new whizzy javascript stuff

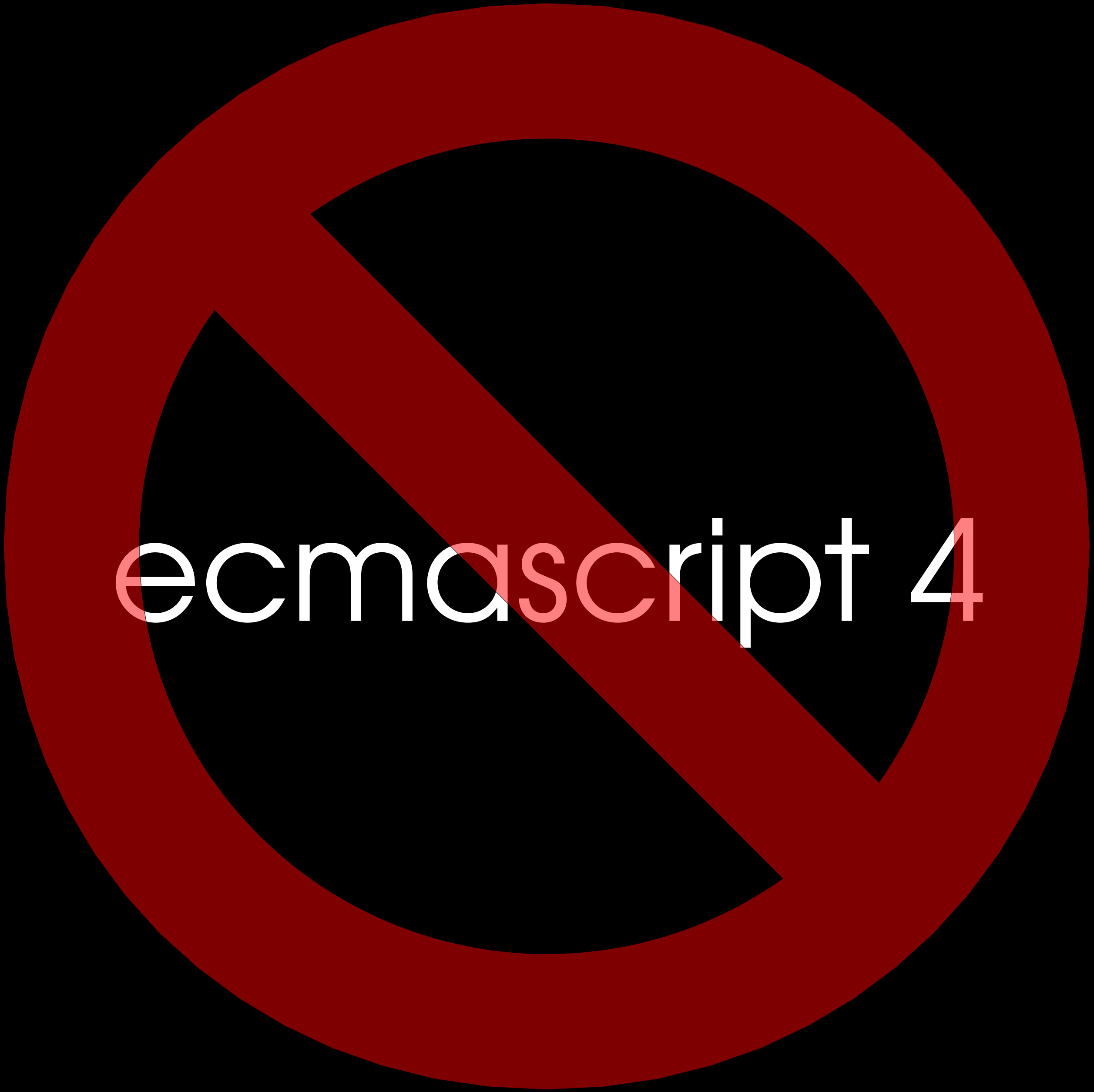
stuart langridge
http://www.kryogenix.org/
@sil

