CS+Social Good - CS 106 Section

Section 0

1. Introductions

2. Why Section?

3. Class Information

4. Web Overview

5. JavaScript Overview

6. Tutorial

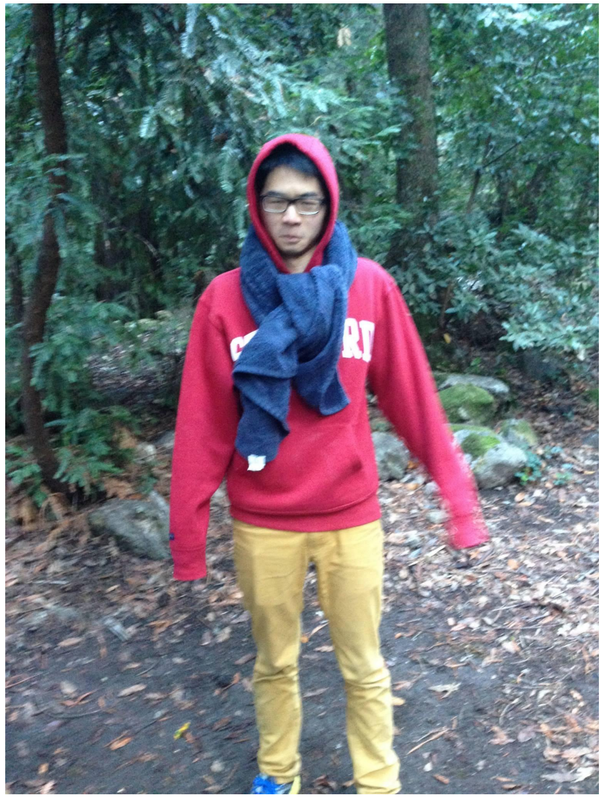
Team

Vicki Niu

Instructor
Stanford ’18
vniu@stanford.edu

Priya Ganesan

Instructor
Stanford ‘17
priyag@stanford.edu



Lawrence Lin Murata

Instructor
Stanford ‘17
lmurata@stanford.edu

Why Section?

Why Section?


You personally are obligated to donate any and all "excess" money
(1) Do all individuals share an equal obligation to donate?
Consider Peter Singer's ethical dilemma
PROPOSITION 3:
STANFORD


CS+SOCIAL GOOD
STANFORD
PRE EDUCATION
SOCIETY
(SPEES)

About This Class

Logistics

- Team: Priya Ganesan (priyag@stanford.edu), Vicki Niu (vniu@stanford.edu), Lawrence Lin Murata (lmurata@stanford.edu)

- Time: 3:30 pm - 5:20 pm, every Wednesday

- Duration: 9 weeks

- Credit: 1 unit of CS 199P under Jerry Cain

- Grading:

    Attendance (8/9 lectures)
    Participation
    Final reflection

Class Timeline

- Week 2 (Today): Overview, Introduction to HTTP and JavaScript
- Week 3: Coding workshop/exercises (semantic analysis)
- Week 4: Google (mapping)
- Week 5: HealthTap (health)
- Week 6: Google (air quality)
- Week 7: CareerVillage (matching students to mentors)
- Week 8: Google (cultural heritage preservation)
- Week 9: Microsoft (VR)
- Week 10: Party!

Bring your laptops to class every week!

Overview of Web

Frontend vs Backend


WEB DESIGN?
THERE'S SPIDERS IN THE BACK YARD YOU CAN EXAMINE
quickmeme.com


I was a web developer
before the Internet.

WHEN YOU VISIT APPLE.COM...

Slides Courtesy : CS 42

Frontend vs. Backend

Client
Your Computer

Server
Apple.com

Frontend vs. Backend

Client
Your Computer

HTTP Request

Server
Apple.com

Frontend vs. Backend

Client
Your Computer

HTTP Request

Response data

Server
Apple.com

Client-side

Client
Your Computer

- on client's machine

Client-side

Client
Your Computer

- on client's machine
- front-end

Client-side

Client
Your Computer

- on client's machine
- front-end
- requests to a server

Client-side

Client
Your Computer

- on client's machine
- front-end
- requests to a server
- processes, manipulates and/or renders data from response

Client-side

Client
Your Computer

- on client's machine
- front-end
- requests to a server
- processes, manipulates and/or renders data from response

Server-side

Server

Apple.com

- run on server

Server-side

Server

Apple.com

- run on server
- back-end

Server-side

Server

Apple.com

- run on server
- back-end
- handles requests from a client and sends response data

Server-side

Server

Apple.com

- run on server
- back-end
- handles requests from a client and sends response data
- often interacts with a database

Server-side

Server

Apple.com

- run on server
- back-end
- handles requests from a client and sends response data
- often interacts with a database

Client vs. Server

Client

Your Computer

Client-side code

1) sends HTTP request

GET /

2) renders response data

Response data

Server

Apple.com

Server-side code

1) handles HTTP request

2) generates/sends response

HTTP Requests

Common request types: GET, POST, PUT, DELETE

We’ll focus on GET and POST

HTTP Examples

HTTP Get Request:

Viewing web pages

Making searches (e.g. Google, Bing)

Accessing files (PDFs, movies, PowerPoints, etc.)

