered lists — A list of items, where every list items are marked with bullets.
  - Ordered lists — A list of items, where each list items are marked with numbers.
  - Definition list — A list of items, with a description of each item.

**Styling Lists with CSS**

- CSS provides the several properties for styling and formatting the most commonly used unordered and ordered lists.
- These CSS list properties typically allow you to:

**1. LIST-STYLE**

- The list-style CSS property defines the display style for list items.
- It is a shorthand property for setting the individual list properties i.e. list-style-type, list-style-position, and list-style-image in a single declaration.

- **Example**
  ul { list-style: circle inside; }

  ol { list-style: upper-latin outside;}

**2. LIST-STYLE-IMAGE**

The list-style-image CSS property specifies an image to be used as a list-item marker.

**3. LIST-STYLE-POSITION**

- The list-style-position CSS property specifies the position of list-item marker with respect tothe list item's principal block box.

- **Example**

  Ol {list-style-position:inside;}

  Ul{list-style-position:outside;}

- **WRITE A NOTE ON CSS 3 (3 M)**

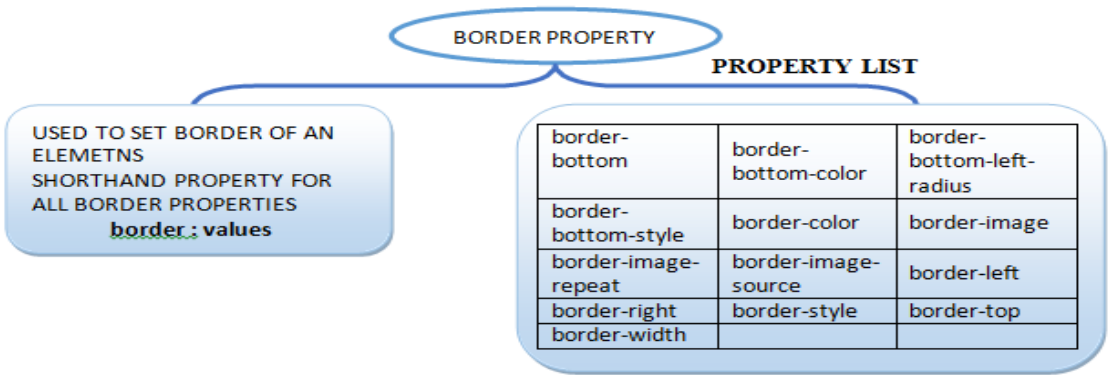
- Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a markup language.CSS3 is a latest standard of css earlier versions(CSS2).The main difference between css2 and css3 is follows
- Media Queries
- Namespaces
- Selectors Level 3
- Color

➢ **CSS3 modules**

- CSS3 is collaboration of CSS2 specifications and new specifications, we can called this collaboration is **module**. Some of the modules are shown below

- Selectors
- Box Model
- Backgrounds
- Image Values and Replaced Content
- Text Effects
- 2D Transformations
- 3D Transformations
- Animations
- Multiple Column Layout
- User Interface

## ➢ EXPLAIN CSS 3 BORDER PROPERTY. (3 M)




**DETAIL**

- The border property sets the width, style, and color for all four sides of an element's border.
- It is a shorthand property for setting the individual border properties i.e. border-width, border-style, and border-color in a single declaration.

