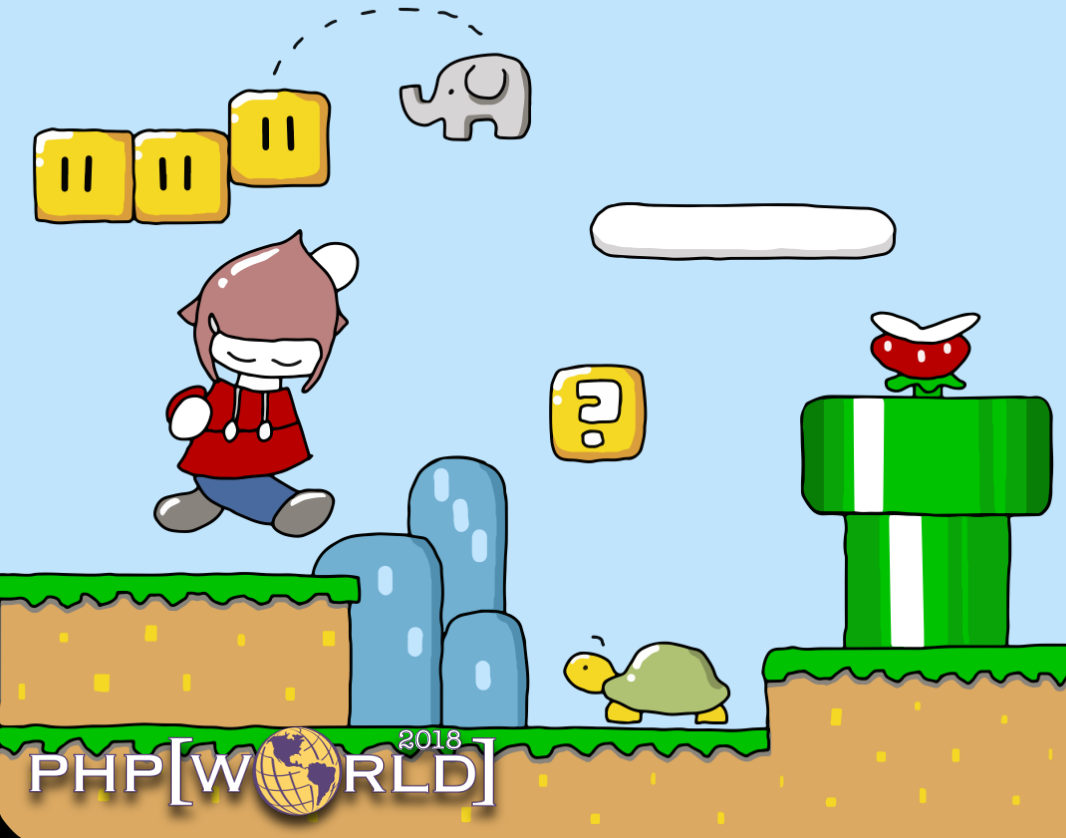


JEN KRAMER • HARVARD UNIVERSITY EXTENSION SCHOOL
@JEN4WEB

MODERN LAYOUTS WITH FLEXBOX, GRID, CALC, AND CUSTOM FIELDS

PART 3: CALC AND CUSTOM FIELDS

Thanks to Our Sponsors



2018
PHP[WORLD]

AUTOMATTIC



WHAT IS CSS CALC?

- Ability to do math in CSS
- Compatible with length, frequency, angle, time, number, and integer
 - Addition: +
 - Subtraction: -
 - Multiplication: *
 - Division: /
- `width: calc(100% - 80px);`

calc() as CSS unit value - CR

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

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

Usage	% of all users	
Global	92.57%	+ 0.58% = 93.15%
unprefixed:	92.42%	+ 0.58% = 93%

[illegible]

ADVANTAGES

- Can mix units when performing calculations (not possible in Sass)

```
.thing {  
    width: 90%; /* fallback if needed */  
    width: calc(100% - 3em);  
}
```

- Documentation of math (how do you derive a number?)
- <https://css-tricks.com/a-couple-of-use-cases-for-calc/>

ADVANTAGES: MIXING UNITS, CALCULATING BOX SIZES

```
.area-one {  
    width: 40%;  
  
    float: left;  
  
    margin-right: 1em;  
  
}  
  
.area-two {  
  
    width: calc(60% - 1em);  
  
    float: right;  
  
}
```