aka the glorious new future

stuart langridge, 2009

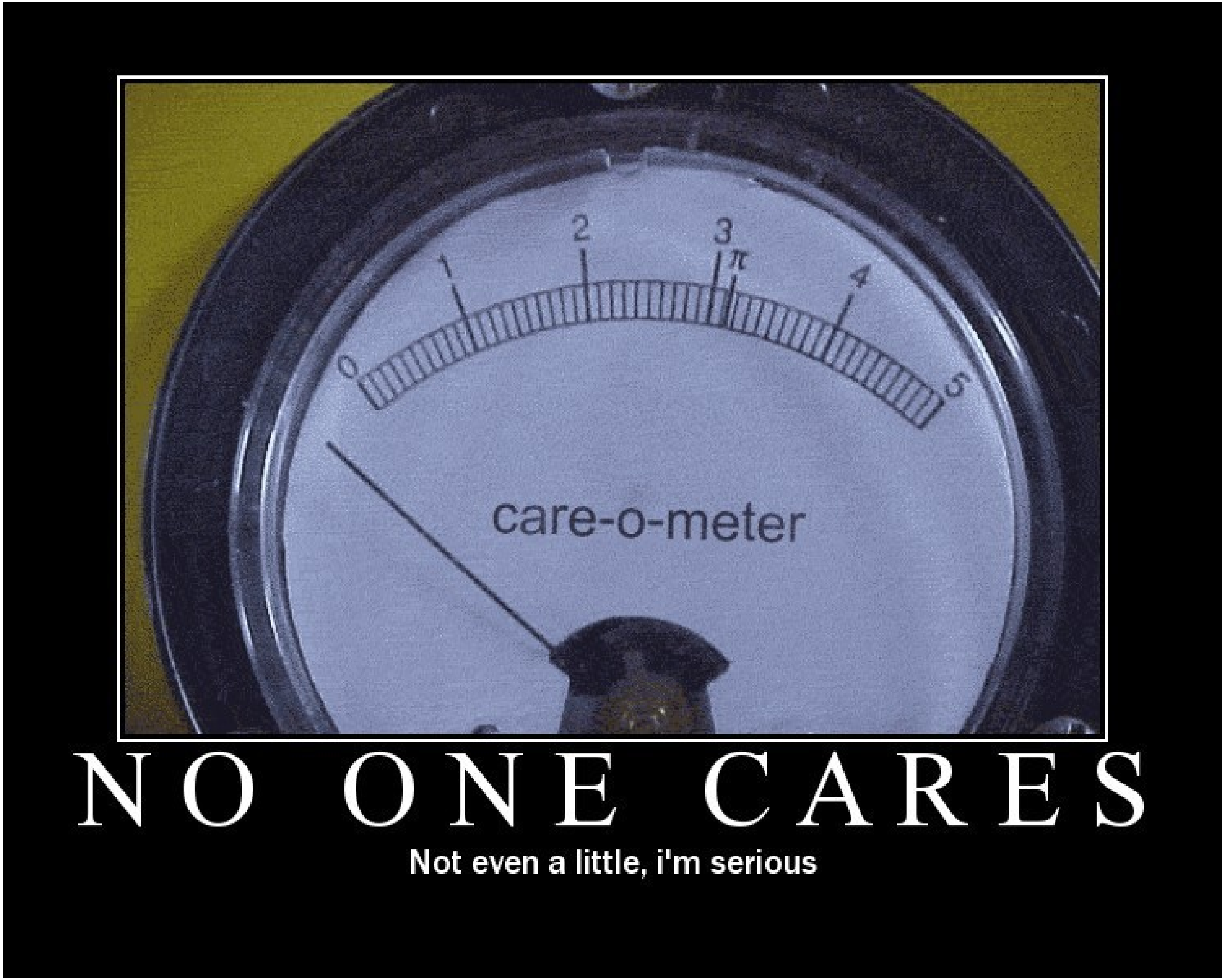
aka one day my son all this will be yours

ecmascript 4



ecmascript 4

stuart langridge, 2009



stuart langridge, 2009

real things

stuart langridge, 2009

Array functions

QUERYSELECTORALL

getters

DOM Local Storage

querySelector

LOCAL DB STORAGE

getComputedStyle

<event-source>

getElementsByClassName

setters

stuart langridge, 2009

getElementsByClassName

```
var elements = document.getElementsByClassName(“class”);
```

