

# In this course

---

- Making a web page with HTML and CSS
- Writing HTML text tags and CSS selectors
- Laying out a web page with multiple sections
- Working with images on a web page
- Customizing fonts and building HTML forms



# **LEVEL 1**

# **HTML**



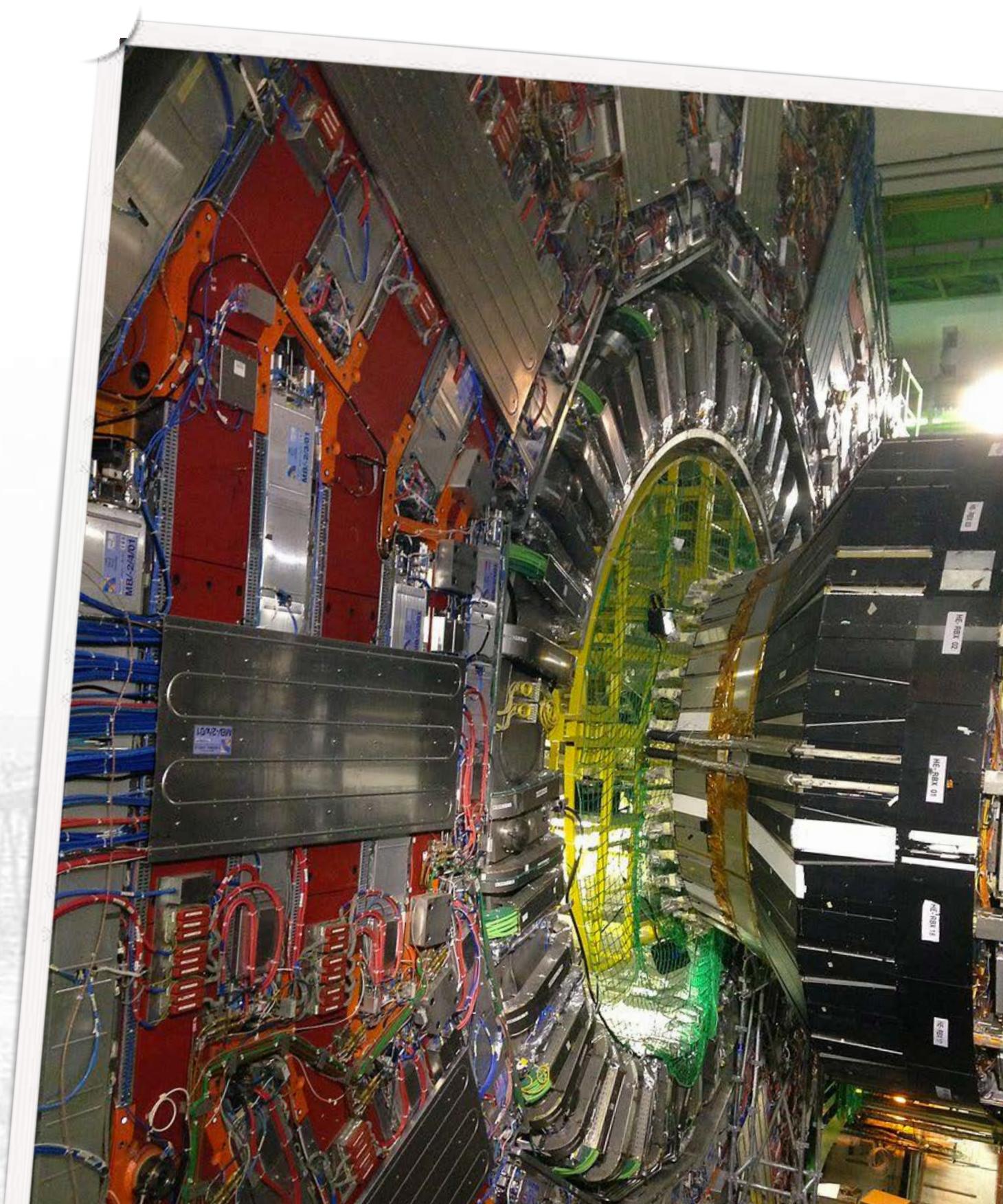
# A Quick Story of the Web



**Tim Berners-Lee**

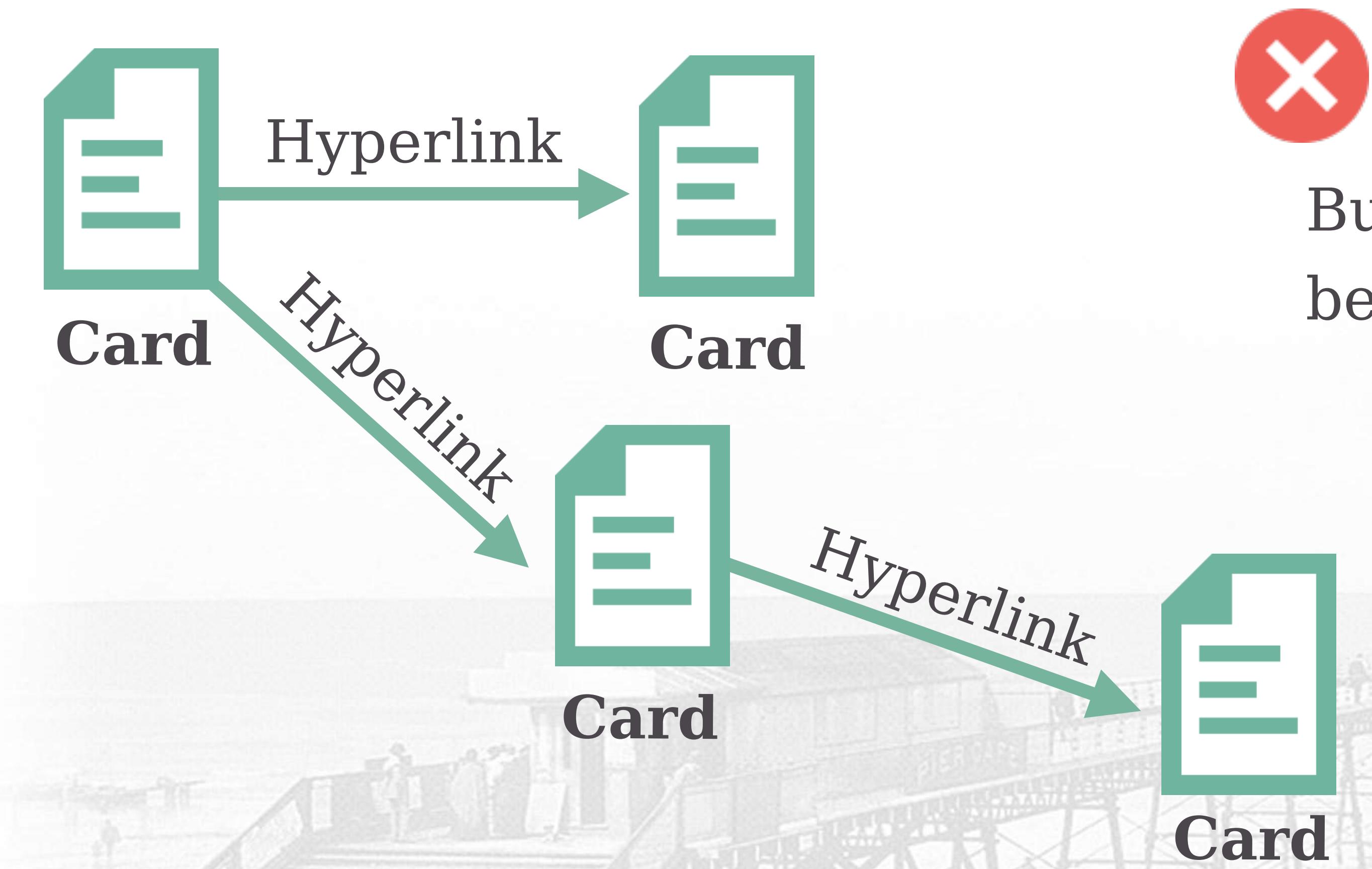
CERN  
You know, the same people who run  
the Large Hadron particle  
accelerator.

In 1980, Tim worked for CERN, where he proposed a better way for researchers there to share and read documents.



# ENQUIRE

So in 1984, he built something called ENQUIRE for CERN, which was made up of Cards (Documents) and Hyperlinks (which connected the Documents).



But Tim realized he could do better.



# The World Wide Web

---

With Robert Cailliau, Tim Berners-Lee wrote a proposal in 1990 for the creation of the **World Wide Web**.

When it started, the WWW was a way for scientists around the world to create and share their own webpages.

A graphic representation of the first logo for the World Wide Web. It consists of two stylized, overlapping 'W' shapes. The top 'W' is dark green, and the bottom 'W' is light green, both outlined in black. They overlap to form a shape reminiscent of a stylized 'A' or a series of connected 'V's.

The World Wide Web's  
First Logo



# The language of a webpage is HTML

HTML allows you to take a plain text document created in any simple text editor...

Learn By Doing

No setup. No hassle. Just Learning.

Code School teaches web technologies in the comfort of your browser with video lessons, coding challenges, and screencasts.

.. and organize it into lists, link to other webpages, include images, and more.



