

CSS Imp. Points (for theory refer .pdf)REVISION

\* level: 1

- use style tag in head tag only.
- color property :- (inside style)

**"element"** selector      **h1 {**  
 color: green;      Declaration  
 property      value  
**<h1> learning color </h1>**

- Types of styling :-

1. Inline styling (in body tag) → Directly

<h1 style = "color: red;"> Inline styling </h1>

2. Internal styling

```

<head>
<title> ... </title>
<style>
  h1 { color: red; }
  p { color: green; }
</style>
</head>
<body>
  <h1> ... </h1>
  <p> ... </p>
</body>
  
```

3. External styling

make file name xyz.css

→ isme lichho jo style tag mein likhle the in internal styling

⇒ Then inside head tag :-

<link rel = "stylesheet" href = "xyz.css" >

## Tags and Attributes

< tag attribute = "value" > content </tag>

DOM = document object model = structure of HTML

Comments same as of HTML.

Universal selector:- \* {

(use in style tag) color: red;

↓ }

This makes all content of HTML until & unless specified individually.

\* id and class ↗ grouping of multiple HTML elements to style them collectively. ("son of universal selector")  
 unique identifier to single HTML element

< style >

#first { color: red; }

.second { color: green; }

</style>

</head>  
 <body>

<div id = "first" > first Div </div>

D/P

First Div → red color

Second Div

Third Div } green color

<div class = "second" > . . . </div>

Shortcut: #first ↪

" " . second ↪

" " third " "

. third ↪

## group selector :-

h1, h2, h3 { color: red; }

## descendant selector:-

div p { color: red; }

p { color: blue; }

∴ <p> - . . . </p> → red color

<p> - . . . </p> → blue color

## \* Level 2

- background color :-

```
# first { color: black; background-color: red; }  
button { " " " " : blue; }
```

```
.. <div id = "first" > First </div>  
<button> Click Me </button>
```

- Color system:- ↗

### 1. RGB model (0,255)

→ ranges from 0 to 255.

e.g. color: rgb(255,0,0); → red.

e.g.

```
<div style = "background-color: green  
rgb(0,255,0);> green </div>
```

automatically text color → black

### 2. Hex color model (0,f)

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

e.g. color: # ff 00 ff ;

e.g. <div style = "background-color: #ff0000;" > red </div>.

### 3. alpha channel → ranges from 0 to 1 (0,1)

rgba(255, 0, 0, 0.0) → fully transparent

" " " " 0.2) fade

0.4) fade

!

1.0) → fully opaque

- Height + Width + clarification of class + id

<style>

```
.box { height: 40px; width: 40px; }
```

```
#box { background-color: red; }
```

```
#box2 { " " " : blue; }
```

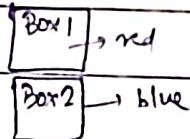
</style>

</head>

<body>

O/P

<div id="box1" class="box"> Box1 </div> <br> <" " "2" " " "2" " " .



## background image

background-image: url(-----.png/.jpg/.etc) ;  
link of image

visibility: hidden;

visibility: visible;

## \* Level 3

or left or right (e.g. Box1 written in 50px x 50px box)

text-align: center;

text-decoration: underline / overline / line-through ;

used ~~for~~ for removing underline from link.

e.g. <a href="#" style="text-decoration: none;"> Link2 </a>  
random link

.box { text-decoration: underline; }

#box1 { text-decoration-style: dashed / double / solid / wavy; }

text-decoration-color: red; → after box if text-decoration:underline  
then box1 if "—" - color: red;

text-transform: uppercase / lowercase / capitalize / none ;

line-height: 6px; → lorem ipsum dolor sit amet

line-height: 30px; → lorem ipsum dolor sit  
amet consectetur elit nema

font-size: 5px;

font-weight: 400; → dont add px, determines boldness

font-style: italic / oblique / normal ;

## font family :-

1. Search google fonts for CSS

2. Copy link rel & paste in code.

3. <link rel = "stylesheet" href="https://fonts.googleapis.com/css2?family=Dancing+Script:wght@400;700&display=block" type="text/css" />

lichha hoga website pe (copy, paste)

before  
style  
tag  
in  
read  
tag

4. .article { font-family: "Dancing Script", cursive; }

5. <div class="article" > ..... </div> <br>

## Icons

1. <script src = "....." type="text/javascript"> put in head tag, copy paste from google </script>

2. <body>

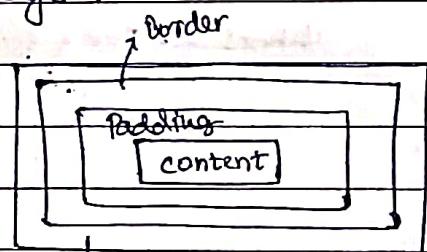
<i class = "....." ></i>

## \* level 4

### • What is Box Model?

→ Central concept in CSS that outlines the design and layout of elements on the web page.

