CSS (Cascading Style Sheets)

GEEKYSHOWS YOUTUBE CHANNEL LEARNING NOTES

Source Code- https://github.com/satyam-seth/css_learning/tree/master/Geekyshows

Playlist Link- https://youtube.com/playlist?list=PLbGui_ZYuhiisW-k9eEON2vxGABv-PafB

SATYAM SETH

25-09-2021

INDEX

Name. Satyam Seth Sub. CSS & CSS3

Std. Div. Roll No.

S. No:	Date	Title	Page No.	Teacher's Sign / Remarks

Extramarks
Date 18 / 6 / 19
Page No. 1

Cascading Style Sheet (CSS) - CSS is a simple mechanism of describing common presentation semantics for every page in a website. It can control the layout of multiple web pages all at once.

Version of CSS-

- CSS1
- CSS 2
- CSS 3

CSS Syntax-

```
Selector { Property1: property1-value ; Property2: property2-value;
           Property N : propertyN-value; }
```

or

```
Selector {
        Property 1: property1-value;
        Property2: property 2-value;
        Property N : property N-Value;
    }
```

Selector- The selector points to the HTML element or HTML Tag you want to style. It is used to find or select HTML elements based on their element name, id, class, attribute, and more.

Property- It indicates that these properties are defined for that element or selector.

Page No. 2

Property-Value - These are values assigned to Properties.

Ex- p{ color: red; font-size: 22px; }
or p{
color: red;
font-size: 22px;
}

Element Selector - The element selector selects based on the element name.

Syntax - Selector { Property 1: property 1-value ; Property 2: property-value ; Property N: property N-value; }

Ex - (i) p{ color: red; font-size: 22px; }
(ii) h1{ color: blue; font-size: 22em; }

Way of inserting CSS -

1- External style sheet
2- Internal style sheet / Embedded Style Sheet
3- Inline style

1- **External Style Sheet** - An external style sheet is a separate document that contains only CSS rules. An external style sheet helps to change the look of an entire website by changing just on css file. It should not contain any HTML Tags. It has .css extention.

Ex- p{ color: red; font-size: 24px; }
h1{ color: blue; font-size: 24em; }

Page No. 3

(i) Open notepad++ or any other Editor or IDE
(ii) Write CSS Code
(iii) Save with .css extension for example geekyshows.css

How to link web Page to an External Style Sheet - The href attribute with <link> element inside the <head> tag is used to link web page to an external style sheet.

Syntax - <head>
< link rel = " style sheet " href = " cssfile name/ Path " >
</head>

Ex - <html>
<head>
<title> Welcom to Geekyshows </ title>
<link rel="stylesheet" href="geekyshows.css">
</head>
<body>
<h1> I am Heading </h1>
 I am first Paragraph
 I am second Paragraph
</body>
</html>

Note - यदि .css और .html file same location पर नहीं है तो हमें href attribute में .css file का name with path देना होगा।

.CSS1.CSS
→ p{ color: red; font-size: 24px;}
h1 { color: blue; font-size: 4em; }
.CSS1.html
ऊपर का Example.

Page No. 4

2- Internal Style Sheet - Internal Style sheet is a set of style sheet may be used if one single page has a unique style.

Internal Style sheets are created using <style> element, which is added inside the <head> element of the HTML document.

Syntax- <head>

```
<style type="text/css">
    selector { property1: property1 value; ____
    :
    :
</style>
```

→

```
<!DOCTYPE html>
<html>
    <head>
        <title> Hello CSS </title>
        <style type="text/css">
            p{color: red; font-size: 24px;}
            h1{color: blue; font-size: 6em;}
        </style>
    </head>
    <body>
        <h1> I am Heading </h1>
        <p> I am first Paragraph. </p>
        <p> I am second Paragraph. </p>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 5

3- Inline Style- Inline style is useful when we need to define specific style for individual elements present on a web page. The style attribute in a specific Tag or element, is used to create inline style. The style attribute can contain any CSS property between double quote.

Syntax- <tagname style="Property1: property1value; ——— ">
——————— </tagname>.

Ex- For paragraph -
 I am first paragraph

For heading:-
<h1 style="color: red; font-size: 22px;" > I am Heading </h1>

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> Hello CSS </ title>
    </head>
    <body>
        <h1 style="color: blue; font-size: 6em;" >
                I am Heading </h1>
        <p style="color: red; font-size: 24px;" >
                I am first Paragraph</p>
        <p> I am second Paragraph </p>
    </body>
</html>
```

Multiple Style Sheet -
geekyshows.CSS- h1{ color: orange; font-size: 25em; }
p{ color: green; font-size: 20px;}

Page No. 6

```
→ <html>
    <head>
        <title>Hello CSS</title>
        <link rel="stylesheet" href="geekyshows.css">
        <style type="text/css">
            h1{color: blue; font-size: 35em;}
            p{color: red; font-size: 30px;}
        </style>
    </head>
    <body>
        <h1 style="color: cyan; font-size: 45 em;">
                I am Hading</h1>
        <p style="color: yellow; font-size: 40px;">
                I am 1st Para</p>
    </body>
</html>
```

External Style Sheet

Internal Style Sheet

Inline Style

Priority of Style Sheets -

1- Inline Styles
2- External or Internal Style Sheets
3- Browser default

Priority - External or Internal -

- If the internal style is defined after the link to the external style sheet then Internal Style has highest priority.

Ex-
```
<html>
    <head>
        <title>Hello CSS</title>
```

Extramarks
Date ___/___/___
Page No. 7

External Style Sheet - <link rel="stylesheet" href="geekyshows.css">

Internal Style Sheet
```
<style type="text/css">
    p{ color: red; font-size: 30px;}
    h1{ color: blue; font-size: 35 em;}
</style>
</head>
```

- If the internal style is defined before the link to the external style sheet then External Style has highest priority.

Ex-
```
<html>
    <head>
        <title> Hello CSS</title>
        <style type="text/css">
            p{ color: red; font-size: 30px;}
            h1{ color: blue; font-size: 35 em;}
        </style>
        <link rel="stylesheet" href="geekyshows.css">
    </head>
```

Internal Style Sheet

External Style Sheet

Comment- A CSS comment starts with /* and ends with */.

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> Hello CSS</title>
        <style>
            # geekyshows{
                /* I am Comment */
                color: red; font-size: 60px;
            }
        </style>
```

Date 19/6/19
Page No. 8

```
        <p id="geekyshows">Hello Word</p>
        <p> This is Paragraph</p>
    </body>
</html>
```

id Selector- The id selector uses the id attribute of an HTML element to select a specific element.

- The id of an element should be unique within a page, so the id selector is used to select one unique element.
- To select an element with a specific id, write a hash (#) character, followed by the id of the element.

Rules-

- Must begin with a letter A-Z or a-z
- A id name cannot start with a number
- Must not contain any space characters
- Can be followed by: letters (A-Z, a-z), digits (0-9), hyphens ("-"), and underscores ("_").
- In HTML, all values are case-insensitive.

Syntax- <head>

```
        <style>
            #id-selector-name{
                property1:value;property2:value;
            }
        </style>
```

Note- इसका use External StyleSheet के साथ ही कर सकते हैं।

Extramarks
Date ___/___/___
Page No. 9

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
    ~~</head>~~<style>
            #geekyshows{
                color:red; font-size: 60px;
            }
        </style>
    </head>
    <body>
        <p id="geekyshows">Hello World</p>
        <p>Example of Second Paragraph</p>
    </body>
</html>
```

CSS6ii.css - #geekyshows{color: blue; font-size:20px;}

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <link rel="stylesheet" href="CSS6ii.css">
    </head>
    <body>
        <p id="geekyshows">Example of First Paragraph</p>
        <p>Example of Second Paragraph</p>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 10

Class ~~Selectors~~ Selector / Style Class - Class selectors are used when you want to apply a style either on some of tag or across the several tags without repeating the style rule in an HTML document. Using this method of creating styles, you can create styles in the form of style classes in an external style sheet or as an internal style sheet. The class attribute is used to apply style ~~calls~~ class in HTML Tag.

Type of Style Class -

1. Universal style classes
2. Element specific style class

Rule -

- Must begin with a letter A-Z or a-z.
- A class name cannot start with a number.
- Must not contain any space characters.
- Can be followed by: letters (A-Z, a-z), digits (0-9), hyphens ("-"), and underscores ("_").
- In HTML, all values are case-insensitive.

Universal style classes - Universal style class starts with a dot operator followed by the class name.

Syntax- <head>

<style>

.class-name { class definition }

</style>

</head>

Extramarks
Date ___/___/___
Page No. 11

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            .red {color:#FF0000; font-size:60px;}
        </style>
    </head>
    <body>
        <h1 class="red">Example of Heading h1</h1>
        <p class="red">Example of First Paragraph</p>
        <p> Example of Second Paragraph</p>
    </body>
</html>
```

CSS7ii.css – .red {color:#FF0000; font-size:60px;}

→
```
<!DOCTYPE html>
<html>
<head>
    <title>CSS</title>
    <link rel="stylesheet" href="CSS7ii.css">
</head>
<body>
    <h1 class="red">Example of Heading h1</h1>
    <p class="red">Example of First Paragraph</p>
    <p>Paragraph</p>
</body>
</html>
```

Page No. 12

Element Specific Style Classes- An element specific style class starts with the element name, followed by a dot operator, which is followed by the class name.

Syntax- <head>
<style>
element_name.class_name{ class definition }
</style>

Ex- <style>
p.red{color:#FF0000; font-size:60px;}
</style>

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            p.red{color:#FF0000; font-size:60px;}
        </style>
    </head>
    <body>
        <h1 class="red">Example of Heading h1</h1>
        <p class="red">Example of First Paragraph</p>
        <p>Example of Second Paragraph</p>
    </body>
</html>
```

CSS8ii.css- p.red{color:#FF0000; font-size:60px;}

→
```
<!DOCTYPE html>
<html>
```

Extramarks
Date ___/___/___
Page No. 13

```
<head>
    <title>CSS</title>
    <link rel="stylesheet" href="CSS8.i.css">
</head>
<body>
    <h1 class="red">Example of Heading h1</h1>
    <p class="red">Example of First Paragraph</p>
    <p>Example Of Second Paragraph</p>
</body>
</html>
```

Note- p.red class केवल p tag के लिये ही काम करेगा।

Use two or more class -

```
<!DOCTYPE html>
<html>
<head>
    <title>CSS</title>
    <style>
        p.red {color:#FF0000;}
        p.look {font-size:60px;}
        .algn {text-align:center;}
    </style>
    <body>
        <h1 class="algn">Example Of Heading h1</h1>
        <p class="red look">Example Of First Paragraph</p>
        <p class="red algn">Example Of Second Paragraph</p>
    </body>
</html>
```

Page No. 14

CSS9ii.CSS - p.red { color: #FF0000; }
p.look { font-size: 60px; }
.algn { text-align: center; }

```
→ <!DOCTYPE html>
<html>
<head>
    <title> CSS </title>
    <link rel="stylesheet" href="CSS9ii.CSS">
</head>
<body>
    <h1 class="algn">Example Of Heading h1</h1>
    <p class="red look">Example Of First Paragraph
                                    </p>
    <p class="red algn">Example Of Second
                                    Paragraph </p>
</body>
</html>
```

Grouping Selectors - To group selectors, seperate each selector with a comma.

CSS10ii.CSS

```
→ h1 {
        color: red; font-size: 30px;
    }
  p {
        color: red; font-size: 30px;
    }
  h2 {
        color: red; font-size: 30px;
    }
  h3 {
        color: red; font-size: 30px;
    }
```

Extramarks
Date ___/___/___
Page No. 15

Syntax- h1, p, h2, h3 {
color: red; font-size: 30 px;
}

→ <!DOCTYPE html>
<html>
<head>
<title> CSS </title>
~~<sta~~ <style type="text/ css">
p, h1{ color: red; font-size: 24px;}
</style>
</head>
<body>
<h1 ~~class="align"~~ > Example of Heading h1</h1>
~~<p class="red-bee"~~
 Example of Paragraph
</body>
</html>

CSS10.ii.css- p, h1{ color: red; font-size: 24px}

→ <!DOCTYPE html>
<html>
<head>
<title> CSS </title>
<link rel="stylesheet" href="CSS10.ii.css">
</head>
<body>
<h1> Example of Heading h1 </h1>
 Example of Paragraph
</body>
</html>

Page No. 16

Backgroud - We can use background properties to enhance the background style of the element presents in the HTML web page.

Background Properties -

1- background - color
2- background - image
3- background - repeat
4- background - position
5- background - attachment
6- background - clip
7- background - size
8- background - origin
9- background

} Newly added in CSS 3

1- background - color - The background-color property allows to set the background color of an HTML element. We can use this property to a predefined color name, color value or transparent.

Syntax - color name - red, green, blue etc.

Color Value -

Hx Value - #F0E68C

RGB value - rgb(240,230,140)

Ex - (i) body {background-color : khaki;}

(ii) body {background-color : #F0E68C;}

(iii) body {background - rgb(240,230,140);}

Extramarks
Date ___/___/___
Page No. 17

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
            body {background-color: khaki;}
            p {background-color: blue;}
        </style>
    </head>
    <body>
        <h1 style="background-color: red">Example Of
                Heading h1</h1>
        <p>Example Of Paragraph</p>
    </body>
</html>
```

2- <u>background-image</u> – It is used to set background image of an HTML element. By default, the background-image property repeats an image both horizontally and vertically so it covers the ~~either~~ entire element. While setting the background image for an HTML element, we should also specify a background color that will be used if the image is not available. We can set this property to ~~none~~ none or url of an image.

<u>Syntax</u>- · body {background-image: url(image/geek.png);}
· body {background-image: url(geek.png);}

<u>Note</u>- image Same location पर नहीं होने पर उसका Path भी देना होगा।

Extramarks
Date ___/___/___
Page No. 18

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</titl>
        <style type="text/cse">
            body{ background-image: url(dog.bmp);
                background-color: red; }
        </style>
    </head>
    <body>
        <h1 style="background-image: url(image/jpg)">
            Exampl of Heading h1 </h1>
        <p>Example of Paragraph</p>
    </body>
</html>
```

3- background-repeat - It specifies whether the background image is repeated or not. We can set this property to repeat, repeat-x, repeat-y or no-repeat. By default, the background-image property repeats an image both ~~horig~~ horizontally and vertically so it covers the entire element.

Syntax- Horizontally - background repeat: repeat-x
Vertical - background repeat: repeat-y
No repeat - background repeat: no-repeat

Ex-
```
body{
    background-image: url(image/guk.jpg);
    background-repeat: repeat-x;
}
```

Extramarks
Date ___/___/___
Page No. 19

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/ccs">
            body{background-image: url(imagi.jpg)
            ;background-repeat: no-repeat;}
        </style>
    </head>
    <body>
        <h1>Example of Heading h1</h1>
        <p>Example of Paragraph</p>
    </body>
</html>
```

4- background-position - It specifies the initial position of a background image. We can set this property to Left top, Center top, right top, center left, center center, center right, left bottom, center bottom, right bottom, x% y%, x-position y-position. ~~bode~~

Syntax- background-position: value;

Note- by default ~~डिफॉल्ट~~ value left top <डि डि

Ex-
```
body{
    background-image: url(imagi/geek.jpg);
    background-repeat: no-repeat;
    background-position: right top;
}
```

Extramarks
Date 20/6/2019
Page No. 20

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style type="text/css">
          body {
                background-image: url (image.jpg);
                background-repeat: no-repeat;
                background-position: right top;
            }
          h1 {
                background-image: url (dog.bmp);
                background-repeat: no-repeat;
                background-position: 50% 50%;
            }
          p {
                background-image: url (image2.jpg);
                background-repeat: no-repeat;
                background-position: x-50 y-70
            }
        </style>
    </head>
    <body>
        <h1> Example of Heading h1 </h1>
        <p> Example of Paragraph </p>
    </body>
</html>
```

5- background-attachment- It specifies whether a background image is fixed or scrolls when the user scrolls the rest of the page. We can set this

Extramarks
Date ___/___/___
Page No. 21

property to either scroll or fixed.

Syntax- background-attachment : value;

Ex- body {
background-image: url(image/geeki.jpg);
background-repeat: no-repeat;
background-position: right top;
background-attachment: fixed;
}

→ <!DOCTYPE html>
<html>
<head>
<title> CSS </title>
<style type="text/ css">
body {
background-image: url(image.jpg);
background-repeat: no-repeat;
background-position: right top;
background- attachment: fixed;
}
</style>
</head>
<body>
 Example of Paragraph
</body>
</html>

Extramarks
Date ___/___/___
Page No. 22

9- background- It works as a shorthand property for all background properties such as background-color, background-image, background-repeat, background-position and background-attachment etc. It sets all these background properties in one declaration.

Syntax- background: value1, value2 value3 ——— ;

Ex- body { background: Khaki url(image/geek.jpg) no-repeat } .

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
            body { background: url(image.jpg) no-repeat
                                right top; }
        </style>
    </head>
    <body>
        <h1>Example of Heading h1</h1>
        <p>Example of Paragraph</p>
    </body>
</html>
```

7- background-size- The background-size property allows to specify the size of background image. The size can be specified in lengths, percentages, or by using one of the two keywords: contain or cover.

- Contain- scales the background image to be as large as posible.

Extramarks
Date ___/___/___
Page No. 23

- cover - scale the background image so that the content area is completely covered by the background image.

Syntax - background-size: value;

Ex.(i)
```
body {
    background-image: url(geek.jpg);
    background-size: 80px;
    background-repeat: no-repeat;
}
```

(ii)
```
body {
    background-image: url(geek.jpg);
    background-size: cover;
    background-repeat: no-repeat;
}
```

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
            body {
                background: url(image.jpg);
                background: contain;
                <!-- background: cover;
                    background: 100px; -->
                background-repeat: no-repeat;
                background-position: center top;
            }
            ~~background~~
        </style>
    </head>
```

Extramarks
Date ___/___/___
Page No. 24

```
    <body>
        <h1>Example of Heading h1</h1>
        <p>Example of Paragraph</p>
    </body>
</html>
```

8- background-origin- The background-origin property specifies where the background image is positioned. We can set this property to border-box, padding-box, content-box.

