

# HACK UNIVERSITY

< WEB DEVELOPMENT />

WWW.HACKUNIVERSITY.COM

# HACK



## UNIVERSITY

< WEB DEVELOPMENT />

**WEEK 1**  
HOW THE INTERNET WORKS, HTML, AND CSS



TODAY



## Getting set up

- Joining the Lore

# LORE

- A web application for course management
  - Like ClassesV2
  - Used to be called CourseKit

# LORE

- A web application for course management

- Liked Classes Hub
- Used to be called Coursekick

# LET'S JOIN



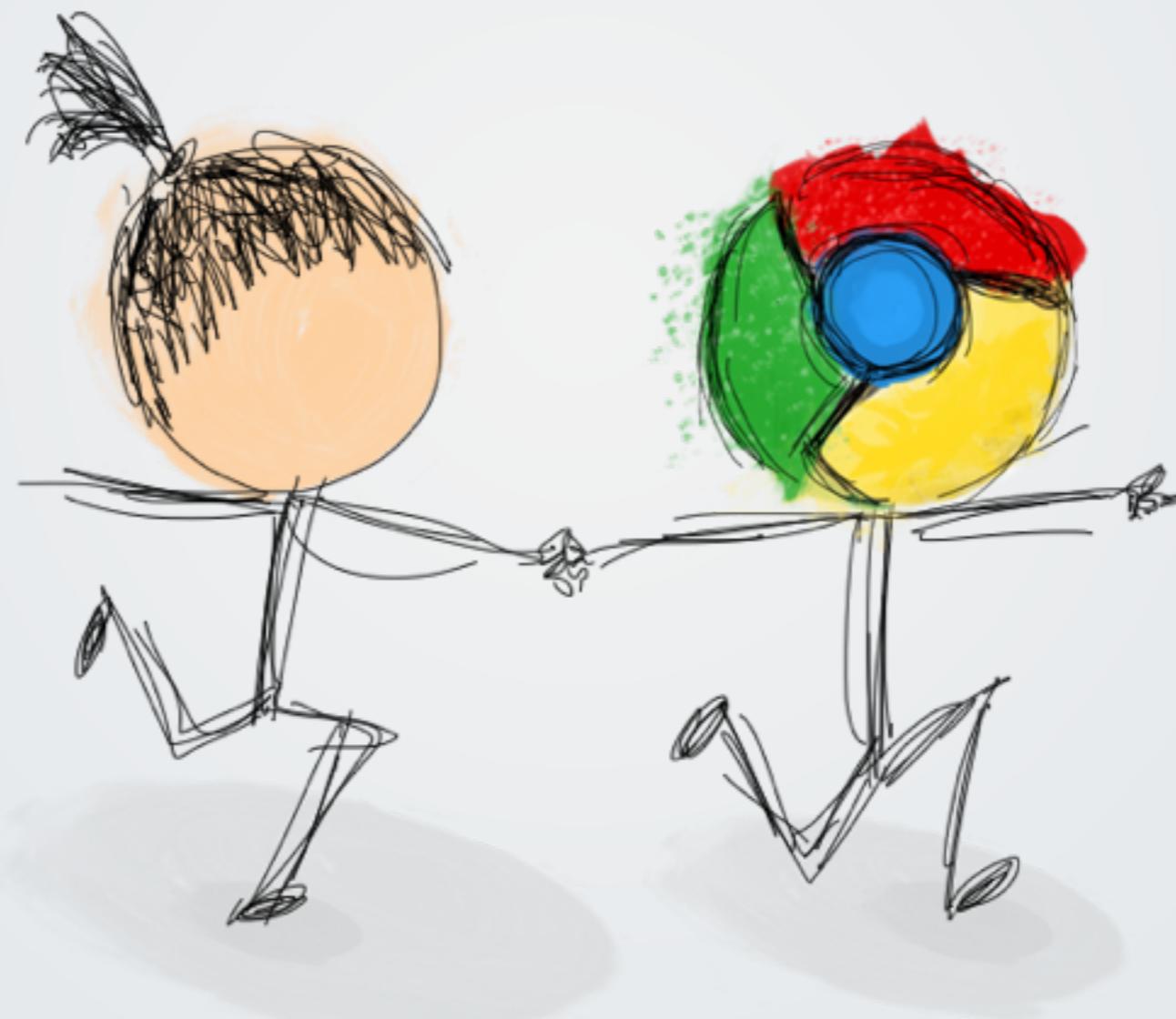
[join.lore.com/VCXPCW](https://join.lore.com/VCXPCW)

