

# CSS BASICS

Yevhen Isakov

2019

# CSS SYNTAX

```
h1 {  
  color: blue;  
  background-color: yellow;  
  border: 1px solid black;  
}  
  
p {  
  color: red;  
}  
  
p, li {  
  text-decoration: underline;  
}
```

```
p {  
    border: 1px solid black;  
    color: red;  
}
```

CSS Syntax Module Level 3

Selector

p

{

border: 1px solid black;

color: red;

}

## Brackets

p

{

border: 1px solid black;

color: red;

}

p {

Property

border

: 1px solid black;

color

: red;

}

```
p {
```

Colon

```
border: 1px solid black;
```

```
color: red;
```

```
}
```

p {

border: 1px solid black;

color: red;

}

Value



```
p {
```

```
border: 1px solid black;
```

```
color: red;
```

```
}
```

Semicolon



p {

Declaration

border: 1px solid black;

color: red;

}

Block

```
p {  
    border: 1px solid black;  
    color: red;  
}
```

# CSS PROPERTIES

- color
- font-weight
- width
- height
- border
- padding-top
- background-color
- display
- transform
- ...etc

```
h1 {  
    display: inline-block;  
    width: 80%;  
    border-bottom: 1px solid gray;  
    background-color: yellow;  
    color: #333;  
    font-weight: bold;  
}
```

```
div {  
    padding-top: 20px;  
    padding-right: 0;  
    padding-bottom: 10%;  
    padding-left: 12px;  
}
```

```
div {  
    padding: 20px 0 10% 12px;  
}
```

L

T

B

R

# VENDOR PREFIXES

```
div {  
    -webkit-border-radius: 10px;  
    -moz-border-radius: 10px;  
    border-radius: 10px;  
}
```



# VENDOR PREFIXES

- `-webkit-` Chrome, Safari (WebKit/Blink)
- `-moz-` Firefox (Gecko)
- `-ms-` Internet Explorer / Edge (Trident/Edge)
- ~~`-o-` Opera (Presto)~~

Can I Use?



## Inherited

font-family font-size font-style font-weight color  
text-align line-height direction etc

## Not inherited

display margin background-color position etc

# CSS INHERITANCE

```
em {  
    background: inherit;  
}
```

# CSS CASCADING

The cascade takes a unordered list of declared values for a given property on a given element, sorts them by their declaration's precedence, and outputs a single cascaded value.

# CSS CASCADING - ORIGINS

- user-agent stylesheet
- author styles
- *reader* styles

# CSS CASCADING - SPECIFICITY

1. Tag selector ( `h1` ), pseudo element ( `::before` )
2. Class selector ( `.example` ), attribute selector ( `[type="radio"]` ), pseudo class ( `:hover` )
3. ID selector ( `#example` )
4. Element styles ( `style="font-weight:bold"` )

# CSS CASCADING - SPECIFICITY

- Universal selector (`*`) and combinators (`+`, `~`, `>`) and `:not()` **do not affect** the specificity
- `!important` overrides all other entries. **DO NOT USE**  
**!IMPORTANT**
- CSS Cascading and Inheritance Level 3



# CSS SPECIFICITY

Selector	Specificity	Specificity in base 10
<code>style=""</code>	1, 0, 0, 0	1000
<code>#wrapper #content</code>	0, 2, 0, 0	200
<code>#content .list</code>	0, 1, 1, 0	110
<code>div#content</code>	0, 1, 0, 1	101
<code>p.comment .date</code>	0, 0, 2, 1	21
<code>p.comment</code>	0, 0, 1, 1	11
<code>ul li</code>	0, 0, 0, 2	2
<code>div</code>	0, 0, 0, 1	1



[http://www.stuffandnonsense.co.uk/archives/css\\_specificity\\_wars.html](http://www.stuffandnonsense.co.uk/archives/css_specificity_wars.html)