- borderbox- The ~~border~~ background image starts from the upper left corner of the border.
- paddingbox- The background image starts from the upper left corner of the padding edge. Its default value.
- Content-box- The background image starts from the upper left corner of the content.

Syntax- background-origin: value.
Note- इसका use border बनाने के बाद करते है।

Ex- body {
```
    background: url(geek.jpg);
    background-size: 80px;
    background-repeat: no-repeat;
    background-origin: content-box;
}
```

Note- border बनाने के लिये border का तथा Padding देने के लिये padding का use करते है।

Extramarks
Date ___/___/_____
Page No. 25

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style type="text/css">
          p {
              border: 5px solid black;
              padding: 40 px;
              background: url(image.jpg);
              background-size: 80px;
              background-origin: content-box;
              background-repeat: no-repeat;
          }
        </style>
    </head>
    <body>
        <p> Example of Paragraph. Example Of Paragraph.<br>
            Example of Paragraph. Example of Paragraph<br>
        </p>
    </body>
</html>
```

6- background-clip - The ~~border~~ background-clip property specifies the painting area of the background. We can set this property to border-box, padding-box, content-box.

- border-box - The background is painted to the outside edge of the border. It's default value.

- padding-box - The background is painted to the outside edge of the padding.

Page No. 26

- Content-box - The background is painted within the content box.

Syntax - background-clip: ~~content-box~~ value;

Ex- body {
    border: 5px dotted black;
    padding: 40 px;
    background-color: khaki;
    background-clip : content-box;
}

→ <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style type="text/ccs">
            p{
                border : 5px dotted black;
                padding : 40 px;
                background-color : khaki;
                background-clip : padding-box;
            }
        </style>
    </head>
    <body>

            Example of Paragraph. Example of Paragraph.
            Example of Paragraph. Example of Paragraph.

    </body>
</html>

Extramarks
Date ___/___/___
Page No. 27

Multiple Background Images -

Syntax- ~~tagname~~ selector {
background-image: url(path1), url(path2), ——— ;
background-position: value1, value2, ——— ;
background-size: value1, value2, ——— ;
background-repeat: norepeat, norepeat, ——— ;
}

Ex- body {
background-image: url(geeky.jpg), url(geek.jpg);
background-position: left top, right top;
background-size: 80 px, 60px;
background-repeat: no-repeat, no-repeat;
}

Note- यदि एक से ज्यादा images कि position Same रख दें तो जिस order में images का url लिखा है उसी order में सबसे पहला सबसे ऊपर और फिर उसके नीचे एक के बाद एक बाकी सभी ~~imag~~ images दिखायी देंगे।

Shorthand method-

Syntax- selector { background ~~image~~ : url(~~path~~1) position value1 repeat value1, url (path2) position value2 repeat value2 ——— ; }

Ex- body { background : url(geeky.jpg) left top no-repeat, url(geeky.jpg) right top no-repeat; }

→ <!DOCTYPE html>
<html>
<head>
<title> CSS </title>
<style type = "text/css">

Extramarks
Date ___/___/___
Page No. 28

```
body{
        ~~body~~ background-image: url(dog.bmp), url(image.jpg);
        background-position: left top, right top;
        background-size: 80px, 60px;
        background-repeat: no-repeat, no-repeat
    }
  p{ background: url(image1.jpg) left top no-repeat,
        url(image2.jpg) right top no-repeat;}
</style>
<body>
    <h1>Example of Heading h1</h1>
    <p>Example of Paragraph.<br>
        Example of Paragraph.<br>
        Example of Paragraph.<br>
        Example of Paragraph.<br>
        Example of Paragraph.<br>
        Example of Paragraph
    </p>
  </body>
</html>
```

Full Background Image -

Syntax- html {

```
        background: url(path) no-repeat center fixed;
        background-size: cover;
    }
```

→ <!DOCTYPE html>

```
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
```

Extramarks
Date ___/___/___
Page No. 29

```
html<
        background:url(image.jpg) center fixed
                                  no-repeat;
        background-size: cover;
        }
    </style>
  </head>
  <body>
    <p>Example of Paragraph</p>
  </body>
</html>
```

<u>Text</u> - Text Properties various text formating options.

- color
- direction
- text-align
- letter-spacing
- line-height
- text-decoration
- text-indent
- text-shadow
- text-transform
- vertical-align
- white-space
- word spacing
- text-align-last
- text-overflow
- word-break
- word-wrap

} Newly added in CSS 3

Extramarks
Date ___/___/___
Page No. 30

color- Color property set the color of text. We can set this property either a color name or color value.

- color name- red, green, blue etc.
- Color value-

Hex value - #F0E68C
RGB value - rgb(240,230,140)

Ex-(i) h1 {color: red;}
(ii) p{color: #F0E68C;}

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <Style type="text/cse">
            p{color: red;}
            h1{color: #F0E68C;}
        </style>
    <head>
    <body>
        <p>Example of Paragraph</p>
        <h1>Example of Heading h1</h1>
    </body>
</html>
```

direction- It specifies the text direction. We can set this property, to either ltr (left to right) or rtl (right to left).

Syntax- direction: value.

Note- by default value ltr

Ex- h1 {direction: rtl;}

Extramarks
Date 21 / 6 / 19
Page No. 31

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/cse">
            h1{direction: rtl;}
        </style>
    </head>
    <body>
        <p>Example Of Paragraph</p>
        <h1> Example of Heading h1</h1>
    </body>
</html>
```

text-align - It specifies alignment of text. We can set this property to left, right center or justify.

Syntax- text-align: value

Ex- h1{text-align : right;}

Note- bydefault इसकी value left रहती है।

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/cse">
            h1{text-align: center;}
            p{text-align: right;}
        </style>
    </head>
    <body>
        <h1> Example of Heading h1</h1>
        <p> Example of Paragraph</p>
</head></body>
```

Extramarks
Date ___/___/___
Page No. 32

Letter-spacing - It specifies the space between text characters. We can set this property to either normal or specify it in the form of the distance between two consecutive letters using identifier, such as px (pixels), in (inches), pt (points) and cm (centimeters).

Length - px, in, pt, cm, em.

Syntax - letter-spacing: value;

Ex - h1{ letter-spacing: 3px}

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style type="text/ccs">
            h1{ letter-spacing: 50 px;}
            p{ letter-spacing: 5 px;}
        </style>
    </head>
    <body>
        <h1> Example of Heading h1 </h1>
        <p> Example of Paragraph </p>
    </body>
</html>
```

line-height - It specifies the distance between two line. We can set this property to normal or specify it in the form of length, number or percentage.

Syntax - line-height: value

Ex - p{line-height: 60 %;}

h1{line-height: 50 px;}

Note - value px, in, pt, cm, em

Extramarks
Date ___/___/___
Page No. 33

→ <!DOCTYPE html>
<html>
<head>
<title>CSS</title>
<style type="text/css">
p{ line-height: 100px; }
</style>
</head>
<body>
Example of Paragraph. Example of Paragraph
Example of Paragraph. Example of Paragraph.

</body>
</html>

~~text-direction~~ - ~~It specifies~~

text-decoration - It specifies the ~~direct~~ decoration added to the text. We can set this property to none, underline, overline, line-through.

Syntax- text-decoration: value;

Ex- h1{ text-decoration: underline; }

→ <!DOCTYPE html>
<html>
<head>
<title>CSS</title>
<style type="text/css">
h1{ text-decoration: underline; }
h2{ text-decoration: overline; }
h3{ text-decoration: line-through; }
</style>
</head>

Page No. 34

```
<body>
    <h1> Example of Heading h1 </h1>
    <h2> Example of Heading h2 </h2>
    <h3> Example of Heading h3 </h3>
</body>
</html>
```

text-indent - It specifies the indention of the first line of text in a block. We can specify this property in the form of either length or percentage. Negative values are allowed. The first line will be indented to the left if the value is negative.

Syntax - text-indent : value;

Ex - p{ text-indent : 30px; }

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style type="text/css">
            h1{ text-indent: -30px; }
            p { text-indent : 30px; }
        </style>
    </head>
    <body>
        <h1> Example Of Heading h1 </h1>
        <p> Example of Paragraph. Example of Paragraph.
            Example of Paragraph. Example Of Paragraph. </
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 35

text-shadow - It specifies the shadow effect added to text. We can add more than one shadow on the text, separated by comma.

Syntax- text-shadow : h-shadow v-shadow blur-radius color;

h-shadow - The position of the horizontal shadow. Negative value are allowed.

v-shadow - The position of the vertical shadow. Negative value are allowed.

blur-radius - The blur radius. Default value is 0

color - The color of the shadow

none - No shadow

Note- h-shadow और v-shadow की value देना Cumpalory है।

Ex- h1{ text-shadow: 2px 2px 5px red;}

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
            <style type = " text/ cse " >
                h1{ text-shadow : 3px 3px 8px green;}
                h2 { text-shadow: 2px 2px 5px red, 5px
                                5px 8px blue;}
            </style>
    </head>
    <body>
        <h1> Example of Heading h1</h1>
        <h2> Example of Heading h2</h2>
    </body>
</html>
```

Page No. 36

text-transform - text-transform is used to turn everything into uppercase or lowercase letters, or capitalize the first letter of each word. We can set this property to none, uppercase, lowercase or capitalize.

Syntax - text-transform : value;

Ex- h1{ text-transform : uppercase;}

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style type="text/ css">
            h1{ text-transform : uppercase;}
            h2{ text-transform: lowercase;}
            h3{ text-transform: capitalize;}
        </style>
    </head>
    <body>
        <h1> Example of Heading h1</h1>
        <h2> Example of Heading h2 </h2>
        <h3> Example of Heading h3</h3>
    </body>
</html>
```

~~Syntax~~

Vertical-align - It specifies the vertical positioning of text in the element. We can set this property to baseline, sub, super, top, middle, bottom, text-bottom or specify in the form length or percentage value.

Syntax - vertical-align : value.

Ex- h1{ vertical-align: super;}

Extramarks
Date ___/___/___
Page No. 37

→ <!DOCTYPE html>

```
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
            .sp{vertical-align:super;}
            .sb{vertical-align:sub;}
        </style>
    </head>
    <body>
        <p>(a+b) <span class="sp">2</span></p>
        <p>H<span class="sb">2</span>O</p>
    </body>
</html>
```

Output- $(a+b)^2$
H_2O

Note- tag का use करके हम पूरे paragraph में केवल selected content पर ही CSS का प्रभाव डाल सकते हैं यदि यहाँ tag का use ना करके <p> tag में ही sp या sb class का use करते हैं तो पूरा Paragraph का Content super या sub में हो जायेगा।

White-space- It specifies how white space inside an HTML element is handled. We can set this property to normal, pre, nowrap, pre-line or pre-wrap.

- normal- Sequence of whitespace will collapse into a single whitespace. Text will wrap when ~~nec~~ necessary.

- pre- Whitespace is preserved by the browser. Text

Page No. 38

will only wrap on line breaks. It is just like the <pre> tag in HTML.

- nowrap - Sequence of whitespace will collapse into a single whitespace. Text will never wrap to the text line. The text continues on the same line until a <br> tag is ~~encounted~~ incountered.

- pre-line - Sequence of whitespace will collapse into a single whitespace. Text will wrap when ~~nece~~ necessary, and on line breaks.

- pre-wrap - Whitespace is preserved by the browser. Text will ~~wo~~ wrap when necessary, and on line breaks.

Syntax - white-space: value;
Ex - p{ white-space: nowrap; }

```
→ <!DOCTYPE html>
  <html>
      <head>
          <title>CSS</title>
          <style type="text/ccs">
              p{ white-space: pre-wrap; }
          </style>
      </head>
      <body>
          <p>Example of Paragraph.
              Example of Paragraph. Example of Paragraph
              Example of Paragraph. Example of Paragraph. </p>
  </body>
  </html>
```

Date 22/6/19
Page No. 39

Word-Spacing- The word-spacing property increase or decrease the white space between words. We can set this property to normal or specific in the form of length value.

Syntax- word-spacing: value;

Ex- p{word-spacing: 20px;}

```
→ <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
            h1{ word-spacing: 20px;}
        </style>
    </head>
    <body>
        <h1> Example Of Heading h1 </h1>
    </body>
</html>
```

text-align-last- The text-align-last property specifies how to align the last line of a text. The text-align-last property will only work for elements with the text-align property set to "justify". We can set this property to auto, left, right, center, justify, start, or end.

Syntax- selector{ text align: justify, text-align-last: value;}

Ex- p{ text-align: justify, text-align-last: right;}

Note- last line browser के window के अनुसार तय होगा।

Extramarks
Date ___/___/___
Page No. 40

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
            p{text-align:justify; text-align-last:
                                    center;}
        </style>
    </head>
    <body>
        <p>Example of Paragraph. Example of Paragraph
           Example of Paragraph. Example of Paragraph
           Example of Paragraph. Example of Paragraph
        </p>
    </body>
</html>
```

word-wrap – word-wrap property allows long words to be able to be broken and wrap ~~onto~~ onto the ~~text~~ next line.
We can set this property to either normal or break-word. By default value is ~~break~~ normal.

Syntax- word-wrap: value;
Ex- p { word-wrap: break-word;}

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
            p{
```

```
            width : 180 px
            border:1 px solid black;
            word-wrap: break-word;
        }
    </style>
  </head>
  <body>
    <p> Example of Paragraph. Example of Paragraph
      abcdefghijklmnopqrstuvwxyz Example
      of paragraph. Example of Paragraph.
    </p>
  </body>
</html>
```

<u>word-break</u> - The word-break property specifies line breaking rules for non-CJK scripts. CJK scripts are Chinese, Japanese, and Korean ("CJK") scripts. We can set this property to normal, breakall or keep-all.

- <u>normal</u> - Default value. Break words according to their usual rules.
- <u>break-all</u> - Lines may break between any two letters.
- <u>keep-all</u> - Break are prohibited between pairs of letters.

<u>Syntax</u> - wordbreak : value;

<u>Ex</u> - p{ wordbreak : break-all;}

Page No. 42

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style type="text/ess">
            p {
                width: 180px;
                border: 1px solid black;
                word-break: break all;
            }
        </style>
    </head>
    <body>
        <p> Example of Paragraph. Example of Paragraph.
            Example of Paragraph. Example of Paragraph.
        </p>
    </body>
</html>
```

Font - The Font Properties is used to set font related styles for the text present on an HTML.

1- font-family
2- font-size
3- font-stretch
4- font-style
5- font-variant
6- font-weight
7- font

Extramarks
Date ___/___/___
Page No. 43

1- font-family- We can specify a list of font names for the text contained inside an HTML element, using font-family property.

Syntax- font-family: value1, value2. ——;

Ex- p{ font-family: Arial, serif, "Times New Roman"; }

Note- If the name of a font family is more than one word, it must be in quotation marks.

Ex- "Time New Roman"

Note- हम font-family में more than one value देते हैं ताकि एक value कोई नहीं available हो वर के system पर तो दूसरा use हो सके। यह order के position के अनुसार Priority रखता है।

```
→ <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p{ font-family: Arial, sirif, "Time New
                                Roman"; }
        </style>
    </head>
    <body>
        <p> Example of Paragraph </p>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 44

2- font-size - We can specify a font size in the form of percentage, px, em.
The default text size in browser is 16 px or 1 em.
Other value - xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger.

Syntax- font-size : value
Ex- (i) p{ font-size : 50%; }
(ii) p{ font-size : 16px }
(iii) p{ font-size : 2 em; }
(iv) p{ font-size : xx-small; }

```
→ <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p{ font-size : xx-small; }
        </style>
    </head>
    <body>
        <p> Example of Paragraph. </p>
    </body>
</html>
```

3- font-stretch - It specifies a normal, condensed or extended font face. We can set this property to normal, wider, narrower, ultra-condensed, extra-condensed, condensed, semi-condensed, semi-expanded, expanded, extra-expanded, ultra expanded.

Extramarks
Date ___/___/___
Page No. 45

~~Syntax- It specif font-stretch: value~~
Note- यह सभी browser में support नहीं करता है।
Syntax- ~~font-size: va~~ font-stretch: value;
Ex- p{ font-stretch: condensed;}

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p{ font-stretch: expanded;}
        </style>
    </head>
    <body>
        <p> Example of Paragraph </p>
    </body>
</html>
```

4- font-style- It specifies the style of the font. We can set this property to normal, italic, oblique. By default text is normal.
Syntax- font-style: value;
Ex- p{ font-style: italic;}

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p{font-style: oblique; }
        </style>
    </head>
```

Extramarks
Date ___/___/___
Page No. 46

```
<body>
    <p> Example of Paragraph </p>
</body>
</html>
```

5- font-variant - It specifies whether or not a font is a small-caps font. We can set this property to either normal (by default) or small-caps.

Syntax-- font-variant : value;

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

Note- small-caps में text caps जो generally होता है उससे थोड़ा छोटा caps में बदल जाता है। Ex- AA

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p{ font-variant: small-caps; }
        </style>
    </head>
    <body>
        <p> Example of Paragraph </p>
    </body>
</html>
```

6- font-weight- The font weight property allows to set the font weight of the text present on an HTML Page. We can set this property to normal, bold, bolder, lighter, number (100, 200, 300 upto 900)

Extramarks
Date ___/___/___
Page No. 47

Syntax- font-weigth : value;
Ex- (i) p { font-weight : bold; }
(ii) p { font-weight : 500; }

→ <!DOCTYPE html>
<html>
<head>
<title>CSS</title>
<style>
p { font-weight: bold; }
</style>
</head>
<body>
Example of Paragraph.
</body>
</html>

7- font- It defines a shorthand property for font-style, font-variant, font-weight, font-size, line-height, and font-family ~~property~~. properties.
Syntax- font: value1, value2, - ——— ;

Ex- p { font : bold 20px serif; }

Note- The font-size and font-family values are required. The properties must be set in order : "font-style font-variant font-weight, font-size / line-height font-family". ~~We can so~~ हम चाहें तो required properties को छोड़ कर किसी भी property को skip कर सकते पर order maintain रहना जरूरी है।

Extramarks
Date ___/___/___
Page No. 48

→ <!DOCTYPE html>
<html>
<head>
<title>CSS</title>
<style>
p{ font: bold 20px serif; }
</style>
</head>
<body>
 Example of Paragraph
</body>
</html>

List- List property is used to create list.
There are two type of lists:-

- Unordered Lists- An unordered list use ~~bull~~ bullets as the list item markers.
- Ordered List- An ordered list uses numbers or letters as the list item markers.

