

PROGRAMMING FOR NON PROGRAMMERS

Jeremiah Alexander | Lead Instructor | WDI

Learning Objectives



Learn the technical vocabulary
to communicate effectively
with developers



Program your own basic
websites in HTML/CSS



Think like a programmer and
write code in Javascript

Schedule & Lesson Format

16/11 - Part 1	Programming the World Wide Web
17/11 - Part 2	Building Websites with HTML & CSS
23/11 - Part 3	Adding Interactivity with Javascript
24/11 - Part 4	Deploying Websites using Github

Each Lesson

7:00pm - 7:30pm	Introductions / Recap
7:30pm - 8:00pm	Lecture Topic 1
8:00pm - 8:30pm	Lab Topic
8:30pm - 9:00pm	Lecture Topic 2
9:00pm - 9:30pm	Lab Topic

PROGRAMMING FOR NON PROGRAMMERS

INTRODUCTIONS

GOALS & RULES

PROGRAMMING FOR NON PROGRAMMERS

PART 1

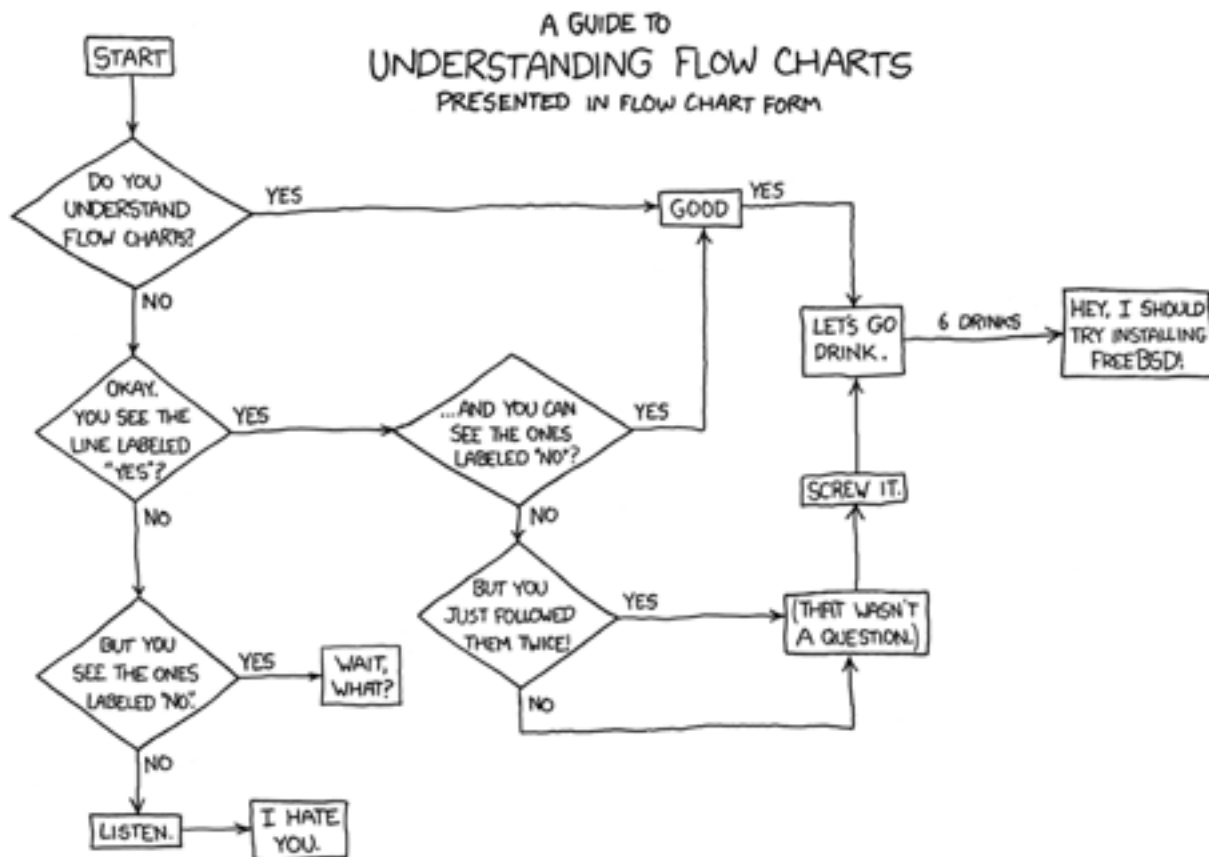
PROGRAMMING THE WORLD WIDE WEB

Learning Objectives

- Define what Programming is
- Give examples of different Programming Languages
- Describe how the Web work
- Identify the stages of Web Development

PROGRAMMING FOR NON PROGRAMMERS

HOW TO PROGRAM A COMPUTER



Programming Languages

Human

SQL, HTML & CSS

JavaScript / Python / Ruby

Java / C#

C/C+

Assembly

Machine

High Level vs. Low Level

Ease of Use vs. Performance

Portability vs. Specificity

Machine Code

```
8B542408 83FA0077 06B80000 0000C383
FA027706 B8010000 00C353BB 01000000
C9010000 008D0419 83FA0376 078BD98B
B84AEBF1 5BC3
```

