

## What is CSS

Page No.	
Date	

- CSS Stands for Cascading Style Sheets
- If HTML is the Structure of the house then CSS is the look and feel of the house
- It's the language to make our pages Presentable
- Designed to make style sheets forall web.
- Now let's try to break the acronym :

Cascading : Falling of Styles

Style : Adding design / styling our HTML tags

sheets : Working our tags style in different documents

→ is a style sheet language used to describe the presentation of a document written in HTML

→ CSS describes how elements should be rendered on screen, on paper, in speech or on other media

# History

Page No. \_\_\_\_\_

Date \_\_\_\_\_

1994 : First proposal by Hakon Wium Lie on 10th October

↓  
1996 : CSS was published on 17th November  
with influences Bert Bos co-author

Later he became Co-author of CSS

1996 : CSS became official with CSS was  
published in December

1997 : Created CSS levels 2 and 4 in Nov,

1998 : Published on 12th May.

## Editors :-

- Atom
- Brackets
- Espresso (Mac OS X)
- Notepad++ (Great for HTML and CSS)
- Komodo Edit (Simple)
- Sublime Text

## CSS in PW Skills

Page No. \_\_\_\_\_

Rating \_\_\_\_\_

↓ mainly used for style

→ CSS is a stylesheet language used to describe the look and formatting of a document written in HTML. It is used to define the styles of elements on a web pages, such as

- Font
- Color
- Size
- Layout of texts and others.

CODING BUGS NOTES GALLERY

## # Why CSS important for Web Developers?

- Styling and Responsive Design

Responsive to change in Size of Dimensions

CSS gives  
to support

- Separation of Concern and appearance,  
→ not creating in only single file by giving some quality
- Reusability and animation

I have ID  
and Class

Animations

Support have

With user  
in multiple  
time

→ Selecting  
multiple  
elements

Pi,

Pg,

hi {

Color : red ;

must use  
commas

apply  
rule  
to all  
of them.

## # Reasons to learning CSS



- Fast Page Speed

Page No.	_____
Date	_____

(load time)

- Better UX and Quick Dev time

(maintain)

- Easy formatting changes and Compatibility across devices. → works compatible you can open works different devices

Bonus tips :- Create more thins as possible, use uppercase your command on CSS

CODING BUGS NOTES GALLERY

## Anatomy of a CSS

Format

Selectors

Structure  
Control

P { → each Selector wrapped in curly braces }

Color: red;

Property  
(key)

Property Value

contain many choices

You can choose which properties you want to effect in the rule

Declaration

24 Jan

Page No. \_\_\_\_\_

Date \_\_\_\_\_

# ① Different ways to bring CSS into the HTML file.

Note :- In Programming world every name is given with sense of meaning, so if you feel its not understandable then search on google translate, slowly slowly you will understand that.  $\Rightarrow$  Fact

- (a) Inline styling
- (b) Internal styling
- (c) External styling

## Inline

inline styling is a technique in HTML that allows you to apply style directly to an HTML element using the style attribute. It is called "HTML" "inline" because the style is applied directly to the element, in the same line of HTML code.

```
<p style = "color: blue; font-size: 24px; border: 1px solid black; padding: 5px; margin: 10px; border-radius: 10px; background-color: #f0f0f0; text-align: center; width: fit-content; margin-left: auto; margin-right: auto;">Hello</p>
```

→ When you write code inside the HTML code — called inline

→ 3rd element of 3rd

(b)

## Internal Styling



Internal styling refers to the practice of using a `style` element within the head of an HTML document to define styles for the elements on the page. The `style` element should contain a list of CSS rules that specify the styles for the elements on the page.

internal internal → 3rd file or 3rd  
inside the → head  
CSS (HTML)

(c)

## External Styling

External styling refers to the practice of linking to an external CSS file from an HTML document using the `link` element.

The `link` element should be placed within the head of the HTML document, and it should have a `rel` attribute with a value of "stylesheet" to indicate that it is linking to a style sheet.

only use wrens  
you need give ~~unval~~  
style so the particular element.

Page No.			
Date			

inline :-

### Single element

- If you have to give style in one particular part, then use inline, but if the same thing requires in multiple part so, inline is not a good method to use.

#### Advantage

- If you use all styling (inline, internal, external) then the highest priority go the inline, if inline absent then it goes to external.
- Inline is easy to apply
- There is no need to create an additional file.

#### Disadvantage :-

- difficult to update
- no provide browser cache advantage
- pseudo classes / classes cannot be styled with
- not versatile — Can't be applied elsewhere
- if use it one place, then can't use that change in another place.

(b)

## Internals in detail

- The internal CSS is used to add a certain style from a single document.
- It is defined in `<head>` section of the HTML page inside the `<style>` tag. → used because tells the browser, we are now using internal CSS

### Advantages:-

- ① ID and classes can be used, if you need to change, then apply it, no need to go on whole code of HTML
- ② you do not need to upload multiple file

### Disadvantages :-

- ① at once if they are useful for only the page they are specific on. → you have write same style in all pages  
→ you can't not update automatically others page.
- ② Internal style sheets increase page load time. bulky — hard to understand code

⑥

## External CSS

not forget  
the single element.

Page No.			
Date			

- When you need to make changes to several pages, you often use the external style sheet.
- It uses the `<link>` tag on every page and the `<link>` tag should be put inside the `<head>` section.

Example :-

`<head>` → Link: CSS - tag

`<link>`

`</head>`

# Selectors in CSS

Select ~~HTML~~ Tags in

(1)

## Part 1.

Why we use Selectors?

→ HTML Selectors are so long, in real life,  
so it's not possible or wise to give a tag to every element to beautify.

Class is most use Selector in CSS

Selectors :- help you to manipulate HTML code,

We will be using Internal CSS to demonstrate CSS Selectors

→ If you want give different ID in two `<p>`, `<p>` different tag, so Selector solve this problem effectively

### Simple Selectors

(1)

Types of  
Selectors  
(most used)

(2) Combinations

(3) Attribute "

(4) Pseudo - class "

(5) Pseudo - elements "

not used

mostly

rarely

deeply

Note :- CSS Selectors cannot be used with inline CSS.

## ① Simple Selectors :— pretty straight forward.

Page No.			
Date			

most used

- 
- ① Universal " " →
  - ② Element " " → h1, p, etc
  - ③ Class " " → (class = "") = . prior { } not good in long
  - ④ ID " " → # (unique) implement more as more
  - ⑤ Selector list → = # id's multiple class
- Ex :- .class1, .class7, .class10 { } not used in multiple places.

## Part 2

### ② Combinators

between the Simple Selectors  
contain more than one Simple Selector.

almost same

- 
- ① Descendant → parent child selection main > p = main p { }
  - ② Child → h1 > p { } child Selectors are not parent Selectors
  - ③ Adjacent Siblings → section + p { } Section at first (अंतिम) then next Section (नेक्स्ट)
  - ④ General → section ~ p { } (2nd next)
  - ⑤ Section →

### ③ Attribute Selectors (+)

→ [+] { } not use generally

### ③ Explore and Bring in fonts :-

Page No.	_____
Date	_____

- in starting we all are using → default fonts  
but if you want specific (poppins etc) so, then  
how to use it.
- So, How to apply // to another browser / website,  
then take CDN and apply in your  
HTML code. → Content delivery Network,  
~~(addreses)~~

Font :- whatever we write text

Note :- 99% developers using Google fonts  
→ Google fonts CDN have trust  
→ speed in loading

always remembers 2 things :-

- ① fonts in meta head in HTML
- ② fonts in use part in mpf.css  
file in which in link at all part of

google  
fonts.

