

# HTML



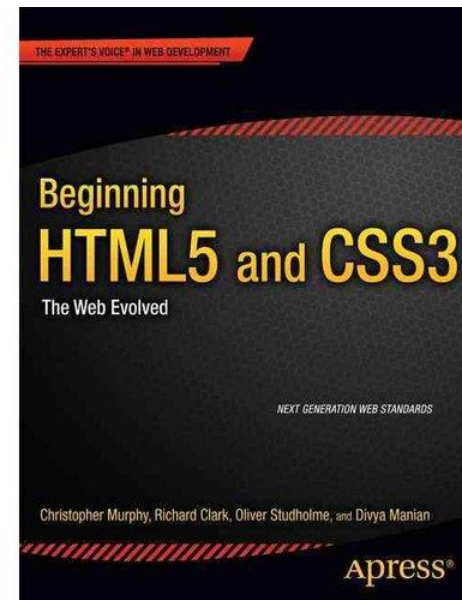
# HTML5 & CSS3

for web designers

Web sites for mobile devices  
& JavaScript use

# Course materials

- Book
  - **Beginning HTML5 and CSS3: The Web Evolved**  
Christopher Murphy; Richard Clark; Oli Studholme;  
Divya Manian  
Apress  
November 21, 2012



# Resources

- <http://github.com/doughoff>



# Introduction

# Web directions

- Web 1.0 – file downloads
- Web 2.0 – dynamic web with data
- Web 3 – page rendered apps on client
- Web 4 – development environment with WebAssembly

# HTML5

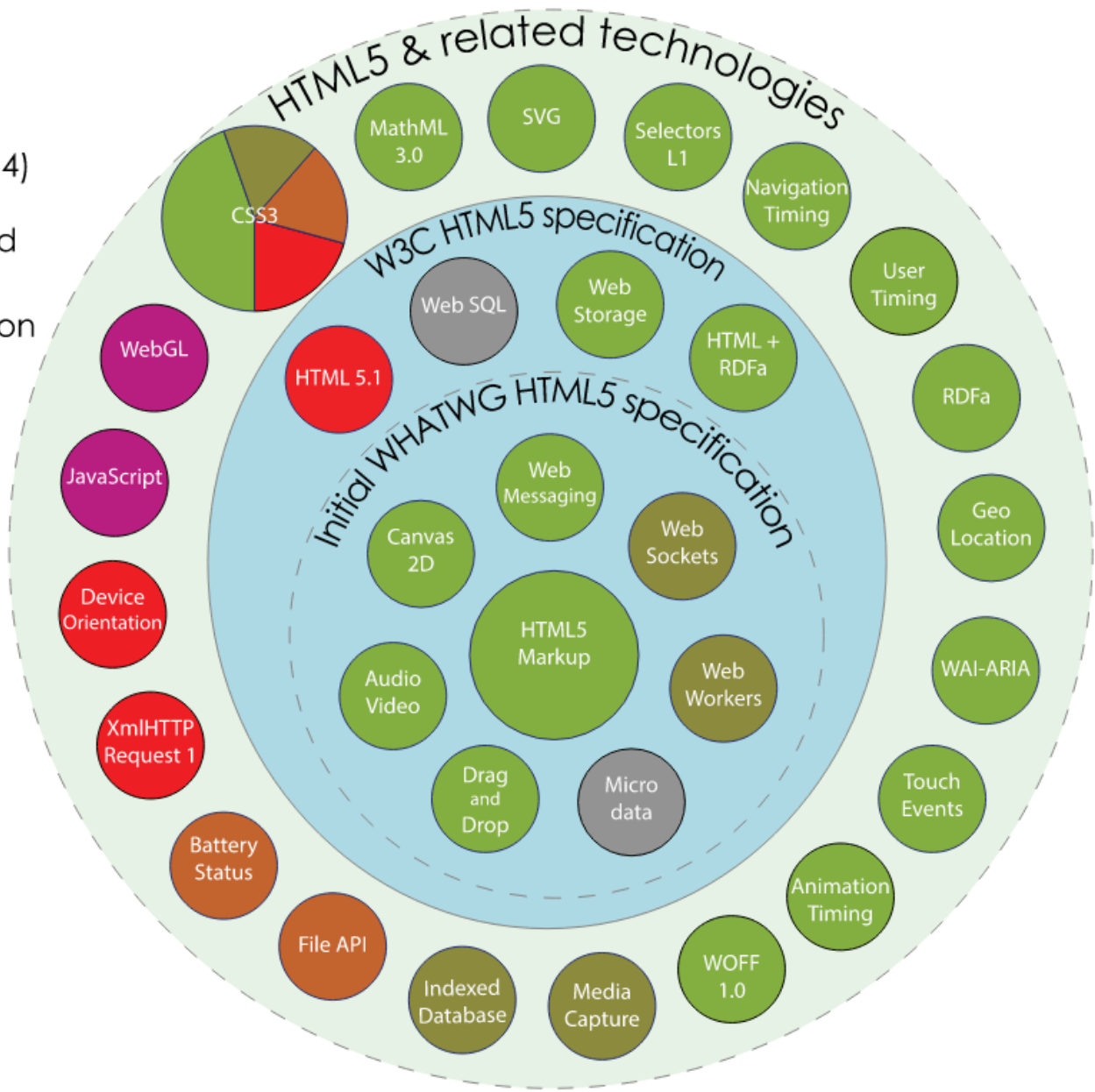
---

- Oct 2014
- Not an HTML4 (2000) upgrade
  - A simplified redesign with acceptance of common usage
- Addresses needs for mobile and integration with JavaScript
- Associated with many standards being agreed upon
  - Drawing
  - Storage
  - Media
  - ...

# HTML5

Taxonomy & Status (October 2014)

- Recommendation/Proposed
- Candidate Recommendation
- Last Call
- Working Draft
- Non-W3C Specifications
- Deprecated or inactive



# Why should I care about HTML5?

- Better ways for web designer to express what they mean. (semantic)
  - Better SEO
- Allows features previously only available through JavaScript
  - Tons of APIs coming usually associated with HTML5



# What is CSS3?

---

- This **is** an upgrade from CSS 2.1
- New features
  - Selectors
  - Web fonts
  - Effects: drop shadows, rounded corners, transparency
  - Multiple backgrounds
  - Animation

# Why should I care about CSS3?

- Cleaner markup
- Fewer images means a little faster loading.
- Eliminating browser specific markup eventually.
  - Specs are not final

# Compatibility

- Browsers

- <http://caniuse.com/>

- <http://HTML5test.com>

compatibility for current browser

- FindMeByIP

<http://fmbip.com> browser compatibility,  
better for developers &  
<http://findmebyip.com/litmus/> all browsers

## HTML5

- ◊ Audio element
- ◊ Canvas (basic support)
- ◊ Color input type
- ◊ contenteditable attribute (basic support)
- ◊ Datalist element
- ◊ dataset & data-\* attributes
- ◊ Date/time input types
- ◊ Details & Summary elements
- ◊ Drag and Drop
- ◊ Form validation
- ◊ HTML5 form features
- ◊ input placeholder attribute
- ◊ New semantic elements
- ◊ Number input type
- ◊ Offline web applications
- ◊ Progress & Meter
- ◊ Range input type
- ◊ Ruby annotation
- ◊ sandbox attribute for iframes
- ◊ Session history management
- ◊ Text API for Canvas
- ◊ Toolbar/context menu
- ◊ Video element
- ◊ WebGL - 3D Canvas graphics
- ◊ **All HTML5 features**

# HTML5 – compatibility web sites

- Mobile
  - <http://mobilehtml5.org/> - HTML5 compatibility charts on mobile and tablet browsers

# Browser testing

- The CSS3 Test
  - <http://css3test.com/>
- CSS3.info selectors test
  - <http://www.css3.info/selectors-test/>
- modern.IE
  - <http://www.modern.ie/en-us>
- Browserling
  - <https://browserling.com/>

# HTML5 info

- W3C HTML WG editors' drafts - <http://dev.w3.org/html5/>
- HTML 5.1 - <http://www.w3.org/TR/2012/WD-html51-20121217/>
- <http://www.html5rocks.com/> - great interactive slide presentation, interactive code, samples and tutorials
- <http://diveintohtml5.info/> - Mark Pilgrim's online book
- <http://html5forwebdesigners.com/> - Jeremy Keith's online book for designers
- <http://html5please.us/> - recommendations to use features

# HTML / CSS Info

- Adobe's The Expressive Web
  - <http://beta.theexpressiveweb.com>
- CSS Tricks
  - <http://css-tricks.com/>
- CSS3.info
  - <http://www.css3.info/>
- Simpl.info - examples of new HTML, CSS, & JavaScript
  - <http://simpl.info/>
- HTML5 Weekly - searchable newsletter
  - <http://html5weekly.com>

# Exercise

- Follow instructions in Exercises handout for
  - 1. Set up development environment





# Markup and Structure

# A demo site

1. Native video
2. @font-face
3. CSS3 hover/transform
4. Geolocation & Local Storage
5. Form validation
6. Drag & drop

<http://thehtml5herald.com/>



# The doctype

- Document Type Declaration tells the browser what's coming if it cares.
- **Before:**
  - `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">`
- **After:**
  - `<!doctype html>`

# <html>

- lang = "en" (optional)
- no XML namespace



# <head>

- Character encoding and CSS link

- **Before:**

- `<meta http-equiv="Content-Type" content="text/html; charset=utf-8">`
- `<link rel="stylesheet" href="css/styles.css" type="text/css">`

- **After:**

- `<meta charset="utf-8">`
- `<link rel="stylesheet" href="css/styles.css">`



# Metadata for SEO

- Original metadata was `<meta>` tags
- Spammers misused keywords tags
- Died in 2009
  - Google stopped using tags as a ranking signal

“Google has ignored the keywords meta tag for years and currently we see no need to change that policy.”  
--Matt Cutts, Google programmer

# charset

- best is UTF-8, database must match
- old charsets
  - windows-1252 includes ISO 8859-1
    - includes curly quotes
    - replaces control characters with displayable ones 80-9F
  - iso-8859-1 (Latin1) includes ASCII
    - eight bits
    - still almost 10% use
  - ASCII
    - seven bits

to receive future event informa  
subject  Not Interested .

from Sept. 11th eve  
of Kansas City's top  
to find out about hot st

## Manage your EasyPay account

Login and go to Edit Easy Pay Options from  
your Overview screen.

Verified email@doughoff.com

Secret Question: What is your maternal grandmother's maiden name?