1 - list-style-type
2 - list-style-image
3 - list-style-position
4 - list-style

2- list-style-image- In this an image is used as the list item marker. We can set this property to none or URL address of an image.

Syntax- list-style-image: url('Path/name');

Ex- ul{ list-style-image: url('book.png'); }

Extramarks
Date ___/___/___
Page No. 49

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            ul { list-style-image: url('bullet.png'); }
        </style>
    </head>
    <body>
        <ul>
            <li> Linux </li>
            <li> Windows </li>
            <li> Mac </li>
            <li> Android </li>
        </ul>
    </body>
</html>
```

list-style-type - It specifies the appearance of the list item maker if the list-style-image property has the value none or if the image specified through this property cannot be displayed. We can set the property to none, disc, ~~circle~~ circle (default value for unordered list), square, decimal (default value for ordered list), decimal-leading-zero, lower-roman, upper-roman, lower-alpha, upper alpha, lower-~~greek~~ greek, lower-latin, upper-latin, hebrew, armenian, georgian, cjk-ideographic, hiragana, katakana, hiragana-iroha, or katakana-iroha.

Syntax ~~list-style-type:~~ list-style-type: ~~square;~~ value;

Ex- ul { list-style-type: square; }

Extramarks
Date 23 / 6 / 19
Page No. 50

Note- list-style-image की Priority list-style-type से ज्यादा होती है।

```
→ <!DOCTYPE html>
  <html>
      <head>
          <title>CSS</title>
          <style>
              ul {
                  list-style-image: url('bullet.png');
                  list-style-type: square;
              }
              ol { list-style-type: lower-alpha; }
          </style>
      </head>
      <body>
          <ul>
              <li> Linux</li>
              <li> Windows</li>
              <li>Mac</li>
              <li>Android</li>
          </ul>
          <ol>
              <li> Linux</li>
              <li> Windows</li>
              <li> Mac</li>
              <li> Android</li>
          </ol>
      </body>
  </head>
```

Extramarks
Date ___/___/___
Page No. 51

3- list-style-position- It specifies the position of the list item marker. You can set this property to either inside or outside.

Syntax- list-style-position: ~~inside~~ value;

Ex- ul{ list-style-position: inside;}

Note- By default value is outside.

```
→ <!DOCTYPE html>
  <html>
      <head>
          <title> CSS </title>
          <style>
              ol{ list-style-position: inside;}
          </style>
      </head>
      <body>
          <ol>
              <li> Linux </li>
              <li> Windows </li>
          </ol>
          <ul>
              <li> Mac </li>
              <li> Android </li>
          </ul>
      </body>
  </html>
```

4- list-style- It is a shorthand property for list-style-type, list-style-image and list-style-position.

and- list-style-type, list-style-position, list-style-image.

Syntax- list-style: value1, value2, value3.

Ex- ul{ list-style: square inside;}

Page No. 52

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            ul { list-style: square inside; }
        </style>
    </head>
    <body>
        <ul>
            <li> Linux </li>
            <li> Windows </li>
            <li> Mac </li>
            <li> Android </li>
        </ul>
    </body>
</html>
```

Border-

1- border-style- This property sets border style of an element. We can set this property to none, hidden, dotted, dashed, solid double, groove, ridge, inset, outset.

1 value- Syntax- border-style: value;

Ex- p { border-style: dashed; }

Note- We can set different-different border-style for all side.

4 Value- Syntax- border-style: top value, right value, bottom value, left value

Ex- p { border-style: dotted solid double dashed; }

3 Value- Syntax- border-style: top value, right & left value, bottom value;

Ex- p { border: style: dotted solid dashed; }

Extramarks
Date ___/___/___
Page No. 53

2 values- Syntax- border-style : top & bottom value, left & right value;
Ex- p{ border-style: dotted solid;}

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            p{ boorder-style: solid;}
            h1{ border-style: dotted groove dashed outset;}
            h2{ border-style: ridge double inset;}
            h3{ border-style: dotted double;}
        </style>
    </head>
    <body>
        <h1> Example of Heading h1</h1>
        <h2> Example of Heading h2</h2>
        <h3> Example of Heading h3</h3>
        <p> Example of Paragraph </p>
    </body>
</html>
```

2- border-width- It sets width of element's border. We can set this property to thin, medium, thick, length in the form of px.

Syntax- ~~Selectorborder~~ Selector{ border-style: value; border-width: value;}
Ex- (i) p{ border-style: solid; border-width: thick;}
(ii) p{ border-style: solid; border-width: 5px;}

4 values- Syntax- border-width: top size, right size, bottom size, left-size;
Ex- p{ border-width: thick medium thin 5px;}

Page No. 54

3 Values- Syntax- border-width: top size right & ~~bottom~~ left size bottom size
Ex- p { border-width : thick medium 5px; }

2 values- Syntax- border-width: top & bottom size right & left size
Ex- p { border-width : thick thin; }

1 value- Syntax- border-width : all side size;
Ex- p { border-width : 5px; }

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/ess">
            h1{ border-style: dotted groove dashed outset
                border-width: thinck medium thin 5px
            h2{ border-style: double ridge inset;
                border-width: 5px medium thin;}
            h3{ border-style: dotted double ;
                border-width thinck 5px;}
            h4{ border-style: dotted ;
                border-width: 10px thick medium thin;}
            p{ border-style: solid ; border-width: 5px;}
        </style>
    </head>
    <body>
        <h1>Example of Heading h1</h1>
        <h2>Example of Heading h2</h2>
        <h3>Example of Heading h3</h3>
        <h4>Example of Heading h4</h4>
        <p>Example of Paragraph</p>
    </body>
</html>
```

3- border-color- It sets color of border. We can set this property to transparent and color value. By default value is black.

Syntax- selector { border-style: value ; border-color: value;}

Ex. (i) p{ border-style: solid; border-color: blue; }

(ii) p{ border-style: solid; border-color: #5A4218; }

4 value- Syntax- border-color: topcolor rightcolor bottomcolor leftcolor;

Ex- p{ border-color: blue red black green; }

3 value- Syntax- border-color: topcolor right & left color bottom color;

Ex- p{ border-color: blue green orange; }

2 value- Syntax- border-color: top & bottom color left & right color; }

Ex- p{ border-color: green orange; }

1 value- Syntax- border-color: all sides color

Ex- p{ border-color: blue; }

```
→ <!DOCTYPE html>
<html>
<head>
    <title> CSS </title>
    <style>
        h1{
            border-style: dotted groove dashed outset;
            border-color: blue red green yellow;
        }
        h2{
            border-style: ridge double inset;
            border-color: blue green yellow;
        h3{ border-style: dotted double;
            border-color: blue red;
        }
```

Page No. 56

```
h4{
    border-style : dotted;
    border-color: blue red green yellow;
    border-width : 15 px;
}
p{
    border-style : solid ;
    border-color : blue;
</style>
<body>
    <h1>Example Of Heading h1</h1>
    <h2> Example Of Heading h2 </h2>
    <h3> Example of Heading h3 </h3>
    <h4> Example of Heading h4 </h4>
    <p> Example Of Paragraph. </p>
</body>
</html>
```

4- <u>border</u>- It is shorthand of all border properties. The order should be: border-width, border-style and border-color.

<u>Syntax</u>- border: widthvalue stylevalue colorvalue;

Ex- (i) p{ border: 10px solid blue; }

(ii) p{ border: solid blue; }

<u>Note</u>- ~~border~~ border-style लिखना required है।

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            h1{ border: 5 px solid blue; }
        </style>
    </head>
```

Extramarks
Date ___/___/___
Page No. 57

```
        <body>
            <h1> Example of Heading h1 </h1>
        </body>
    </html>
```

Table border- For specifying table we use border property.

Ex- table, th, td { border: 2px solid blue; }

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/ccs">
            table, th { border: 2px solid blue; }
            td { border: 5px dotted red; }
        </style>
    </head>
    <body>
        <table>
            <th>Name</th>
            <th>Roll</th>
            <th>Result</th>
            <tr>
                <td>Rahul</td>
                <td>101</td>
                <td>Pass</td>
            </tr>
        </table>
    </body>
</html>
```

Page No. 58

border-collapse - This property sets whether the table borders are collapsed into a single border or detached. We can set this property to separate (default) and collapse.

Syntax - table { border-collapse: value; }

Ex- table { border-collapse: collapse; }

Note- इसे table selector के साथ ही use करना है।

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style type="text/Css">
            table, th, td { border: 5px solid red; }
            table { border-collapse: collapse; }
        </style>
    </head>
    <body>
        <table>
            <th> Name </th>
            <th> Roll </th>
            <th> Result </th>
            <tr>
                <td> Rahul </td>
                <td> 101 </td>
                <td> Pass </td>
            </tr>
        </table>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 59

border-spacing- This property is used to set the distance between the borders of adjacent cells. This will work only when border-collapse is separated. We can set this property to horizontal spacing and vertical spacing in the form of cm, px.

Syntax- table{border-collapse : separate; border-spacing : horizontal space vertical space;}

Ex- table{ border-collapse : separate; border-spacing : 5px 10px;}

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
            table, th, td{ border: 5px solid blue;}
            table{
                border-collapse: separate;
                border-spacing: 50px 10px;
            }
        </style>
    </head>
    <body>
        <table>
            <th>Name</th>
            <th>Roll</th>
            <th>Result</th>
            <tr>
                <td>Rahul</td>
                <td>101</td>
                <td>Pass</td>
        </table>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 60

caption-side- This property is used to place a caption for table. We can set this property to top (default) or bottom.

Syntax- caption-side: bottom;

Ex- caption { caption-side: bottom; }

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            table, th, td { border: 5px solid blue; }
            caption { caption-side: bottom; }
        </style>
    </head>
    <body>
        <table>
            <caption> Student Details </caption>
            <th> Name </th>
            <th> Roll </th>
            <th> Result </th>
            <tr>
                <td> Rani </td>
                <td> 102 </td>
                <td> Pass </td>
            </tr>
        </table>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 61

empty-cells - This property is used to set whether or not to display border and background on empty cells in a table. It will work only if the border-collapse is set to separate. We can set this property to show (default) or hide.

Syntax - empty-cell: value;

Ex - table{ empty-cells: hide;}

```
→ <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            table,th, td {border: 5px solid blue;}
            table{ empty-cells: hide;}
        </style>
    </head>
    <body>
        <table>
            <th>Name</th>
            <th> Roll</th>
            <th> Result</th>
            <tr>
                <td>Rahul</td>
                <td> Pass</td>
            </tr>
        </table>
    </body>
</html>
```

Page No. 62

table-layout - This property is used to set table layout. We can set this property either auto (default) or fixed.

Syntax - table-layout: value;

Ex - table{ table-layout: fixed;}

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            table {
                border-collapse : separate;
                width: 100%;
                border : 1px solid black;
                table-layout: fixed;
            }
            td {border: 1px solid black; }
        </style>
    </head>
    <body>
        <table>
            <tr>
                <td width="10%" >Rahul Kumar Sharma</td>
                <td width="90%" >Lalpur Ranchi</td>
            </tr>
        </table>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 63

How to set Table width and Height - We use width and height property to set table width and height. We can also set table heading (th) width and height.

~~Syntax~~ (i) table{width: 100;}
Ex - (ii) th{height: 10px;}

```
→ <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            table, th, td{
                border: 5px solid red;
                border-collapse: collapse;
            }
            table{width: 100%;}
            th{height: 50px;}
        </style>
    </head>
    <body>
        <table>
            <th>Name</th>
            <th>Roll</th>
            <th>Result</th>
            <tr>
                <td>Rahul</td>
                <td>101</td>
                <td>Pass</td>
            </tr>
        </table>
    </body>
</html>
```

Page No. 64

How to set Table Padding - This property is used to control the space between the border and the content in a table, this is used on <td> and <th> elements.

Syntax- padding: value;

Ex- th, td { padding: 10 px; }

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            table, th, td {
                border: 5px solid red;
                border-collapse: collapse;
            }
            td, th { padding: 10px; }
        </style>
    </head>
    <body>
        <table>
            <th>Name</th>
            <th>Roll</th>
            <th>Result</th>
            <tr>
                <td>Rahul</td>
                <td>102</td>
                <td>Pass</td>
            <tr>
        </table>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 65

How to set Table color - We use background-color property for this.
~~Syntax~~ Ex- table { background-color: pink; }

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            table, th, td {
                border: 5px solid red;
                border-collapse: collapse;
            }
            table { background-color: pink; }
            th { background-color: yellow }
        </style>
    </head>
    <body>
        <table>
            <th> Name </th>
            <th> Roll </th>
            <th> Result </th>
            <tr>
                <td> Rahul </td>
                <td> 101 </td>
                <td> Pass </td>
            </tr>
        </table>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 66

Display- This property is used to define how ~~to~~ an element should display. Every HTML element has a default display value depending on ~~with~~ what type of element it is.

- The default display value for most elements is block or inline.

- inline
- block
- flex
- inline-block
- inline-flex
- inline-table
- list-item
- run-in
- table
- table-row-group
- table-caption
- table-column-group
- table-header-group
- table-footer-group
- table-cell
- table-column
- table-row
- none

Block-level Elements- A Block-level element always starts on a new line and takes up the full width available.

Ex-
<h1>
<p>

Extramarks
Date __/__/__
Page No. 67

Inline Element- An inline element does not start on a new line and only take up as much width as necessary.

Ex-
<a>
<img>

→ <!DOCTYPE html>
<html>
<head>
<title>CSS</title>
<style>
p { border: 1px solid blue; }
a { border: 1px solid red; }
</style>
</head>
<body>
Example of Paragraph
<a href="#">Home</a>
</body>
</html>

Inline- When we set this value, the element does not start on a new line and only takes up as much width as necessary (we can't set width / height it won't work)

block- When we set this value, element always starts on a new line and takes up the full width available (we can set width / height)

Extramarks
Date ___/___/___
Page No. 68

inline-block - It is combination of inline and block value. It doesn't start on new line but we can set width and height.

none - The element will not be displayed at all (has no effect on layout)

Syntax- display : value.
Ex- p { display: inline; }

```
→ <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p {
                border:1px solid blue;
                display : inline ;
            }
            div {
                width: 500px;
                background-color: pink;
                display: inline-block;
            }
            a {
                width: 100px;
                background-color: pink;
                display: block;
            }
            h1 { display: none; }
        </style>
    </head>
```

Extramarks
Date ___/___/___
Page No. 69

```
<body>
    <h1> Example of Heading h1</h1>
    <p> Example of Paragraph</p>
    <div> Example of div. </div>
    <a  href="#" >Home</a>
</body>
</html>
```

Visibility - This property is used to specify whether or not an element is visible. We can set this property to visible (default), hidden or collapse.

- hidden - The element is invisible not removed.
- collapse - This value is only used for table elements. collapse removes a row or column, but it does not affect the table layout. The space taken up by the row or column will be available for other content. If collapse is used on other elements, it will be treated as "hidden".

Syntax - visibility: value;

Ex - p{ visibility: hidden;}

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p.p2 { visibility: hidden; }
            table, th, td { border: 5 px solid red;}
            tr. row { visibility: collapse; }
        </style>
    </head>
```

Extramarks
Date ___/___/___
Page No. 70

```
<body>
    <p class="p1">1. Example Of Paragraph</p>
    <p class="p2">2. Example Of Paragraph</p>
    <p class="p3">3. Example Of Paragraph</p>
    <table>
        <th> Name</th>
        <th>Roll</th>
        <th>Result</th>
        <tr>
            <td>Rahul</td>
            <td>101</td>
            <td>Pass</td>
        </tr>
        <tr class="row">
            <td>Rani</td>
            <td>102</td>
            <td>Pass</td>
        <tr>
            <td>Ram</td>
            <td>103</td>
            <td>Fail</td>
        </tr>
    </table>
</body>
</html>
```

display: none vs Visibility: hidden -

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
```

Extramarks
Date ___/___/___
Page No. 71

```
<style>
    p.p2 {visibility: hidden;}
    p.p4 {display: none;}
</style>
<head>
<body>
    <p class="p1">1. Example of Paragraph</p>
    <p class="p2">2. Example of Paragraph</p>
    <p class="p3">3. Example of Paragraph</p>
    <p class="p4">4. Example of Paragraph</p>
    <p class="p5">5. Example Of Paragraph</p>
</body>
</html>
```

Link-

- a:link - a normal, unvisited link.
- a:visited - a link ~~when~~ the user has visited.
- a:hover - a link when the user mouse over it.
- a:active - a link the moment it is clicked.