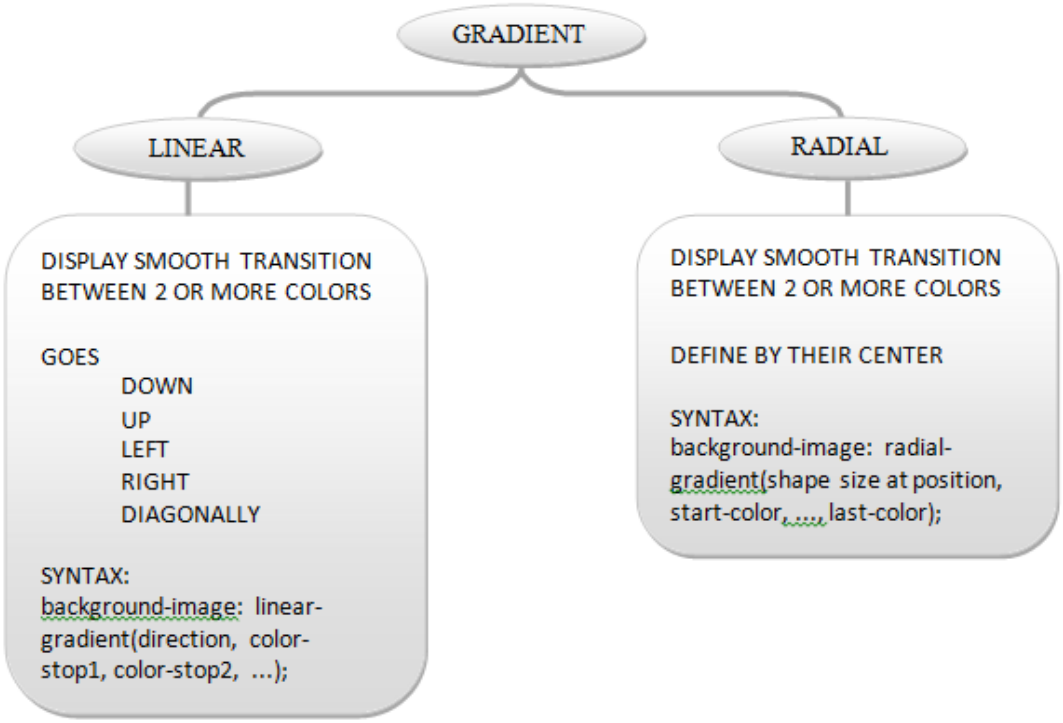
- **Example**

```
h1 {
border: 5px solid #ff0000;
}
p {
color: #00ff00; border: 5px
solid;
}
```

- **Property Values**

| Value | Description |
|---|---|
| border-width | Sets the width of the border of an element. |
| border-style | Sets the line style of the border of an element. |
| border-color | Sets the color of the border of an element. |
| initial | Sets this property to its default value. |

- **EXPLAIN CSS 3 GRADIENT PROPERTY. (3 M) (IMP)**




**DETAIL**

- The CSS3 gradient feature **provides a flexible solution to generate smooth transitions betweentwo or more colors**.
- Earlier, to achieve such effect we had to use the images.
- Using CSS3 gradients you can reduce the download time and saves the bandwidth usages.
- The elements with gradients can be scaled up or down to any extent without losing the quality, also the output will render much faster because it is generated by the browser.

- **Gradients are available in two styles: linear and radial.**

- **Creating CSS3 Linear Gradients**
  - To create **a linear gradient you must define at least two color stops**.

**linear-gradient(direction, color-stop1, color-stop2, ...)**

  - **Linear Gradient - Top to Bottom**
- The following example will create a linear gradient from top to bottom. This is default.

```
.gradient {
/* Fallback for browsers that don't support gradients */
background: red;
/* For Safari 5.1 to 6.0 */
background: -webkit-linear-gradient(red, yellow);
/* For Internet Explorer 10 */
background: -ms-linear-gradient(red, yellow);
/* Standard syntax */
background: linear-gradient(red, yellow);
}
```

  - **Linear Gradient - Left to Right**
- The following example will create a linear gradient from left to right.

```
.gradient {
/* Fallback for browsers that don't support gradients */background: red;
/* For Safari 5.1 to 6.0 */
background: -webkit-linear-gradient(left, red, yellow);
/* For Internet Explorer 10 */
background: -ms-linear-gradient(left, red, yellow);
/* Standard syntax */
background: linear-gradient(to right, red, yellow);
}
```

  - **Linear Gradient - Diagonal**
- You can also create a gradient along the diagonal direction.
- The following example will create a linear gradient from the bottom left corner to the top right corner of the element's box.

```
.gradient {
/* Fallback for browsers that don't support gradients */
background: red;
/* For Safari 5.1 to 6.0 */
```

```
background: -webkit-linear-gradient(bottom left, red, yellow);
/* For Internet Explorer 10 */
background: -ms-linear-gradient(bottom left, red, yellow);
/* Standard syntax */
background: linear-gradient(to top right, red, yellow);
}
```

- **EXPLAIN CSS 3 DROP SHADOW PROPERTY. (3 M)**

**DETAIL**

- The CSS3 gives you ability to add drop shadow effects to the elements like you do in Photoshop without using any images.
- Prior to CSS3, sliced images are used for creating the shadows around the elements that was quite annoying.
- **CSS3 box-shadow Property**

o The box-shadow property can be used to add shadow to the element's boxes.

o You can even apply more than one shadow effects using a comma-separated list of shadows.