Answer:

New Password:



# IE issues

- Older browsers ignore non-HTML tag names and style using CSS except <IE9
- The HTML shiv for IE.
  - `<!--[if lt IE 9]>...`
- Or use Modernizr (better)
  - <https://modernizr.com>
  - Includes the IE shiv and polyfills you specify
  - features exposed through html element class names and JavaScript queries



# Valid HTML5

---

- Looser but still rules to follow
- Not required
  - case sensitivity
  - attribute quotes!
  - empty element slash!
  - an action for a form
  - & can now be &

# A basic HTML5 template

---

- `<!doctype html>`
- `<html lang="en">`
- `<head>`
  - `<meta charset="utf-8">`
  - `<title>The HTML5 Herald</title>`
  - `<meta name="description" content="The HTML5 Herald">`
  - `<meta name="author" content="SitePoint">`



# A basic HTML5 template

- `<link rel="stylesheet" href="css/site-name .css">`
- `<!--[if lt IE 9]><script  
src="http://html5shiv.googlecode.com/svn/trunk/html5.js"><  
/script><![endif]-->`
- `</head>`
- `<body>`
  - **content here**
  - `<script src="js/scripts.js"></script>`
- `</body>`
- `</html>`

# Exercise

- Follow instructions in Exercises handout for
  - 2. Basic HTML5 template



Chapter 3

# Markup and Semantics

# Redefining the page's structure

---

- semantics = the meaning of the content
- **<main>**
  - 5.1 - content directly related to the central topic of a document. Once per page. Top level under body. (not for IE)
- **<header>**
  - repeated navigational aids and logo at top of page or each chunk on page
- **<footer>**
  - secondary info to place "out of the way", not just bottom

# Redefining the page's structure

- **<section>**
  - a related group of content with a heading
  - use <div> if no meaning is inferred
  - nesting is OK
- **<article>**
  - intended for publication – RSS, post, blog, comment, tweet
- **<aside>**
  - related but separate information
  - sidebars, link lists, advertising

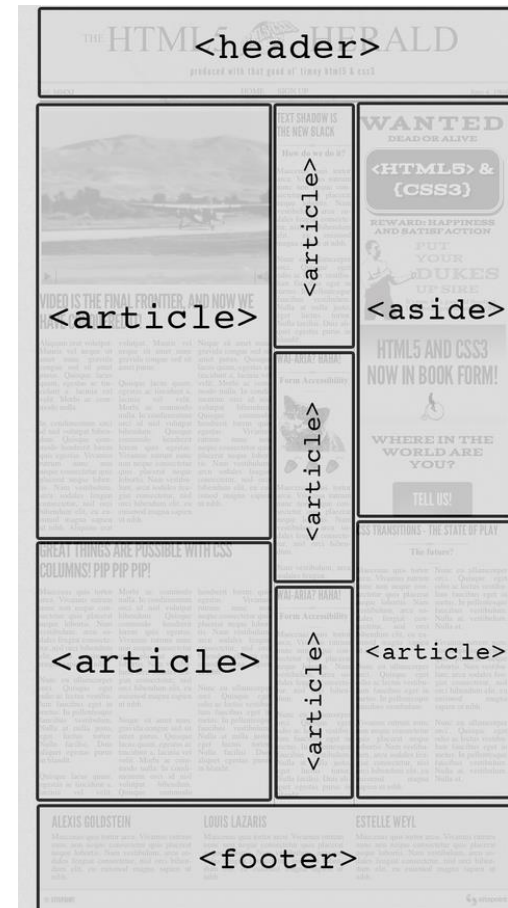
# Redefining the page's structure

- `<nav>`
  - important navigation links



# Structuring a page

- header, nav, column divs, article, aside, footer
  - **header** for repeatable content at top of page with nav
  - **nav** for repeatable navigation area
  - **divs** for major columns in body
  - **articles** for text units
  - **aside** for related articles
  - **footer** with sections and sub-sections



# Exercise

- Follow instructions in Exercises handout for
  - 3. Show structure of HTML5Herald

# Changes to existing features

---

- Visual vs. content
  - `<b>` is just styled different, but `<strong>` has strong importance
  - `<i>` is also styled differently, but `<em>` has emphasized meaning

# Changes to existing features

- Previous meaning (visual)
  - `<big>` - a big font
  - `<small>` - a small font
- New meaning (content)
  - Don't use `<big>`
  - `<small>` - "the fine print"

# Changes to existing features

- `<dl>` (definition list), `<dt>` (term), and `<dd>` (description) are now **description** list elements
  - Term-definition
  - Topic-value
  - Question-answer

# Changes to existing features

- `<cite>` “forbids” people’s names and only can be used for citing titles of works.
- `<ol>` now supports **start** attribute for a starting number (again).
- `<ol>` adds **reversed** attribute to reverse order, future implementation

# Changes to existing features

- Block elements are now valid inside an `<a>`
  - bigger targets for mobile

# New elements

- `<q>` is just like `<blockquote>`
  - use for quoted text
- `<abbr title="laugh out loud">lol</abbr>`
  - Tooltips
- `<menu type="toolbar">`
  - `<li class="new">New</li>`
  - `<li class="open">Open</li>`
  - `<li class="save">Save</li>`
  - `<li class="quit">Quit</li>`
- `</menu>`



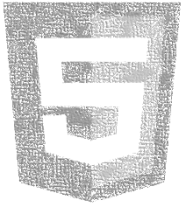
# New elements - figure

- `<figure id="fig1">`
  - ``
  - `<figcaption>Screen Reader Support for WAI-ARIA</figcaption>`
- `</figure>`
- Optional secondary content for text, images, code, etc.

# New elements - picture

- Will make the <img> adaptive to screen size capabilities.
- <http://arstechnica.com/information-technology/2014/09/how-a-new-html-element-will-make-the-web-faster/>

IE	Edge *	Firefox	Chrome	Safari
			29	
			49	
			50	
8	13	47	51	
11	14	48	52	9.1
		49	53	10
		50	54	TP
		51	55	



# Exercise

- Follow instructions in Exercises handout for
  - 4. Associate a caption with a photo



Better markup than HTML5 elements for business data

**Microdata**

# Microdata

- semantic markup for specific data types by consenting authors and readers
- XML in thinking, HTML5 in practice
  - uses a custom vocabulary of attributes
  - no new elements

# History

---

- Google Rich Snippets - basic (2009)
- Schema - better (June 2011)
  - <http://www.schema.org>
  - Supported by Bing, Google, Yahoo!

# Schemas

- Common (hundreds more)
  - [Review](#), [AggregateRating](#)
  - [Event](#)
  - [Organization](#)
  - [Place](#), [LocalBusiness](#), [Restaurant](#) ...
  - [Person](#)
  - [Product](#), [Offer](#), [AggregateOffer](#)
  - Creative works: [CreativeWork](#), [Book](#), [Movie](#), [MusicRecording](#), [Recipe](#), [TVSeries](#)
  - Embedded non-text objects:  
[AudioObject](#), [ImageObject](#), [VideoObject](#)

# Declaration

- declare scope
  - `<element itemscope>`
    - defines the beginning of the microdata scope
    - one vocabulary in use in one scope
- declare namespace
  - `<element itemtype="http://www.data-vocabulary.org/Person">`
- declare data
  - `<element itemprop="name">value</element>`



# Declaration

- `<div itemscope itemtype="http://schema.org/Event">`
- `<a itemprop="url" href="nba-miami-philidelphia-game3.html">`
- NBA Eastern Conference First Round Playoff Tickets:
- `<span itemprop="name"> Miami Heat at Philadelphia 76ers - Game 3 (Home Game 1) </span></a><meta itemprop="startDate" content="2016-04-21T20:00">Thu, 04/21/16 8:00 p.m.`
- ...
- `</div>`

# Testing

- Use Google's Rich Snippets testing tool
  - shows data parsed from page
  - <http://www.google.com/webmasters/tools/richsnippets>
- Markup helper
  - <https://www.google.com/webmasters/markup-helper/u/0/>



# Markup for Programmers

# <script>

- Before:

- `<script src="js/scripts.js" type="text/javascript"></script>`

- After:

- `<script src="js/scripts.js"></script>`

- Location

- anywhere on page
- if doing DOM manipulation, it must follow the elements it references
- most recommended: place at bottom of body
- most common: use in jQuery ready function at top of page

# <script>

- <script>
  - code is fetched and executed immediately
- <script defer>
  - for external scripts , execution after parsing when DOM is loaded
  - Buggy support in IE8
- <script async>
  - HTML5 - for external scripts, will run when available without waiting, no particular order
  - no support IE9

# Programming aids

---

- Many elements are used with JavaScript to prepare a place which has meaning for the data.
- Most support is semantic and not view oriented

## **<mark>**

- In response to a user's actions, this element captures a location on the page.
  - Search results could be delimited with this element

# <progress> and <meter>

- Marking data that is changing
  - **progress** indicates what we've done so far without a goal.
    - attributes: max="" value=""
  - **meter** indicates how far we are towards a goal.
    - attributes: max="" value="" min="" high="", low="", optimum=""



# <time> and <data>

- To designate a precise date & time
- To create a placeholder for a formatted date & time
- To mark a publication date of this page or closest <article>
- [ISO 8601](#) format
  - <time>1904-06-04</time>
  - <time datetime="1904-06-04">the fourth of June, 1904 </time>
  - <time datetime="1904-06-04T16:24:34Z" pubdate>June 4, 1904 </time>
- 10/2011 – replaced by <data> element and then reinstated

# data-\* custom attributes

- for custom data private to the page when there are no more appropriate attributes or elements
- data attribute key (name)
  - must be at least one character long
  - must be prefixed with 'data-'.
  - no uppercase letters
- data attribute value can be any string
- declarative code like
  - jQuery's programmatic data( )
  - jQuery Mobile markup



Chapter 4

# Form elements

# HTML5 form attributes

---

- `<input placeholder="type stuff like this">`
  - My website is located at:
- `<input multiple>`
  - `<select>` more than one `<option>` in a drop down, now for email and file inputs.



# jQuery plugin

- float label pattern
  - Flylabel.js (above)
    - <http://athaeryn.github.io/flyLabel.js>
  - Floatlabels.js (inside)
    - <http://clubdesign.github.io/floatlabels.js/>
  - JvFloat.js
    - <http://bootsnipp.com/iframe/Ov>

Destroy all humans?

DESTROY ALL HUMANS?

Not today.

This is a floated Label. It stays there until the field is cleared.

