

RESPONSIVE DESIGN





number of devices on which people view Web content is exploding



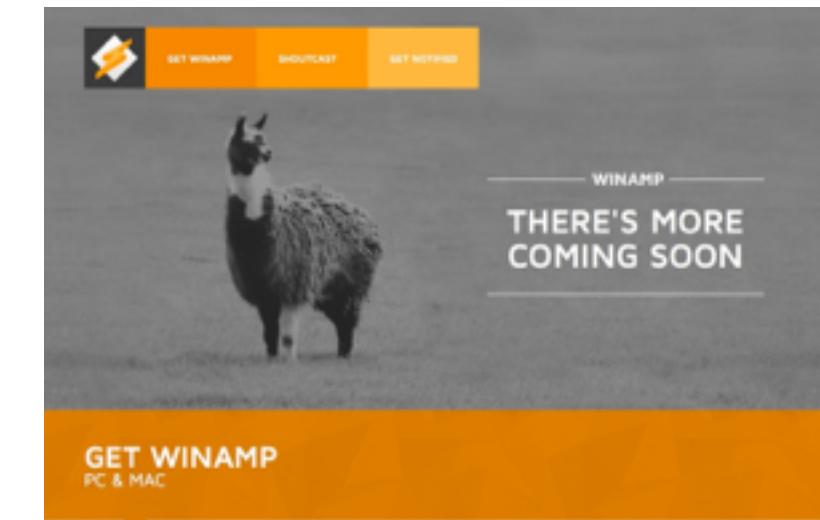
In the US, ~30% of mobile
Web users are mobile-only