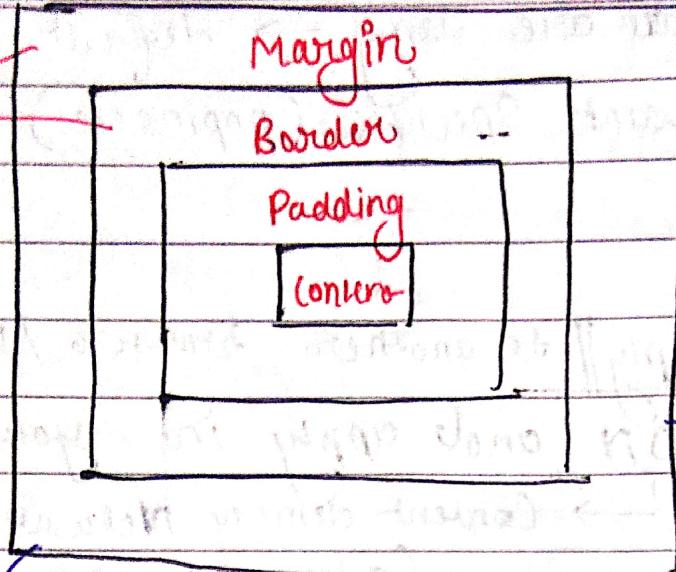
First link :- tell the browser  
use  
Second link / tell the .css file

2  
4

## Box Model

Page No.	
Date	

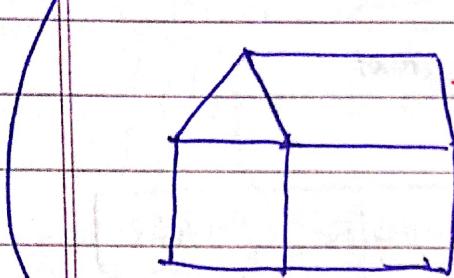
take very less space



→ term "box Model"

is used when discussing a web application's design and layout.

→ box that wraps around every HTML element



→ Bricks → Wall → margin, Border etc  
(mixture of this make home (content))

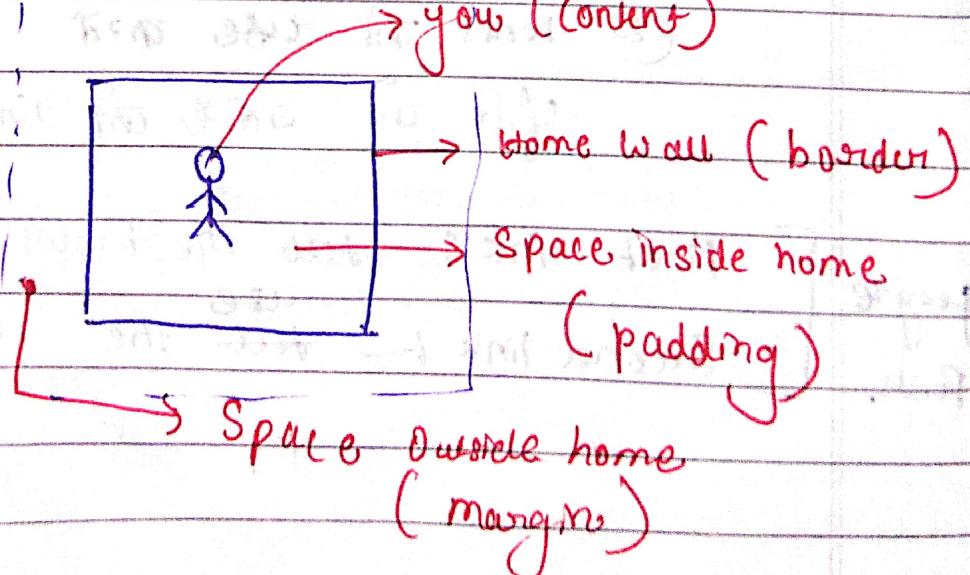
inspire → user agent Style sheet

clockwise

understand how it work

Also should know how to calculate! —

350px



③

## Colors and Styles in CSS

Page No. \_\_\_\_\_

Date \_\_\_\_\_



↳ How to make beautiful colors in your browser  
to beautify your web page.

Color → give great identity the brand, emotion,  
mood etc.

google browser variation Red = mozilla

(chance to show diff)

# There are different ways of taking colors  
in your website.

Problem in that, every  
browser, treat different

using Predefined Color names colors different  
of same name.

① " hexadecimab Colour Codes (99.9% percent used)

