# IDV460

INTERACTIVE DATA VIZ SPRING 16

# HTML + CSS

# WHAT IS A WEB PAGE?

A web page is a document or information resource that is suitable for the World Wide Web and can be accessed through a web browser and displayed on a monitor or mobile device.
Googles indexes about 48 billion web pages, roughly seven for every single person on Earth. Each of these pages has a unique URL.

http://mediaschool.indiana.edu/

# WHAT IS MADE OF?

Web pages are made up of a combination of the three core web technologies:

**HTML** = structure

# WHAT IS MADE OF?

Web pages are made up
of a combination of the
three core web
technologies:

**HTML** = structure
**CSS** = presentation

```
</div>
<br /><a target="_blank" href="tryit.asp?filename=tryhtml_intro" class="tryitbtn">Try
it yourself &raquo;</a>
</di

<h2>

<l
HTML
is n

tags

<h2>

&lt;
&lt;

tags
```

```css
#page-wrap {
    width: 860px;
    padding: 50px;
    position: relative;
    z-index: 2;
    }
#page-wrap ul { list-style: none; }
#logo {
    background: url(../images/logo.png) no-repeat;
    width: 800px;
    height: 168px;
    margin: 0px 30px;
    }
header h1, header p {
     display:none;
    }
#archive {
    margin: 25px 0 0 0;
    font: 16px Impact, sans-serif;
    text-transform: uppercase;
    text-align: center;
    position: relative;
    }
#archive span {
    color: #BF2200;
}
```

# WHAT IS MADE OF?

Web pages are made up of a combination of the three core web technologies:

**HTML** = structure
**CSS** = presentation
**JS** = functionality

```html
</div>
<br /><a target="_blank" href="tryit.asp?filename=tryhtml_intro" class="tryitbtn">Try it yourself &raquo;</a>
</di
```

```css
#page-wrap {
    width: 860px;
```

```
<h2>
  <l
HTML
is n

#p
tags-    #l

<h2>

&lt;l
&lt;,   hec

tags-    #a


#a
```

```javascript
var dismiss = '[data-dismiss="alert"]'
var Alert   = function (el) {
  $(el).on('click', dismiss, this.close)
}

Alert.VERSION = '3.3.6'

Alert.TRANSITION_DURATION = 150

Alert.prototype.close = function (e) {
  var $this    = $(this)
  var selector = $this.attr('data-target')

  if (!selector) {
    selector = $this.attr('href')
    selector = selector && selector.replace(/.*(?=#[^\s]*$)/, '') // strip for ie7
  }

  var $parent = $(selector)

  if (e) e.preventDefault()

  if (!$parent.length) {
    $parent = $this.closest('.alert')
  }

  $parent.trigger(e = $.Event('close.bs.alert'))

  if (e.isDefaultPrevented()) return

  $parent.removeClass('in')

  function removeElement() {
    // detach from parent, fire event then clean up data
    $parent.detach().trigger('closed.bs.alert').remove()
  }
```

# WHAT IS MADE OF?

Another way of thinking about this is to consider these languages as parts of speech:
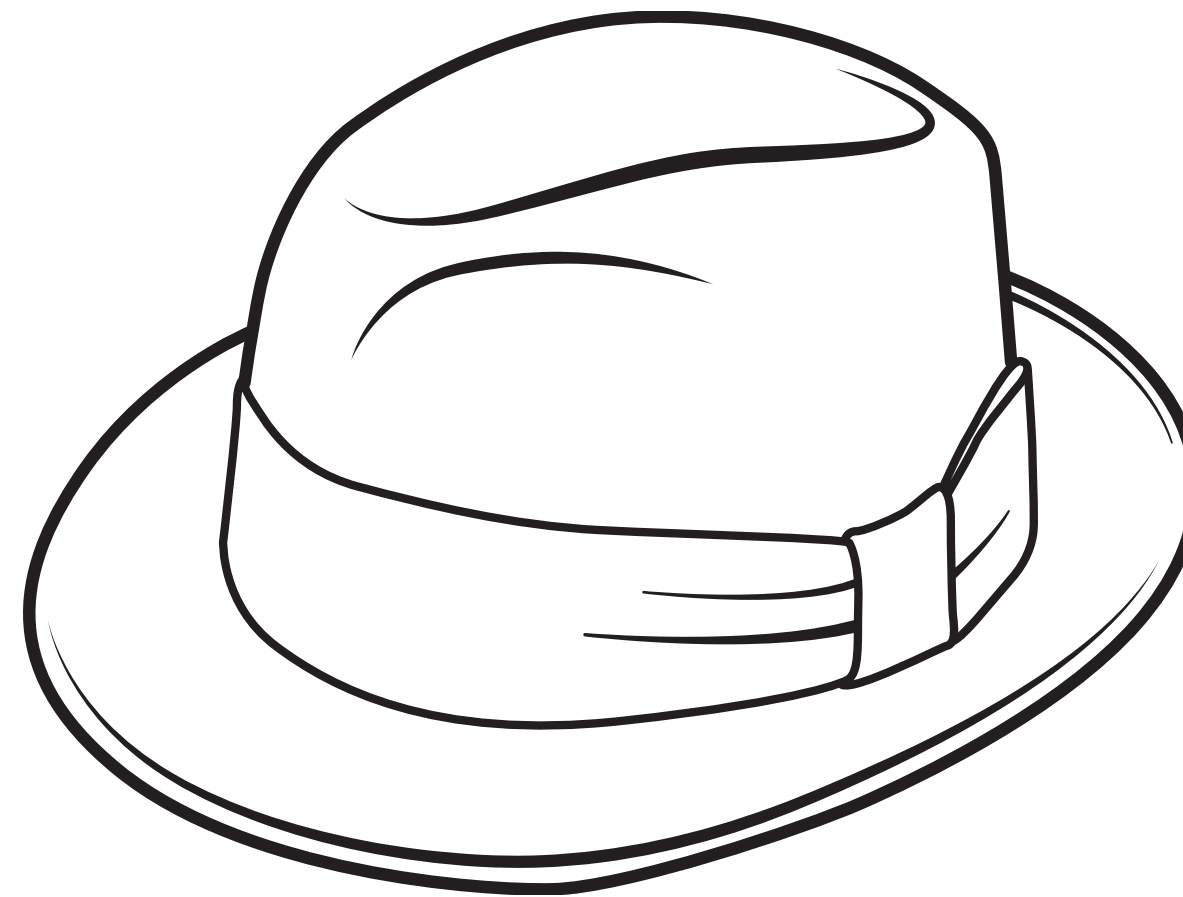
**HTML** = nouns



A hat.