Text text text

Name

My name

Email

# HTML5 form attributes

- `<input disabled>` - used to disable buttons
  - Use with `:disabled` in CSS selector
  - Value is not submitted, no focus
- `<input readonly>`
  - Value is submitted & can have focus

# HTML5 form attributes

- `<input form="form1 form2 form3">` - shows membership
  - lets you work around lack of nested form support
- `<input autofocus>` - give focus on page load
- IE9 lack of support, better to use jQuery

# HTML5 form attributes

- **<input autocomplete>** - like Google's search, supported but now in specs.
  - Default is on (if user has allowed in browser)
- **<form autocomplete = 'OFF'**
  - automatically complete values based on values that the user has entered before

User ID:

[Forgot UserID or Password?](#)

Password:

*This webpage has disabled automatic filling for this form.*



# REST & multiple form support

- Future support possible for actions of REST support
  - method="delete", method = "put" (update)
- Example of multiple form use
  - `<form method = 'get' id = 'get' action = '/thing'>...</form>`
  - `<form method = 'post' id = 'post' action = 'thing'>`
    - `<input type = 'button' value = 'delay 1 day'`  
`formaction = '/thing/delay1'`
  - `</form>`
  - `<input type = 'hidden' value = '123' name = 'id' form = 'get post' />`

# datalist ~= autocomplete

- `<label for="favcolor">Favorite Color</label>`
- `<input type="text" list="colors" id="tb_color" name="req_colors" >`
- `<datalist id="colors">`  
    `<option value="Blue">`  
    `<option value="Green">`  
    `<option value="Pink">`  
    `<option value="Purple">`  
    `</datalist>`
- Not supported in Safari, IE9 - use jQuery autocomplete

# HTML5 form validation

- **<input required ...>** must have a value which is valid for the data type
  - Failure when form is submitted will put focus on first invalid input element
  - Chrome will provide error messages

• My name is:

• My email address is:

• My website is located at:

• I would like my password to be:

**Please fill out this field.**

# HTML5 form validation

- Data specific types for `<input type="datatype">`
  - search, email, url, tel, datetime, date, month, week, time, datetime-local, number, range, color
- type="search" – appearance like the o/s search style
- type="email" – one or more (use @ multiple), better error messages, hints to iOS to show better keypad
- type="url" – browser does basic validation

IE	Firefox	Chrome	Safari	Opera
8.0				
9.0	23.0		5.1	
10.0	24.0	29.0	6.0	
11.0	25.0	30.0	7.0	17.0
	26.0	31.0		18.0
	27.0	32.0		

# HTML5 form validation

- type="tel" – no validation
- type="range" min="0" max="10"
- type="color">
  - mostly mobile support now
- Many unsupported

On a scale of 1 to 10, my knowledge of HTML5 is:



# HTML5 form validation

- Type="number"
  - causes keyboard to show numbers on iOS
  - causes decimal numbers to be invalid
    - use step = "any" to fix
  - use min="0" for positive numbers only
  - causes spinner to crowd input field on Android
    - remove with `input[type=number]::-webkit-inner-spin-button, input[type=number]::-webkit-outer-spin-button { -webkit-appearance: none; margin: 0; }`

I would like to receive  copies of *The HTML5 Herald*.

# HTML5 form validation

- type="date" – year-month-day: 2004-06-24
- type="month" – year-month: 2004-06
- type="week" – year-weekNumber: 2004-W09
- type="time" – time 24h clock: 22:00
- type="datetime" – 2004-06-24T22:00
- type="datetime-local" – no time zone

Please start my subscription on:

- Support only in Chrome

# HTML5 form validation

- **<input pattern="<regex pattern>"**
  - Common regex patterns – <http://regexlib.com/>
    - \S{5} – exactly five non-whitespace characters
    - \d{5} – exactly five digits
    - \d{5}(-\d{4})? – exactly five digits optionally followed by a dash and four digits
    - \d{5,10} – between five and ten digits
- **<output>**
  - programming support to show calculation results



# Changes to existing form controls & attributes

---



- **<form novalidate>**
  - stops HTML5 validation attributes
- **<form action="">**, **<form>**
  - requests current page, now in specs
- **<optgroup>** - can be nested
- **<textarea>**
  - no rows="#" or cols="#". Use CSS.
  - wrap="soft | hard" – hard adds CR from word wrap and requires cols="#"

# Exercise

- Follow instructions in Exercises handout for
  - 5. Form
  - 6. Find browser support for input element
  - 7. Form field testing on mobile



# HTML5 Audio and Video

# A bit of history

---

- Required third party player
  - QuickTime, RealPlayer, Silverlight, Flash
- Required special tags
- Required special formats

# Current state

- Browsers must
  - Deconstruct the container
  - Decode the video to display it
  - Decode the audio to play it
- IE 8 does not support `<video>` but still has market share

# Video containers / codecs

---

- A **container** holds the video and audio streams
  - .flv (Flash), .mp4, .m4v (MPEG-4), .avi (AVI)
  - HTML5 relevant containers are MPEG-4, OGG, WebM
- A **codec** is the engine that encodes or decodes the video stream
  - HTML5 relevant video codecs are H.264, Theora, and VP8
  - HTML5 relevant audio codecs are AAC and Vorbis



# Video support in browsers

CODECS/ CONTAINER	Firefox	Safari	IE	Chrome	Opera	iPhone	Android
<b>H.264+AAC +MP4</b>	.	3.0+	9.0+	.	.	3.0+	2.0+
<b>Theora+Vorbis+ Ogg</b>	3.5+	†	.	5.0+	10.5+	.	.
<b>WebM</b>	4.0+	†	9.0 +*	6.0+	10.6+	.	2.3‡

when the user has installed a VP8  
codec

# DiveIntoHTML5 recommendation

---

- <http://diveintohtml5.info/video.html>
- Make three versions
  - H.264 baseline video and AAC "low complexity" audio in an MP4 container.
  - Theora video and Vorbis audio in an Ogg container.
  - WebM (VP8 + Vorbis).
- Link to all three video files from a single `<video>` element, and fall back to a Flash-based video player.



# My recommendation

- Use YouTube or Vimeo Plus

Like About **Share** Add to

Share this video **Embed** Email







```
<iframe width="420" height="315"
src="//www.youtube.com/embed/t1I022aUWQQ" frameborder="0"
allowfullscreen></iframe>
```

Video size: 420 × 315

- ☒ Show suggested videos when the video finishes
- ☐ Enable privacy-enhanced mode [?]
- ☐ Use old embed code [?]

Share This Video

Link <http://vimeo.com/67809013>

Social      

Add E-mail

Embed [+ Show options](#)

```
<iframe src="//player.vimeo.com/video/67809013" width="500"
```

This embedded video will include a text link.

# The markup

---

- Simplest - `<video src="example.webm"></video>`
- Better - `<video src="example.webm" width="375" height="280"></video>`
- Add controls - `<video src="example.webm" width="375" height="280" controls></video>`



# The markup - attributes

- `<video autoplay>` - not polite unless requested
- `<video loop>` - infinite repeat
- `<video preload = "auto | none | metadata">` - auto loads both video and metadata before playing.
- `<video poster = "image.jpg">` - "splash screen"
- `<video audio = "muted">` - no other options

# The real markup

- `<video id="movie" width="320" height="240" preload controls>`
  - `<source src="pr6.mp4" />`
  - `<source src="pr6.webm" type='video/webm; codecs="vp8, vorbis"' />`
  - `<source src="pr6.ogv" type='video/ogg; codecs="theora, vorbis"' />`
  - `<object width="320" height="240" type="application/x-shockwave-flash" data="flowplayer-3.2.1.swf"> <param name="movie" value="flowplayer-3.2.1.swf" /> <param name="allowfullscreen" value="true" /> <param name="flashvars" value='config={"clip":{"url": "http://wearehugh.com/dih5/pr6.mp4", "autoPlay":false, "autoBuffering":true}}' /> <p>Download video as <a href="pr6.mp4">MP4</a>, <a href="pr6.webm">WebM</a>, or <a href="pr6.ogv">Ogg</a>.</p> </object>`
- `</video>`

# Miro VC video encoder

---

- Download [Miro Video Converter](#) for Mac OS X or Windows.
  - open source, GPL-licensed program for encoding video in multiple formats.
  - offers no options beyond choosing a video file and choosing an output format
  - can take virtually any video file as input, including DV video produced by consumer-level camcorders
  - produces reasonable quality output from most videos

# Audio player

- `<audio controls="controls">`
  - `<source src="song.ogg" type="audio/ogg" />`
  - `<source src="song.mp3" type="audio/mpeg" />`
  - Your browser does not support the audio element.
- `</audio>`

Format	IE 9	Firefox 3.5	Opera 10.5	Chrome 3.0	Safari 3.0
Ogg Vorbis	No	Yes	Yes	Yes	No
MP3	Yes	No	No	Yes	Yes
Wav	No	Yes	Yes	Yes	Yes

# Web Audio API

- JavaScript
- Web Audio API - Working Draft
  - no support in IE, Opera
- [https://developer.mozilla.org/en-US/docs/Web\\_Audio\\_API](https://developer.mozilla.org/en-US/docs/Web_Audio_API)
- Links
  - Beat box -
    - [http://box.cloudymedia.com/?utm\\_source=html5weekly&utm\\_medium=email#!song-1544](http://box.cloudymedia.com/?utm_source=html5weekly&utm_medium=email#!song-1544)
    - <http://box.cloudymedia.com/#!song-589>

# Video plugins

---

- Background videos
  - [jQuery.VideoBG plugin](#)
- Good looking video player
  - [ttwVideoPlayer jQuery Plugin](#)
- All in one solution
  - [jPlayer](#)
- VideoJS - <http://videojs.com/>



# Web sites – audio & video

- **Simon Pieters**

- <http://dev.opera.com/articles/view/everything-you-need-to-know-about-html5-video-and-audio/>

- **Kaltura community**

- [Html5video.org](http://Html5video.org)

# Exercise

- Follow instructions in Exercises handout for
  - 7. Test video formats
  - 8. Test audio formats (on system with speakers)

# HTML 5.1

- Dec 2016
- <https://www.w3.org/TR/2016/REC-html51-20161101/changes.html#changes>
- picture and srcset attributes
- details and summary elements



# CSS3 Introduction

# Intro

---

- CSS3 spec has been around since 2006
- Few CSS3 modules have reached Recommendation status (approved).
  - CSS3 Color
  - CSS2.1
- Prediction for final approval is 2022.
- CSS Zen Garden - <http://www.csszengarden.com/>

# Vendor extension prefixes

- Used to implement and test vendor specific extensions to CSS
- **Chrome, Safari:** -webkit-
- **Firefox:** -moz-
- **Opera:** -o-
- **IE:** -ms-
- **Microsoft Office:** -mso-

# Initialize your CSS

- Set all values to zero, make HTML5 new elements block
- <http://www.cssreset.com/>
  - Recommended for HTML5 use  
<http://html5doctor.com/html-5-reset-stylesheet/>
  - Full featured CSS framework use YUI 3  
<http://yuilibrary.com/>
  - Smallest file size  
normalize.css

# HTML5Boilerplate

- IE fixes, mobile setup, favicons, touch icons
- Uses Modernizr
  - contains HTML5 shiv, respond.js
- Uses jQuery, Chrome Frame (IE6), Google Analytics
- Uses normalize.css to initialize CSS
  - Contains CSS defaults and helpers, media query placeholders, print styles with page breaks



# Migration - Modernizr

- JavaScript library used to test for HTML5 features
- 2.0 version – June 1, 2011
- added to **HTML5 Boilerplate 2.0**
- adds class names to all elements to show or not show support for behavior.
  - ... localStorage sessionStorage svg no-inlinesvg ...