② " RGB Values

③ " HSL Values

④ " the "rgba" function

⑤ " the "hsla" function

⑥ " the "transparent" keyword →

also used  
most highly

Backgrounds in CSS

also

explore  
this

11

## Polarity in C8S

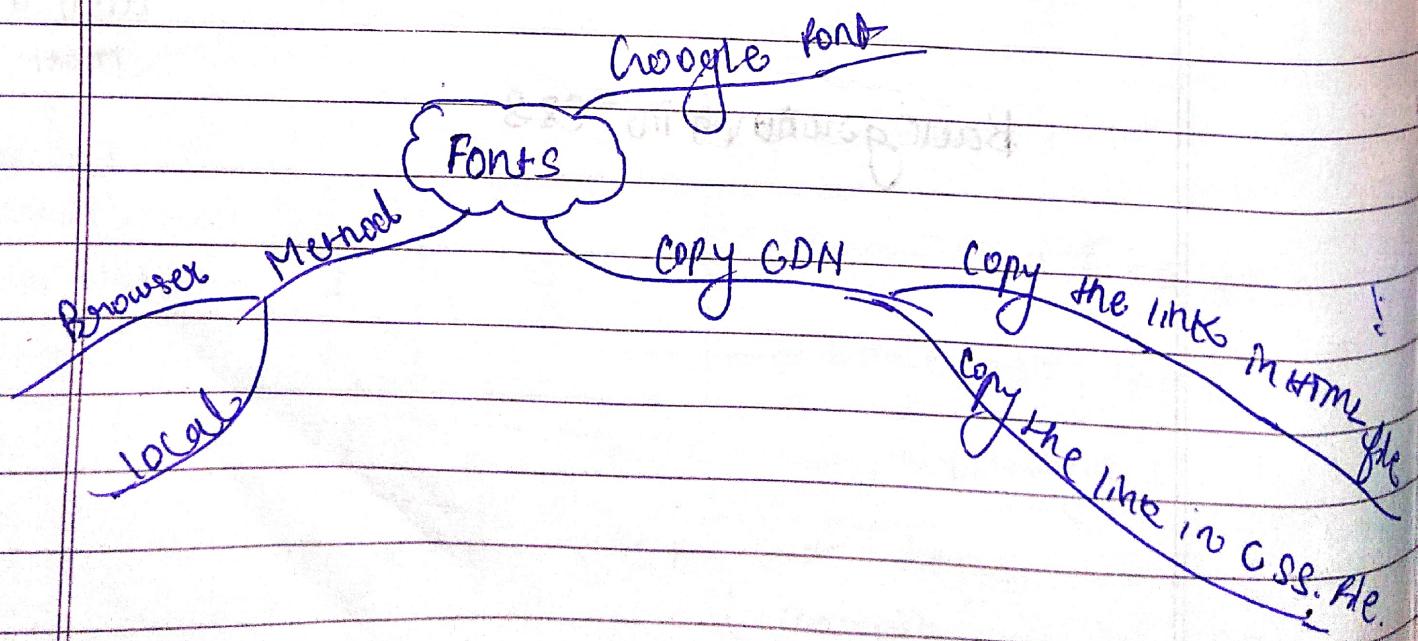
Page No. \_\_\_\_\_

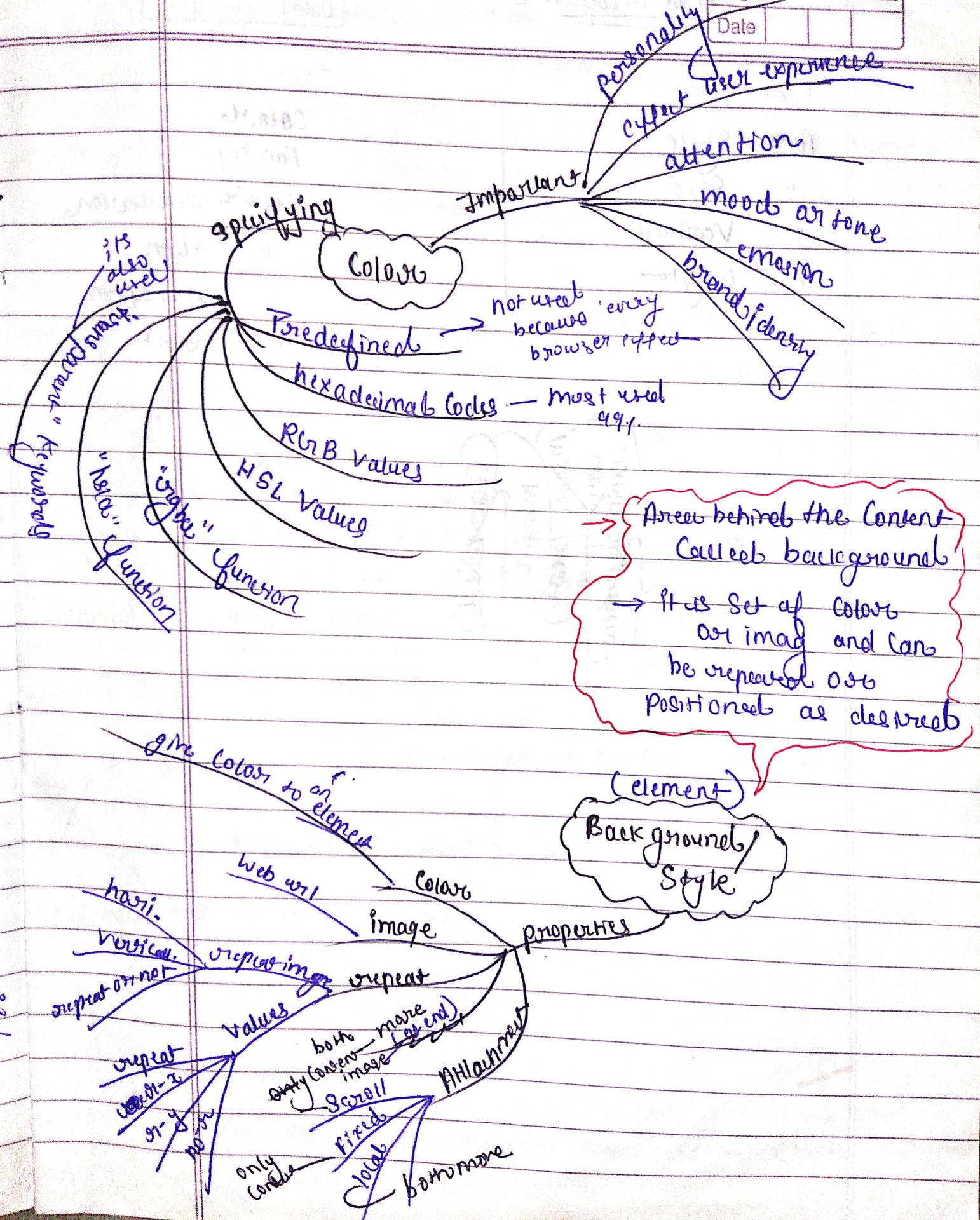
First priority → inline (own parents)

seconds → intervals (our relative)

third " → External (our Society)

if you  
give  
some think





Margin → Create extra space around an element

Page No.			
Date			

Padding → Create extra space within an element

Margins

- You have more control over all four margins (margin area)  
→ [top, right, bottom, left]

margin-top

margin-right

margin-bottom

margin-left

Negative values are also allowed

(default)

auto → When the browser calculates margins

length → Specifies → px, pt, cm etc / Fixed value

% → Width of the containing element

inherit → parent elements

An element's padding creates the space b/w its content and its borders.

→ top, right, bottom, left

Padding

→ Padding -> top

" - top

" - bottom

" - left

Padding and element width → Understand the clash

→ (watch video)

Page No. \_\_\_\_\_

Date \_\_\_\_\_

## [Height and Width]

- Widths ← min  
= max  
→ widths by default sets the content area

→ but manually you fix by width by

px, em, %, auto (default)  
length → image, text

→ keyword values :- max-content | Global → inherit  
min - " | initial  
max - " | revert  
width - " | unset.

• Height

stable point E

(horizontal) 10px

width 30px

min-width

max - "

Height (vertical)

min-height

max - "

Box-sizing

Hole :-

overflow

if the content over then, hidden that

Positioning the main page

Centering the img

• my-classes

→ styling the body  
→ changing the page  
→ color (HTML)

Page  
HTML

→ box model  
→ CSS layout

→ four and two  
→ CSS : box all

→ types of S

→ subset

CSS rules

All HTML  
elements of  
the specified  
type.

< p >

each id  
value should  
be unique

# my-id

Selectors

ID

multiple  
instances of  
the same  
class can appear  
on a page.

use  
specified  
attribute

img [src]

The specified element(s)  
but only when on the  
specific state,

Part when a cursor is  
hovering over a link

pseudo-class  
(eg: :hover)

Attributes

Class

=> a : hover

## Display Property

Change the clear default  
behaviour of HTML  
Page No. \_\_\_\_\_

Date			
------	--	--	--

- The Display CSS Property Sets whether an element is treated as block or inline elements and the layout used for its children, such as → flow layout  
→ grid  
→ flex
- Display Property sets an element's inner and outer display types.