# WHAT IS MADE OF?

Another way of thinking about this is to consider these languages as parts of speech:
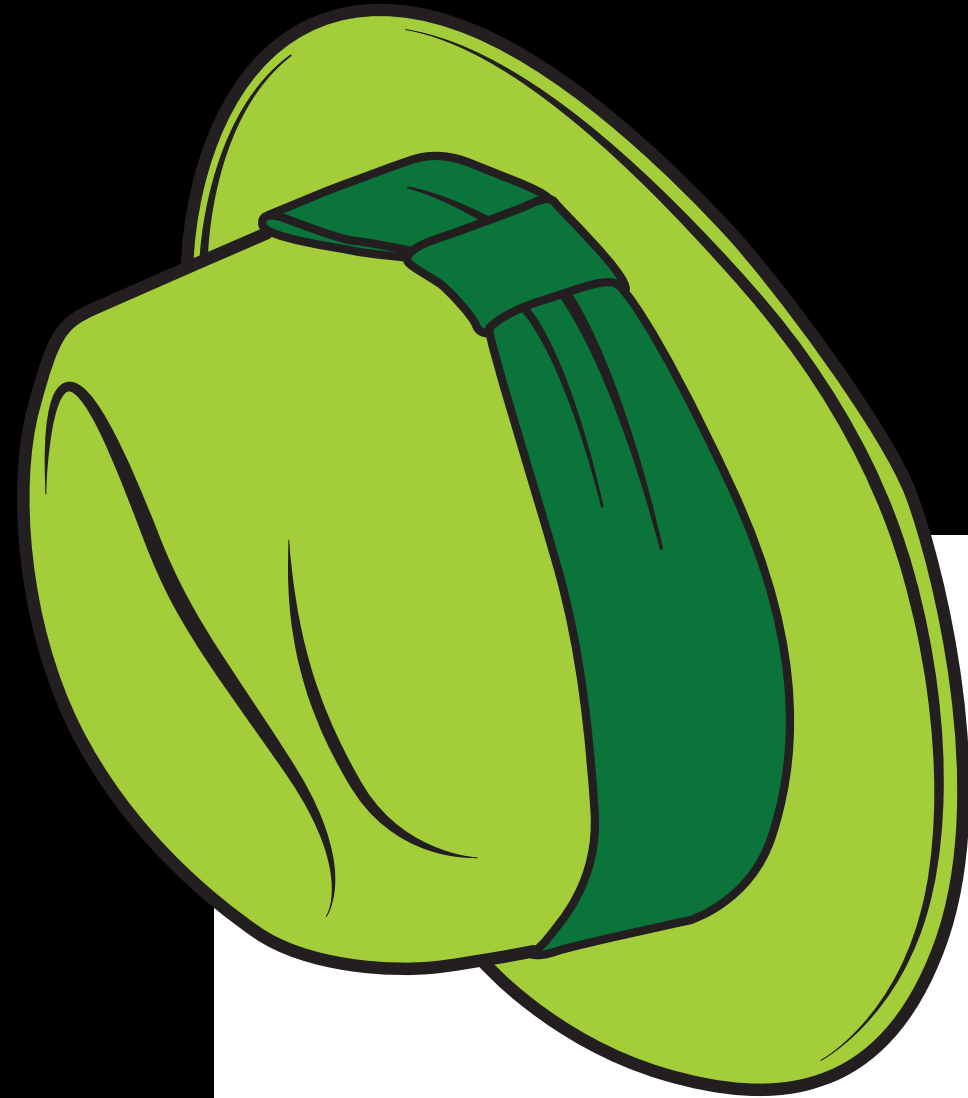
**HTML** = nouns
**CSS** = adjectives

A green hat.

# WHAT IS MADE OF?

Another way of thinking about this is to consider these languages as parts of speech:

**HTML** = nouns
**CSS** = adjectives
**JS** = verbs

*Throw* the green hat.

# A TYPICAL WEB PAGE

Here are some standard components used by many pages on a typical website.

# A TYPICAL WEB PAGE

**Header:** Includes the name of the site and often some sort of graphic.

# A TYPICAL WEB PAGE

**Navigation:** A set of links that take the user to other pages on the site. Navigation can be horizontal, vertical or both, but should be consistent across all the pages of the site.

# A TYPICAL WEB PAGE

**Content:** The "meat" of the page, typically set according to a grid in one or more columns of varying widths.
In HTML5, these columns have descriptive names (section, main, aside, etc.)

# A TYPICAL WEB PAGE

**Footer:** The bottom of the page, it frequently includes basic, small-type information such as an address, phone number or e-mail as well as text-based site navigation.



to shut the pop-holes. With the ring of light from his lantern dancing from side to side, he lurched across the yard, kicked off his boots at the back door, drew himself a last glass of beer from the barrel in the scullery, and made his way up to bed, where Mrs. Jones was already snoring.

*Read more ...*

## 1984
*(written in 1949)*

It was a bright cold day in April, and the clocks were striking thirteen. Winston Smith, his chin nuzzled into his breast in an effort to escape the vile wind, slipped quickly through the glass doors of Victory Mansions, though not quickly enough to prevent a swirl of gritty dust from entering along with him.

*Read more ...*

**COVER STORY**

Here are a bunch of cool 1984 covers.

A small sampling of 1984 covers ◀ ▶

**BIOGRAPHY | NOVELS | NON-FICTION | JOURNALISM | LEGACY | CONTACT ME**

ALL COPYRIGHTS RESERVED UNLESS OTHERWISE STATED

# HTML

# HTML

HTML (hypertext markup language) forms the building blocks of all Web pages. Created by Tim Berners-Lee, HTML is relatively simple: it consists of about 100 code elements that are used to "tag" certain kinds of content — defining what that content is so that CSS can be used to style it.

```
<a href="#">anchor, for hyperlinks</a>
<body>The entire visible portion of the page</body>
<br>Inserts a line break without creating a new paragraph
<div>Page division, creating boxes around content</div>
<em>Emphasis, displays as italics</em>
<h1>Largest bold heading</h1>
<h2>Second-largest</h2>
<h6>Smallest heading</h6>
<head>Document head, code that performs background functions
<html>Root element, contains entire web page</html>
<img src="image.jpg">An image file placed on an HTML page
<li>List item</li>
<ol>Ordered list, creates an ordered list</ol>
<p>Paragraph, creates a stand-alone paragraph</p>
<script>Contains a script or points to external script</scri
<span>Used to apply formatting to a portion of the document<
<strong>Emphasizes by displaying in bold</strong>
<style>Calls CSS style rules</style>
<td>A table cell</td>
<th>A table header cell</th>
<tr>A table row</tr>
<ul>Unordered list, creates a bulleted list</ul>
```

## Tags

HTML is a markup language, using a series of tags.

In HTML, tags are **keywords** surrounded by angled brackets.

`<html>`

# Tags

HTML tags normally come in pairs: an opening tag and a closing tag.

They look the same, except that a closing tag has a slash after the first angled bracket.

This is an HTML *element*.

`<title>Hello.</title>`
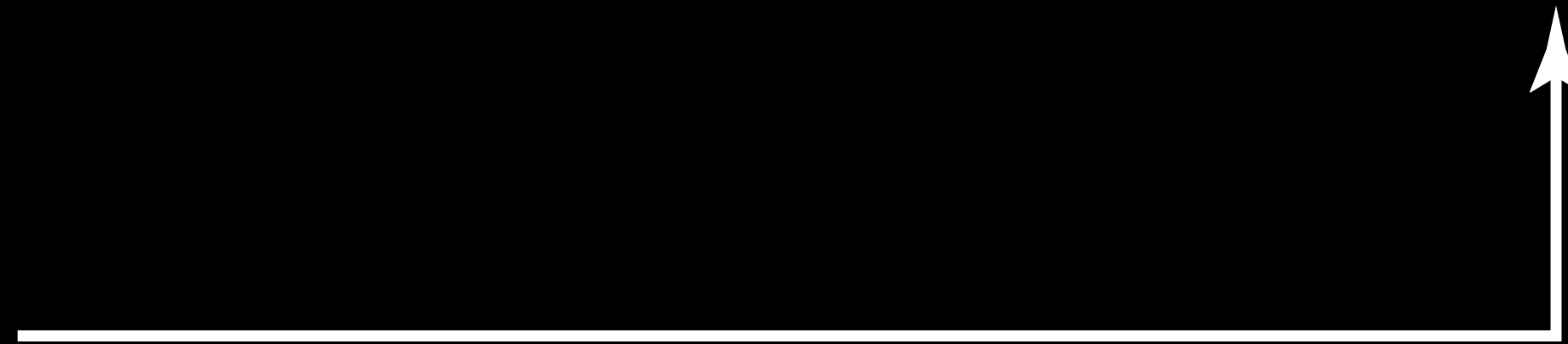
# Elements

There are two kinds of elements: empty and non-empty.

**Non-empty** elements are those that contain content — the information between the tags. Most tags are non-empty tags.

Some elements have no content. They are **empty** — they are closed within the start tag.

```html
<p>I am a paragraph.</p>
```

```html
<img src="images/example.png">
<br>
```

# Elements

Most HTML elements can be nested —
that is, they can contain other elements.
The three elements here are common to
every page:

The **<html>** element defines the entire
document.

```html
<html>
 <head>
  <title>First page</title>
 </head>
 <body>
  <h1>Headline</h1>
  <p>Paragraph</p>
 </body>
</html>
```

# Elements

Most HTML elements can be nested — that is, they can contain other elements. The three elements here are common to every page:

The **<html>** element defines the entire document.