# Migration - Modernizr

- create CSS styles based on support
- use in JavaScript as a boolean property
  - if (Modernizr. geolocation) {...}
- eliminates need for HTML5 shiv when using semantic elements
- uses a reset CSS

# Migration – Modernizr web sites

- <http://www.modernizr.com>
- HTML5 Cross Browser Polyfills - <https://github.com/Modernizr/Modernizr/wiki/HTML5-Cross-browser-Polyfills>
- Initializr 2 - <http://www.initializr.com/> for more template options
- Selectivizr – emulates CSS3 selectors in IE6-8 <http://selectivizr.com/>

# Exercise

- Follow instructions in Exercises handout for
  - Run CSS3 browser score test



**CSS selectors**

# Major selector types

CSS	HTML
tagname	<code>&lt;tagname&gt;</code> <code>&lt;tagname&gt;</code>
<code>.className</code>	<code>&lt;tagname class="className"&gt;</code> <code>&lt;otherTagname class="className"&gt;</code>
<code>#idName</code>	<code>&lt;tagname id="idName"&gt;</code> (can only be one id per doc)
<code>[attributeName="attributeValue"]</code>	<code>&lt;tagname</code> <code>attributeName=attributeValue&gt;</code>

# CSS2 Selector operators

---

- union selector1, selector2
  - combination of the two selections
  - h1, h2, h3 { }
- descendant: selector1 selector2
  - all elements at any child sublevel of selector1
  - ul.menu li
- child: selector1 > selector2
  - all elements at first child sublevel only
- adjacent / sibling: selector1 + selector2
  - first element at the same peer level coming after selector1

# Relational selectors – CSS3

- general sibling: `selector1 ~ selector2`
  - all elements at the same peer level coming after selector1
  - a little buggy in IE 7+



# CSS2 Attribute selectors

---

- `[attributeName=attributeValue]`
- All value matching is case insensitive
- No quotes are used for the attribute value
- Compatibility
  - IE6 and lower do not support attribute selection
  - IE7 and IE8 support attribute selectors only if a !DOCTYPE is specified.

# Attribute selectors – CSS2

- `[attributeName]` – all tags with this attribute
- `tag[attributeName]` – all tags with this tag name and attribute having any value
- `tag[attributeName=value]` – all tags with this tag name and attribute matching this value
- `tag[attributeName|=value]` – all tags with this tag name and attribute matching or beginning with this value
- `tag[attributeName~=value]` – all tags with this tag name and attribute containing a word with this value

# Attribute selectors – CSS3

- `tag[attributeName^=value]` – all tags with this tag name and attribute beginning with this value
- `tag[attributeName$=value]` – all tags with this tag name and attribute ending with this value
- `tag[attributeName*=value]` – all tags with this tag name and attribute containing any portion with this value

# CSS3 selectors – pseudo classes

- LVHA - a:link, a:visited, a:hover, a:active (love-hate)
- Add :focus when using :hover (mobile)
  - a:focus:hover
- New pseudo-classes (states of elements)
  - :enabled, :disabled
  - :checked, :indeterminate
  - :target (http://domain.com#item1 matches button#item1:target or #item1:target)
  - :default
  - :valid, :invalid

# CSS3 selectors – pseudo classes

- Firefox already has :invalid red shadow default style. Watch.

Email Address: \*

*johndoe@example.com*

# CSS3 selectors – pseudo classes

- New pseudo-classes
  - :in-range, :out-of-range (for min and max)
  - :required, :optional
  - :read-only, :read-write

# CSS3 selectors – pseudo classes

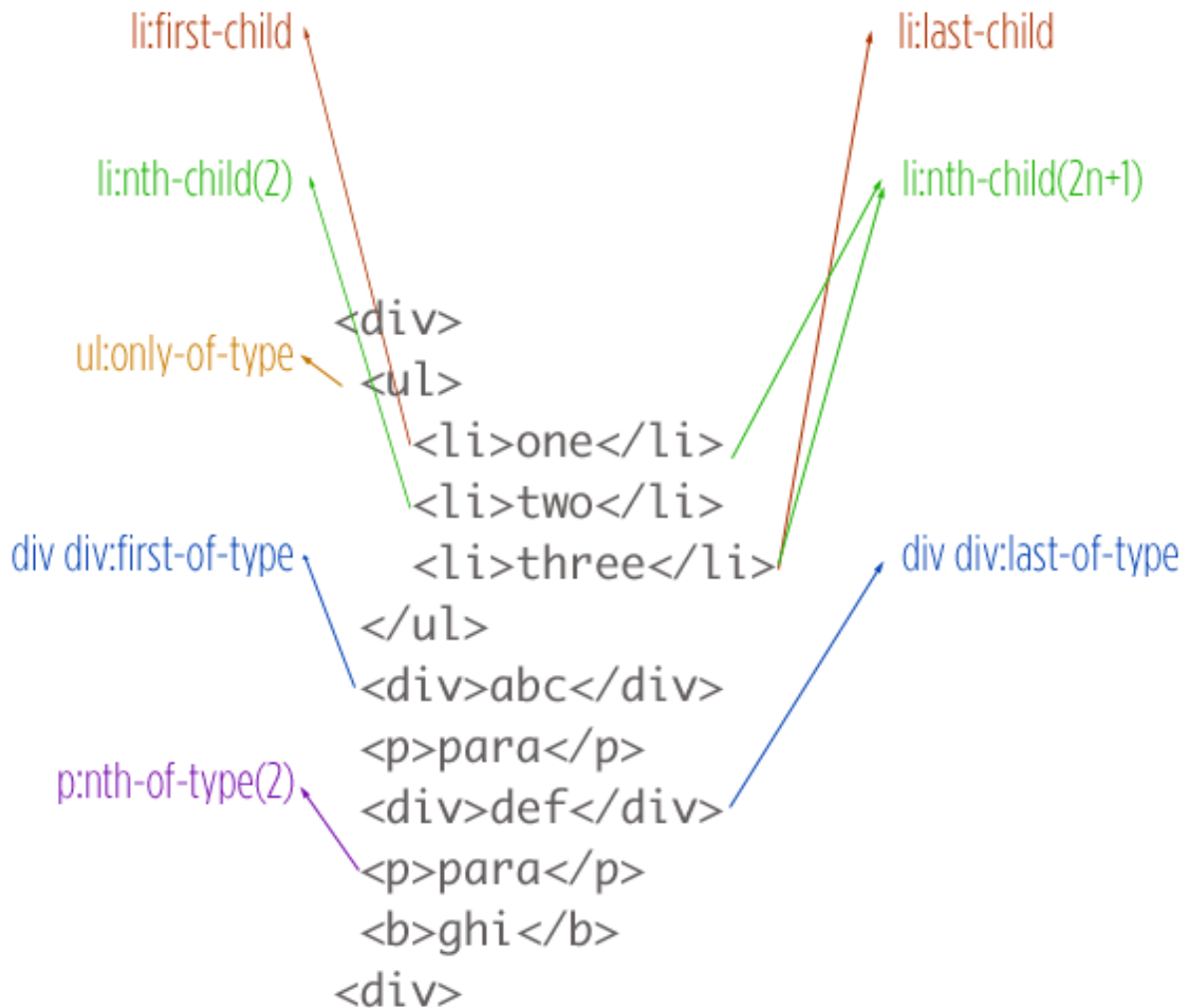
- Structural
  - :root
  - element1:nth-of-type(#) // also use odd & even  
element1:nth-last-of-type(#)
  - element1 element2:nth-child(#)  
element1 element2:nth-last-child(#)
  - element1:first-of-type  
element1:last-of-type  
element1:first-child  
element1:last-child

# CSS3 selectors – pseudo classes

- Structural
  - `element1:only-child`  
`element1:only-of-type`
  - `element1:empty`
  - `element1:lang(en)`
  - `element1:not(selector)`



# CSS3 selectors – pseudo classes



# CSS3 selectors – pseudo classes

- implicit structure in element
  - :first-letter – useful for drop caps
  - :first-line – useful for magazine style lead-in
  - :lang (IE8+)
- selected text
  - ::selection (also use ::-moz-selection for Firefox)

# Pseudo-elements

- generated content (pseudo element)
  - :before { content: "..."} }
  - :after {content: "..."} }
- can be any combination of: "text" or attr(key)
- <=IE8
  - supports the single-colon CSS 2.1 syntax (i.e. pseudo-class)
  - does not support the double-colon CSS3 syntax (i.e. pseudo-element)

# Exercise

- Follow instructions in Exercises handout for
  - Create a style with :after
  - Change the selection style
  - Alternate by 2s and 3s in bullet lists



## **CSS3 colors & effects**

# CSS Colors

- #3e3e3e (RRGGBB)
  - #213
- rgb(40, 40, 40)
- hsl(0, 0%, 25%)
- grey

# Colors & opacity

- RGB is now RGBA (alpha/opacity)
  - no hex version (#RGB, #RRGGBB)
  - <=IE8 will ignore and use last color it knows
- HSL
  - hue – 0 to 360, starting at red
  - saturation – 0% - 100%, starting at no gray
  - lightness – 0% - 100%, starting at black, 50% is normal
- HSLA
  - alpha – 0% - 100%, starting at transparent

0° Reds					
Saturation					
	100%	75%	50%	25%	0%
100					
88					
75					
63					
50					
38					
25					
13					
0					

# Colors



- `<div style="background: rgba(255, 0, 0, 0.2);"></div>`  
`<div style="background: rgba(255, 0, 0, 0.4);"></div>`  
`<div style="background: rgba(255, 0, 0, 0.6);"></div>`  
`<div style="background: rgba(255, 0, 0, 0.8);"></div>`  
`<div style="background: rgba(255, 0, 0, 1) ;"></div>`



- `<div style="background: rgb(243, 191, 189);"></div>`  
`<div style="background: rgb(246, 143, 142);"></div>`  
`<div style="background: rgb(249, 95 , 94) ;"></div>`  
`<div style="background: rgb(252, 47, 47) ;"></div>`  
`<div style="background: rgb(255, 0, 0) ;"></div>`