Outer type sets → flow layout

Inner display set → children

✓ PW

<P> → ~~block~~ block element

Inside style

P {

Display: inline;

}

Change into  
block

< Span >

Inside style

Span {

Display: block;

{

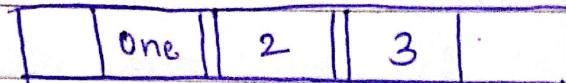
Change into  
inline

**impacts**

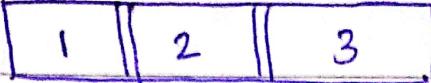
- Display : None → not visible in browser
- Display : hidden → watch the video
- Display : inline-block
- Display : inline

Page No.		
Date		

Display : block ;



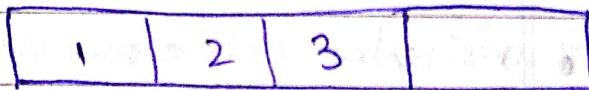
Display : inline-block ;



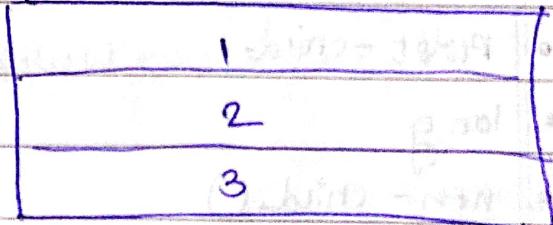
" : none ;

blank

" : flex ;



" : grid ;



Must  
read  
Ppt

(जो पूर्णांतर न हो)

## Pseudo Classes — [::]

Page No. \_\_\_\_\_

Date \_\_\_\_\_

→ Some work is done by other CSS elements, then the discovery come is called — Pseudo classes

Most frequent used Selector in PC —

- hover → button and link *(cursor tag)* → happens when visit
- focus → change in small area → generally used on input
- link → underline links
- visited → how you clicked *there*
- active → happen when click Ex: Google

nth-child()  
1, 2, 3  
(1)  
(2)  
(even)  
(odd)  
 $(2n+1)$   
 $2n+1$   
= 3

Syntax : —

Selector : pseudo class {

Property : Value ;

Can be used  
to style a  
specific  
part of  
an element

Ex : P : hover {

apply  
when  
you

bgc : black ;

color : white ;

{

go to  
the  
content

→ Pseudo-classes are used to Select elements based on their state, such as when the element is hovered over, when it has focus or when it is the first or last child of its parent element.

CODING BUGS NOTES GALLERY

## Pseudo Pseudo-elements [::]

is a keyword added to a selector that lets you style specific parts of the selected elements.

Ex : — change the first line of `<p>`

`P::first-line {`

→ Can be used

Colour : blue ;

to style an

text : uppercase ;

element based

}

On its state.

Syntax =

`Selector :: Pseudo-element {`

`Property : Value ;`

only use  
One Selector  
at a time



CSS allows developers to create special types of selectors called pseudo-elements, which can be used to style specific parts of an HTML element.

Page No.			
Date			

Some are :-

::first-line → change first line

::first-letter

::after → change → add msg = "Priyanka"

::before → opposite of after

::marker

::selection

# Difference in both

# CSS Specificity.

## Gradient

use < gradient >

Page No.			
Date			

→ The < gradient > CSS data type is a special type of < image > that consists of a progressive transition between two or more colors.

↳ Smooth transitions between two or more specific colors.

- ① Linear (goes down/up/left/right/diagonally)
- ② Radial (defined by the center)
- ③ Conic (rotated around a center point)

### Linear



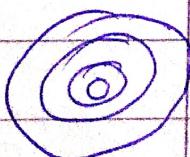
Set starting point and a direction (or an angle) along with the gradient effect.

### Radial



- It is defined by its center
- to create a ~~smooth~~ radial gradient you must also define at least two color stops

### Conic



- to repeat radial gradients

# Transition

→ hover के बाद - कोई animation नहीं।

Page No. \_\_\_\_\_

Date \_\_\_\_\_

The Transition is CSS Property is a shorthand Property for : all the

transition - Property - all

transition - duration - 0s (speed)

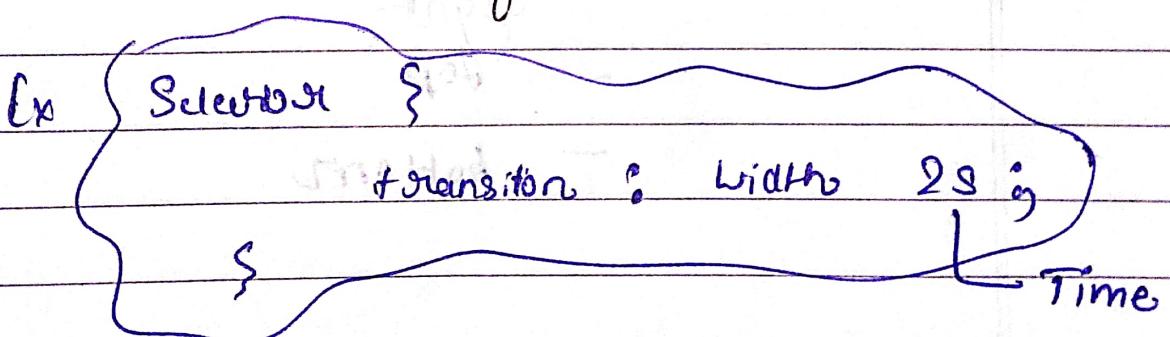
transition - function - eas - slow / fast

transition - delay - 0s

→ hover को तो क्या  
किसी वेर बाबू

→ CSS transitions allow you to change Property values smoothly, over a given duration.

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



ease → slow start → fast → ends slowly

ease-in → slow start

ease-out → slow end

ease-in-out → slow start and slow end

linear → constant start and end speed

Cubic-bezier

## Tooltip

hover → in → HTT

HTM HTT

Page No. \_\_\_\_\_

Date

→ is used to specify extra information about something when the user moves the mouse pointer on element.

<title> — used

→ Generally used in → image  
→ links

→ a tooltip that appears when the user moves the mouse over an element.

↳ placed — left

— right

— top

— bottom

# Media Queries

Page No. \_\_\_\_\_

Date \_\_\_\_\_

- Media Queries are a feature of CSS that allows you to apply different styles to a webpage on the characteristics of the device it's being displayed on.

This can be useful for creating responsive designs that look good on a variety of different devices  
Such as :- Smartphone  
Tablets  
Desktop Computers

Advantages to using media queries

in your web development projects:-

- Responsive Design
- Customized user experience → increase traffic
- improved performance → low internet usage
- enhanced accessibility
- Simplified maintenance

CODING BUGS NOTES GALLERY

Note :- always start your Responsive design.