Assembly

fib:

```
mov edx, [esp+8]  
cmp edx, 0  
ja @f  
mov eax, 0  
ret
```

@@:

```
cmp edx, 2  
ja @f  
mov eax, 1  
ret
```

C

```
if (n <= 0)
    return 0;
else if (n <= 2)
    return 1;
else {
    unsigned int a,b,c;
    a = 1;
    b = 1;
    while (1) {
        c = a + b;
        if (n <= 3) return c;
        a = b;
        b = c;
        n--;
    }
}
```

SQL

```
SELECT title  
  FROM   Book  
 WHERE price > 100.00  
ORDER BY author;
```

Same Same but Different

Receiving Input	Text, Clicks, Speech, Location, Big Data
Remembering	Data Stores: Variables, Cookies, Files, Databases
Making Decisions	Conditionals: if, else, then
Repeating Processes	Iteration/Loops: for, each, while
Producing Output	Text, Image, Sound, Movement

Programming Languages on the Web

Human

SQL, HTML & CSS

JavaScript / Python / Ruby

Java / C#

C/C+

Assembly

Machine

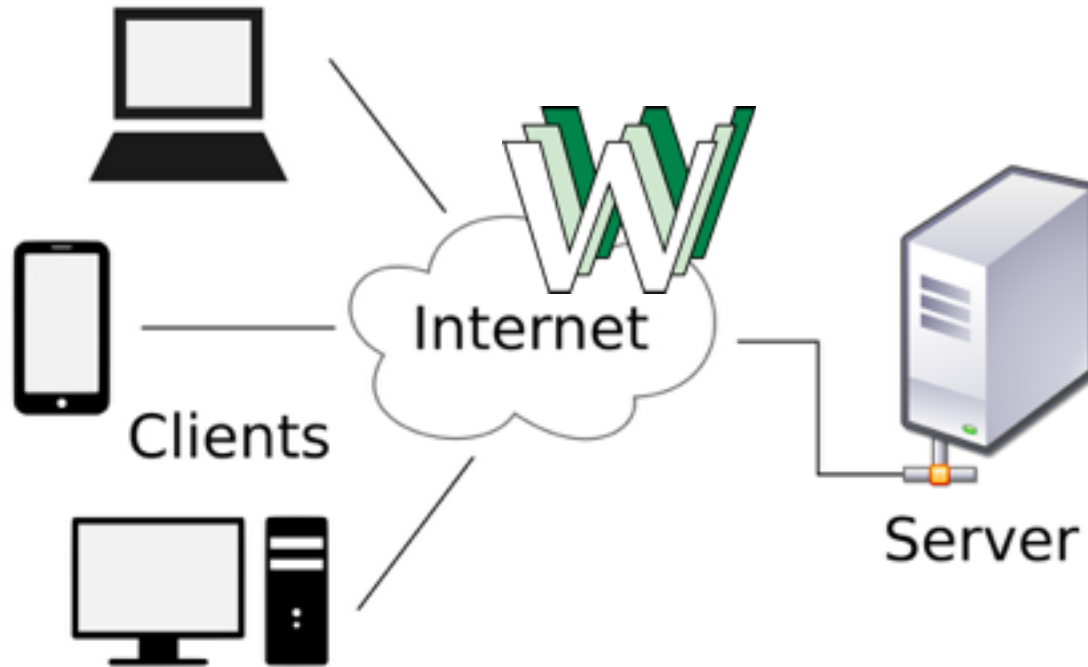
High Level vs. Low Level

Ease of Use vs. Performance

Portability vs. Specificity

PROGRAMMING FOR NON PROGRAMMERS

HOW THE WEB WORKS



World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#) , [Policy](#) , November's [W3 news](#) , [Frequently Asked Questions](#) .

[What's out there?](#)

Pointers to the world's online information, [subjects](#) , [W3 servers](#), etc.

[Help](#)

on the browser you are using

[Software Products](#)

A list of W3 project components and their current state. (e.g. [Line Mode](#) ,X11 [Viola](#) , [NeXTStep](#) , [Servers](#) , [Tools](#) ,[Mail robot](#) ,[Library](#))

[Technical](#)

Details of protocols, formats, program internals etc

[Bibliography](#)

Paper documentation on W3 and references.

[People](#)

A list of some people involved in the project.

[History](#)

A summary of the history of the project.