The basic syntax of creating a box shadow can be given with: box-shadow: offset-x offset-y blur-radius color;

o The components of the box-shadow property have the following meaning:

**offset-x** — Sets the horizontal offset of the shadow.

**offset-y** — Sets the vertical offset of the shadow.

**blur-radius** — Sets the blur radius. The larger the value, the bigger the blur and more the shadow's edge will be blurred. Negative values are not allowed.

**color** — Sets the color of the shadow. If the color value is omitted or not specified, it takes the value of the color property.

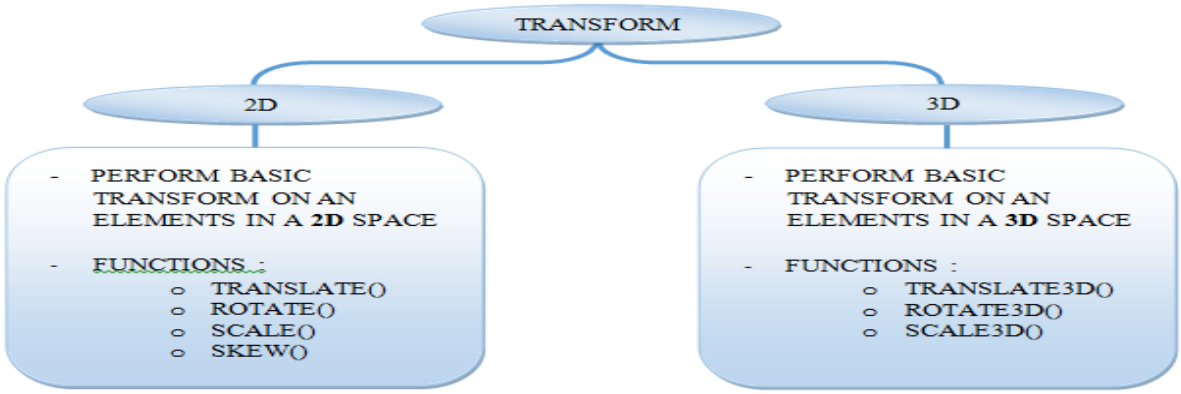
**Example:**

```
<!DOCTYPE html>
<html ><head><style>
.box{
width: 200px; height:150px;
```

```
background: #112BCC;
box-shadow: 5px 5px 10px red, 10px 10px 20px yellow;
  }
```

**output**

- **EXPLAIN CSS 3 2D & 3D TRANSFORM PROPERTY. (3 OR 5M) (MIMP)**




**DETAIL**

1. **2D Transformation of Elements**

- With CSS3 2D transform feature you can perform basic transform manipulations such as move, rotate, scale and skew on elements in a two-dimensional space.
- A transformed element doesn't affect the surrounding elements, but can overlap them, just like the absolutely positioned elements.
- Using CSS Transform and Transform Functions

a. **The translate() Function**

i. Moves the element from its current position to a new position along the X and

Y axes. This can be written as translate(tx, ty). If ty isn't specified, its value is assumed to be zero.

**Example**

```
img {
  -webkit-transform: translate(200px, 50px);  /* Chrome, Safari,Opera */
-moz-transform: translate(200px, 50px);  /* Firefox */
-ms-transform: translate(200px, 50px);  /* IE 9 */
```

**b. transform: translate(200px, 50px); /* Standard syntax */ The rotate() Function**

**i.** The rotate() function rotates the element around its origin (as specified by the transform-origin property) by the specified angle. This can be written as rotate(a).

**Example**

```
img {
-webkit-transform: rotate(30deg);  /* Chrome, Safari, Opera */
-moz-transform: rotate(30deg);  /* Firefox */
-ms-transform: rotate(30deg); /* IE 9 */ transform: rotate(30deg);  /* Standard syntax */
}
```

**The function rotate(30deg) rotates the image in clockwise direction about its origin by the angle 30 degrees. You can use negative values to rotate the element counter-clockwise.**

**c. The scale() Function**

**i.** The scale() function increases or decreases the size of the element. It can bewritten as scale(sx, sy). If sy isn't specified, it is assumed to be equal to sx.

