HTML5
Huh, What is it good for?
Plug: Introducing HTML5
It's more than

- HTML
- New HTML5 elements
- Buzzword
- Simple doctype
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>My HTML5 page</title>
  <style>
    body {
      font-family: sans-serif;
    }
  </style>
</head>
<body>
<article>
  <h1>My story</h1>
  <p>With JavaScript</p>
</article>
<script src="funky.js"></script>
<script>
  makeFunky(document.querySelector('article'));
</script>
</body>
</html>
Homepage

- Offline
- App cache
- WebStorage
- WebSQL
- IndexedDB
- Multimedia
- Video & Audio

JavaScript
○ X-domain messaging
○ CORS
○ File API
○ Drag & Drop
○ History API
○ Lots, lots more.

JavaScript
JavaScript

- Drawing
- Offline
- URLs
- Real-time
when can I use "HTML5"?
Node utility to inline images, CSS and JavaScript for a web page - useful for mobile sites — Read more
click here to add a homepage

reverting back to online jQuery. Closes #7

remy (author)
2 days ago

<table>
<thead>
<tr>
<th>file</th>
<th>date</th>
<th>message</th>
</tr>
</thead>
<tbody>
<tr>
<td>bin/</td>
<td>May 16, 2011</td>
<td>fixed up binary [remy]</td>
</tr>
<tr>
<td>.gitignore</td>
<td>May 15, 2011</td>
<td>ignore file [remy]</td>
</tr>
<tr>
<td>README.md</td>
<td>3 days ago</td>
<td>Updated docs [remy]</td>
</tr>
<tr>
<td>inliner.js</td>
<td>2 days ago</td>
<td>reverting back to online jQuery. Closes #7 [remy]</td>
</tr>
<tr>
<td>package.json</td>
<td>2 days ago</td>
<td>reverting back to online jQuery. Closes #7 [remy]</td>
</tr>
</tbody>
</table>
What's happening?

@Mentions Retweets Searches Lists

emmoboulton Emma Boulton
Getting flashbacks to 2007 and my old life at the BBC, as I write a report on online news consumption.
45 seconds ago

danwrong Dan Webb
Ha! Got two passes to a preview screening of Attack The Block in SF on Wednesday night. Anyone want to join me?
3 minutes ago

emmoboulton Emma Boulton
Hungry people of the internet, I present to you, 5 min cup cake in a mug: howto.wired.com/wiki/Make_Cake... (via @aexmo)
6 minutes ago

taxsharp Tristan Sharp
Call me what you like, just make sure you don't forget dim as I've just realised why every second home in Brighton has an "open house" banner
8 minutes ago

mollydotcom Molly E. Holzschlag
Third Red Flag: Web is Down More Details E...

Your Tweets 14,206

1 hour ago: It appears to have stopped raining in Bergen. People look confused.

Following 98

Followers 14,075

Favorites 1,196

Starred: gekitsuu @rem FYI when you're doing your demos you can edit the java script li...

Trends
United Kingdom • change
#baftas #whydoesomegirls #20bandsilove

Listed 1,687

Recently listed in: jscript-gurus, People-I-Retweet, Nosailor, developer, web

Who to follow
Suggestions for you • view all

Disney • Follow

Emily Toop • Follow

David Flanagan • Follow

Refresh suggestions
Browse interests • Find friends
A Tale Of Two Cities
Making it work in the "other" browser
Polyfills will/might save you

A shim that mimics a future API providing a fallback to older browsers

http://goo.gl/0Z9eI
Tools

Modernizr to detect support

yepnode.js to conditionally load
(available as part of Modernizr)
Modernizr.load({
  test: Modernizr.geolocation,
  nope: 'geo-polyfill.js',
  complete: initMyGeoApp
});
Oh, and learn JavaScript.
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8" />
<title>Canvas</title>
</head>
<body>
<canvas></canvas>
</body>
</html>
2D API

ctx = canvas.getContext('2d')
A Tale Of Two Cities

London vs. Berlin

Price: £10k - £4,888k
£11k - £5,605k

powered by nestoria

built by LeftLogic
- Gradients
- Pixel manipulation
- Paths
- Shadows
- Export to data url
- State saving
Gradients
Gradients

```javascript
var rainbow = ctx.createLinearGradient(0, 0, w, h),
colours = {
  0: 'red',
  0.167: 'orange',
  0.333: 'yellow',
  // etc
};

for (var key in colours) {
  rainbow.addColorStop(key, colours[key]);
}

ctx.fillStyle = rainbow;
ctx.fillRect(0, 0, w, h);
```
Pixel Pushing
var pixels = ctx.getImageData(0, 0, w, h),
len = pixels.data.length;