→ Mobile first Design → Now most people are in mobile screen

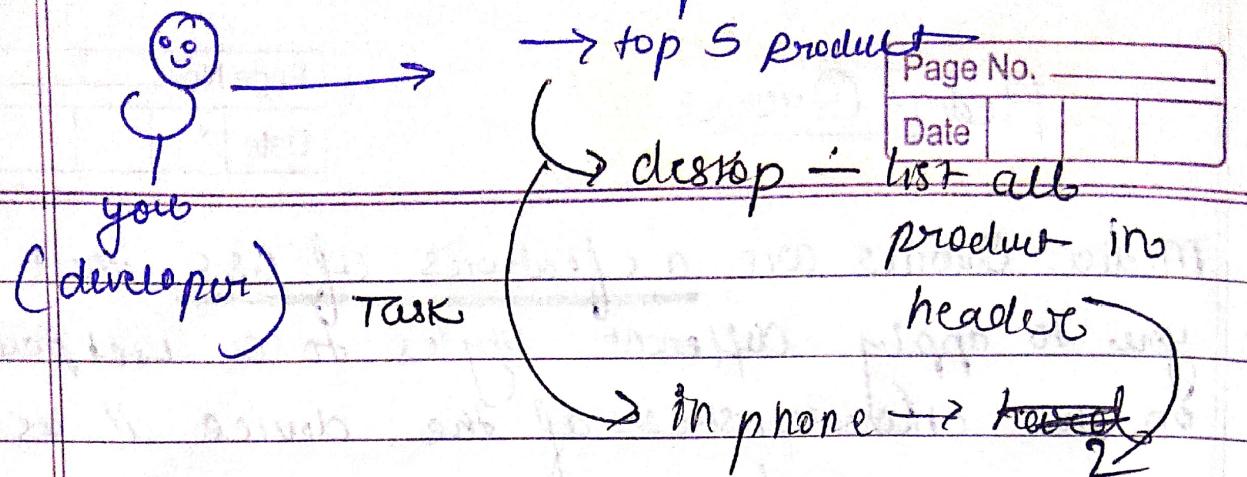
iOS Apple — various Screen size

Tablet — different size

Android — different size of different Company

Smart tv — different sizes

#



#

heavy animation

not works good

in phone well,

5 products  
efficiently

because people travel to different areas → then internet connection not well, so its hard to make it fast the animation in slow network

#

Must try → increase and decrease .....

Slowly your web browser and see the margin, - (viewport)

See how the layout change

in different screen sizes .....

And also explore

Toggle "device tool bar"

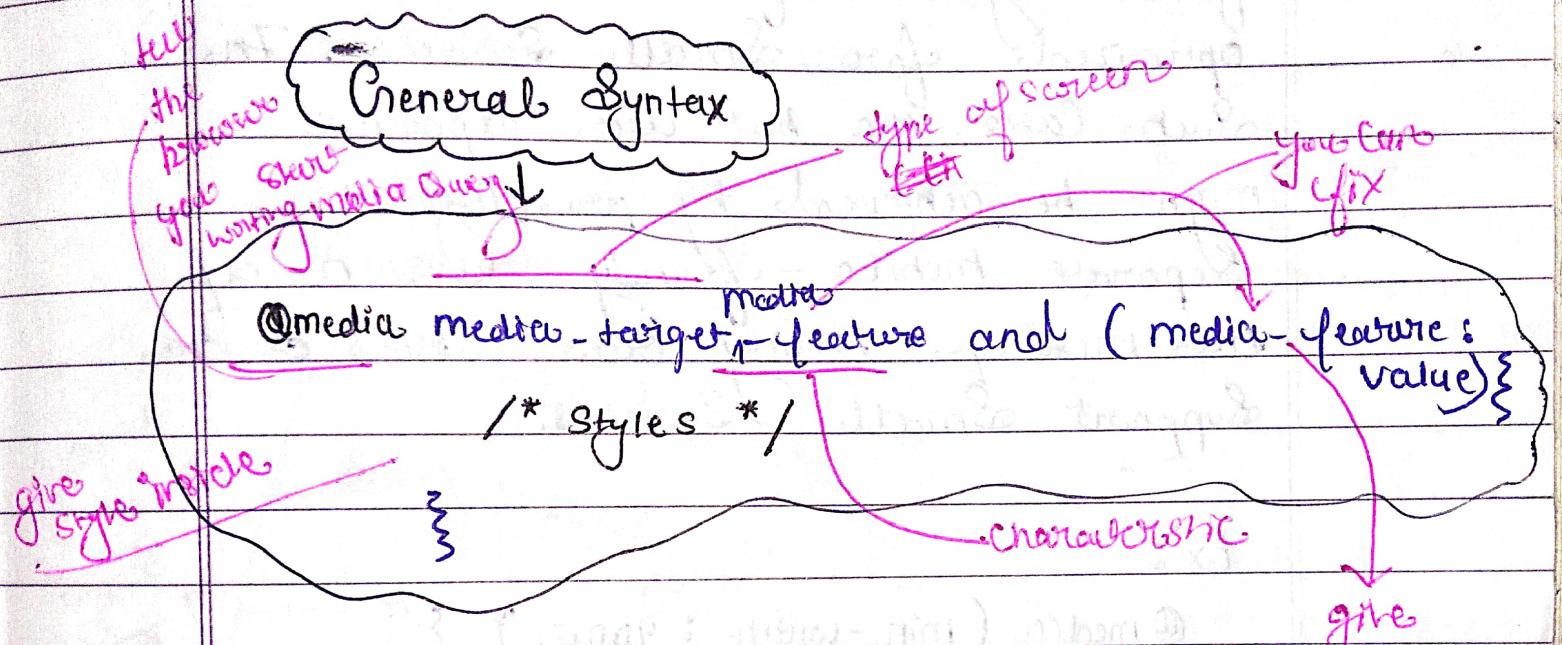
help in Debugging

## Part - 2

Page No. \_\_\_\_\_

Date \_\_\_\_\_

→ adjust your webpage in different sizes of screen.



Note : viewport understand must

- Whatever you see in web page is called viewport
- Whatever the content renders on the browser is called viewport

phone → very <sup>small</sup> <sub>big</sub> viewport  
tablet → <sup>small</sup> <sub>big</sub> viewport  
Desktop top → big viewport  
TV → very big viewport

} Set in meta

## Media Query in action

⇒ In some cases, websites may not support small screens because the website's functionality and features are not optimized for small screens. In such cases, the best user experience might be achieved by providing a separate mobile-friendly version of the message telling the user that the website does not support small screens.

Ex :-

```
@media (min-width: 900px) {
```

// if viewport is atleast 900px

then this style will apply //

Note → give size in media query

in increasing order,

it reduces the chance of overlapping

## # Media Query In Different Device Screens.

- Why popular?
- Typical Breakpoints
- Different Screens.

increase no. device

run most devices as possible →

use of mobile browsing

Benefits to business

improving experience

Better accessibility

Better performance

mobile vs most possible device,

Form supp

then browsing increase in mobile device

- Typical Breakpoints: —

There are tons of screens and devices with different heights and widths, so it is hard to calculate one exact breakpoint for each device.