**Example**

```
img {
-webkit-transform: scale(1.5);  /* Chrome, Safari, Opera */
-moz-transform: scale(1.5);  /* Firefox */
-ms-transform: scale(1.5);  /* IE 9 */ transform: scale(1.5);  /*
Modern Browsers */
}
```

**The function scale(1.5) proportionally scale the width and height of the image 1.5 times to its original size. The function scale(1) or scale(1, 1) has no effect on the element.**

**d. The skew() Function**

**i.** The skew() function skews the element along the X and Y axes by the specified angles. It can be written as skew(ax, ay). If ay isn't specified, its value is assumed to be zero.

**Example**

```
img {
-webkit-transform: skew(-40deg, 15deg);  /* Chrome, Safari, Opera */
-moz-transform: skew(-40deg, 15deg);  /* Firefox */
-ms-transform: skew(-40deg, 15deg);  /* IE 9 */
transform: skew(-40deg, 15deg);  /* Modern Browsers */
}
```

The function skew(-40deg, 15deg) skews the element -40 degree horizontally along the x-axis, and 15 degree vertically along the y-axis.

**3D Transformation of Elements**

- With CSS3 3D transform feature you can perform basic transform manipulations such as move, rotate, scale and skew on elements in a three-dimensional space.
- A transformed element doesn't affect the surrounding elements, but can overlap them, just like the absolutely positioned elements. However, the transformed element still takes space in the layout at its default (un-transformed) location.
- Using CSS Transform and Transform Functions
- The CSS3 transform property uses the transform functions to manipulate the coordinate system used by an element in order to apply the transformation effect.

**a. The translate3d() Function**

- Moves the element from its current position to a new position along the X, Y and Z-axis. This can be written as translate(tx, ty, tz). Percentage values are not allowed for third translation-value parameter (i.e. tz).

**Example:**

```
.container { width: 125px;
```

```
height: 125px;
perspective: 500px; border: 4px solid
#e5a043;background: #fff2dd;
}
.transformed {
-webkit-transform: translate3d(25px, 25px, 50px); /* Chrome, Safari,
Opera */
transform: translate3d(25px, 25px, 50px); /* Standard syntax */
}
```

The function translate3d(25px, 25px, 50px) moves the image 25 pixels along the positive X and Y-axis, and the 50 pixels along thepositive Z-axis.

**a. The scale3d() Function**

- The scale3d() function changes the size of an element. It can be written as scale(sx, sy, sz). The effect of this function is not evident unless you use itin combination with other transform functions such as rotate and the perspective, as shown in the example below.

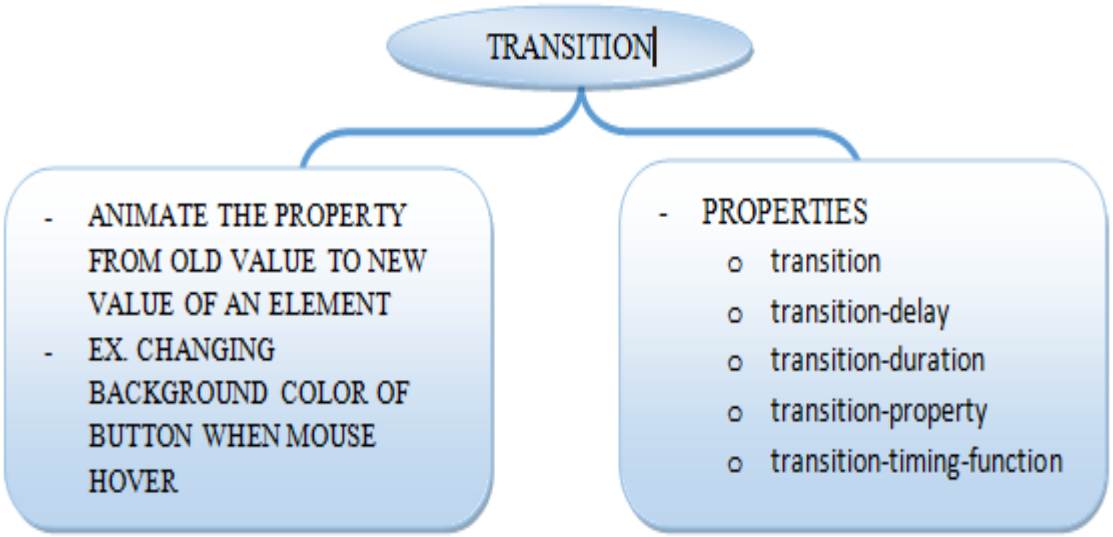
**Example**

```
.container{ width: 125px;
height: 125px;
perspective: 300px; border: 4px solid
#6486ab;background: #e9eef3;
}
.transformed {
-webkit-transform: scale3d(1, 1, 2) rotate3d(1, 0, 0, 60deg); /* Chrome,
Safari, Opera */
transform: scale3d(1, 1, 2) rotate3d(1, 0, 0, 60deg); /* Standard syntax
*/
}
```

