

CSS

CSS FUNDAMENTALS

Context



IN A ROCKET

Learn front-end development at *rocket speed*

PAST

+

PRESENT

+

FUTURE



+



MATURITY OF SPECIFICATIONS BY STATUS CODE



- ♦ Box properties
- ♦ Cascade
- ♦ Color + background
- ♦ CSS syntax
- ♦ Font properties
- ♦ Pseudo-classes
- ♦ Text properties

CSS 1

CSS Level 1

RECOMMENDATION

DEC 1996

CSS 2

CSS Level 2

RECOMMENDATION

MAY 1998

CSS 1 + CSS 2 +

- ♦ Box model
- ♦ Generated content
- ♦ Media types
- ♦ Paged media
- ♦ Selectors
- ♦ Tables
- ♦ Visual effects

CSS 2.1

CSS Level 2
Revision 1

RECOMMENDATION

JUN 2011

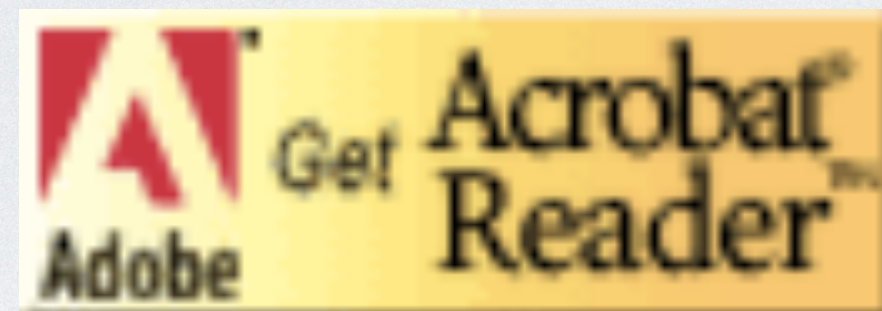


+



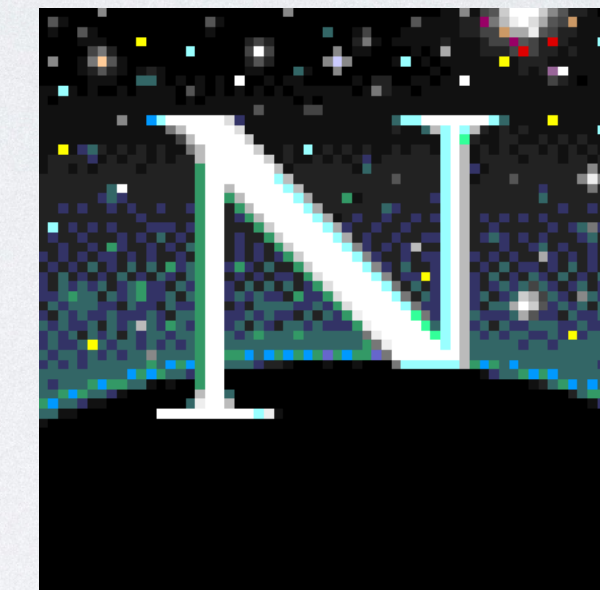
STANDARDS WERE OFTEN IGNORED

SITE OPTIMIZED FOR:



etc.