# RESOURCES

A QUICK ASIDE

# KEY CONCEPT

GOOGLE IS YOUR FRIEND



# KEY CONCEPT

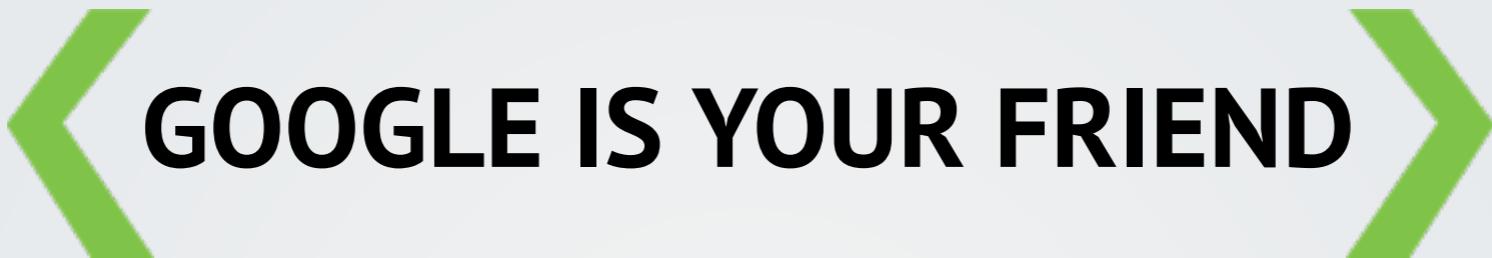
A large portion of web development is knowing where to look

Most common answer = Google

- Things to Google:
  - Syntax
  - Entire problems. Ex: “javascript dropdown menu”
  - Think of whatever question you would ask an expert, and ask it to Google first.

***WHAT DO WE DO  
WHEN WE ENCOUNTER  
A PROBLEM WE CAN'T  
IMMEDIATELY SOLVE?***





**GOOGLE IS YOUR FRIEND**

**BUT WE ARE HERE FOR YOU TOO!**

**THIS IS AN INTERACTIVE CLASS. PLEASE STOP  
US WHENEVER YOU HAVE A QUESTION.**

# **ANATOMY 101**

## WEB APPLICATIONS



# TERMINOLOGY

## Client-server model

- Client == (you and your) browser
- Server == machine sending (or “serving”) you the data and files you request

HOST ~== “server”

- “to host” (code, files, applications) ~== “to serve”

LOCAL == hosted on the machine in question

REMOTE == hosted on a different machine



**HACK** >  
UNIVERSITY

# REQUEST-RESPONSE CYCLE

(1) Client (browser) makes a “request”

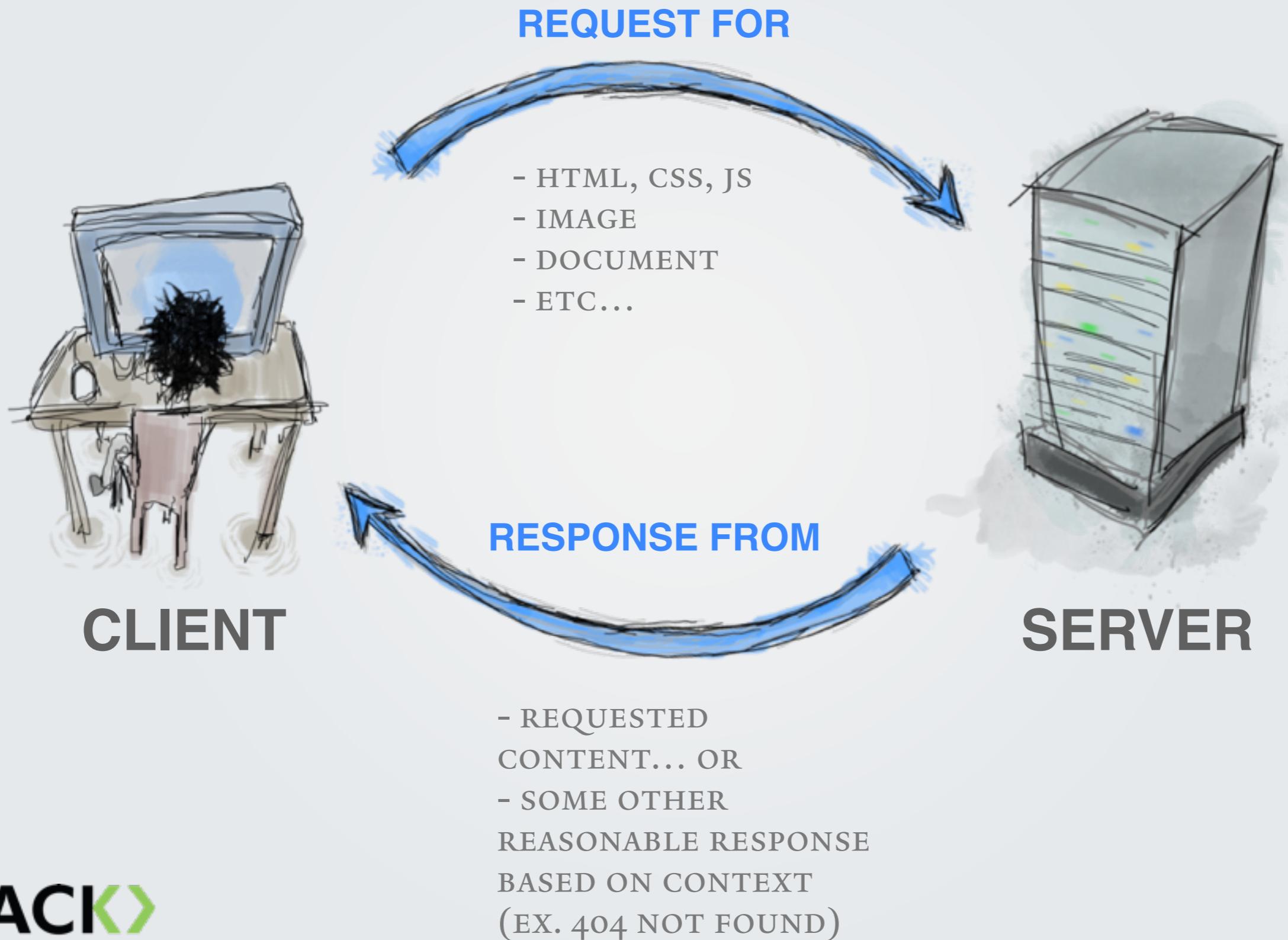
- REQUEST == textual message whose syntax and semantics are defined by HyperText Transfer Protocol (*HTTP*)

