

CSS and RWD basics

Miloš Maksimović,
Senior Software Engineer,
Toptal



CSS history

- First proposed by Håkon Wium Lie on 1994
- CSS1, a W3C recommendation, released in 1996
- CSS Level 2 – 1997 – added positioning, z-index, media types etc.
- CSS Level 3 – 1998 – still under development, de-facto standard
 - Added media queries, namespaces, entire list of selectors
 - Maintained backwards compatibility

Selectors

universal selectors

```
* {  
  color: #000000;  
}
```

element selectors

```
h1 {  
  color: #36CFFF;  
}
```

descendant selectors

```
ul em {  
  color: #36CFFF;  
}
```

class selectors

```
h1.black {  
  color: #000000;  
}
```

id selectors

```
h1#first-title {  
  color: #36CFFF;  
}
```

child selectors

```
div > p {  
  color: #36CFFF;  
}
```

grouping selectors

```
h1, h2, h3 {  
  color: #36C;  
  font-weight: normal;  
}
```

pseudo-class selectors

```
h1:hover {  
  color: #36CFFF;  
}
```

pseudo-element selectors

```
p::before {  
  content: "Read this: ";  
}
```

Inclusion in HTML

- Hierarchy of overriding styles in html (most important – last to be processed first)
 - Inline styles - style attribute of any HTML element
 - Embedded styles - <style> HTML tag
 - External - <link> tag with href attribute set to point to external .css file
 - Additionally we usually set rel="stylesheet" (required) and type="text/css" when referencing an external file
- Note: using *!important* flag trumps the hierarchy

Measurements

- % - ex: line-height:125%
- cm/mm/in - ex: margin-bottom: 2cm;
- em – times the original measure - ex: letter-spacing: 7em;
- rem – newer, times the original measure of the root element
- px - ex: line-height: 55px
- pt – a point = 1/72nd of an inch - ex: line-height: 55pt

Dimensions

- There are some dimensions that are unique to all elements. Those are set with the following CSS properties:
 - height
 - width
 - line-height - space between lines of text
 - min/max-height/width – these contain the element to defined max/min

Fonts

- font-family - ex. georgia,garamond,serif
- font-style - normal, italic, oblique
- font-variant - normal, small-caps
- font-weight - normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900
- font-size - xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger, size in px/in/%
- font-stretch - wider, narrower, condensed...
- font - a shorthand property

```
body {  
    font: font:italic small-caps bold 15px georgia;  
}
```

Colors

- #RRGGBB
- #RGB
- color: rgb(50%,50%,50%)
- color: rgb(0,0,255);
- color: teal;
- Browser/web safe - all computers would display the colors correctly when running a 256 color palette. There are 216 of them.
- <https://color.adobe.com/create/color-wheel>

The Browser Safe Colors					
#000000	#000033	#000066	#000099	#0000CC	#0000FF
#003300	#003333	#003366	#003399	#0033CC	#0033FF
#006600	#006633	#006666	#006699	#0066CC	#0066FF
#009900	#009933	#009966	#009999	#0099CC	#0099FF
#00FF00	#00FF33	#00FF66	#00FF99	#00FFCC	#00FFFF

Display

- inline
- block
- inline-block – the inside is formatted as block and the element itself is inline
- none – use it to hide the element altogether
- list-item – behavior of a element
- table - mimics the same table model as HTML Table but with CSS properties.
- flex – latest and greatest from CSS. Check out tutorials for more details

<http://flexboxfroggy.com/>

Positioning and layers

- position: relative
 - relative to its usual position
 - and then using left, right, top, bottom
- position: absolute
 - relative to screen top left corner
 - and then using left, right, top, bottom
- position: fixed
 - relative to screen top left regardless of scrolling
 - and then using left, right, top, bottom
- Layers: z-index - used in combination with position to create the effect of layers
- http://www.w3schools.com/cssref/playit.asp?filename=playcss_position

 <p>Default position</p>	 <p>Moved across 1 Moved down 2</p>	 <p>Centered horizontally automatically</p>
	<code>position:relative; 1. left:100px; 2. top:50px;</code>	<code>margin:0 auto;</code>
 <p>Side-by-side</p>	 <p>Overlapping</p>	 <p>Out of order</p>
<code>1: position:absolute; top:40px;left:40px; 2: position:absolute; top:40px;left:100px;</code>	<code>position:absolute; 1: top:10px;left:30px; 2: top:30px;left:50px; 3: top:50px;left:70px;</code>	<code>position:absolute; 1: top:30px;left:50px; 2: top:50px;left:70px; 3: top:10px;left:30px;</code>

