Time: 10pm - 2am Date 27-05-23

Expt. No. HTML Page.No. 12

- Meta Tags : used in search engine (SEO).
  → It helps any particular website to rank better in google or different search engines.
  → It helps to get more traffic on any website.

- HTML body

```
<!Doctype html>
<html lang="en">
<head>
    <title></title>
</head>
<body> </body>
</html>
```

- Heading tag (<h1> ..... <h6>)
  → heading tag is used for headings
- Paragraph tag (<p>) : used for paragraph.
- Lorem 4 : It will give 4 random words in VScode.
- : It is used for bold.
- : This tag is used for Italic.
-
 : It is used for new line starting.
- <hr> : It is used for horizontal raw.
- <b> : used for bold.
- <i> : for italic
- Comment : <!-- Comment -->

Teacher's Signature : _____

Boiler plate means template.

Img and Anchor Tag.

- <a> Anchor tag :

```
<a href = "https://google.com" > Go to google </a>
```

→ This will provide a link to go to google

→ for open the link in new tab we used target attribute.

```
<a href = " https://google.com" > Go to google </a>
<a href = " https://google.com" target = "_blank" > google </a>
```

→ for adding link of another html file [use '/' in href and then select file]

```
<a href = "/web-D/heading.html" > Heading page </a>
```

- <img> Image tag :

→ alt attribute used for giving message if img not loaded.

```
<img src = "sanjay.jpg" alt = "Error loading image" >
```

→ width attribute is used for width of an image. (——)

→ height attribute is used for height of an image. (|)

```
<img src = "sanjay.jpg" alt = "Error" width = "1080" height = "720" >
```

Lists and Tables

- Ordered list

→ type : It is used for choose the type of an ordered list.

→ start : This attribute is used from which the ordered list should be start.

Ex;

```
<ol type = "1" start = "5" >
    <li> Mario </li>
    <li> Contra </li>
</ol>
```

- Unordered list

```
<ul >
    <li></li>
</ul >
```

Expt. No..

- Tal
→ <t
→ <-
Exampl
→ tr
→ th
→ td
outpu
For
< fo
- lab
<
<
outp
- typ
the
- tex
<t
<

Date _____

Expt. No. _____ Page.No. _____

- Table tag <table>

→ <thead> : used for table head. [<tr>] [<th>]
→ <tbody> : used for table body. [<tr>] [<td>]

Example :
```
<table>
    <thead>
        <tr><th></th>
            <th></th></tr></thead>
    <tbody>
        <tr><td></td>
            <td></td></tr></tbody>
</table>
```

→ tr : used to create raw.
→ th : used to create heads in the table [Name, ID]
→ td : Enter the data in columns. [Sanjay, 41221148]

output :

Name	ID
Sanjay	41221148

Forms and Input Tags

```
<form action="backend.php"></form>
```

- label : It is used to give the label.

```
<label for="Name">Name:</label>
<input type="text" name="Name" id="">
```

output : Name : [ ]

- types for input : date, number, checkbox, radio [use the same name for choose one option], Email ... etc.
- textarea : for put box to write something.

```
<textarea name="about" id="" cols="30" rows="10">
</textarea>
```

Teacher's Signature : _____

```
<select name="education" id="">
    <option value="11th">11th class</option>
    <option value="12th">12th class</option>
</select>
```

Output: education : [11th class ▾] — 11th class / 12th class

- for button :

→ Submit : <input type="Submit" value="Submit">

→ reset : <input type="Reset" value="Reset">

Inline & Block Elements : They acquired only required space.

- span : It is used for continue from same line.
- strong : used for bold

ID's and classes in HTML

→ id : It is an identifier and it can give only to one element bcoz it can't be same.

→ '.' for class
'#' for id

Ex: `<div id="mainbox" class="redbg">`

Note: we can give multiple class to one element by giving space

`<div class="redbg blackborder">`

HTML Entities

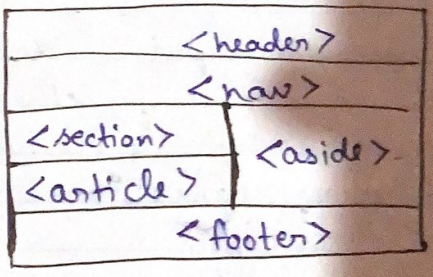
-   : It is used to give a space of one character.

