CSS

CSS setand for cascading stylesheet.

CSS describes how HTML elements / tag are to display on screen (web Browser).

CSS was introduced by hakon Wium in 1995.

Types of css

1.Internal css

2. External css

3. Inline css

**1. Internal CSS:**

**Syntax :**

<Style>

selector{

}

</style>

Selector:- is used to select Html tags to apply CSS.

**Types of Selector:-**

1.tag Selectors / Element Selector

2. id selector

3.class selector

4. group selector

5.universal selector

6.Child selector

7.Direct selector

8.Odd selector

9.Even selector

10.Nth-child selector

11.Adjacent selector

12.Pseudo selector

13. Attribute selector etc.

**Border properties:-**

Border-width:10px;

Border-style: solid

Border-color: red;

======

Border:width style color;

==

**Writing Mode:-** The writing-mode property in CSS defines the direction in which text and inline content flow. It is useful for supporting vertical text layouts and different writing systems.

Writing-mode:horizontal -tb;

**Values:**

**horizontal-tb (default):** Text flows horizontally from left to right (LTR) or right to left (RTL).

**vertical-rl:** Text flows vertically from top to bottom, with lines ordered right to left.

**vertical-lr:** Text flows vertically from top to bottom, with lines ordered left to right.

**sideways-rl:** Like vertical-rl, but the text remains upright.

**sideways-lr:** Like vertical-lr, but the text remains upright.

**Text-transform**

**Values:**

* **none** → Default, no transformation.
* **capitalize** → First letter of each word capitalized.
* **uppercase** → Converts text to all uppercase.
* **lowercase** → Converts text to all lowercase.
* **full-width** → Expands characters (mainly for Asian scripts).

**Text-shadow:-**

text-shadow: x-offset y-offset blur-radius color;

Text-shadow: horizontal ,vertical , blur, color;

**Values:**

** x-offset** → Horizontal shadow position (positive = right, negative = left).

** y-offset** → Vertical shadow position (positive = down, negative = up).

** blur-radius (optional)** → Blurs the shadow (higher = softer).

 **color** → Specifies shadow color.

**Hover:-** The :hover pseudo-class in CSS applies styles when a user hovers over an element with a mouse.

**Syntax:-**

Selector:hover

{

Property:value;

}

**Yah sab hover ke sath use hota hai [**

**The transition property in CSS allows smooth animation between property changes.**

**Value:-**

**[**

* **property →** CSS property to animate (e.g., background-color).
* **duration →** Time of transition (e.g., 0.5s).
* **timing-function (optional) →** Speed curve (ease, linear, ease-in, ease-out, etc.).
* **delay (optional) →** Time before transition starts.

]

Transition:property time in second, property time in sec

----===

Transition:color 2s, text-shadow 5s

Transition:all time in second;

Transition:all 4s;

]

**Box-shadow:** The box-shadow property in CSS adds shadow effects around an element’s box.

box-shadow: x-offset y-offset blur-radius spread-radius color;

Box-shadow: horizontal vertical blur spred color side;

Padding:50px

Top bottom left right

Box-sizing:content-box;

Border-box=> applicable on padding and border.

Background:

Linear-gradient

Repeating- Linear-gradient

Radial-gradient

Repeating- radial-gradient

--

Transform:rotate| skewX|skewY|skew | translate

Position: static default value |absolute| relative | fixed| sticky

Left| right

Top | bottom

**Animation CSS:-**

**Syntax:-** @keyfram animationName

{

From{

Property:value;

Property:value;

}

To

{

Property:value;

Property:value;

}

}

Call karne ke liye

1. Animation – name: animationName;
2. Animation-duration:time in second |5.5s
3. Animation delay: time in second | 4s;
4. Animation-iteration-count: Number| infinite
5. Animation-direction:normal| reverse | alternate| alternate-reverse
6. Animation-play-state:running by default | paused
7. Animation-timing-function: linear | ease-in | ease-out| ease-in-out
8. Animation-fill-mode: forward|

* Animation: name duration timing-fuction delay direction

**Multiple colore use:-**

@keyframes animationName{

0%{

Property:value;

Property:value;

}

25%{

Property:value;

Property:value;

}

50%{

Property:value;

Property:value;

}

75%{

Property:value;

Property:value;

}

100%{

Property:value;

Property:value;

}

}

**Animation-timing-function:-**

Liner => same speed from start to end

Ease ( by default value)=> start slow then fast and end slow

Ease-in => start slow

Ease-out=> end slow

Ease-in-out=> start and end slow

Types:\_

List-style