BROWSERS REQUIRED MANUAL UPDATES



PAST

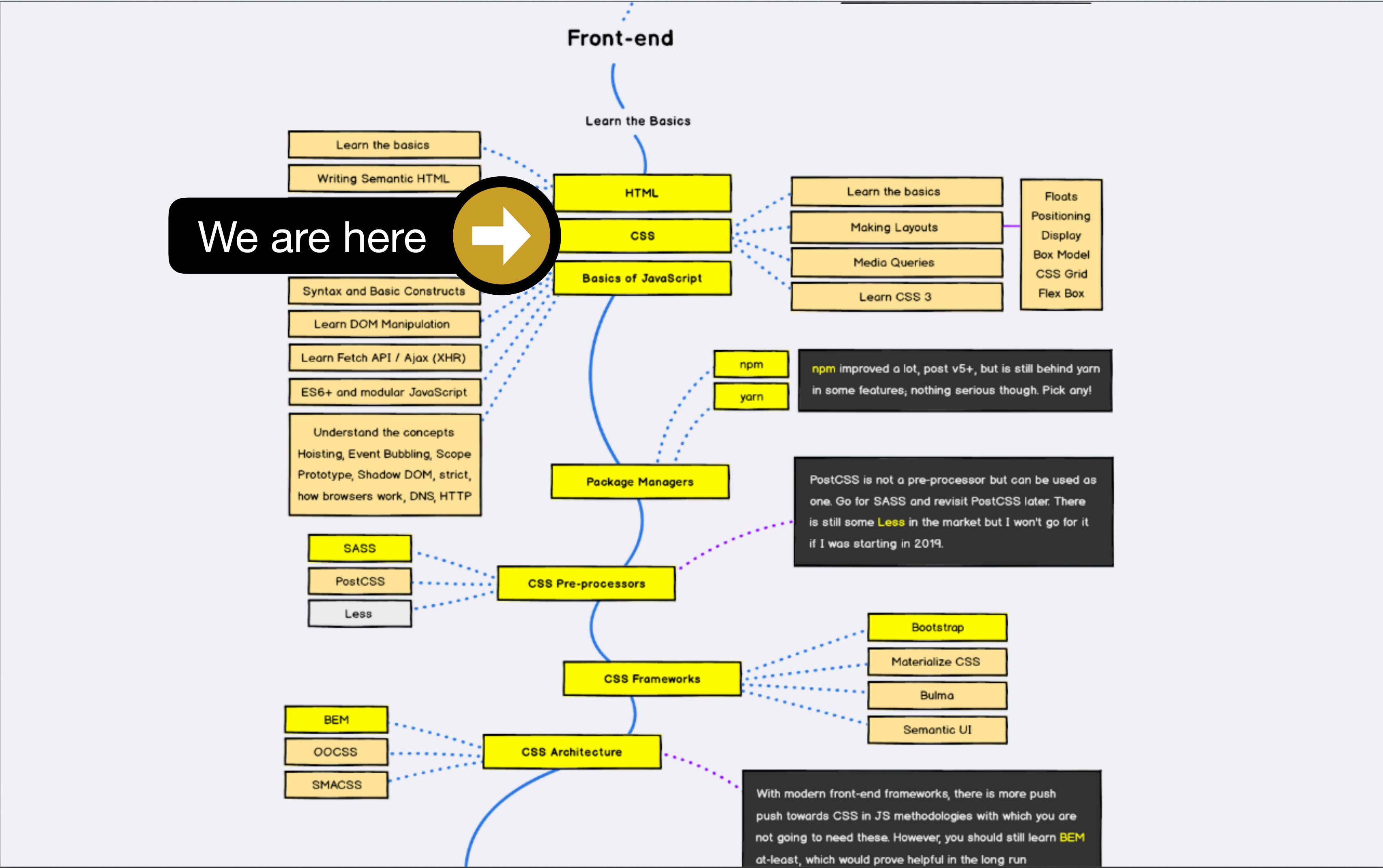
+

PRESENT

+

FUTURE

FRONT-END DEVELOPER ROADMAP



SOURCE: [Front-end developer roadmap by Kamran Ahmed](#).