Ex-
```
a:link {
    color: red;
}
```

Order- a:hover MUST come after a:link and a:visited
a:active MUST come after a:hover

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            a:link { color: red; }
```

Page No. 72

```
a:visited { color: green; }
a:hover { color: yellow; }
a:active { color: blue; }
</style>
<head>
<body>
<a href="#" > Click Me </a>
</body>
</html>
```

Syntax-

- left
- right
- top
- bottom

Auto (Default value)
Length (cm, px, %)

left: value;
right: value
bottom: value
top: value

Ex- left: auto; right: auto; top: auto; bottom: auto;
left: 55px; right: 55px; top: 23px; bottom: 23 cm;
left: 50%; right: 50%; top: 40%; bottom: 40%;
left: 14cm; right: 30cm; top: 25cm; bottom: 30 px;

Note- These properties are used to position an element. We can not use any of these properties without setting up position property. हम left के साथ right और top के साथ bottom का use एक साथ नहीं कर सकते हैं।

→
```
<!DOCTYPE html>
<html>
<head>
<title> CSS </title>
<style>
img {
```

```
        position: fixed;
        right : 50px;
        bottom : 20px;
    }
  ~~static~~ </style>
  </head>
  <body>
        <img  src="icon.png" ></img>
        <p> Example of Paragraph. ____
                                  </p>
  </body>
</html>
```

Position-

1. static (default value)
2. fixed
3. relative
4. absolute

Syntax position: value;

1. static- When we set static for an element then it is not positioned in any special way. it is always positioned according to the normal flow of the page.
left, right, top and bottom property won't work with static position.

~~Syntax~~ Ex.

```
h1 {
        position: static;
        border: 5px solid red;
    }
h1 {    position: static;
        border 5px solid red;
        left : 50px;
    }
```

Page No. 74

2. fixed - When we set fixed position for an element it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element.

Ex- img {
position: fixed
right: 0;
top: 50%
border: 3px solid red;
}

3. relative - When we set relative position for an element, elements positioned to its normal flow.

Ex- h1 {
position: relative;
left: 50px;
border: 3px solid red;
}

4. absolute - When we set absolute position for an element, element is positioned to the nearest positioned ancestor. If there is no positioned ~~absolute~~ ancestor then it follow the normal position according to browser.

Ex- div {
position: relative;
width: 400px;
height: 200px;
border: 3px solid red;
}

Extramarks
Date / /
Page No. 75

```
div {
        position : absolute;
        top: 20px;
        right : 0;
        width : 200px;
        height : 100px;
        border : 3px solid red;
    }
```

1.2. →

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </ title>
        ~~<head~~ <style>
            img {
                position : static or fixed;
                right : 50px ;
                bottom : 20px ;
            }
        </ style>
    </head>
    <body>
        <img  src = " icon.png " > </ img >
        <p > Example of Paragraph ——— </p>
    </body>
</head>
```

3.4. →

```
<! DOCTYPE  html>
<html>
    <head>
        <title > CSS  </ title >
```

Date ___/___/___
Page No. 76

```
<style>
    div {
        position: ~~relative or absolute~~ fixed;
        width: 400px;
        height: 200px;
        background-color: red;
    h1 {
        position: relative or absolute;
        top: 50px
    }
</style>
<head>
<body>
    <div>
        <h1> Example of Heading </h1>
    </div>
    <p> Example of Paragraph ~~</h1>~~ ————————
        ————————————————— </p>
</body>
</html>
```

How to set element Layer (z-index) -

z-index- z-index property is used to set the layer of elements. We can set this property to auto (default) and number. It works only with position property value fixed, relative, absolute.
The element which has higher number appear at the top.

Syntax- z-index: value;

Extramarks
Date 25/6/19
Page No. 77

Ex- img {
position: fixed;
z-index: 585;
}

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div {
                position: fixed;
                width: 400px;
                height: 300px;
                background-color: red;
            }
            img { position: relative; }
            h1 {
                position: relative;
                top: 100 px;
                z-index: 24;
                color: blue;
        </style>
    </head>
    <body>
        <div> </div>
        <h1> Example of heading h1 </h1>
        <img src="icon.png"> </ img>
        <p> Example of Paragraph. _____
            _____ </p>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 78

Overflow- This property is used to define what happens if content overflows an element's box. It means it will define whether to clip content or to add scrollbars when an element's content is too big to fit in a specified area. This property only works for block elements with a specified height. We can set this property to visible (default), scroll, hidden, auto.

Syntax- overflow: value;

Ex- p{ overflow: hidden; }

Note- scroll से horizontal और vertical scroll देता है चाहे जरूरत हो या ना हो पर auto जरूरत होने पर ही देगा। hidden से जो नहीं fit हो रहा उस content को hide कर देगा।

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            p{
                height: 200px;
                width: 400px;
                border: 3px solid red;
                overflow: auto;
            }
        </style>
    </head>
    <body>
```

Extramarks
Date ___/___/___
Page No. 79

```
<p> Start
        Example of Paragraph.
                End. </p>
    </body>
</html>
```

float- This property is used to specify that an element should be placed along the left or right side of its container, where text and inline elements will wrap around it. We can set this property to left, right, none (default).

Syntax- float: value

```
img{ float: left; }
```

margin-right & **margin-left** - right और left में margin set करने के लिये

Syntax margin-right: value; margin-left: value;

Ex- margin-right: 10px; margin-left: 1px;

clear- This property is used to specify whether an element can be next to floating elements that precede it or must be moved down (cleared) below them. We can set this property to left, right, both, none (default).

Syntax- clear: value;

Ex- p{ clear: left; }

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS </title>
```

Page No. 80

```
<style>
    #img1{
        float: left;
        margin-right: 10px;
    }
    #img2{
        float: right;
        margin-left: 10px;
    }
    p{ clear: both; }
</style>
</head>
<body>
    <img id="img1" src="image1.jpg"></img>
    <img id="img2" src="image2.jpg"></img>
    <h1>
        Example of Heading h1.
    </h1>
    <p>
        Example of Paragraph.
    </p>
</body>
</html>
```

div tag - The div tag is used ~~is used~~ to group various other HTML elements. It is a block level element. We can say it also create a block. Syntax - ———

Ex- This is div tag

```
<div>
    <h> Heading </h2>
    <p> Paragraph </p>
</div>
```

Extramarks
Date ___/___/___
Page No. 81

span tag- The span tag is used to group inline elements. It is an inline element.

Syntax- ———

Ex- I am example

→ <!DOCTYPE html>
<html>
<head>
<title> CSS </title>
<style>
div { color: red; }
span { color: green; }
</style>
</head>
<body>

<h1> Example of Heading h1. </h1>
 Example of ~~para~~ Paragraph. .

</body>
</html>

Selector-

Example of Element ~~Sel~~ Selector-

<!DOCTYPE html>
<html>
<head>
<title> CSS </title>

Page No. 82

```
        <style>
            h1 { color: red; }
        </style>
    </head>
    <body>
        <h1> Example of Heading h1 </h1>
    </body>
</html>
```

Example of Id Selector-

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            #divgeek #geek { color: red; }
            #divgeek { color: green; }
            #blue { color: blue; }
        </style>
    </head>
    <body>
        <div id="divgeek">
            <p id="geek"> Example of First Paragraph. </p>
            <p> Example Of Second Paragraph. </p>
            <h1> Example Of Heading h1 </h1>
        </div>
    </body>
</html>
```

इसे और भी nested child के साथ यूज कर सकते हैं और इसकी Priority सबसे height होगी।

Note- div tag के Child tag <p id="geek"> का color red ही रहेगा चाहे जो हो जाये।

Extramarks
Date / /
Page No. 83

Example Of Class Selector -

```
→ <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            .red{ color: red;}
            h2.red{ color: blue; }
            .look{ background-color: blue; }
        </style>
    </head>
    <body>
        <div>
            <p class="red">Example Of First Paragraph.</p>
            <p> Example Of Second Paragraph.</p>
            <h1 class="red look" > Example Of Heading h1</h1>
            <h2 class="red" >Example Of Heading h2 </h2>
        </div>
    </body>
</html>
```

Conflicts - जो सब से last में property लिखते है वही apply होती है और यदि किसी element के लिये कोई property specified नहीं है तो वही Apply होगा।

```
→ <!DOCTYPE html>
<html>
    <head>
```

Extramarks
Date ___/___/___
Page No. 84

```
<title>CSS</title>
<style>
    #maindiv p{color:blue;}
    p{color:red;}
    p{color:green;}
</style>
</head>
<body>
    <p> Example Of ~~Heading~~ Paragraph.</p>
    <div id="maindiv">
        <h1>Example Of Heading h1</h1>
        <p>Example Of Paragraph</p>
    </div>
</body>
</html>
```

Inheritance- Parant tag ~~class~~ की property child tag ~~class~~ पर भी Apply होता है पर कुछ tags like <a></a> tag पर यह apply नहीं होता है

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            #divgeek{color:green;}
        </style>
    </head>
    <body>
        <div id="divgeek">
            <p>Example of Paragraph.</p>
            <h1>Example Of Heading h1</h1>
```

Extramarks
Date ___/___/___
Page No. 85

```
            <a  href="#">Click Here</a>
        </div>
    </body>
</html>
```

Descendant Class Selector-

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            #maindiv .mainpara {color: red;}
            #maindiv #childdiv . childpara{color: yellow;}
            #maindiv #childdiv .gdiv p{color: blue;}
            p{color: green;}
        </style>
    </head>
    <body>
        <div id="maindiv">
            <p class="mainpara">I am para 1 and child
                                of main div</p>
            <div id="childdiv">
                <div class="gdiv">
                    <p>Example of Paragraph</p>
                    <h2> Example Of Heading</h2>
                </div>
                <p class="childpara">This is 1st para
                    and child of child div</p>
                <h1> This is heading 1 and child of
                    child div</h1>
            </div>
```

Page No. 86

```
        <h2>I am heading 2 and child of main div</h2>
        <p>I am para2 and child of main div</p>
    </div>
  </body>
</html>
```

Selector Priority -

Id Selector > Class Selector > Element Selector

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            #maindiv {color: red;}
            p{color: blue;}
            ~~#maindiv p{color: red;}~~
            ~~h2~~.
            #mindiv h2{ color: red;}
            h2{color: blue;}
            .cl{color: green;}
            h1{color: khaki;}
            span{color: yellow;}
        </style>
    </head>
    <body>
        <div id="maindiv">
            <p>Example Of <span>Paragraph</span>
            .</p>
        </div>
        <div id="mindiv">
```

```
        <h2>Example Of Heading h2.</h2>
    </div>
    <h1 class="cl">Example of Heading h1.</h1>
  </body>
</html>
```

Important declaration- इसकी priority सब से ~~high~~ अधिक होती है Id Selector से भी high.

→
```
<!DOCTYPE html>
<html>
  <head>
    <title> CSS</title>
      <style>
        #maindiv .cl { color: red; }
          p { color: green !important; }
      </style>
  </head>
  <body>
      <div id="maindiv">
        <p class="cl> This is para</p>
      </div>
  </body>
</html>
```

Universal Selector (*) - इसका use करने पर सभी element जो उस webpage में है उस पर हम कोई भी CSS property apply कर सकते है चाहे वो <a> tag हि क्यूं न हो।

Syntax- * { property1: value1; property2: value2; ——— ; }

Extramarks
Date 26 / 6 / 19
Page No. 88

```
→ <!DOCTYPE html>
  <html>
      <head>
          <title>CSS</title>
          <style>
              *{color:red;}
          </style>
      </head>
      <body>
          <div id="maindiv">
              <h1>Main Heading</h1>
              <p>Main Paragraph</p>
              <div>
                  <h1>Child Heading</h1>
                  <p>Child Paragraph</p>
              </div>
              <a href="#">Click Here</a>
          </div>
      </body>
  </html>
```

Child Selector(>) - ये केवल direct child पर ही work करता है।

Syntax- #Father > child { property1 : value1 ; property2 : value2 ; ~

```
→ <!DOCTYPE html>
  <html>
      <head>
          <title>CSS</title>
          <style>
              #maindiv > p {color:yellow;}
          </style>
      </head>
```

Extramarks
Date ___
Page No. 89

```
<body>
    <div id="maindiv">
        <h1>Main Heading</h1>
        <p> Main Paragraph </p>
        <div>
            <h1> Child Heading</h1>
            <p> Child Paragraph</p>
        </div>
        <a href="#">Click Here</p>
    </div>
</body>
</html>
```

Adjacent Selector(+) - किसी tag के Adjacent (ठीक बाद में) tag पर work करता है।

Syntax tag-name + Adjacent tag name {property: 'value'; ---;}

```
<!DOCTYPE html>
<html>
    <head>
        <titl> CSS</title>
        <style>
            #maindiv h1 + p{ color: yellow;}
        </style>
    </head>
    <body>
        <div id="maindiv">
            <h1>Main Heading</h1>
            <p>Main Paragraph</p>
            <div>
                <h1> Child Heading</h1>
                <p> Child Paragraph</p>
            </div>
```

Extramarks
Date ___/___/___
Page No. 90

```
        <p>Example of Paragraph </p>
        <h1>Example Of Heading </h1>
    </div>
    <h3> Example Of Heading h3 </h3>
  </body>
</html>
```

Sibling Selector (~) - किसी tag के sibling tag पर property apply करने के use करते हैं। किसी parent tag के सभी child आपस में siblings होते हैं।

Syntax - tag-name ~ sibling tag-name { property : value ; ---- }

```
→ <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            h1 ~ p { color: red; }
        </style>
    </head>
    <body>
        <div>
            <p> Example of Paragraph </p>
        </div>
        <h1> Example of Heading h1. </h1>
        <p> Example of Paragraph. </p>
    </body>
</html>
```

Pseudo-class - A pseudo-class is used to define a special state of an element.

Extramarks
Date ___/___/___
Page No. 91

Syntax- selector: pseudo-class { property: value; }
Ex- div: hover { background-color: red; }
Note- हम चाहे तो div (selector) के स्थान पर id या class का भी use कर सकते हैं

Some important Pseudo classes-

:link
:visited
:hover
:active
:focus
:checked
:first-child
:first-of-type
:nth-child(n)

first-child - This will set CSS rule to every first element of selector which is someone's child.

last-child- This will set CSS rule to every last element of selector which is someone's child.

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            p: first-child { color: red; }
            p: last-child { color: blue; }
            div p: first-child { color: yellow; }
            div p: last-child { color: green; }
        </style>
```

Date ___/___/___
Page No. 92

```
</head>
<body>
    <p> Example Of First Paragraph.
    <div>
        <p> Example Of First Paragraph.</p>
        <p> Example Of Second Paragraph.</p>
        <p> Example Of Last Paragraph.</p>
    </div>
    <hr>
    <div>
        <p> Example Of First Paragraph.</p>
        <p> Example Of Second Paragraph.</p>
        <p> Example Of Last Paragraph.</p>
    </div>
    <hr>
    <p> Example Of Second Paragraph.</p>
    <p> Example Of Last Paragraph.</p>
</body>
</html>
```

first-of-type - This will set CSS rule to every first selector type of element.

last-of-type - This will set CSS rule to every last selector type of element.

```
--> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p: first-of-type {color: red;}
            p: last-of-type {color: blue;}
```

Extramarks
Date ___/___/___
Page No. 93

```
div p:first-of-type{color:yellow;}
div p:last-of-type{color:green;}
</select>
<head>
<body>
    <h1>Example of Heading</h1>
    <p>Example of Paragraph</p>
    <div>
        <h1>Example Of Heading</h1>
        <p>Example of First Paragraph</p>
        <p>Example of Last Paragraph</p>
    </div>
    <hr>
    <div>
        <p>Example Of First Paragraph</p>
        <p>Example Of Last Paragraph</p>
        <h1>Example Of Heading h1</h1>
    </div>
    <hr>
    <div>
        <p>Example Of First Paragraph.</p>
        <h1>Example Of Heading h1</h1>
        <p>Example Of Last Paragraph</p>
    </div>
    <hr>
    <p>Example of Last Paragraph.</p>
    <h1>Example Of Heading h1</h1>
</body>
</html>
```

nth-child(n) – इसका use करके हम nth element में CSS apply कर सकते हैं। Parameter में integer, odd, even और कोई function (like 3n+1) (where n=0 ... till) ले सकता है।

Extramarks
Date ___/___/___
Page No. 94

```
→ <!DOCTYPE html>
  <html>
      <head>
          <title>CSS</ /title>
          <style >
              li: nth-child(4) { color: red; }
              li :nth-child (3), li: nth-child (8) { color: blue
          </style>
      </head>
      <body>
          <ul>
              <li> Item 1 </li>
              <li> Item 2 </li>
              <li> Item 3 </li>
              <li> Item 4 </li>
              <li> Item 5 </li>
              <li> Item 6 </li>
              <li> Item 7 </li>
              <li> Item 8 </li>
              <li> Item 9 </li>
          </ul>
      </body>
  </html>

→ <!DOCTYPE html>
  <html>
      <head>
          <title> CSS </ /title>
          <style>
              li:nth-child (even) { color: red; }
              li:nth-child (odd) { color: blue; }
          </style>
      </head>
```

Extramarks
Date ___/___/___
Page No. 95

```
<body>
    <ul>
        <li> Item 1 </li>
        <li> Item 2 </li>
        <li> Item 3 </li>
        <li> Item 4 </li>
        <li> Item 5 </li>
        <li> Item 6 </li>
        <li> Item 7 </li>
        <li> Item 8 </li>
        <li> Item 9 </li>
    </ul>
</body>
</html>
```

→

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            li:nth-child(3n+2) {color: red;}
        </style>
    </head>
    <body>
        <ul>
            <li> Item 1 </li>
            <li> Item 2 </li>
            <li> Item 3 </li>
            <li> Item 4 </li>
            <li> Item 5 </li>
            <li> Item 6 </li>
            <li> Item 7 </li>
```

Extramarks
Date ___/___/___
Page No. 96

```
            <li> Item 8 </li>
            <li> Item 9 </li>
        </ul>
    </body>
</html>
```

Content - It is used to generate content at run time dynamically with the :: before and :: after pseudo-elements, to insert generated content.