The **<head>** element contains *invisible* information about the page.

```
<html>
<head>
 <title>First page</title>
</head>
<body>
 <h1>Headline</h1>
 <p>Paragraph</p>
</body>
</html>
```

# Elements

Most HTML elements can be nested — that is, they can contain other elements. The three elements here are common to every page:

The **<html>** element defines the entire document.

The **<head>** element contains *invisible* information about the page.

The **<body>** element contains all the *visible* content of the page.

```html
<html>
<head>
  <title>First page</title>
</head>
<body>
  <h1>Headline</h1>
  <p>Paragraph</p>
</body>
</html>
```

# Containers

An HTML document is essentially a container of containers.
Each container is an HTML element, and most containers can and do contain other containers.

```
<html>

  <head>

    <title>                          </title>

                                       </head>

  <body>

    <p>

      <img>

      <a>                              </a>

                                        </p>

                                       </body>

                                      </html>
```

# Containers

The relationship between these elements is *familial*. We say that the <html> element is the **parent**. The <head> and <body> elements are **children** of the <html> element. The <p> element is a **child** of the <body> element, and the <img> and <a> elements are **children** of the <p> element. The <head> and <body> are **siblings**, or parallel.

```
<html>
    <head>
        <title>                          </title>
    </head>

    <body>
        <p>
            <img>

            <a>                          </a>
        </p>
    </body>
</html>
```

# Attributes

HTML elements can also have **attributes**, which provide additional information about the element.

Attributes are always structured as **name="value"**

Attribute values must *always* be enclosed in quotes (single or double, but they have to match).

```
<img src="images/example.jpg" alt="not here" >
```

**ATTRIBUTE**      **ATTRIBUTE**

# Attributes

There are different kinds of attributes, too.

**Global** attributes can be used on *any* HTML element.

```
class="something"
id="something"
style="color:blue"
```

# Attributes

Other (non-global) attributes are specific to certain tags only.

For example, the **<img>** element can have several attributes, including the URL of the image (**src**) relative to the HTML document.

```
<img src="smiley.gif" alt="smiley face"
height="42" width="42">
```

The **<a>** element, which is used for hyperlinks, includes the **href** attribute to specify the URL you're linking to.

```
<a href="http://www.j563.com" target="_blank"></a>
```

# Common tags

Here are some of the most commonly used tags in HTML, and how they are structured.

**Headings**

```
<h1>The largest headline</h1>
<h6>The smallest headline</h6>
```

**Paragraphs**

```
<p>As the alarm clock on the chest of drawers
exploded like a horrid little bomb of bell metal,
Dorothy, wrenched from the depths of some complex,
troubling dream, awoke with a start and lay on her
back looking into the darkness in extreme
exhaustion.</p>
```

**Links**

```
<a href="http://www.w3schools.com">This is a link></a>
```

**Images**

```
<img src="images/example.jpg" alt="not here" >
```
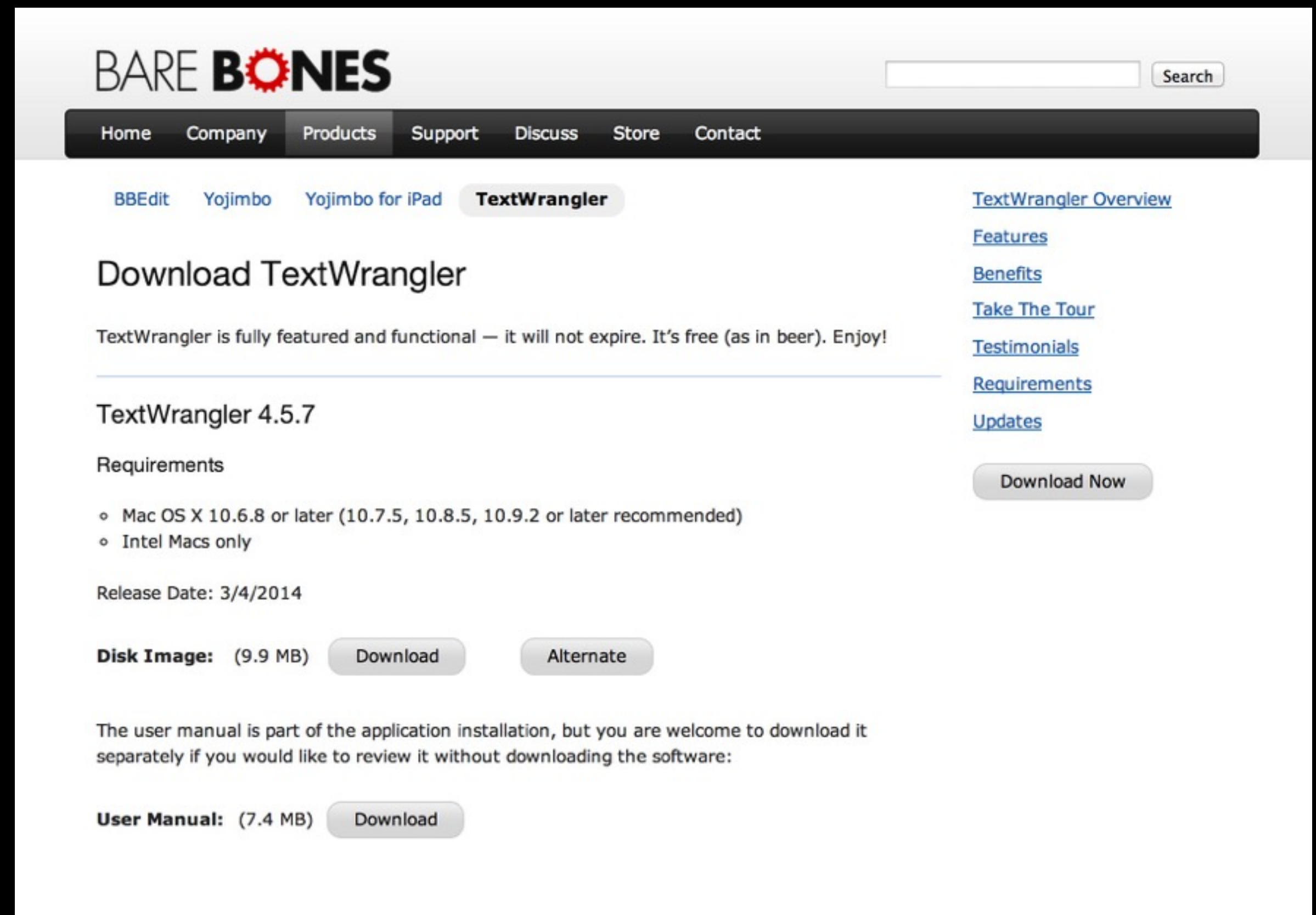
# Getting started

You can write HTML in a number of different programs, including web design software like Dreamweaver and code-writing software like Sublime. You can also apply HTML tags to Microsoft Word documents and even Text Edit.

In our class, we'll use **Text Wrangler**, which you can download (at home) for free at **http://www.barebones.com/products/textwrangler/download.html** Text Wrangler is Mac only; for PCs, you can try Notepad++ (also free, at **notepad-plus-plus.org**).