+



CSS3




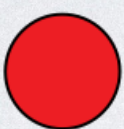
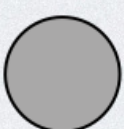
CSS4

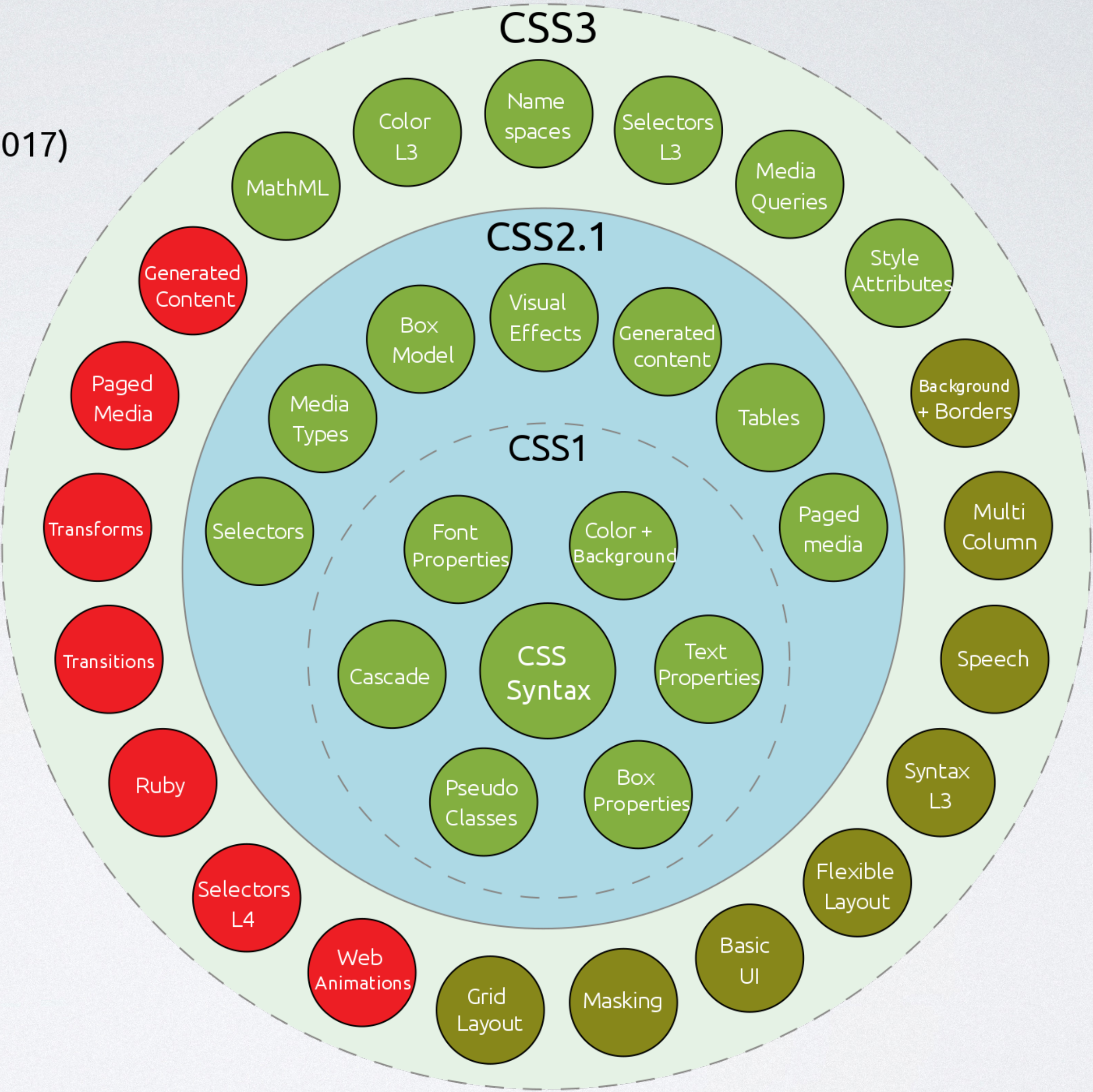
CSS?

