

JavaScript: The Good Parts vs. JavaScript: The Definitive Guide



For next class, read <http://eloquentjavascript.net/> chapters 1-4.

CS 152: *Programming Language Paradigms*



JavaScript

Prof. Tom Austin

San José State University

History of JavaScript



Brendan Eich

1995: Netscape hired
Brendan Eich.

His job: implement Scheme for
the web browser.

After a few meetings,
Scheme was deemed too weird...

**In 10 days, Brendan Eich wrote
the initial version of JavaScript
for Netscape 2.0 Beta.**

JavaScript

- *Superficially* similar to Java
- Primarily client-side programming
- Server-side variants:
 - JVM: Rhino & Nashorn
 - Node.js
- <http://w3schools.com/js/default.asp>

JavaScript is *multi-paradigm*:

- *Imperative*
- *Functional* – "Scheme in C's clothing"
- *Object-oriented* – Prototype-based

Imperative JavaScript

```
function addList(list) {  
    var i, sum=0;  
    for (i=0; i<list.length; i++) {  
        sum += list[i];  
    }  
    return sum;  
}
```

Functional JavaScript

```
var addList = function(list) {  
  if (list.length === 0) {  
    return 0;  
  }  
  return list[0]  
    + addList(list.slice(1));  
}
```

Object-Oriented JavaScript

```
function Adder (amount) {  
    this.amount = amount;  
}  
Adder.prototype.add = function(x) {  
    return this.amount + x;  
}  
var myAdder = new Adder(1);  
var y = myAdder.add(7);
```

Extended JavaScript Examples

(in-class)

Introduction to Node.js



Node.js

- Server-side JavaScript
- Based on Google's V8 engine
- npm: Node.js package manager
- <http://nodejs.org/>



myFile.txt

This is my file.

There are many like it,
but this one is mine.

File I/O in Node.js

Callback
function

```
var fs = require('fs');  
fs.readFile('myFile.txt',  
function(err, data) {  
    if (err) throw err;  
    console.log(""+data);  
});  
console.log('all done');
```

Resulting Output

all done

This is my file.

There are many like it,
but this one is mine.

Synchronous File IO in Node

```
var data = fs.readFileSync(  
    './myFile.txt');  
console.log(data.toString());  
console.log('all done');
```

Lab: Intro to JavaScript

Today's lab explores both the functional and object-oriented aspects of JavaScript.

See Canvas for details.