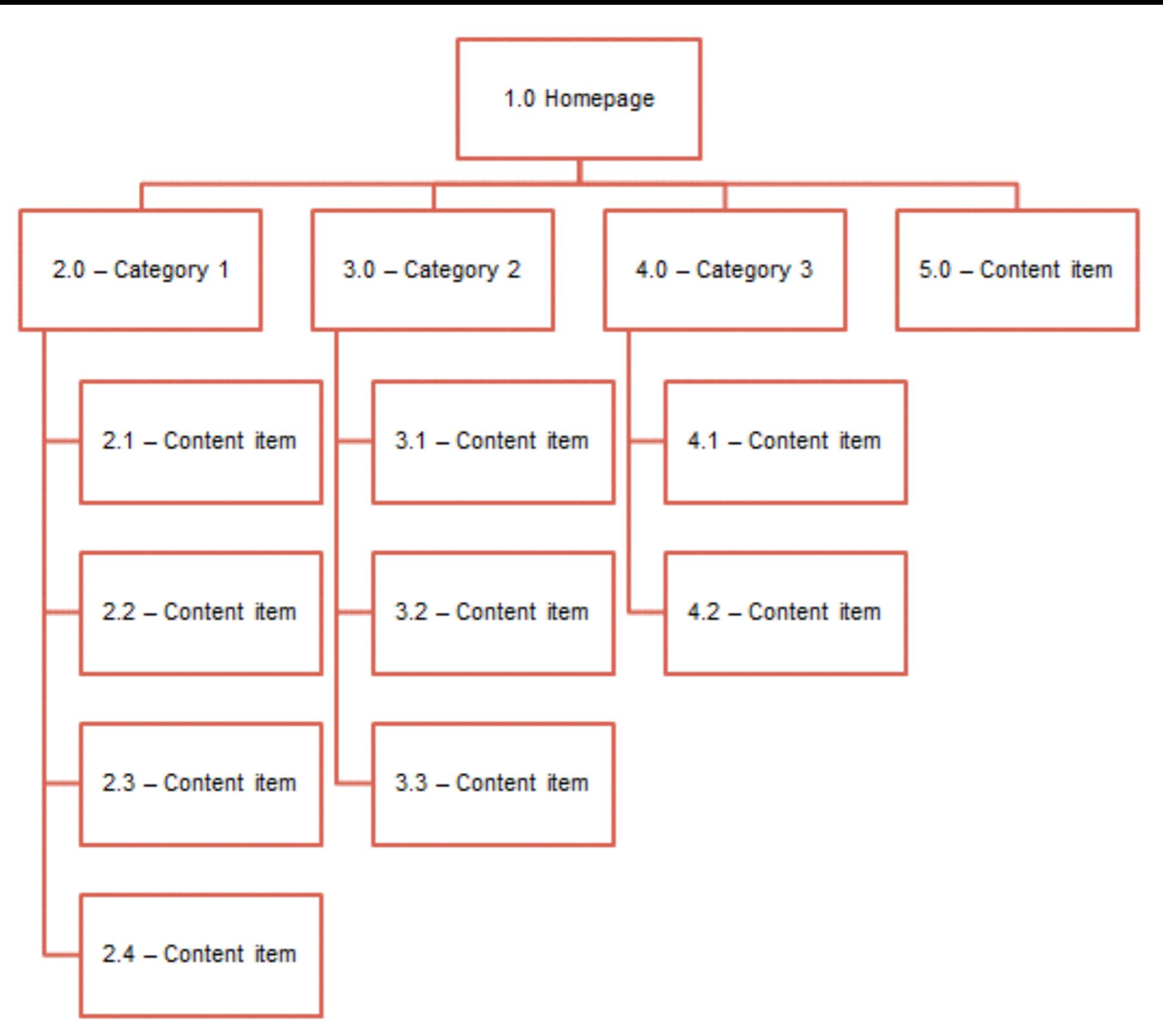
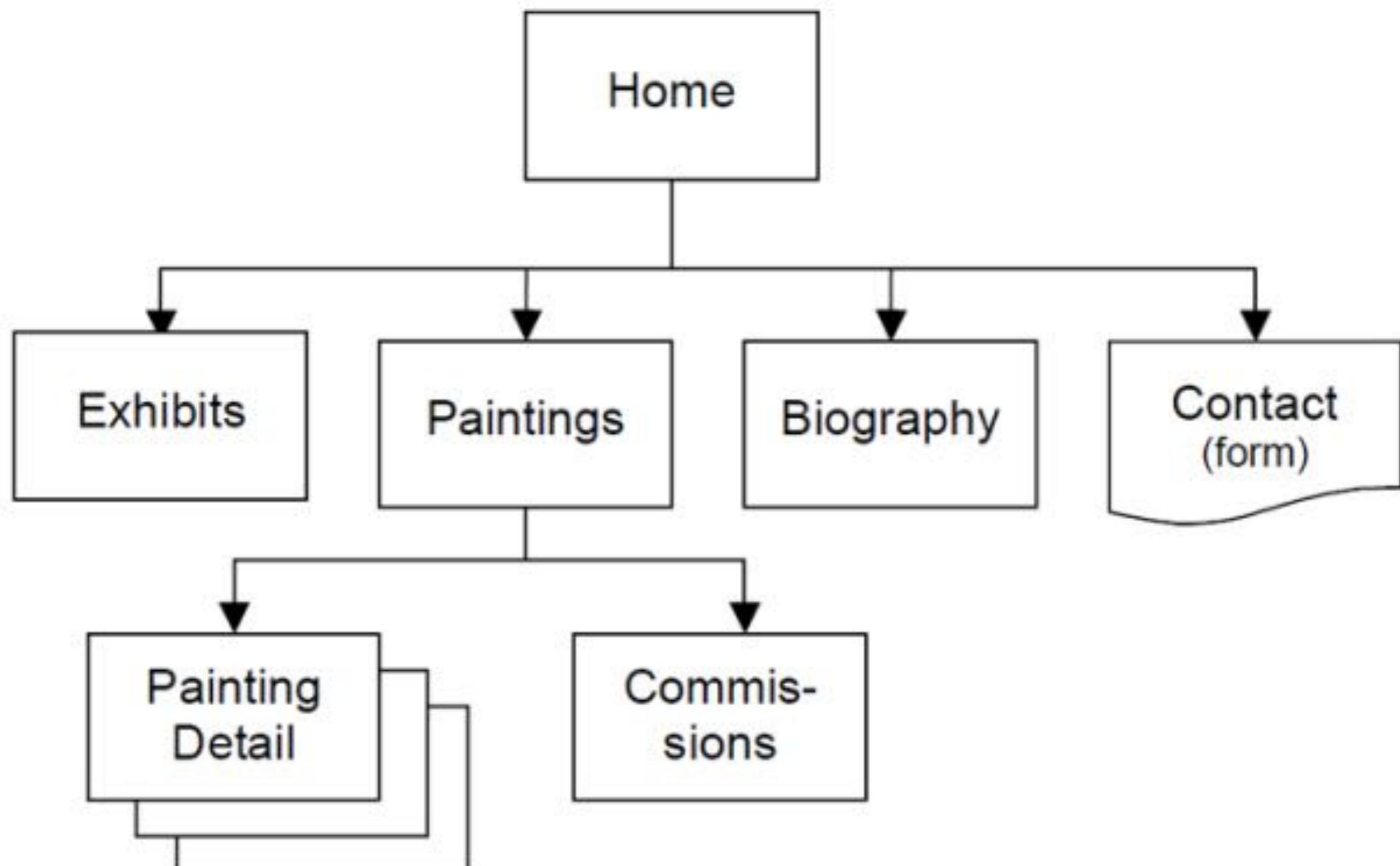


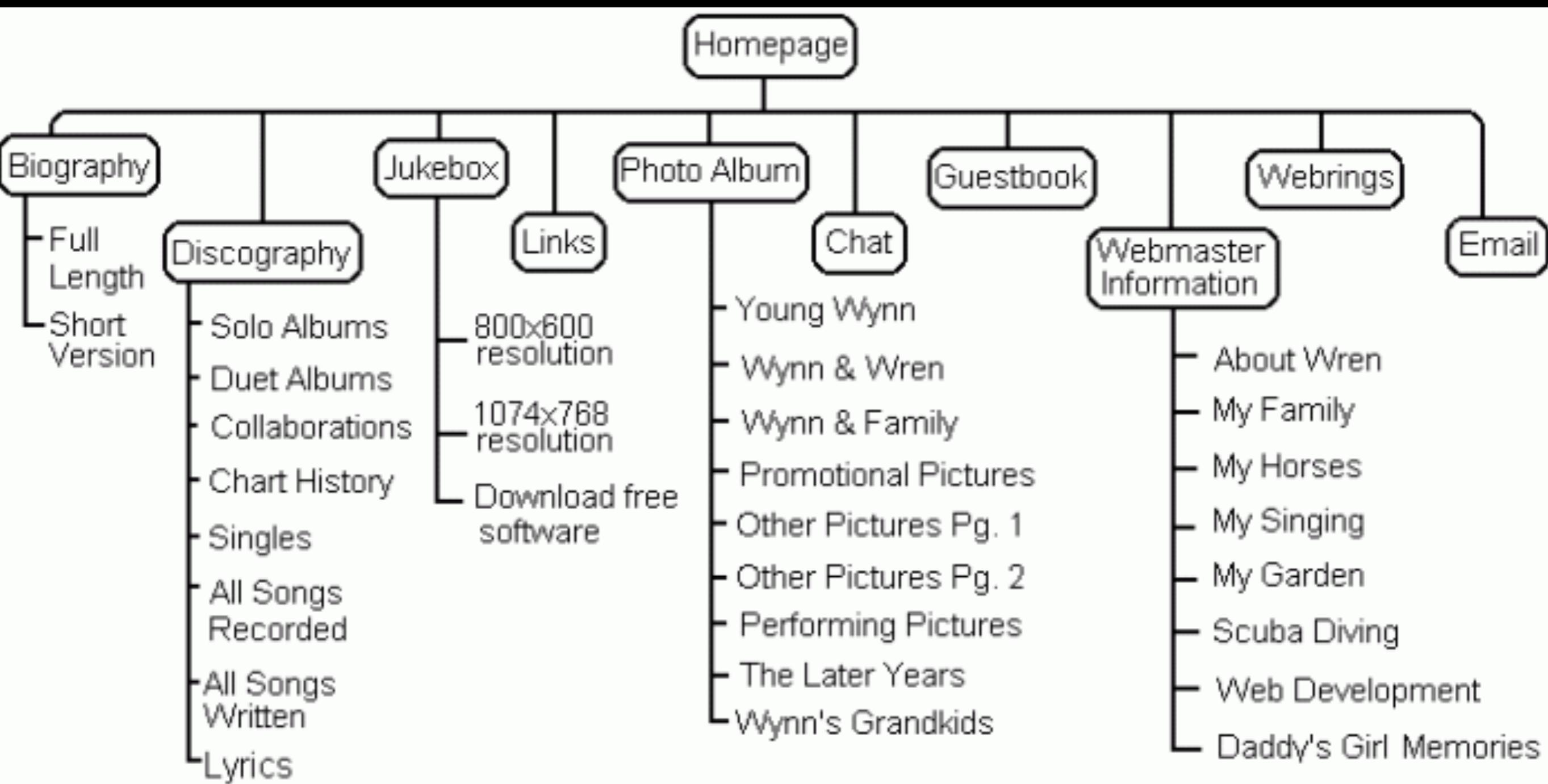
Running a local **HTTP server while developing is **VERY** important for designing for the web.**