Ex- p::before{ content: "Geekyshows Says"; }

Syntax- ~~Selector:: pseudo~~ (Note- it run time par dynamically create hote hai)

Selector:: pseudo-element { content: value; }

value-

- normal (default)
- none
- counter
- attr(attribute)
- string
- open-quote
- close-quote
- no-open-quote
- no-close-quote
- url(url)

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
    ~~</head>~~ <style>
        p::before{ content: "GeekyShows" }
        p::after{ content: " End." }
```

```
        p::before{content:open-quote}
        p::after{content:close-quote}
        h1::after{content:url(bullet.png)}
    </style>
  </head>
  <body>
      <p>
          Example Of Paragraph.
      </p>
      <p>
          Example Of Paragraph.
      </p>
      <h1>Example of Heading h1</h1>
  </body>
</html>
```

Pseudo Element - This is used to style specified parts of an element.

Syntax- selector::pseudo-element{property:value;}

Ex- p::first-letter{color:red;}

Pseudo Elements-

1- ::first-letter
2- ::first-line
3- ::selection
4- ::before
5- ::after

Note- CSS3 से पहले Pseudo classes और Pseudo elements को use करने के लिये : का use करते थे पर CSS3 से हम Pseudo classes के लिये : का और Pseudo elements के लिये :: का use करते हैं।

Extramarks
Date 27 / 6 / 19
Page No. 98

1- :: first-letter- This is used to add a style to the first letter of the specified selector (each.). This can only be used with block-level elements.
property use with ::first-letter-

- background properties
- font properties
- padding properties
- color properties
- margin properties
- border properties
- line-height
- text-transform
- text-decoration
- vertical-align (only if float is 'none')
- float
- clear

Ex- p::first{color:red;}

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p::first-letter{ color: red;}
        </style>
    </head>
    <body>
        <p>Example Of Paragraph </p>
        <h1> Example Of Heading h1 </h1>
        <div>
```

```
        <p>Example Of Paragraph.</p>
        <h1>Example of Heading h1</h1>
    </div>
  </body>
</html>
```

2- ::first-line - This is used to add a style to the first line of the specified selector (each). This can only be used with block-level elements.

properties used with ::first-line -

- background properties
- font properties
- text-decoration
- color properties
- word-properties
- letter-spacing
- text-transform
- vertical-align
- line-height
- clear

Ex- p::first-line { color: red; }

->
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            p::first-line { color: red; }
        </style>
    </head>
```

Extramarks
Date ___/___/___
Page No. 100

```
<body>
    <p> Example of Paragraph.
    </p>
    <p> Example Of Paragraph.
    </p>
</body>
</html>
```

3- :: selection - This is used to matche the portion of an element that is selected by a user then apply the CSS rule in the selected content.

properties use with :: selection -

- background properties
- color properties
- cursor properties
- outline properties

Ex- p::selection {background-color: red;}

Note - bydefault background-color: blue and ~~text~~-color: white रहता है। :: selection {property : value;} लिखने से property सबके लिये apply हो जायेगा। ये सभी browsers support नहीं करते।

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            p::selection {color: red;}
        </style>
    </head>
```

Extramarks
Date: ___/___/___
Page No. 101

```
<body>
    <p> Example of Paragraph </p>
    <h1> Example of Heading h1 </h1>
    <p> Example Of Paragraph </p>
</body>
</html>
```

4- **::before**- This is used to insert something before the content of each specified selector (each). We use content property to specify the content to insert.

Ex- p::before { content: "Geekyshows"; }

5- **::after** This is use to insert something after the content of each specified selector (each). We use content property to specify the content to insert.

Ex- p::after { content: " End."; }

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p::before {
                content: "Geekyshows";
                color: green;
            }

            p::after {
                content: " By Geekyshows";
                color: red;
            }
        </style>
    </head>
```

Extramarks
Date / /
Page No. 102

```
<body>
    <p>Example of Paragraph.</p>
    <p>Example Of Paragraph.</p>
    <h1>Example Of Heading h1</h1>
</body>
</html>
```

Margin - Browser और किसी element के बीच के gap को margin कहते हैं

Margin Property -

1- margin-left
2- margin-right
3- margin-top
4- margin-bottom
5- margin

1- margin-left - This property is used to set left margin of element. We can set this property to auto or length in the form of %, cm, px, etc. It is by default set 0px; . When we set auto value browser decide the margin.

Syntax- margin-left: value;

Ex- (i) img{ margin-left: 20%; }
(ii) h1{margin-left: 40px;}
(iii) p{margin-left: auto;}

→ <!DOCTYPE html>
<html>
    <head>

Extramarks
Date ___
Page No. 103

```
<title> CSS </title>
<style>
    p { margin-left: 100px; margin-right: 100px;
    div {
        background-color: green;
        margin-left: 100px;
    }
</style>
</head>
<body>
    <div>
        <p> Example Of Paragraph. </p>
    </div>
    <p> Example Of Paragraph </p>
    <h1> Example Of Heading h1 </h1>
</body>
</html>
```

2- margin-right- This property is used to set right margin of element. We can set this property to auto or length in the form of %, cm, px etc. It is by default set 0px; When we set auto value browser decide the margin. Syntax- margin-right: value;

Ex-(i) img { margin-right: 20% }
(ii) h1 { margin-right: 40px; }
(iii) p { margin-right: auto; }

3&4- margin-top & margin-bottom- margin-top property is used to set top margin and margin-bottom is used to set bottom margin of element. We can set these properties to auto or length in

Extramarks
Date ___
Page No. 104

the form of %, cm, px etc. It is by default set 0px, when we set auto value browser decide the margin.

Syntax- margin-top: value; & margin-bottom: value;

Ex- p{margin-top: 10px;} & p{margin-bottom: 10px;}

→
```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            p{
                margin-bottom: 100px;
                margin-top: 100px;
            }
            div{
                background-color: green;
                margin-top: 100px;
            }
        </style>
    </head>
    <body>
        <div>
            <p>Example Of Paragraph</p>
        </div>
        <hr>
        <p>Example Of Paragraph</p>
        <hr>
        <h1>Example Of Heading1</h1>
    </body>
</html>
```

for 3 val

for 2 val

for singl

→

Extramarks
Date ___/___/___
Page No. 105

* margin- It's shorthand margin property. It declares all margin properties in one single line. We can set this property to auto or length in the form of %, cm, px, etc. It is by default set 0px; when we set auto value browser decide the margin.

Syntax-

for 4 value- selector { margin: top value right value bottom value left value; }

Ex- p { margin: 10px 20px 30px 40px; }

for 3 value- selector { margin: top value right & left value bottom value; }

Ex- p { margin: 10px 20px 30px; }

for 2 value- selector { margin: top & bottom value left & right value; }

Ex- p { margin: 10px 20px; }

for single value- selector: for all value;

Ex- p { margin: 20px; }

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
        p{
            background-color: red;
            margin: 10px 20px 30px 40px;
        }
        h1{
            background-color: pink;
            margin: 10px 20px 30px;
        }
```

Extramarks
Date ___/___/___
Page No. 106

```
h2{
    background-color: yellow;
    margin: 10 px 20 px;
}
h3{
    background-color: blue;
    margin: 10 px;
}
</style>
</head>
<body>
<p>Example Of Paragraph.</p> <hr>
<h1>Example Of Heading h1</h1> <hr>
<h2>Example Of Heading h2</h2> <hr>
<h3>Example Of Heading h3</h3> <hr>
</body>
</html>
```

Pdding Property - This property is used to set the padding of specified element.

1- padding-left
2- padding-right
3- padding-top
4- padding-bottom
5- padding*

182- padding-left & padding-right - padding-left is used to set ~~bottom~~ left padding of element. ~~We can set this~~ and padding-right is used to set right padding of element. We can set these properties to length in the form of %, cm, px etc.

Syntax- padding-left: value; & padding-right: value;

Ex- img{padding-left: 20%;} & h1{padding-right: 40 px;}

Page No. 107

38. padding-top & padding-bottom – padding-top is used to set top padding and padding-bottom is used to set bottom padding of element. We can set these properties to length in the form of %, cm, px etc.

Syntax- padding-top: value; & padding-bottom: value;

Ex- p{ padding-top: 10%;} & ing{ padding-bottom: 10px;}

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </ title>
        <style>
            div{
                background-color: khaki;
                padding-left: 10px;
                padding-right: 20px;
                padding-top: 30px;
                padding-bottom: 40px;
            }
        </style>
    </head>
    <body>
        <div>
            <p>
                Example Of Paragraph.
            </p>
        </div>
    </body>
</html>
```

Extramarks
Date
Page No. 108

5- padding- It's shorthand padding property. It declares all padding properties in one single line, we can set this property to left in the form of %, cm, px etc.

Syntax-

for 4 values- selector { padding : top value right value bottom value left value }

Ex- p { padding : 10px 20px 30px 40px; }

for 3 values- selector { padding : top value right & left value bottom value; }

Ex- p { padding : 10px 20px 30px; }

for 2 values- selector { padding : top & bottom value left & right value; }

Ex- padding : 20px 10px; }

for single value- selector { padding : value for all; }

Ex- p { padding : 10px; }

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            p{
                background-color : red;
                padding : 10px 20px 30px 40px;
            }
            h1{
                background-color : pink;
                padding : 10px 20px 30px;
            }
            h2{
                background-color : yellow;
                padding : 10px 20px;
```

```
h3{
    background-color: 10px;
    padding: 10px;
}
</style>
</head>
<body>
    <p> Example Of Paragraph. </p>
    <h1> Example Of Heading h1 </h1>
    <h2> Example Of Heading h2 </h2>
    <h3> Example Of Heading h3 </h3>
</body>
</html>
```

Difference between Margin & Padding-

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </ title>
        <style>
            #firstdiv{
                float: left;
                background-color: pink;
                width: 200px;
                height: 100px;
                margin: 70px;
                padding 20px;
            }
            #seconddiv{
                float: left;
                background-color: gray;
```

Extramarks
Date ___ / ___ / ___
Page No. 110

```
            width: 200px;
            height: 100px;
        }
    </style>
</head>
<body>
    <div id="firstdiv">
        <p> Example of Paragraph.</p>
    </div>
    <div id="secondiv">
        <p>Example Of Paragraph.</p>
    </div>
</body>
</html>
```

Width- This property is used to set width of an element. We can set this property to auto and length in the form of px, cm, % etc.

Note- इसका use हम केवल Block-level elements के साथ ही कर सकते है और यदि इसका use हमे inline element के साथ करना हो तो पहले उसके display value को block करना होगा यदि काम height के लिये भी करना होगा।

Syntax- width: value;

Ex- p {width: 100px;}

Height:- This property is used to set height of an element. We can set this property to auto and length in the form of px, cm, % etc.

Syntax- height: value;

Ex- p{height: 40%;}

Extramarks
Date ___/___/___
Page No. 111

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style type="text/css">
            #idiv {
                width: 30%;
                height: 300px;
                background-color: pink;
            }
            a {
                display: block;
                width: 30%;
                height: 200px;
                background-color: pink;
            }
        </style>
    </head>
    <body>
        <div id="idiv">
            <p>
                Example Of Paragraph :-
            </p>
        </div>
        <h1> Example Of ~~Paragraph~~ Heading h1 </h1>
        <a href="#">Click Here</a>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 112

min-width- This property used to set the minimum width of an element. We can set this property to auto and length in the form of px, cm, % etc.
Syntax- min-width: value;

max-width- This property used to set the maximum width of an element. We can set this property to auto and length in the form of px, cm, % etc.
Syntax- max-width: value;

min-height- This property used to set the minimum height of an element. We can set this property to auto and length in the form of px, cm, % etc.
Syntax- min-height: value;

max-height- This property used to set the maximum height of an element. We can set this property to auto and length in the form of px, cm, % etc.
Syntax- max-height: value;

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            p {
                min-width: 600px;
                max-width: 800px;
```

Extramarks
Date ___/___/___
Page No. 113

```
            min-height: 200px;
            max-height: 400px;
            background-color: pink;
        }
    </style>
</head>
<body>
    <p>
        Example Of Paragraph.
    </p>
</body>
</html>
```

Opacity- से element के transparency को set करने के लिये use करते है। इसमे value 0 से 1 के बीच मे कुछ भी देसकते है (like 0.5)। 0 पर pure transparent और 1 पर normal दिखायी देगा।

Syntax- opacity: value;

Opacity with img tag & div tag-

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            img { opacity: 0.5; }
            div {
                width: 60px;
                height: 30px;
                background-color: blue;
            }
```

Extramarks
Date 28/6/19
Page No. 114

```
div: hover { opacity: 0.5; }
</style>
<body>
    <img src="dog.bmp" ></ img>
    <div></div>
</body>
</html>
```

Opacity with text- अगर हम चाहे कि text पर काम ना करे ~~बल्कि~~ बल्कि उसके background पर काम करे तो हमें background color को rgba (redvalue, greenvalue, bluevalue, opacity value); में देना होगा।

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS</title>
        <style>
            div {
                width: 300px;
                height: 200px;
                background-color: rgba (192,192,64,0.5)
            }
        </style>
    </head>
    <body>
        <div>
            <p> Example of Paragraph.
            </p>
        </div>
    </body>
</html>
```

Border-Left-

1- border-left-style - left border को एक specific style देने के लिये use करते है।

values-

- none- This is default which specifies no border.
- hidden- This is very similar to none value, except in border conflict resolution for table elements.
- solid- It specifies a solid border.
- dotted- It specifies a dotted border.
- double- It specifies a double border.
- dashed- It specifies a dashed border.
- groove- It specifies a 3D grooved border.
- ridge- It specifies a 3D ridged border.
- inset- It specifies a 3D inset border.
- outset- It specifies a 3D outset border.

Syntax- border-left-style: value;

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div { border-left-style: solid; }
        </style>
    </head>
    <body>
        <div>
            <p> Example Of Paragraph. </p>
        </div>
    </body>
</html>
```

Extramarks
Date ___
Page No. 116

2- border-left-width - इसको use करने से पहले left-style की value देना होगा, border-

values-

- medium- This is default value which specifies a medium left border.
- thin- It specifies a thin left border.
- thick- It specifies a thick left border.
- length- We can define left border length in the form of %, px etc.

Syntax- border-left-width: value;

3- border-left-color- left-border को Color देने के लिये

values,-

- Color value
- transparent.

Syntax- ~~bordercolor~~ border-left-color: value;

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            div {
                border-left-style: solid;
                border-left-width: 10 px;
                border-left-color: red;
            }
        </style>
    </head>
    <body>
        <div>
            <p> Example of Paragraph. </p>
        </div>
    </body>
</html>
```

Extramarks
Date: / /
Page No. 117

4- **border-left-** It is shorthand of border-left-style, border-left-width and border-left-color.

Syntax- selector border-left: widthvalue stylevalue colorvalue;

Note- हम चाहें तो color या width में से किसी एक को नहीं भी लिख सकते हैं पर order सही रहेगा।

Ex- p{ border-left: 10px solid red; }

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            div{ border-left: 10px solid red;}
            h1{ border-left: 10px solid;}
            h2{ border-left: solid red;}
            h3{ border-left: solid;}
        </style>
    </head>
    <body>
        <div>
            <p>Example Of Paragraph.</p>
        </div>
        <h1> Example Of Heading h1</h1>
        <h2>Example Of Heading h2</h2>
        <h3> Example Of Heading h3</h3>
    </body>
</html>
```

Border-Right-

1- **border-right-** right border को एक specific style देने के लिये use करते हैं।

Values-

Extramarks
Date ___/___/___
Page No. 118

- none- this is default which specifies no border.
- hidden- This is very similar to none value except in border conflict resolation for table element.
- solid- It is specifies a solid border.
- dotted- It is specifies a dotted border.
- double- It is specifies a double border.
- dashed- It is specifies a dashed border.
- groove- It is specifies a 3D groove border.
- ridge- It is specifies a 3D ridge border.
- inset- It is specifies a 3D inset border.
- outset- It is specifies a 3D outset border.

Syntax- border-right-style : value;

2- border-right-width-
value- वही सभी value जो border-left-width में use कर सकते हैं।
Syntax- border-right-width: value;

3- border-right-color-
values- color value, transparent.
Syntax- border-right-color: value;

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div {
                background-color: red;
                border-right-style: dotted;
                border-right-width: 10px;
                border-right-color: pink;
            }
        </style>
```

Extramarks
Date / /
Page No. 119

```
</head>
<body>
    <div>
        <p>Example of Paragraph</p>
    </div>
</body>
</html>
```

4- **border-right** - It is shorthand of border-right-style, border-right-width and border-right-color.

