Beginning Programming in JavaScript

Clarissa Littler

April 2, 2016

What You'll Learn Here

- What programming is

What You'll Learn Here

- What programming is
- The basics of Javascript

What You'll Learn Here

- What programming is
- The basics of Javascript
- What to learn next

This Course

- Three weeks

This Course

- Three weeks
- Two hours each

This Course

- Three weeks
- Two hours each
- Lecture and small exercises

This Course

- Three weeks
- Two hours each
- Lecture and small exercises
  - Piggybacking on Code Academy

Why Programming?

- Jobs

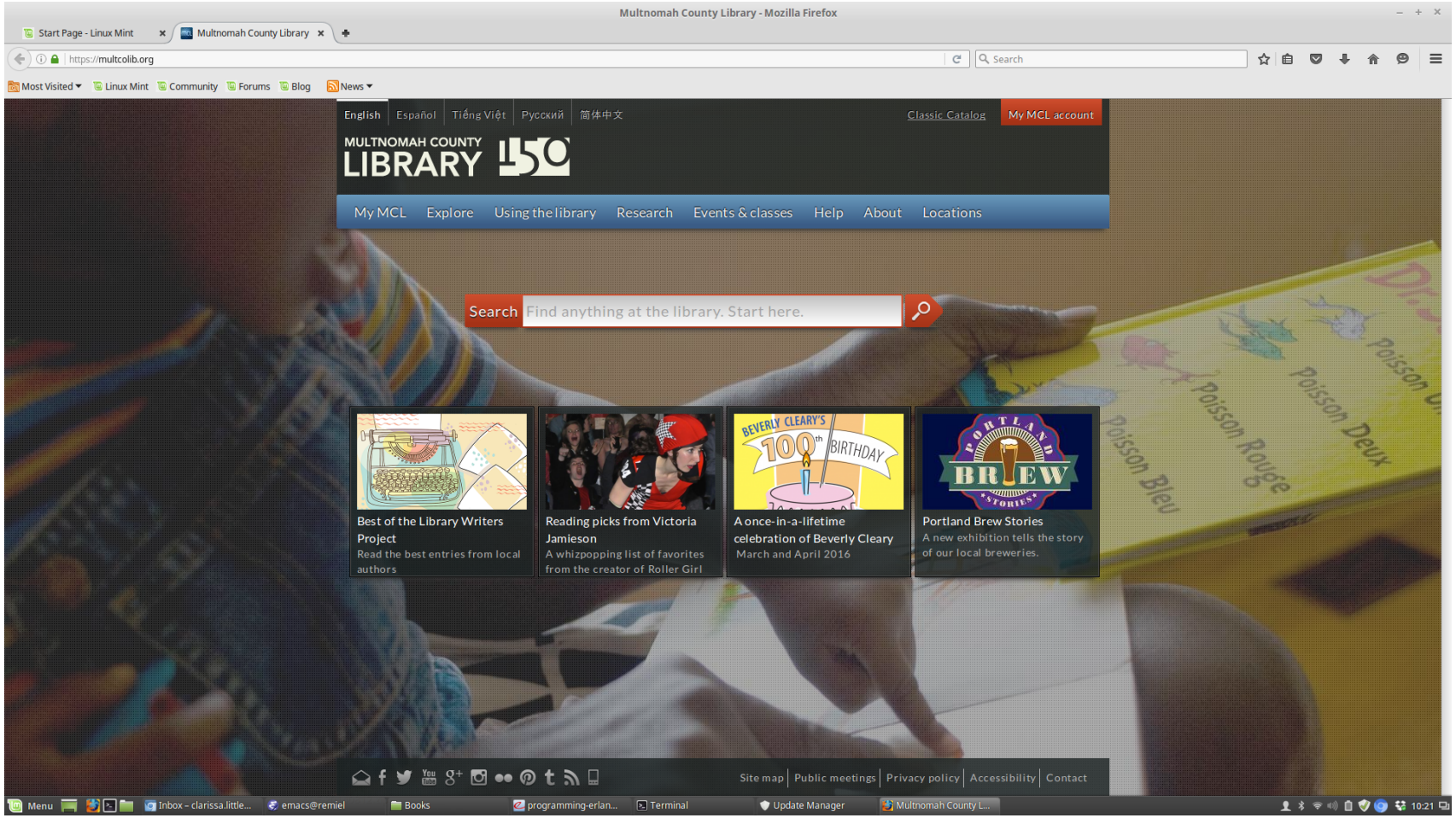
Why Programming?