→ to write <p> on web <p>
we use & for write symbols like :-
- &pound - £
- &copyright - ©

Semantic Tags

```
<details>
    <summary> Name </summary>
    I'm Sanjay Verma
</details>
```

Output: ▶ Name
▼ Name :
I'm Sanjay Verma



Time : 9:30pm - 12am

Date 28/5/23

Expt. No. CSS

Page.No.

CSS : Cascading Style Sheet

- It gives style to raw HTML
- used to give style to our web-page and make it responsive.

Syntax

selector (where) → property (This) → value (set this value)

```
p { color : blue ; }
```

→ Three ways to add CSS to the markup(html code).

1. directly using style attribute
2. Internal : kept inside head tags in <style> tags.
3. External : .css style sheet.

Inline, Internal & External CSS

- Inline CSS

```
<p style="color : red ; ">Mr.adroit</p>
```

- Internal CSS

→ In head tag

```
<style>
    p{ color : red ; }
</style>
```

- External CSS

```
< link rel="stylesheet" href="External.CSS ">
```

Note : Last command execute first, But if you want to execute particular command first write '!important'.

Selectors in CSS

They are used to find the element whose property will be set. They are used to target the html elements. It makes easy for us

'*' → It is an universal selector.

Teacher's Signature : _____

- Types of selectors :-
  → CSS element selector → id selector → class selector
  → CSS grouping selector.
- /* comment */ → this comment is used in CSS.

→ element selector : p { color : red ; }
→ ID selector : #secondPara { color : purple ; }
→ class selector : .Third { color : pink ; }
→ class grouping selector : footer, span { Color : yellow ; }

CSS Fonts

```
p { font-family : 'Courier New', Courier, monospace;
    font-size : 30px;
    line-height : 7.3em;
    font-weight : bold;
    font-style : italic;
  }
```

CSS Colors

```
Color : red ;
color : rgb ( 8 , 9 , 71 )
color : #ff4532
```

Borders and Backgrounds

```
p { background-color : yellow;
    border-width : 4px ;
    border-height : 8px ;
    border-style : solid ;
    border-top : 2px solid voilet ;
    bord-top-left-radius : 50px ;
    background-repeat : no-repeat ;
    background-position : center center ;
  }
```

..... etc There are many

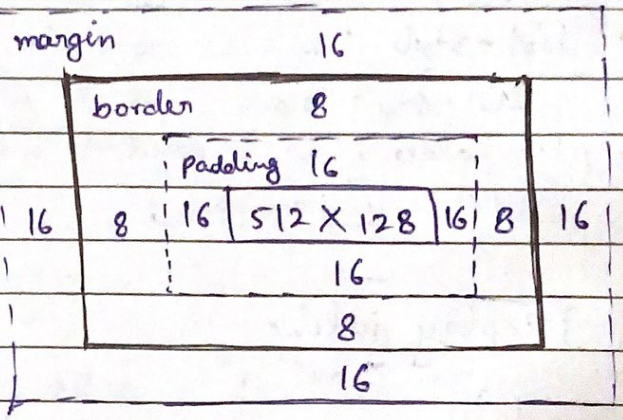
→ background : url ('img/logo.jpg'); → used to set image as background

Time: 10pm - 12:15am Date 29-05-23

Expt. No. Page.No.

Box Model

padding = 79px;
padding top = 56px;
/* padding = top right bottom left; */
padding = 23px 56px 6px 9px
/* top & bottom | left and Right */
/* top | left and Right | bottom */
/* y & x */



Aligment (Float and Clear)

float: It is used to align left or right
float: right;

clear: It is used to not allow the floating element to overlap.
clear: left; clear: both; clear: right;

text-align: It is used to align the text.
text-align: right; text-align: justify;

Styling Links & Buttons

- cursor: to add the cursor style. (cursor: pointer;)
- text-decoration: to decorate the text.
  text-decoration: none; → It will remove underline from hyperlink.
- hover: It is used to change the color when pointer
  Ex: a:hover { color: red;
  background-color: pink; }
- visited: used for visited link (~~a:visi~~ a:visited { })
- active: typically starts when the user presses down the primary mouse button.

pseudo-class

<button></button> : for creating button.

Teacher's Signature :

Navigation Menu

- <nav></nav> : It is used for navbar
- list-style : It is used for style the list
  list-style : none → It will remove the bullet point from list
- placeholder : It is used to write the msg in textbox for showing.
  Example: Enter Your Name.

Display inline

display : inline ; → It will take only required space.
display : block ; → It helps to set width

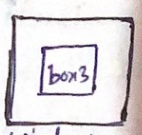
Display Property

box-sizing : border box ; → total width including margin and padding.
display : inline-block ;

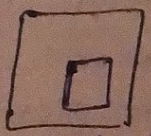
Positions

→ Relative position : relative to its normal position.
when you need a gap on its original position.
by this we can set top, left, right, bottom positions.