The function scale3d(1, 1, 2) scales the elements along the Z-axis and the function rotate3d(1, 0, 0, 60deg) rotates the image along the X-axis by the angle 60 degrees.

**3D Transform Functions**

## ➢ EXPLAIN CSS TRANSITION PROPERTY. (3 M)




**DETAIL**

- Normally when the value of a CSS property changes, the rendered result is instantly updated. A common example is changing the background color of a button on mouse hover.
- In a normal scenario the background color of the button is changes immediately from the old property value to the new property value when you place the cursor over the button.
- CSS3 introduces a new transition feature that allows you to animate a property from the old value to the new value smoothly over time.
- The following example will show you how to animate the background-color of an HTML button on mouse hover.

**Example**

```
button {
background: #fd7c2a;
/* For Safari 3.0+ */
-webkit-transition-property: background;
-webkit-transition-duration: 2s;
/* Standard syntax */
```

```
transition-property:        background;
transition-duration: 2s;
}
button:hover  {  background:
#3cc16e;
}
```

**Performing Multiple Transitions**

Each of the transition properties can take more than one value, separated by commas, which provides an easy way to define multiple transitions at once with different settings.

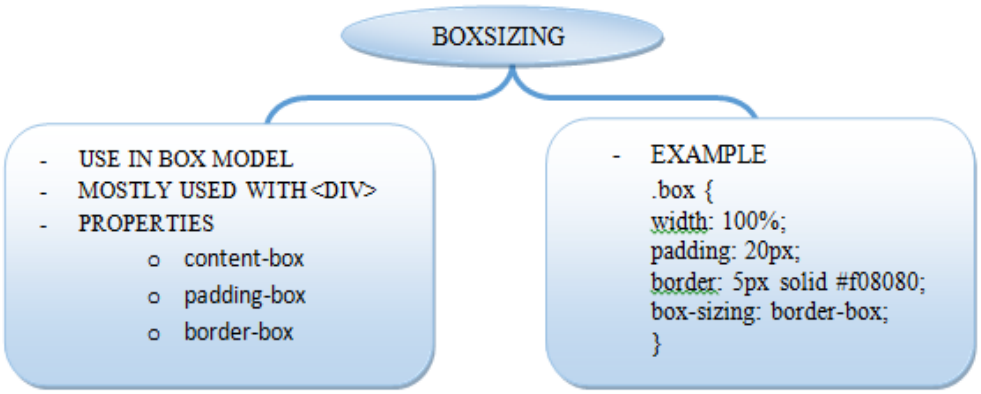
**Example**

```
button {
background:  #fd7c2a;  border:  3px
solid #dc5801;
/* For Safari 3.0+ */
-webkit-transition-property: background, border;
-webkit-transition-duration: 1s, 2s;
/* Standard syntax */
transition-property: background, border;transition-
duration: 1s, 2s;
}
button:hover { background: #3cc16e;
border-color: #288049;
}
```

**CSS3 Transition Properties**

| **roperty** | **)escription** |
|---|---|
| transition | A shorthand property for setting all the four individual transitionproperties in a single declaration. |
| transition-delay | Specifies when the transition will start. |
| transition-duration | Specifies the number of seconds or milliseconds a Transition animationshould take to complete. |
| transition-property | Specifies the names of the CSS properties to which a transition effectshould be applied. |
| transition-timing function | Specifies how the intermediate values of the CSS properties eing affectedby a transition will be calculated. |