GOAL OF RESPONSIVE DESIGN

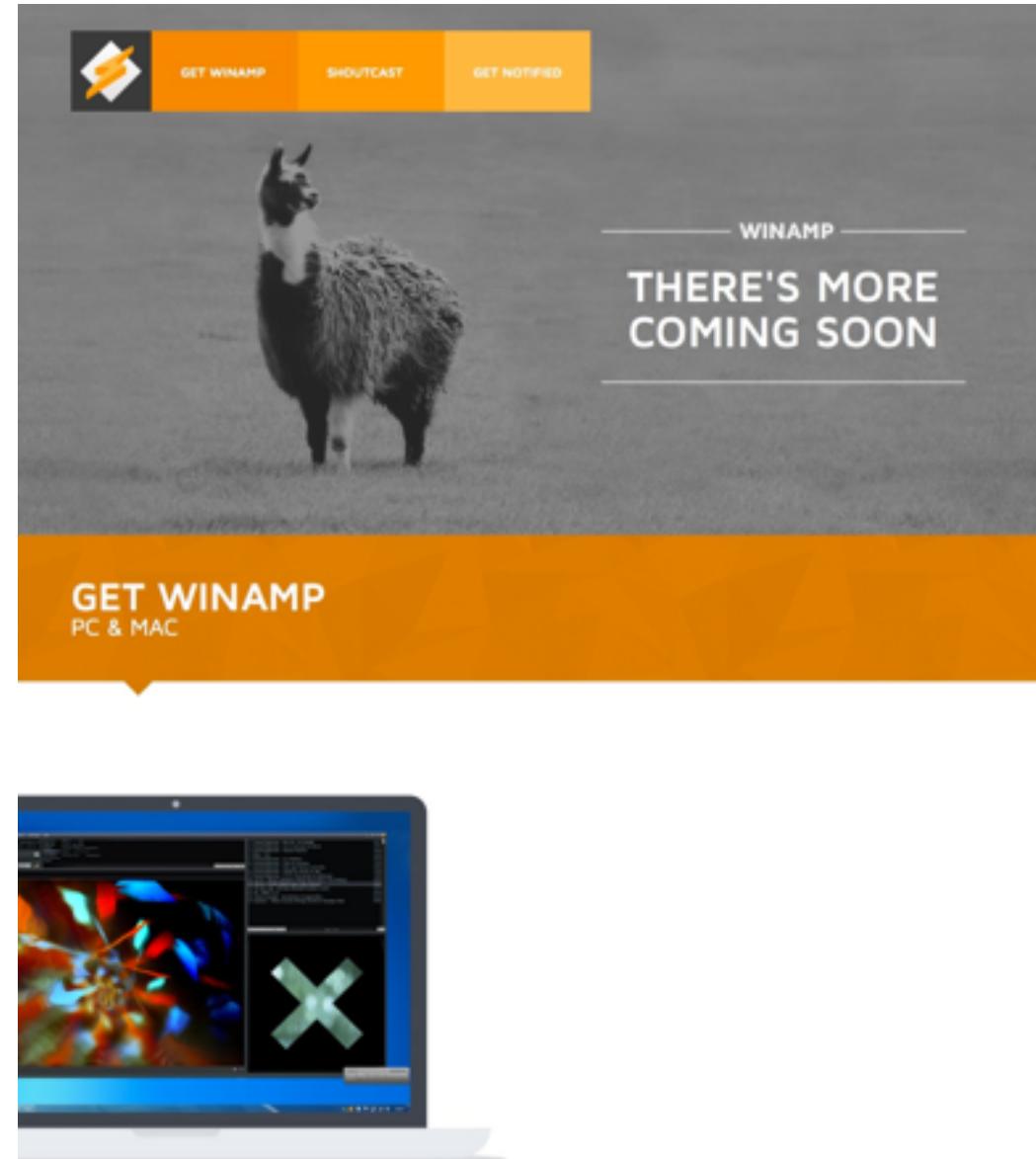
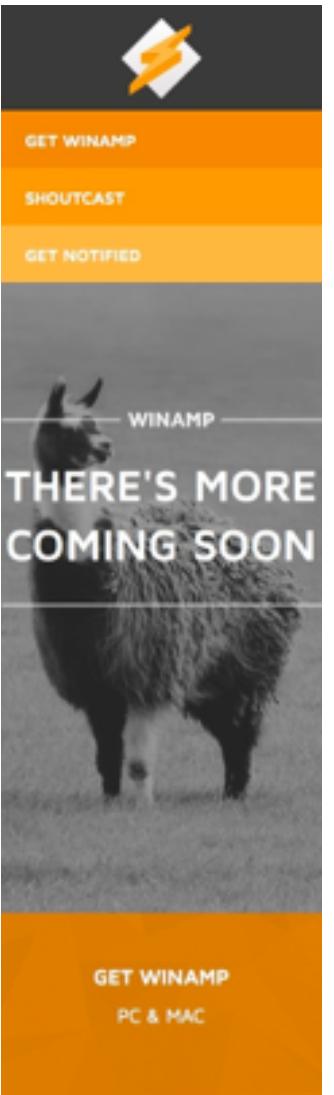
optimize design for different form factors



and everything
