DANIEL MIKESCH

JAVA SCRIPT 101

JAVA SCRIPT

- since 1995
- Entworfen von Brenden Eich
- in 10 Tagen für Netscape
- Konzepte aus Scheme und Self
- Syntax abgeleitet aus C und Java
- MS JScript in IE 3 in 1996
- Standardisiert in ECMA Script

JavaScript is a high-level, dynamic, untyped, and interpreted programming language.

. . .

supporting object-oriented and functional programming . . .

Wikipedia

Wien

Neusiedler See

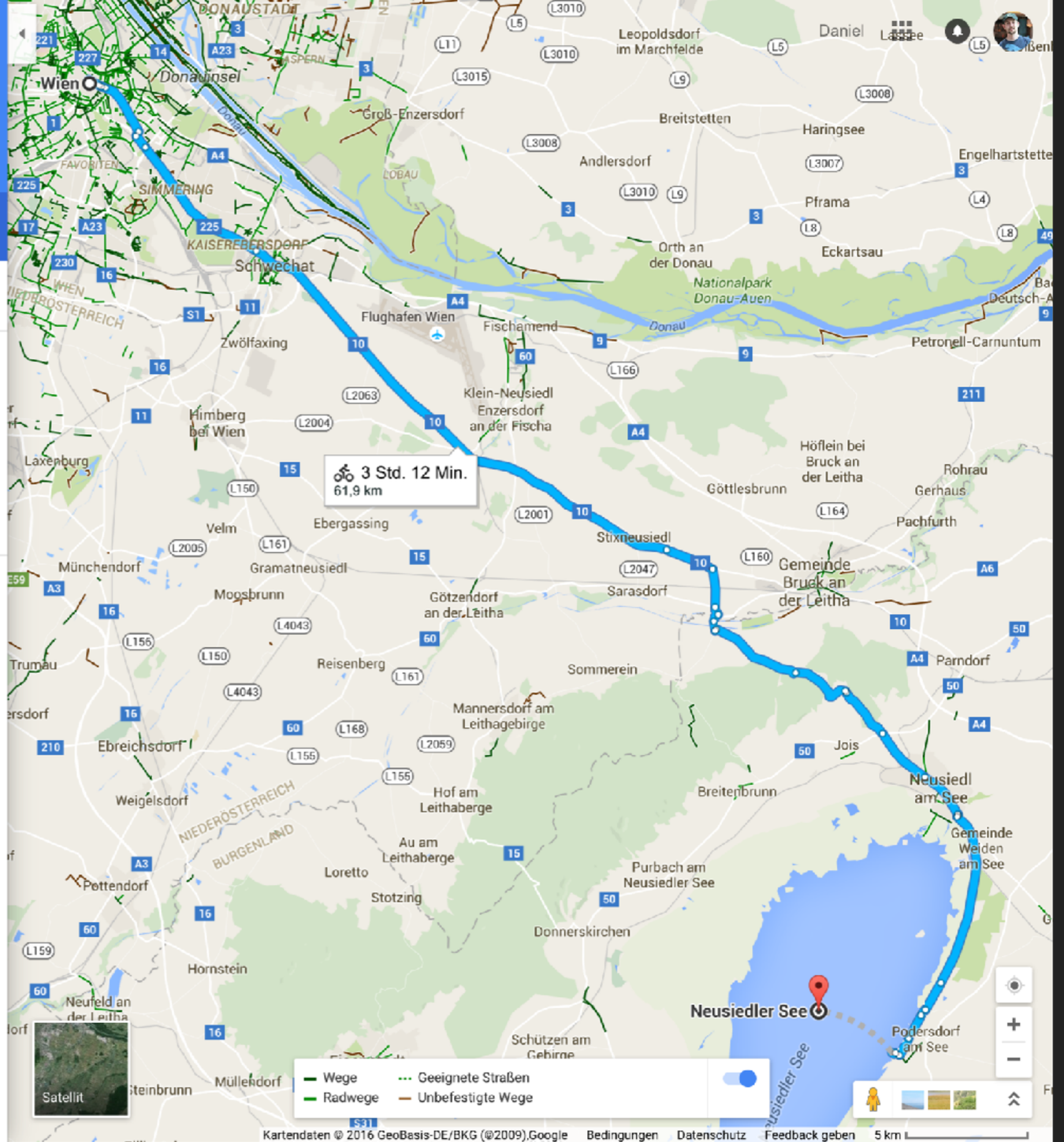
OPTIONEN

Wegbeschreibung auf mein Smartphone senden

via B10 — 3 Std. 12 Min.