(2) Server issues a “response”

- RESPONSE == textual message defined by HTTP
- Contains status code. Ex: 404 (“*Not Found*”), 200 (“*Okay*”), 500 (“*Internal Server Error*”)

(3) Cycle repeats itself

# THE CLIENT-SERVER MODEL



# **WHOA WHOA, BACK UP**

OKAY, LET'S WALK AND TALK THROUGH  
WHAT GOES ON BEHIND THE SCENES  
WHEN YOU VISIT A WEBSITE,  
LIKE FACEBOOK, OR GOOGLE, OR ESPN

# **WELCOME TO THE FRONT-END**

LET'S GET LEARNED



# FRONT-END DEVELOPMENT

An interaction between three “languages”

- HTML - the content of the internet
- CSS - the style of the internet
- Javascript - the logic and action of the internet

DEMO: Let's go to a favorite website of ours and see what each technology does there

# HTML

The “content” of the Internet

- Just text, no formatting
- In a week or two, you’ll be able to make Google!

# HTML

## The “content” of the Internet

- Just text, no formatting
- In a week or two, you’ll be able to make Google!



# CODING IN HTML

---

Tags delineate content blocks (which may contain yet more tags)

- `<h1> I'm inside a tag! </h1>`
  - Tags state “properties,” and these properties are then passed on to the content within the tags
  - Mr. `<tag>` says “abide by my laws until further notice”
  - Then Ms. `</tag>` says “further notice”
-

# POPULAR TAGS

<div> the content blocks of HTML (an empty shell)

- <div>This content will be in an div block!</div>

<a> links

- <a href="http://www.hackyale.com">Home</a>

<img> images

- 

<p> paragraphs

- <p>This text will be in a nice paragraph</p>

```
<!DOCTYPE html>
<html>
<head>
    <title>Welcome To HackYale | HTML</title>
</head>
<body>
    <h1>Welcome To HackYale!</h1>
    <h3>This course will focus on front end technologies</h3>
    <ul>
        <li>HTML</li>
        <li>CSS</li>
        <li>JavaScript</li>
    </ul>
    <h2>HTML</h2>
    <p>
        HTML is the <em>content</em> of the web. Without it, none of your favorite sites would exist. You can write links to places like <a href="http://www.google.com/">Google</a>, <a href="http://www.nytimes.com/">The New York Times</a> or anywhere you like!
    </p>
    <p>
        If you go to the 'view' menu in chrome, scroll to 'developer', and click 'view source', you can see the HTML (among other things) that your browser rendered. (shortcut: command + option + u.)
    </p>
    <p>
        This page is <b>just</b> HTML (except for some browser defaults, but we'll get to those later).
    </p>
    <p>
        Next, let's check out <a href="/welcome/css">CSS</a>.
    </p>
</body>
</html>
```

# OUR SECOND WEBSITE

AN EXERCISE

# LET'S BUILD OUR FIRST WEBPAGE

There are about 20-25 tags you will need to know

But for today, we'll use:

- !DOCTYPE
- html
- head
- title
- body
- div
- h1, h2, ..., h6
- p
- img
- br
- <!-- ... -->

THAT'S ALREADY 1/2 OF WHAT 'EXPERTS' KNOW!

Now lets **fire** up our text editors  
and get coding!