## Learn By Doing

**No setup. No hassle. Just learning.**



Code School teaches web technologies in the comfort of your browser with video lessons, coding challenges, and screencasts.

[View Our Courses](#)

Learn:

- [Git](#)
- [Backbone.js](#)
- [Sass](#)
- [Rails](#)
- [jQuery](#)



# But what does HTML mean?

---

HTML stands for HyperText Markup Language

## Hypertext

Overcoming the constraints of written text. *Interactive.*

## Markup Language

A way to literally "Mark Up" a document to specify attributes, like different font sizes, lists, links to other webpages, and images.



# HTML is written in text files

---

Just like most programming languages, we type a bunch of HTML into a file (aka. document) so we can send it around.



index.html



Name of the file

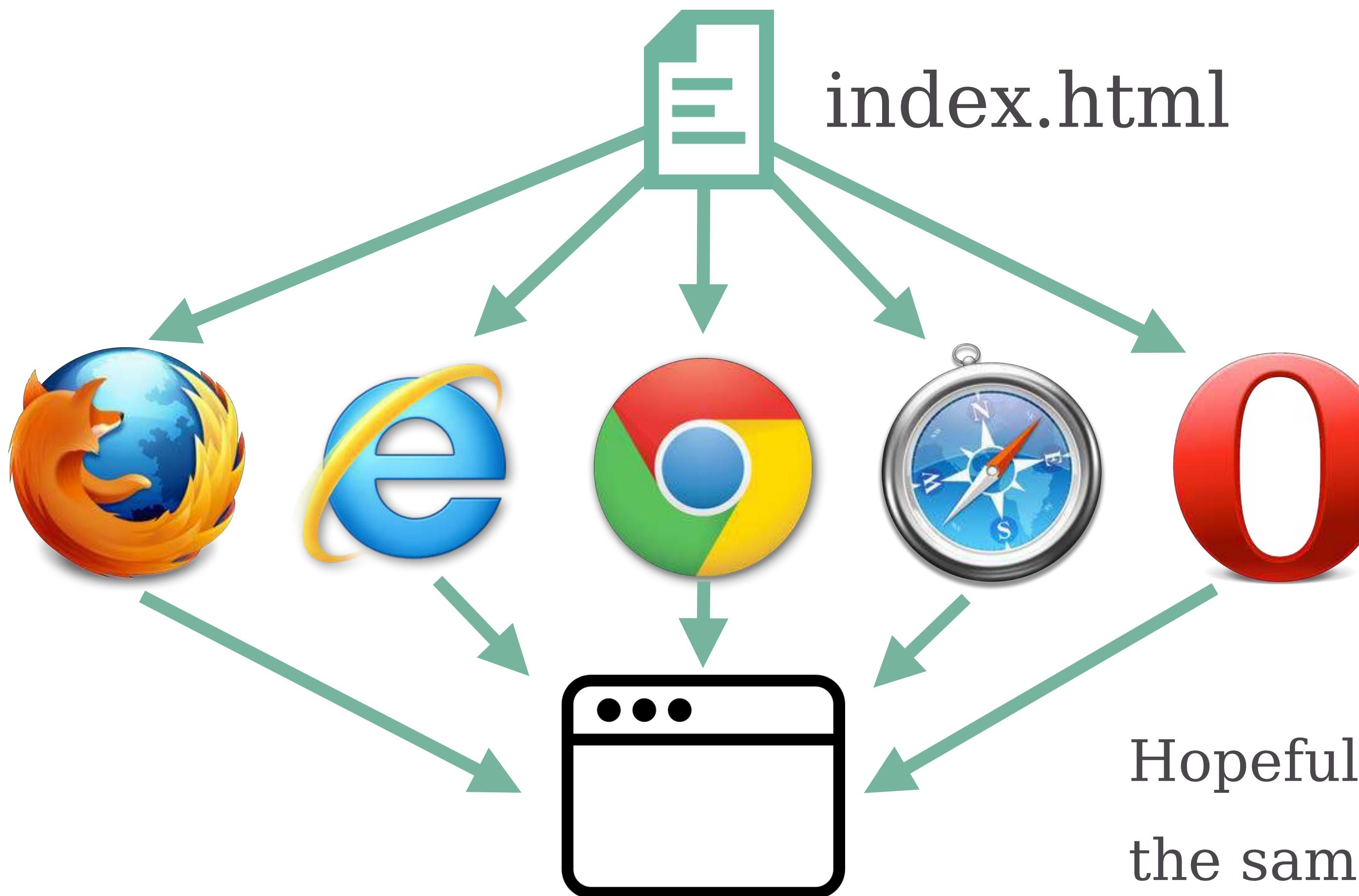
File extension  
(tells the browser the file contains HTML)



# To display an HTML file we need a browser

**Web Browsers** are basically "HTML Readers."

They understand how to read HTML and display it for you.