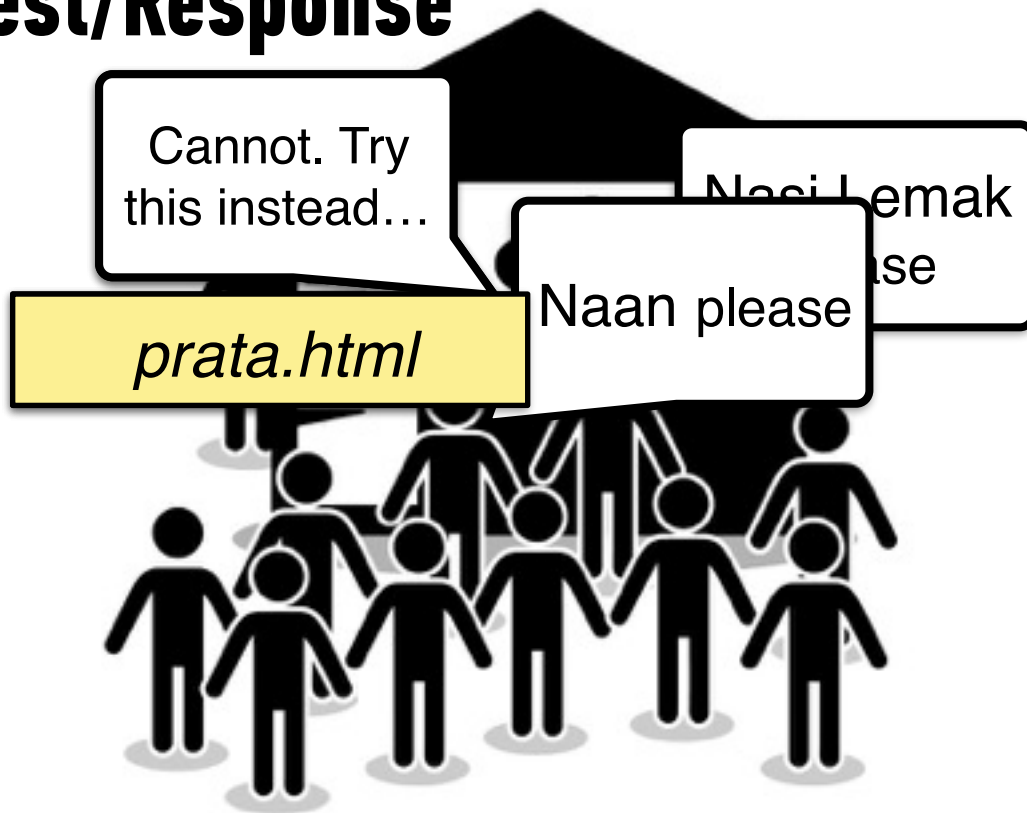
[How can I help ?](#)

If you would like to support the web..

[Getting code](#)

Getting the code by [anonymous FTP](#) , etc.

HTTP Request/Response



What is a Web Client?



Client-side / Front-end Development

HTML



structure

CSS



look

JS



dynamism

Server-side / Back-end Development



PostgreSQL



mongoDB®

PROGRAMMING FOR NON PROGRAMMERS

LAB

PAPER PROGRAMMING

ATM Program Design

Design a basic computer program for an ATM: PIN validation, Balance Enquiries, Withdrawals etc