# Getting started

Our goal today is to prepare your basic website for this class, which will have a URL of

**pages.iu.edu/~name/idv460**

# Getting started

Open TextWrangler from your
Applications folder (or from
the dock).
Type the following lines:
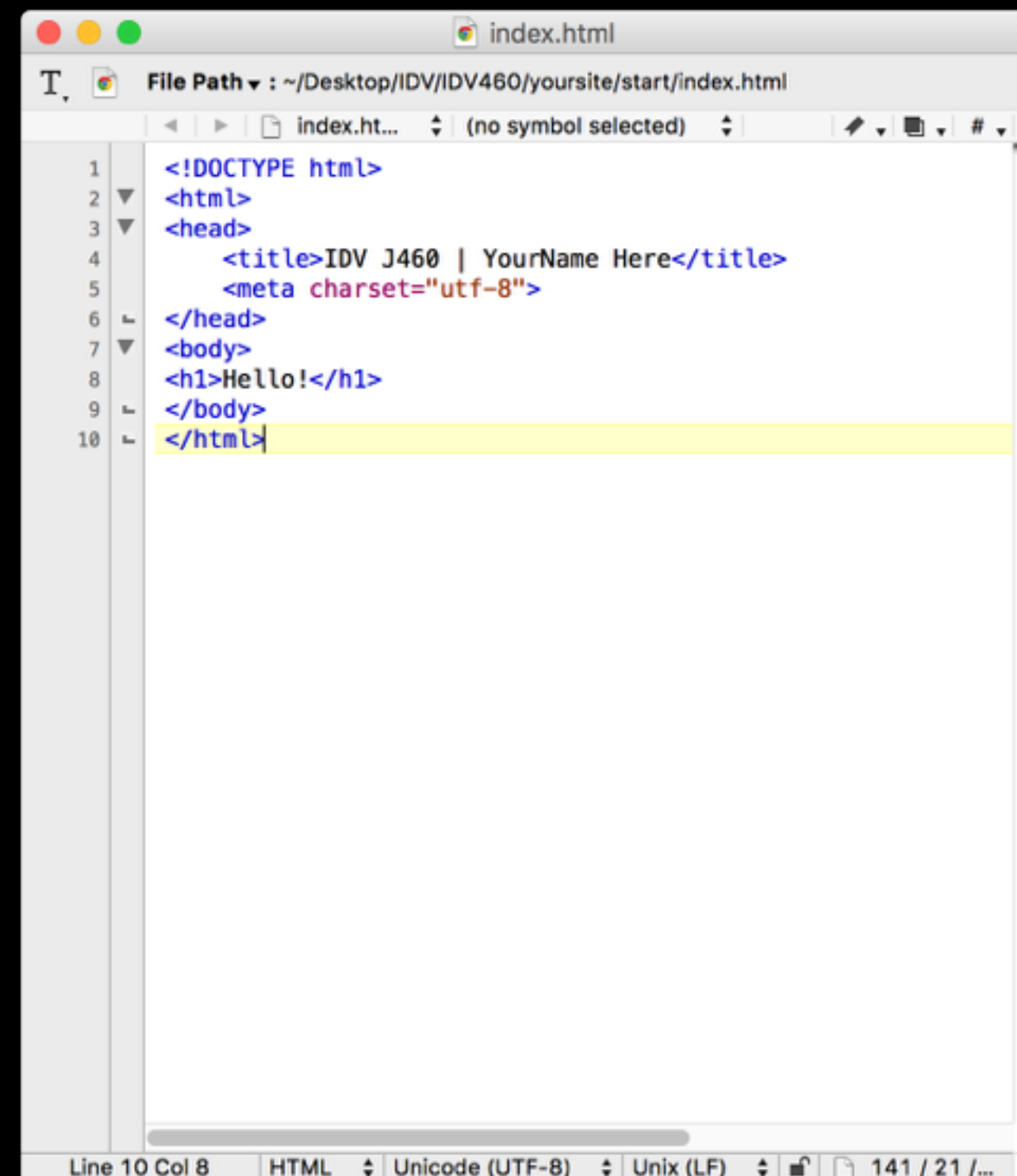


```
<!DOCTYPE html>
<html>
  <head>
    <title>IDV J460 | YourName Here</title>
    <meta charset="utf-8">
  </head>
  <body>
    <h1>Hello!</h1>
  </body>
</html>
```

# Getting started

Save the file onto your desktop as **index.html** Double-click the file there to open it in a browser, and resize the browser window so that it and TextEdit are side by side.

index.html

**File Path** ▾ **:** ~/Desktop/IDV/IDV460/yoursite/start/index.html

index.ht...   (no symbol selected)

```
1    <!DOCTYPE html>
2    <html>
3    <head>
4        <title>IDV J460 | YourName Here</title>
5        <meta charset="utf-8">
6    </head>
7    <body>
8    <h1>Hello!</h1>
9    </body>
10   </html>
```

Line 10 Col 8    HTML    Unicode (UTF-8)    Unix (LF)    141 / 21 /...

IDV J460 | YourName Here ×    Steve ⚠

file:///Users/stlayton/Desktop/IDV/IDV460/yo...

Apps    User Dashboard    Lynda    Chase.com    »    Other Bookmarks
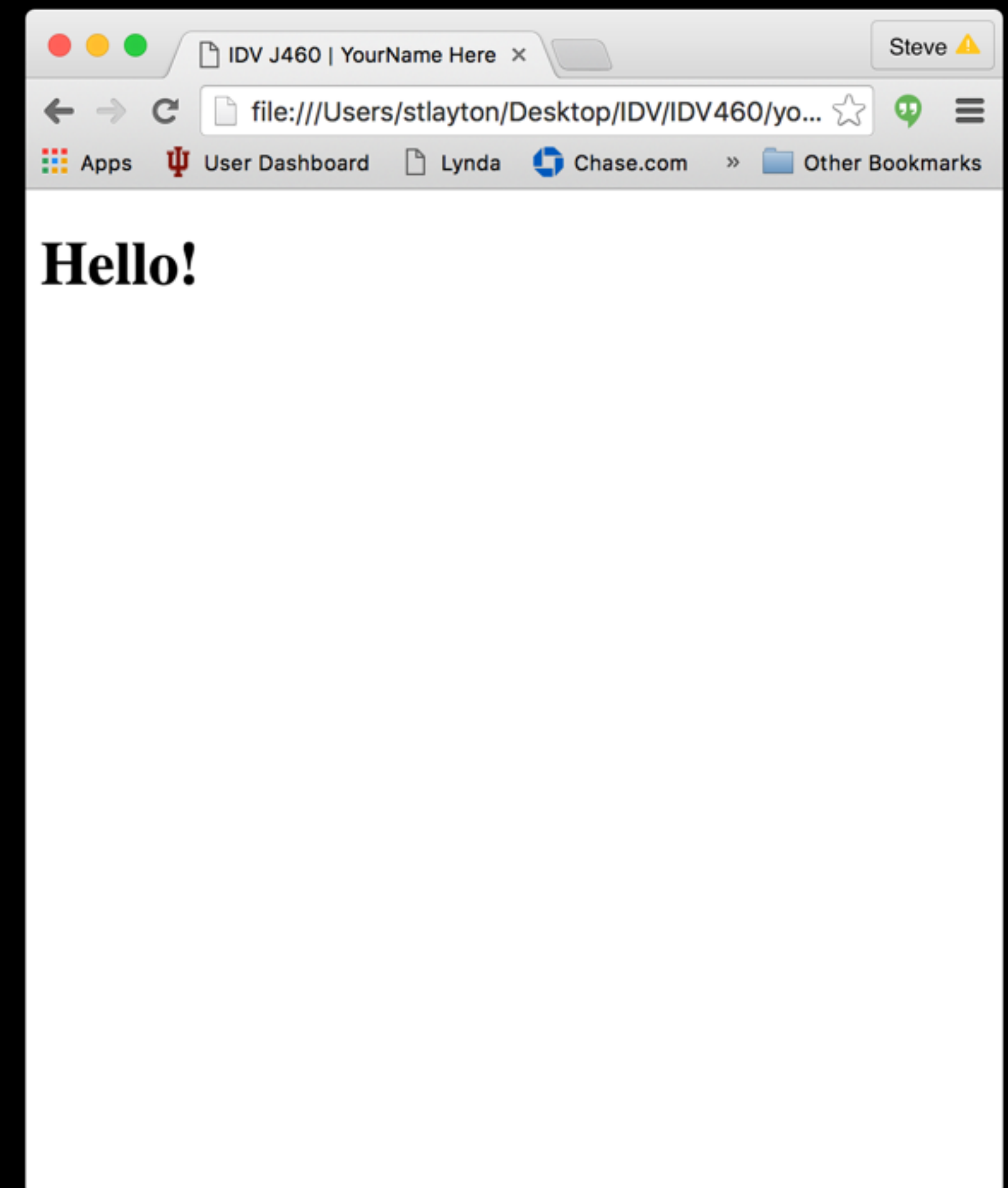
# Hello!