CSS3: TAXONOMY AND STATUS BY SERGEY MAVRODY

CSS3

Taxonomy & Status (September 2017)

-  W3C Recommendation
-  Candidate Recommendation
-  Last Call
-  Working Draft
-  Obsolete or inactive

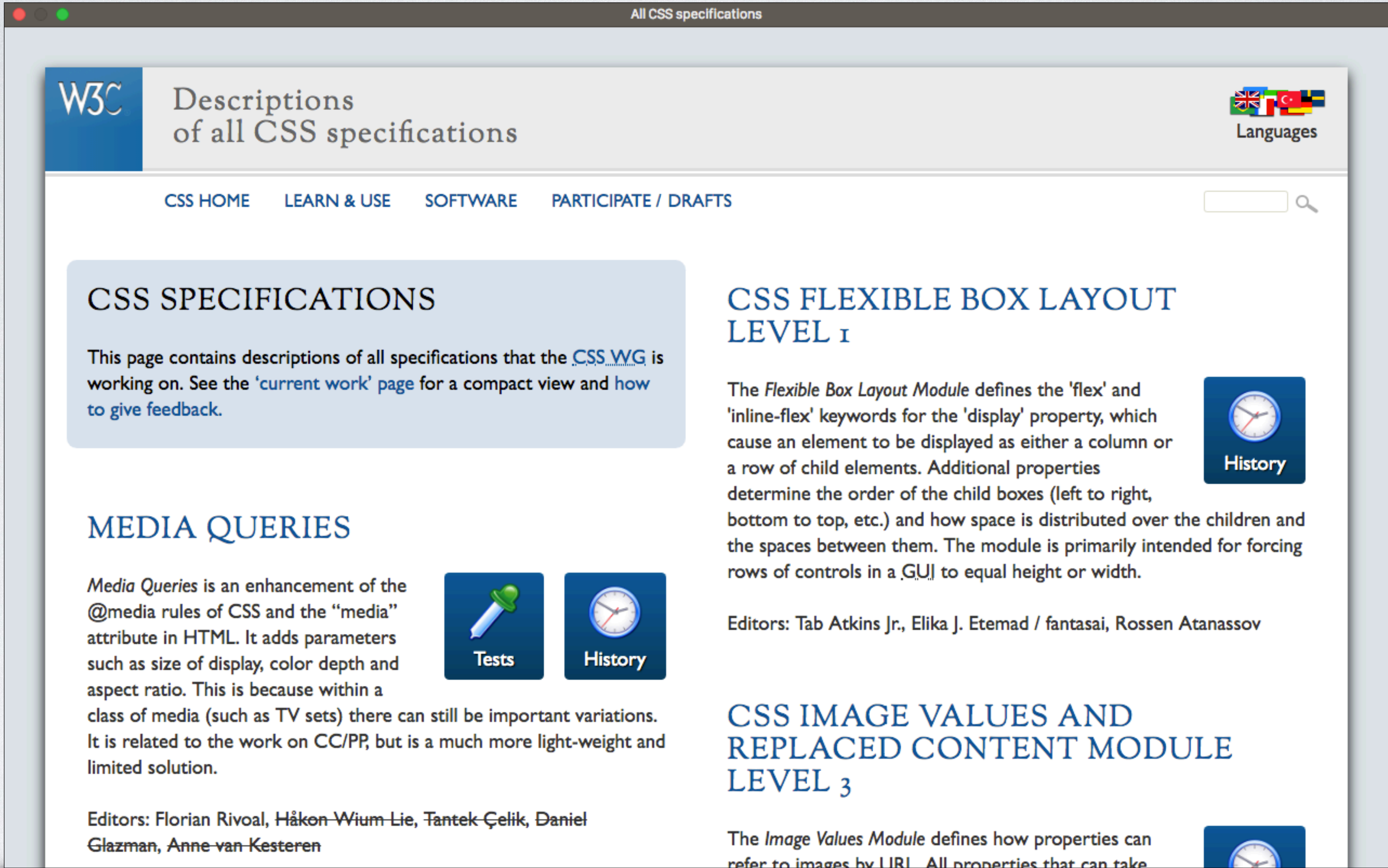


SOURCE: [HTML5 & CSS3: Quick Reference by Sergey Mavrody](#).