Background

- background-color
- background-image - url(/images/pattern1.gif)
- background-repeat - no-repeat/repeat (default)/repeat-x/repeat-y
- background-position - ex. right top
- background-attachment - fixed/scroll
- background - a shorthand property for all

```
body {  
    background: #ffffff url("img_tree.png") no-repeat right top;  
}
```

Text

- color
- direction - rtl, ltr (right-to-left, left-to-right)
- letter-spacing
- word-spacing
- text-indent
- text-align - right, left, center
- text-decoration - none, underline, overline, line-through, blink
- text-transform - capitalize, uppercase, lowercase
- white-space - pre, nowrap, normal
- text-shadow – ex. 4px 4px 8px 8px blue

Images

- border
- height
- width
- opacity – careful here! - not inherited from parent

Borders

- border-color - can be infixed with a particular border: border-left-color etc.
- border-style - none, solid, dotted etc.)
- border-width - in px/in/mm/%
- border - shorthand property, ex:

```
div.bordered {  
    border:4px solid red;  
}
```


Margins

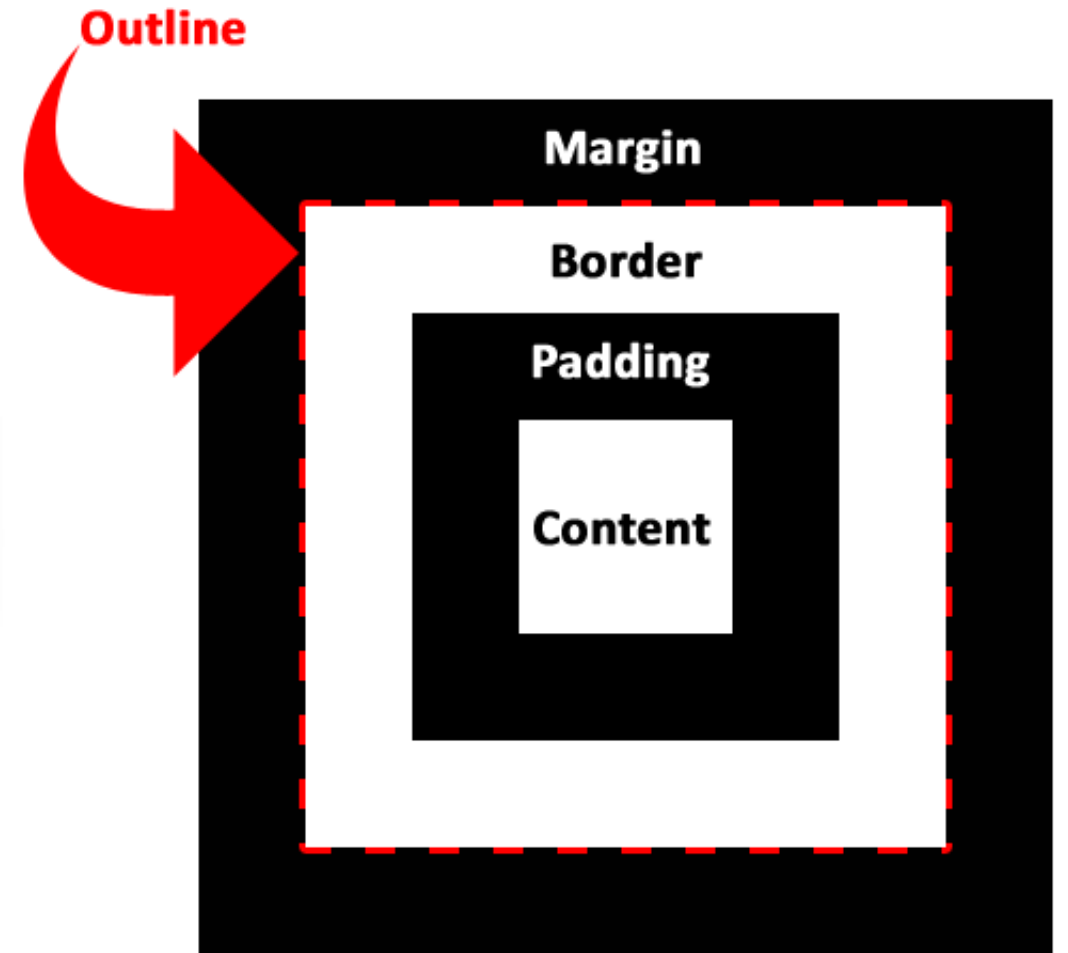
- `margin (-top/bottom/left/right)`
- `margin: 15px` - all margins
- `margin: 15px 10px` - top and bottom 15, left and right 10
- `margin: 15px 10px 20px` - top 15, left and right 10 and bottom 20
- `margin: 15px 10px 20px 5px` - top/right/bottom/left