# Colors - names

- Basic - aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow
- Extended - antiquewhite, aqua, aquamarine, azure, beige, bisque, black, blanchedalmond, blue, blueviolet, brown, burlywood, cadetblue, chartreuse, chocolate, coral, cornflowerblue, cornsilk, crimson, cyan, darkblue, darkcyan, darkgoldenrod, darkgray, darkgreen, darkgrey, darkkhaki, darkmagenta, darkolivegreen, darkorange, darkorchid, darkred, darksalmon, darkseagreen, darkslateblue, darkslategray, darkslategrey, darkturquoise, darkviolet, deeppink, deepskyblue, dimgray, dimgrey, dodgerblue, firebrick, floralwhite, forestgreen, fuchsia, gainsboro, ghostwhite, gold, goldenrod, gray, green, greenyellow, grey, honeydew, hotpink, indianred, indigo, ivory, khaki, lavender, lavenderblush, lawngreen, lemonchiffon, lightblue, lightcoral, lightcyan, lightgoldenrodyellow, lightgray, lightgreen, lightgrey, lightpink, lightsalmon, lightseagreen, lightskyblue, lightslategray, lightslategrey, lightsteelblue, lightyellow, lime, limegreen, linen, magenta, maroon, mediumaquamarine, mediumblue, mediumorchid, mediumpurple, mediumseagreen, mediumslateblue, mediumspringgreen, mediumturquoise, mediumvioletred, midnightblue, mintcream, mistyrose, moccasin, navajowhite, navy, oldlace, olive, olivedrab, orange, orangered, orchid, palegoldenrod, palegreen, paleturquoise, palevioletred, papayawhip, peachpuff, peru, pink, plum, powderblue, purple, red, rosybrown, royalblue, saddlebrown, salmon, sandybrown, seagreen, seashell, sienna, silver, skyblue, slateblue, slategray, slategrey, snow, springgreen, steelblue, tan, teal, thistle, tomato, turquoise, violet, wheat, white, whitesmoke, yellow, yellowgreen

# Opacity

- same values as alpha: 0% - 100%
  - opacity: 0.5;
- changes all child elements
  - different than alpha in color

# Rounded Corners

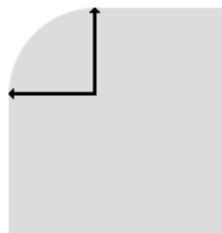
---

- border-radius: ##px
  - border-top-left-radius: ##px; border-top-right-radius: ##px;  
border-bottom-right-radius: ##px; border-bottom-left-radius: ##px;

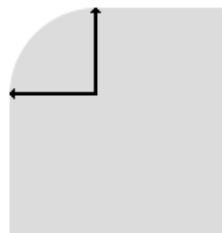


# Rounded Corners - unequal

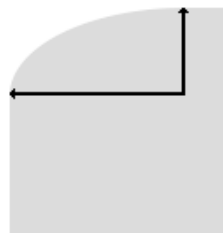
- `border-radius: ##px / ##px;`
  - horizontal / vertical radius
- `border-radius: ##px ##px ##px ##px / ##px ##px ##px ##px ;`



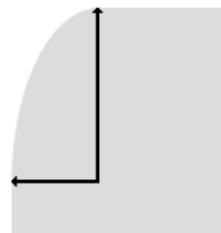
`border-top-left-radius: 50px;`



`border-top-left-radius: 50px 50px;`



`border-top-left-radius: 100px 50px;`



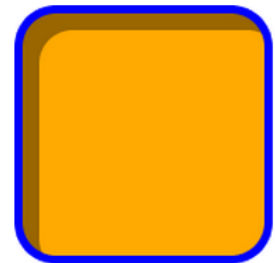
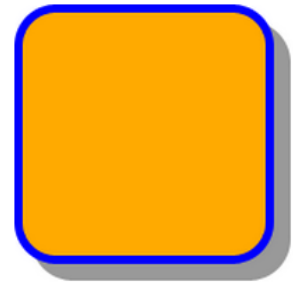
`border-top-left-radius: 50px 100px;`

# Filters

- manipulate the color and shape by blur, brightness, contrast, drop-shadow, grayscale, hue-rotate, invert, opacity, saturate, sepia
  - <http://html.adobe.com/webplatform/graphics/customfilters/>
    - Not old IE 'filter' property
- SVG object filters : > IE9
- CSS filters like SVG: >Edge 12
- SVG transforms, filters with CSS: > IE8

# Drop shadows

- box-shadow: inset #px #px # # rgba(##,##,##,##)
  - inset – optional
  - 1<sup>st</sup> – horizontal offset
  - 2<sup>nd</sup> – vertical offset
  - 3<sup>rd</sup> – optional, blur distance
  - 4<sup>th</sup> – optional , spread distance, pos or neg
  - 5<sup>th</sup> – recommended, color of shadow
- inset shadow effect
  - box-shadow: inset #px...



# Drop shadows

- Multiple shadows, use comma between
  - ```
form {  
  -webkit-box-shadow:  
    inset 1px 1px 84px rgba(0,0,0,0.24), inset -1px -1px 84px  
    rgba(0,0,0,0.24);  
  -moz-box-shadow:  
    inset 1px 1px 84px rgba(0,0,0,0.24), inset -1px -1px 84px  
    rgba(0,0,0,0.24);  
  box-shadow:  
    inset 1px 1px 84px rgba(0,0,0,0.24), inset -1px -1px 84px  
    rgba(0,0,0,0.24); }
```
- Images do a better job but
  - aren't reusable
  - require bandwidth

# Text shadows

- `text-shadow: #px #px # rgba(##,##,##,##)`
  - 1<sup>st</sup> – horizontal offset
  - 2<sup>nd</sup> – vertical offset
  - 3<sup>rd</sup> – optional, blur distance
  - 4<sup>th</sup> – recommended, color of shadow
- multiple shadows, use comma
- IE9 does not support
- @supports
  - example: `text/shadows-supports.html`



Drop Shadows



Multiple shadows are Hot



# CSS3 - Web sites

- Codrops tutorials HTML5 / CSS3 effects - <http://tympanus.net/codrops/category/tutorials/>
  - Background full size, slideshow - <http://tympanus.net/codrops/2012/01/02/fullscreen-background-image-slideshow-with-css3/>
- Menus, buttons
  - CSS Menu Maker - <http://css3menu.com/>
  - CSS Round/Shadowed Buttons - <http://cssround.com/>

# Exercise

- Follow instructions in Exercises handout for
  - 14. Contrast opacity and transparency
  - 15. Buttons with rounded corners and shadows



Chapter 7

# CSS3 Gradients and Backgrounds

# Gradients

---

- smooth transitions between start and stop colors
- color stops are intermediate colors at a specified position
- use anywhere a background would be used
- two types
  - linear
  - radial



# Gradients - linear


- two different syntaxes becoming one W3C standard
  - recommended to use vendor prefixes
- Smooth transitions or stripes
- IE9 can use SVG
  - Scalable vector graphics is an XML language
- IE can use filters for linear gradients only



# Gradients - linear

- Code generation tool
  - <http://www.colorzilla.com/gradient-editor/>
    - SCSS, import file to use
- Other tools
  - [John Allsopp](#)
  - [Damian Galarza](#) - generator
  - [Paul Irish](#) – CSS3, please

Preview



Orientation:  Size:  x  ☒ IE

CSS

```
background: rgb(255,255,255); /* Old browsers */
background: -moz-linear-gradient(top,
  rgba(255,255,255,1) 0%, rgba(0,0,0,1) 99%); /*
  FF3.6+ */
background: -webkit-gradient(linear, left top, left
  bottom, color-stop(0%,rgba(255,255,255,1)),
  color-stop(99%,rgba(0,0,0,1))); /*
  Chrome, Safari4+ */
background: -webkit-linear-gradient(top,
  rgba(255,255,255,1) 0%,rgba(0,0,0,1) 99%); /*
  Chrome10+, Safari5.1+ */
background: -o-linear-gradient(top,
  rgba(255,255,255,1) 0%,rgba(0,0,0,1) 99%); /*
  Opera11.10+ */
background: -ms-linear-gradient(top,
  rgba(255,255,255,1) 0%,rgba(0,0,0,1) 99%); /*
  IE10+ */
filter: progid:DXImageTransform.Microsoft.gradient(
  startColorstr='#ffffff',
  endColorstr='#000000',GradientType=0 ); /* IE6-9
  */
background: linear-gradient(top,
  rgba(255,255,255,1) 0%,rgba(0,0,0,1) 99%); /*
  W3C */
```

# Gradients - repeating

- `repeating-linear-gradient( )`
- `repeating-radial-gradient( )`
- same syntax as non-repeating

# Background – multiple images

---

- Compose a background from multiple images
  - to reduce bandwidth using smaller files
  - to change position of images independent of each other (parallax effects)
- Before:
  - nested divs with backgrounds



# Background – multiple images

- After
  - background-image:  
url(firstImage.jpg) no-repeat x-position y-position,  
url(secondImage.gif),  
url(thirdImage.png);
  - background-image:  
url(firstImage.jpg), url(secondImage.gif),  
url(thirdImage.png);  
background-repeat: no-repeat, repeat, repeat;  
background-position: left top, 0 0, 50% 50%
- 1st is top image, last is bottom image

# background-size

- when included in background declaration, not supported
- -prefix-background-size: 100px, auto; (100 wide, proportionate height)
- -prefix-background-size: cover; (fills but overflows)
- -prefix-background-size: contain; (fits, no overflow)

| Value      | Description                                                                                                                                                                                                   |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| length     | Sets the height and width of the background image. The first value sets the width, the second value sets the height. If only one value is given, the second is set to "auto"                                  |
| percentage | Sets the height and width of the background image in percent of the parent element. The first value sets the width, the second value sets the height. If only one value is given, the second is set to "auto" |
| cover      | Scale the image to the smallest size such that both its width and its height can fit inside the content area                                                                                                  |
| contain    | Scale the image to the largest size such that both its width and its height can fit inside the content area                                                                                                   |

# Exercise

- Follow instructions in Exercises handout for
  - 16. Use gradients



Chapter 8

# CSS3 Transforms and Transitions

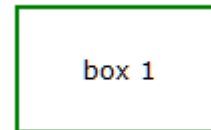
# Transforms

---

- Move (translate), scale, rotate, skew
- Supports 2D and 3D
- Additive
  - transform: translate(##) scale(#) rotate(#) skew(##);
  - applied in order
- Does not reflow text
- Needs vendor prefixes
- 2D – no IE8, uses filters
- 3D – no IE9 and no nesting at all.