Syntax- selector{ border-right : width value style value color value ;}

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style >
            div { border-right : 10px solid red;}
            h1{ border-right : 10px solid;}
            h2{ border-right : solid red;}
            h3{ border-right : solid;}
        </style>
    </head>
    <body>
        <div>
            <p> Example Of Paragraph. </p>
        </div>
        <h1> Example Of Heading h1</h1>
        <h2> Example Of Heading h2</h2>
        <h3> Example Of Heading h3</h3>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 120

Border Top-

1- border-top-style-
Value- इसमें भी वही सब value use कर सकते हैं जो border-left-style में use कर सकते हैं।
Syntax- border-top-style:

2- border-top-width-
Value- इसमें भी वही सब value use कर सकते हैं जो border-left-width में use कर सकते हैं।
Syntax- border-top-width; value

3- border-top-color-
Value- ~~सभी भी वही सब~~ color value & transparent.
Syntax- border-top-color: value;

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div{
                border-top-style: solid;
                border-top-weight: 10px;
                border-top-color: pink;
            }
        </style>
    </head>
    <body>
        <div>
            <p> Example of Paragraph.</p>
        </div>
    </body>
</html>
```

4- border-top- It is shorthand of border-top-style, border-top-width And border-top-color.

Syntax- selector { border-top: width value style value color value;}

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            div{ border-top:10px solid red;}
            h1{ border-top: 10px solid;}
            h2 { border-top: solid red;}
            h3{ border-top: solid;}
        </style>
    </head>
    <body>
        <div>
            <p> Example of Paragraph</p>
        </div>
        <h1> Example Of Heading h1</h1>
        <h2> Example Of Heading h2 </h2>
        <h3> Example Of Heading h3 </h3>
    </body>
</html>
```

Border - Bottom -

1- border-bottom-style- सब value use कर सकते हैं जो value इसमें हैं वही में use कर सकते हैं।

Syntax- border-bottom-style: value;

Date ___ / ___ / ___
Page No. 122

2- ~~bot~~ border-bottom-~~w~~width -
Value- इसमें भी वहीं सब value use कर सकते हैं जो हम border-~~bot~~left-width के साथ use कर सकते हैं।
Syntax- border-bottom-width: value;

3- ~~bot~~ border-bottom-color-
value- color value & transparent
Syntax- border-bottom-color: value;

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div{
                border-bottom-style: solid;
                border-bottom-width: 10px;
                border-bottom-color: pink;
            }
        </style>
    </head>
    <body>
        <div>
            <p> Example Of Paragraph. </p>
        </div>
    </body>
</html>
```

4- border-bottom- It is shorthand of ~~bottom~~ border-bottom-style, border-bottom-width and border-bottom-color.
Syntax- selector { border-bottom: width value style value color value }

Extramarks
Date / /
Page No. 123

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </ title>
        <style>
            div { border-bottom :10px solid pink;}
                ~~border-bottom~~
            h1 { border-bottom : 10px solid ;}
            h2 { border-bottom : solid pink ; }
            h3 { border-bottom : solid ;}
        </style>
    </head>
    <body>
        <div>
            <p> Example Of Paragraph. </p>
        </div>
        <h1> Example Of Heading h1 </h1>
        <h2> Example Of Heading h2 </h2>
        <h3> Example Of Heading h3 </h3>
    </body>
</html>
```

Border - Rounded Corners-

1- border-top-left-radius
2- border-top-right-radius
3- border-bottom-left-radius
4- border-bottom-right radius
5- border-radius

Note- इसमें हम केवल length value in form of %, px, cm etc. देते हैं। हम चाहें तो दो value भी दे सकते हैं पहली value border के Horizontal line को और दूसरी value border के vertical line को corner से round करेगी।

Extramarks
Date 29/6/19
Page No. 124

1- border-top-left-radius-
Syntax-
for 2 value- border-top-left-radius: Horizontal value vertical value;
for 1 value- border-top-left-radius: Horizontal & vertical value;

2- border-top-right-radius-
Syntax-
for 2 value- border-top-right-radius: Horizontal value Vertical value;
for 1 value- border-top-right-radius: Horizonal & vertical value;

3- border-bottom-left-radius-
Syntax-
for 2 value- border-bottom-left-radius: Horizontal value vertical value;
for 1 value- border-bottom-left-radius: Horizontal & vertical value;

4- border-bottom-right-radius-
Syntax-
for 2 value- border-bottom-right-radius: Horizontal value vertical value
for 1 value- border-bottom-right-radius: Horizontal & vertical value;

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS</title>
        <style>
            div {
                border: 10px solid red;
                border-top-left-radius: 10px;
                border-top-right-radius: 10px;
                border-bottom-left-radius: 10px;
                border-bottom-right-radius: 10px;
```

Page No. 125

```
h1{
    border: 10 px solid green;
    border-top-left-radius: 10px 20px;
    border-top-right-radius: 10px 20px;
    border-bottom-left-radius: 10px 20px;
    border-bottom-right-radius: 10px 20px;
}
</style>
</head>
<body>
<div>
    <p> Example Of Paragraph. </p>
</div>
    <h1> Example Of Heading h1 </h1>
</body>
</html>
```

5- **border-radius-**

Syntax-

for 4 value- border-radius: topleft value topright value bottom-right value bottom left value

or border-radius: H topleft H topright H bottom left H bottom right / V top left V top right V bottom left V bottom right.

for 3 value- border-radius: topleft topright & bottom left bottom right

or border-radius: H topleft H topright & H bottom left H bottom right / V top left V top right & V bottom left V bottom right

for 2 value- border-radius: topleft & bottom right topright & bottom left

or border-radius: H top-left & H bottom right H topright & H bottom left / V top left & V bottom right V top right & V bottom left

for 1 value- border-radius: for all values

or border-radius: H all value / V all value

Extramarks
Date ___/___/___
Page No. 126

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div {
                border: 10px solid red;
                border-radius: 10px 10px 10px 10px;
            }
            h1 {
                border: 10px solid green;
                border-radius: 10px 10px 10px 10px
                /20px 20px 20px 20px;
            }
        </style>
    </head>
    <body>
        <div>
            <p>Example Of Paragraph. </p>
        </div>
        <h1> Example Of Heading h1 </h1>
    </body>
</html>
```

Border-Radius with img Tag-

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            img {
```

Page No. 127

```
            width: 400px;
            height: 200px;
            border-radius: 50%;
        }
    </style>
  </head>
  <body>
        <img src="dog.bmp"></img>
  </body>
</html>
```

box-shadow - This property is used to attach one or more shadows to an element.

Syntax- box-shadow: h-shadow v-shadow blur spread color inset;

for multiple shadow- box-shadow: h-shadow | v-shadow | blur | spread | color | inset;

Note- h-shadow और v-shadow देना अनिवार्य है। by default value none रहती है। इसका use करके हम text में shadow नहीं दे सकते हैं।

- h-shadow- The position of the horizontal shadow. Negative values are allowed.
- v-shadow- The position of the vertical shadow. Negative values are allowed.
- blur- The blur distance. (only in px).
- spread- Size of shadow. Negative values are allowed. (only in px).
- color- The color of the shadow.
- none- No shadow. (Default.)
- inset- Changes the shadow from an outer shadow (outset) to an shadow.

Ex- (i) div { box-shadow: 2px 2px 5px red; }
(ii) div { box-shadow: 2px 2px 5px red, 5px 5px 5px blue; }

Extramarks
Date ___/___/___
Page No. 128

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </ title>
        <style>
            #firstdiv {
                width : 300px;
                height : 200px;
                background-color: pink;
                box-shadow: 5px 3px 10px
            red, 10 px 6 px 20px blue;
            }
            #seconddiv {
                width : 300 px;
                height : 200 px;
                background-color: pink;
                box-shadow : 5 px 3 px 10 px
                green inset;
            }
        </ style>
    </head>
    <body>
        < div id = " firstdiv > </ div >
        <hr>
        < div id = " seconddiv " > </ div >
    </ body >
</html>
```

box-shadow with img tag-

```
-> <!DOCTYPE html>
<html>
    <head>
```

Extramarks
Date ___/___/___
Page No. 129

```
<title> CSS </title>
<style>
        img {
                width: 400px;
                height: 200px;
                border: 3px solid;
        }
        img:hover { box-shadow: 20px x 10px
                                20px; }
</style>
</head>
<body>
        <img src" dog.png" > </img>
</body>
</html>
```

<u>Transition</u> - Transition is used to change property values from one value to another, over a given duration. The duration effect will start when the specified CSS property changes value.

How to use Transition?
: CSS property you want to add an effect to
: Duration of the effect.

<u>Note</u>- If the duration part is not specified, the transition will have no effect, because the default value is 0.

Extramarks
Date ___/___/___
Page No. 130

1- transition-timing-function
2- transition-property
3- transition-duration
4- transition-delay
5- transition

2- transition-property - This property is used to specify the name of the CSS property the transition effect is for (the transition effect will start when the specified CSS property changes). Always specify the transition-duration property, otherwise the ~~dur~~ duration is and transition will have no effect. We can set this property to none, all and ~~other CSS~~ property.

Syntax- transition-property: value;

Ex- div { transition-property: all;}

- none- No property will get a transition effect.
- all- All properties will get transition effect. This is default value.
- property- Define a comma separated list of CSS property name the transition effect is for.

3- transition-duration- This property is used to specify how many seconds or milliseconds a transition effect take to complete. We can set this property to time in the form of second (s) or millisecond (ms). The default value of this property is 0s which means there will be no effect.

Page No. 131

Syntax- transition-duration : value;
Ex- (i) div { transition-duration : 100ms;}
(ii) div { transition-duration : 10s;}

1. transition-timing-function- This property is used to specify the speed curve of the transition effect to change speed over its duration. We can set this property to ease, linear, ease-in, ease-out, ease-in-out, step-start, step-end, steps (int, start, or end), cubic-bezier(n,n,n,n).
Syntax- transition-timing-function: value;
Ex- div { transition-timing-function: ease-in;}

- ease- It specifies a transition effect with a slow start, then fast, then end slowly (equivalent to cubic-bezier(0.25, 0.1, 0.25, 1)). This is default value.

- linear- It specifies a transition effect with the same speed from start to end (equivalent to cubic-bezier(0,0,1,1)).

- ease-in- It specifies a transition effect with a slow start (equivalent to cubic-bezier(0.42,0,1,1)).

- ease-out- It specifies a transition effect with a slow end (equivalent to cubic-bezier(0,0,0.58,1))

- ease-in-out- It specifies a transition effect with a slow start and end (equivalent to cubic-bezier(0.42,0,0.58,1))

- step-start- Equivalent to steps(1, start)

Date / / Page No. 132

- step-end- Equivalent to steps (1, end).

- steps (int, start or end)- Specifies a stepping function, with two parameters. The first parameter specifies the number of intervals in the function. It must be a positive integer (greater than 0). The second parameter, which is optional, is either the value "start" or "end", and specifies the point at which the change of value occurs within the interval. If the second parameter is omitted, it is give the value "end".

- cubic-bezier(n,n,n,n) - Define your own values in the cubic-bezier function. Possible values are numeric values for 0 to 1.

4- transition-delay- This property is used to specify when the transition effect will start. We can set this property to time in the form of seconds (s) or milliseconds (ms).

Syntax- transition-delay: value;

Ex- (i) div{transition-delay: 2px;}

(ii) div { transition-delay: 100ms;}

```
→ <!DOCTYPE html>
  <html>
      <head>
          <title> CSS </title>
          <style>
```

Page No. 133

```
div {
        width : 100px ;
        height : 100px;
        background-color : pink ;
    }
div:hover {
        width : 300px;
        height : 200px;
        background-color : green;
    }
div {
        transition-property : all;
        transition-duration : 2s;
        transition-timing-function : cubic-
                bezier : (0.885, 0.03, 0.685, 0.23)
        transition-delay : 1s;
    }
</style>
</head>
<body>
    <div></div>
</body>
</html>
```

5- **transition-** The transition property is a ~~shorthand~~ shorthand property for the four ~~trans~~ transition ~~property~~ properties.

Syntax- Selector{ transition : transition-property , transition-duration, transition-timing-function, transition-delay;}

Ex- div{ transition : width 200ms ease;}

Note- Always specify the transition-duration property, otherwise the duration is 0s, and the transition will have no effect. transition-property की value किसी specific element के लिये देने के लिये, अलग से या comma ,, से separate करके लिखना होगा।

Extramarks
Date ___/___/___
Page No. 134

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            #fdiv {
                width: 100px;
                height: 100px;
                background-color: pink;
                transition: width 2s ease-in,
                height 6s ease-in;
            }
            #fdiv:hover {
                width: 300px;
                height: 200px;
            }
            #ldiv {
                width: 100px;
                height: 100px;
                background-color: pink;
                transition: all 2s ease-in;
            #ldiv:hover {
                width: 300px;
                height: 200px;
                background-color: green;
            }
        </style>
    </head>
    <body>
        <div id="fdiv"></div><hr>
        <div id="ldiv"></div>
    </body>
</html>
```

Page No. 135

Cursor-

Syntax- cursor: value;

- default- The default cursor.
- auto- Default. The browser sets a cursor.
- all-scroll- The cursor indicates that something can be scrolled in any direction.
- none- No cursor is rendered for the element.
- not-allowed- The cursor indicates that the requested action will not be executed.
- text- The cursor indicates text that may be selected.
- URL- A comma separated list of URL-defined cursors can be used.
- row-resize- The cursor indicates that the row can be resized vertically.
- s-resize- The cursor indicates that an edge of a box is to be moved down (south).
- se-resize- The cursor indicates that an edge of a box is to be moved down and right (south/east).
- sw-resize- The cursor indicates that an edge of a box is to be moved down and left (south/west).
- n-resize- The cursor indicates that an edge of a box is to be moved up (north).
- ne-resize- The cursor indicates that an edge of a box is to be moved up and right (north/east).
- nesw-resize- Indicates a bidirectional resize cursor.
- nw-resize- The cursor indicates that an edge of a box is to be moved up and left (north/west)
- ns-resize- Indicates a bidirectional resize cursor.

Extramarks
Date ___/___/___
Page No. 136

- nwse-resize - Indicates a bidirectional resize cursor.
- copy - The cursor indicates something to be copied.
- * alias - The cursor indicates an alias of something is to be created.
- e-resize - The cursor indicates that an edge of box is to be moved right (east).
- ew-resize - Indicate a bidirectional resize cursor.
- * pointer - The cursor is a pointer and indicates a link.
- * progress - The cursor indicates that the program is busy (in progress).
- * zoom-out. The cursor indicates that something can be zoomed out.
- * zoom-in - The cursor indicates that something can be zoomed in.
- vertical-text - The cursor indicates vertical-text that may be selected.
- w-resize - The cursor indicates that an edge of a box is to be moved left (west).
- * wait - The cursor indicates that the program is busy.
- Context-menu - The cursor indicates that a context-menu is available.
- col-resize - The cursor indicates that the column can be resized horizontally.
- cell - The cursor indicates that a cell (or set of cells) may be selected.
- * no-drop - The cursor indicates that the dragged

item cannot be dropped here.

- crosshair - The cursor render as a crosshair.
- grab - The cursor indicates that something can be grabbed.
- grabbing - The cursor indicate that something can be grabbed.
- help - The cursor indicates that help is available.
- move - The cursor indicate something is to be moved.

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div {
                width: 400 px;
                height: 200px;
                background-color: pink;
            }
            div:hover { cursor: not-allowed; }
        </style>
    </head>
    <body>
        <div> </div>
    </body>
</html>
```

Extramarks
Date
Page No. 138

Text-decoration- This property is used to specify the decoration added to text.

1- text-decoration-line
2- text-decoration-style
3- text-decoration-color
4- text-decoration.

1- text-decoration-line- This property is used to specify what type of line the decoration will have. We can set this property to none, underline, overline, line-through. We can combine more than one value, like underline and overline to display lines both under and over the text.

Syntax- text-decoration-line: value;

for multi- text-decoration-line: value1, value2, ——;

Ex- (i) h2 { text-decoration-line: underline; }

(ii) h2 { text-decoration-line: underline, overline; }

2- text-decoration-style- This property is used to specify how the line will display. We can set this property to solid, double, dotted, dashed, wavy.

Syntax- text-decoration-style: value;

Ex- h1 { text-decoration-line: underline;
text-decoration-style: dotted; }

3- text-decoration-color- This property is used to specify the color of the text-decoration-line.

Syntax- text-decoration: value;

Extramarks
Date
Page No. 139

Ex- h1 { text-decoration-line: underline;
text-decoration-color: red; }

→ <!DOCTYPE html>
<html>
<head>
<title> CSS </title>
<style>
h1{
text-decoration-line: underline;
text-decoration-style: wavy;
text-decoration-color: red;
}
h2 { text-decoration-line: underline
overline line-through; }
</style>
</head>
<body>
<h1> Example Of Heading h1</h1>
<h2> Example Of Heading h2</h2>
</body>
</html>

4- text-decoration- This ~~property~~ is shorthand of all three text-decoration properties.

Syntax- Selector { text-decoration : text-decoration-line value
text-decoration-style value text-decoration-color value; }

Ex- h1 { text-decoration : underline dotted red; }

Note- ~~text-dec~~ text-decoration-style और text-decoration-color की value लिखना optional है।

Extramarks
Date
Page No. 140

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</ title >
        <style>
            h1{ text-decoration: underline dotted red;
            h2{ text-decoration: underline solid;}
            h3{ text-decoration: green;}
            h4{ text-decoration: underline overline
                line-through wavy blue;}
            a{ text -decoration: none;}
        </style>
    </head>
    <body>
        <h1>Example Of Heading h1 </h1>
        <h2> Example Of Heading h2 </h2>
        <h3> Example Of Heading h3 </h3>
        <h4> Example Of Heading h4 </h4>
        <a href="#" >Click Here </a>
    </body>
</html>
```

Outline- An outline is a line that is drawn around elements, outside the borders. Outlines never take up space, as they are drawn outside of an element's content.

1- outline-color
2- outline-style
3- outline-width
4- outline
5- outline-offset

Page No. 141

2- **outline-style-** This property is used to specify the style of an outline. We can set this property to none (default), hidden, dotted, dashed, solid, double, groove, ridge, inset, outset.

Syntax- outline-style : value;

Ex- div { outline-style : solid; }

1- **outline-color-** This property is used to specify the color of an outline. We can set this property to invert (default) or color.

Syntax- outline-color : value;

Ex- div { outline-color : red; outline-style : solid; }

3- **outline-width-** This property is used to specify width of an outline. We can set this property to medium (default), thin, thick, length in the form of %, cm, px etc.

Syntax- outline-width : value;

Ex- div { outline-style : solid; outline-width : thin; }

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style >
            div {
                border:solid red;
                outline-style : dotted;
                outline-color:green;
                outline-width : 10px;
            }
        </style>
    </head>
```

Extramarks
Date ___/___/___
Page No. 142

```
<body>
    <div>Example of div Tag.</div>
</body>
</html>
```

4- **outline** - This ~~property~~ is shorthand of other outline properties.