# HTML DOCUMENT STRUCTURE

# HTML DOCUMENT STRUCTURE

<!DOCTYPE>

# HTML DOCUMENT STRUCTURE

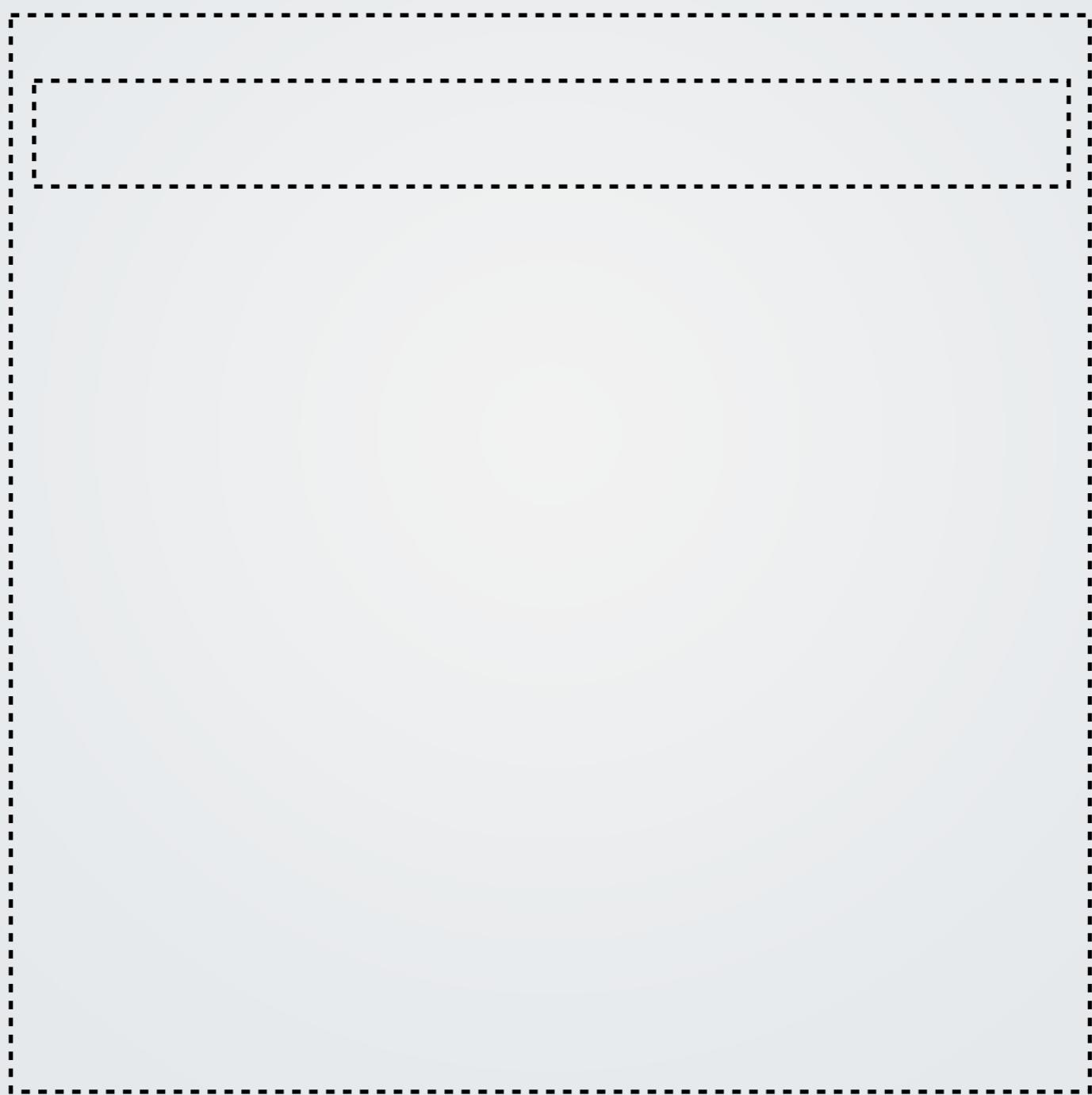
<!DOCTYPE>

<html>

</html>

# HTML DOCUMENT STRUCTURE

<!DOCTYPE>

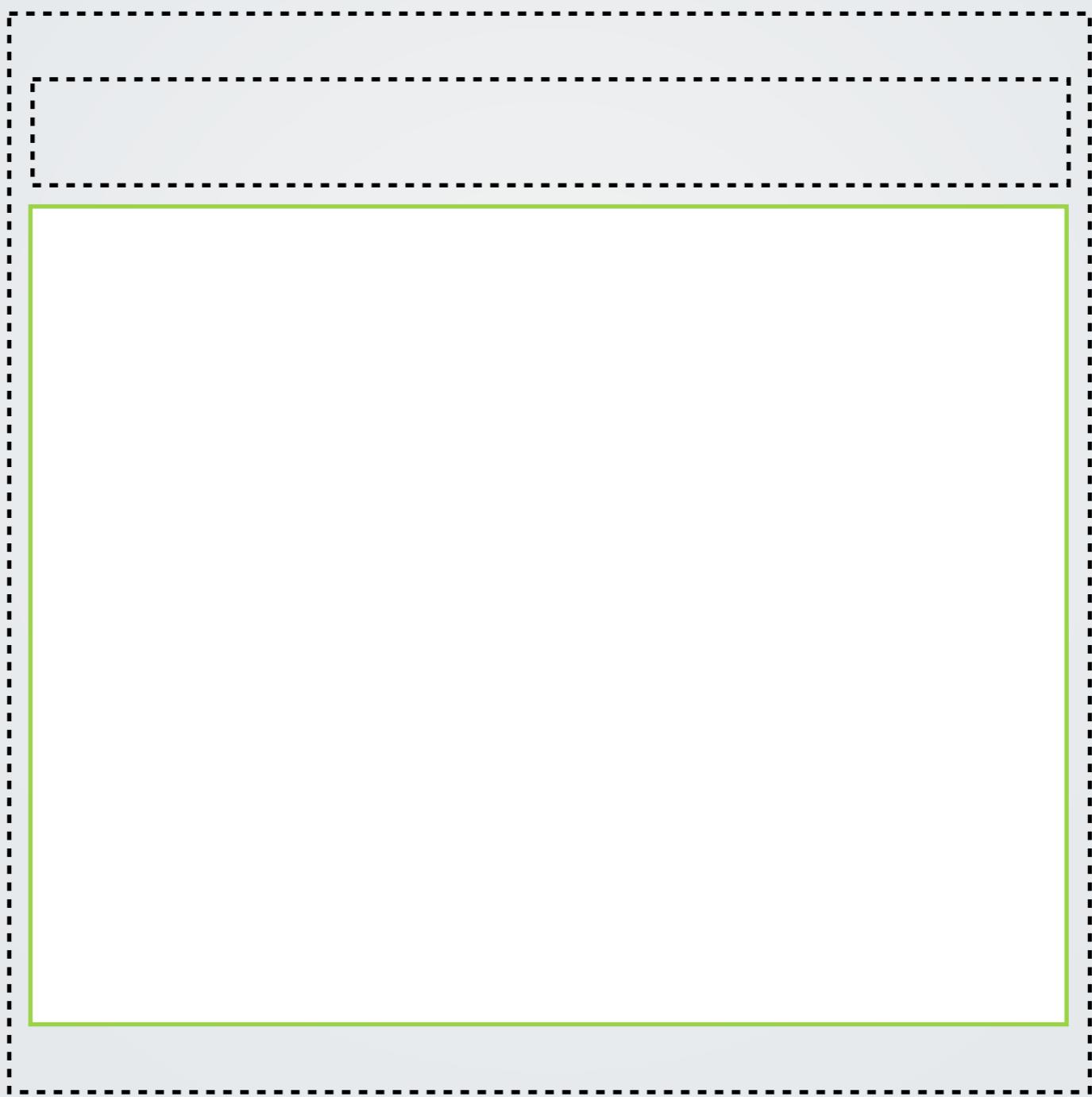


<html>  
<head>  
</head>

</html>

# HTML DOCUMENT STRUCTURE

<!DOCTYPE>



<html>  
<head>  
</head>  
<body>

</body>  
</html>

# ONLY A WEBPAGE? THAT'S SO 1994!

---

So, how about a website?

- But what is a website? It's just a collection of webpages!

New tag you'll need: <a>

---

Okay, that was cool, but my site was still kind of ugly.



# CASCADING STYLE SHEETS (CSS)

# ENTER, CSS

```
<p> Hello World! </p>
<p> Paragraphs are great! </p>
<p> Totally. </p>
```

Hello World!  
Paragraphs are great!  
Totally.

# ADDING STYLES

```
<p>  
  Hello World!  
</p>  
<p>  
  Paragraphs are great!  
</p>  
<p>  
  Totally.  
</p>
```

+

```
p {  
  color: red;  
}
```



Hello World!  
Paragraphs are great!  
Totally.

LESSON:  
Styling

# CSS SYNTAX

```
p {  
    font-size: 18px;  
    color: blue;  
}
```

# CSS SYNTAX

```
p {  
    font-size: 18px;  
    color: blue;  
}
```

SELECTOR

# CSS SYNTAX

```
p {  
    font-size: 18px;  
    color: blue;  
}
```

SELECTOR

DEFINITION BLOCK

# CSS SYNTAX

```
p {  
    font-size: 18px;  