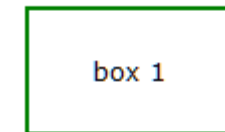
# Translate

- Measured from left top
- Use block or inline-block elements in WebKit
- Move from current position with
  - `transform: translate(x, y);`
  - `transform: translateX(#);`
  - `transform: translateY(#);`



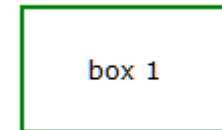
# Scale

- Scales from center of element by default
  - change with transform-origin
- Use percentage (1.5 or 150%)
- Grow/shrink from current position with
  - `transform: scale(#);` // proportional x & y
  - `transform: scale(x, y);`
  - `transform: scaleX(#);`
  - `transform: scaleY(#);`



# Rotate

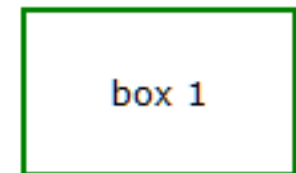
- Rotates from center (+clockwise, -countercw)
- Use degrees, grads, radians, or turns
- Rotate from current position with
  - transform: rotate(#deg);
- IE6, 7, 8 requires a filter
  - use a tool to get values like [css3please](#)





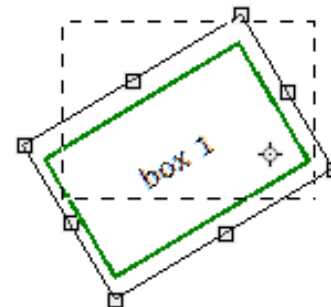
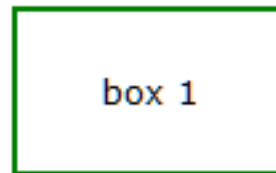
# Skew

- Skews from center of element
- Use degrees, grads, radians, or turns
- Skew from current position with
  - `transform: skew(#deg, #deg)`
  - `transform: skew(#deg)` or `transform: skewX(#deg)`
  - `transform: skewY(#deg)`



# Changing the origin

- The point which is nailed down
- Change the origin with respect to left top
  - `transform-origin: #px #px;`



# Tools - Transformations

- John Allsopp's CSS3 transform tool
  - <http://westciv.com/tools/transforms/index.html>
- Matthew Lein's Easing Animation tool
  - <http://matthewlein.com/ceaser/>

# Demo - transforms

- Unfolding the Box - a demo of transforms
  - <http://rupl.github.io/unfold/>
- Magic - a library of transforms
  - [http://minimamente.com/example/magic\\_animations](http://minimamente.com/example/magic_animations)

# Transitions

- Transitions follow the pattern of
  - `-prefix-transition: property duration timingFunction <delay>;`
- Properties are
  - all, opacity, zoom,
- Duration is in #s (seconds)
- Timing functions can be
  - ease (default), ease-in, ease-out, ease-in-out (popular)
- Delays are in #s also.



# Demos - animation

- 2011 Madmanimation “movie” demo
  - <http://stuffandnonsense.co.uk/content/demo/madmanimation/>
- Hyperspace effect demo
  - <http://codepen.io/noahblon/full/GKflw>

# Animation - CSS effects libraries

- Cloud-eight element animations
  - <http://www.cloud-eight.com/github/animations/>
- Animate.css
  - <http://daneden.me/animate/>
- Effeckt.css (Paul Irish)
  - <https://github.com/h5bp/Effeckt.css>
- LESS mixins
- SpinKit - animated icons when loading page|file
  - <http://tobiasahlin.com/spinkit/>

# Exercise

- Follow instructions in Exercises handout for
  - 17. Create overlapping Polaroids





# Web Fonts

# Web loaded fonts

---

- Best way
  - CSS3's @font-face loaded from CDN, requested by browser
    - also can be locally loaded from web server
- Old school
  - Arial, Verdana, Times, Georgia fonts loaded from client
  - enhance with images to use other fonts
- Old workarounds
  - sIFR, Cufón – loaded from web, processed outside of browser

# Two choices

- Use content hosting service
  - You don't want to create all the font versions.
  - You want a simple link and CSS
  - You want lots of font options.
- Host fonts yourself
  - You own the fonts.
  - You need a font that is not available publicly.
  - You aren't always online.

# Implementing @font-face

---

- The "at-rule"
  - maybe you've seen the `@import url(otherStyleSheet.css)`
  - a group of CSS engine processing instructions
- For each face use a declaration
  - `@font-face {`
    - `font-family: 'fontName';`
    - `src: source;`
    - `font-weight: weight;`
    - `font-style: style; }`

## @font-face { font-family: ...

- Not the real font name, but one you or the OS give it.
  - a programming variable
- Link to font file in the src: descriptor
- @font-face {  
    font-family: Gentium;  
    src:  
    url(http://example.com/fonts/Gentium123.ttf); }
- p { font-family: Gentium, serif; }

## @font-face { src:

- Use multiple sources separated by commas
  - First one matched wins
- src: url('fontName.???') format(?????),  
url('fontName.???') format(?????)
- Formats
  - EOT – proprietary for IE4-8
  - WOFF – Web Open Font Format
  - OTF – OpenType, iPhone 4.2+
  - TTF – TrueType
  - SVG – Scalable Vector Graphics, iPhone <4.2

# Tricking IE8 and before

- Thinks it's given one URL between the first quote to the last quote.
- Fix: Add a query string to the EOT URL
  - `src: url('fontName.eot?#iefix') format('eot'),...`
- Ignores the rest of the text for 99.9% of cases
- The other 0.1%, use a `<head>` tag
  - `<meta http-equiv="X-UA-Compatible" content="IE=Edge">`

# Font properties

- font-variant (small-caps)
- font-weight: bold selects a thicker face in the font-family.



- font-style: italic is the font variant not making it look oblique



Veggieburger Light - Lorem ipsum  
eiusmod tempor incididunt ut la  
quis nostrud exercitation ullamc  
aute irure dolor in reprehenderi  
pariatur. Excepteur sint occaec  
deserunt mollit anim id est labor

Veggieburger Regular - Lorem  
do eiusmod tempor incididunt ut  
veniam, quis nostrud exercitati  
consequat. Duis aute irure dol  
fugiat nulla pariatur. Excepteu  
officia deserunt mollit anim id e

Veggieburger Bold - Lorem ip  
eiusmod tempor incididunt ut l  
quis nostrud exercitation ullarr  
aute irure dolor in reprehende  
pariatur. Excepteur sint occae  
deserunt mollit anim id est labi



# Font properties

- font-stretch – condensed, normal, expanded
  - no webkit yet, in development

e e **e** e e e e e e

condensed normal expanded

- font-size-adjust – preserving x-height on font fallback
  - only Firefox, experimental Chrome

b b b b b b

# Font properties

- font-synthesis
  - should the browser oblique and overstrike (synthetic faces)?
  - = style, = weight, or =none

# Font download resources

- [Font Squirrel](#) download kit includes
  - TrueType, EOT, WOFF, SVG, Cufón and fonts
  - Demo.html and stylesheet.css
  - Upload your own and generate kit

Undercover | 1 Font  
[View Font](#) | [View @ff Demo](#) | [Get Kit](#)

UNDERCOVER

Vanilla | 1 Font  
[View Font](#) | [View @ff Demo](#) | [Get Kit](#)

Vanilla AaBb

Veggieburger | 3 Fonts  
[View Font](#) | [View @ff Demo](#) | [Get Kit](#)

Veggieburger A

# Free font service resources

- Google Fonts online - <http://www.google.com/webfonts>

Google Fonts

DIRECTORY FEATURED ABOUT

Viewing 217 of 808 font families

Open Sans  
Steve Matteson (10 styles)

All their equipment and instruments are alive.

Roboto  
Christian Robertson (12 styles)

Sent... Regul... 40px

Almost before we knew it, we had left the ground.

SEE SPECIMEN

Lato  
Łukasz Dziedzic (10 styles)

It was going to be a lonely trip back.

Oswald  
Vernon Adams (3 styles)

My two natures had memory in common.

Roboto Condensed  
Christian Robertson (6 styles)

Silver mist suffused the deck of the ship.

Source Sans Pro  
Paul D. Hunt (12 styles)

The face of the moon was in shadow.

Categories

☐ Serif

☒ Sans Serif

☐ Display

☐ Handwriting

☐ Monospace

Sorting

Trending

Languages

All Languages

Number of styles

Thickness

Slant

Width

Add to Collection

# Free font service resources

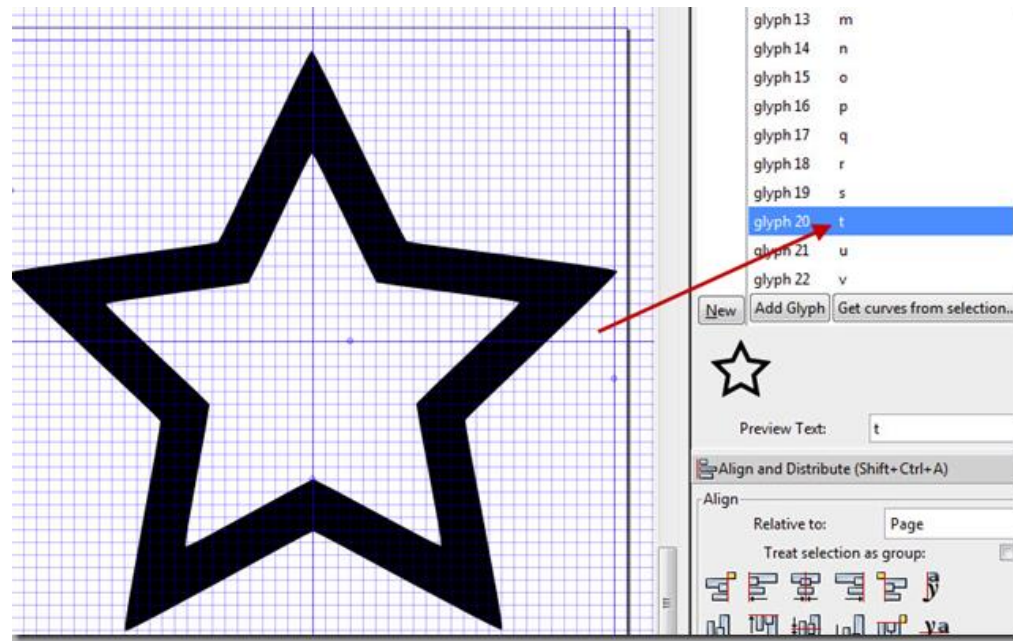
- Adobe Web Fonts – 500+  
<https://edgewebfonts.adobe.com/>
- Paid / free
  - [Typotheque](#), [Webtype](#), [Fontdeck](#), and [Fonts.com](#)

# Web font testing

- Typecast - <http://typecast.com/>
  - link from Google
- Type Wonder - preview font on site
  - <http://typewonder.com>
- Chrome extensions
  - <https://chrome.google.com/webstore/detail/google-font-previewer-for-engndlnldodigdjamndkplafgmkkencc>