"Mithai Bhandar & chalo"  
↓ ↓ ↓  
Margin Border Padding Content



dimension: Top right bottom left

margin

• margin do not include border in dimensions of the box.

e.g. <style>

\* { margin: 0; padding: 0; }

#button {

margin: 15px 10px 5px 20px;  
border: 2px solid black;

padding: 10px;

, background-color: aquamarine;

</style>

</head>

<body> <button id="button1" > button1 </button>

</body>

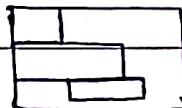
## border property :-

```
inside id "button1" { height: 100px; width: 100px;
border: 5px dashed black;
border-radius: 30px;
box-sizing: border-box;
box-sizing: content-box; }
```

## \* Level : 5

### Display property :-

#### 1. Block elements



i.e. start on a newline

Takes all horizontal space

e.g. div, p, h1, ul, li



i.e. stay in line with text

as wide as content

no new line b/w element

can't be set size easily

e.g. span, a, strong, em.

### \* e.g. style.css file :-

```
.box { height: 100px;
width: 100px;
background-color: blue;
margin: 10px;
text-align: center;
border: 5px solid black;
border-radius: 10px; }
```

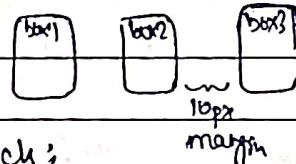
### index.html file :-

```
<head>
<title>...</title>
<link rel="stylesheet" href="style.css">
</head>
<body>
<div id="parent">
  <div id="div1" class="box">box1</div>
  <div id="div2" class="box">box2</div>
  <div id="div3" class="box">box3</div>
</div>
```

display: block / inline / inline-block;



10px margin



10px margin

O/P

O/P

• responsive website : adapt layout for different screen sizes

• relative units: px, em, rem, %, vh, vw  
 ↓      ↓      ↓      ↓      ↓      ↓      ↓  
 w.r.t.    font size    font size of    viewport    viewport  
 jo pehle likha    root element    height    width  
 hai

• % e.g. #first { height: 200px ;

width: 200px ;

background-color: aqua ;

font-size: 25px ;

#second {

background-color: violet ;

width: 50% ; → Jo pehle likha & usse relative .

height: 30% ;

emeg:-

font-size: 2em ;

if → (font-size: 2em;) →

first  
second

first  
Second

• vh/vw e.g. #first { height: 50vh ; } Jitna part dekh rhe ho uska !

width: 90vw ;

background-color: red ;



Total size  
of  
website/  
browser

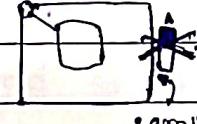
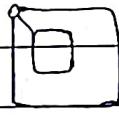
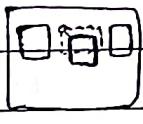
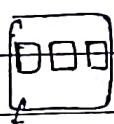
#### • 4. Position property :-

static

relative

absolute

fixed



browser

scrollbar

e.g. file in styles.css

div {

height: 40px ;

width: 40px ;

background-color: red ;

border: 5px solid black ;

margin: 20px ;

#div1 { position: static ; }

#div2 { position: relative ; }

top: 20px ;

left: 90px ;

e.g. file in index.html

<head>  
<title> ... </title>

<link rel="stylesheet" href="style5.css">

</head>

<body>

<div id="div1"> 1 Static </div>

" " " 2 " 2 Relative "

" " " 3 " 3 fixed "

" " " 4 " 4 absolute "

```
#div3 { position: fixed; }
```

```
    top: 20px;
```

```
    left: 90px;
```

```
#div4 {
```

```
    position: absolute;
```

```
    top: 200px;
```

```
    left: 200px;
```

```
}
```

- z-index : determines the stacking order of elements along the z-axis.

e.g. `.box1, .box2 { container { position: relative; } }`

```
.box1, .box2 {
```

```
    position: absolute;
```

```
    border: 3px solid black;
```

```
    width: 100px;
```

```
    height: 100px;
```

```
    text-align: center;
```

```
    font-size: 25px;
```