Ex: #box3{ position : relative;
top : 34px;
left : 34px; }



window


top and left

→ Asolute position : relative to its parent position.

```
<div class="container">
    <div class="box" id="box1">1</div>
    <div class="box" id="box2">2</div>
</div>
```

- Here the parent is container for the box 1 and 2 so they change position relative to container position.

Time: 12pm - 4pm Date 30-05-23

Expt. No. Page.No.

→ fixed position: It will fix the position, means the position will not be changed even we are doing scrolling.
→ sticky position: It is used to stick the element which will not be moved even we are scrolling.
→ static position: Default position.

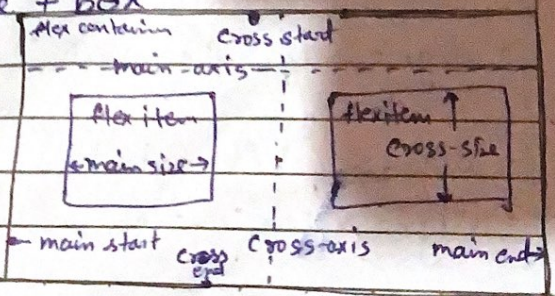
Visibility and Z-index

- div.box*4 = Here $ is used to give the incrementation of no. to the boxes by 1, 2...
- display: none; will hide the element and the space
- visibility: hidden; will hide the element but will show its empty space.
- Z-index:
  → It will work only for position: relative, absolute, fixed or sticky;
  → It specifies the stack level of the box in current stacking.
  → In simple z-index work as a layer which layer should come to front and which will be on backside.

```
z-index: 34;   } highest z-index will appear on
z-index: 30;   } upper layer.
```

Flexbox: is a one-dimensional layout method for laying out items in rows or columns.

- css flexbox is a better way to align items into a container. Flexbox = flexible + box
- display: flex; used to make the flex container.
- flex-direction: row; it will show the items in row.



Teacher's Signature:

→ flex properties for a flex container.

- flex-direction : column; It will show the items in a column.
- flex-direction : row-reverse; It will reverse the rows.
- flex-wrap : wrap; It controls whether the flex container is single line or multi lines, and the direction of cross axis.
- flex-flow : row-reverse wrap; It is used to give flex direction and flex-wrap.
- justify-content : center; to center the flex items.
- justify-content : space-between; to give space between items
- justify-content : space-evenly; to give space between items and the same space at the starting and ending.
- align-items : center; for vertical center.
- align-items : flex-end; for bottom
- align-items : flex-start; for top
- align-items : stretch;

→ flex properties for a flex items.

- order : 2; give the order to the items higher the order, later it shows up in the container.
- flex-grow : 2; It is used to give the space to the item, like in this the item will take space of two items.
- flex-shrink : 2; To shrink the item
- flex-basis : 320px; to control the size of an item.

→ when flex-direction is set to row flex-basis : will control width same as column.

flex : 2 (↑ grow) 2 (↓ shrink) 320px (↔ basis);

- align-self : flex-end; It will send the item in the flex end.
- align-self : flex-start;
- align-self : center;

Date _____

Expt. No. _____ Page.No. _____

Responsive Design and Font Units.

em, rem, vh and vw units.

→ em : It is used to give the font size, n times of its parents. Ex: font-size : 10em; → It will increase 10 times.

Note : font-size increase 10 times according to their parent class like parent font-size is 3px so this will be 30px.

Note: padding : 3em → If we write padding after font size then it will not follow parent it will increase 3 times of font-size means it will be 90px.

→ rem : It is follow the size of html.

```
<html lang="en">
<style>
        html { font-size : 25px } </ style>
#second { font-size : 3rem; } → It will be 3x25 =75px.
```

→ vh : It is used to take full height of the window.

Ex: height : 100 vh;

→ vw : It is used to take full width of the window

Ex: width : 100vw;

Media Queries : It uses the @media rule to include a block of CSS properties only if a certain condition is true.

```
@media (max-width : 300px) { #box1 {    } }
@ media (min-width : 300px) and (max-width : 500px) {
   #box 1 { display : block;
            background-color : blueviolet;
          }
}
```

@media only screen and (max-width : 300px) → It is used when we want to support only screen media types.