Syntax- Selector{ outline: outline-color outline-style outline-width

Ex- div { outline: red solid 3px; }

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div {
                border: solid red;
                outline: yellow dotted 3px;
            }
            h1 {
                border: solid green;
                outline: dotted 3px;
            }
            h2 {
                border: solid blue;
                outline: yellow dotted;
            }
        </style>
    </head>
    <body>
        <div> Example Of div Tag </div>
        <h1> Example of Heading h1 </h1>
```

Extramarks
Date ___/___/___
Page No. 143

```
        <h2>Example of Heading h2</h2>
    </body>
</html>
```

5. **outline-offset** - This property is used to add space between an outline and the edge or border of an element.

Syntax- outline-offset: value;

Ex. div {outline-offset: 5px;}

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div {
                border: solid red;
                outline: blue dotted 3px;
                outline-offset: 5px;
            }
        </style>
    </head>
    <body>
        <div>Example of div Tag</div>
    </body>
</html>
```

Columns - This property is used to define columns.

1- column-count
2- column-width
3- columns

Extramarks
Date ___/___/___
Page No. 144

1- Column-count- This property is used to specify the number of columns an element should be divided into. We can set this property to number or auto (default).

Syntax- column-count: value;

Ex- div { column-count: 2; }

2- Column-width- This property is used to specify width for the columns. We can set this property to length or auto (default).

Syntax- column-width: value;

Ex- div { column-count: 2;
column-width: 100px; }

Note- हमें column-count या column-width में से एक ही use करना चाहिये नहीं तो ज्यादातर दोनों में से कोई एक ही work करता है।

3- Columns- This is shorthand which is used to set column-width and column-count

Syntax- Selector { columns: column-width value column-count value; } ③ →

Ex- div { columns: 200px, 2; }

(122) →
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div {
                column-count: 3 px;
                column-width: 500px;
            }
```

Extramarks
Date ___
Page No. 145

```
        p{ column-width : 100px; }
        h1{ column-count: 4; }
    </style>
</head>
<body>
    <div>
        Example of div Tag.
    </div>
    <h1>
        Example of Heading h1
    </h1>
    <p>
        Example of Paragraph.
    </p>
</body>
</html>.
```

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div { columns : 3 500 px; }
            h1{ columns : 100 px; }
            p{ columns : 4; }
        </style>
    </head>
    <body>
        <div>
```

Extramarks
Date ___/___/___
Page No. 146

Example of div Tag.

```
</div>
<p>
    Example of Paragraph.
</p>
<h1>
    Example of Heading h1..
</h1>
</body>
</html>
```

Column-rule-

1- Column-rule-style
2- Column-rule-width
3- column-rule-color
4- column-rule

1- Column-rule-style- This property is used to specify the style of the rule between columns. We can set this property to none (default), hidden, dotted, dashed, solid, double, groove, ridge, inset and outset.
~~Ex- div~~ Syntax- Column-rule-style: value;
Ex- div { column-rule-style: double;

2- Column-rule-width- This property is used to specify width of the rule between columns. We can set this property to medium

Date 1 7 19
Page No. 147

(default), thin, thick and length.

Syntax- column-rule-width: value;

Ex- column-rule-width: 5px;

3. column-rule-color- This property is used to specify color of the rule between columns. We can set this property to color value.

Syntax- column-rule-color: value;

Ex- div { column-rule-color: red;

```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            div {
                columns: 3;
                ~~column-rule~~ column-rule-style
                : solid

                column-rule-width: 10px
                column-rule-color: red;
            }
        </style>
    </head>
    <body>
        <div>
            Example of div Tag.

        </div>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 148

4- column-rule - This is shorthand of other column-rule-* properties.

Syntax- Selector { column-rule : column-rule-width column-rule-style column-rule-color }

Ex- div { column-rule : 5px solid red;

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            div{
                columns:3;
                column-rule: 10px dotted blue;
            }
            p{
                columns: 100px;
                column-rule: double green;
            }
            h1{
                columns: 4
                column-rule: 15px dashed;
            }
        </style>
    </head>
    <body>
        <div>
            Example of div Tag.
        </div>
        <p>
            Example of Paragraph.
        </p>
```

Extramarks
Date ___/___/___
Page No. 149

```
<h1>
        Example of Heading h1</h1>
</h1>
</body>
</html>
```

column-gap- This property is used to specify the gap between the columns. We can set this property to normal (default) and length.

Syntax- column-gap: value;

Ex- div { column-gap: 30px; }

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS </title>
        <style>
            div {
                column: 3;
                column-gap: 50px;
            }
        </style>
    </head>
    <body>
        <div>
            Example of div Tag.

        </div>
    </body>
</html>
```

Extramarks
Date ___/___/___
Page No. 150

column-span- This property is used to specify how many columns an element should span across. We can set this property to none (default) and all.

Syntax- column-span: value;

Ex- div { column-span: all;

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            div { column: 3; }
            p { column-span : all; }
        </style>
    </head>
    <body>
        <div>
            <p>
                Example of Paragraph.
            </p>
            Example of div Tag.
        </div>
    </body>
</html>
```

must

must

filter- This property is used to define visual effects like blur and brightness to an element.

Syntax- filter: value;

Ex- img { filter: blur (10px;); }

Extramarks
Date
Page No. 151

values -

- none - Default value. Specifies no effects.
- blur (px) - This is used to apply blur effect to the image. A larger value will create more blur.
- brightness (%) - This is used to adjust the brightness of the image. 0% will make the image completely black. 100% is default and represents the original image. Values over 100% will provide brighter ~~outer~~ results.
- contrast (%) - This is used to adjust the contrast of the image. 0% will make the image completely black. 100% is default and represents the original image. Values over 100% will ~~provid~~ provide results with less contrast.
- drop-shadow (h-shadow v-shadow blur spread color) - This is used to apply a drop shadow effect to the image. This is very ~~simil~~ similar to box-shadow property.
  - must - h-shadow - This is used to specify a fixed value for the horizontal shadow. Negative values place the shadow to the left of the image.
  - must - v-shadow - This is used to ~~add~~ specify a fixed value for the vertical shadow. Negative values place the shadow above the image.
  - blur (px) - This is used to add blur effect to the shadow. A larger value will create more blur.
  - spread (px) - This is used to specify size of ~~shado~~ shadow. Negative values will cause the shadow to shrink. If not specified, it will be

Page No. 152

- color - This is used to add a color to the shadow. If not specified, the color depends on the browser (often black).
- grayscale(%) - This is used to convert the image to grayscale. 0% is default and represents the original image. ~~Maximum value is 360 deg~~ 100% will make the image completely gray. Negative values are not allowed.
- hue-rotate(deg) - This is used to specify a hue rotation on the image. 0 deg is default, and represents the original image. Maximum value is 360 deg.
- invert(%) - This is used to invert the samples in the image. 0% is default and represents the original image. 100% will make the image completely inverted. Negative values are not allowed.
- opacity(%) - This is used to set the opacity level for the image. This is similar to the opacity property. 0% is completely transparent. 100% is default and represents the original image. Negative values are not allowed.
- saturate(%) - This is used to saturate the image. 0% will make the image completely un-saturated. 100% is default and represents the original image. Value over 100% provides higher-saturated results. Negative values are not allowed.
- sepia(%) - This is used to convert the image to sepia. 0% is default and represents

the original image. 100% will make the image completely sepia. Negative values are not allowed.

. url()- The url() function takes the location of an XML file that specifies an SVG filter, and may include an anchor to a specific filter element.

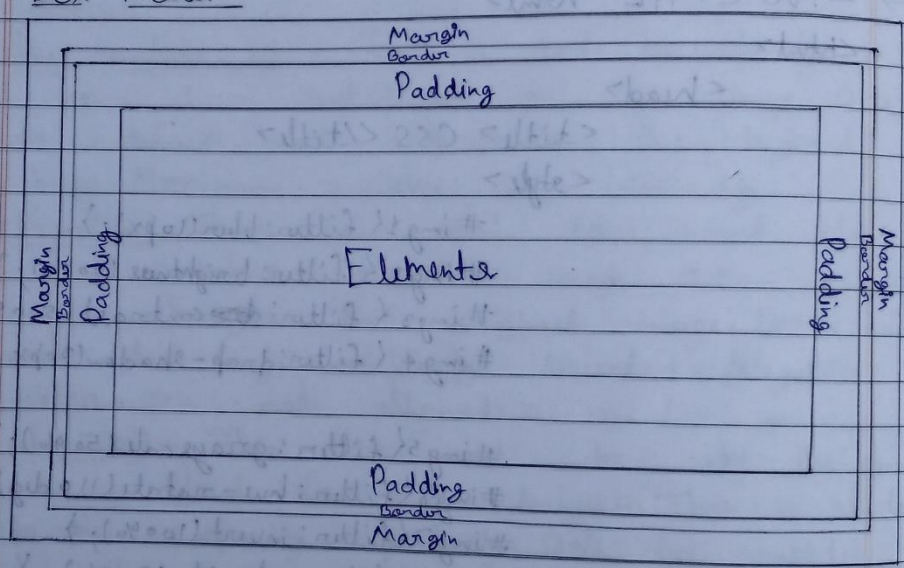
Ex- filter: url(svg-url#element-id)

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            #img1{ filter: blur(10px); }
            #img2{ filter: brightness(20%); }
            #img3{ filter: contrast(500%); }
            #img4{ filter: drop-shadow(20px 10px 10px red); }

            #img5{ filter: grayscale(50%); }
            #img6{ filter: hue-rotate(110deg); }
            #img7{ filter: invert(100%); }
            #img8{ filter: opacity(30%); }
            #img9{ filter: saturate(50%); }
            #img10{ filter: sepia(80%); }

        </style>
    </head>
    <body>
        <img id="img1" src="example.jpg"></img><hr>
        <img id="img2" src="cat.jpg"></img><hr>
        <img id="img3" src="car.jpg"></img><hr>
        <img id="img4" src="dog.jpg"></img><hr>
        <img id="img5" src="error.jpg"></img><hr>
```

Extramarks
Date ___/___/___
Page No. 154

```
<img id="img6" src="dog.bmp"></img><hr>
<img id="img7" src="image1.jpg"></img><hr>
<img id="img8" src="image3.jpg"></img><hr>
<img id="img9" src="image4.jpg"></img><hr>
<img id="img10" src="download.png"></img>
</body>
</html>
```

Box Model –



width= left margin + left border + left padding + width + right padding + right border + right margin

Height- top margin + top border + top padding + height + bottom padding + bottom border + bottom margin

→
```
<!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
```

Page No. 155

```
<style>
    div {
        width: 300px;
        height: 200px;
        margin: 5 px;
        border: 5px solid green;
        padding: 5px;
    }
</style>
</head>
<body>
    <div style="background-color: pink">This is content area. </div>
    <div style="background-color: grey">This is content area.</div>
</body>
</html>
```

box-sizing- The box sizing property tells the browser what the sizing properties should include. We can set this property to content-box (default) or border-box. The box-sizing property is used to include the padding and border in an element's total width and height.

Syntax- border-sizing: value;

Ex div { box-sizing : border-box; }

values-

- Content-box- The width and height properties (and min/max properties) includes only the content. Border, padding, or margin are not included.
- border-box- The width and height properties (and min/max properties) includes content, padding and border, but not the margin.

Extramarks
Date ___/___/___
Page No. 156

```
--> <!DOCTYPE html>
<html>
    <head> <title> CSS </title>
        <style>
            div {
                box-sizing: border-box;
                width: 300px;
                height: 200px;
                margin: 5 px;
                border: 5 px solid green;
                padding: 5px;
            }
        </style>
    </head>
    <body>
        <div style="background-color: pink"> This is content area. </div>
        <div style="background-color: grey"> This is content area. </div>
    </body>
</html>
```

Attribute Selector- We can set style using attribute or its value.

- Set style using attribute-

  Syntax- Selector [attribute] { property: value; }

  Ex- div[id] { color: red; }

```
=> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
```

Page No. 157

- Set style using attribute and its value -
  Syntax- Selector [attribute = "value"] { property: value; }
  Ex- div [id="data"] { color: red; }

- Set Style using an attribute value containg a specified word - (जो भी value में word होगा वही apply होगा।
  Syntax- [attribute ~= "value"] { property: value; }
  Ex- [class ~= "game"] { color: red; }

- Set Style using with the specified attribute starting with the specified value -
  Syntax- [attribute |= "value"] { property: value; }
  Ex- [class |= "game"] { color: red; }
  Note- The value must be a whole word. Ex- "game", "game-super"

- Set Style whose attribute value begins with a specified value -
  Syntax- [attribute ^= "value"] { property: value; }
  Ex- [class ^= "game"] { color: red; }
  Note- It is not necessary that the value should be a whole word.

- Set Style whose attribute value ends with a specified value -
  Syntax- [attribute $= "value"] { property: value; }
  Ex- [class $= "game"] { color: red; }
  Note- It is not necessary that the value should be a whole word.

- Set Style whose attribute value contains a specified value -
  Syntax- [attribute *= "value"] { property: value; }
  Ex- [class *= "ga"] { color: red; }

Extramarks
Date ___/___/___
Page No. 158

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </ title>
        <style>
            div [id] { color: blue; }
            div[id = "dv"] { color: green; }
            div[class = "d" ] { color: yellow; }
            div[class ~= "game" ] { color: pink; }
            div[class |= "work" ] { color: khaki; }
            div[class $= "code" ] { color: grey; }
            div[class ^= "joy" ] { color: purple; }
            div[class *= "za" ] { color: orange; }
        </style>
    </head>
    <body>
        <div id="data" > This is 1st div. </div>
        <div class="dv" > This is 2nd div. </div>
        <div id="dv" > This is 3rd div. </div>
        <div class=" d" > This is 4 th div. </div>
        <div class="game super" > This is 5th div. </div>
        <div class="game" > This is 6th div. </div>
        <div class="work" > This is 7th div. </div>
        <div class="work-space" > This is 8th div. </div>
        <div class="joy-space" > This is 9th div. </div>
        <div class="joy space" > This is 10th div. </div>
        <div class="source code" > This is 11th div. </div>
        <div class="zero-code" > This is 12th div. </div>
        <div class="abzanan" > This is 13th div. </div>
        <div class="zajid-bahe" > This is 14th div. </div>
    </body>
</html>
```

Page No. 159

FlexBox or Flexible Box -

What is it?

- It Layout mode a CSS3 concept

Why we need it?

- It helps to create responsive web pages.

Flexbox Properties -

- display
- flex-direction
- justify-content
- align-items
- flex-wrap
- align-content
- flex-flow
- order
- align-self
- flex

Display - नीचे सभी value बतायी गयी है अब यहाँ हम flex and inline-flex value के बारे में जानेंगे।

- flex - Displays an element as a block-level flex container
- inline-flex - Displays an element as an inline-level flex container.

flex-direction - This property is used to set the direction of flexible items. If the element is not a flexible item, the flex-direction property has no effect. We can set this property to row (default), row-reverse, column, column-reverse.

Extramarks
Date ___/___/___
Page No. 160

Syntax- flex-direction: ~~column-reverse~~ value;

- Ex- div { flex-direction: column-reverse; }
- values- row- This is used to display the flexible items horizontally as row.
- row-reverse- This is used to display the flexible items same as row, but in reverse order.
- column- This is used to display the flexible items vertically as column.
- column-reverse- This is used to display the flexible items same as column, but in reverse order.

```
-> <!DOCTYPE html>
<html>
    <title> CSS </title>
    <style>
        #container {
            border: 3px solid black;
            display: flex;
            flex-direction: row-reverse;
        }
        div {
            text-align: center;
            padding: 20 px;
            margin: 10px;
        }
        #firstdiv { background-color: red; }
        #seconddiv { background-color: green; }
        #thirddiv { background-color: yellow; }
        #Fourthdiv { background-color: purple; }
        #fifthdiv { background-color: blue; }
```

```
        </style>
    </head>
    <body>
        <div id="container">
            <div id="firstdiv">1</div>
            <div id="seconddiv">2</div>
            <div id="thirddiv">3</div>
            <div id="fourthdiv">4</div>
            <div id="fifthdiv">5</div>
        </div>
    </body>
</html>
```

Justify-content - The property is used to align the flexible container items when the items do not use all available space on the main-axis (horizontally). We can set this property to flex-start (default), flex-end, center, space-between, space-around.

Syntax- justify-content: value;

Ex- div { justify-content - center; }

Values-

- **flex-start** - This is used to position the item at the beginning of the container.
- **flex-end** - This is used to position the item at the end of the container.
- **center** - This is used to position the item at the center of the container.
- **space-between** - This is used to position the item with space between the lines.
- **space-around** - This is used to position the item with space before, between, and after the lines.