ADVANTAGES: MAKE MATH EASIER TO UNDERSTAND

```
.column-1-7 {  
    width: 14.2857%  
}
```

```
.column-2-7 {  
    width: 28.5714%  
}
```

```
.column-3-7 {  
    width: 42.8571%  
}
```

```
.column-1-7 {  
    width: calc(100% / 7);  
}  
  
.column-2-7 {  
    width: calc(100% / 7 * 2);  
}  
  
.column-3-7 {  
    width: calc(100% / 7 * 3);  
}
```

SYNTAX QUIRKS

- The + and - operators must be surrounded by whitespace.
 - `calc(50% -8px)` will be parsed as a percentage followed by a negative length—an invalid expression—while `calc(50% - 8px)` is a percentage followed by a subtraction operator and a length.
 - Likewise, `calc(8px + -50%)` is treated as a length followed by an addition operator and a negative percentage.
- The * and / operators do not require whitespace, but adding it for consistency is both allowed and recommended.
- In short: a `calc()` inside of a `calc()` is identical to just parentheses.
- <https://developer.mozilla.org/en-US/docs/Web/CSS/calc>

CALC CAN BE USED WITH SASS

```
$a: 4em;
```

```
div {
```

```
    height: calc(#{ $a } + 7px);
```

```
    background: #e53b2c;
```

```
}
```

<https://codepen.io/thebabydino/pen/hgFno>

PEDMAS

```
.col-3 {
```

```
    flex: 0 0 calc( 20% * 3 + 4% * 2);
```

```
}
```

- Please Excuse My Dear Aunt Sally (PEDMAS)
 - Parenthesis
 - Exponents
 - Multiplication/Division, left to right
 - Addition/Subtraction, left to right

SUMMARY

- Use `calc()` anywhere in CSS to perform numeric calculations.
 - Don't forget about whitespace around the operator.
 - You can mix and match your units.
 - You can use parentheses or nest `calc()` statements.
 - PEDMAS applies.
 - Well-supported by most browsers in use.

PART 2

CSS CUSTOM FIELDS (VARIABLES)

WHAT IS CSS CUSTOM FIELDS?

- Kind of like variables, but with weird quirks
 - Must declare custom field within a CSS property – just like other CSS declarations
 - Values inherit – just like elsewhere in CSS
 - Can be used in JavaScript

 - CR

Permits the declaration and usage of cascading variables in stylesheets.

Usage	% of all users
Global	87.57% + 0.06% = 87.63%

[illegible]

DECLARING A VARIABLE

```
:root {  
    --primary-color: blue;  
    --gap: 1.75rem;  
}  
  
.wrapper {  
    background-color: var(--primary-color);  
}  
  
h1 {  
    color: var(--primary-color);  
    margin-bottom: calc(var(--gap) * 5px);  
}
```

DECLARING A VARIABLE

```
:root {  
  --primary-color: blue;  
  --default-size: 1rem;  
  --bkgd-img: url(..img/myimage.jpg);  
  --center-margin: 0 auto;  
  --bezier: cubic-bezier(.17,.67,.83,.67);  
}
```


CSS VARIABLES VS SASS VARIABLES

```
$primary-color: #123456;
```

```
h1 {
```

```
    color: $primary-color;
```

```
}
```

```
article {
```

```
    border: 1px solid $primary-color;
```

```
}
```

After compiling:

```
h1 {
```

```
    color: #123456;
```

```
}
```

```
article {
```

```
    border: 1px solid #123456;
```

```
}
```

<https://css-tricks.com/difference-between-types-of-css-variables/>

CSS VARIABLES VS SASS VARIABLES

```
:root {  
    --primary-color: #123456;  
}  
  
h1 {  
    color: var(--primary-color);  
}  
  
article {  
    border: 1px solid var(--primary-color);  
}
```

<https://css-tricks.com/difference-between-types-of-css-variables/>

CSS VARIABLES AND JAVASCRIPT

<https://developers.google.com/web/updates/2016/02/css-variables-why-should-you-care> (scroll down halfway)

CSS VARIABLES HAVE INHERITANCE BUT IT'S WEIRDER THAN YOU THINK

- Sass: <https://codepen.io/jen4web/pen/rQMVKr?editors=1100>
- Variables: <https://codepen.io/jen4web/pen/RqGPxw?editors=1100>

USING INHERITANCE IN MEDIA QUERIES WITH VARIABLES

<https://codepen.io/jen4web/pen/oQzxr?editors=1100>

(also rudimentary type scaling)

QUESTIONS?

Jen Kramer

Watertown, MA, USA

Phone: 802-257-2657

jen@jenkramer.org

www.jenkramer.org

Twitter: @jen4web

Facebook: facebook.com/webdesignjen

Code available at
www.github.com/jen4web/phpworld

Slides available at
www.slideshare.net/jen4web

CSS
IS
AWESOME