in between



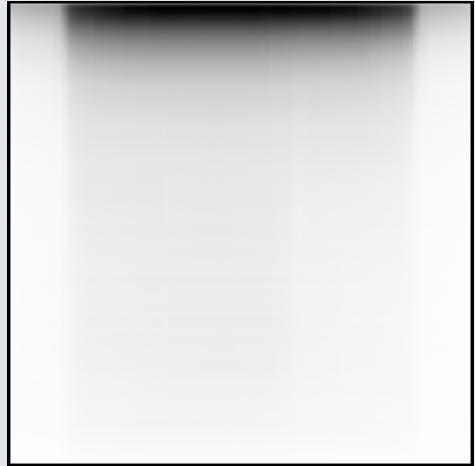
DESIGN CONSIDERATIONS



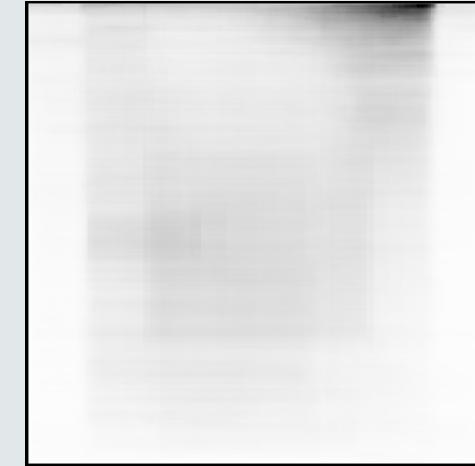
How do styles and layouts change?