- 1. It ensures all relative links will work when not running “locally” on yr machine**
- 2. Some interactivity will not work if not running from HTTP**
- 3. Web technologies change FAST, this way you will understand what’s going on “under the hood” + will be more able to adapt.**

designing for the web: site maps







A simple Sitemap depicting Website Global Navigation

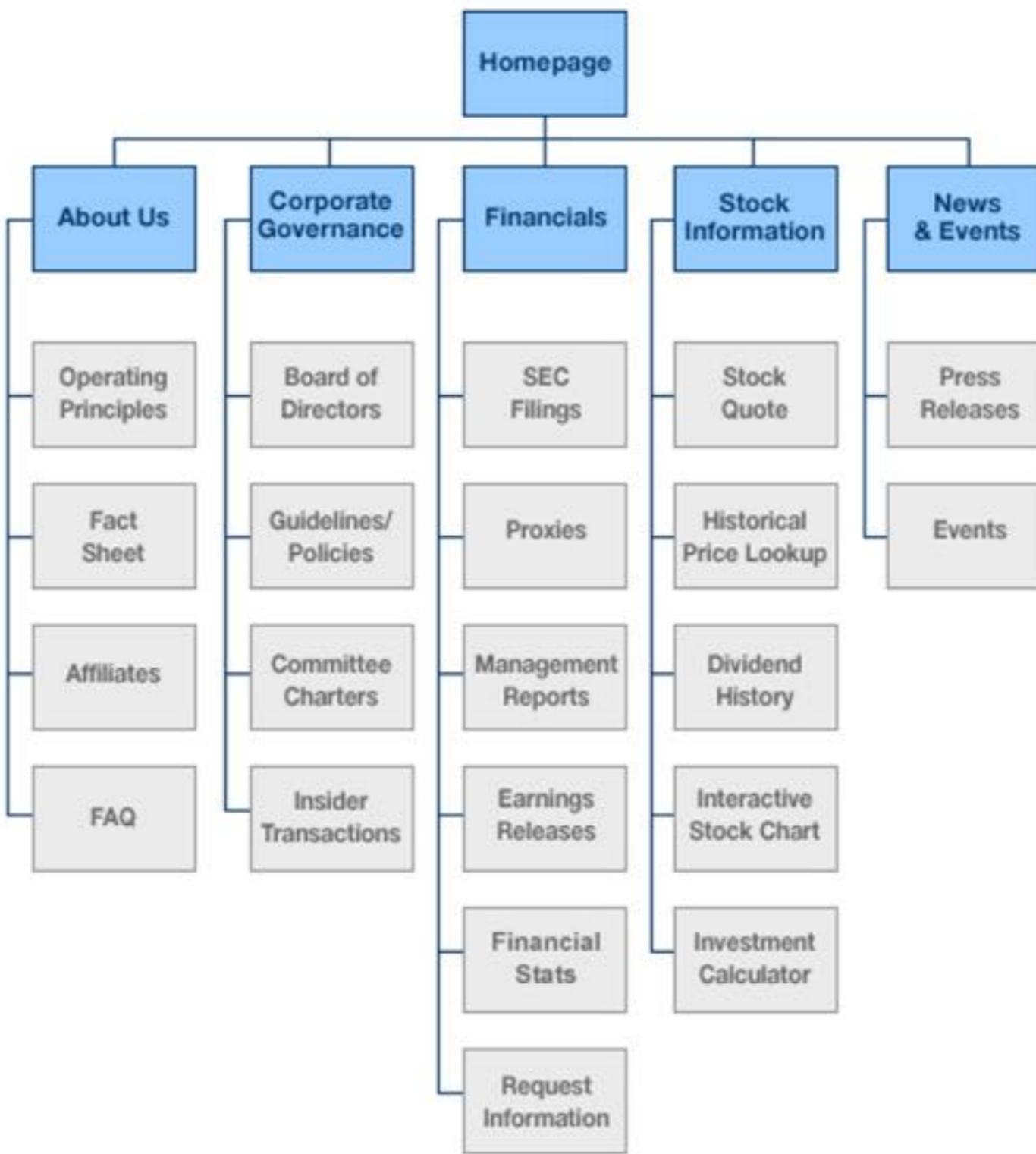
Links available from every page

[Site Map](#)

[Safe Harbor Statement](#)

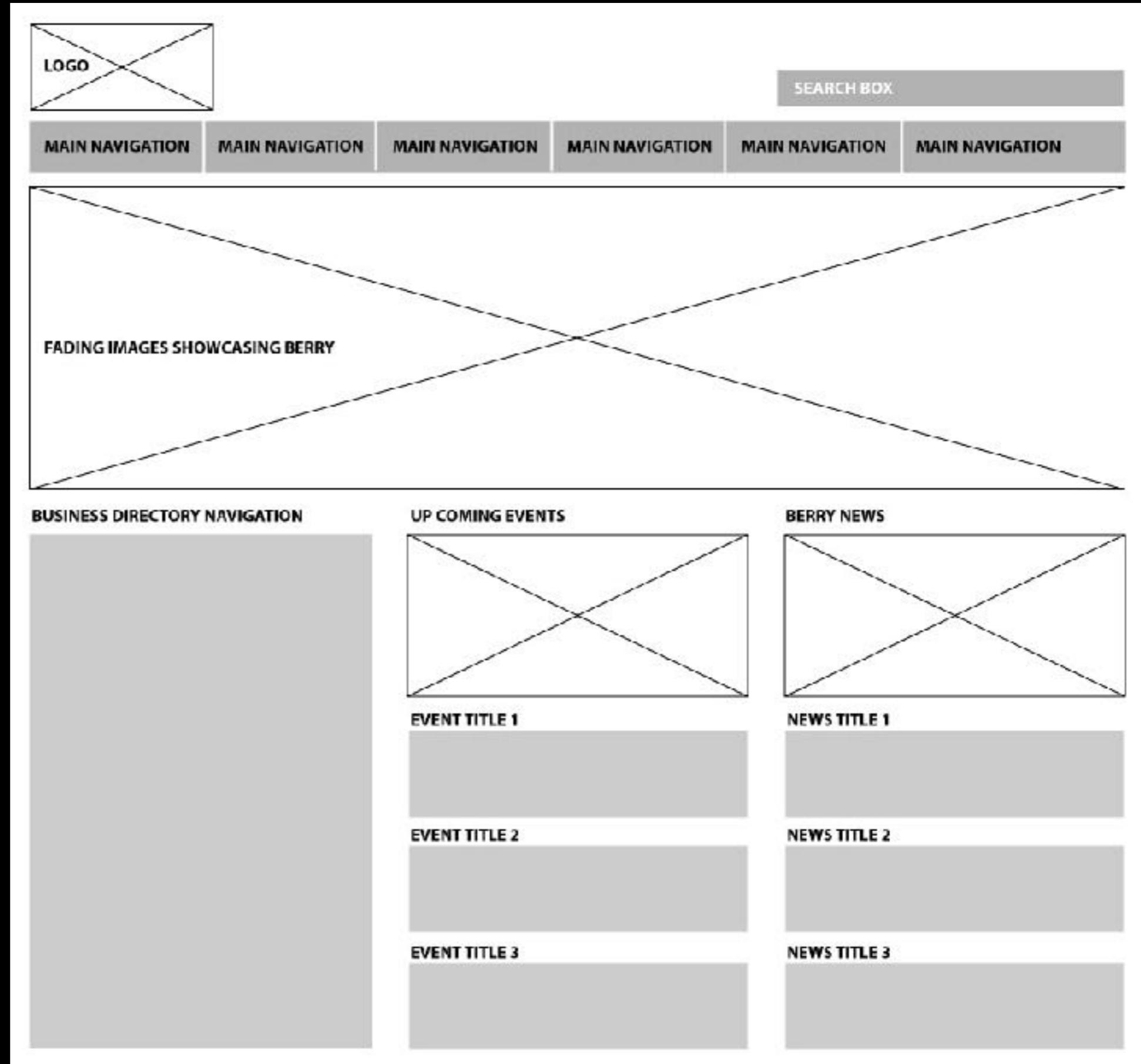
[Privacy Policy](#)

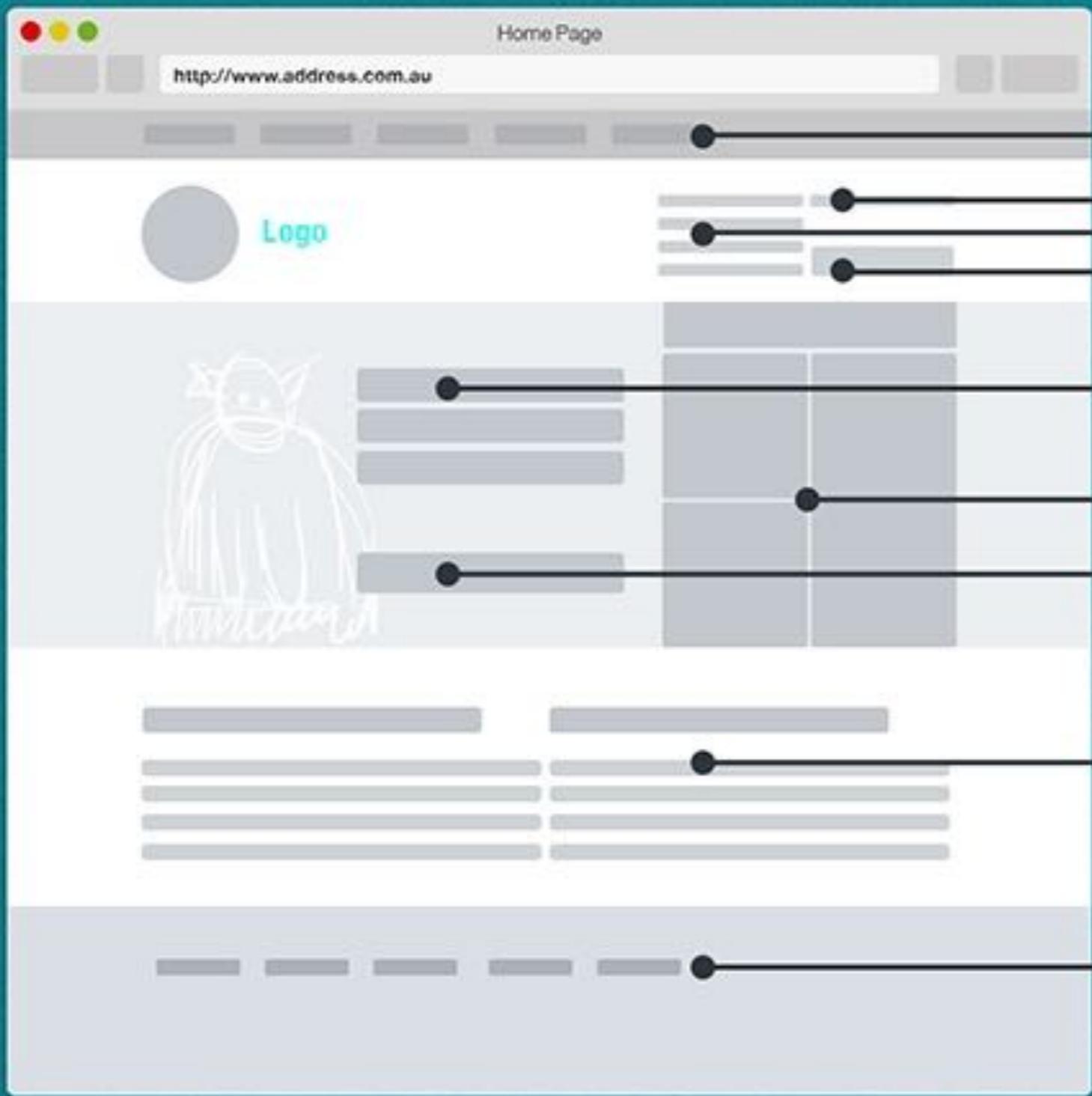
[Contact Us](#)



designing for the web: wire frames

Website Wire Frame





Navigation

Main navigation.