- Jobs
- Automation

Why Programming?

- Jobs
- Automation
- Fun

What You Can Do



What You Can Do

What You Can Do

emacs@remiel

File Edit Options Buffers Tools Jabber Org Tbl Egg (Git) Help

Save Undo

```org
#+TITLE: What Are Computers, Really?
#+AUTHOR: Clarissa Littler
#+EMAIL: clarissa.littler@gmail.com
#+DATE: 6-23-2015
#+OPTIONS: H:2
#+BEAMER_THEME: Madrid
#+LATEX_CLASS: beamer

#+startup: beamer
#+LaTeX_CLASS: beamer
#+LaTeX_CLASS_OPTIONS: [bigger]

#+COLUMNS: %45ITEM %10BEAMER_ENV(Env) %10BEAMER_ACT(Act) %4BEAMER_COL(Col) %8BEAMER_OPT(Opt)

* Outline and Goals...
* Computers and Programs
** Computation Sounds Like Computer...
** Examples of Finite Processes
*** Recipe as Finite Process                                  :B_block:...
** Examples of Finite Processes...
** Examples of Finite Processes
file:DirectionsScreenshot.png
** Examples of Finite Processes...
** Examples of Finite Processes...
** Examples of Finite Processes...
** Examples of Finite Processes...
* What Makes a Finite Process Finite...
* Models of Computation...
* Unsolvable Problems...
* The Legacy of Church-Turing...
* Finale...
```

-:--- OSBridgeTalk.org All L67 Git-master (Org Egg Hi Bm Helm)

```org
#+TITLE: Beginning Programming in JavaScript
#+AUTHOR: Clarissa Littler

#+startup: beamer
#+BEAMER_THEME: Madrid
#+LaTeX_CLASS: beamer
#+LaTeX_CLASS_OPTIONS: [bigger]
#+BEAMER_FRAME_LEVEL: 2

* Why Learn Programming?
** What You'll Learn Here
   + What programming *is* \pause
   + The basics of Javascript \pause
   + What to learn next
** This Course
   + Three weeks \pause
   + Two hours each \pause
   + Lecture and small exercises \pause
     + Piggybacking on Code Academy
** Why Programming?
file:website.png
** Why Programming?
file:servers.jpg
** Why Programming?
```

-:--- lecture1.org All L25 Git:master (Org Git:master Hi Bm Helm Wrap)

Wrote /home/clarissa/projects/multcolib-lectures/lecture1.org

Menu | Tumblr - Chromium | emacs@remiel | Books | programming-erlan... | Terminal | Update Manager | Multnomah County L... | multcolib-lectures | 10:43

What You Can Do

Programming Is...

- a way to instruct a computer

Programming Is...

- a way to instruct a computer
- to do *what*?

Programming Is...

- a way to instruct a computer
- to do *what*?
- are there limits?

Computers as people

- Computer as a job title

Computers as people

- Computer as a job title
- Computers calculated as a series of steps

Computers as people

- Computer as a job title
- Computers calculated as a series of steps
- Mostly number crunching

Computers as people

- Computer as a job title
- Computers calculated as a series of steps
- Mostly number crunching
- Computation

Computation

Computation is anything that uses

Computation

Computation is anything that uses

- Finite instructions

Computation

Computation is anything that uses

- Finite instructions
- Finite time

Computation

Computation is anything that uses

- Finite instructions
- Finite time
- Finite resources

More Than Number Crunching

- Computation > arithmetic

More Than Number Crunching

- Computation > arithmetic
- 3D printers

More Than Number Crunching

- Computation > arithmetic
- 3D printers
- Robots

More Than Number Crunching

- Computation > arithmetic
- 3D printers
- Robots
- Computation is physical

Other Examples of Computation

Spinach Celery Soup

Cook celery and onion together til soft, then add frozen spinach and cook to get some of the moisture out and reduce volume add broth lentils cilantro and other spices, stir thoroughly, throw bay leaves on top.
Cook for 40 minutes
Turn off heat, wait til it stops bubbling and blend thoroughly.
Cook for 5-10 minutes after blending

Other Examples of Computations

Many more examples exist in the wild including:

Other Examples of Computations

Many more examples exist in the wild including:

- counting on your fingers

Other Examples of Computations

Many more examples exist in the wild including:

- counting on your fingers
- long division

Other Examples of Computations

Many more examples exist in the wild including:

- counting on your fingers
- long division
- sorting your vinyl collection

Limits of Computation

- Are there things not programmable?