How is content remapped?

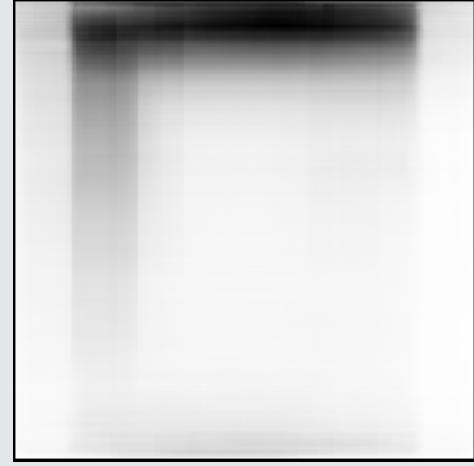
DESKTOP



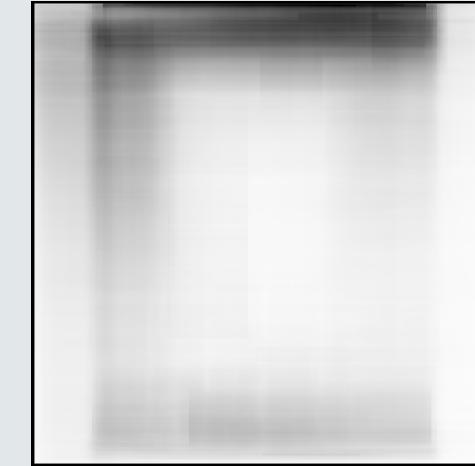
header



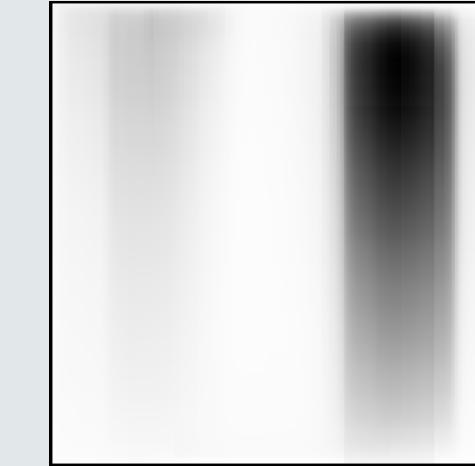
login



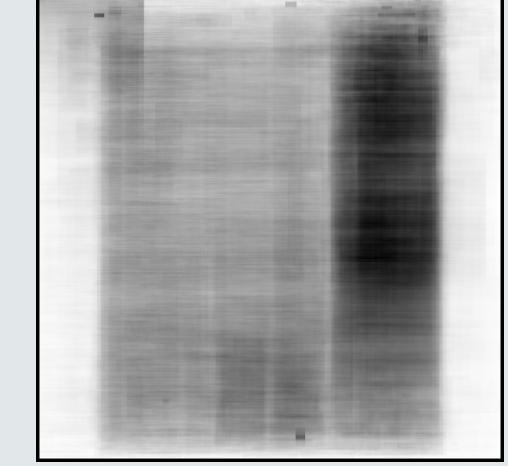
nav



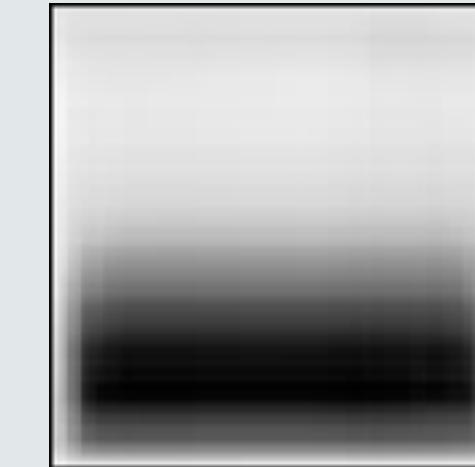
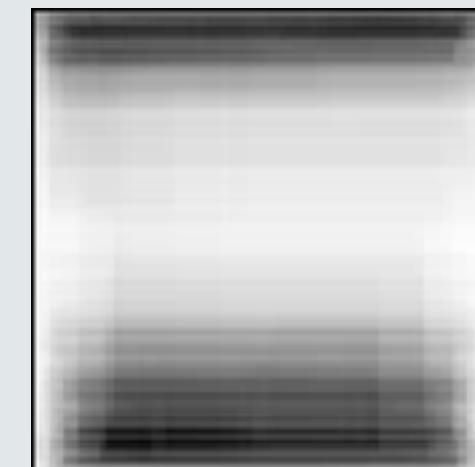
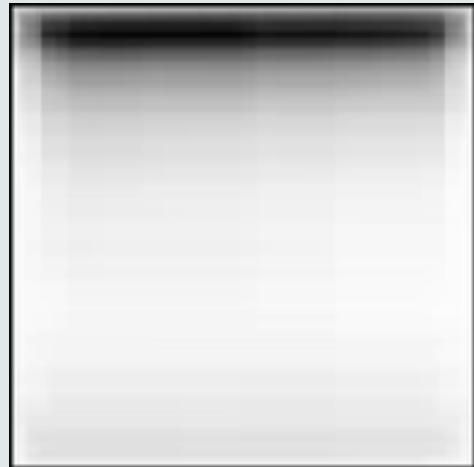
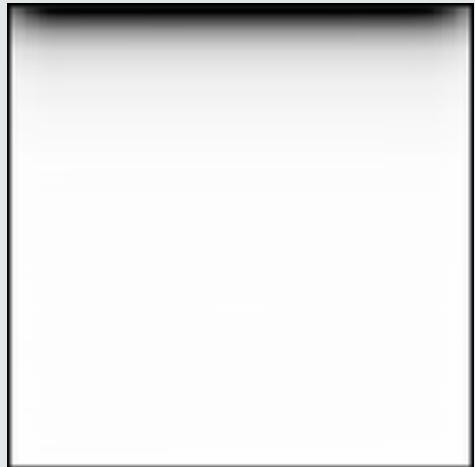
menu



sidebar



twitter



MOBILE

@media

CSS media queries are used to define different style rules for different devices

usually defined on browser width and/or height

can also query device width, height, orientation, resolution, etc.

@media SYNTAX

```
@media not|only mediatype and  
(media feature) {  
    CSS-Code;  
}
```

screen, print, tv
max-width, min-width, width,
max-height, min-height, height,
max-resolution, ...

@media EXAMPLE

```
Hide style sheets from  
older user agents  
@media only screen and  
(max-width: 640px) {  
    body {color: blue; }  
}
```

A red curved arrow points from the text "Hide style sheets from older user agents" down to the opening brace of the CSS code block. Another red curved arrow points from the text "breakpoint" down to the media query condition "(max-width: 640px)".