Padding

- padding (-top/bottom/left/right)
- padding : 15px - all margins
- padding : 15px 10px - top and bottom 15, left and right 10
- padding : 15px 10px 20px - top 15, left and right 10 and bottom 20
- padding : 15px 10px 20px 5px - top/right/bottom/left

Outlines

- similar to borders, except:
 - they do not take up space
 - do not have to be rectangular
 - always same on all sides
- outline-width
- outline-style - none, solid, dashed etc.
- outline-color
- outline - shorthand property



Lists

- list-style-type - style of the marker - none, disc, circle, square for ul, decimal, lower-alpha, upper-roman etc. for ol
- list-style-position - none, inside, outside - wrapping underneath the marker or the start of the first line
- list-style-image - url(/images/bullet.gif)
- list-style - list-style: outside upper-alpha
- marker-offset - distance between marker and text

Pseudo-classes

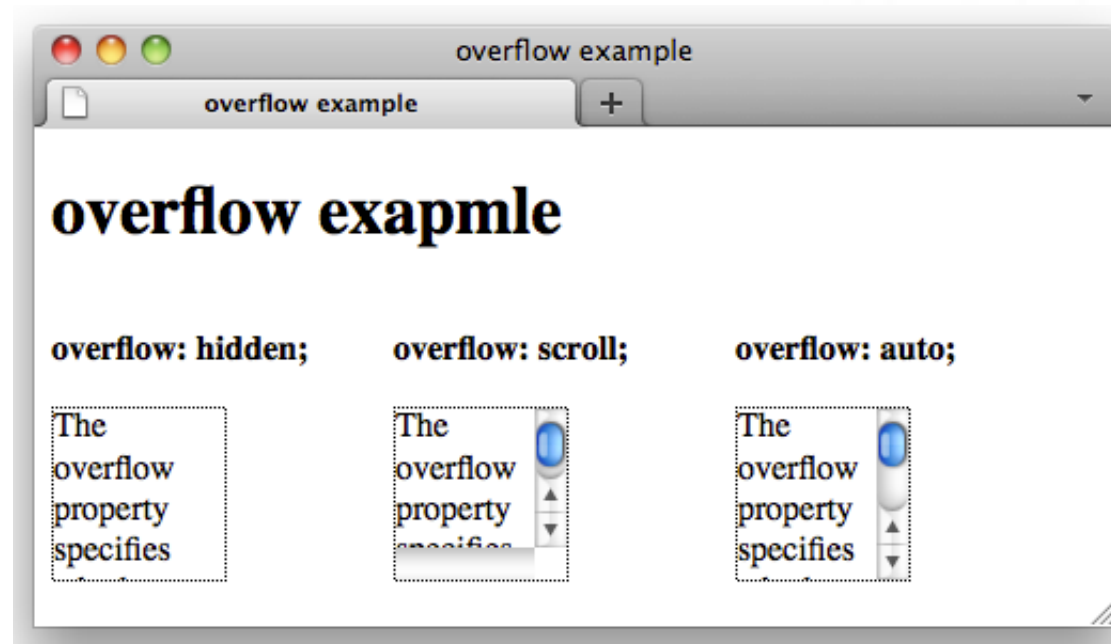
- :link
- :visited
- :hover
- :active
- :first-child
- :last-child
- :nth-child(2)
- :focus
- :required
- :valid
- :read-only

Pseudo-elements

- `::before`
- `::after` (usually used by icons)
- `::selection`
- `::first-letter`
- `::first-line`

Cursors and scrollbars

- cursor (auto, crosshair, default, pointer, move, wait, help etc)
- overflow (visible, hidden, scroll, auto)