Limits of Computation

- Are there things not programmable?
- Yes!

Limits of Computation

- Are there things not programmable?
- Yes!
  - But not much you care about

Limits of Computation

- Are there things not programmable?
- Yes!
  - But not much you care about
- Programs that analyze programs

Limits of Computation

- Physical limits

Limits of Computation

- Physical limits
- No better computer can save you

Limits of Computation

- Physical limits
- No better computer can save you
  - Not even quantum computers

What are Programs?

Programs are

What are Programs?

Programs are descriptions of computations

What are Programs?

Programs are descriptions of computations in a special language

What are Programs?

Programs are descriptions of computations in a special language that's understandable to a computer

Programming Languages

- Human languages are too complicated

Programming Languages

- Human languages are too complicated
- "y'all'd've"

Programming Languages

- Human languages are too complicated
- "y'all'd've"
- Ever had trouble with directions?

Programming Languages

- Simpler than human ones

Programming Languages

- Simpler than human ones
- Inflexible

Programming Languages

- Simpler than human ones
- Inflexible
- Precise

Programming Languages

- Simpler than human ones
- Inflexible
- Precise
- Harder to think in

Why JavaScript?

- A JavaScript interpreter in every browser

Why JavaScript?

- A JavaScript interpreter in every browser
- JavaScript for servers

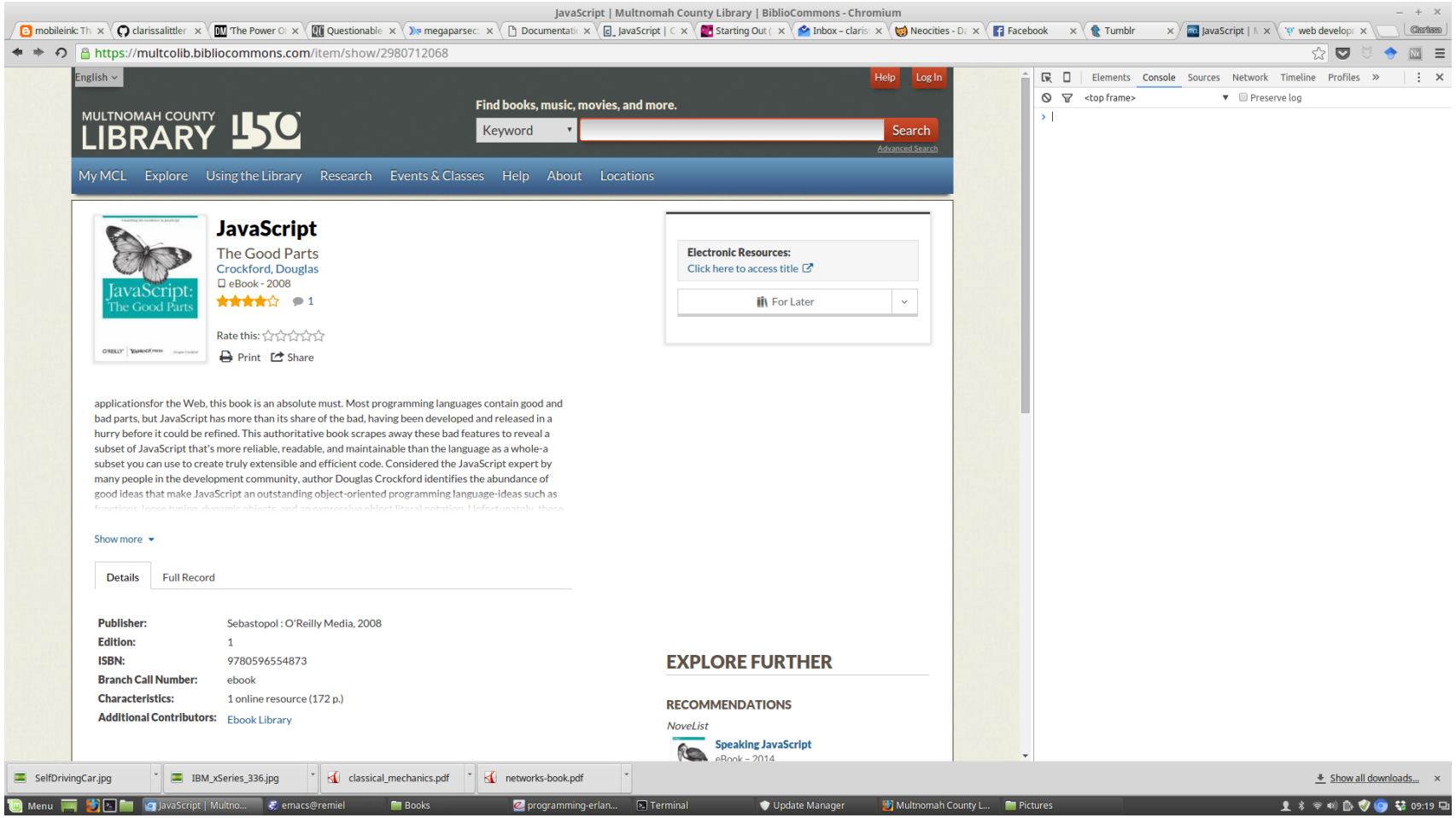
Why JavaScript?