Quick Contact / Social Media

Phone/Address /Links to Social media accounts.

After Hours Highlight

Make people aware of your extended hours service.

Friendly Expertise Message

Reassuring people of your experience / care.

Links to Main Service

Graphical illustration / links to more content.

Quality / Take Action

Link, follow up on skills / expertise in the about us section

Two Highlight Boxes

Highlight pet care and expertise here, would be latest articles added to the news section

Footer Links

Contact / address / deeper linkage into website

intro 2 CSS

* which is super awesome

**bc CSS is all about:

DESIGN

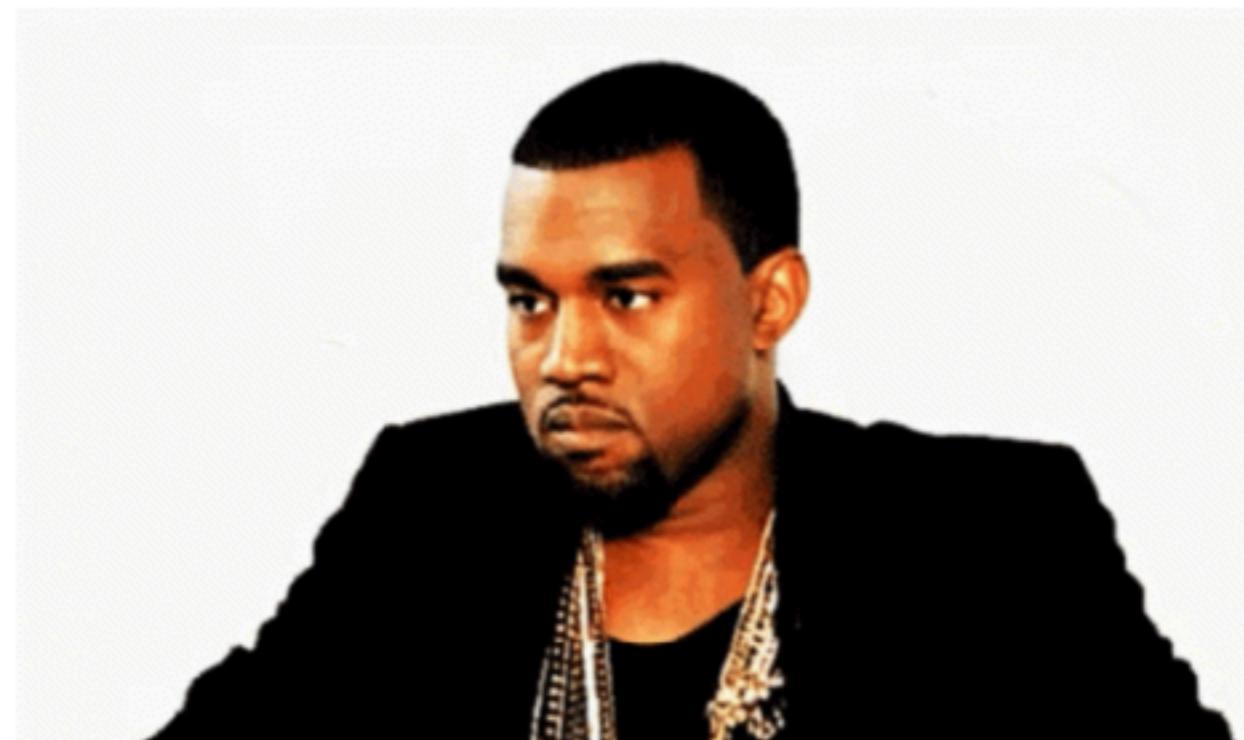
CSS works by associating rules with HTML elements. These rules govern how the content of specified elements should be displayed.

A CSS rule contains two parts: a selector and a declaration.

It takes 5% to learn how to write CSS rule and 95% to learn different properties that you can use.

the world w/out css

This Webpage though...



The key to understanding how **CSS** works is to imagine that there is an invisible box around every **HTML** element.

Block level elements are outlined w/ red + inline elements in green.

<body> creates 1st box, then **<h1>, <h2>, <p>, <i> + <a>** each create their own boxes within it.

The Cottage Garden

The *cottage garden* is a distinct style of garden that uses an informal design, dense plantings, and a mixture of ornamental and edible plants.

The Cottage Garden originated in [England](#) and its history can be traced back for centuries, although they were re-invented in 1870's England, when stylized versions were formed as a reaction to the more structured and rigorously maintained [English estate gardens](#).

The earliest cottage gardens were more practical than their modern descendants, with an emphasis on vegetables and herbs, along with some fruit trees.

The Cottage Garden

The *cottage garden* is a distinct style of garden that uses an informal design, dense plantings, and a mixture of ornamental and edible plants.

The Cottage Garden originated in [England](#) and its history can be traced back for centuries, although they were re-invented in 1870's England, when stylized versions were formed as a reaction to the more structured and rigorously maintained [English estate gardens](#).

The earliest cottage gardens were more practical than their modern descendants, with an emphasis on vegetables and herbs, along with some fruit trees.

Border

All boxes have borders even if invisible or 0px wide. It separates the edge of one box from another.

Margin

Margins sit outside the edge of the border. You can set the width to create a gap btw borders of adjacent boxes.

Padding

Padding is the space btw the border + any content contained within it. More padding increases the readability of its contents.

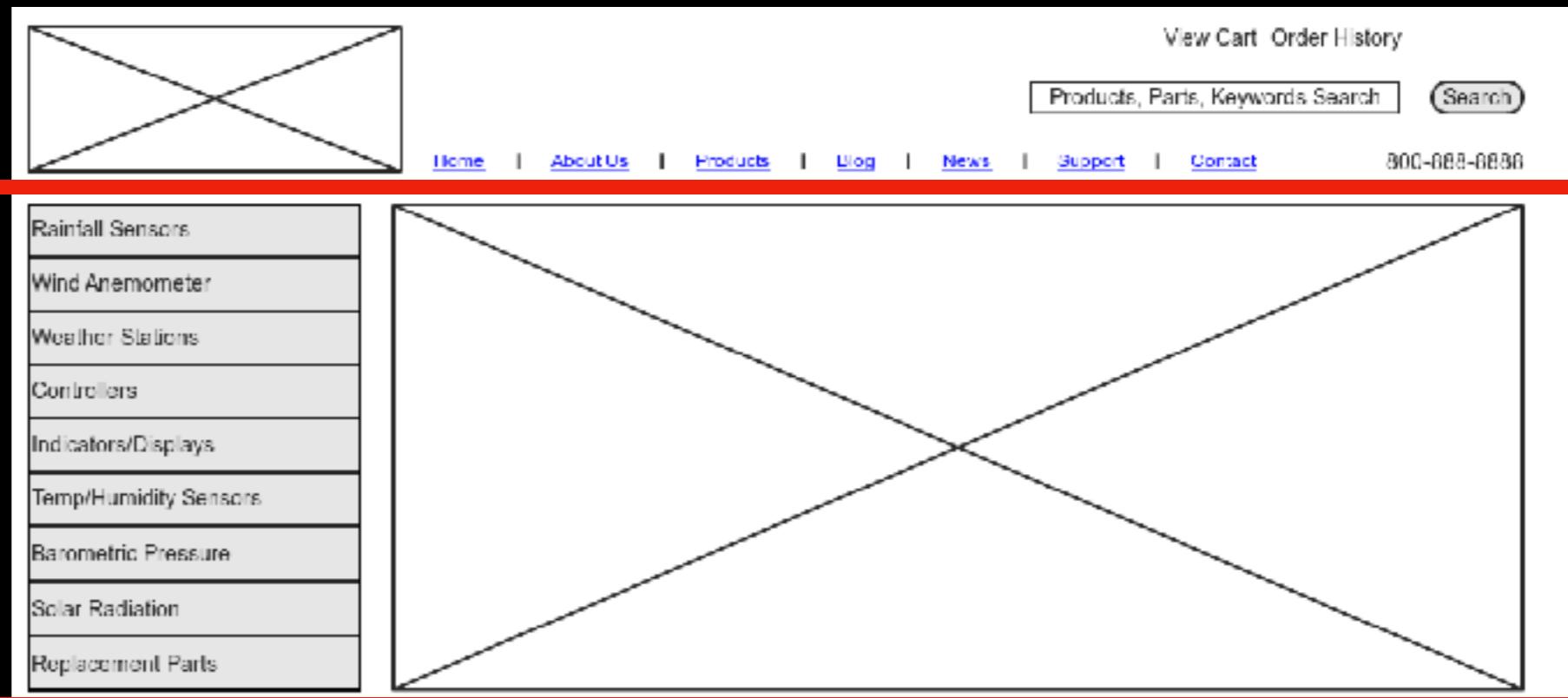


Containing Elements

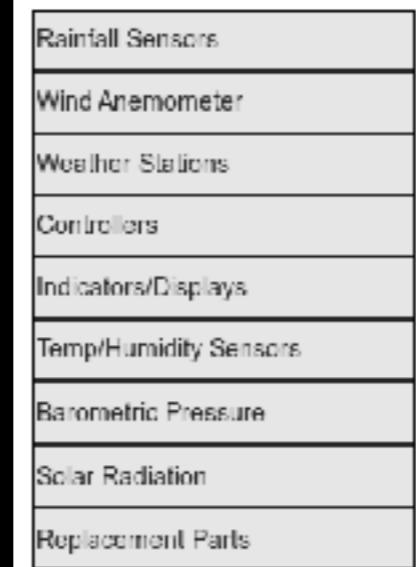
Block level

elements start on a new line – if a block-level element sits inside another then the outer box is the containing or parent element.

