

CSS3

4 parts

1. Element styles
2. Page layout
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, %, em, rem)
font-style, font-weight
letter-spacing, word-spacing
text-transform

text-shadow
line-height
text-indent (negative value)
text-decoration
font-variant
small-caps

:first-line, :first-letter

CSS box model

box model: content, padding, border, margin

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 block and inline elements

How margin works when set width

margin: collapse, negative

Element styles

cursor
outline
user-select

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

background: linear-gradient(...), url(image)

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

repeating-linear-gradient
repeating-radial-gradient

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
appearance

box-shadow

background image

background-image
background-repeat
background-position
top, right, bottom, left
center

background-size
auto, cover, contain

background-attachment
scroll, local, fixed

background-clip
padding-box, content-box, text

list-style

border-collapse (for table)

Element position

vertical-align 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

float, inline-block cornerstone
column-count, column-gap

CSS selectors

Syntax examples:
*, tag, #id, .class
s1, s2 (select all s1 and all s2)
s1 s2, s1>s2, s1+s1, s1~s2
[att], [attr="val"]
:active, :focus, :hover,
::after, ::before
:required,
:first-child, :last-child,
:nth-child(n), :nth-last-child(n)
::first-letter, ::first-line, :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

Responsive design

Syntax example
@media all and (min-width:768px) {
... } cornerstone

Web fonts, Web icons, SVG

google fonts
font awesome
SVG

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

flex box

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

flex-flow

For flex item:

order
 flex

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

Grid

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;

```
@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