just like getElementsByTagName

querySelector, querySelectorAll

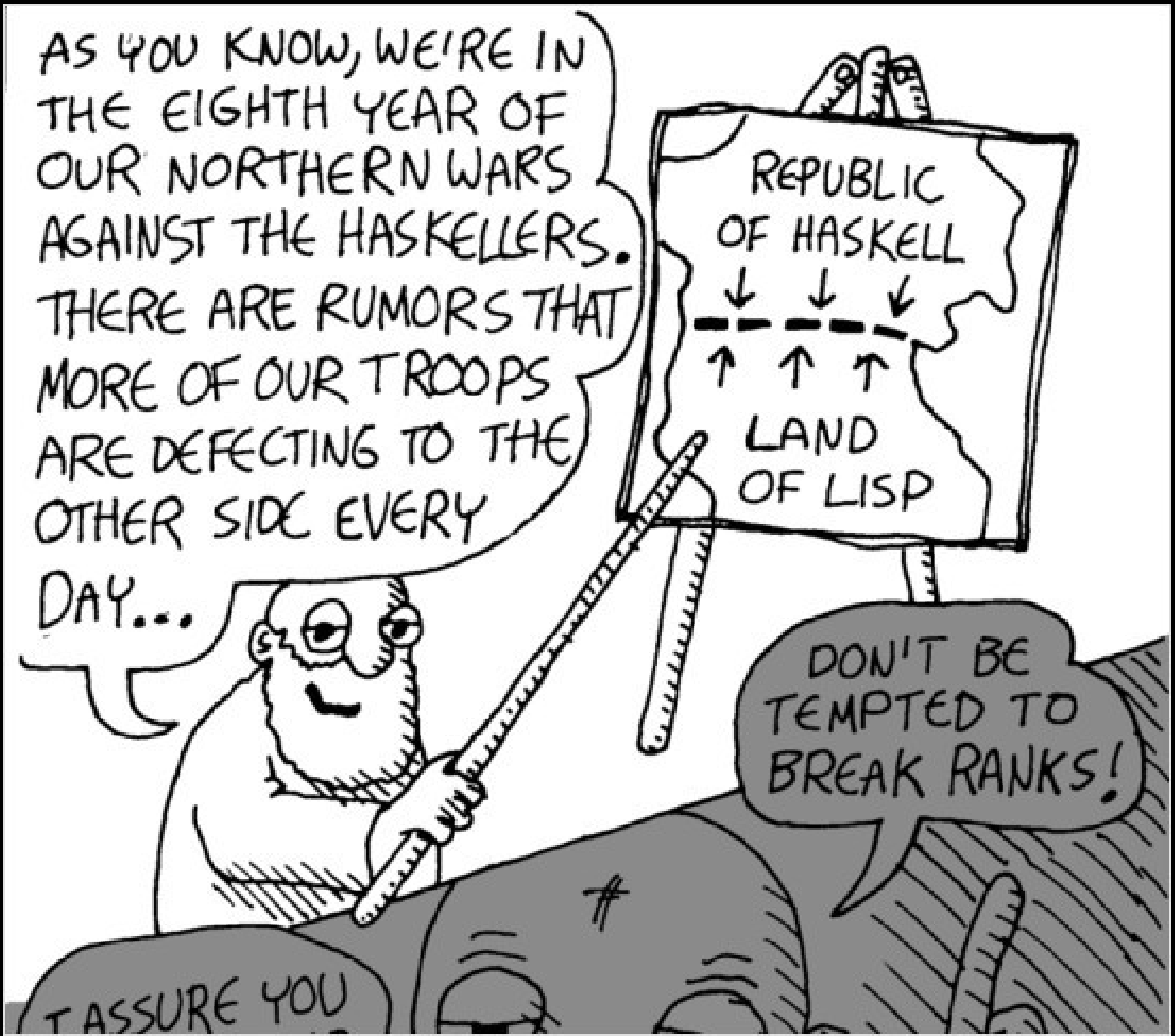
```
var link = document.querySelector(“.nav p a”);

var links = document.querySelectorAll(“.nav p a”);

var l = e.querySelectorAll(“.large:nth-child(even)”);
```

use any CSS selectors you like

Functional programming


AS YOU KNOW, WE'RE IN THE EIGHTH YEAR OF OUR NORTHERN WARS AGAINST THE HASKELLERS. THERE ARE RUMORS THAT MORE OF OUR TROOPS ARE DEFECTING TO THE OTHER SIDE EVERY DAY...
REPUBLIC OF HASKELL
LAND OF LISP
DON'T BE TEMPTED TO BREAK RANKS!
I ASSURE YOU

stuart langridge, 2009

Functional programming

```
not that scary...
```

Array functions

```
[“a”,”b”,”c”].forEach(function(val, idx, arr) {
    item (idx) in array (arr) is (val)
});
```