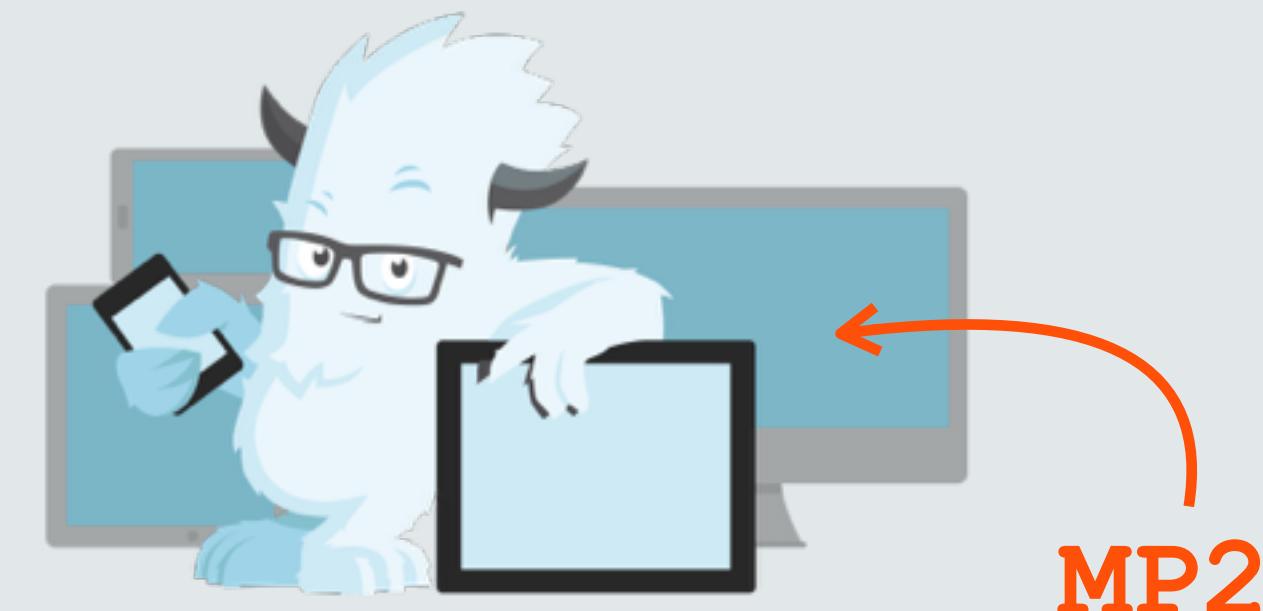
CODEPEN

writing media queries by hand is hard!

CSS FRAMEWORKS



Twitter
Bootstrap



Zurb
Foundation



GRID-BASED DESIGNS

divide screen into equal-width columns

columns separated by gutters

margins on both sides of window

elements may span multiple columns

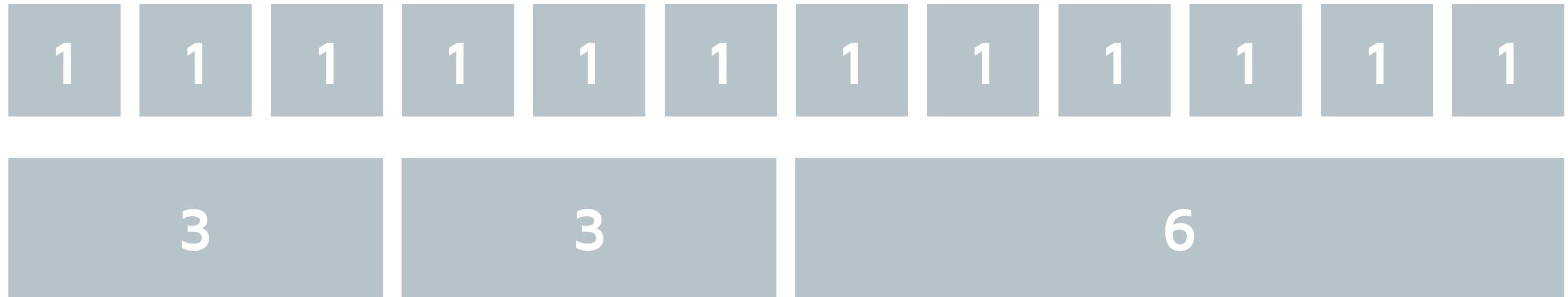
elements start or end at column boundaries

alignment reduces visual clutter



WHY ARE GRIDS USEFUL IN RESPONSIVE DESIGN?