+80

SPECIFICATIONS

ALL CSS SPECIFICATIONS



CSS current work & how to participate

W3C

CSS current work & how to participate

UK

FR

DE

IT

ES

PT

RU

JP

IN

BR

US

CA

MX

AR

CL

CO

PE

VE

EC

UY

PA

CR

GT

HN

DO

PR

VI

US

CA

MX

AR

CL

CO

PE

VE

EC

UY

PA

CR

GT

HN

DO

PR

VI

Languages

CSS HOME

LEARN & USE

SOFTWARE

PARTICIPATE / DRAFTS

CSS SPECIFICATIONS

This page contains a list of all completed specifications and drafts by the [CSS WG](#) (formerly 'CSS & FP WG'). If you want to follow the development of CSS3, this is the place to start. You have ideas? Contributions? See 'If you want to help' on this page.

A *specification* is not a manual. There is no excuse for badly written drafts and please complain if you find one. But specs do target a specific audience. See fantasai's *Understanding the CSS Specifications..* J. David Eisenberg has written another useful *How to read W3C specs*. Or you can read about 'modules,' 'levels,' 'snapshots' and the CSS process.

Worklets Level 1

FPWD

WD

i

CSS Color Level 4

FPWD

WD

i

CSS Fonts Level 4

WD

WD

i

CSS Rhythmic Sizing Level 1

FPWD

WD

i

CSS Image Values and Replaced Content Level 4

WD

WD

i

CSS Fill and Stroke Level 3

FPWD

WD

i

CSS Logical Properties and Values Level 1

FPWD

WD

i

CSS Overflow Level 4

FPWD

WD

i

CSS Grid Layout Level 2

FPWD

WD

i

CSS Text Decoration Level 4

FPWD

WD

i

CSS Layout API Level 1

FPWD

WD

i

WHAT'S NEW?

2018-07-03

Updated Candidate Recommendation: *CSS Text Decoration Level 3*.

2018-06-28

Updated Working Draft: *CSS Grid Layout Level 2*.

2018-06-26

Updated Candidate Recommendation: *CSS Fonts Level 3*.

2018-06-21

New Recommendation: *CSS Basic User Interface Level 3*.