## Getting started

In your editor, begin to set up the basic structural composition of your page.



```html
<!DOCTYPE html>
<html>
  <head>
    <title>IDV J460 | YourName Here</title>
    <meta charset="utf-8">
  </head>
  <body>
    <header>
    </header>
    <nav>
    </nav>
    <main>
      <h1>Hello!</h1>
    </main>
    <footer>
    </footer>
  </body>
</html>
```

## Changing styles

We will begin learning to using CSS for presentation in earnest shortly.
For now, though, to understand how it works, we'll add a quick style rule to our HTML head tag.

```html
<!DOCTYPE html>
<html>
 <head>
  <title>IDV J460 | YourName Here</title>
  <meta charset="utf-8">
  <style>
  body {
   background-color: peachpuff;
  }
  h1 {
   font-family: sans-serif;
   color: orange;
   text-align: center;
   position: fixed;
   top: 50%;
   left: 50%;
   transform: translate(-50%, -50%);
  }
  </style>
```

```
<!DOCTYPE html>
<html>
 <head>
  <title>IDV J460 | YourName Here</title>
  <meta charset="utf-8">
```

# Upload!

We will go ahead and publish this simple html page as your (temporary) website for IDV460.

| 🖴 SFTP (SSH File Transfer Protocol) | ↕ |

Server: `mercury.uits.indiana.edu`  Port: `22`

URL: sftp://stlayton@mercury.uits.indiana.edu:22/

Username: `stlayton`

Password: `••••••••••••••••••••`

☐ Anonymous Login

☑ Add to Keychain

? Cancel **Connect**

▶ **More Options**

```
   left: 50%;
   transform: translate(-50%, -50%);
  }
 </style>
```

# Building your site

Go ahead and quit Text Wrangler, and return to the finder. Make a new folder there called website, and create the following folders inside it:
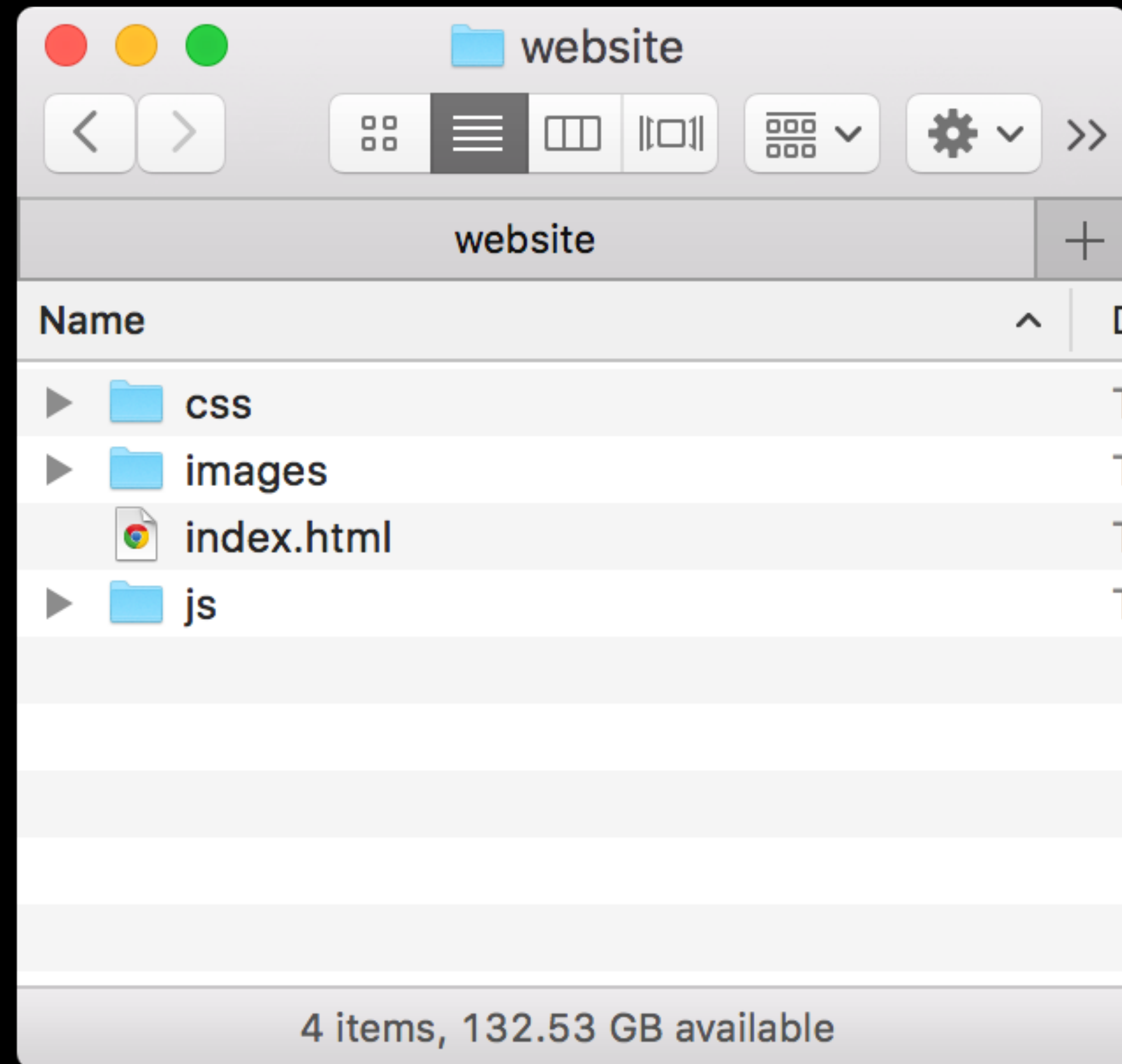
# CSS

# Folder

Drag the entire folder you just made on top of the Text Wrangler icon in the dock to launch it again.
You will be able to see the other folders you made in the sidebar, which makes it easy to go back and forth between documents.



website
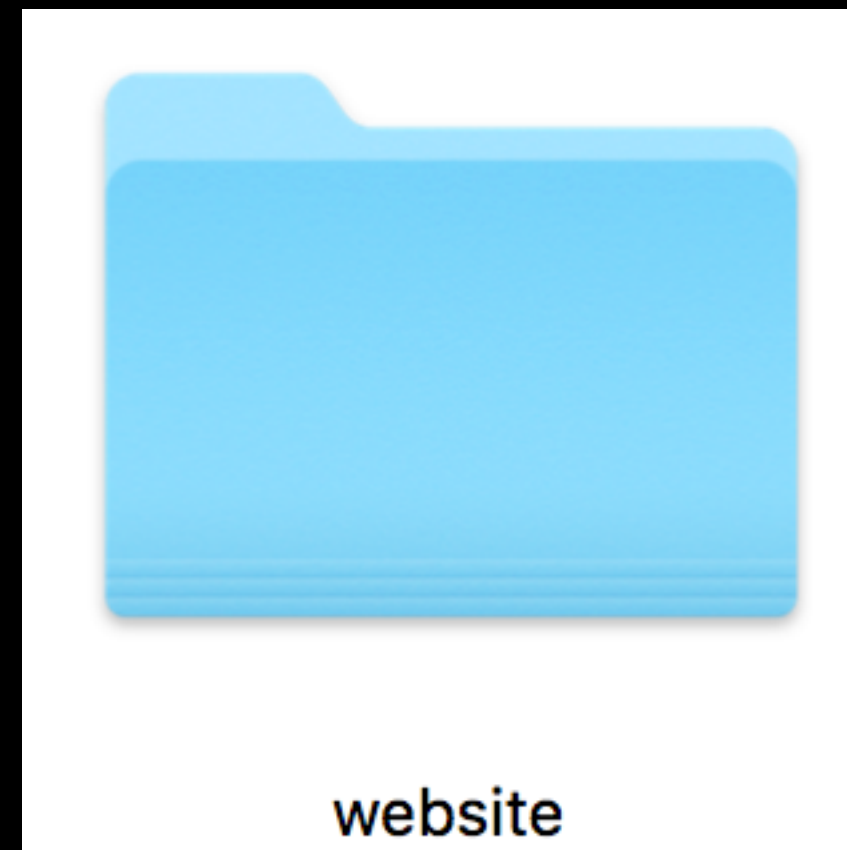
# Folder

Drag the entire folder
you just made on top of
the Text Wrangler icon
in the dock to launch it
again.
You will be able to see
the other folders you
made in the sidebar,
which makes it easy to
go back and forth
between documents.

