

CSS

→ `list-style-type: circle;`

• HTML
• CSS
• JS } for unordered list

→ `list-style-type: square;`

• HTML
• CSS
• JS } for ordered list.

→ `list-style-type: upper-roman;`

I HTML
II CSS
III JS }

→ `list-style-type: lower-alpha;`

a) HTML
b) CSS
c) JS } for ordered list

ul {

→ `list-style-type: url('Image.png');` (for unordered list)

→ Remove Bullets

ul {

`list-style-type: none;
margin: 0;
padding: 0;`

Output

HTML
CSS
JS

→ ul {

list-style: square inside ul ("arrow.png");
}

→ li {

display: inline;

}

→ li {

display: block;

}

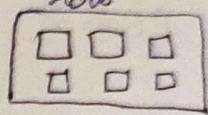
Output

html CSS JavaScript

We can use it
to make
cards in one
row

Output

HTML
CSS
JavaScript



→ we can set background color and margin, padding of
ordered list and unordered list

Example:

ol li {

background:

padding:

}

ol {

}

ol li {

}

ul {

}

ul li {

}

* → Responsive Table

Add table in div tag to make it responsive

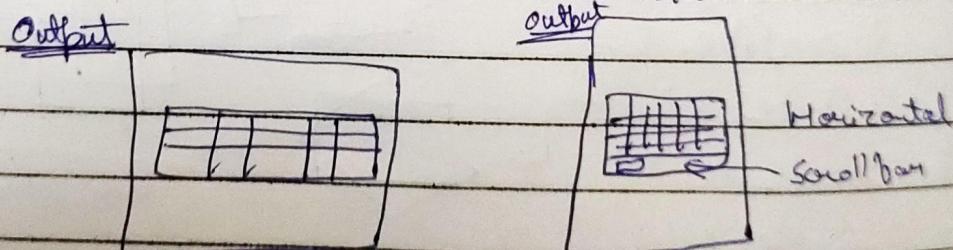
→ `<div style = "overflow-x: auto;">`
`<table>`

Table content
`<table>`
`</div>`

~~CSS~~ → `border-collapse: collapse;`
`width: 100%;`

~~CSS~~ → `tr:nth-child(even) { background-color: # f2f2f2; }`

In ~~the~~ small screen we can see horizontal scroll bar to view table data



→ Block - Level Elements

`<div>`
`<h1> to <h6>`
`<p>`
`<form>`
`<header>`

`<footer>`
~~or~~ `<section>`

→ Inline Elements

<a>

* image

display : none

<! --- to hide an element --->

It is mostly used with JavaScript

→ ~~Span~~ Display span elements as block elements

→ <Style> ~~Span { display : block; }~~ </Style>
 Text 1 is here Text 2 is
 here

Output

Text 1 is here

Text 2 is here

Note

[display : block with span is used in html 5
 and br tag is used in html 4]

→ Opacity

img ↗

opacity : 0.5;

{

grey color: #888888

* CSS

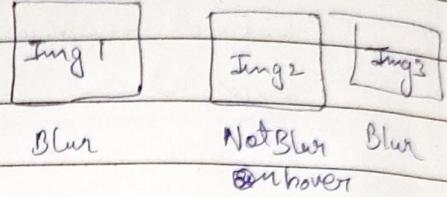


img {
 opacity: 0.5;
}

^{Can}
I used this in my
"editing photos" page.



img:hover {
 opacity: 1.0;
}



Img

div.background {

background: url(img.jpg) repeat;

border: 2px solid black;

}

* * glass effect

div.transbox {

margin: 30px;

background-color: #ffffff;

border: 1px solid black

opacity: 0.6;)

}

div.transbox p {

margin: 5px;

font-weight: bold;

color: #000000;

}

* CSS Gradients

background-image: linear-gradient (red, blue);

by default it is top to bottom

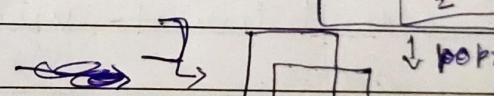
→ background-image: linear-gradient (top right, red, blue);

" " " " (top right, red, blue);
↳ diagonal

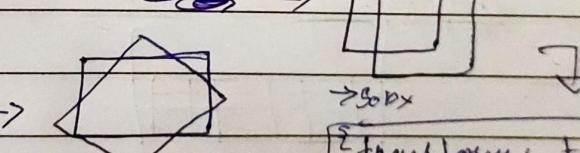
CSS 2D transformation methods

transform

translate()



rotate()



scaleX()

→ 50px
transform: transform(50px, 0deg);

scaleY()

→ we can put negative values here
transform: rotate(20deg);

scale()

→ we can put -2 values here
transform: scale(2, 3);

skew()

→ transforms : skewX(20deg);
transform : skew(20deg, 10deg);

for width

skewX()

matrix()

for height

skewY()

matrix()

matrix(scaleX(), skewY(), skewX(),

scaleY(), translateX(), translateY())

CSS Animation

- @keyframes
- animation-name
- animation-duration
- animation-delay
- animation-iteration-count
- animation-direction
- animation-timing-function
- animation-fill-mode
- animation

div{}

width: 100px;

height: 100px;

background-color: red;

animation-name: example;

animation-duration: 4s;

{}

Both
work
together

@keyframes example {

from {background-color: red; }

to { background-color: blue; }

{}

Both work
together

@ Keyframes example {

0%. { background-color: red; }

25%, {

yellow; }

50%, {

green; }

100%, {

blue; }

{}

→ normal

→ reverse

→ alternate

→ alternate-reverse

} to use this add
animation-direction: reverse