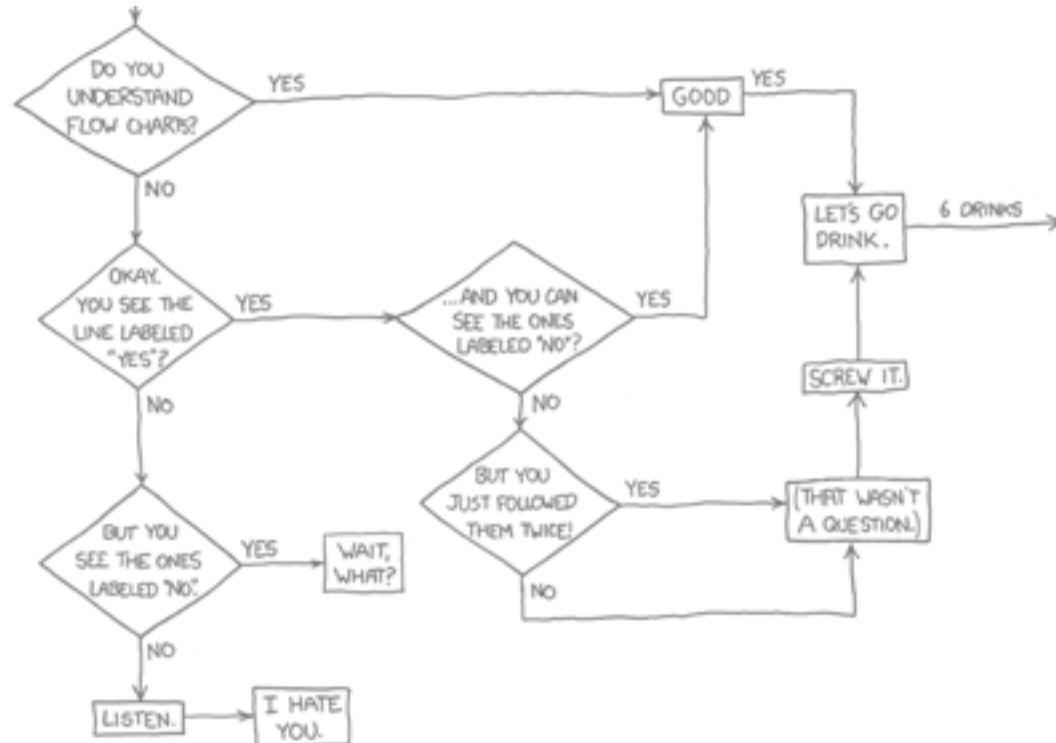
Receiving Input

Remembering

Making Decisions

Repeating Processes

Producing Output



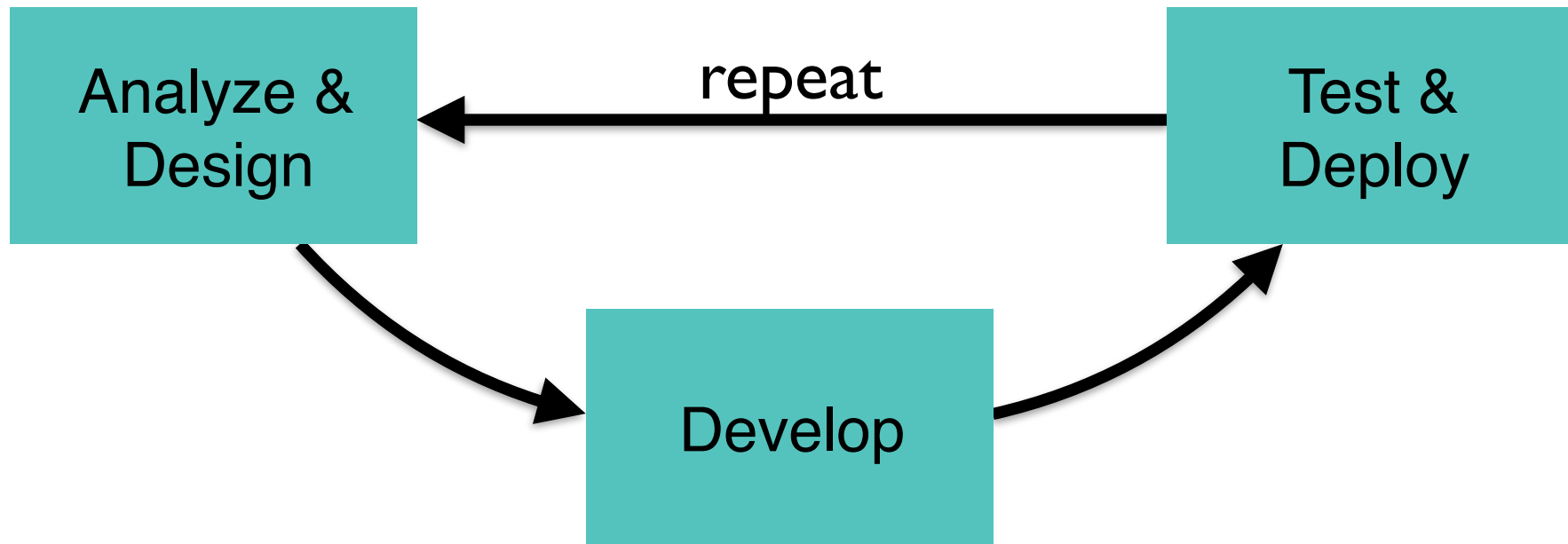
PROGRAMMING FOR NON PROGRAMMERS

THE DEVELOPMENT PROCESS

Agile Development

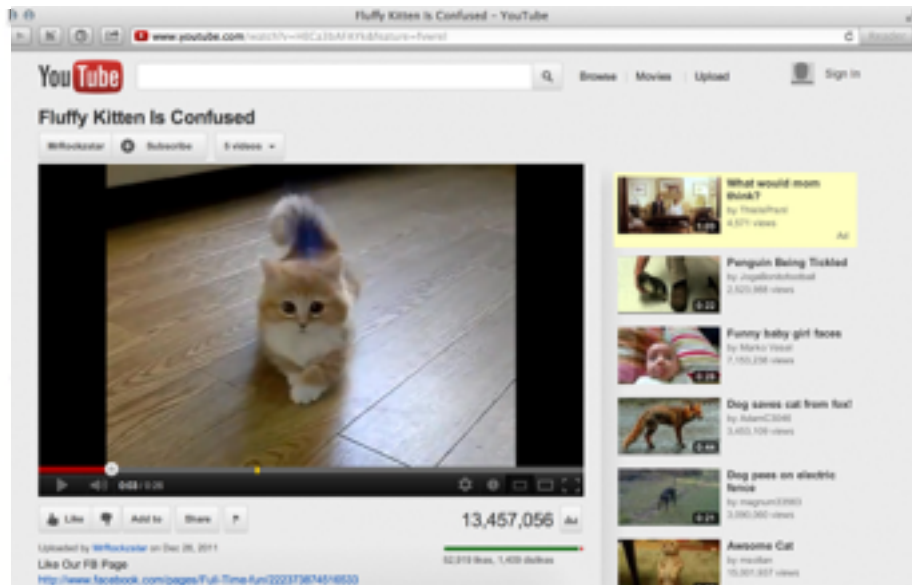


Agile vs. Waterfall



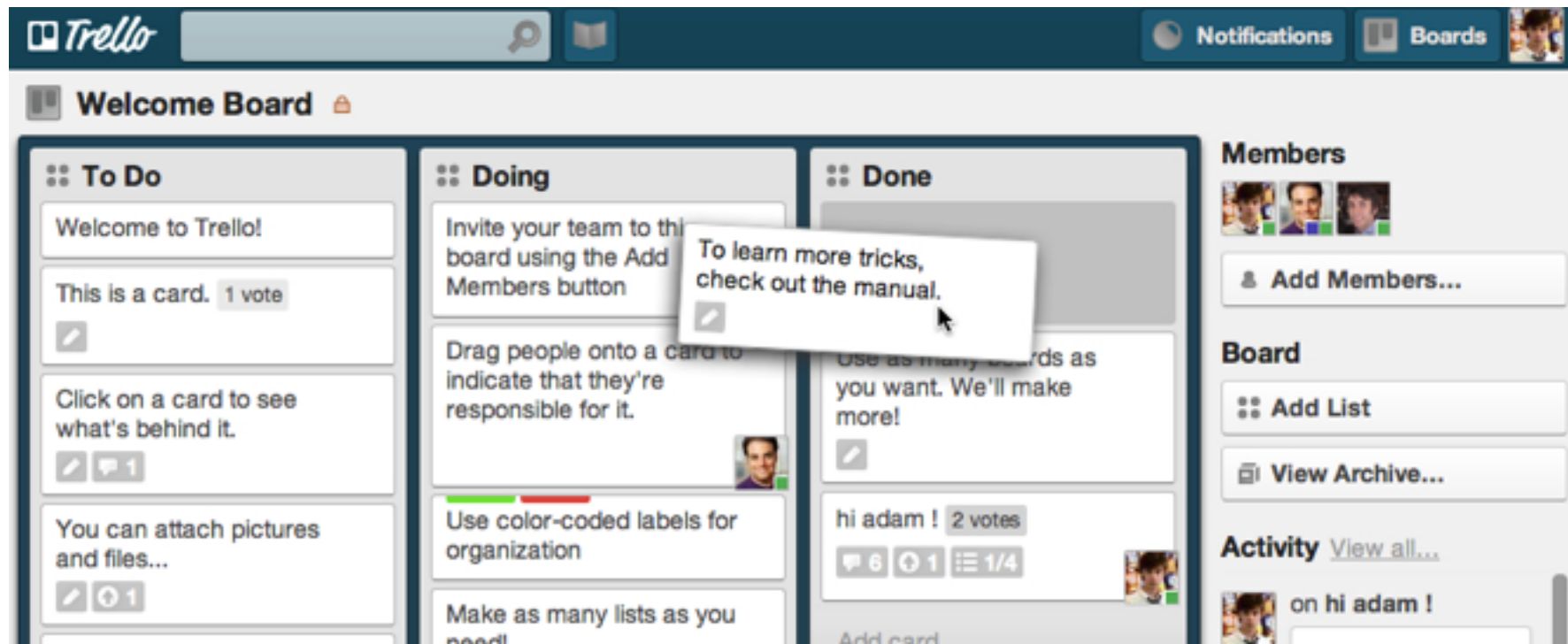
Analyze & Design

- User Research
- Information Architecture
- User Experience Design
- Visual Design



Wireframe vs. Mockup?

Develop



Trello

Notifications Boards

Welcome Board

To Do

- Welcome to Trello!
- This is a card. 1 vote
- Click on a card to see what's behind it.
- You can attach pictures and files...

Doing

- Invite your team to this board using the Add Members button
- Drag people onto a card to indicate that they're responsible for it.
- Use color-coded labels for organization
- Make as many lists as you need!

Done

- Use as many boards as you want. We'll make more!
- hi adam ! 2 votes

To learn more tricks, check out the manual.

Members

- Add Members...

Board

- Add List
- View Archive...

Activity [View all...](#)

- on hi adam !