↑ 148 m · ↓ 208 m — 61,9 km

DETAILS

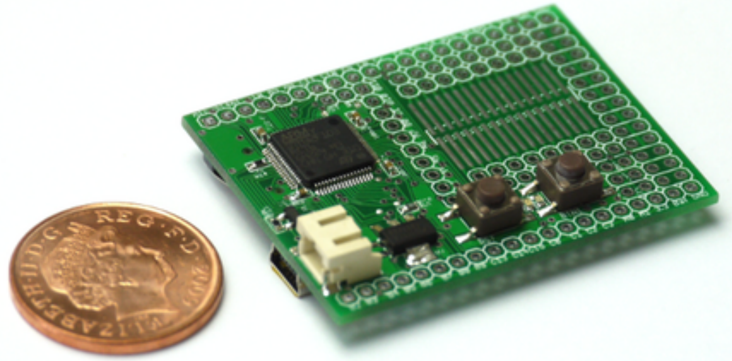



GO!
Tappy to Flappy
Tappy Chicken
2


1750
HTML
5

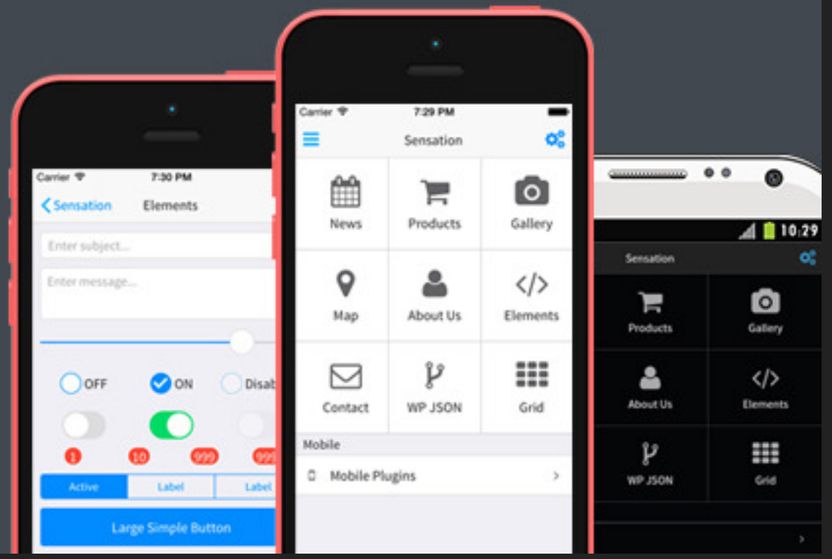
Espruino

JavaScript for Things



Sensation

PhoneGap/Cordova
Full Hybrid App with
AngularJS + Onsen UI



webserver.js

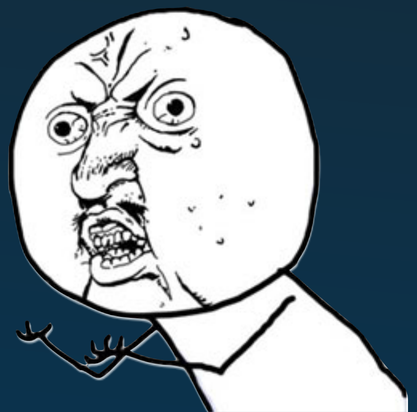
```
1  var express = require('../node_modules/express'),
2      redis = require('../node_modules/socket.io/node_modules/redis'),
3      // io = require('../node_modules/socket.io').listen(app),
4      fs = require('fs');
5
6  var app = express.createServer();
7  app.use(express.bodyParser());
8
9  // var cache = redis.createClient(6379, "127.0.0.1");
10 var cache = redis.createClient(7010, "192.168.100.18");
11
12 var port = 8888;
13
14 // get html page ok
15 app.get('/*', function (req, res) {
16     req.setMaxListeners(0);
17     res.sendfile(__dirname + '/html/' + req.params[0]);
18 });
19
20 // get js page ok
21 app.get('/js/*', function (req, res){
22     res.sendfile(__dirname + '/html/js/' + req.params[0]);
23 });
24
25 // get image ok
26 app.get('/img/*', function (req, res){
27     res.sendfile(__dirname + '/html/img/' + req.params[0]);
28 });
29
30 app.listen(port);
```