    color: blue;  
}
```

SELECTOR  
DEFINITION BLOCK  
PROPERTY

# CSS SYNTAX

```
p {  
    font-size: 18px;  
    color: blue;  
}
```

SELECTOR  
DEFINITION BLOCK  
PROPERTY  
VALUE

# CSS SYNTAX

```
p {  
    font-size: 18px;  
    color: blue;  
}
```

SELECTOR  
DEFINITION BLOCK  
PROPERTY  
VALUE  
END OF DEFINITION

# THE SOURCE

```
<!DOCTYPE html>
<html>
<head>
  <title>Welcome To HackYale | HTML</title>
</head>
<body>
  <h1>Welcome To HackYale!</h1>
  <h3>This course will focus on front end technologies</h3>
  <ul>
    <li>HTML</li>
    <li>CSS</li>
    <li>JavaScript</li>
  </ul>
  <h2>HTML</h2>
  <p>
    HTML is the <em>content</em> of the web. Without it, none of your favorite sites would exist. You can write links to places like <a href="http://www.google.com/">Google</a>, <a href="http://www.nytimes.com/">The New York Times</a> or anywhere you like!
  </p>
  <p>
    If you go to the 'view' menu in chrome, scroll to 'developer', and click 'view source', you can see the HTML (among other things) that your browser rendered. (shortcut: command + option + u.)
  </p>
  <p>
    This page is <b>just</b> HTML (except for some browser defaults, but we'll get to those later).
  </p>
  <p>
    Next, let's check out <a href="/welcome/css">CSS</a>.
  </p>
</body>
</html>
```