HTTP Post Request:

Submitting forms

Creating/updating new posts/comments (e.g. Facebook)

Registering a user (e.g. Twitter)

Time to learn some JavaScript!


JAVASCRIPT?
SOUNDS LIKE SOME STRONG
COFFEE
DIYLOL.COM

JavaScript Variables

```
var count;
count = 42;

// A variable can be a number or a string or a boolean.
var valleyQuote = "Our startup allows you to share texts
with each other and it’s going to change the world
forever.";

var precision = 0.42;
var condition = true;
```

JavaScript Strings

```
var phoneNumber = "(650) 456-7890";

var name = "Benedict Cumberbatch";
// is the same as:
var name = 'Benedict Cumberbatch';

var firstName = 'Benedict';
var lastName = 'Cumberbatch';
firstName + ' ' + lastName; // 'Benedict Cumberbatch'
```

Equality

```
var socialGoodSectionRocks = true;
var socialGoodSectionSucks = false;
```

New `===` operator, which compares **type and value**:

```
3 == 3;     // true
3 === 3;    // true
3 == '3';   // true;
3 === '3'; // false; types are different
''     ==   '0';     // false
0      ==   '';      // true
'0'    ==   0;       // true
false  ==   '0';     // true
```

Conditionals and Loops

```
if (condition) {
    // do something
}

for (var i = 0; i < 10; i++) {
    // do something 10 times
}

while (condition) {
    // do something while condition holds true
}
```

Arrays

```
var numbers = [1, 2, 3];
var moreNumbers = ["four", "five", "six", "seven"];

numbers[0]; // 1
numbers.length; // 3

// adds 4 to the end of the array
numbers.push(4); // returns new length, 4

numbers.pop(); // removes 4 and returns it
```

Functions

```
// note the lack of types for parameters:
function add(a, b) {
    return a + b;
}

add(5, 10); // returns 15
```

Objects

```
var object = {};
object.key = 'value';
object['key'] = 'value';

var sectionStaff = [
    { name: 'Priya',
      status: "I got a blank space,
                and I'll write your name" },
    { name: 'Lawrence',
      status: "I'm too hot (hot damn)" },
    { name: 'Vicki',
      status: "I am fresher than you." },
];
```

Objects

```
var object = {};
object.key = 'value';
object['key'] = 'value';
```

```
var sectionStaff = [
    { name: 'Priya',
      status: "I got a blank space,
                and I'll write your name" },
    { name: 'Lawrence',
      status: "I'm too hot (hot damn)" },
    { name: 'Vicki',
      status: "I am fresher than you." },
];
```

Objects

```
var object = {};
object.key = 'value';
object['key'] = 'value';

var sectionStaff = [
    { name: 'Priya',
      status: "I got a blank space,
               and I'll write your name" },
    { name: 'Lawrence',
      status: "I'm too hot (hot damn)" },
    { name: 'Vicki',
      status: "I am fresher than you." },
];
```

Objects

```
var object = {};
object.key = 'value';
object['key'] = 'value';

var sectionStaff = [
    { name: 'Priya',
      status: "I got a blank space,
                and I'll write your name" },
    { name: 'Lawrence',
      status: "I'm too hot (hot damn)" },
    { name: 'Vicki',
      status: "I am fresher than you." },
];
```