node.js

JavaScript ist eine der populärsten Programmiersprachen 😍

Aber auch eine der am meisten gehassten 🙃

JAVASCRIPT U LOOK LIKE C
Y U NO BEHAVE LIKE C !!? 😡



C

```c
/* Code segment 11*/
int curDist, newDist;
int newU1[100], newU2[100];

curDist = Dist(U);
do {
     int i;
     oldDist = curDist;
     for (i=1; i<=K+1; i = i+ 1) {
          newU1 = U + F[i];
          newU2 = U - F[i];
          newDist = Dist(newU1);
          if (newDist == 0 || newDist < curDist) {
               curDist = newDist;
               U = newU1;
          }
          newDist = Dist(newU2);
          if (newDist == 0 || newDist < curDist) {
               curDist = newDist;
               U = newU2;
          }
     }
} while (curDist > 0 && curDist < oldDist);
```

LISP

```lisp
(in-package :cl-mw.examples.hello-world)
(define-mw-master (argv)
    (unwind-protect
         (let ((num-tasks 10)
               (num-results 0))
           (dotimes (x num-tasks)
             (let ((str (format nil "Task ~A" x)))
               (mw-funcall-hello (str))))
           (while (/= num-results num-tasks)
             (mw-master-loop)
             (when-let ((results (mw-get-results)))
               (dolist (result results)
                 (let ((payload (mw-result-packet
                                 result)))
                   (incf num-results)
                   (format t "Got result from slave:
                              ~S~%"
                           payload)))))
           0)
      (format t "Master algo cleanup form.~%")))
```


Activate Your Web Pages
6th Edition
JavaScript
The Definitive Guide
David Flanagan
O'REILLY®
Unearthing the Excellence in JavaScript
JavaScript:
The Good Parts
O'REILLY® | YAHOO! PRESS
Douglas Crockford

WE HAVE COOKIES!

AND HTML5 OFFLINE-STORAGE

WHITESPACE ?

VARIABLEN?

VARIABLEN UND FUNKTIONSNAMEN

- bestehen aus Buchstaben und Ziffern
- keine reservierten Wörter
- beginnen mit einem Buchstaben (oder $ _)
- Groß - Kleinschreibung ist wichtig!

RESERVIERTE WÖRTER

abstract
arguments
boolean
break
byte
case
catch
char
class
const
continue
debugger
default
delete
do
double
else
enum
eval
export
extends
false
final
finally
float
for
function
goto
if
implement
super
import
in
instanceof
int
interface
let
long
native
new
null
package
private
protected
public
return
short
static
super
switch
synchronized
this
throw
throws
transient
true
try
typeof
var
void
volatile
while
with
yield

DATENTYPEN ?

DATENTYPEN

Boolean (true/false)

Number

Strings

Function

Array

Object

RegExpr

undefined

null

```
var isComplicated = false;
var theUltimateAnswer = 42;
var greetings = “Hello World”;
var foo = function(){ console.log(x) };
function foo(){ console.log(x) }
var stuff = [“A”, 42, “Yess”, 3.14];
var car = {color:”red”, drive: function(){}};
var regex = /^[a-z0-9]$/;
var nochnix;
var empty = null;
```

UNDEFINED VS NULL == VS === != VS !==

```
typeof undefined                //undefined
typeof null                     //object

null === undefined              //false
null == undefined               //true
```

== THE EVIL TWIN OF ===

```
'' == '0'    // false
0 == ''      // true
0 == '0'    // true

false == 'false'// false
false == '0'  // true

false == undefined // false
false == null      // false
null == undefined  // true

' \t\r\n ' == 0    // true
```

KONTROLLSTRUKTUREN?

JS IN HTML

```
<script>alert(“Hallo!”);</script>

<script src=“main.js”></script>

<button onclick=“alert(‘you clicked :D’);”>
  Click me!
</button>

<a href=“javascript:alert(’you clicked :D’)”>
  Click this link
</a>
```

“HELPER OBJECTS”

- Date `var myDate = new Date(); myDate.getDay();`
- Math `Math.PI  Math.abs(3.2)  Math.random()`
- JSON `JSON.strigify( someVariable );`

“BROWSER API OBJECTS”

- Document `var el = document.getElementById(“id”);`
- Window `window.location.href =  “http://google.com”;`

```
<html>
  <head>
    <title>My Doc</title>
  </head>
  <body>
    <h1>Header</h1>
    <p>Paragraph</p>
  </body>
</html>
```