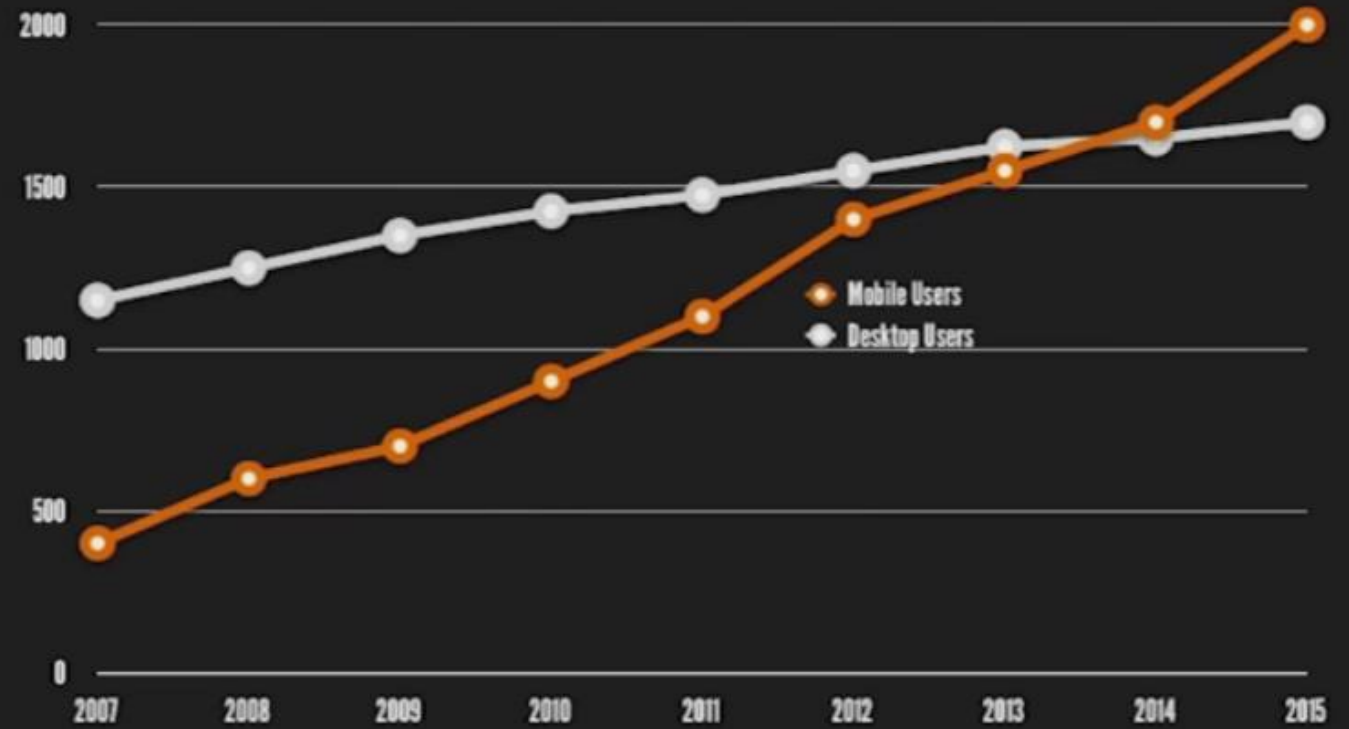
Effects and animations

- animation, blur, glow, gradient etc
- Google it! 😊

Responsive web design (RWD)

- More people are using their mobile devices to access internet than they are computers.
- Make a difference between responsive web and adaptive web!
- 3 techniques:
 - Fluid foundation
 - Flexible content
 - Media Queries

RWD 101



RWD – Fluid foundation

- Using percentages instead of explicit dimensions.
- Target over context.
- Examples: 1-2
- IE before version 8 always rounded up decimal pixels. Modern browsers do the math better.

RWD – Flexible content

- Target over context.
- Images scale really well.
- Do not set height to maintain aspect ratio.

```
img {  
    max-width: 100%;  
}
```

- Examples: 3-4, ratio.html

RWD – Media queries

- Media types – Common: all, screen, print. Other: braille, embossed, speech etc.
- Media features – Common: width, orientation. Other: height, aspect-ratio, device-width, resolution etc.
- Viewport (width) vs device-width!
- Example: 5
- Two ways to approach writing CSS:
 - large resolution first – (most common, issue: small devices)
 - small resolution first (more logical – build up)
- https://seesparkbox.com/foundry/media_query_bookmarklet

```
@media screen and (min-width: 768px) {  
    body { line-height: 1.2 }  
}  
  
@media (orientation: portrait) {  
    body { line-height: 1.2 }  
}
```


RWD – Other considerations

- Touch target area – Apple recommends minimum 44x44px. Use paddings!
- Hover states – Don't hide content behind :hover! Think of mobile devices.
- Contrast – sounds funny, but try your site in the sun and in the dark.
- Readability – small screen != small type – check font size.

Pitanja?

Kontakt

Miloš Maksimović,

Senior Software Engineer, Toptal

milos.m@toptal.com

maksapn@gmail.com