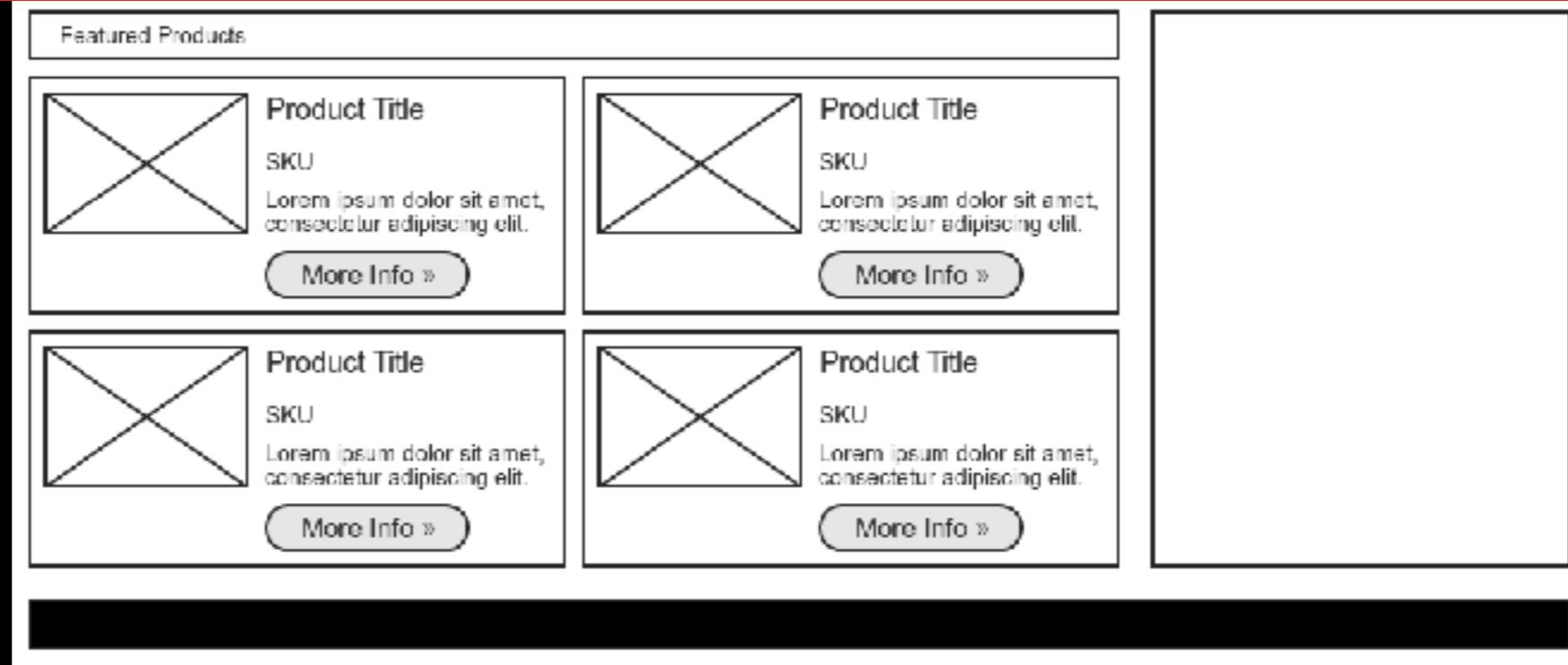
level 1



level 2



level 3

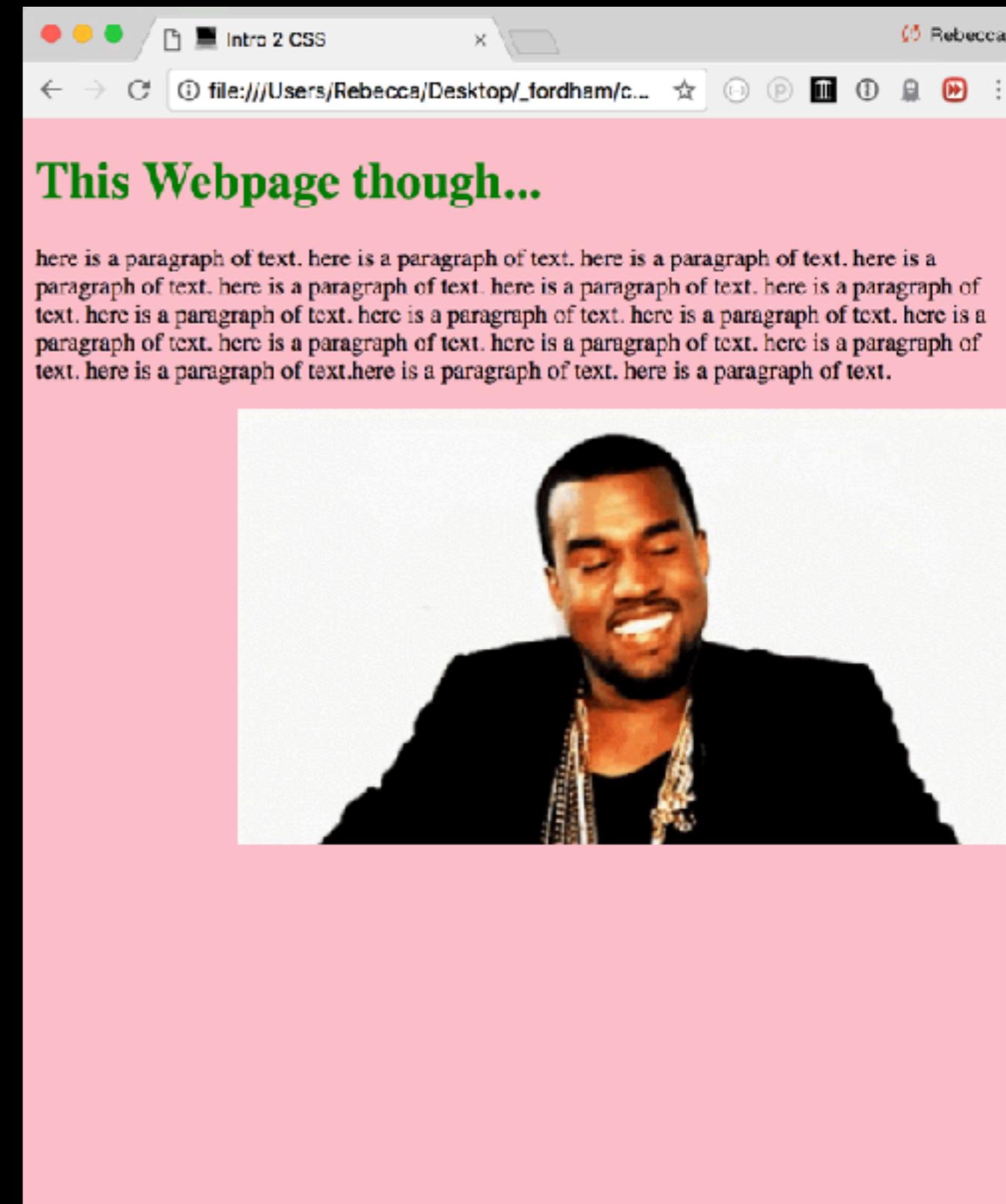


You can write CSS 3 Different Ways:

Inline Styles

Embedded Styles

Externals Styles



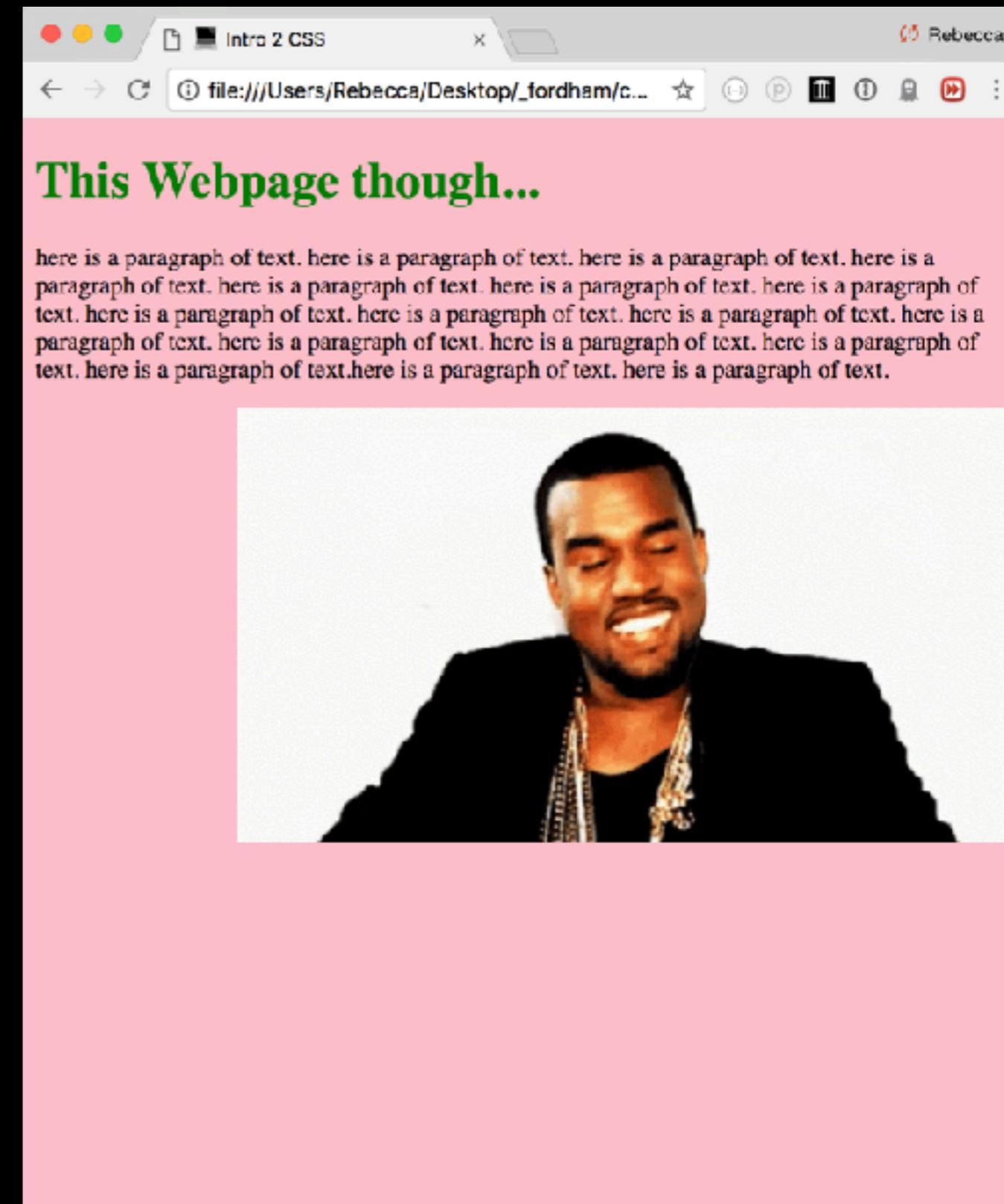
You can write CSS 3 Different Ways:

Inline Styles

```
<h1 style="color:green;">This Webpage though...</h1>
<body style="background-color: pink;">
```

Embedded Styles

Externals Styles



Inline Styles

```
<h1 style="color:white;">This Webpage though...</h1>
<body style="background-color: green;">
```

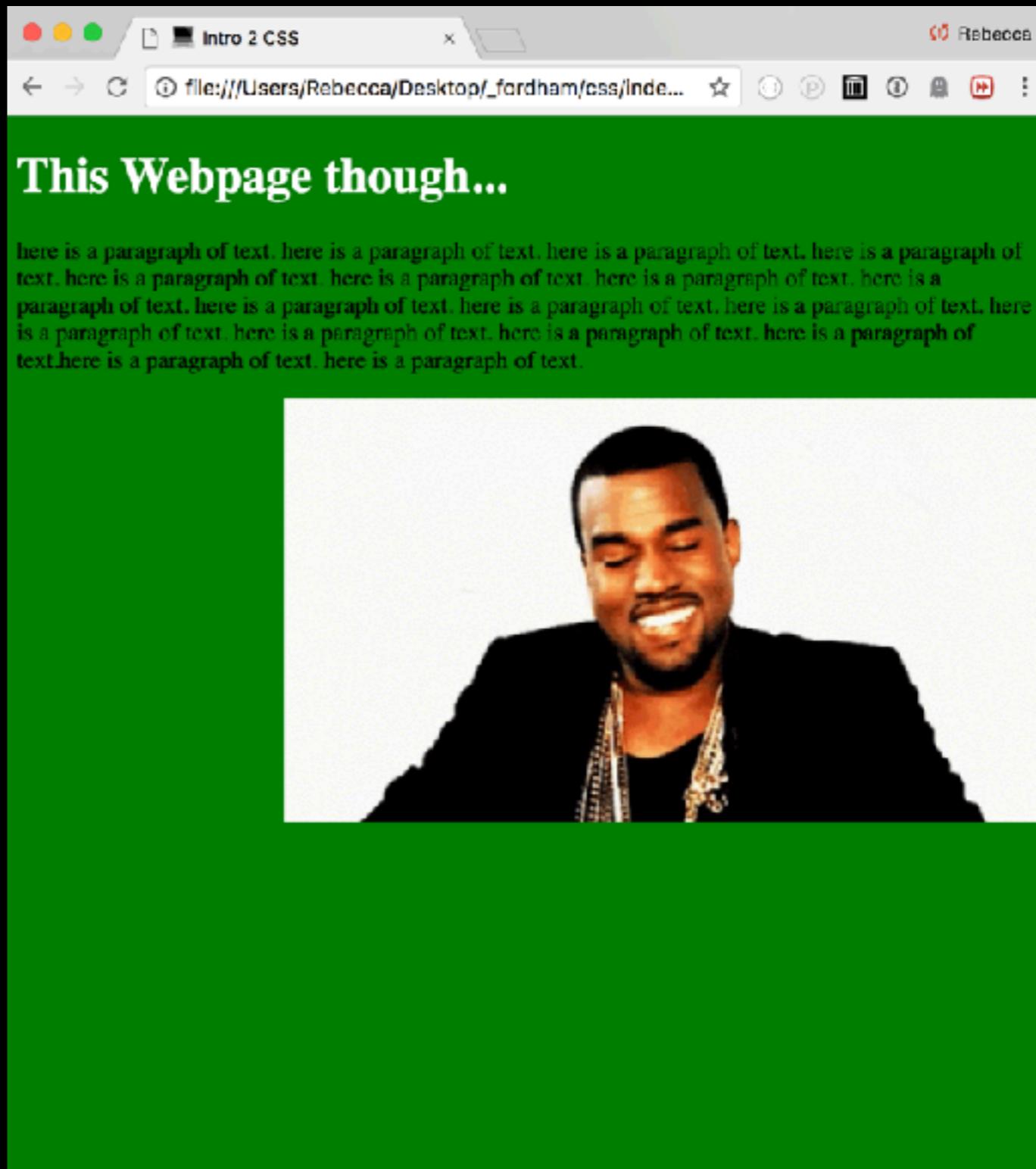
Embedded Styles

```
<html>
  <head>
    <title>  Intro 2 CSS </title>
    <style type="text/css">
      h1 {
        color: white
      }

      body{
        background: green;
      }

    </style>
  </head>
```

Externals Styles



Inline Styles

```
<h1 style="color:#FF4500;">This Webpage though...</h1>
<body style="background-color: #000080;">
```

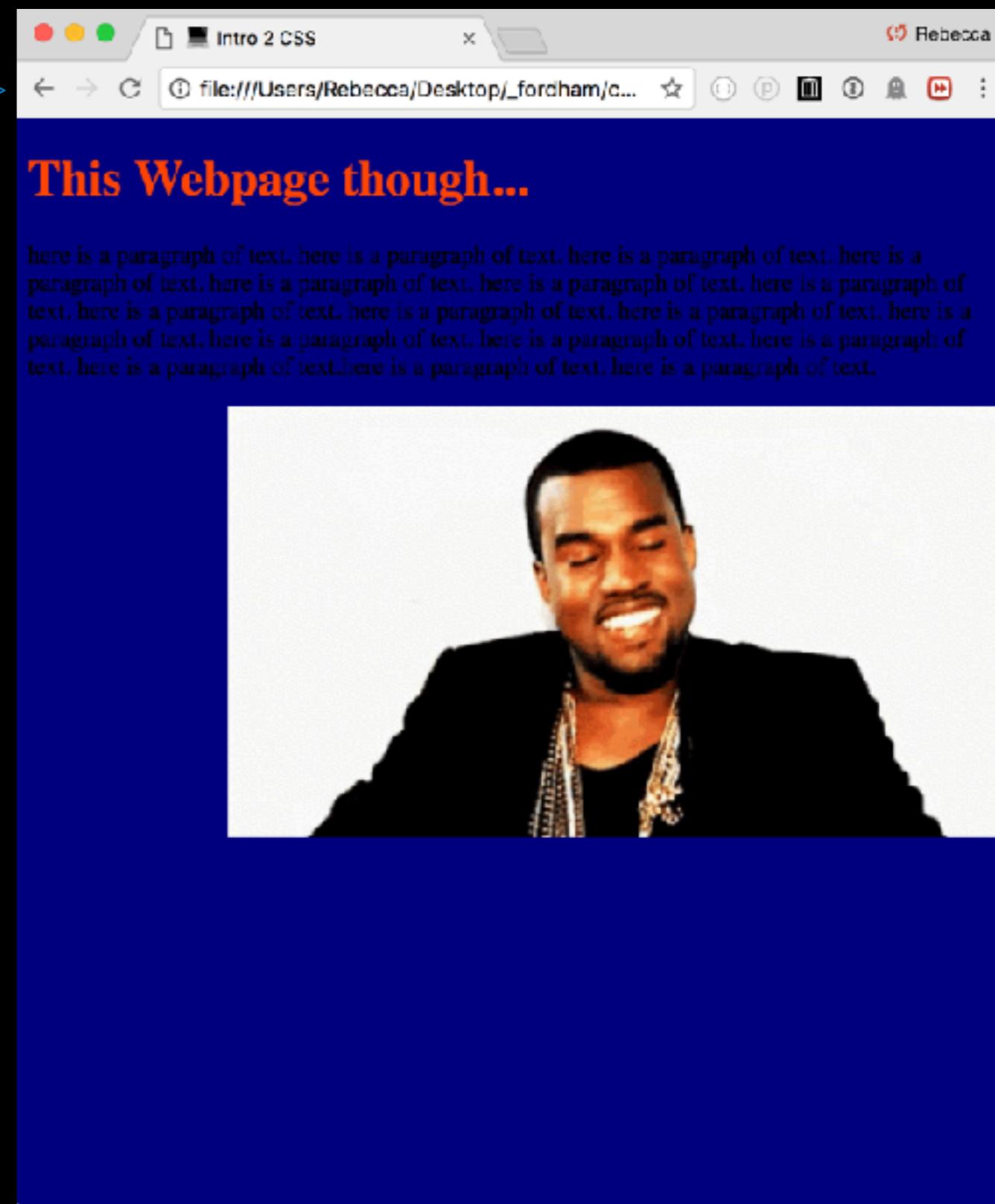
Embedded Styles

```
<html>
  <head>
    <title>  Intro 2 CSS </title>
    <style type= "text/css">
      h1 {
        color: #FF4500
      }

      body {
        background: #000
      }
    </style>
  </head>
```

External Styles *

```
<head>
    <title>  Intro 2 CSS </title>
    <link rel="stylesheet" type="text/css" href="theStyle.css">
</head>
```



style.css

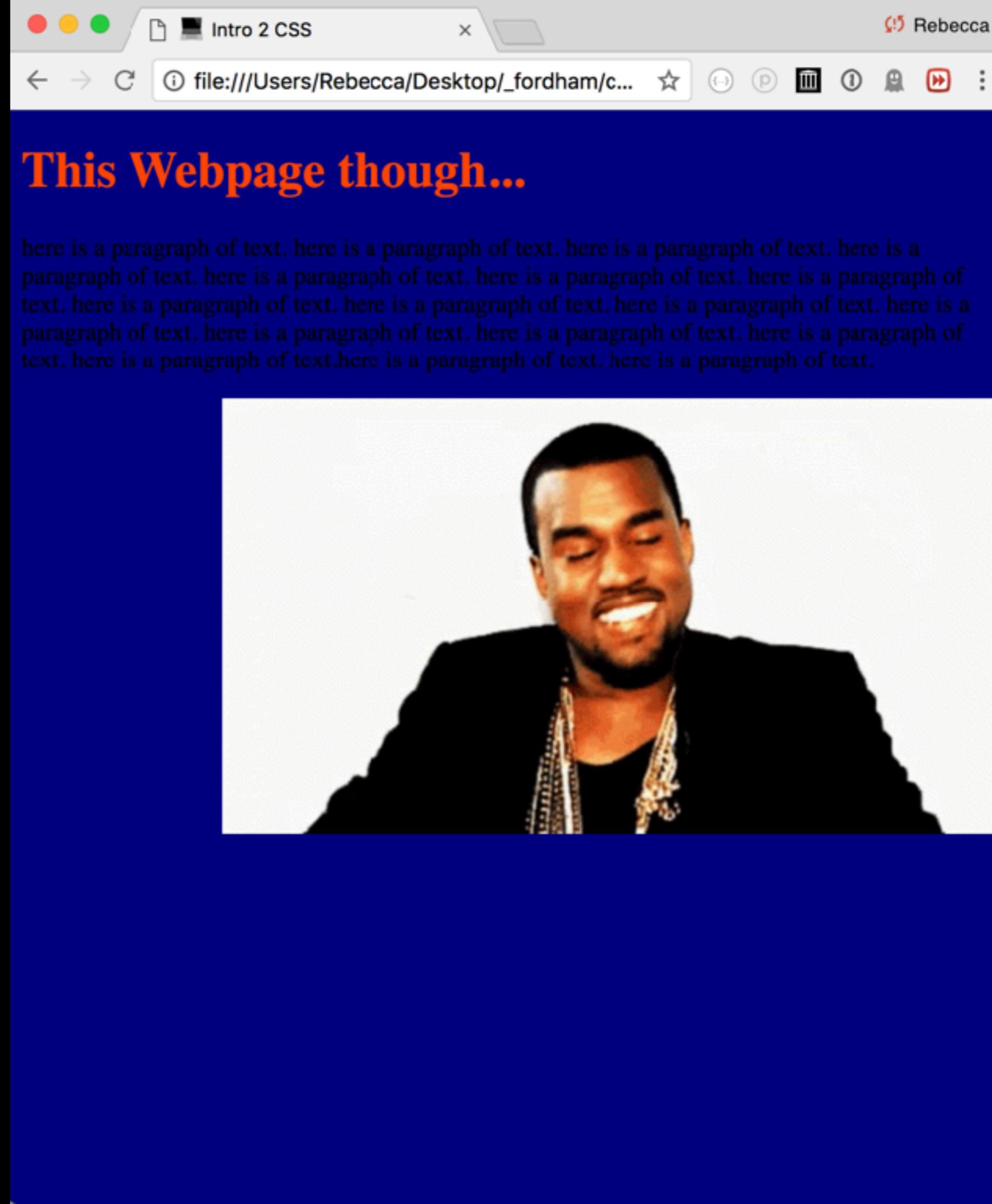
CSS SYNTAX:

selectors are used to find (select) HTML elements based on their element name, id, etc...

```
h1 {  
    color: #FF4500  
}
```

```
body {  
    background: #000080;  
}
```

```
selector { declaration }
```



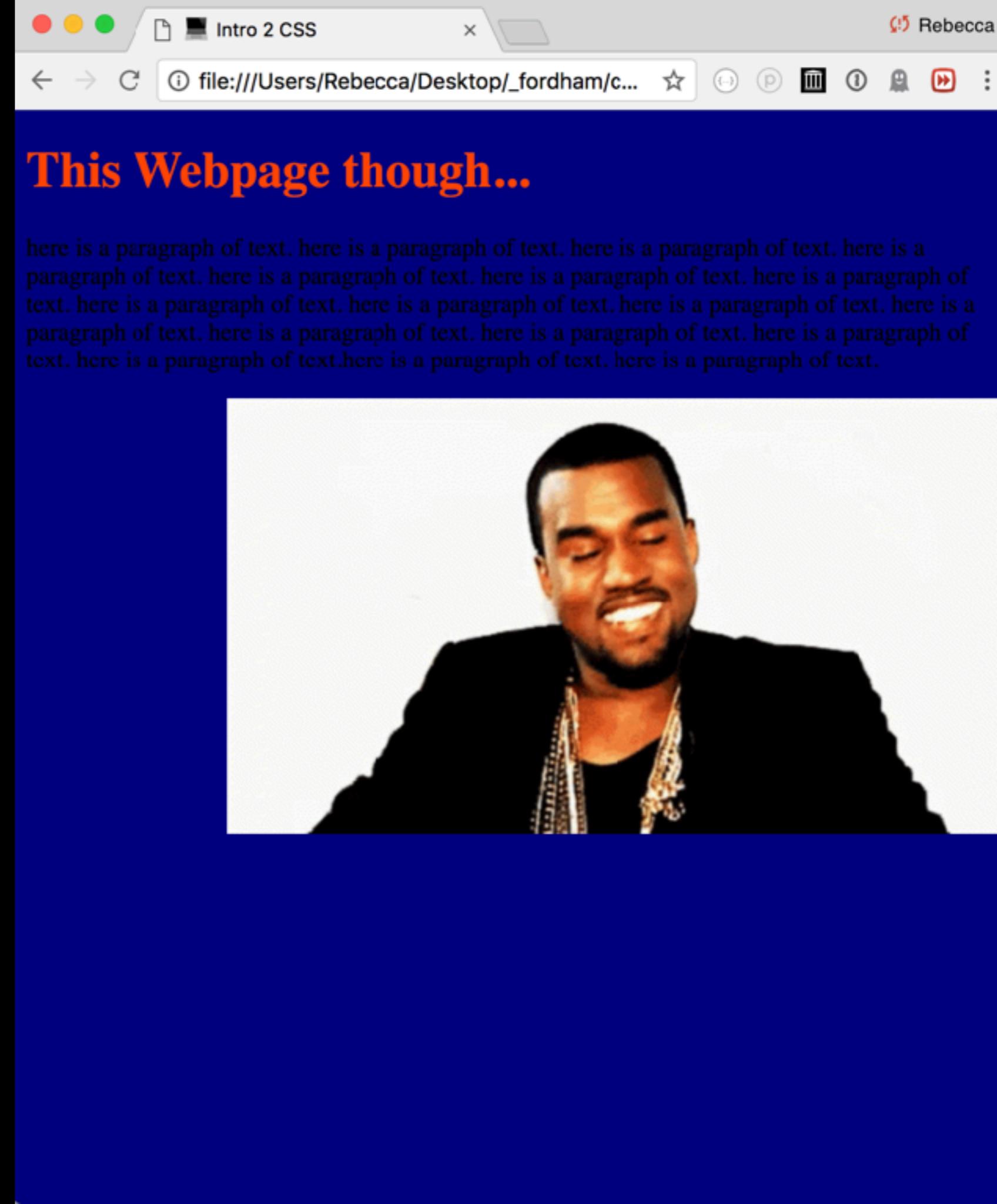
CSS SYNTAX:

selectors are used to find
(select) HTML elements
based on their element
name, id, etc...

```
h1 {  
    color: #FF4500  
}
```

```
body {  
    background: #000080;  
}
```

```
selector {  
  
    property: value ;  
  
}
```



Selector is a term such as **p**, **h1**, **div** that identifies the HTML element you want to format or apply a rule to. You can add multiple selectors in a declaration.

Selector	Meaning	Example
Universal Selector	Applies to all elements in the document	* { }
Type Selector	Matches element names	h1, h2, h3 { }
Class Selector	Matches an element whose class attribute has a value that matches the one specified after the period (or full stop) symbol	<p>.note {} targets any element whose class attribute has a value of "note}</p> <p>p.note {} targets only <p> elements whose class attribute has a value of "note"</p>
ID Selector	Matches an element whose id attribute has a value that matches then specified after the # symbol	<p>#introduction {} targets the element whose id attribute has value of "introduction"</p>

Selector

Meaning

Example

Child Selector

Matches an element that is a direct child of another

li > a {}
targets any `<a>` element that are children of an `` element (but not other `<a>` elements in the page).

Descendant Selector

Matches an element that is a descendent of another specified element (not just a direct child of that element)

p a {}
targets any `<a>` elements that sit inside a `<p>` element, even if there are other elements nested btw them

Selector	Meaning	Example
----------	---------	---------

Adjacent Sibling Selector

Matches an element that is the **next sibling of another**

`h1+p {}`
targets the first `<p>` element after any `<h1>` element (but not other `<p>` elements)

General Sibling Selector

Matches an element that is a **sibling of another, although it does not have to be the directly preceding element**

`h1~p {}`
if you have two `<p>` elements that are siblings of an `<h1>` element, this rule would apply to both

```
/* type/element selector */
```

```
p {  
    color: blue;  
}
```

```
/* class attribute selector */
```

```
.blue-text {  
    color: blue;  
}
```

```
/* id attribute selector */
```

```
#blue-par {  
    color: blue;  
}
```

```
/* BONUS: grouping  
selector */
```

```
p,  
.blue-text,  
#blue-par {  
    color: blue;  
}
```

selecting multiple elements:

```
h1, h2, h3 {
```

```
  color: red;  
  background-color: blue;  
  width: 500px;
```

```
}
```

```
p,
```

```
li {
```

```
  background-color: red;  
  font-color: blue;
```

```
}
```

Units + Dimensions

Length & Size

The most common units we'll be using for setting the of an element or property are:

- `px`
- `%`
- `em`
- `rem`

HTML comments are written like this

<!-- This is a comment -->

CSS comments are written like this

/* This is a comment */

Project — ~/Desktop/teach/frdhm/_F_2018/week04

Project — ~/Desktop/teach/frdhm/_F_2018/week04

index.html style.css

```
4  <title>Week 4</title>
5  <!-- you will need to link to the css file -->
6  <link href="style.css" rel="stylesheet"
7   type="text/css">
8  </head>
9
10 <body>
11   <h2>Intro to CSS</h2>
12   <h3> Welcome to Color! </h3>
13   <h4>where things get real</h4>
14   <div>
15     <ol>
16       <li>CSS treats each HTML element as if it
17        appears inside its own box + uses rules to
18        indicate how that element should look</li>
19       <li>Rules are made up of selectors (that specify
20        the elements the rule applies to) + declarations
21        (that indicate what these elements should look
22        like).</li>
23       <li>Different types of selectors allow you to
24        target your rules at different elements.</li>
25       <li>Declarations are made up of two parts: the
26        properties of the element that you want to
27        change, and the values of those properties. For
28        example, the font-family property sets the choice of
29        font, and the value arial specifies Arial as
        the preferred typeface.</li>
       <li>CSS rules usually appear in a separate
        document, although they may appear within an
        HTML page.</li>
     </ol>
   </div>
   <div class="theAuthor">
     — from John Duckett's <span>HTML + CSS</span>
   </div>
</body>
</html>
```

Week 4

localhost:12345

INTRO TO CSS

Welcome to Color!

where things get real

1. CSS treats each HTML element as if it appears inside its own box + uses rules to indicate how that element should look.
2. Rules are made up of selectors (that specify the elements the rule applies to) + declarations (that indicate what these elements should look like).
3. Different types of selectors allow you to target your rules at different elements.
4. Declarations are made up of two parts: the properties of the element that you want to change, and the values of those properties. For example, the font-family property sets the choice of font, and the value arial specifies Arial as the preferred typeface.
5. CSS rules usually appear in a separate document, although they may appear within an HTML page.