Test & Deploy

- Automated vs. Manual
- Test First & TDD
- Continuous Integration



PROGRAMMING FOR NON PROGRAMMERS

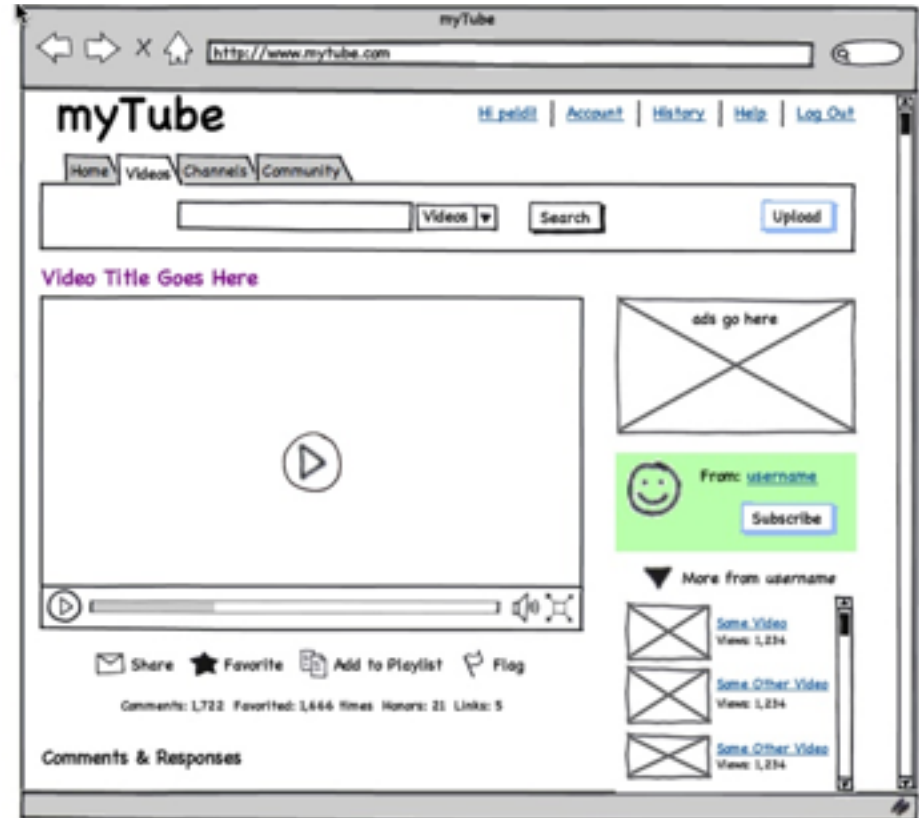
LAB

WEBSITE DESIGN

Website Design Lab

Using wireframe.cc design a 2 page biography website.

- Home Page
 - Person's name
 - Profile Picture
 - Description
 - Key Achievements
- Projects/Publications Page
 - Each: Name, Description, Image & External Link



PROGRAMMING FOR NON PROGRAMMERS

PART 2

BUILDING WEBSITES WITH HTML & CSS

Learning Objectives

- Explain the purpose of HTML, CSS & JS
- Develop a Basic Website using HTML & CSS

PROGRAMMING FOR NON PROGRAMMERS

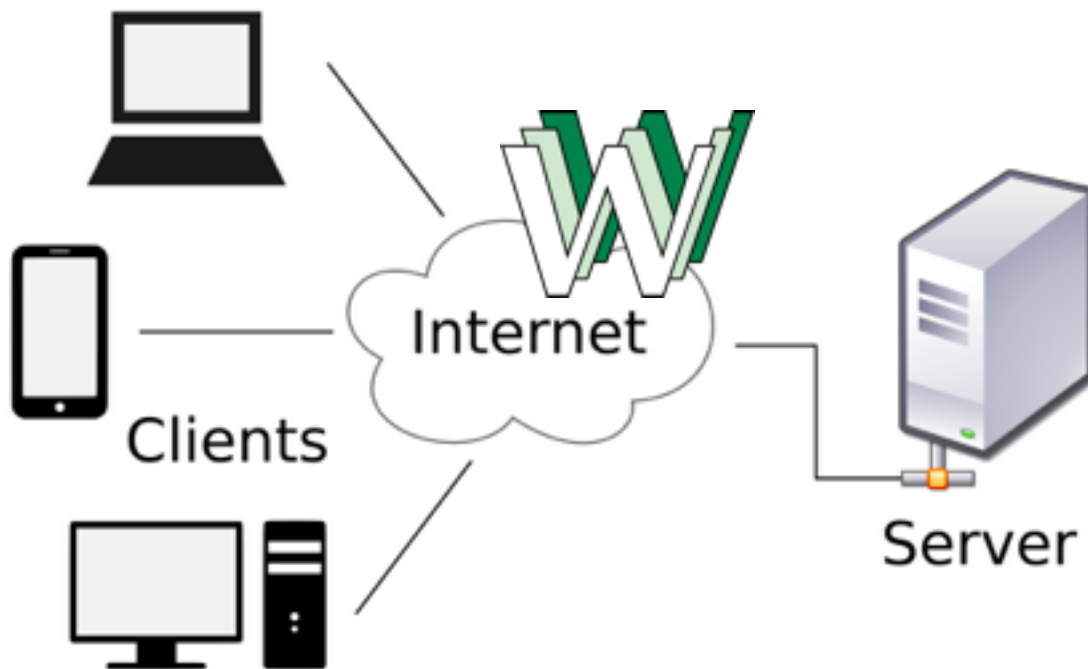
RECAP

PROGRAMMING THE WEB

What is Programming?

Receiving Input	Text, Clicks, Speech, Location, Big Data
Remembering	Data Stores: Variables, Cookies, Files, Databases
Making Decisions	Conditionals: if, else, then
Repeating Processes	Iteration/Loops: for, each, while
Producing Output	Text, Image, Sound, Movement