`.box1 { background-color: red; }`

```
    left: 20px;
```

```
    top: 60px;
```

```
    z-index: 2;
```

`.box2 { " " : aqua; }`

```
    left: 60px;
```

```
    top: 20px;
```

```
    z-index: 1;
```

### non-semantic tags

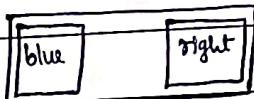
e.g. `<header>, <footer>, <article>, <section>, <nav>.`

e.g. `<div>, <span>, <i>, <b>.`

\* Level : 6

float property :-

- container { height: 110px; width: 300px; border: 1px solid black }
- box { width: 100px; height: 100px; margin: 5px; }
- box1 { background-color: red; float: right; }
- box2 { " " blue " " left; }



Flexbox : • 1-D layout for arranging items in rows or columns.

\* Items flex (expand) to fill additional space or shrink to fit into smaller spaces.

eg. \* { margin: 0; padding: 0; }

• box { height: 100px;  
width: 100px;  
border: 3px solid black;  
margin-right: 5px; }

#heading { margin-left: 200px; }

#container {  
height: 150px;  
width: 600px;  
padding: 10px;  
margin: 20px;  
border: 3px solid black; }

display: flex;

flex-direction: row-reverse;

}

#box1 { background-color: aqua; }

" 2

" 3

" 4



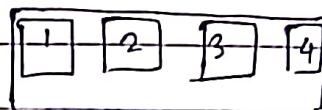
if  
width  
of container  
be 200px  
is adjusted



(after `display: flex;`)

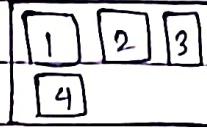
• if in place of `flex-direction:-`

1. `justify-content: space-between;`



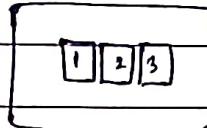
2. `flex-wrap: wrap;`

flex-wrap-reverse



3. `flex-direction: row;`

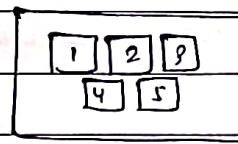
`justify-content: center;`



4.

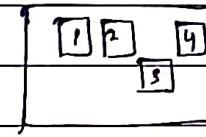
" " "

`flex-wrap: wrap;`



5.

" " "



`#box3 { background-color: yellow;`

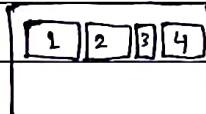
`align-self: center; }`

6.

" all

`#box3 { " " " ;`

`flex-shrink: 4; }`

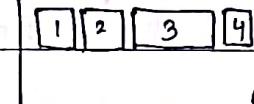


7.

" all

`#box3 { " " " ;`

`flex-grow: 1; }`



8. `#box1 { background-color: aqua;`

`order: 3; }`

`#box2 { " " " red;`

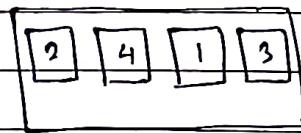
`order: 1; }`

`#box3 { " " " yellow;`

`order: 4; }`

`#box4 { " " " blue;`

`order: 2; }`



Grid :- multidimensional-layout



but flexbox is L-D layout



e.g. → • container {

display: grid;

grid-template-columns: 50px 50px;

    " " rows: 50px 50px;

}

• item1 {

    grid-column: 1/2;

    grid-row: 1/2;

    background-color: blue;

1	2
3	4

• item2 {

    grid-column: 1/2;

    grid-row: 1/2;

    background-color: red;

• item3 {

    grid-column: 1/2;

    grid-row: 2/2;

    background-color: green;

• item4 {

    grid-column: 1/2;

    grid-row: 2/2;

    background-color: yellow;

• Media Queries : tailor styles for specific device characteristics

Syntax: @media screen and (max-width: 768px) {

• container {

    amount of screen  
    to cover

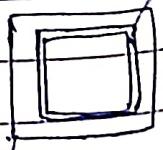
    styles to  
    apply when  
    all conditions are  
    met.

• box {

e.g. height: 100px; width: 100px; background-color: lightblue; }

@media screen and (width: 250px) {

    • box { border: 5px solid red; }



browsers

Date : \_\_\_\_\_  
PAGE NO. : \_\_\_\_\_

**VAANI**

all same  
@ media screen and (min-width: 300px){  
    box{ height: 150px; width: 150px; }

all combination, all same

@ media screen and (min-width: 250px) and (max-width: 300px){  
    box{ border-radius: 50%;  
        }

\* Level: 7