- A JavaScript interpreter in every browser
- JavaScript for servers
- Relatively simple language

Why JavaScript?

- A JavaScript interpreter in every browser
- JavaScript for servers
- Relatively simple language
  - At least the "good parts"

JavaScript Console



Syntax vs. Semantics

- syntax is the grammar of a programming language

Syntax vs. Semantics

- syntax is the grammar of a programming language
- what it does is the *semantics*

Syntax vs. Semantics

- syntax is the grammar of a programming language
- what it does is the *semantics*
- need to be very careful understanding both

Nouns and Verbs

- Human languages have nouns and verbs

Nouns and Verbs

- Human languages have nouns and verbs
- So do computer's languages

Nouns and Verbs

- Human languages have nouns and verbs
- So do computer's languages
- Nouns => data

Nouns and Verbs

- Human languages have nouns and verbs
- So do computer's languages
- Nouns => data
- Verbs => control constructs

Kinds of Data

- Numbers

Kinds of Data

- Numbers
- Text

Kinds of Data

- Numbers
- Text
- Lists

Kinds of Data

- Numbers
- Text
- Lists
- Labeled collections (think dictionaries)

Arithmetic

Numbers In JavaScript

- Numbers

Arithmetic

Numbers In JavaScript

- Numbers
- Arithmetic operations

Arithmetic

Numbers In JavaScript

- Numbers
- Arithmetic operations
  - Addition
  - Multiplication
  - Division
  - Subtraction

Arithmetic Operations

- 20 + 5
- 20 * 5
- 20 / 5
- 20 - 5

Strings

What Are Strings?

- Computers need to manipulate text

Strings

What Are Strings?

- Computers need to manipulate text
- Text between double quotes

Strings

What Are Strings?

- Computers need to manipulate text
- Text between double quotes
  - or single quotes

Strings

What Are Strings?

- Computers need to manipulate text
- Text between double quotes
  - or single quotes

Examples of Strings

- "this is a 'string'"

Strings

What Are Strings?

- Computers need to manipulate text
- Text between double quotes
  - or single quotes

Examples of Strings

- "this is a 'string'"
- 'this is also a "string"'

Strings

What Are Strings?