12-COLUMN GRIDS



GRID CLASSES

use predefined class libraries to
create layouts and define
responsive behaviors



GRID CLASSES



```
<div class="row">  
  <div class="small-3 columns">...</div>  
  <div class="small-3 columns">...</div>  
  <div class="small-6 columns">...</div>  
</div>
```

All children of rows
must be columns!



COLUMN CLASSES

small-# **max-width: 640px**

medium-# **min-width: 641px and max-width: 1024px**

large-# **min-width: 1025 and max-width: 1440px**

xlarge-# **min-width: 1441 and max-width: 1920px**

xxlarge-# **min-width: 1920px**



COMBINING CLASSES



```
<div class="row">  
  <div class="small-3 columns">...</div>  
  <div class="small-3 columns">...</div>  
  <div class="small-6 columns">...</div>  
</div>
```

why don't we see any
responsive behavior?



MOBILE FIRST

design mobile first, then desktop

Google's approach, Eric Schmidt 2010

graceful degradation vs progressive enhancement



MOBILE FIRST BEHAVIOR



larger devices will inherit
styles from smaller ones

```
<div class="row">  
  <div class="medium-3 columns">...</div>  
  <div class="medium-3 columns">...</div>  
  <div class="medium-6 columns">...</div>  
</div>
```

CODEPEN



MOBILE FIRST BEHAVIOR

```
<div class="row">  
    <div class="small-3 large-2 columns">...</div>  
    <div class="small-3 large-2 columns">...</div>  
    <div class="small-6 large-8 columns">...</div>  
</div>
```

what does this do?



NESTING GRIDS

```
<div class="row">  
  <div class="small-3 columns">...</div>  
  <div class="small-9 columns">  
    <div class="row">  
      <div class="medium-6 columns">...</div>  
      <div class="medium-6 columns">...</div>  
    </div>  
  </div>  
</div>
```

CODEPEN



MORE GRID CLASSES

```
<div class="row">  
  <div class="small-4 small-offset-2 columns">...</div>  
</div>
```

offset columns

```
<div class="row">  
  <div class="small-3 small-centered columns">...</div>  
</div>
```

center columns

CODEPEN



REORDER COLUMNS

```
<div class="row">  
  <div class="medium-9 medium-push-3 columns">...</div>  
  <div class="medium-3 medium-pull-9 columns">...</div>  
</div>
```

CODEPEN



CSS PREPROCESSOR INTEGRATION

Bootstrap is built on LESS,
but also has SASS port

Foundation works with SASS



CUSTOMIZING WITH SASS

customize grid with SASS variables in `_settings.scss`

```
$row-width: rem-calc(1000);
```

```
$column-gutter: rem-calc(30);
```

```
$total-columns: 12;
```



Device Apple iPhone 5 Network No throttling
320 x 568 2 Fit UA Mozilla/5.0 (iPhone; CPU iPhone O...
0 100 200 300 400 500 600 700 800 900 1000 1100 1200 1300
ANNOUNCEMENT: Enrollment is at max capacity --
keep checking online for openings.
PIAZZA LABS AND MPS RESOURCES
THE ART AND
SCIENCE OF WEB
PROGRAMMING
CS498RK SPRING 2015 UIUC MONDAY AND
WEDNESDAY 11AM-12.15PM SIEBEL 0216
OVERVIEW
Presents client- and server-side technologies that enable modern Web applications. Topics include the building blocks of the Web (HTML, CSS, the Document Object Model, Javascript) and data exchange (HTTP, JSON, RESTful APIs, and SQL/NoSQL databases). Programming
☰

The screenshot shows a mobile web application for a course titled "THE ART AND SCIENCE OF WEB PROGRAMMING". The page includes course details: CS498RK, SPRING 2015 UIUC, MONDAY AND WEDNESDAY 11AM-12.15PM, SIEBEL 0216. A section titled "OVERVIEW" describes the course content. At the top, there is an announcement: "ANNOUNCEMENT: Enrollment is at max capacity -- keep checking online for openings." The interface includes a navigation bar with links for PIAZZA, LABS AND MPS, and RESOURCES. The developer tools are open at the bottom, showing the Elements tab selected. A red circle highlights the "Elements" tab icon. The element tree on the left shows the HTML structure, and the styles panel on the right shows the CSS rules applied to the selected element.