Objects

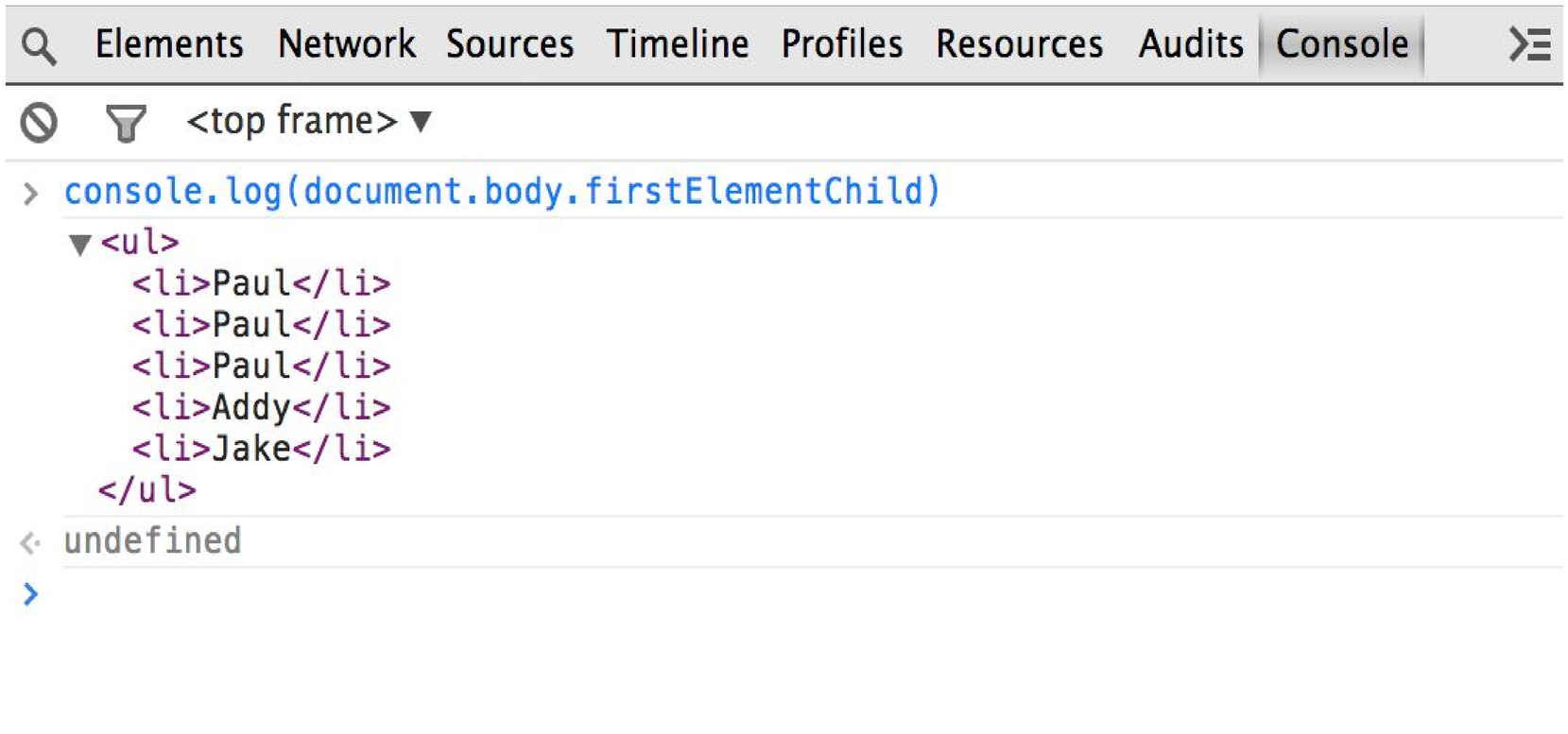
```
var object = {};
object.key = 'value';
object['key'] = 'value';

var sectionStaff = [
    { name: 'Priya',
      status: "I got a blank space,
               and I'll write your name" },
    { name: 'Lawrence',
      status: "I'm too hot (hot damn)" },
    { name: 'Vicki',
      status: "I am fresher than you." },
];
```

Console

Chrome: **Right click → Inspect Element** or **command + alt + i**

Firefox / Safari / IE / Microsoft Edge : Same as Chrome but first download Chrome.



Console

```
console.log('This is a log message.')
// you should see 'This is a log message.' in your console
```

```
console.log('Some other message.', 5 * 3);
// you should see 'Some other message.', 15 in your console
```

```
var numbers = [2, 4];
numbers.push(6);
console.log(numbers);
// you should see '[2, 4, 6]' in your console
```

Let’s write a political speech

```
// Step 1: Be vague.
function start() {
    var intro = "This is a very important topic.\n";
    intro += "It’s a real problem. We have to do something
about it \n";
    return intro;
}
```

Let’s write a political speech

```
function sayItAgain() {
    // Step 2: Say It Again.

}
```

Let’s write a political speech

```
function sayItAgain() {
    // Step 2: Say It Again.
    var repeat = "What will I do about this problem? We need
    to think about this. \n";

    // Step 3: Get as close to saying something logical
    // without actually doing so.
}
```

Let’s write a political speech

```
...
    // Step 3: Get as close to saying something logical
    // without actually doing so.
    for (var i = 0; i < 50; i++) {
      repeat += "This is the biggest problem our great country
faces today. We should build a wall.";
    }

    return repeat;
}
```

Let’s write a political speech

```
function giveASpeech() {
    return start() + sayItAgain();
}
```

Tutorial

Dynamic String Replacement with JavaScript

Tutorial

1. Go to the Nuclear Physics wikipedia page: https://en.wikipedia.org/wiki/Nuclear_physics

2. Open the console on your browser

3. Paste the following code

document.documentElement.innerHTML = document.documentElement.innerHTML.replace(/[nN]uclear/g, "butt");

Tutorial

4. Open a new tab and go to http://larry.lawrencemurata.com/
5. Paste the following code on your console. You should replace the string “Hello” with the new text from the Wikipedia page

responsiveVoice.speak("Hello");

CS 106 Social Good Section

Priya Ganesan
Vicki Niu
Lawrence Lin Murata