— from John Duckett's *HTML + CSS*

style.css — ~/Desktop/teach/frdhm/_F_2018/week04

style.css — ~/Desktop/teach/frdhm/_F_2018/week04

Project

Index.html

style.css

```
2
3 h2 {
4   font-family: Times;
5   background: DarkCyan;
6   /* lowercase + capitalize */
7   text-transform: uppercase;
8   text-decoration: underline;
9   letter-spacing: 0.2em;
10 }
11
12 h3 {
13   background: rgba(238,62,200);
14   font-weight: bold;
15 }
16
17 h4 {
18   background: rgba(130,62,200);
19   font-size: 75%;
20   text-decoration: line-through;
21   word-spacing: 1em;
22 }
23 body {
24   font-family: Avenir;
25   font-size: 15px;
26   background: rgb(200,200,200);
27 }
28
29 div{
30   background: rgb(255,255,255);
31   line-height: 1.8em;
32 }
33
34 .theAuthor{
35   background: rgb(255,255,255);
36   text-align: center;
37 }
38 span {
39   font-style: italic;
40 }
```

style.css 36:20

INTRO TO CSS

Welcome to Color!

where things get real

1. CSS treats each HTML element as if it appears inside its own box – uses rules to indicate how that element should look.
2. Rules are made up of selectors (that specify the elements the rule applies to) + declarations (that indicate what these elements should look like).
3. Different types of selectors allow you to target your rules at different elements.
4. Declarations are made up of two parts: the properties of the element that you want to change, and the values of those properties. For example, the font-family property sets the choice of font, and the value arial specifies Arial as the preferred typeface.
5. CSS rules usually appear in a separate document, although they may appear within an HTML page.

-- from John Duckett's HTML + CSS

```
{  
text-align:  
  
    left ;  
    right ;  
    center ;  
    justify ;  
  
}
```

```
{
```

```
vertical-align:
```

```
baseline ;
```

```
sub ;
```

```
super ;
```

```
top ;
```

```
text-top ;
```

```
middle ;
```

```
bottom ;
```

```
text-bottom ;
```

```
}
```

This property is NOT intended to allow you to vertically align text in the middle of a block level elements such as `<p>` + `<div>`, although it does have this effect when used with table cells `<td>` + `<th>` elements.

It is more commonly used w/ inline elements such as ``, `` or ``. When used with these elements, it performs a task very similar to the HTML align attribute used on the `` element.

Interaction Design

a: link {

a: visited {

: hover { Applied when a user hovers over an element w/ a mouse. This changes the appearance of links and buttons when a user places their cursor over them.
Does not work on mobile.

: active { Applied when an element is bingo activated by a user, like when a button is pressed or a link clicked.
This added to UX.
Applied when an element has focus. Any thing you can interact with.

: focus { Focus occurs when a browser discovers that you are ready to interact w/ an element. For example when yr cursor is in an input - that element is said to have focus.

}

More on Cascading

From John Duckett book:

Last rule

If the two selectors are identical, the latter of the two will take precedence. For example, if there were two **i** elements in style sheet, the second one would take precedence over the first.

Specificity

If one selector is more specific than the others, the more specific rule will take precedence over more general ones. For example, **h1** is more specific than body tag and so one.

Important

You can add **!important** after any property value to indicate that it should be considered more important than other rules that apply to the same element.

Classes and IDs

Two common attributes used to single out certain HTML elements are **class** and **id**, both are used to identify particular elements when adding CSS styling rules. **You author class + id names!!** They have no particular meaning in themselves, besides a puzzle - or code - you are creating.

Use a **class** when you have more than one element you want to share the same styling - perhaps across multiple pages.

Use an **id** when there is only one element on the page with that id, for example id="header" With a class you can have as many elements with that styling as you like.

An element can have more than one **class**, but not more than one **id**. When there is more than one class, the class names are separated by spaces.

```
<h1 id="myHeader">Hello World!</h1>
```

IDs

Every HTML element can carry the id attribute. It is used to uniquely identify that element from other elements on the page.

Its value should start with a letter or an underscore (not a number or any other character). It is important that no two elements on the same page have the same value for their id attributes (otherwise the value is no longer unique).

More to read on ID naming: <https://mathiasbynens.be/notes/css-escapes>

IDs

To select these IDs in CSS
you would do so with
#myHeader syntax

(IDs may become particularly useful when it comes to media elements - photos, videos + sound files.)

```
#myHeader{  
    color: blue;  
}
```

Classes

Every HTML element can also carry a **class** attribute. Sometimes, rather than uniquely identifying one element within a document, you will want a way to identify several elements as being different from the other elements on the page.

```
<div class="cities">
<h2>London</h2>
<p>London is the capital of England. It is the most populous city in the United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>
<p>Standing on the River Thames, London has been a major settlement for two millennia, its history going back to its founding by the Romans, who named it Londinium.</p>
</div>

<div class="cities">
<h2>Paris</h2>
<p>Paris is the capital and most populous city of France.</p>
<p>Situated on the Seine River, it is at the heart of the Île-de-France region, also known as the région parisienne.</p>
<p>Within its metropolitan area is one of the largest population centers in Europe, with over 12 million inhabitants.</p>
</div>

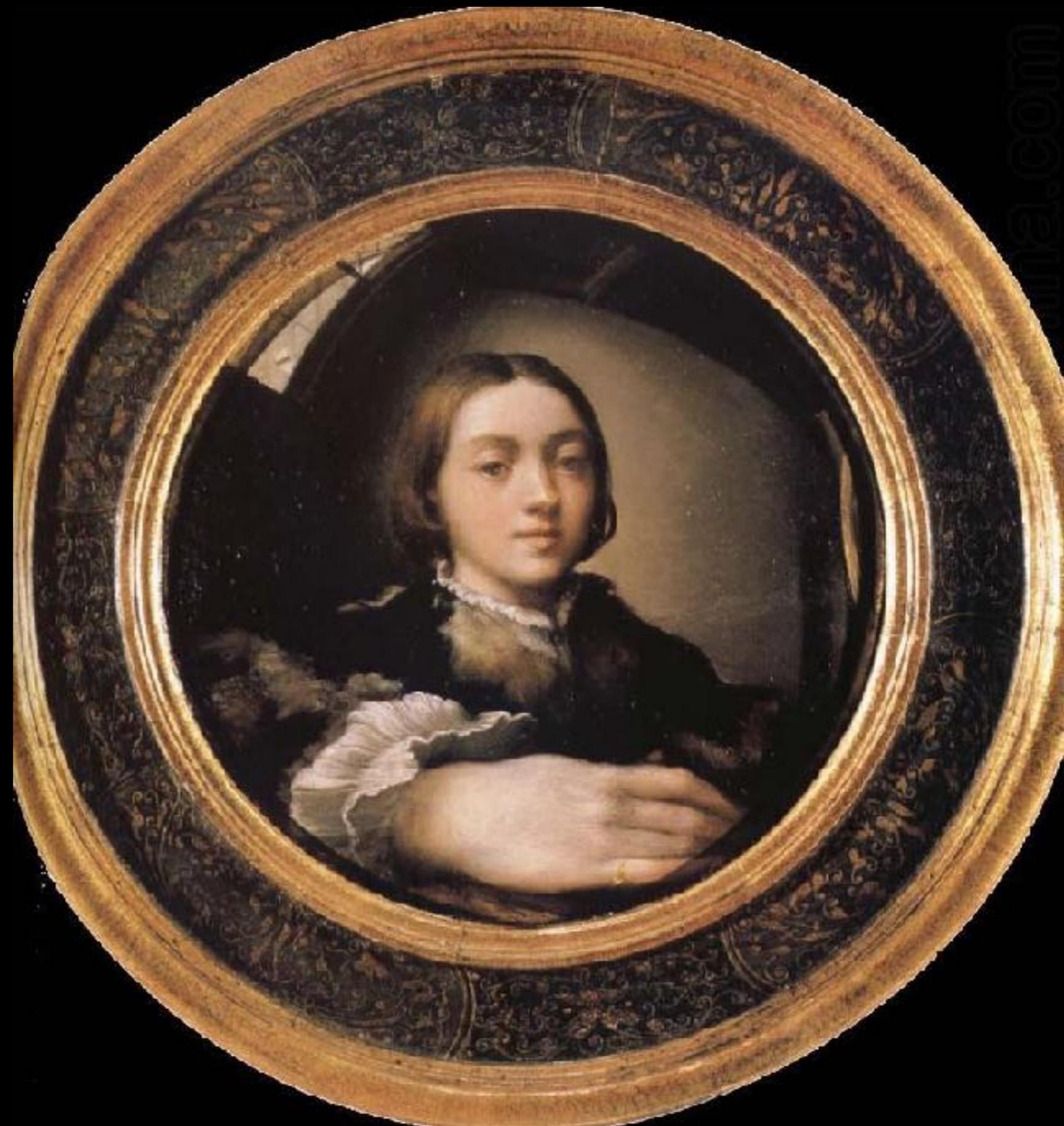
<div class="cities">
<h2>Tokyo</h2>
<p>Tokyo is the capital of Japan, the center of the Greater Tokyo Area, and the most populous metropolitan area in the world.</p>
<p>It is the seat of the Japanese government and the Imperial Palace, and the home of the Japanese Imperial Family.</p>
<p>The Tokyo prefecture is part of the world's most populous metropolitan area with 38 million people and the world's largest urban economy.</p>
</div>
```

Classes

```
<div class="theAuthor">
  --- from John Duckett's <span><a
  href="https://www.amazon.com/Web-Design-HTML-JavaScript-jQuery/dp/1118907442
  /ref=sr_1_3?ie=UTF8&qid=1526310943&sr=8-3&keywords=html+and+css"
  target="_blank">HTML + CSS</span></a>
  <br>
</div>
```

