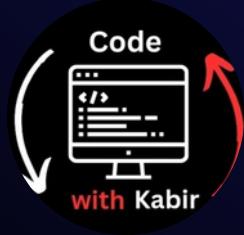


# CSS CHEATSHEET

2024



Swipe →

@codewithkabir

# WHAT IS CSS?

- CSS stands for Cascading Style Sheets.
- CSS is used to style and layout web pages by controlling their visual appearance and design.

## SYNTAX

```
selector {  
    property : value ;  
}
```

# SELECTORS

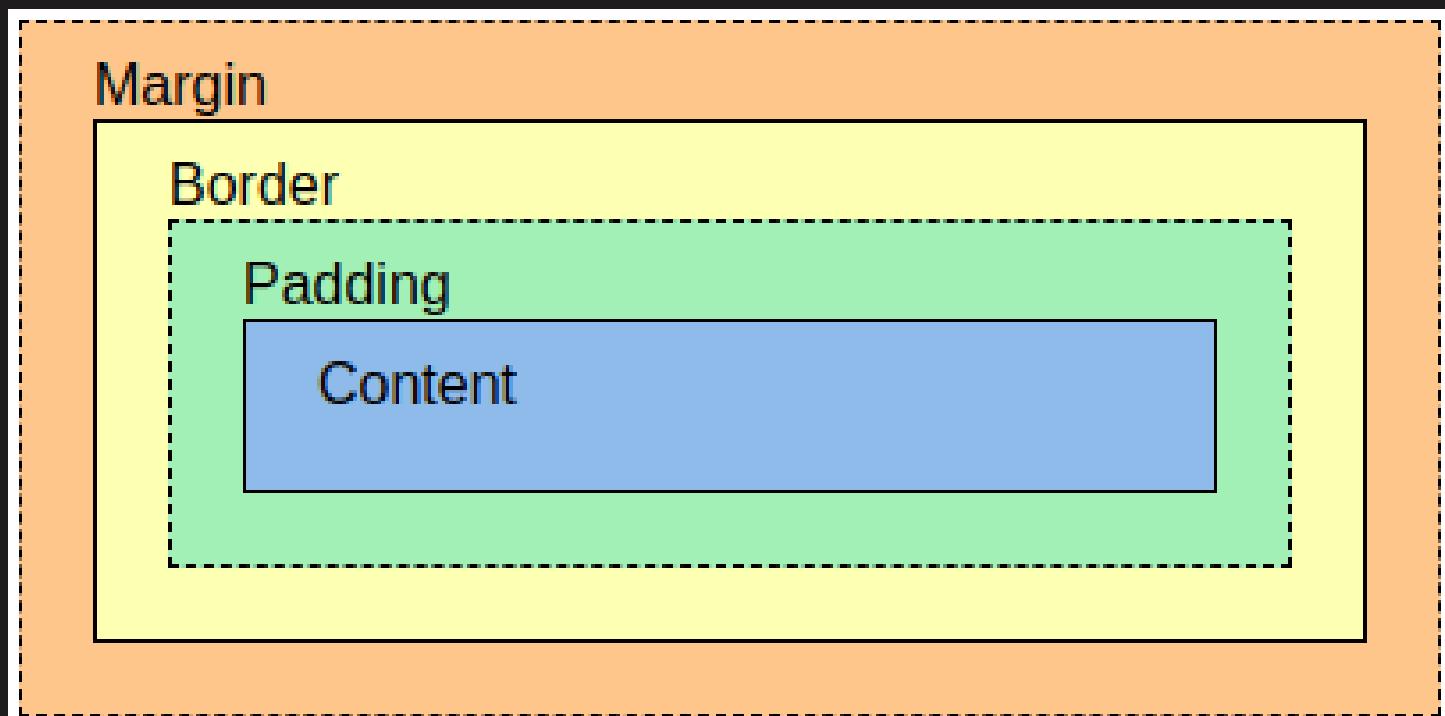
- **\*** Universal selector
- **element** Type selector
- **.class** Class selector
- **#id** ID selector
- **element, element** Group selector
- **element element** Descendant selector
- **element > element** Child selector
- **element + element** Adjacent sibling selector
- **element ~ element** General sibling selector

# UNITS

<b>%</b>	Percentage
<b>cm</b>	Centimeter
<b>in</b>	Inch
<b>mm</b>	Millimetre
<b>pc</b>	Pica (1 pica = 12 points)
<b>pt</b>	Point (1 point = 1/72 inch)
<b>px</b>	Pixel (1 pixel = 1/96 inch)
<b>ch</b>	width of 0 character in font being used.
<b>ex</b>	X-height of element's font
<b>em</b>	1 em = current font size
<b>gd</b>	grid defined by layout grid
<b>rem</b>	Font size of the root element
<b>vh</b>	Viewport's height
<b>vw</b>	Viewport's width
<b>vm</b>	Smaller of viewport's height or width

# BOX MODEL

```
.element {  
    width : 100px;  
    height : 100px;  
    padding : 10px;  
    border : 1px solid black;  
    margin : 20px;  
}
```



# TEXT

```
color: #333;      // Text color  
font-size: 16px;  // Text size  
font-family: Arial; // Font  
font-weight: bold; // Font weight  
text-align: center; // Aligns the text  
text-decoration: underline; // Adds decoration to text  
text-transform: uppercase; // Transforms the text
```

# BACKGROUND

```
background-color: #f0f0f0; // color of background  
background-size: cover; // size of background  
background-image: url( image.jpg ); // background image  
background-repeat: no-repeat; // background repetition  
background-position: center; // background position
```

# FLEXBOX

```
.element {  
  display: flex;    // Defines a flex container  
  align-items: center; // Aligns items to center horizontally  
  justify-content: center; // Aligns items to center vertically  
  flex-direction: row; // direction of the items  
  flex-wrap: wrap; // determines whether items wrap  
}
```

```
.child {  
  flex: 1;    // sets how a flex item will grow  
}
```

# GRID

```
display: grid;      // Enables Grid  
display-template-columns: // Define columns  
display-template-rows:      // Define rows  
grid-gap:   // space between grid items  
grid-row:   // item's size and location within a grid row  
grid-column: // item's size and location within grid column
```

# DISPLAY AND POSITION

display: block | inline | inline-block | none | flex | grid  
position: static | relative | absolute | fixed | sticky  
top, left, right, bottom: positioning offsets

# BORDER AND OUTLINE

border: border-width border-style border-color  
border-radius: Rounding corners  
outline: Outline around the elements  
box-shadow: Shadow around the elements

# PSEUDO-CLASSES AND PSEUDO ELEMENTS

**:hover** // applied when an element is hovered

**:focus** // applied when the element is focused

**:active** // applied when the element is activated

**::before** // inserts content before element's content

**::after** // inserts content after element's content

## MISCELLANEOUS

**opacity** // Transparency level of an element

**overflow** visible | hidden | scroll | auto

**z-index** stacking order of positioned elements

**cursor** Mouse's cursor to be displayed

# TRANSITION AND ANIMATION

```
.transition {  
    transition: all 0.3s ease;      // Transition effect  
}  
  
.animation {  
    animation: anim 0.3s infinite;  
}  
  
 @keyframes anim {  
     from { background-color: red; }  
     to { background-color: aqua; }  
}
```

# MEDIA QUERIES

```
@media ( max-width: 600px ) {  
    .responsive{  
        background-color: lightblue;  
    }  
}
```