# THE BORING RESULT

## Welcome To HackYale!

This course will focus on front end technologies

- HTML
- CSS
- JavaScript

### HTML

HTML is the *content* of the web. Without it, none of your favorite sites would exist. You can write links to places like [Google](#), [The New York Times](#) or anywhere you like!

If you go to the 'view' menu in chrome, scroll to 'developer', and click 'view source', you can see the HTML (among other things) that your browser rendered. (shortcut: command + option + u.)

This page is **just** HTML (except for some browser defaults, but we'll get to those later).

Next, let's check out [CSS](#).

# A DASH OF CSS

```
body {  
    padding: 0;  
    font: 14px "Lucida Grande", Helvetica, Arial, sans-serif;  
    background-color: #ffffff;  
    background-image: -moz-radial-gradient(50% 50%, circle farthest-side, white, #eaeaea 70%);  
    background-image: -webkit-radial-gradient(50% 50%, circle farthest-side, white, #eaeaea 70%);  
    background-image: -o-radial-gradient(50% 50%, circle farthest-side, white, #eaeaea 70%);  
    background-image: -ms-radial-gradient(50% 50%, circle farthest-side, white, #eaeaea 70%);  
    background-image: radial-gradient(50% 50%, circle farthest-side, white, #eaeaea 70%);  
    margin: 0;  
}  
  
a {  
    color: #00B7FF;  
    text-decoration: none;  
}  
  
a:hover {  
    text-decoration: underline;  
}  
  
ul, li {  
    list-style-type: none;  
}  
  
p {  
    margin-left: 40px;  
}  
  
.emph {  
    font-size: 20px;  
    color: #20AA20;  
    font-family: "Zapf Chancery", Parkavenue, cursive;  
}
```

# THE LESS BORING RESULT

## Welcome To HackYale!

This course will focus on front end technologies

HTML

CSS

JavaScript

## CSS

CSS is the *style* of the web.

With CSS, you can color, arrange, and generally style your page. In addition, you can add small amounts of dynamic activity, like the underline you'll see as the mouse hovers over the "JavaScript" link.

Next, let's check out [JavaScript](#).