# Using fonts instead of sprites

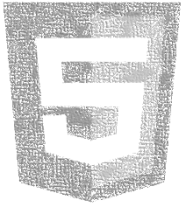
- Use SVG editor to make initial font file
  - <http://inkscape.org/>
- Convert SVG to a font kit
  - <http://www.fontsquirrel.com/fontface/generator>



# Icon fonts

- Font Awesome - <http://fontawesome.io/icons/>
  - Use <https://cdnjs.com/> to get resource
  - <https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.6.3/css/font-awesome.min.css>
- Fontello - icon fonts generator, multiple icon fonts - <http://fontello.com/>
- Using fonts as bullets
  - <http://blogs.sitepointstatic.com/examples/tech/icon-bullets/index.html>





# Exercise

- Follow instructions in Exercises handout for
  - 18. Google Web Fonts



# Media queries

# Media queries

---

- Adaptable layouts for mobile devices, print, ...
- @media screen {
  - /\* layout, typography, colors rule sets \*/ }
- @media print {
  - /\* layout, typography, colors rule sets \*/ }
- }

# Media queries

- Two ways
  - `@import url("color.css") screen and (color);`
  - `@media print{ /* styles */ }`  
or
  - `<link rel="stylesheet" href="style.css" media="screen and (color)">`
- Media types
  - all, braille, embossed, handheld, print, projection, screen, speech, tty, tv

# Features

| Value       | Description                                                                                                                                                            |
|-------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| width       | Specifies the width of the targeted display area.<br>Example: <code>media="screen and (min-width:500px)"</code>                                                        |
| orientation | Specifies the orientation of the target display/paper.<br>Possible values: "portrait" or "landscape"<br>Example: <code>media="all and (orientation: landscape)"</code> |

Others: height, aspect-ratio, device-height, device-width, device-aspect-ratio, color, color-index, monochrome, resolution, scan, grid

# Feature operators

- and
  - `@media screen and (min-width: 400px) and (max-width: 700px) { ... }`
- not
- ,
  - OR operator
  - `@media handheld and (min-width: 20em), screen and (min-width: 20em) { ... }`
- Only
  - Used to hide style sheets from older browsers

# Responsive web design

- **Change**
  - Layouts – 3 columns → 1 column
  - Fonts – change sizes
  - Images – change size, resolutions
  - Widgets – change type (horizontal nav bar → drop down)
- **Based on**
  - Browser type
  - Screen width
- **Executed by**
  - CSS media queries
  - JavaScript
  - Back end processing

# Exercise

- Follow instructions in Exercises handout for
  - 19. Responsive web design with media queries



**HTML**



**Other**

# Multicolumns

- Columns dynamically resize by width and length
- <IE9 – no support
- LESS

## VIDEO IS THE FINAL FRONTIER, AND NOW WE HAVE CONQUERED IT!

Aliquam erat volutpat. Mauris vel neque sit amet nunc gravida congue sed sit amet purus. Quisque lacus quam, egestas ac tincidunt a, lacinia vel velit. Morbi ac commodo nulla.

Mauris vel neque sit amet nunc gravida congue sed sit amet purus. Quisque lacus quam, egestas ac tincidunt a, lacinia vel velit. Morbi ac commodo nulla.

Neque sit amet nunc gravida congue sed sit amet purus. Quisque lacus quam, egestas ac tincidunt a, lacinia vel velit. Morbi ac commodo nulla.

## VIDEO IS THE FINAL FRONTIER, AND NOW WE HAVE CONQUERED IT!

Aliquam erat volutpat. Mauris vel neque sit amet nunc gravida congue sed sit amet purus. Quisque lacus quam, egestas ac tincidunt a, lacinia vel velit. Morbi ac commodo nulla.

In condimentum orci id nisl volutpat bibendum. Quisque commodo hendrerit lorem quis egestas. Vivamus rutrum nunc non neque consectetur quis placerat neque lobortis. Nam vestibulum, arcu sodales feugiat consectetur, nisl orci bibendum elit, eu euismod magna sapien ut

# Multicolumns - properties

- column-count: max number of columns allowed, auto will be one.
- column-width: min width of column, used to best fit text
- columns: combines count and width
- column-gap: gutter width
- column-rule: thin solid black (width style color)
- height: setting height will limit vertical size
  - text will overflow, or disappear (overflow:hidden)

# Animation

- @keyframes: no support <IE10, Safari, Chrome
  - Safari, Chrome support @-webkit-keyframes
- **@keyframes mymove** {  
    from {top:0px;} to {top:200px;} }
- **div** {  
    animation:mymove 5s infinite;  
    position:relative; }
- CSS3 Animation Cheat Sheet
  - <http://justinaguilar.com/animations>

# Flexbox

- A better table with flow used for layouts
  - 3 syntax styles - old, tweener, new

| IE   | Edge * | Firefox | Chrome | Safari | Opera |
|------|--------|---------|--------|--------|-------|
|      |        |         | 29     |        |       |
|      |        |         | 49     |        |       |
|      |        |         | 50     |        |       |
| 8    | 13     | 47      | 51     |        |       |
| 4 11 | 14     | 48      | 52     | 9.1    | 39    |
|      |        | 49      | 53     | 10     | 40    |
|      |        | 50      | 54     | TP     | 41    |
|      |        | 51      | 55     |        |       |

# Flexbox simple grid

- A table-tr-td layout but with divs and classes
- `<div class="grid">`
  - `<div class="grid-row">`
    - `<div class="grid-item">1</div>`
    - `<div class="grid-item">2</div> ...`
  - `</div>`
  - `<div class="grid-row">...</div>`
- `</div>`

|                                                                                                                                                                                                                                                       |                                                                                                                                                                                                                                         |                                                                                                                                                                                                                                           |                                                                                                                                                                                                    |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Lorem ipsum dolor sit amet, consectetur adipisicing elit. Nisi sunt nihil aspernatur provident quos ipsam, omnis vitae similique officia aliquam quibusdam, incidunt tenetur. Pariatur reprehenderit aut, unde rerum beatae, consectetur.             |                                                                                                                                                                                                                                         | Lorem ipsum dolor sit amet, consectetur adipisicing elit. Tempore facilis maiores fuga error, et dolorum quo. Quas doloribus nesciunt excepturi voluptatibus ullam ex quasi. Quia repellat, cumque repellendus quisquam error.            |                                                                                                                                                                                                    |
| Lorem ipsum dolor sit amet, consectetur adipisicing elit. Praesentium, sunt, voluptatem? Sit molestiae voluptas nobis sequi deleniti assumenda quam qui doloribus, iste rem laborum repellendus nemo velit placeat similique, atque?                  | Lorem ipsum dolor sit amet, consectetur adipisicing elit. Incidunt quis, harum recusandae, consequatur necessitatibus distinctio neque at hic nesciunt assumenda! Pariatur illum sunt placeat. Quam impedit neque sequi voluptates eum! | Lorem ipsum dolor sit amet, consectetur adipisicing elit. Necessitatibus, vitae tempore, ut libero facere nam fuga corporis distinctio error esse autem, doloribus eos voluptatem odit consequatur. Totam provident, quisquam eius.       |                                                                                                                                                                                                    |
| Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sapiente dignissimos, rem quisquam laboriosam consectetur laborum architecto nulla ullam? Eveniet ipsam temporibus adipisci est perspiciatis soluta corrupti obcaecati esse magni incidunt! | Lorem ipsum dolor sit amet, consectetur adipisicing elit. Non magnam, reiciendis sunt delectus et natus quisquam tenetur quasi optio eius nisi nemo cupiditate vitae iste officis nam repellat expedita. Inventore.                     | Lorem ipsum dolor sit amet, consectetur adipisicing elit. Sed voluptate consequatur enim quidem deleniti blanditis, modi ipsum repellat. Praesentium dolores tempore enim! Error a assumenda sed, officia Architecto, repellat, incidunt. | Lorem ipsum dolor sit amet, consectetur adipisicing elit. Ea saepe obcaecati, nobis quam ipsum quisquam illum qui voluptatem ipsa, rem ex cum nulla nostrum ut, fugit, repudiandae autem sed nisi. |

# Flexbox simple grid CSS

- Row layout
  - `.grid { border: solid 1px #e7e7e7; }`
  - `.grid-row { display: flex; }`
  - `.grid-item { flex: 1; padding: 12px; border: solid 1px #e7e7e7; }`
- Column layout
  - `.grid-row { display: flex; flex-direction: column; }`

# Flexbox responsive grid HTML

- `<div class="grid-row collapse-lt-480 ">`
  - `<div class="grid-item">lorem</div><div class="grid-item"> lorem </div>`
- `</div>`
- `<div class="grid-row collapse-lt-720 ">`
  - `<div class="grid-item"> lorem </div><div class="grid-item"> lorem </div><div class="grid-item"> lorem </div>`
- `</div>`
- `<div class="grid-row collapse-lt-960 ">`
  - `<div class="grid-item"> lorem </div><div class="grid-item"> lorem </div><div class="grid-item"> lorem </div><div class="grid-item"> lorem </div>`
- `</div>`



# Flexbox responsive grid CSS

- @media all and ( min-width: 480px ) {
  - .collapse-lt-480 { flex-direction: row; } }
- @media all and ( min-width: 720px ) {
  - .collapse-lt-720 { flex-direction: row; } }
- @media all and ( min-width: 960px ) {
  - .collapse-lt-960 { flex-direction: row; } }

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nisi sunt nihil aspernatur provident quos ipsam, omnis vitae similique officia aliquam quibusdam, incidunt tenetur. Pariatur reprehenderit aut, unde rerum beatae, consectetur.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Tempore facilis maiores fuga error, et dolorum quo. Quas doloribus nesciunt excepturi voluptatibus ullam ex quasi. Quia repellat, cumque repellendus quisquam error.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesentium, sunt, voluptatem? Sit molestiae voluptas nobis sequi deleniti assumenda quam qui doloribus, iste rem laborum repellendus nemo velit placeat similique, atque?