Note- इसके use से horizontally alignment होता है।

Page No. 162

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            #container{
                border:3px solid black;
                display:flex;
                justify-content:flex-end;
            }
            div{
                text-align:center;
                padding:20px;
                margin:10px;
            }
            #firstdiv{background-color:red;}
            #seconddiv{background-color:blue;}
            #thirddiv{background-color:green;}
        </style>
    </head>
    <body>
        <div id="container">
            <div id="firstdiv">1</div>
            <div id="seconddiv">2</div>
            <div id="thirddiv">3</div>
        </div>
    </body>
</html>
```

align-items- This property is used to specify the default alignment for items inside the container. This property can be override using align-self property for each item. We can set this property to stretch (default), center, flex-start, flex-end, baseline

Extramarks
Date ___ / ___ / ___
Page No. 163

Syntax- align-items : ~~center~~ value;
Ex- div { align-items: center; }
values - { center, flex-start, flex-end ये सब पीछे वाले की तरह काम करेंगे।
. stretch- This is used to struch the item to fit the container. (पर यदि hight दिया होगा height से नया नहीं stretch होगा।)
. baseline- This is used to position the item at the baseline of the container.
Note- इसके use से vertically align होता है।

```
<!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        <style>
            #container{
                border: 3px solid black;
                display: flex;
                head: 300px;
                justify-content: center;
                align-items: center;
                align-items: baseline;
            }

            div {
                text-align: center;
                padding: 20px;
                margin: 10px;
                height: 30px;
            }
            #firstdiv{ background-color: red;}
            #seconddiv{
                background-color: blue;
                font-size: 50px;
            }
```

Page No. 164

```
        # thirddiv { background-color:green;}
        ~~background-color~~
    </style>
</html>
<body>
    <div id="container">
        <div id="firstdiv">1</div>
        <div id="seconddiv">2</div>
        <div id="thirddiv">3</div>
    </div>
</body>
</html>
```

flex-wrap - This property is used to specify whether the flexible items should wrap or not. If the elements are not flexible items, the ~~flex~~ flex-wrap property has no effect. We can set this property to nowrap (default) and wrap-reverse.

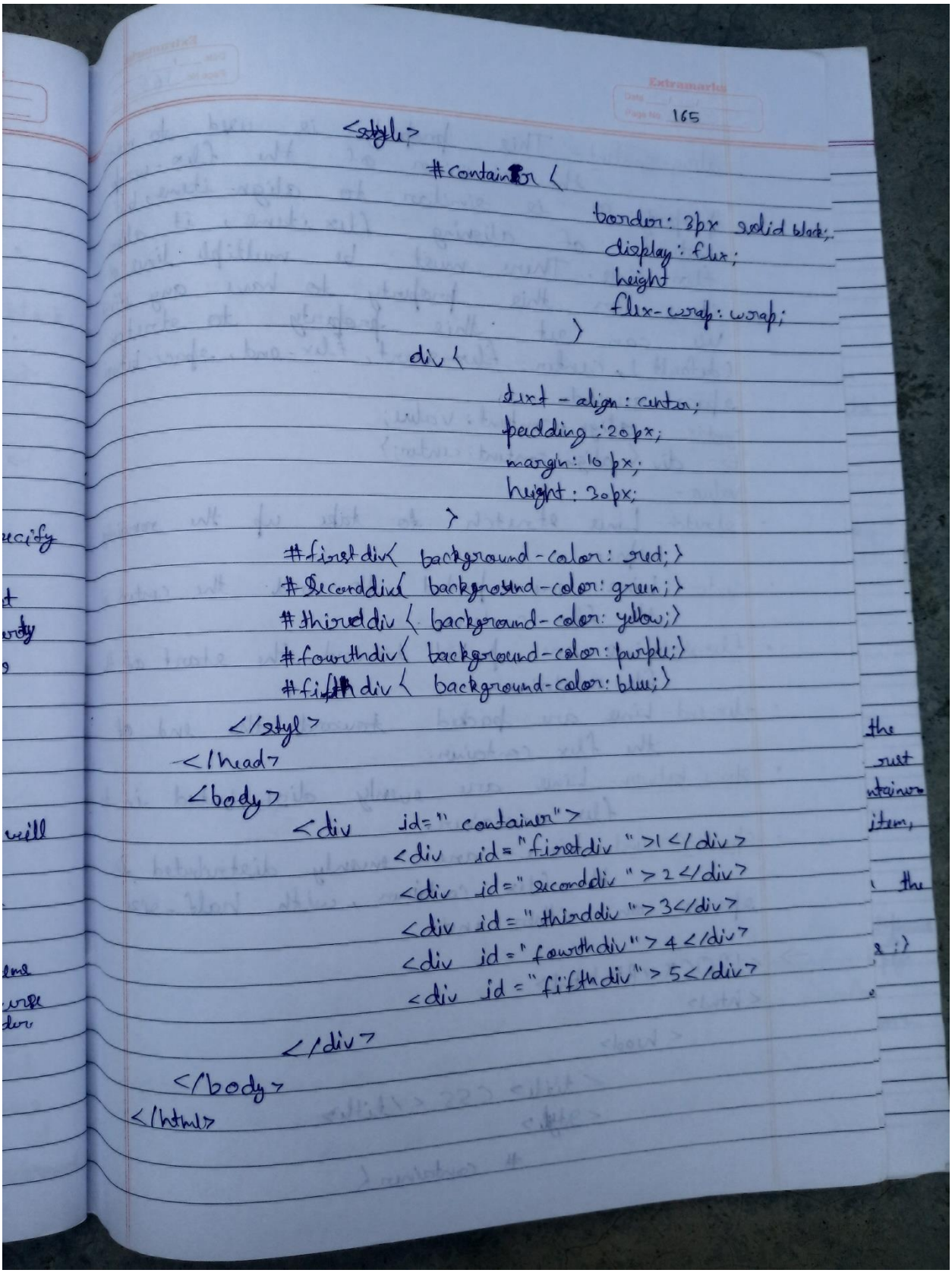
Syntax - flex-wrap: value;

Ex - div { flex-wrap: wrap; }

Value -

- nowrap - It specifies that the flexible items will not wrap.
- wrap - It specify that the flexible items will wrap if necessary.
- wrap-reverse - It specifies that the flexible items will wrap, if necessary, in reverse order.

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
```



```
<style>
    #container {
        border: 3px solid black;
        display: flex;
        height
        flex-wrap: wrap;
    }
    div {
        text-align: center;
        padding: 20px;
        margin: 10 px;
        height: 30px;
    }
    #first div{ background-color: red;}
    #Second div{ background-color: green;}
    #third div { background-color: yellow;}
    #fourth div{ background-color: purple;}
    #fifth div { background-color: blue;}
</styl>
</head>
<body>
    <div id="container">
        <div id="firstdiv">1</div>
        <div id="seconddiv">2</div>
        <div id="thirddiv">3</div>
        <div id="fourthdiv">4</div>
        <div id="fifthdiv">5</div>
    </div>
</body>
</html>
```

Extramarks
Date ___/___/___
Page No. 166

align-content - This property is used to modify the behavior of the flex-wrap property. It is similar to align-items, but instead of aligning flex items, it aligns flex lines. There must be multiple lines of items for this property to have any effect. We can set this property to stretch (default), center, flex-start, flex-end, space-between, space-around.

Syntax- align-content: value;

Ex- div { align-content: center; }

Values-

- stretch- Lines stretch to take up the remaining space.
- center- Lines are packed toward the center of the flex container.
- flex-start- Lines are packed toward the start of the flex container.
- flex-end- Lines are packed toward the end of the flex container.
- space-between- Lines are evenly distributed in the flex container.
- space-around- Lines are evenly distributed in the flex container, with half-size spaces on either end.

```
→ <!DOCTYPE html>
  <html>
      <head>
          <title> CSS </title>
          <style>
              # container {
```

```
                border:3px solid black;
                display:flex;
                hight:300px;
                flex-wrap:wrap;
                align-content: flex-end;
            }
        div{
                text-align:center;
                padding:20px;
                margin:10px;
                hight:30px;
            }
        #fdiv { background-color: red; }
    </style>
  </head>
  <body> <div id="container">
        <div id="fdiv">1</div>
        <div id="fdiv">2</div>
        <div id = fdiv">3</div>
        <div id="fdiv">4</div>
        <div id="fdiv">5</div>
    </div>
  </body>
</html>
```

flex-flow- It is a shorthand property for the flex-direction and the flex-wrap properties. If the elements are not flexible items, the flex-flow property has no effect.

Syntax- Selector { flex-flow: flex-flow flex-wrap; }

Ex- div { flex-flow: row-reverse wrap; }

Page No. 168

Order- This property is used to specify the order of a flexible item relative to to the rest of the ~~property~~ flexible items inside the same container. If the element is not a flexible item, the order property has no effect. We can set this property to number (default 0).

Syntax- order: value;

Ex- div { order: 1; }

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            #container {
                border: 3px solid black;
                display: flex;
                height: 300px;
                flex-flow: row wrap;
            }
            div {
                text-align: center;
                padding: 20px;
                margin: 10px;
                height: 30px;
            }
            #firstdiv {
                background-color: red;
                order: 2;
            }
            #seconddiv {
```

```
                    background-color: green;
                    order: -1;
                }
            #thirddiv {
                    background-color: yellow;
                    order: 3;
                }
            #fourthdiv { background-color: purple; }
            #fifthdiv { background-color: blue; }
        </style>
    </head>
    <body>
        <div id="container">
            <div id="firstdiv">About Us</div>
            <div id="seconddiv">Home</div>
            <div id="thirddiv">Contact Us</div>
            <div id="fourthdiv">Tutorial</div>
            <div id="fifthdiv">Video</div>
        </div>
    </body>
</html>
```

align-self- The property is used to specify the alignment for the selected item inside the flexible container. The align-self property overrides the flexible container's align-items property. We can set this property to auto (default), stretch, center, flex-start, flex-end, baseline.

Syntax- align-self: value;

Ex- div { align-self: center; }

Values-

auto- The element inherits its parent container's

Extramarks
Date ___/___/___
Page No. 170

align-items property, or "stretch" if it has no parent container.

- stretch - This is used to position the element to fit the container.
- center - This is used to position the element at the center of container.
- flex-start - This is used to position the element at the beginning of the container
- flex-end - This is used to position the element at the end of the container.
- baseline - This is used to position the element at the baseline of the container.

flex-grow - This property is used to specify how much the item will grow relative to the rest of the flexible items inside the same container. If the element is not a flexible item, the flex-grow property has no effect. We can set this property to number (default 0).

Syntax - flex-grow : value;

Ex - div{flex-grow : 1;}

```
-> <!DOCTYPE html>
<html>
    <head>
        <title>CSS</title>
        ~~<head>~~ <style>
            #container{
                border: 3px solid black;
                display: flex;
                height: 300px;
                align-items: flex-end;
            }
```

```
div {
        text-align: center;
        padding: 20px;
        margin: 10px;
        height: 30px;
        flex-grow: 1;
    }
#firstdiv { background-color: red; }
#seconddiv {
        background-color: green;
        flex-grow: 2;
    }
#thirddiv {
        background-color: yellow;
        ~~flex-grow~~: align-self: flex-start;
    }
#fourthdiv { background-color: blue; }
#fifthdiv { background-color: purple; }
</style>
</head>
<body>
    <div id="container">
        <div id="firstdiv">1</div>
        <div id="seconddiv">2</div>
        <div id="thirddiv">3</div>
        <div id="fourthdiv">4</div>
        <div id="fifthdiv">5</div>
    </div>
</body>
</html>
```

Page No. 172

flex-shrink- This property is used to specify how the item will shrink relative to the rest of the flexible items inside the same container. If the element is not a flexible item, the flex-shrink property has no effect. We can set this property to number (default 1).

Syntax- flex-shrink: value;

Ex- p{flex-shrink: 2;}

```
-> <!DOCTYPE html>
<html>
    <head>
        <title> CSS </title>
        <style>
            #container {
                border: 3px solid black;
                display: flex;
                height: 300px;
                width: 500px;
            }
            div {
                text-align: center;
                padding: 20px;
                margin: 10px;
                height: 30px;
                width: 100px;
            }
        #firstdiv{ background-color:red;}
        #seconddiv{
            background-color: blue;
            flex-shrink: 2;
        }
```

```
#thirddiv{
        background-color: yellow;
        flex-shrink: 2;
    }
#fourthdiv { background-color: purple; }
#fifthdiv { background-color: blue; }
</style>
<head>
<body>
    <div id="container">
        <div id="first div">1</div>
        <div id="second div">2</div>
        <div id="thirddiv">3</div>
        <div id="fourthdiv">4</div>
        <div id="fifth div">5</div>
    </div>
</body>
</html>
```

flex - This property is used to specify the length of the item, relative to the rest of the flexible items inside the same container. If the element is not a flexible item, the flex property has no effect.
The flex property is a shorthand for the flex grow, flex-shrink, and the flex-basis properties.

Syntax - Selector { flex : flex-grow-shrink flex-basis ;}

Ex - div { flex: 2,3 }

```
<!DOCTYPE html>
<html>
    <head>
```

174

```
<title> CSS </ title>
<style>
    #container{
            border: 3px solid black;
            display: flex;
            height: 300px;
            align-items: flex-end;
        }
    div {
            text-align: center;
            padding: 20 px;
            margin: 10px;
            height: 30px;
            flex: 1;
        }
  #firstdiv { background-color: green; }
  #seconddiv {
            background-color: yellow
            align-self: flex-start;
        }
  #thirddiv {
            background-color: red;
            flex: 2 3 100 px;
        }
  #fourthdiv {
            background color: blue; }
  </style>
</head>
<body>
    <div id="container">
        <div id="firstdiv">1 </div>
        <div id="seconddiv">2 </div>
```

Note-

175

```
        <div id="thirddiv">3</div>
        <div id="fourthdiv">4</div>
    </div>
  </body>
</html>
```

Media Query - A media query consist of an optional media type and zero or more expressions that limits the style sheets' scope by using media features, such as width, height and color

Syntax - ~~@media screen and (min-width:~~

@media not/only mediatype and (expressions/media features) { property: value; — ; }

Ex (i)
```
@media screen and(min-width:480px)
{
    body{ background-color: lightgreen;}
}
```

(ii)
```
@media (min-width:480px)
{
    body{ background-color: lightgreen;}
}
```

values -

- all - Used for all media type devices.
- print - Used for ~~for~~ printers
- screen - Used for computer screens, tablets, smart-phones etc.
- speech - Used for screenreaders that "reads" the page out load.

Note - If you ~~use~~ use the not or only operators, you must specify an explicit media type.

- And-
  - @media (min-width: 800px) { property: value; }
  - @media (min-width: 800px) and (orientation: landscape) { property: value; }
  - @media screen and (min-width: 800px) and (orientation: landscape) { property: value; }

- Or (Comma-,) -
  @media (min-width: 800px), (orientation: landscape) { property: value; }

- Not-
  - @media not all and (min-width: 80px) { property: value; }
  = @media not (all and (min-width: 800px)) { property: value; }

And, or (,) and Not -
- @media not screen and (color), print and (color) { property: value; }
= @media (not (screen and (color))), print and (color) { property: value; }

Media Features-

- max-aspect-ratio- The maximum ratio between the width and the height of the display area.
- max-color- The maximum number of bits per color component for the output device
- max-color-index- The maximum number of colors the device can display.
- max-device-aspect-ratio- The maximum ratio between the width and the height of the device.

- max-device-height- The maximum height of the device, such as a computer screen.
- max-device-width- The maximum width of the device such as a computer screen.
- max-height - The.
- max-monochrome- The maximum number of bits per "color" on a monochrome (grayscale) device.
- max-resolution- The maximum resolution of the device, using dpi or dpcm maximum height of the display area, such as a browser window.
- scan- The scanning process of the output device
- scripting- Is scripting (eg. javaScript) available? (added in Media Queries Level 4)
- any-hover- Does any available input mechanism allow the user to hover over elements? (added in Media Queries Level 4)
- any-pointer- Is any available input mechanism a pointing device, and if so, how accurate is it? (added in Media Queries - Level 4)
- aspect-ratio - The ratio between the width and the height of the viewport.
- color - The number of color component for the output device.
- color-index- The number of colors the device can display.
- grid- ~~with~~ whether the device is a ~~good~~ or ~~bit~~ bitmap.
- height- The viewport height.
- hover - Does the primary input mechanism allow the user to hover over elements? (added in Media Queries Level 4).

178

- inverted-colors- Is the browser or underlying OS inverting colors? (added in Media Queries Level 4).
- light-level- Current ambient light (added in Media Queries Level 4)
- max-width- The maximum width of the display area, such as a browser window.
- min-aspect-ratio- The minimum ratio between the width and the height of the display area.
- min-color- The minimum number of bits per color component for the output device.
- min-color-index- The minimum number of colors the device can display.
- min-device-width- The minimum width of the device, such as a computer screen.
- min-device-height- The minimum height of the device, such as a computer screen.
- min-device-aspect-ratio- The minimum ratio between the width and the height of the device.
- min-height- The minimum height of the display area such as a browser window.
- min-monochrome- The minimum number of bits per "color" on a monochrome (greyscale) device.
- min-resolution- The minimum resolution of the device, using dpi or dpcm.
- min-width- The minimum width of the display area such as a browser window.
- ~~monochrome~~ monochrome- The number of bits per "color" on a monochrome (greyscale) device.

179

- update-frequency - How quickly can the output device modify the appearance of the content. (added in Media Queries Level 4).
- width - The viewport width.
- orientation - The orientation of the viewport (landscape or portrait mode)
- overflow-block - How does the output device handle content that overflows the viewport along the block axis (added in Media Queries Level 4)
- overflow-inline - Can content that overflows the viewport along the inline axis be scrolled (added in Media Queries Level 4)
- pointer - Is the primary input mechanism a pointing device, and if so, how accurate is it? (added in Media Queries Level 4)
- resolution - The resolution of the output device, using dpi or dpcm.