To select these classes in CSS you would do so with **.theAuthor** syntax

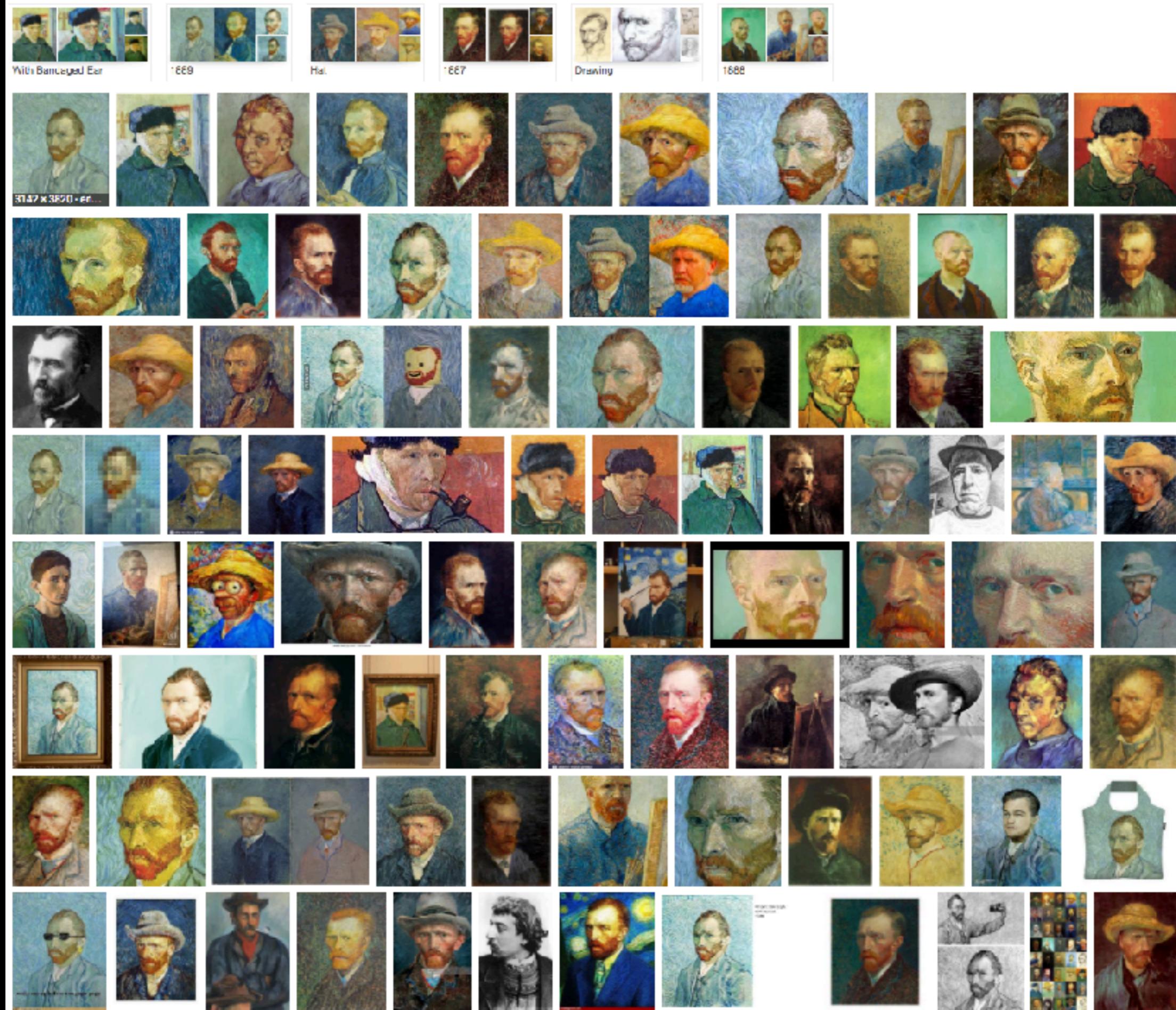
```
.theAuthor{
  background: rgb(255,255,255);
  /* HSL: Hue, Saturation + Lightness
  Hue – as an angle between 0 + 360
  Saturation – as a percentage
  Lightness – as a percentage: 0% = white, %50 = normal + 100% is black
  Alpha – expressed btw 0 _ 1.0 : 0.5 = 50% transparency, .75 is 75%
  transparency*/
  background: hsl(0,100%,100%, 0.2);
  text-align: center;
}
```



Parmigianino

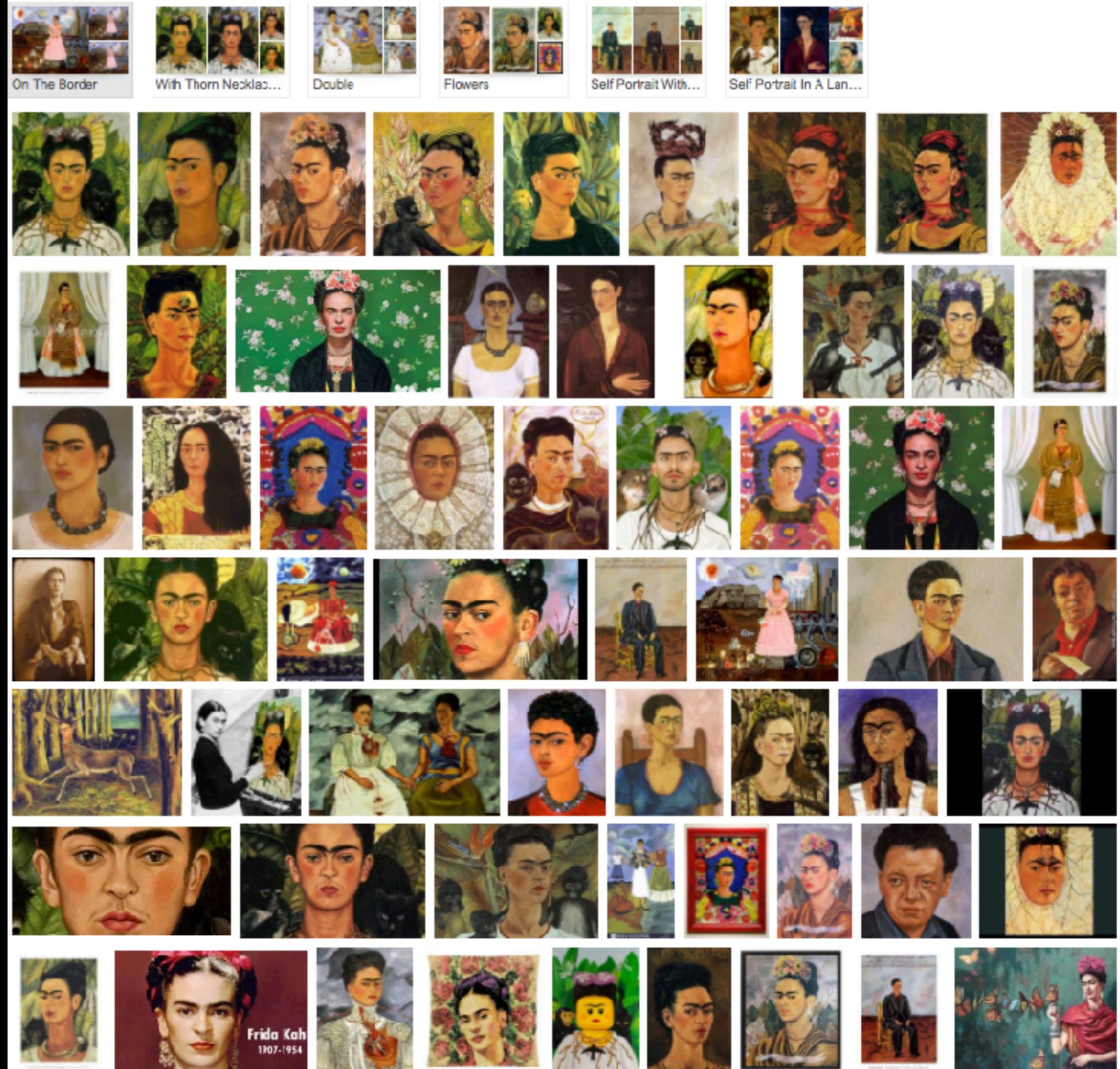
Self-Portrait in a Convex Mirror c. 1524

Van Gogh, 1853 - 1890



Frida Kahlo, 1907 - 1954

All Images Shopping Videos News More Settings Tools





Robert Cornelius, 1839



Yves Klein
Leap into the Void, 1960



Buzz Aldrin
space selfie, 1966



Ana Medieta
Imagen de Yagul, 1973



Andy Warhol
six self portraits, 1986

"In the future, everyone will be world-famous for 15 minutes." - 1968



Pipiloti Rist, born 1962

I'm Not a Girl Who Misses Much, 1986



Pipiloti Rist

Be Nice To Me, 2000

TWENTY QUESTIONS
(A SAMPLER)



IS SHE AS
PRETTY AS A
PICTURE

OR
CLEAR AS
CRYSTAL

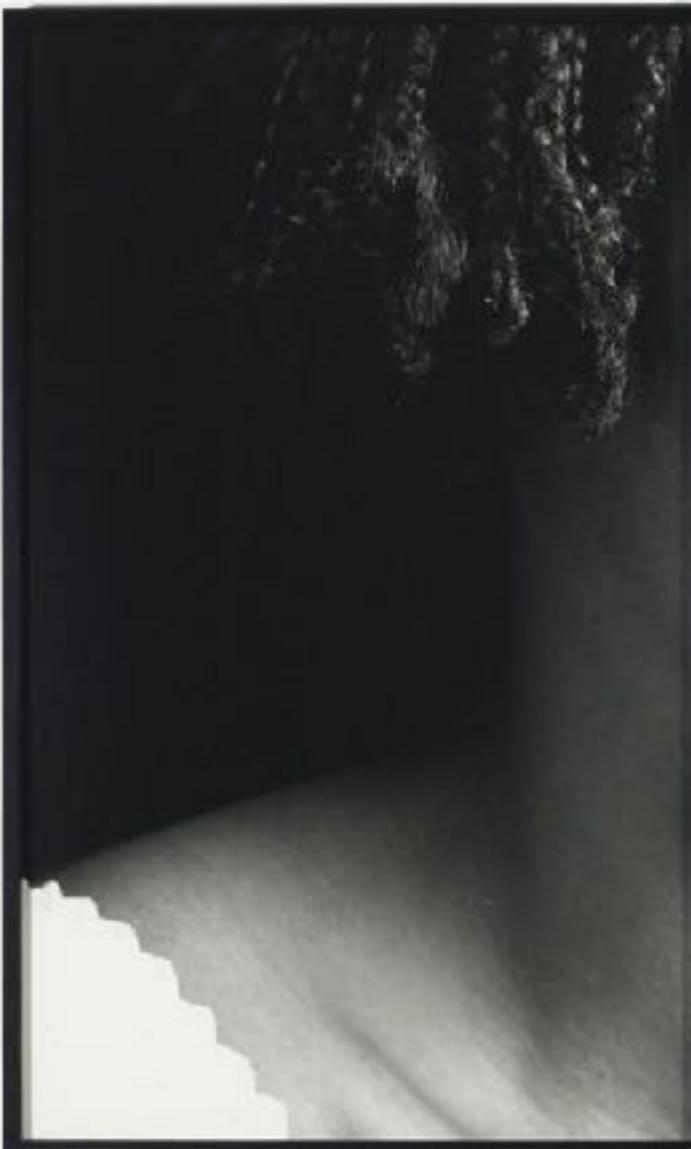


OR
PURE
AS A LILY



OR
BLACK
AS COAL

OR
SHARP
AS A RAZOR



necktie
neck & neck
neck-ed
neckless

necking
neckline
necklace
breakneck



Jennifer Ringley
Jennicam, 1996 - 2003

Mark Zuckerberg

Follow Message ...

Timeline About Friends Photos More ▾

Follow Mark to get his public posts in your News Feed.

97,299,559 Followers

Follow

About

Follow Mark to get his public posts in your News Feed.

97,299,577 Followers

Follow

Overview

Work and Education

Places He's Lived

Contact and Basic Info

Family and Relationships

Details About Mark

Life Events

Works at Facebook and Chan Zuckerberg Initiative

zuck (Instagram)

May 14, 1984

Lives in Palo Alto, California
From Dobbs Ferry, New York

Married to Priscilla Chan
Married since May 19, 2012

Facebook, February 2004
<http://www.fb.com>

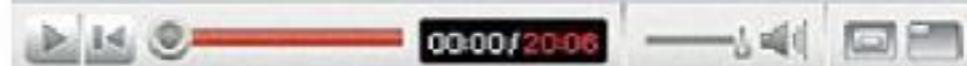
DECEMBER 25, 2006 / JANUARY 1, 2007

www.Time.com

TIME

PERSON OF THE YEAR

You.



Yes, you.

You control the Information Age.
Welcome to your world.

Youtube, February 2005

You: Person of the Year
Time Magazine, December 2006