Array functions

```
[“a”,”b”,”c”].forEach(function(val, idx, arr) {
    item (idx) in array (arr) is (val)
});

[1,2,3].every(function(val, idx, arr) {
    return (val < 4)
})     //  true: see also “some”
```

Array functions

```
[“a”,”b”,”c”].forEach(function(val, idx, arr) {
    item (idx) in array (arr) is (val)
});

[1,2,3].every(function(val, idx, arr) {
    return (val < 4)
})     //  true: see also “some”

[1,2,3,4,5].filter(function(val, idx, arr) {
    return (val < 4)
})     //  [1,2,3]
```

Array functions

```
[“a”,”b”,”c”].forEach(function(val, idx, arr) {
    item (idx) in array (arr) is (val)
});

[1,2,3].every(function(val, idx, arr) {
    return (val < 4)
})     //  true: see also “some”

[1,2,3,4,5].filter(function(val, idx, arr) {
    return (val < 4)
})     //  [1,2,3]

[1,2,3,4,5].map(function(val, idx, arr) {
    return (val + 6)
})     //  [7,8,9,10,11]: see also “reduce”
```

Array functions

```
[“a”,”b”,”c”].forEach(function(val, idx, arr) {
    item (idx) in array (arr) is (val)
});

[1,2,3].every(function(val, idx, arr) {
    return (val < 4)
})     //  true: see also “some”

[1,2,3,4,5].filter(function(val, idx, arr) {
    return (val < 4)
})     //  [1,2,3]

[1,2,3,4,5].map(function(val, idx, arr) {
    return (val + 6)
})     //  [7,8,9,10,11]: see also “reduce”

[“a”,”b”,”c”].indexOf(“b”)   // 1
```

getComputedStyle

```
var el = document.querySelector(“p”);

var css = document.defaultView.getComputedStyle(el,null);

var height = css.getPropertyValue("height");
```

getComputedStyle

```
var el = document.querySelector(“p”);

var css = document.defaultView.getComputedStyle(el,null);

var height = css.getPropertyValue("height");
```

but...

```
el.style.height
```

getComputedStyle

```
var css = document.defaultView.getComputedStyle(el,null);

var height = css.getPropertyValue("height");
```

just not necessary

runtimeStyle in IE

```
var el = document.getElementsByTagName(“p”)[0];

var height = el.runtimeStyle.height;

el.runtimeStyle.cssText = “height: 200px;”;
```

Internet Explorer ftw!

getters and setters

```
var dt = Date.prototype;
dt.__defineGetter__(“sinceIWasBorn”, function() {
    var silborn = Date.parse(“30/01/1976”);
    return this.getTime() - silborn;
});

var now = new Date();
alert(now.sinceIWasBorn); // property, not method
```

getters and setters

```
var dt = Date.prototype;
dt.__defineGetter__(“sinceIWasBorn”, function() {
    var silborn = Date.parse(“30/01/1976”);
    return this.getTime() - silborn;
});

var obj = {
  a: 100,
  get b(): { return this.a + 1 }
  set c(val): {
    this.a = val + 50;
  }
};
```

DOM local storage

```
var visits = globalStorage[“kryogenix.org”].visits;

if (!visits) visits = 0;

globalStorage[“kryogenix.org”].visits = visits + 1;
```

IE has userData behaviours

local database storage

```
var db = openDatabase("My Database", "1.0");

db.executeSql("SELECT * FROM test", function(result1) {
   // do something with the results

   db.executeSql("DROP TABLE test", function(result2) {
     // do some more stuff

     alert("The second query has finished");
   });
});
```