Client vs. Server



Front-end / Client-side Development

HTML



structure

CSS



look

JS



dynamism

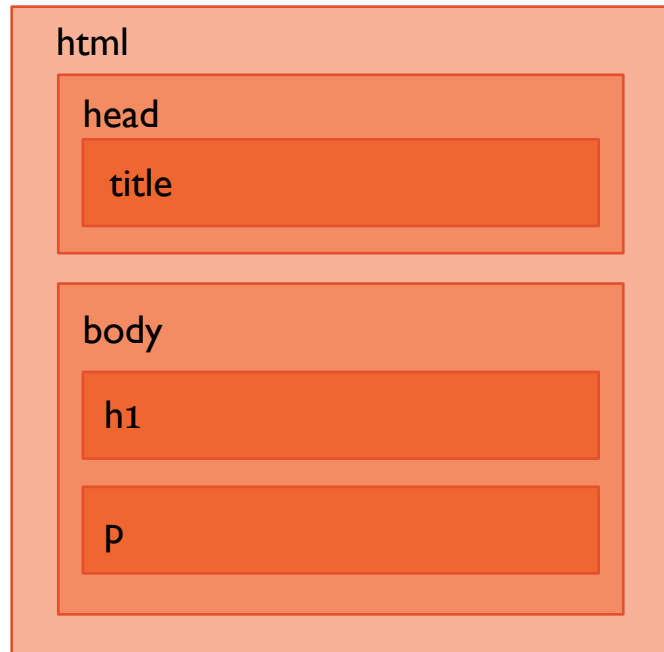
PROGRAMMING FOR NON PROGRAMMERS

HYPertext MARKUP LANGUAGE



Basic Document Structure

```
<html>  
  <head>  
    Meta Data goes here  
  </head>  
  
  <body>  
    Document Content goes here  
  </body>  
</html>
```



Headings

`<h1>Most Important Title</h1>`

`<h2>Second Most Important</h2>`

`<h3>A Less Important Title</h3>`

`<h4>Even Less Important Title</h4>`

`<h5>No One Really Uses This Title</h5>`

`<h6>Title Too Deep, Time To Stop</h6>`

Text

`<p>`A paragraph of text`</p>`

``Stress this point``

``Pay Attention Here``

`<hr>`

`
`

`<pre>`Do not Change

My Formatting browser!`</pre>`

Lists

```
<ol>
    <li>1st item</li>
    <li>2nd item</li>
</ol>
<ul>
    <li>an item</li>
    <li>another item</li>
</ul>
<dl>
    <dt>definition term</dt>
    <dd>description</dd>
</dl>
```

Containers

``simple grouping``

`<div>`simple container`</div>`

`<header>`container for header content`</header>`

`<nav>`container for navigation links`</nav>`

`<main>`container for main content`</main>`

`<article>`Self standing content group`</article>`

`<section>`a section of content`</section>`

`<footer>`container for footer content`</footer>`

Images

```

```



Links

`Internal Link`

`External Link`

↑
url goes here

↑
display text goes here

PROGRAMMING FOR NON PROGRAMMERS

LAB

BIOGRAPHY SITE

PART 1: HTML STRUCTURE

Biography Site Part 1: HTML

Make 2 HTML pages (NO CSS):
[index.html](#) & [projects.html](#)

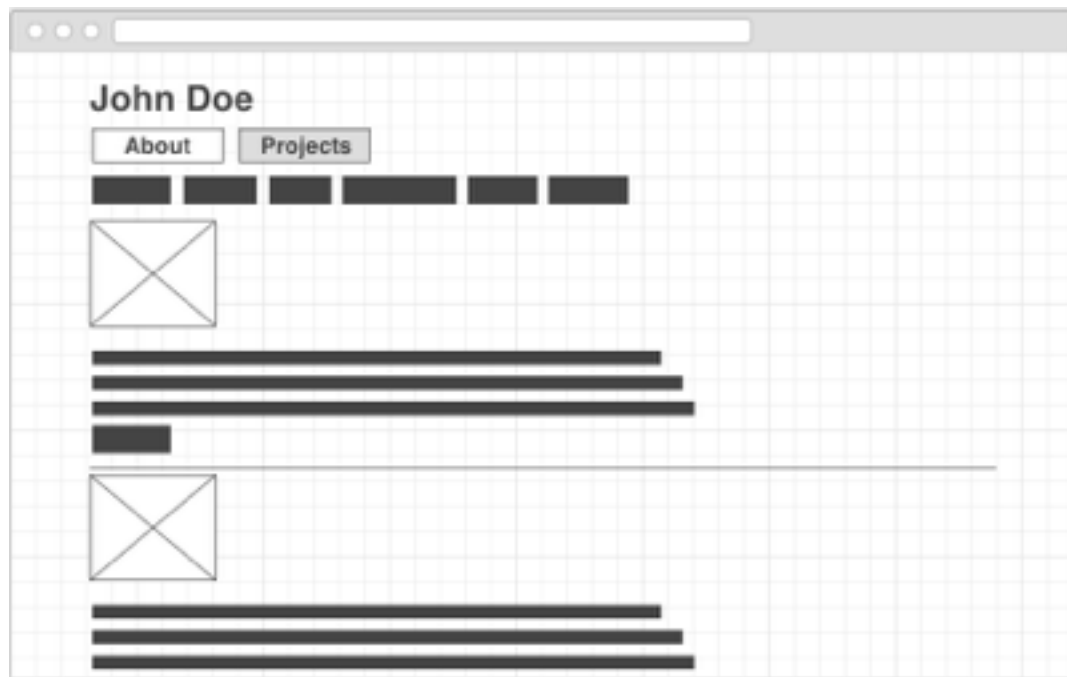
- Home Page

- Person's name
- Profile Picture
- Short Biography
- List of Key Achievements

- Projects/Publications Page

- List: Name, Description, Image & Link For Each

<http://www.w3schools.com/tags>



PROGRAMMING FOR NON PROGRAMMERS

CASCADING STYLESHEETS



SELECTOR

body {

DECLARATION

color: black;