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Incidunt quis, harum recusandae, consequatur necessitatibus distinctio neque at hic nesciunt assumenda! Pariatur illum sunt placeat. Quam impedit neque sequi voluptates eum!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Necessitatibus, vitae tempore, ut libero facere nam fuga corporis distinctio error esse autem, doloribus eos voluptatem odit consequatur. Totam provident, quisquam eius.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sapiente dignissimos, rem quisquam laboriosam consectetur laborum architecto nulla ullam? Eveniet ipsam temporibus adipisci est perspiciatis soluta corrupti obcaecati esse magni incidunt!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Non magnam, reiciendis sunt delectus et natus quisquam tenetur quasi optio eius nisi nemo cupiditate vitae iste officiis nam repellat expedita Inventore

# Flexbox - links

- Cheatsheet - <http://www.sketchingwithcss.com/samplechapter/cheatsheet.html>
- CSS-Tricks - <http://css-tricks.com/snippets/css/a-guide-to-flexbox/>
- Smashing - <http://coding.smashingmagazine.com/2013/05/22/centering-elements-with-flexbox/>
- Mozilla - [https://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Flexible\\_boxes](https://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Flexible_boxes)
- Examples (used here) - <http://callmenick.com/post/flexbox-examples>



## Additional APIs

uses  
JAVASCRIPT

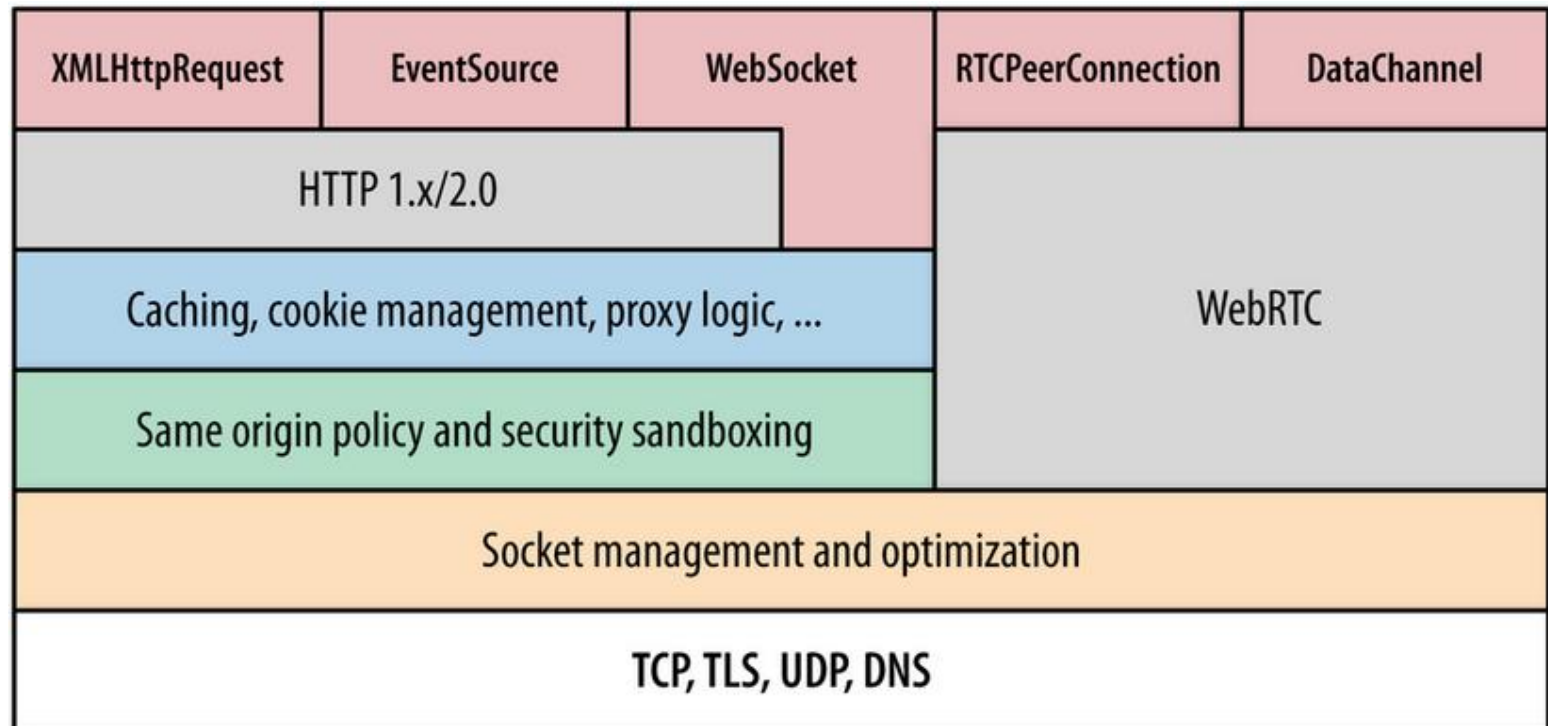
# Web workers

- Concurrent (threaded) JavaScript
- speeds up JS heavy pages
- not supported in Android 4.3, and of course, IE 9
  - 12/2013 - 72%
- based on XMLHttpRequest

# I/O



- Server Sent Events (SSE) - server to client only
- WebRTC (Real Time Comm.) - audio, video, and data sharing between browsers (peers)



# I/O - WebSocket

- bidirectional, message-oriented streaming of text and binary data
- low level, good for custom app protocol messaging
- supported by web servers like IIS...
- not supported in Android browser, and.... IE 9
- JavaScript libraries
  - <http://pusher.com>

# Reusable components

- Shadow DOM – DOM boundaries
  - make it possible to build widgets which
    - can be reused reliably
    - won't break pages if the next version of the component changes internal implementation details.
  - <http://www.html5rocks.com/en/tutorials/webcomponents/shadowdom/>
- HTML Templates – supported in Edge 13
- Also see Google Polymer, Material Design

# Geolocation

---

- Requires permission
- Gets current latitude and longitude from device or IP through JavaScript



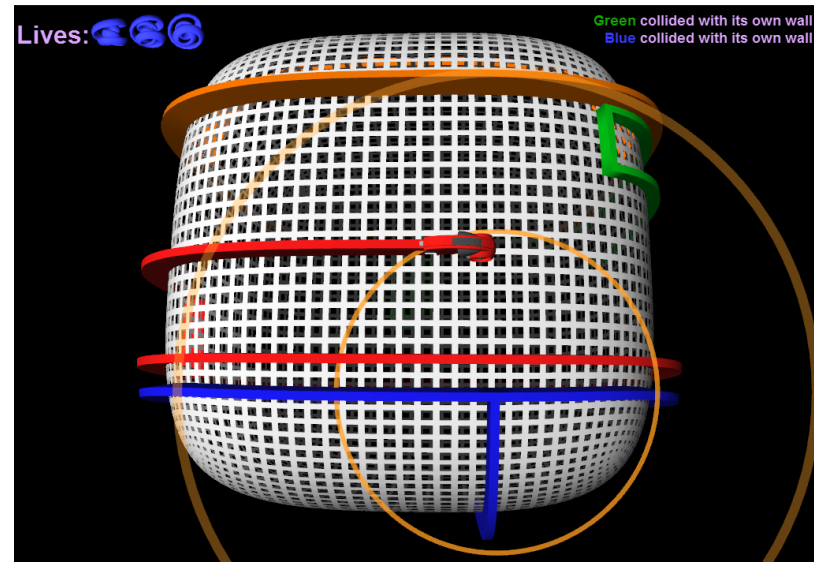
# Offline apps

---

- AppCache
- FileAPI
- localStorage
  - Allows storage through different formats
  - Much like a larger cookie.
  - Not complete support yet.

# Canvas

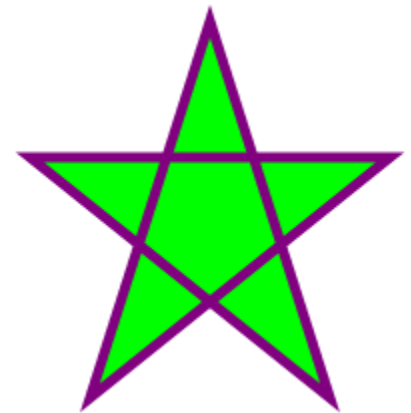
- JavaScript enabled bit-mapped drawing
- speed better than anything except Flash
- Apple developed
- WebGL - 3D JavaScript library, no IE
  - <http://cycleblob.com/>
- <http://hakim.se/experiments>



# SVG

---

- Scalable Vector Graphics, an XML language
- `<canvas>` is bitmapped, SVG is vector (zoomable)
- `>IE8`
  - filters not supported in IE9 (no sharp edges)
- `<svg height="210" width="500">`
- `<polygon points="100,10 40,198 190,78 10,78 160,198" style="fill:lime;stroke:purple;stroke-width:5;fill-rule:nonzero;"/>`
- `</svg>`



# SVG

- use open source images to borrow
  - <http://www.opencart.org/>
- use a tool to create
  - Inkscape - <http://inkscape.org>
  - SVG Edit (web) - <http://code.google.com/p/svg-edit/>
- use a JavaScript library to interact
  - Raphaël - <http://dmitrybaranovskiy.github.io/raphael/>

# Drag and drop

- not supported in iOS, Opera
  - iPad is a major part of the D&D market

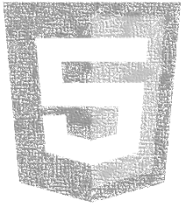
| IE   | Edge * | Firefox | Chrome | Safari | Opera | iOS Safari * | Opera Mini * | Android Browser * | Chrome for Android |
|------|--------|---------|--------|--------|-------|--------------|--------------|-------------------|--------------------|
| 1 8  |        |         | 31     |        |       |              |              | 4.1               |                    |
| 1 9  |        | 38      | 43     |        |       | 7.1          |              | 4.3               |                    |
| 2 10 |        | 39      | 44     |        | 31    | 8.4          |              | 4.4.4             |                    |
| 2 11 | 2 12   | 40      | 45     | 8      | 32    | 9            | 8            | 44                | 44                 |
|      | 2 13   | 41      | 46     | 9      | 33    |              |              |                   |                    |
|      |        | 42      | 47     |        | 34    |              |              |                   |                    |
|      |        | 43      | 48     |        |       |              |              |                   |                    |

# Other APIs - no IE/Edge/Firefox support

- CSS Shapes
  - allows text to flow around content by marking where it should be allowed to go (not restricting based on image, etc.).
- HTML Imports
  - a way to include and reuse HTML in other HTML docs. No IE/Edge/Firefox/Safari support.
- Custom Elements
  - author defined DOM elements
- Speech Recognition API

# Other APIs – no IE/Edge support

- getUserMedia API
  - provides access to multimedia streams (video, audio, or both) from local devices
- RTCPeerConnection API
  - send data across the Internet
- CSS Masks
- CSS Clip Paths



# Exercises

- 20. Geolocation
- 21. Web storage



**HTML**



**End matter**

# Songs

- HTML5 it is a-changin' – Opera web evangelist Bruce Lawson
  - <http://www.youtube.com/watch?v=7gNmFabAOGc> (2010)
- Web Standards Hoedown - Opera Developer Relations
  - <http://www.youtube.com/watch?v=SFvYhEhFsdQ> (2012)
- CSS3 Song from an Opera Developer Relations employee
  - <http://www.youtube.com/watch?v=5b7dNNapMCM> (2010)