To keep things simple  
you could say there are five categories

① Extra Small devices  $\Rightarrow$  max-width : 600px  
(phones, 600px and down)

② Small devices  
(portrait tablet and large phone)  
600px and up  $\Rightarrow$  min-width : 600px

③ medium devices  
(landscape tablets, 768px and up)  
 $\Rightarrow$  min-width : 768px

④ Large devices  
(laptops/desktops, 992px and up)  
 $\Rightarrow$  min-width : 992px

⑤ Extra large  
(large desktops, 1200px and up)  
 $\Rightarrow$  min-width : 1200px

Large  $\rightarrow$  1440px

normal  $\rightarrow$  1024px

Tablet  $\rightarrow$  768px

mobile  $\rightarrow$  425px

## Writing media Queries for different Screens :-

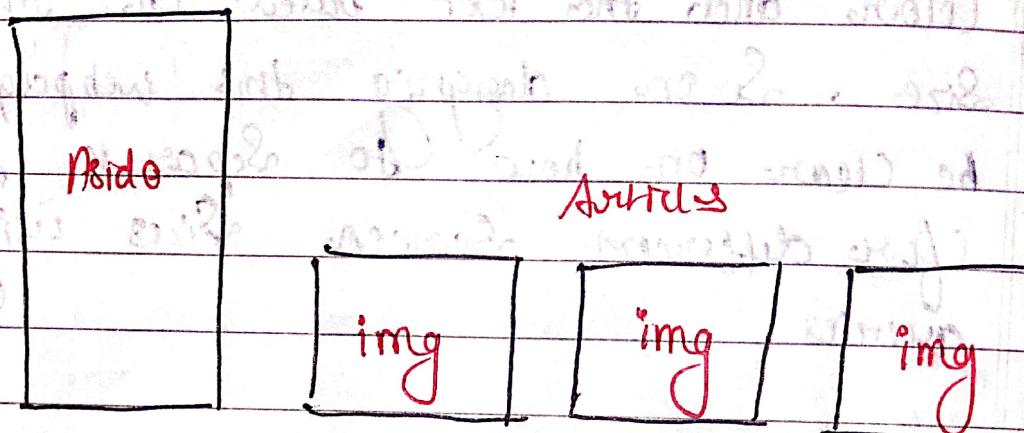
Page No.			
Date			

→ To understand how media queries (for different screens) it's change the background colour and the text bated. on the screen size . so on designing this webpage we will be clear on how to separate the CSS (for different screen sizes using media queries).

# CSS layout

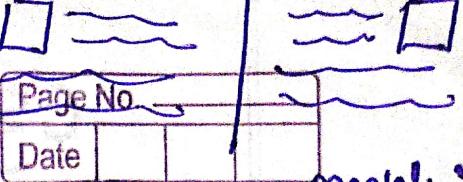
Web  
Page

Header



CSS layout Control → positions  
→ size of elements ] on the web page (viewport)

- Benefits of using
- Better presentation / Structure
  - Better accessibility
  - Flexible and responsive
  - Readability
  - Browser Compatibility



## Types of CSS layout :-

give origin to  
place element  
at any place

- ① Normal flow → what we are using to understand CSS
- ② Float → none, right, left, top, bottom most used
- ③ position → left, top, right, left, bottom
- ④ Flex
- ⑤ Grid

→ float property specifies how can position an element within the container.

→ float is mainly used when we want to align elements horizontally next to each other.

# CSS position

→ position gives vertical align. your elements  
at any place :- top, bottom, right, left  
one?

stick  
to Doc's  
parent

must  
give  
the direction

depends  
on  
origin  
condition

## Types of position

①

Static

②

Sticky

③

Fixed

④

Relative

⑤

Absolute

→ default positions

mixture of fixed and relative



tells where you  
want fix

→ change position according  
to its relatives / current  
position

what only do according to  
parents

# (Property) Overflow

Page No.	
Date	

→ The Overflow Property in CSS determines what happens to Content that is too large to fit in an element's box.

↳ Different Values of Overflow

- Overflow : visible → render outside the box
- Overflow : hidden → Outside Content will be Invisible
- Overflow : scroll → add Scrolling — See all Content
- Overflow : auto → Similar as scroll, but adds scrollbars only when necessary — have choice

→ add scrollbars even if you needed 0% not (no choice)

depends on browsers

## # Overflow-x and Overflow-y

— these properties specify whether to change the overflow of content just — horizontally  
— vertically  
— both

x → left / right

y → top / bottom

# Z-index

Page No. \_\_\_\_\_

Date \_\_\_\_\_

(Property) — work as Stack

- Z-index Property in CSS used to specify the stack order of an element.
- An element with a higher Z-index value will be placed in front of an element with a lower Z-index value.
- Z-index : (Integer Value) — 1, 2, 3 etc
  - ↳ the elements with a higher number is placed on top of the elements with lower numbers.
- default Z-index — always — 1
- always give value in Negative Value
  - ↳ works as transparency
- it only works, if two things are in same position (place)
  - low priority → lower position
  - higher " → higher "

# Flex Box

Page No.

Date

→ part of CSS layout  
(method)

CODING BUGS NOTES GALLERY

→ CSS Flexbox layout allows you to easily format HTML.

Flexbox makes it simple to align items vertically and horizontally using rows and columns.

Items will "flex" to different sizes to fill the space. It makes responsive design easier.

Making all column in a multiple-column layout adopt the same height even if they contain as different amount of content

Why Flexbox  
easily adds space

Vertically centering a block of content inside its parent

One dimension layout

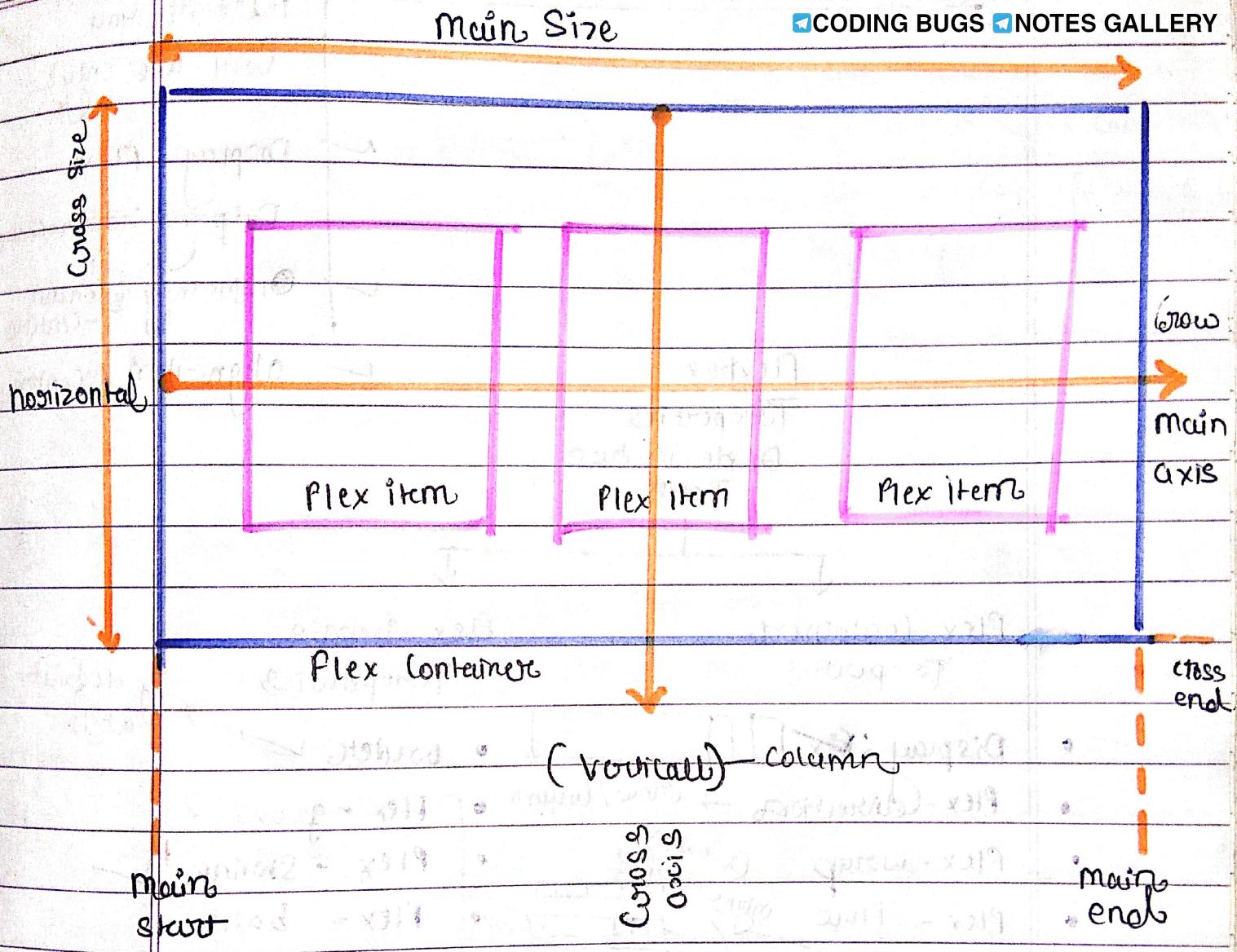
Making all the children of a container take up an equal amount of the available width/height regardless of how much width/height is available.