Teacher's Signature : _____

Time : 12:45 pm - 6:00 pm

31-05-23

Notes :- Practice

- background-size : cover; → to cover the background size.
- margin : auto; → for center
- display : inline-block; → for take only required space.
- filter : invert (100%); → to invert the color.

More about Selectors

- div li p { } → if p is contained by any li and li is contained by div.
- div > p { } → if p is contained directly by div
- div + p { } → if p is next after div

Attribute & nth child pseudo Selectors

- input [type = 'text'] { } → used for style the input which has a 'text' type.
- <style> a [target] { } </style> → used for style the anchor tag which has a target.
- a [target = '_self'] → It will style the target which has '_self'
- <style> li : nth-child (3) { } </style>
  → It will style the 3rd no. li
- li : nth-child (2n+0) { } → style every 2nd li
- li : nth-child (even) { } → style every even no. li
- li : nth-child (odd) { } → style every odd no. li

Before and after Pseudo Selectors

- header :: before { content : " " } → It will add the content before the header (if header is present in body).

To add multiple thing before anything like you want add something before section.

```
section :: before { content : " ";
                    position : absolute;
                    width : 100%;
                    height : 100%; }
```

- same as :: after { content : " " } → to add content after something.

Date

Expt. No. Page.No.

• To control opacity we use → opacity : 0.5;

Box Shadow & Text Shadow

→ Box Shadow positive no. for bottom -ve no. for top.

box-shdow for bottom ; • box-shadow : 10px 13px red;

• box-shdow : 10px 13px 30px red;

• box-shadow : 10px 13px 30px 40px red;

↑ offset-x ↑ offset-y ↑ blur-radius ↑ spread-radius

→ for inside the box

• box-shadow : inset 8px 10px red;

→ for multiple box-shadow we use comma ',' .

• box-shadow : 10px 13px red, 13px 16px purple;

→ Same as for Text-Shadow

• text-shadow : 2p 3px red;

Note: text-shadow has the same properties of box-shadow.

CSS variables and Custom Properties.

→ Local variable

we declare local variables in CSS using '--'.

Ex: .container { --box-color : green;

background-color : var(--box-color);

and we access variable using var(variable-name);

→ Global variable

we declare global variable using ':root { variables }'

Ex: :root { --primary-color : green;

declaring variable → --maxw : 333px; }

.container {

accessing variable → background-color : var(--primary-color);

max-width : var(--maxw);

}

Teacher's Signature :

Time: 7pm - 9pm

Animations & Keyframes

- declare position position : relative ;
- animation-name : sanjay-2; → to give the name of an animation.
- animation-duration : 2s; → time period of the animation.
- animation-iteration-count : 1; → How many times animation will play.
  - ↳ infinite → for infinity count
- aimation-fill-mode : forwards; → to stop animation after it ends without going to its original form.
- aimation-timing-function : ease-in; speed of animation : starts slowly ends fast.
  - ↳ ease-out → starts fast ends slowly :
  - ↳ ease-in-out → starts and ends slow and mid will be fast.
- animation-delay : 3s ; → After how much time animation should start
- aimation-direction : alternate ; → give direction to animation.

Comes in selectors body.

Note : animation can also write this (shorthand property)

animation : animation-name animation-duration animation-timing-function animation-delay animation-iteration-count animation-fill-mode;

Ex: animation : sanjay 5s ease-in 1s 12 backwards ;

Note for animation we use Keyframes

Ex:
```
.box { background color : green;
       animation-name : sanjay;
       animation-duration : 2s;
       animation-iteration-count : 1; }

@keyframes sanjay { for { width : 200px; }
                    to { width : 900px; } }
```

animation name ← sanjay

OR

```
@keyframes sanjay { 0% { top : 0px;
                         left : 0px; }
                    25% { top : 250px;
                          left : 0px; }
                    75% { top : 250px
                          left : 0px }
                    100% { top : 0px
                           left : 250px } }
```

Date _____

Expt. No. _____ Page.No. _____

Transitions

first we will write transition-property : background-color ;
as in this we have write background-color for transition-property, So here transition will only apply on background-color.
→ To apply all the elements of hover we write
transition-property : all ;

- transition-duration : 1s ; → for time-period of transition.
- transition-delay

⇒ Transition shorthand property (transition-timing-function, transition-delay)
transition : all 1s ease-in-out 0.3s ;
(1s = transition-duration)
↳ name on which you want to apply transitions. Here it will apply on all.