# Soft wrap

You will find it more convenient to "soft wrap" the text in the Text Wrangler window, so you can see all the text on the page at the same time.

Go to Text Wrangler > Preferences, and choose Editor Defaults. Make the change you see here.
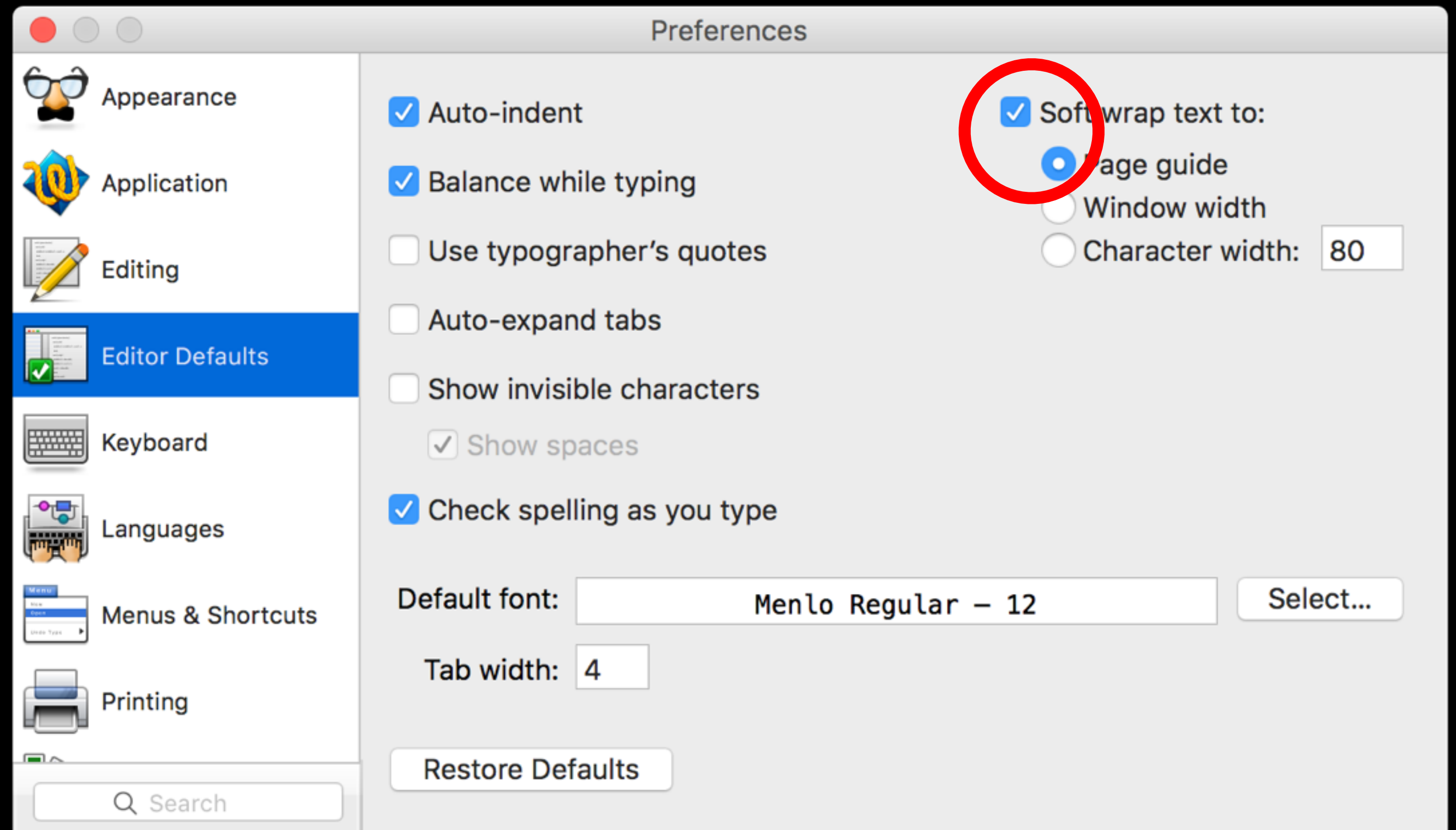
# The power of CSS

Using **Cascading Style Sheets** allows you to control every aspect of the visual presentation of your page *separately* from the content.

It steps between the HTML code and the browser, defining how each element should be rendered.

```
h1, p, a {
    border: 3px solid blue;
    margin: 10px;
    padding: 10px;
    width: 500 px;
    line-height: 150%;
    background-color: #9C0;
    color: white;
}
```

# The CSS box model

CSS imposes an imaginary **box** around each element and then enables you to format every presentational aspect of each element, redefining the default settings from HTML.