for (var i = 0; i < len; i += 4) {
  var rgb = 'rgb(' +
            pixels.data[i],   // red
            pixels.data[i+1], // green
            pixels.data[i+2]  // blue
          ].join(',') + ')';
}
Shadows

ctx.shadowBlur = 2;
ctx.shadowOffsetX = 2;
ctx.shadowOffsetY = 2;
ctx.shadowColor = 'rgba(0,0,0,.7)';
var pinsize = 26,
pinedge = pinsize * 0.1,
centre = pinsize/2,
radius = centre - pinedge,
circle = Math.PI * 2;

ctx.fillStyle = rgb; // from our pixel
ctx.beginPath();
ctx.arc(centre, centre, radius, 0, circle, true);
ctx.closePath();
ctx.fill();
ctx.canvas.toDataURL('image/png')

data:image/png;base64,...
context.canvas

All context contain back-reference to it's canvas
State Saving

```javascript
pinctx.fillStyle = '#fff';
pinctx.save();
pinctx.fillStyle = '#f00';
// do something
pinctx.restore();
// now fillStyle is #fff
```
A Tale Of Two Cities
You are not connected to the Internet.
Safari can't open the page "http://html5demos.com/" because your computer isn't connected to the Internet.

Network Diagnostics can help you solve network connection problems.

Network Diagnostics...
Using a Manifest

<!DOCTYPE html>
<html manifest="my.appcache">
<body>
<!-- my page -->
</body>
</html>
my.appcache

CACHE MANIFEST
app.html
css/style.css
js/app.js
#version 13
1. Serve as text/manifest, by adding to mime.types:

   text/cache-manifest appcache
Firefox caching

<IfModule mod_expires.c>
ExpiresActive on
ExpiresByType text/cache-manifest ↬ “access plus 0 seconds”
</IfModule>
2. First line must be:

CACHE MANIFEST
The Manifest

3. Including page is implicitly included in the cache.
The Manifest

4. Two further namespaces: NETWORK & FALLBACK
CACHE MANIFEST

CACHE:
app.js
app.css
index.html

NETWORK:
http://*
https://*

FALLBACK:
/ offline.html
What to cache

CACHE MANIFEST

CACHE:
app.js
app.css
index.html

NETWORK:
http://*
https://*

FALLBACK:
/ offline.html
App cache requires all resources are accounted for.
Requests for files not found in the cache, are directed to offline.html (when offline).

CACHE MANIFEST

CACHE:
app.js
app.css
index.html

NETWORK:
http://*
https://*

FALLBACK:
/ offline.html
CACHE MANIFEST

CACHE:
app.js
app.css
index.html

NETWORK:
http://*
https://*

FALLBACK:
/ offline.html
5. Include some versioning to cache bust your manifest

# version 16
> :help

:load <url> - to inject new DOM
:load <script_url> - to inject external library
load also supports following shortcuts:
jquery, underscore, prototype, mootools, dojo, rightjs, coffeescript, yui.
eg: :load jquery
:listen [id] - to start remote debugging session
:clear - to clear the history (accessed using cursor keys)
:about

Directions to inject JS Console in to any page (useful for mobile debugging)
When your app updates

```javascript
applicationCache.onUpdateReady = function () {
    if (confirm("New version ready. Refresh?")) {
        // reload
        window.location = window.location;
    }
};
```
Tip

Do offline last
Web storage
Key / Value pair

- localStorage, persistent
- sessionStorage, temporary
- Extremely simple to use
- Storage events
sessionStorage.name = '@rem';

// page is reloaded
var name = sessionStorage.name;
if (name !== undefined) {
    alert('Welcome back ' + name);
}
sessionStorage.name = '@rem';

// page is reloaded
var name = sessionStorage.name;
if (name !== undefined) {
    alert('Welcome back ' + name);
}
sessionStorage.name = '@rem';

// page is reloaded
var name = sessionStorage.name;
if (name !== undefined) {
    alert('Welcome back ' + name);
}
storage coerces to string
var sess = sessionStorage;
sess.data = JSON.stringify(data);
var sess = sessionStorage;
sess.data = JSON.stringify(data);

var data = JSON.parse(sess.data || '{}');
Events

window.onstorage = function(e){
  // e.key
  // e.newValue
  // e.oldValue
  // e.url
};
URLS
> :help

i :load <url> – to inject new DOM
:load <script_url> – to inject external library
load also supports following shortcuts:
jquery, underscore, prototype, mootools, dojo, rightjs,
coffeescript, yui.
    eg. :load jquery
:listen [id] – to start remote debugging session
:clear – to clear the history (accessed using cursor keys)
:about