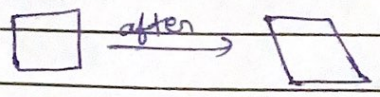
Example :
```
#box { display : flex ;
       width : 200px ;
       height : 200px ;
       transition : all 1S ease-in-out 0.3s ; }

#box : hover { background-color : white ;
               height : 400px ;
               width : 400px ;
               border-radius : 100px ;
               font-size : 30px }
```
there the transition will apply on all elements of hover

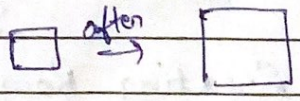
Transform Property

Transform property is used for transformation.

- transform : rotate (360 deg) ; → this will rotate it 360°.
- transform : skew (40 deg) ;
this will look like this



- transform : scale (2) ;
It will increase the size two times



- transform : translateX (123px) ;
In this it will change its position at X-axis by 123px.
- transform : translateY (123 px) ;

Teacher's Signature : _____

→ to translate on x and y both.

- transform : translate (123px, 123px);

Example:

```
.box { display : flex;
       justify-content : center;
       align-items : center;
       ;
       transitions : all 0.5s case-in-out; }
.box : hover { transform : rotate (360deg);
               transform : skew (40deg);
             }
```

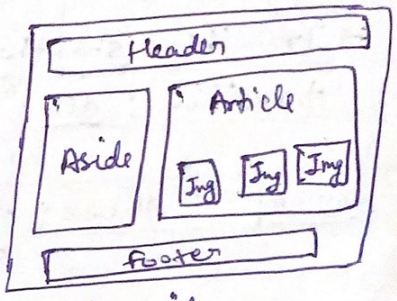
we put transition to see the transformation.

CSS Grids → Layout mechanism

to initialize grid : display : grid;

- grid-template-column : 300px 100px 100px;

It is used to set the width of the columns in pixel.



use of grid

- grid-template-column : 300px auto 100px;
- we can also write grid-template in this form:
  grid-template-column : 1fr 4fr 1fr;
- grid-template-column : repeat (3, auto);
  to set the no. of columns, like in this there will be only 3 columns and the width will be auto.
- grid-gap : 2rem;
  It is used to give the gap between columns.

Example:

```
.container { display : grid;
             grid-template-column : repeat (3, auto);
             grid-gap : 2rem;
           }
```

Grid - Creating Rows and Gaps

- Grid-auto-rows : 3fr;

This will distribute the width in 3fr for every rows.

↳ grid-row-gap : 1rem; ↳ grid-column-gap : 2rem;

Time : 6pm - 8:30pm Date 01-06-23

Expt. No. _____ Page.No. _____

Spanning Rows and Columns in Grid

- grid-column-start : 1; } This will span the column
- grid-colum-end : 3; } from 1 line to 3 means 2 columns.
- grid-row-start : 1; } This will span the rows
- grid-column-end : 3; } from 1 line to 3 means 2 rows.
- grid-column : 1 / span 3 → this will span 3 columns 1, 2, 3
- grid-row : 1 / span 3 → this span 3 rows 1, 2, 3.

Example :

use 'first-child' for spanning

```
.box : first-child {
    grid-column-start : 1;
    grid-column-end : 3;
    grid-row : 1 / span 3;
}
```

Autofit and Minmax

- grid-template-columns : repeat (auto-fit, minmax(300px, 1fr));

→ It is used for responsive design.

→ by using auto-fit and minmax, you can set the min. width and max width.

Layouts in Grid

Layouts using grid-template area.

Ex: grid-template-areas :

```
'navbar navbar navbar'
'section section aside'
'sanjay sanjay sanjay'
;
```

} These we have declared 3 rows and three columns with grid-areas name.

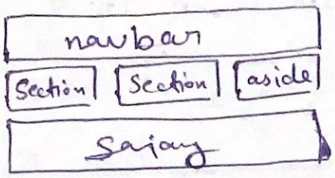
we set name for grid-area

```
# navbar {
    grid-area : navbar
}
```

Teacher's Signature : _____

```
#section { grid-area : section ; }
#aside { grid-area : aside ; }
#footer { grid-area : sanjay ; }
```

Output :



Media Queries using CSS_Grid

Example :

```
@media only screen and (max-width : 300px) { body {
        background-color : red; }
    .container { display : grid;
     grid-template-areas :
     'navbar navbar navbar'
     'aside aside aside'
     'sanjay sanjay sanjay';
    }
   span { display : block ;
      }
   aside { display : none ;
      }
}
#aside { grid-area : aside; }
```

→ creating media query

using grid in media query

This will block the display of span content.

This hide the aside content