- Computers need to manipulate text
- Text between double quotes
  - or single quotes

Examples of Strings

- "this is a 'string'"
- 'this is also a "string"'
- 'this isn't a 'string''

Variables

- Variables are in every language
- Two uses of variables in programming

Variables

- Variables are in every language
- Two uses of variables in programming
  - "So I have a friend, let's call her Alice. . . "

Variables

- Variables are in every language
- Two uses of variables in programming
  - "So I have a friend, let's call her Alice. . . "
  - Containers for your stuff

Variables

- Variables are in every language
- Two uses of variables in programming
  - "So I have a friend, let's call her Alice. . . "
  - Containers for your stuff

Making Variables

```
var thisIsAVariable = "and this is its value"
```

Variables

Using Variables

- The value is retrieved by name

Variables

Using Variables

- The value is retrieved by name
- Type `thisIsAVariable`

Variables

Using Variables

- The value is retrieved by name
- Type `thisIsAVariable`

Setting Variables

- `name = value`

Variables

Using Variables

- The value is retrieved by name
- Type `thisIsAVariable`

Setting Variables

- `name = value`
- `thisIsAVariable = 10`

Undefined

- Try typing `var thisVar`

Undefined

- Try typing `var thisVar`
- What's the value of `thisVar`?

Undefined

- Try typing `var thisVar`
- What's the value of `thisVar`?
- `undefined` is the value of not having a value

Undefined

- Try typing `var thisVar`
- What's the value of `thisVar`?
- `undefined` is the value of not having a value
- Like 0 is the value of having no quantity

Using Functions

- Functions in algebra

Using Functions

- Functions in algebra

```
f(x) = x + 2
```

Using Functions

- Functions in algebra

  ```
  f(x) = x + 2
  ```

- Functions in JavaScript

Using Functions

- Functions in algebra

  `f(x) = x + 2`

- Functions in JavaScript
  - Can do more than crunch numbers

Using Functions

- Functions in algebra

  `f(x) = x + 2`

- Functions in JavaScript
  - Can do more than crunch numbers
  - Change variables

Using Functions

- Functions in algebra

  `f(x) = x + 2`

- Functions in JavaScript
  - Can do more than crunch numbers
  - Change variables
  - Print output

Using Functions

- Functions in algebra

  ```
  f(x) = x + 2
  ```

- Functions in JavaScript
  - Can do more than crunch numbers
  - Change variables
  - Print output
  - Render HTML

Using Functions

- Functions in algebra

  ```
  f(x) = x + 2
  ```

- Functions in JavaScript
  - Can do more than crunch numbers
  - Change variables
  - Print output
  - Render HTML
  - ...

Using Functions

- Functions in algebra

  `f(x) = x + 2`

- Functions in JavaScript
  - Can do more than crunch numbers
  - Change variables
  - Print output
  - Render HTML
  - ...
- `console.log("message")`

Sequencing Operations

Multiple steps are separated by semicolons on separate lines

```
console.log("this");
console.log("is");
console.log("a");
console.log("sequence");
console.log("of");
console.log("steps");
```

Making Choices

- If it's raining, take an umbrella

Making Choices

- If it's raining, take an umbrella
- If you see the purple Little Free Library, take a right, otherwise keep walking

Making Choices

- If it's raining, take an umbrella
- If you see the purple Little Free Library, take a right, otherwise keep walking
- If the username and password match, log the user in

Choices in Programming

- if-statements

Choices in Programming

- if-statements
- Need to formalize truth

Choices in Programming

- if-statements
- Need to formalize truth
- Boolean data

Choices in Programming

- if-statements
- Need to formalize truth
- Boolean data
  - `true`
  - `false`

If-Statement

```
if (2 < 5) {
    console.log("two is less than five");
}
```

If-Statement

```
if (2 === 5) {
    console.log("two is equal to five");
}
else {
    console.log("two wasn't equal to five");
}
```

Code Academy

- Break from lecturing

Code Academy

- Break from lecturing
- I'll guide you through difficulties

Code Academy

- Break from lecturing
- I'll guide you through difficulties
- If you're bored, just say so!

Next Time

- How to change the flow of a program:

Next Time

- How to change the flow of a program: iteration and loops

Next Time

- How to change the flow of a program: iteration and loops
- How to write functions

Next Time

- How to change the flow of a program: iteration and loops
- How to write functions
- A mini-lecture on how web browsers and web servers communicate

Next Time

- How to change the flow of a program: iteration and loops
- How to write functions
- A mini-lecture on how web browsers and web servers communicate
- More Code Academy!