# HOW DO WE INCLUDE CSS ON A PAGE?

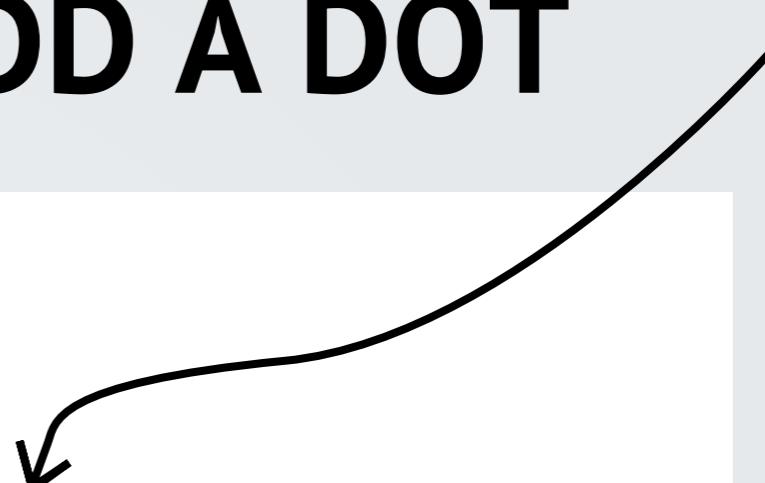
## LINK TAGS

# THE LINK TAG

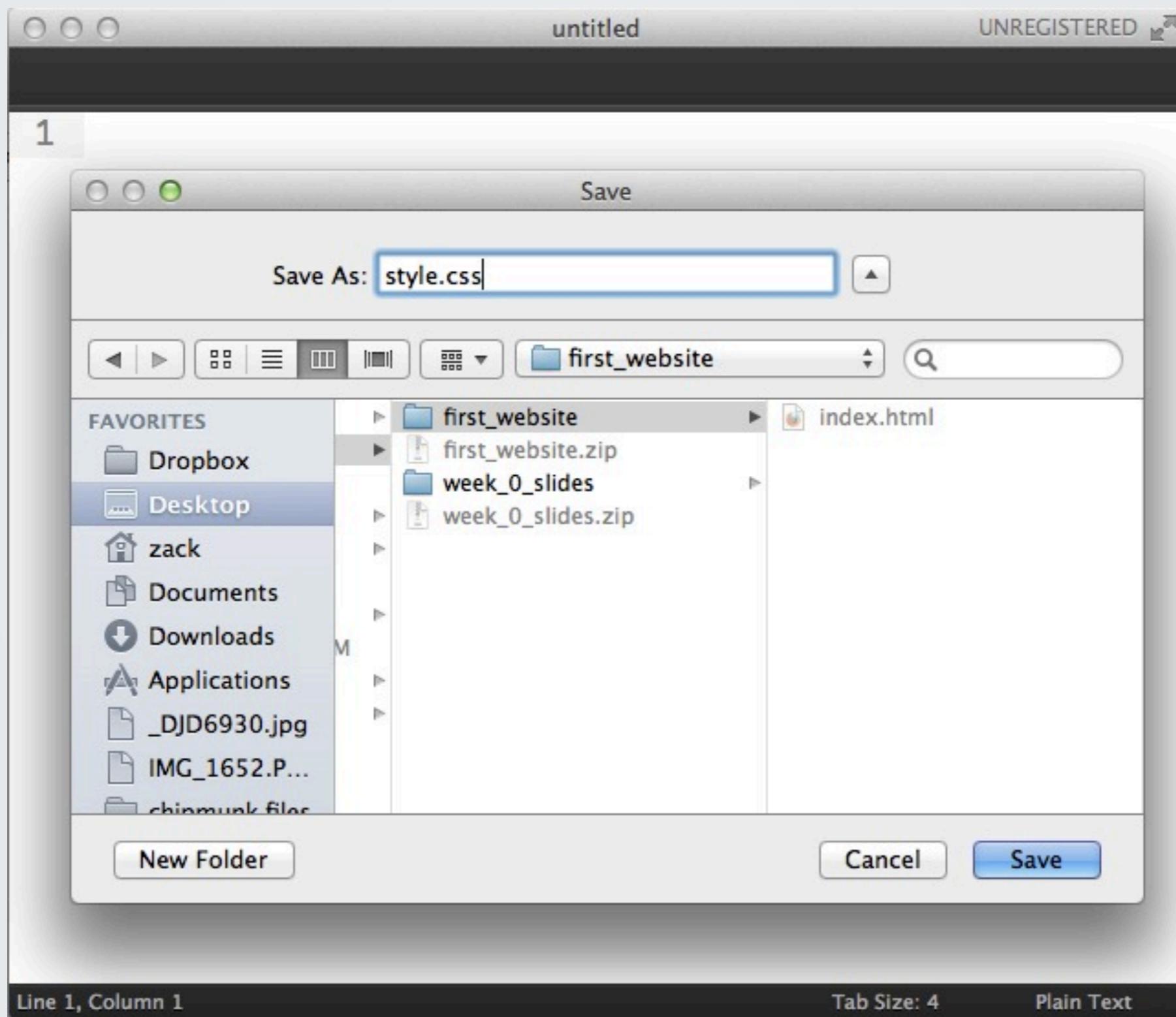
```
<html>
<head>
    <!-- some HTML -->
    <link rel="stylesheet" href="/style.css">
    <!-- some more HTML -->
</head>
<body>
    <!-- even more HTML -->
</body>
</html>
```