Note :- Flexbox Container → Flexbox model

Page No.			
Date			

Main Size

CODING BUGS NOTES GALLERY



Some fundamental terminologies

of flexbox:-

- ① Flex Container
- ② Flex items
- ③ Main axis
- ④ Cross axis

## Different Flexbox Properties

In future you will use only ↴

Display : flex

~~Display : inline-block~~

Justify-content : center

Align-items : center

Flexbox  
Properties  
Divide into two  
Parts

Flex Container  
Properties

Flex items

Properties

by default  
at 0

- Display : flex
  - Flex-direction → row / column
  - Flex-wrap (x-axis) → right to left
  - Flex-flow (y-axis) → right to left
  - justify-content ↴
  - align-items ↴
  - align-content ↴
  - gap
    - (i) row-gap
    - (ii) column-gap
- give  
not  
space

- order ↴
- Flex-grow 0
- Flex-shrink 1
- Flex-basis
- Flex
- align-self.

Commons — Flex

you add  
Property all at  
Same line.

## • Display Property

Display : flex

Display : inline-flex

It us used to define a flex container. After defining a flex container using display flex or inline-flex.

•

## • Justify Content

→ (works in x-axis) — left to right

→ align the flex items in main axis

Values :- flex-start

flex-end

center

space-between



space-around — equal margin

space-evenly

## • Align Item

→ works in y-axis → top to bottom

→ cross axis

values :- stretch

- flex-start

- flex-end

- center

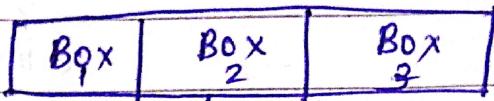
- baseline

## Flex Items Properties

Page No. \_\_\_\_\_

Date \_\_\_\_\_

→ There are the properties that are used over the flex items (child elements)

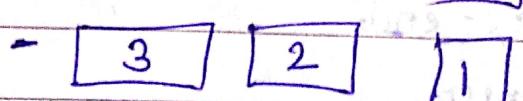
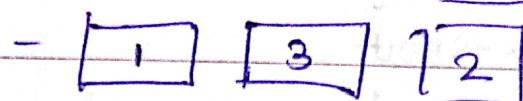


→ work on the particular box, i.e., we use flex item properties.

### • Order

#### Array

→ give the order / positions of individual flex item.



→ align as you want

→ by default → 0

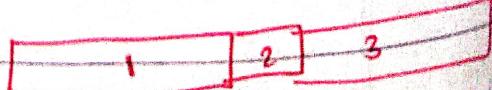
### • flex grow → left to right

→ for increasing and decreasing the individual flex item.



→ default → 0

→ negative value also allow



→ big numbers & big size

Page No.	
Date	

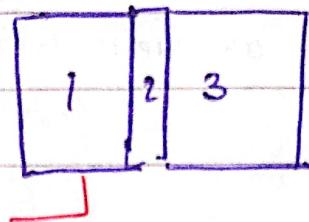
- Flex-Shrink

(top to bottom)

→ work same as flex-grow

- Flex-Shrink

→ left to right



Box 2 ⇒ 2

→ Flex item should be shrink if the flex container is smaller than the total size of flex items.

→ only positive value ✓

→ default value 1 ✓

means elements will shrink equally to fit the container.

→ higher numbers & more shrink

CODING BUGS NOTES GALLERY

- Flex basic

## • flex

in Flex property, you give all property

in One time

- Flex-grow

- Flex-shrink

Flex-base

## • align-self

→ align a flex item its flex container.

(set as you want)

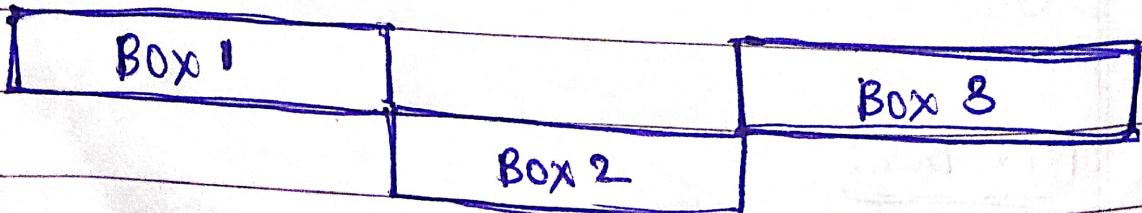
- Stretch

- Flex-start

- Flex-end

- Center

- baseline



# GRID

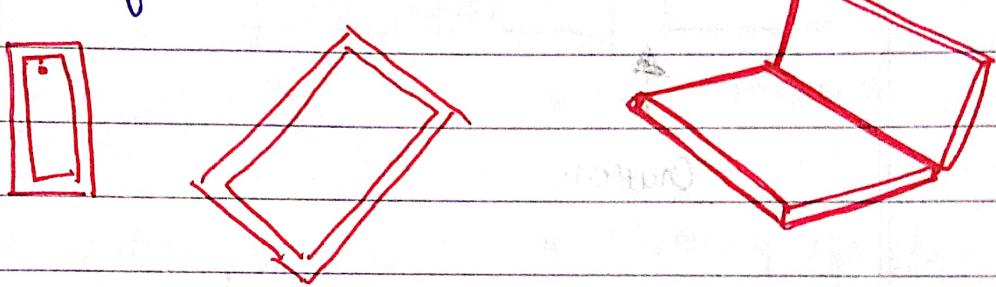
Page No.