Elements Network Sources Timeline Profiles Resources Audits Console

<!DOCTYPE html>
▼ <html class=" js flexbox flexboxlegacy canvas canvastext webgl touch geolocation postmessage websqldatabase indexeddb hashchange history draganddrop websockets rgba hsla multiplebgs backgroundsize borderimage borderradius boxshadow textshadow opacity cssanimations csscolumns cssgradients cssreflections csstransforms csstransforms3d csstransitions fontface">

Styles Computed Event Listeners DOM Breakpoints Properties

element.style {
}

@media only screen and (max-width: 40em)
.nav {

app.css:2231
app.css:2233

Find in Styles

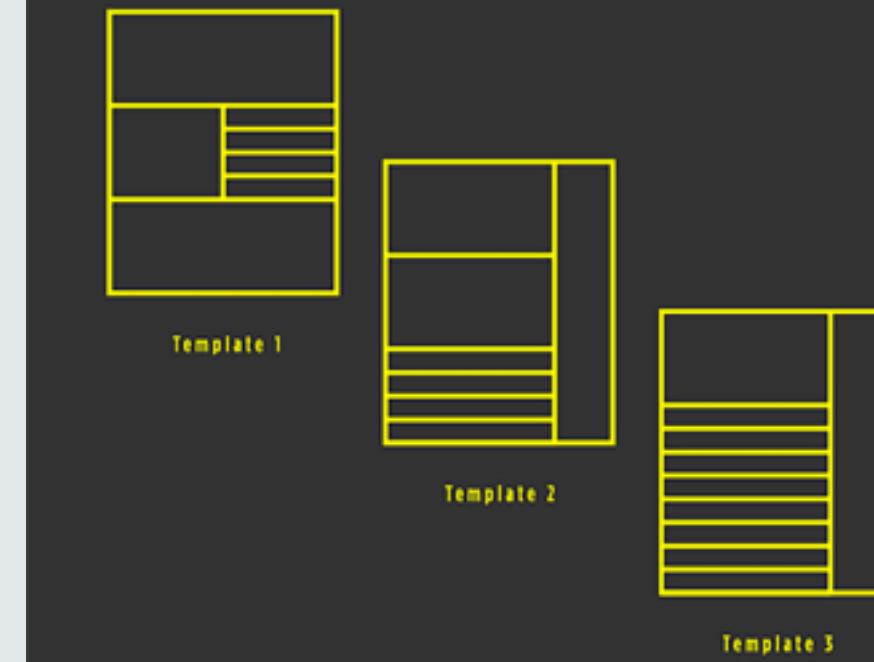
Towards a Responsive Philosophy

www.fastcodesign.com/3036091/the-next-big-thing-in-responsive-design

“Most homepages use the exact same layout, day in and day out. And it’s not just homepages. Article pages—where most users first land on websites—look exactly the same, too. To a user, a day when war breaks out in Iraq can feel exactly the same as a day when the biggest news is a change in Bieber's hairstyle.”



MOST PUBLISHERS ONLINE HAVE THE SAME TEMPLATE EVERY DAY.



have responsive elements instead of responsive pages?

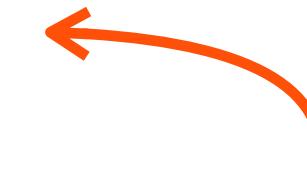
www.fastcodesign.com/3036091/the-next-big-thing-in-responsive-design

TAILORED EXPERIENCES

screen size is just one part of a user's context

there's a lot more you know: where people are in the world, where they've just come from, how long they've been on the site, etc.

design algorithms and interfaces that take into account more of the user's context



percentage of influence specifier?

NEXT CLASS: FOUNDATION LAB

courses.engr.illinois.edu/cs498rk1/