Directions to inject JS Console in to any page (useful for mobile debugging)
History API

```javascript
history.pushState(obj, title, url);
window.onpopstate = fn;
```
Detection

!!(window.history &&
  history.pushState)
Polyfill

History.js

https://github.com/balupton/history.js
window.onhashchange = function(e) {
    // user went back or forward
    // now read location.hash
};
Server Sent Events

"push based XHR"
Prerequisites

- **Persistent connection support**
  (event based are favourable: ie. node, twisted, etc)

- **Support in browser***
  but we can fix that :)

- If not native, fails over to polling
Limitations

- Same origin rules
- No support in Firefox or IE
- Read only
var es = new EventSource(url);

es.onopen = function () {
    console.log('open stream');
};

es.onmessage = function (event) {
    console.log('got message: ' + event.data);
};
app.get('/sse', function (req, res) {
    connections.push(res);

    res.writeHead(200, {
        'Content-Type': 'text/event-stream',
        'Cache-Control': 'no-cache'
    });
    res.write('eventId:0

');
    res.write('eventId:0

');
});
Server code

connections.forEach(function (c) {
    c.write('data:' + JSON.stringify(req.body) + '\n\n');
});

req.body is an object of key/values
>:listen geekninja

i Creating connection...

i Connected to "geekninja"

>:help

i :load <url> - to inject new DOM
    :load <script_url> - to inject external library
    load also supports following shortcuts:
    jquery, underscore, prototype, mootools,
    dojo, rightjs, coffeescript, yui.
    eg. :load jquery

    :listen [id] - to start remote debugging session
    :clear - to clear the history (accessed using
cursor keys)
    :about

Directions to inject JS Console in to any page
(useful for mobile debugging)
Web Sockets
Two way real-time comms

http://www.flickr.com/photos/44442915@N00/4960579336
In a nutshell

- Persistent connection
- Tiny chunks of data exchanged
- Bi-directional & no origin rules
Some Uses

- Chat / "hello world"
- Streaming data like stock share prices
- Multi-player game state
- Google Wave remember? :)


Native or polyfilled

IE6 ✓ : ( 
http://github.com/gimite/web-socket-js/

### web-socket-js /

<table>
<thead>
<tr>
<th>name</th>
<th>age</th>
<th>message</th>
<th>history</th>
</tr>
</thead>
<tbody>
<tr>
<td>.gitignore</td>
<td>August 10, 2009</td>
<td>Initial version. Checked only with Chrome.</td>
<td>[gimite]</td>
</tr>
<tr>
<td>FABridge.js</td>
<td>August 10, 2009</td>
<td>Initial version. Checked only with Chrome.</td>
<td>[gimite]</td>
</tr>
<tr>
<td>README.txt</td>
<td>January 24, 2010</td>
<td>Referring to latest spec.</td>
<td>[gimite]</td>
</tr>
</tbody>
</table>
var url = 'ws://node.remysharp.com:8000',
    conn = new WebSocket(url);

conn.onopen = function () {
    conn.send('@rem says Hi!');
};

conn.onmessage = function (event) {
    console.log(event.data);
};
var url = 'ws://node.remysharp.com:8000',
    conn = new WebSocket(url);

conn.onopen = function () {
    conn.send('@rem says Hi!');
};

conn.onmessage = function (event) {
    console.log(event.data);
};
```javascript
var url = 'ws://node.remysharp.com:8000',
    conn = new WebSocket(url);

conn.onopen = function () {
    conn.send('@rem says Hi!');
};

conn.onmessage = function (event) {
    console.log(event.data);
};
```
var url = 'ws://node.remysharp.com:8000',
    conn = new WebSocket(url);

conn.onopen = function () {
    conn.send('@rem says Hi!');
};

conn.onmessage = function (event) {
    console.log(event.data);
};
var url = 'ws://node.remysharp.com:8000',
    conn = new WebSocket(url);

conn.onopen = function () {
    conn.send('@rem says Hi!');
};

conn.onmessage = function (event) {
    console.log(event.data);
};
var url = 'ws://node.remysharp.com:8000',
    conn = new WebSocket(url);

conn.onopen = function () {
    conn.send('@rem says Hi!');
};

conn.onmessage = function (event) {
    console.log(event.data);
};
var url = 'ws://node.remysharp.com:8000',
    conn = new WebSocket(url);

conn.onopen = function () {
    conn.send('@rem says Hi!');
};

conn.onmessage = function (event) {
    console.log(event.data);
};
Questions?

@rem
remy@leftlogic.com