## ➢ EXPLAIN CSS 3 BOXSIZING PROPERTY. (2 M)



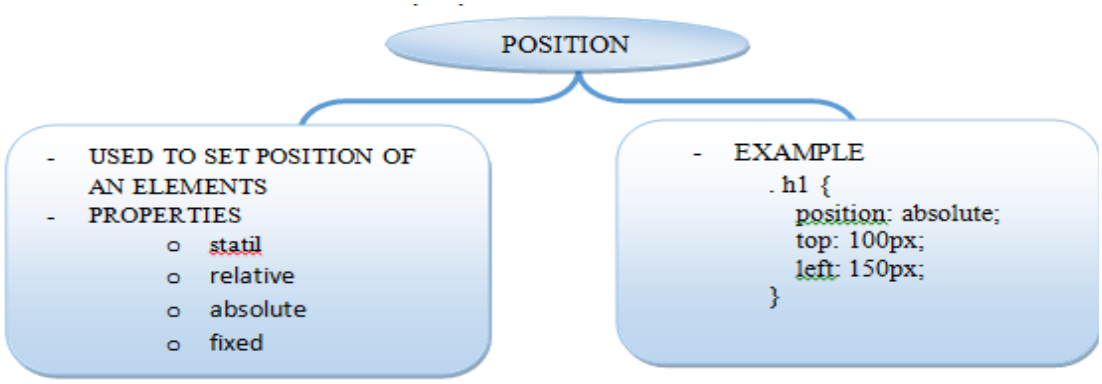

**DETAIL**

- The box-sizing CSS property is used to alter the default CSS box model, which is normally usedby the browser to calculate the widths and heights of the elements.
- **Syntax**

**Example**

```
.box {
                width:50%;
                 padding:15px;
                 float:left;
                 border: 5px solid #000;
box-sizing: border-box;
}
```

## ➢ EXPLAIN CSS 3 POSITION PROPERTY. (2 M)




**DETAIL**

- The position CSS property specifies how an element is positioned.
- Tip: Elements with a position other than static are said to be positioned. Their vertical placement in the stacking context is determined by the z-index property.

**Example**

```
h1 {
position: absolute; top:
100px;
left: 150px;
}
```

## ➢ EXPLAIN CSS 3 MEDIA QUERIES. (3 OR 5 M)(MIMP)

- Media queries allow you to customize the presentation of your web pages for a specific range ofdevices like mobile phones, tablets, desktops, etc. without any change in markups.
- Media queries can be used to check many things, such as:
  - width and height of the viewport
  - width and height of the device
  - orientation (is the tablet/phone in landscape or portrait mode?)
  - resolution
- Using media queries are a popular technique for delivering a tailored style sheet to tablets,iPhone, and Androids.
- A media query consists of a media type and zero or more expressions that match the type and conditions of a particular media features such as device

width or screen resolution.

- Since media query is a logical expression it can be resolve to either true or false.
- The result of the query will be true if the media type specified in the media query matches the type of device the document is being displayed on, as well as all expressions in the media query are satisfied.
- When a media query is true, the related style sheet or style rules are applied to the target device. Here's a simple example of the media query for standard devices.

**Syntax**

- A media query consists of a media type and can contain one or more expressions, which resolveto either true or false.

```
@media not|only mediatype and (expressions) {
CSS-Code;
}
```

**Example**

```
                    /* Smartphones (portrait and landscape)      */
                @media screen and (min-width: 320px) and (max-width: 480px)
                 {
                 /* styles */
                }
                        /* Smartphones (portrait) ------ */
                         @media screen and (max-width: 320px)
                         {
/* styles */
                            }
/* Smartphones (landscape) ----------------------------*/
@media screen and (min-width: 321px){
/* styles */
}
/* Tablets, iPads (portrait and landscape)---------------------------- */
@media screen and (min-width: 768px) and (max-width: 1024px){
/* styles */
}
```

```
/* Tablets, iPads (portrait) ----------------------------*/
@media screen and (min-width: 768px){
/* styles */
}
/* Tablets, iPads (landscape) ---------------------------- */
@media screen and (min-width: 1024px){
/* styles */
}
/* Desktops and laptops ----------------------------*/
@media screen and (min-width: 1224px){
/* styles */
}
/* Large screens ------------------------------- */
@media screen and (min-width: 1824px){
/* styles */
}
```

**Tip: Media queries are an excellent way to create responsive layouts. Using media queries you can customize your website differently for users browsing on devices like smart phones or tablets without changing the actual content of the page.**