PROPERTY VALUE

}

Linking To A Stylesheet

```
<html>  
  <head>  
    <link rel="stylesheet" href="styles.css">  
  </head>  
</html>
```

Selectors: Type

```
body {  
    background-color: pink;  
}  
div {  
    border: 1px solid black;  
}
```

Selectors: Class

```
<p>Normal text</p>
```

```
<p class="special">Special text</p>
```

```
<p class="summary special">Multiple classes</p>
```

```
.special {  
    font-weight: bold;  
}
```

Selectors: ID

```
<h1 id="main-title">Only 1 of These Ever</h1>
```

```
#main-title {  
    color: #ff0000;  
}
```

Selectors: Advanced

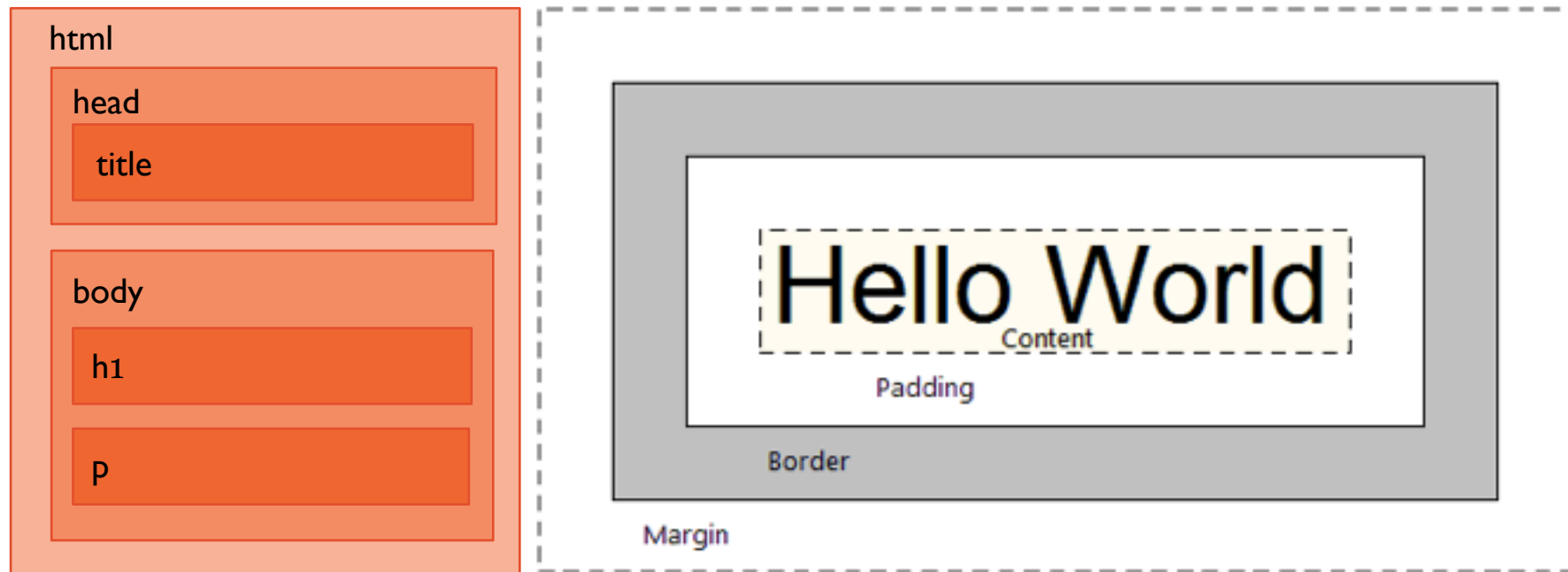
```
<p>First paragraph</p>  
<p>Second paragraph</p>
```

```
p :first-of-type {  
    font-weight: bold;  
}  
p:hover {  
    text-decoration: underline;  
}
```


Common Properties

```
body {  
  border: 1px solid black; /*thickness style color*/  
  background-color: pink;  
  background-image: url(backdrop.png);  
  font: italic 1.5em Georgia;  
  padding: 0 5px 5px 0; /*top left right bottom*/  
  width: 800px;  
}
```

Box Model



Positioning

```
.nav {  
  display: block; /*inline inline-block flex none*/  
  position: static; /*absolute fixed relative*/  
  top: 0;  
  right: 0;  
}
```

PROGRAMMING FOR NON PROGRAMMERS

LAB

BIOGRAPHY PAGE

PART 2: CSS DESIGN

Biography Site Part 2: CSS

Make 1 CSS file (styles.css) and import it in your html.

Styles

- Heading
- Paragraphs
- Lists
- Links
- Sections
- Images

<http://www.w3schools.com/cssref/>