Top border

Top margin

Left border

Left margin

Left padding

Top padding

`<h1>...</h1>`

Right padding

Right margin

Right border

Bottom padding

Bottom margin

Bottom border
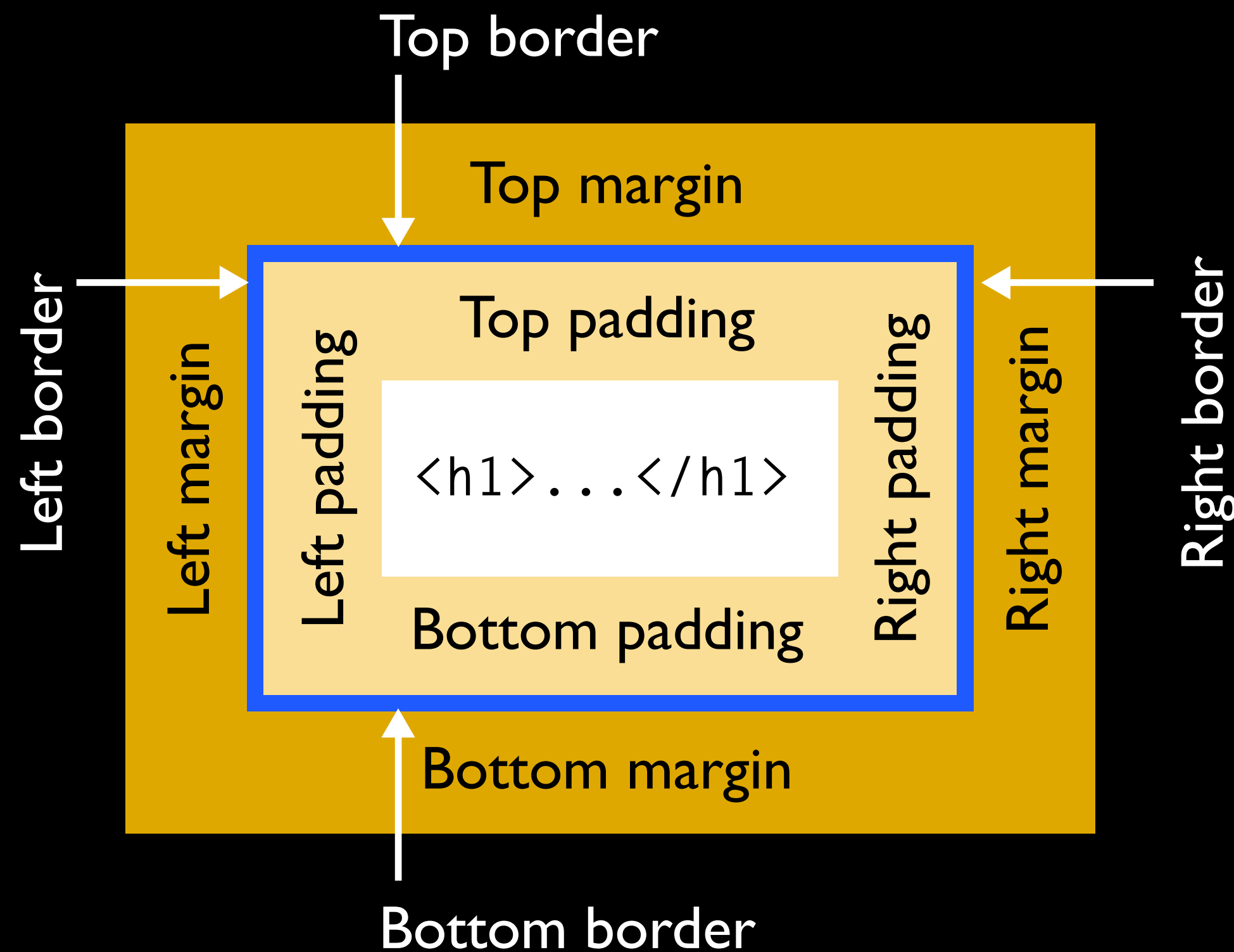
# How CSS is used

There are three ways in which style sheets can be associated with an HTML document:

**Inline styles** can be added within HTML elements as the **value** of a style attribute.

```
<p style="color: red">This is a bad idea.</p>
```

This is a bad idea.

# How CSS is used

You can do this on occasion, but as standard practice it is not recommended, as it fails to separate presentation from content.

```
<p style="color: red">This is a bad idea.</p>
```

This is a bad idea.

# How CSS is used

**Internal** CSS can be added to an HTML document's head, inside a style element, as we did a few minutes ago.

```
<style>
 p {color: red;}
</style>
```

Yet they too are subordinated to the totalitarian cult of personality of Big Brother, the deified Party leader who rules with a philosophy that decries individuality and reason as thoughtcrimes.

# How CSS is used

This is not highly recommended either, as it works only on a per-page basis. You would have to add a <style> tag to the head of every page in your website.

```
<style>
 p {color: red;}
</style>
```

Yet they too are subordinated to the totalitarian cult of personality of Big Brother, the deified Party leader who rules with a philosophy that decries individuality and reason as thoughtcrimes.

# How CSS is used

**External CSS** can be linked to an HTML document using a link element in the head.

This allows you to fully utilize the power of CSS, because a single CSS file can create the presentation for a site that has thousands of pages.

```
<link rel="stylesheet" href="style/style.css"
type="text/css" />
```

# CSS syntax

CSS is made up of statements — commonly referred to as **rules** — that tell the browser how to render the elements of an HTML document.

Each statement has two parts: a **selector** and a **declaration**.

A declaration has two parts: a **property** and a **value**.

SELECTOR

DECLARATION

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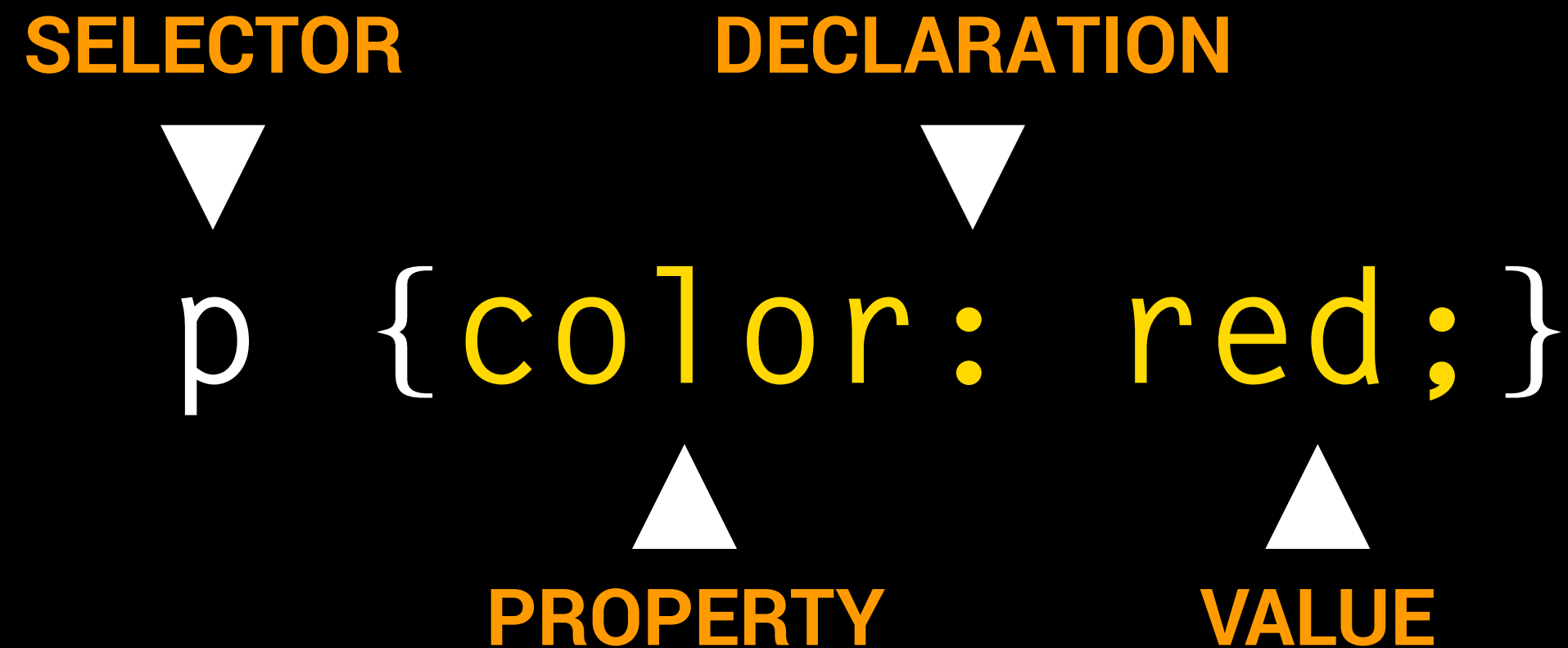
PROPERTY

VALUE

# CSS syntax

The declaration is contained
within a set of curly braces.

`p {color: red;}`

# CSS syntax

The declaration is contained within a set of curly braces.

The property is separated from the value with a colon.

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

## CSS syntax

The declaration is contained within a set of curly braces.

The property is separated from the value with a colon.

The declaration ends with a semicolon.

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

# CSS syntax

A rule can have one declaration or several. Several declarations are referred to as a **declaration block**. The declaration block is contained within opening and closing curly braces.

```css
p {
color: red;
font-weight: bold;
font-family: Helvetica, Arial, sans-serif;
}
```

# CSS selectors

The simplest kind of **selector** uses the name of the HTML element (as shown here) and selects ALL the elements of that type. So in this example, all the paragraphs in the document would be Helvetica, bold, and red.

```
p {
color: red;
font-weight: bold;
font-family: Helvetica, Arial, sans-serif;
}
```

**All my paragraphs are red.**

# CSS selectors

CSS rules can be used to define any single HTML element, or a group of elements (selectors) separated by commas.

```
p      {color: red;}
p, h1 {color: red;}
```

ELEMENT

MULTIPLE

# CSS selectors

Rules can be applied to classes and id's.