Rewriting

CSS Basic Box Model Level 3

WD

WD

Notes

Dangerously outdated; see CSS 2.1

i

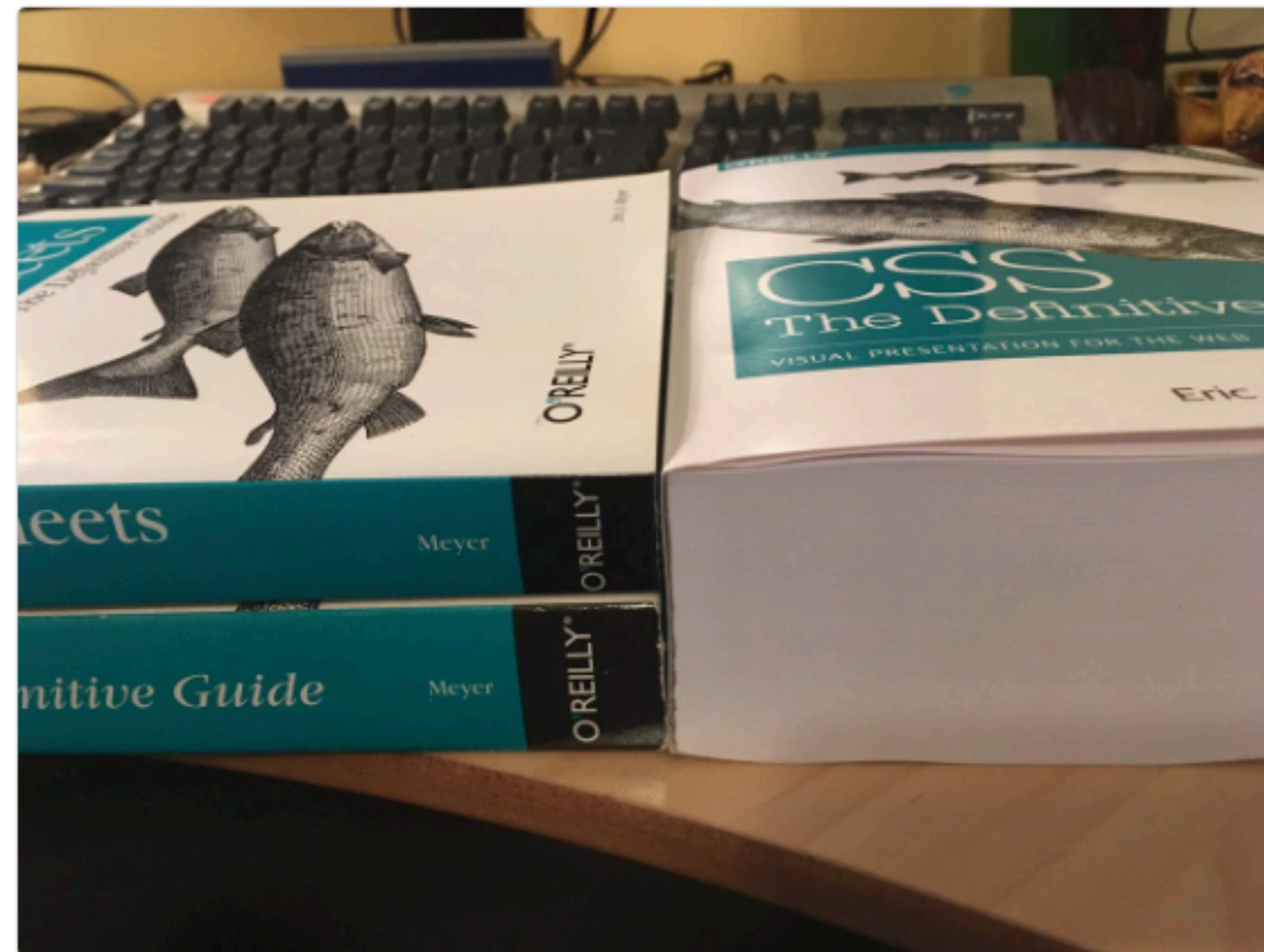


Eric Meyer
@meyerweb

Following



On the left: the 2nd and 3rd editions of “CSS: The Definitive Guide”. On the right, a single copy of the fourth edition.



4:25 PM - 10 Nov 2017

1,316 Retweets 3,303 Likes





+





Evergreen browsers

PAST

+


PRESENT

+

FUTURE



**POSTCSS
PRESET ENV**



**CSS
HOUDINI**

POSTCSS PRESET ENV

You can write future CSS syntax, and the plugin will convert it to CSS that browsers will understand (using polyfills).

postcss-preset-env

Use tomorrow's CSS today.



Features



Try It Now



Support





GitHub


PostCSS Preset Env lets you convert modern CSS into something most browsers can understand, determining the polyfills you need based on your targeted browsers or runtime environments, using [cssdb](#).

```
npm install postcss-preset-env
```


postcss-preset-env

 Features

 Try It Now

 Support

Stage 0+

all property

:any-link pseudo-class

:blank empty-value pseudo-class

break properties

case-insensitive attributes

functional color notation

custom media queries

custom properties

custom selectors

:dir pseudo-class

double position gradients

custom environment variables

:focus-visible pseudo-class

:focus-within pseudo-class

font-variant property

gap properties

gray() function

alpha hex colors

hwb() function

image-set() function

lab() function

all property

STAGE 3

SPECIFICATION

PLUGIN

A property for defining the reset of all properties of an element

```
a {  
  all: initial;  
}
```