WebKit only (Mozilla in XPCOM)

no, you can’t have a consistent database API

server-sent events

PHP on the server:

```
header("Content-Type: application/x-dom-event-stream");
while(true) {
    echo "Event: server-time\n";
    $time = time();
    echo "data: $time\n";
    echo "\n";
    flush();
    sleep(3);
}
```

server-sent events

```
header("Content-Type: application/x-dom-event-stream");
while(true) {
    echo "Event: server-time\n";
    $time = time(); echo "data: $time\n";
    echo "\n"; flush();
    sleep(3);
}
```

```
<event-source src=”getTime.php”>

document.getElementsByTagName("event-source")[0]
   .addEventListener("server-time", eventHandler, false);

function eventHandler(event) {
    alert(event.data);
}
```

Opera only

I know what you're thinking... where can I get these fine new items?

stuart langridge, 2009

querySelector
array functions
server-sent events
getters and setters
DOM local storage
getComputedStyle
local database storage
getElementsByClassName

querySelector

array functions

~~server-sent events~~ (opera only)

getters and setters

~~DOM local storage~~ (not webkit)

getComputedStyle

~~local database storage~~ (webkit only)

getElementsByClassName

stuart langridge, 2009

~~querySelector~~

~~array functions~~

~~server-sent events~~

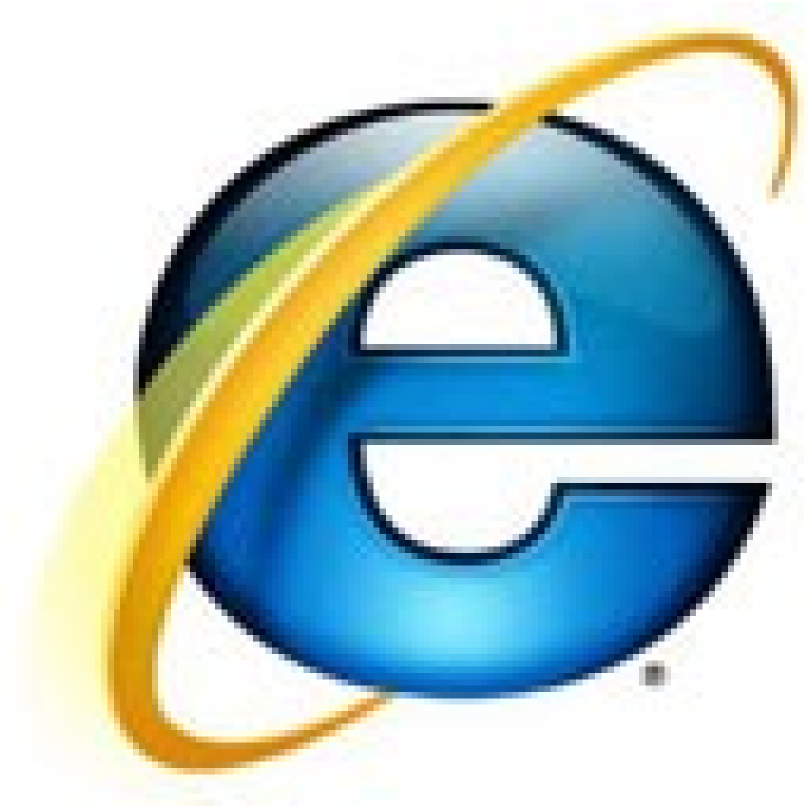
~~getters and setters~~

~~DOM local storage~~

~~getComputedStyle~~

~~local database storage~~

~~getElementsByClassName~~



stuart langridge, 2009

EPIC FAIL

~~querySelector~~

~~array functions~~

~~server-sent events~~

~~getters and setters~~

~~DOM local storage~~

~~getComputedStyle~~

~~local database storage~~

~~getElementsByClassName~~

stuart langridge, 2009

but, but, but...

IE8!

querySelector

~~server-sent events~~
~~getters and setters~~

~~local database storage~~

stuart langridge, 2009

with a library

querySelector

array functions

~~server-sent events~~

~~getters and setters~~

DOM local storage

getComputedStyle

~~local database storage~~

getElementsByClassName

single browser environment

progressive enhancement

get the skills

stuart langridge, 2009

party like it’s 2009

stuart langridge, 2009

fin.

http://tinyurl.com/whizzy

thanks to Conrad Barski

stuart langridge, 2009