

CSS3

5 parts

1. Page layout
2. Element styles
3. Element position
4. Responsive design
5. Animation

3 ways to include CSS

Inline / Embedded / External

CSS rule

Selector, Property, Value

Color model

color

background-color: linear-gradient(to bottom right, red, blue)

rgb, rgba, hex, hsl, name

opacity

Font

font-family (5 basic fonts, typography)

font-size (px)

font-style, font-weight

letter-spacing, word-spacing

text-transform

text-shadow, text-align

line-height **cornerstone**

text-indent (negative value)

text-decoration

font-variant: small-caps

:first-line, :first-letter

%, em, rem

CSS box model

box model: content, padding, border,

margin **cornerstone**

border

border-style: solid/double

border-color

border-width (create arrow)

border-image

border-radius (circle, eclipse, prove)

Examples of block, inline, inline-block elements

display **cornerstone**

inline, inline-block, block, none

default width/height for each type

set width/height for each type

visibility

box-sizing **cornerstone**

How margin and padding work for inline, inline-block and block elements

How margin works when set width

margin: collapse

margin percentage value relative to container width

Element styles

cursor

user-select

outline, outline-offset

width: 100%, auto, 0

applied on block and inline element

min-width, max-width, **width**

min-height, max-height, **height**

width: 1000%

overflow, overflow-x, overflow-y

visible, hidden, scroll, auto

overflow for float child **cornerstone**

box-shadow

list-style

border-collapse (for table)

background image

background-image: url(...)

background: url(...) 100px 100px, url(...)

background: linear-gradient(to right, white 20%, black 20%, white, black);
background-image: linear-gradient(), url();

radial-gradient(circle 100px at 100px 100px, white, blue, black)

repeating-linear-gradient

repeating-radial-gradient

background-repeat: no-repeat

background-repeat: no-repeat, repeat

background-position

top, right, bottom, left, center

background-size: auto, cover, contain

background-size: 100px 100px

background-size: 100px

background-attachment

scroll, local, fixed

background-clip

padding-box, content-box, border-box

-webkit-background-clip: text

color: transparent

filter: grayscale(100%) ...

Element position

linebox **cornerstone**

vertical-align: -4px/top/... **cornerstone**

position **cornerstone**

static, **relative, absolute**, fixed

top, left, right, bottom

z-index (for non static)

Centering (vertical, **horizontal**)

cornerstone

float, clear **cornerstone**

text-align (left, justify, right)

Page layout: grid, flexbox better

float, inline-block **cornerstone**

(use float only for image wrap around text)

column-count, column-gap

CSS selectors

***, tag, #id, .class**

s1, s2 (select all s1 and all s2)

s1 s2, s1>s2

[att], [attr="val"]

:active, :focus, :hover,

:required,

:first-child, :last-child

:nth-child(n), :nth-last-child(n)

:root

:not(selector)

CSS specificity

1. !important > inline style > id selector > pseudo class > class or attribute selector > tag

2. long path > short path

3. The last rule overrides the previous rule

Insert HTML element

Content **cornerstone**

:before, :after

Responsive design

Syntax example

@media all and (min-width: 768px)

{ ... } **cornerstone**

Web fonts, Web icons, SVG

google fonts

font awesome

More properties

all: initial|inherit

background:url(...), url(...)
background-blend-mode: overlay|...
background-origin: padding-box |
border-box | content-box

mix-blend-mode: overlay | ...

object-fit
object-position

caret-color
user-select: none
hyphens:auto
word-break: break-all|keep-all

clip-path: url(#mysvgpath) |
circle(40% at 100% 50%) |
eclipse(200px 100px) | polygon(50%
0, 100% 50%, 50% 100%, 0 50%) |
inset(4% 24% 23% 22%)

count-increment: mycounter 100
content: count(mycount)

display: contents
display: list-item

position: sticky

outline

resize: both
overflow: auto

caption-side:bottom
table-layout: auto

appearance

CSS functions

```
:root {  
  --bg-color: lime  
}  
color: var(--bg-color)
```

```
<div data='data from server'></div>  
content: attr(data)
```

More selectors

[attr~='val'] (space list), [attr*='val']
[attr|= 'val'] (dash list), [attr^='val']
[attr\$='val']
::selection, :target, :visited, :link,
:checked, :disabled, :enabled, :in-
range, :out-of-range, :invalid, :optional,
:read-only, :read-write,
:valid
:empty
:only-of-type, :only-child,
:first-of-type, :last-of-type,
:nth-of-type(n), :nth-last-of-type(n)

More layouts

Grid (2 dimension)

place-content: center

flex box (1 dimension)

For container:

display: flex
flex-direction: row, column, row-
reverse, column-reverse
flex-wrap: wrap

justify-content: flex-end, space-
around, space-between, center
align-items
align-content
place-content

flex-flow

For flex item:

order
flex

flex-basis (0 and auto, compare with
width)
flex-grow
flex-shrink

align-self

Front-end framework

Bootstrap/Foundation

Animation

transition: p1 1s, p2 3s linear;
transition: 1s

transition-property
transition-duration
transition-timing-function
(cubic-bezier.com)
transition-delay (can be negative)

transform for 2D (has order)

translateX(100px),
translateY(200px),
translate(100px, 200px)
scaleX(2)
scaleY(1)
scale(2,1)

rotate(90deg), rotate(10turn)

transform-origin: x-axis y-axis z-axis
top left right bottom/10px 10px/%

skewX(20deg)
skewY(20deg)
matrix(...)

transform for 3D

perspective property
perspective function
perspective-origin

translateZ(200px)==scaleZ(2)
translateZ(100px)

translate3d(x,y,z)

rotateX(2turn)
rotateY(45deg)

rotateZ(45deg)
rotate3d(...)
transform-style: preserve-3d/flat

backface-visibility:hidden;

animation

```
@keyframes move {  
  from {  
    translateX(100px);  
  }  
  to {  
    translateX(200px)  
  }  
}
```

animation-name
animation-duration

animation-timing-function: ease-in,
ease-out, ease-in-out,
step-start, step-end,
steps(2), steps(2,start)

animation-iteration-count:infinite
animation-delay:1s
animation-direction: normal/reverse/
alternate

animation-fill-mode: forwards/
backwards/both

animate spritesheet

Extra

animate.css library
YouTube channel

By TeensProgramming.com