:any-link pseudo-class

STAGE 2

SPECIFICATION

PLUGIN

A pseudo-class for matching anchor elements independent of whether they have been visited

```
nav :any-link > span {  
  background-color: yellow;  
}
```

:blank empty-value pseudo-class

STAGE 1

SPECIFICATION


PLUGIN

A pseudo-class for matching form elements when they are empty

```
input:blank {
```




**POSTCSS
PRESET ENV**



**CSS
HOUDINI**

CSS HOUDINI

Group of APIs for direct access to the CSS Object Model

Box Tree API

Layout API

Painting API

Parser API

Properties & values API

Font metrics API

Typed OM

Worklets

Animation worklet




























CSS HOUDINI: W3C EDITOR DRAFTS









W3C

Home

Login

CSS-TAG Houdini Editor Drafts

Specification	Last Update	By	
Box Tree API 1	2018-08-14 05:14:43 PDT	philip_j_genstedt	 
CSS Animation Worklet 1	2019-08-12 06:57:43 PDT	majidvp	  
CSS Layout API 1	2019-02-20 20:05:55 PST	philip_j_genstedt	  
CSS Painting API 1	2019-07-23 13:59:26 PDT	tabatkins	   
CSS Parser API 1	2017-11-09 13:05:45 PST	dbaron	  
CSS Properties and Values API 1	2019-07-31 10:34:16 PDT	tabatkins	 
css-typed-om			
CSS Typed OM 1 (Current Work)	2019-08-21 22:35:09 PDT	kagami_sascha_rosylight	  
CSS Typed OM 2	2018-05-10 15:34:50 PDT	tabatkins	 
Font Metrics API 1	2019-02-11 05:11:49 PST	philip_j_genstedt	  
Worklets 1	2019-08-20 09:04:58 PDT	kagami_sascha_rosylight	 

Other Documents	Last Update	By	
CODE OF CONDUCT	2018-05-23 13:15:48 PDT	plh	 
CONTRIBUTING	2018-05-23 13:15:47 PDT	plh	 
LICENSE	2018-05-23 13:15:46 PDT	plh	 
README	2018-06-11 07:23:45 PDT	astearns	 

Please send comments, questions, and error reports to www-style@w3.org

CSS HOUDINI: EXPERIMENTS

CSS HOUDINIexperimental showcase by @iamvdo

Need latest Chrome with flag!Support?

BACKGROUNDS

Conic gradient •

Corners gradient •

Slanted backgrounds •

Checkboxes ••

Background properties ••

JS-in-CSS ••

BORDERS

Simple tooltip ••

Inner borders •

Rough boxes ••

Non-rectangular separators ••

MASKS

Corner shapes ••

Avatar polygons ••

Smooth corners ••

Dynamic hover masks ••

Random bubbles mask ••

ANIMATIONS

Animating a gradient •

Ripple animation •

Highlighter marker annotations ••

Double rainbow •

Wobble animation ••

LAYOUT

EQ: Responsive background ••

Random irregular grid •

CSS Houdini Experiments

Here is my experiments with the latest CSS Houdini spec. You'll need the latest Chrome (with Web Platform flag enabled). See [support](#)

[Source code on GitHub](#)

Let's do it!

First, see [how to register a new custom property](#) (and why). Then, check [how to write your own custom paint](#) to use with `background-image`. Or [borders](#). Or [CSS masks](#).

At the end, [interact with user inputs](#) and [animate it](#) !

Learn Houdini


- [Houdini on Smashing Mag](#)
- [Houdini and the paint API](#)
- [Working with the new CSS Typed Object Model](#)
- [Awesome CSS Houdini](#)

Specification links

- [CSS Properties & Values API](#)
- [CSS Typed OM](#)
- [CSS Painting API](#)
- [CSS Layout API](#)
- [All other Houdini specs](#)

CSS HOUDINI: BROWSER SUPPORT

Is Houdini ready yet?