**a**  
1 x element selector  
Sinh: 0, 0, 1



**p a**  
2 x element selectors  
Sinh: 0, 0, 2



**.whatever**  
1 x class selector  
Sinh: 0, 1, 0



**a.whatever**  
1 x element selector  
1 x class selector  
Sinh: 0, 1, 1



**p a.whatever**  
2 x element selectors  
1 x class selector  
Sinh: 0, 1, 2



**.whatever .whatever**  
2 x class selectors  
Sinh: 0, 2, 0



**p .whatever a.whatever**  
2 x element selectors  
2 x class selectors  
Sinh: 0, 2, 2



**#whatever**  
1 x id selector  
Sinh: 1, 0, 0



**a#whatever**  
1 x element selector  
1 x id selector  
Sinh: 1, 0, 1



**.whatever a#whatever**  
1 x element selectors  
1 x class selector  
1 x id selector  
Sinh: 1, 1, 1



**.whatever .whatever #whatever**  
2 x class selectors  
1 x id selector  
Sinh: 1, 2, 0



**#whatever #whatever**  
2 x id selectors  
Sinh: 2, 0, 0

# SELECTORS

Tag selector — E

div

h1

img

Class selector — `.class`

`.header`

`.item`

`.is-active`

ID selector — #id

#header

#search-button

#dropdown

Attribute selector — `[attr]`

`[disabled]`

`[role="dialog"]`

`[data-component*="popup"]`

Universal selector — \*

\*



Cascade selector — E F

ul li

div a

header h1 .link

Child selector — E > F

```
ul > li
```

```
.menu > .item
```

```
footer > img
```

General sibling selector — E ~ F

```
input ~ label
```

```
span ~ em
```

```
.image ~ .link
```

Adjacent sibling selector — E + F

input + label

span + em

.image + .link

# PSEUDO CLASSES

Starts with single colon

# Links

`:link`

`:active`

`:visited`

`:any-link`

## Events

:hover

:focus

:focus-within

:target

:active

:checked

# Forms

:enabled

:disabled

:valid

:invalid

:indeterminate

:required



# Elements

`:first-child`

`:last-child`

`:first-of-type`

`:last-of-type`

`:nth-child(n)`

`:nth-last-child(n)`

Negation

:not()

General

:root

# PSEUDO ELEMENTS

Starts with double colon

## Generated content

::before & ::after

```
div::before {  
    content: "before << ";  
}  
div::after {  
    content: " >> after";  
}
```

before << This is text paragraph >> after

Input placeholder

::placeholder

Enter some text

# AT-RULE SELECTORS

@charset

@media

@page

@keyframes

@supports

CSS Selectors Level 4 @ W3C

CSS Selectors @ KharkivCSS #2

HTML5 Elements and their styling @ KharkivCSS #3

# CSS UNITS

px em % s ms fr vw vh ...etc



# RELATIVE UNITS

`em` font-size of the parent element

`rem` font-size of root element

`ex` height of the "x" letter in lowercase

`ch` width of the "0" symbol

`%` relative to the same property of the parent element

# VIEWPORT-RELATIVE UNITS

`vw` 1% of the width of the viewport

`vh` 1% of the width of the viewport

`vmin` 1% of viewport's\* smaller dimension

`vmax` 1% of viewport's\* larger dimension

# ABSOLUTE UNITS

mm millimeters

in inches ( $1\text{in} = 96\text{px} = 2.54\text{cm}$ )

px pixels ( $1\text{px} = 1/96\text{th of } 1\text{in}$ )

pt points ( $1\text{pt} = 1/72\text{ of } 1\text{in}$ )

pc picas ( $1\text{pc} = 12\text{ pt}$ )

# NOTES

**em** and **px** are most usable units for fonts, **rem** becoming popular

**%** is the most popular unit for layout due to responsive

**vw/vh** usually used for dimensions which relate to viewport

Most of absolute units are used preferably for print page rules

CSS Values and Units Module Level 4

# MATH

Native CSS way to do simple math right in CSS. The most useful ability of `calc()` is its ability to mix units.

```
ul {  
  width: calc(100% - 20px);  
  font-size: calc(0.5em + 6px);  
}
```

CSS Values and Units Module Level 3

## calc() as CSS unit value - CR

Method of allowing calculated values for length units, i.e. width:

```
calc(100% - 3em)
```

### Usage

Global

unprefixed:

% of all users

$$89.12\% + 3.01\% = 92.12\%$$
$$88.97\% + 3.01\% = 91.98\%$$
[illegible]

# CSS VARIABLES

```
:root {  
    --font-stack: Helvetica, sans-serif;  
    --primary-color: #333;  
}  
  
body {  
    font: 100% var(--font-stack);  
    color: var(--primary-color);  
}
```

CSS Custom Properties for Cascading Variables Module Level 1

 - CR

Permits the declaration and usage of cascading variables in stylesheets.

### Usage

Global

% of all users

$$86.81\% + 0.05\% = 86.85\%$$
[illegible]



**THANKS!**

Questions?