Date

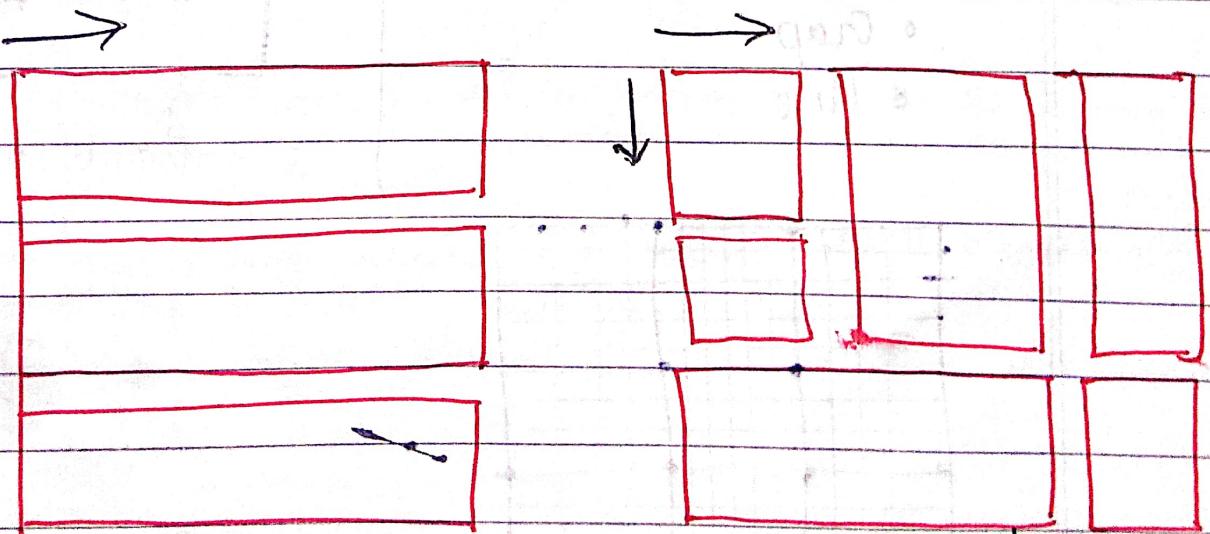
CODING BUGS NOTES GALLERY

## GSS Grid layout

→ is a two-dimensional layout system for the web. It lets you lay content out in rows and columns and has many features that make building complex layout straightforward.



They help us to create designs where elements don't jump around or change width as we move from one page to page, providing greater consistency on our websites.



Flexbox

One Dimension

(Drawback  
big)

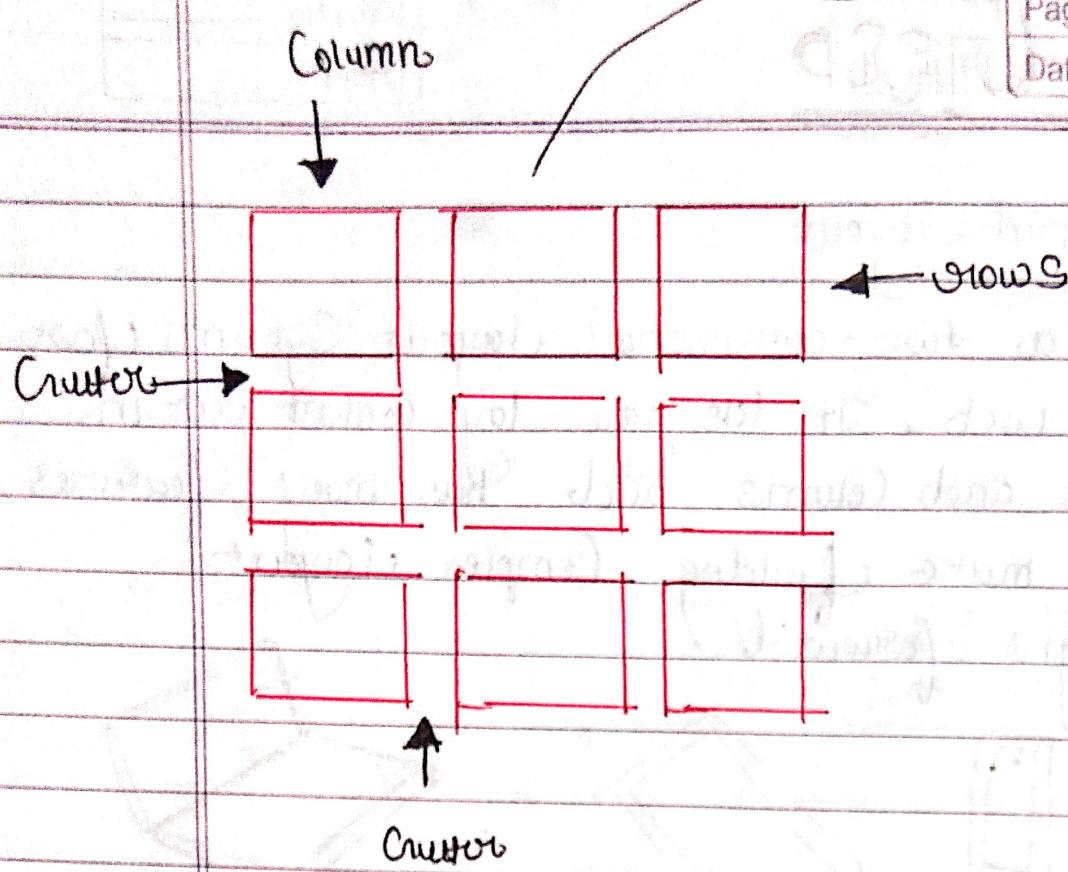
CSS Grid

Two Dimension

Why its come

## Grid Layout

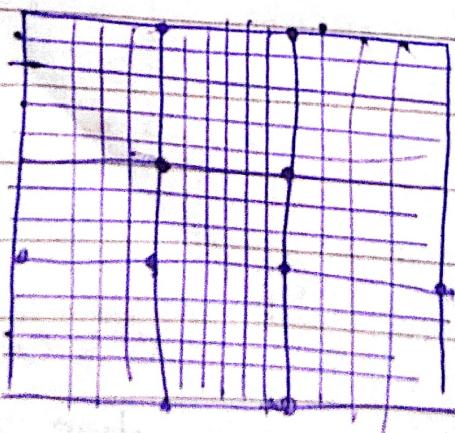
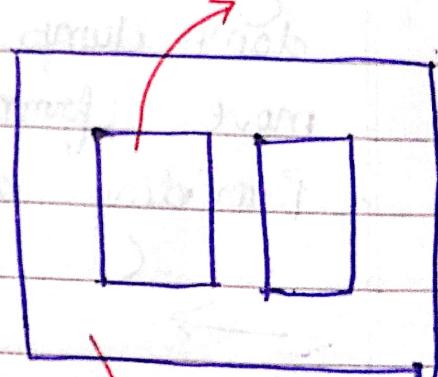
Page No.			
Date			



## Terminologies :-

- Item
- Row
- Column
- Gap
- Line

Grid item



## Properties of Grid

(Parent)

(child)

Grid  
Container

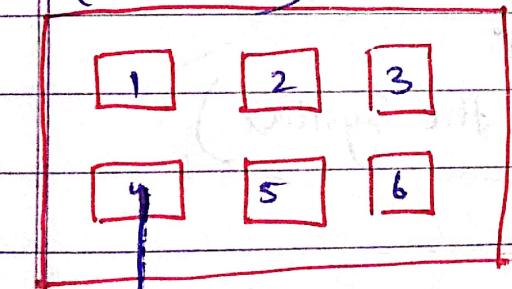
Grid  
items

use one  
gravior  
unit

CODING BUGS NOTES GALLERY

- Grid - template-rows
- Grid - template-columns (1fr)      • Grid - row
- Align - Content (y-axis)      • Grid - Column
- Justify - Content (x-axis)      • Grid - area
- Naming grid

Ex



Pg - Gravior Unit

Grid  
Container