	 Google Chrome	 Mozilla Firefox	 Opera	 Microsoft Edge	 Samsung Internet	 Apple Safari	 Spec
Layout API (Explainer Demos)	Partial support (Canary) Details	Intent to implement Details	No signal	No signal	No signal	No signal	First Public Working Draft Spec
Paint API (Explainer Demos Article)	Shipped (Chrome 65) Details	Intent to implement (Servo) Details	Shipped (Opera 52) Details	No signal	Shipped (Internet 9.2) Details	In Development Details	Candidate Recommendation Spec
Parser API (Explainer)	No signal	No signal	No signal	No signal	No signal	No signal	No spec
Properties & Values API (Demos)	Partial support (Canary) Details	In Development Details	No signal	No signal	No signal	Partial support (Safari TP 67) Details	Working Draft Spec
AnimationWorklet (Explainer Demos Article)	Partial support (Canary) Details	No signal	No signal	No signal	No signal	No signal	First Public Working Draft Spec
Typed OM (Explainer Article)	Shipped (Chrome 66) Details	Intent to implement Details	Shipped (Opera 53) Details	No signal	Shipped (Internet 9.2) Details	In Development Details	Working Draft Spec
Font Metrics API (Explainer)	No signal	No signal	No signal	No signal	No signal	No signal	No spec

PAST



PRESENT



FUTURE



CONTENT IS KING, NOT CSS



SOURCE: [The Web Pyramid](#) by @pinboard.

YOU CAN CONTINUE THIS COURSE FOR FREE ON

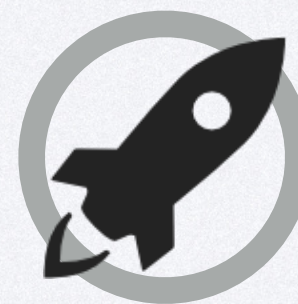
inrocket.com

Learn front-end development at *rocket speed*

+ READY TO USE CODE

+ QUIZZES

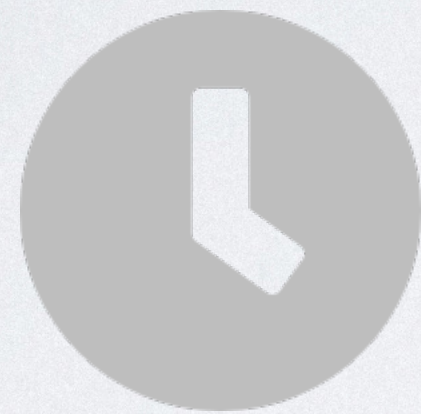
+ FREE UPDATES



by miguelsanchez.com

inrocket.com

Learn front-end development at *rocket speed*



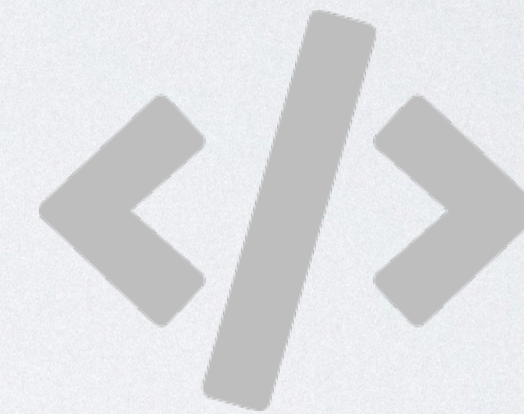
We respect your time

No more blah blah videos. Just straight to the point slides with relevant information.



Step by step guides

Clear and concise steps to build real use solutions. No missed points.



Ready to use code

Real code you can just copy and paste into your real projects.

CSS

CSS FUNDAMENTALS

Context



IN A ROCKET

Learn front-end development at *rocket speed*