**Classes** are referenced with a period.

An **id** is referenced with a hash mark.

```
p       {color: red;} ————————————— ELEMENT
p, h1 {color: red;} ——————————— MULTIPLE
.orwell   {color: red;} ——————— CLASS
#animalfarm   {color: red;} ——— ID
```

# CSS selectors

A CSS rule can also be applied to descendants.

A **descendant selector** specifies an element that is nested within another element. Elements in a descendant selector are separated by a space.

In this descendant selector, all the paragraphs *within* elements that have a class of "orwell" will be red.

```
p       {color: red;} ——————————— ELEMENT
p, h1 {color: red;} ——————————— MULTIPLE
.orwell   {color: red;} ——————— CLASS
#animalfarm   {color: red;} ——— ID
.orwell p {color: red;} ——————— DESCENDANT
```

# Class and id

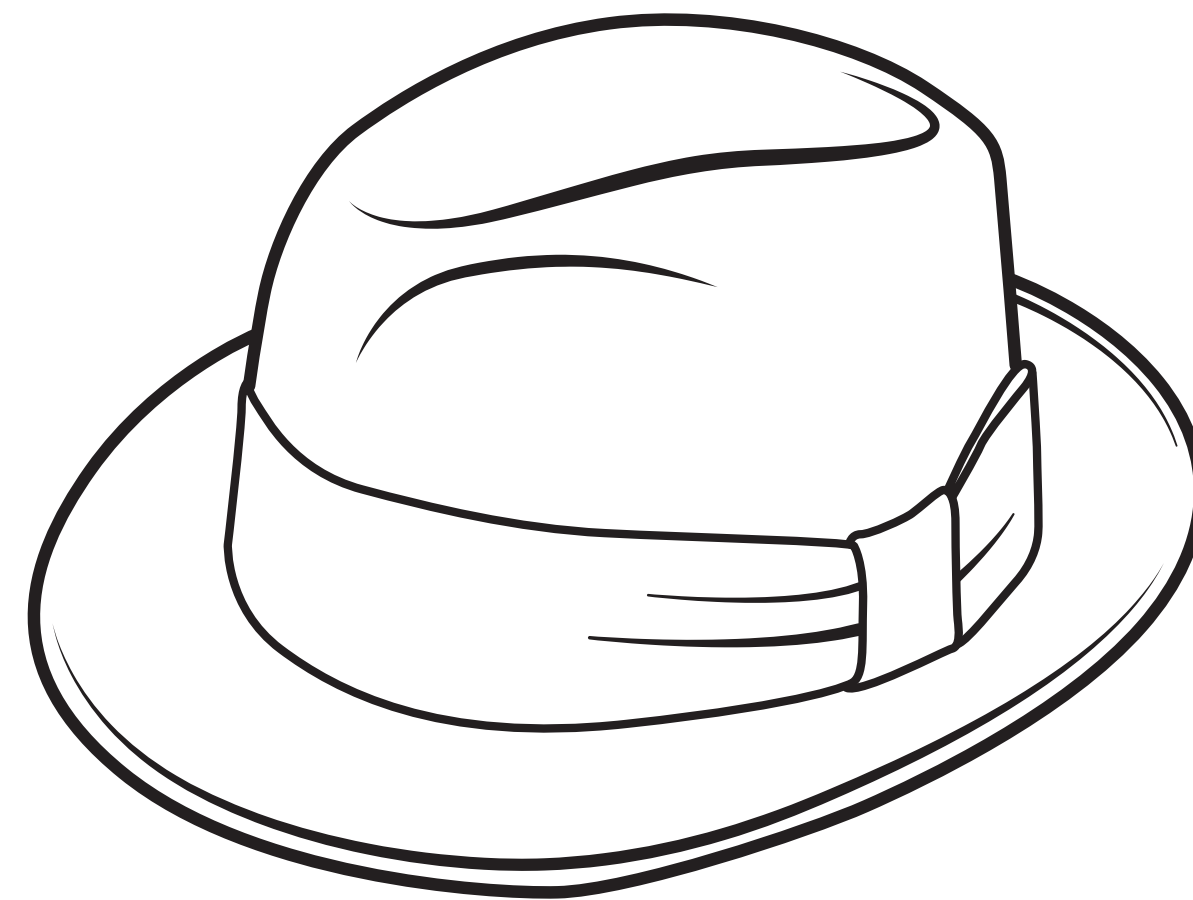Think of it this way. If hat was an HTML element, we would write it as

```
<hat>
```

This hat is actually a fedora; we could consider the fedora to be a class of hat:

```
<hat class="fedora">
```



A hat.

And this is not just any fedora, it's my fedora. That means I can give it an ID of ...

```
<hat class="fedora" id="steves">
```

# Google fonts

Go to Google Fonts

**https://www.google.com/fonts**

and find the font you want to use.

## Google Fonts

708 font families shown

Word | **Sentence** | Paragraph | Post

Preview Text: Grumpy wizards make toxic

**Filters:**

All categories ▾

▸ **Thickness**

▸ **Slant**

▸ **Width**

Normal 400

# Grumpy wizards ma

**Open Sans**, 10 Styles by Steve Matteson

**Script:**

Latin ▾

☒ Reset all filters/search

Normal 400

# Grumpy wizards ma

**Styles:**

⊡ Show all styles

**Roboto**, 12 Styles by Christian Robertson

# Google fonts

Click the button that says "Add to Collection." This works on a visit-by-visit basis rather than a collection you build over time.
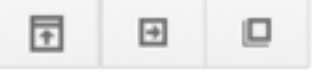
# Google fonts

Click the "Use" button at lower right when you have finished browsing for fonts. If you are adding more than one family to the page, it is best to do them together in a single line of code.



Thin 100
Grumpy wizards make toxic brew for the evil Queen and Jack.

Light 300
Grumpy wizards make toxic brew for the evil Queen and Jack.

Normal 400
Grumpy wizards make toxic brew for the evil Queen and Jack.

Bold 700
**Grumpy wizards make toxic brew for the evil Queen and Jack.**

**Roboto Slab**, 4 Styles by Christian Robertson          Add to Collection

Choose     Review     Use

# Google fonts

Follow the instructions on the next page:

1. Choose the weights you want.

2. Choose the alphabet (usually you will leave this box checked as it is by default).

3. Copy the <link> tag, which you will paste into the <head> of your HTML.

4. Copy the font-family rule, which you will use in your CSS.

1. Choose the styles you want:

⊟ Roboto Slab
  ☐ Thin 100                Grumpy wizards make toxic brew for the evil Queen and Jack.
  ☐ Light 300               Grumpy wizards make toxic brew for the evil Queen and Jack.
  ☑ Normal 400              Grumpy wizards make toxic brew for the evil Queen and Jack.
  ☐ **Bold 700**            **Grumpy wizards make toxic brew for the evil Queen and Jack.**

| Standard | @import | Javascript |

3. Add this code to your website:

```
<link href='https://fonts.googleapis.com/css?family=Roboto+Slab' rel='stylesheet' type='text/css'>
```

4. Integrate the fonts into your CSS:

The Google Fonts API will generate the necessary browser-specific CSS to use the fonts. All you need to do is add the font name to your CSS

```
font-family: 'Roboto Slab', serif;
```

# Google fonts

If you like, you can also download a version of your font to use in programs like InDesign or Illustrator. Just hit the tiny download button at top right of the page, and click on the .zip file link. The font will download, and you then can install it by decompressing it, and double-clicking the font itself.

## Download fonts

> ⚠ You do not need to download the font to use it on your webpages. Instead, refer to the "Use" section.

🔲 Download the font families in your Collection as a .zip file
§ Sync Google fonts to your desktop using SkyFonts
🏠 Download all font families, including source files, at the Google Fonts Github project

Close

Roboto Slab

Bold

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
1234567890

(Not Installed)     Install Font