# WE'RE GOING TO ADD A DOT

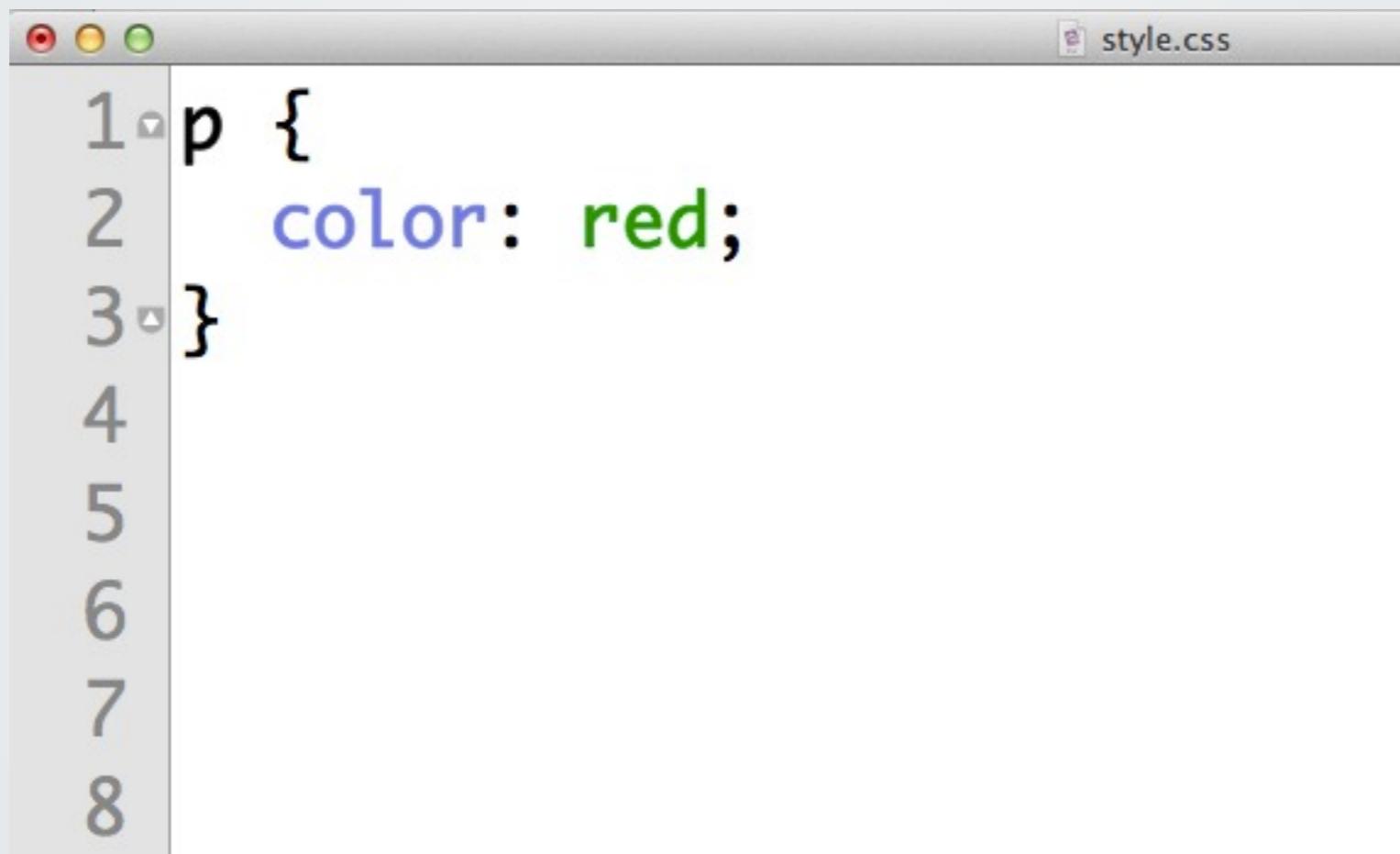
```
<html>
  <head>
    <!-- some HTML -->
    <link rel="stylesheet" href="./style.css">
    <!-- some more HTML -->
  </head>
  <body>
    <!-- even more HTML -->
  </body>
</html>
```



# SAVE CSS FILE IN PROJECT DIRECTORY



# ADD A STYLE DECLARATION



A screenshot of a Mac OS X application window titled "style.css". The window contains the following CSS code:

```
1 p {  
2     color: red;  
3 }
```

The code is numbered from 1 to 8 on the left side. The "color" keyword and its value "red" are highlighted in blue and green respectively.

# WOOHOO!

## Welcome To HackYale!

This course will focus on front end technologies

- HTML
- CSS
- JavaScript

### HTML

HTML is the *content* of the web. Without it, none of your favorite sites would exist. You can write links to places like [Google](#), [The New York Times](#) or anywhere you like!

If you go to the 'view' menu in chrome, scroll to 'developer', and click 'view source', you can see the HTML (among other things) that your browser rendered. (shortcut: command + option + u.)

This page is just HTML (except for some browser defaults, but we'll get to those later).

Next, let's check out [CSS](#).

# STATICCLOUD

# STATICCLOUD

- Free hosting for static websites
- As simple as it gets
- Backed by Amazon Web Services (quick digression)
- Was down last night (womp womp womp...)

# STATICCLOUD

- Free hosting for static websites
  - As simple as git
  - Backed by Amazon Web Services (quick digestion)
  - Was down last night (womp womp womp...)
- ## LET'S TRY IT

**WE MADE A REAL  
LIVE WEBSITE ON  
THE INTERNET!**

**(UNLESS STATICLOUD IS STILL DOWN)**





# ADMIN



# ADMIN

## ➤ Coding Homework

- Make a personal profile website (like what I'll show you)
- Upload it to hw1\_your\_name.staticcloud.com
- Post your url to the Lore

## ➤ Bonus Points

- Have multiple pages on your website

# ADMIN

## ➤ Office Hours

- Paul Fletcher-Hill (Tomorrow 8pm - 10pm)
- Rafi Khan (Thursday 8pm - 10pm)
- Zack Reneau-Wedeen (Monday 3pm - 5pm)
- Location TBA in the wrap-up email

# ADMIN

## ➤ Support Groups

- Small groups you can direct questions to
- More on that to come

# ADMIN

## ➤ Workshops

- Join the mailing list
- Stay tuned!

# THANKS!



# HACK UNIVERSITY

QUESTIONS EVEN GOOGLE CAN'T ANSWER?

TEAM@HACKYALE.COM

WWW.HACKUNIVERSITY.COM