Hopefully they all display  
the same thing (that's the idea).



# A guide to the slides

---

In this course, we're going to show you HTML, CSS  
and what those look like in the browser.

Look for these symbols to help you figure out what you're looking at:

**HTML**

HTML code

**CSS**

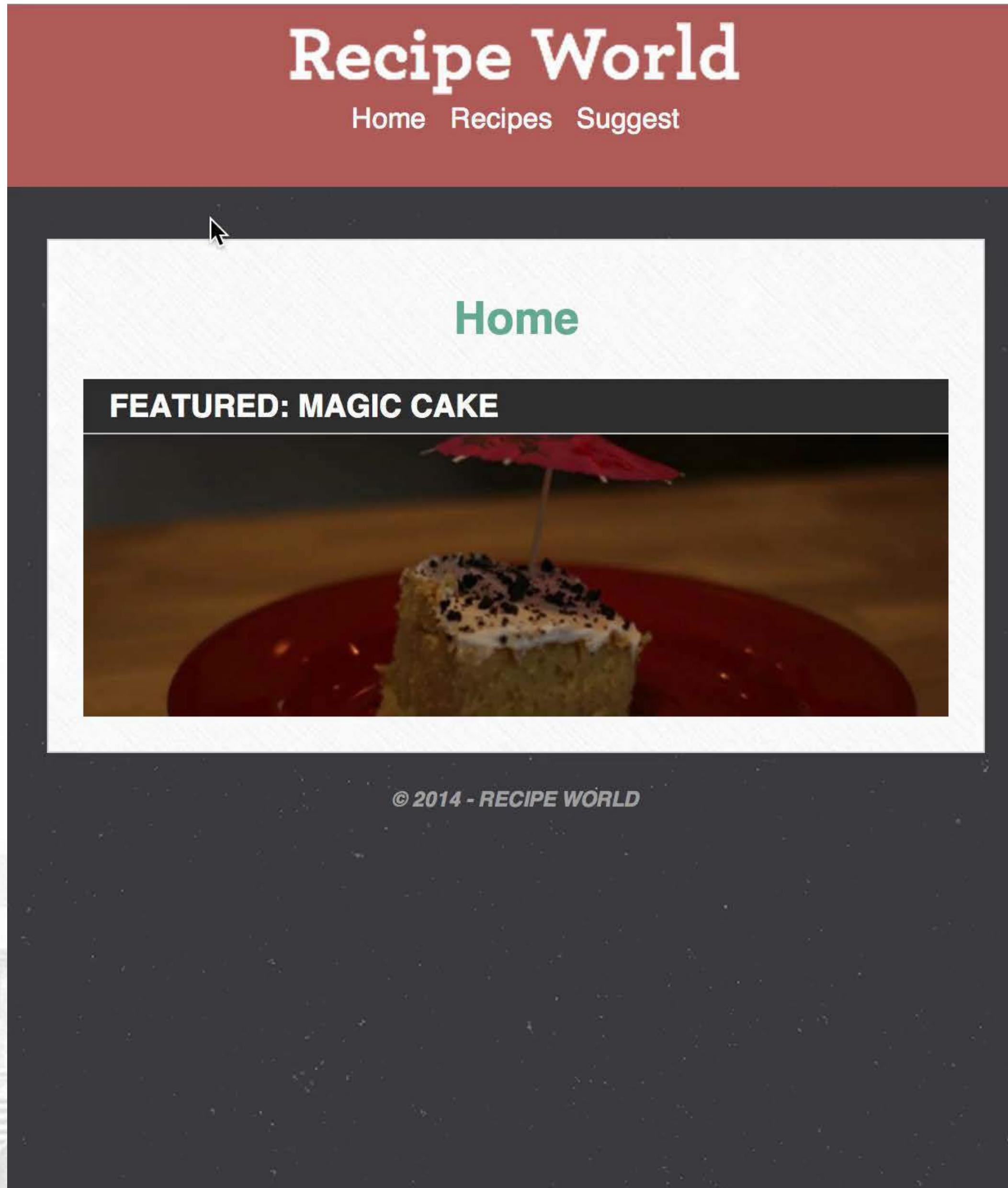
CSS code



What it looks like displayed in the browser



# Building Recipe World



# First things first: get some content!!

---

Before you start making a web page, you've got to figure out what content it is going to show.

We'll start with a single recipe with these sections.

Page Title:

Recipe World

Recipe Name:

Magic Cake

Section Name:

Ingredients

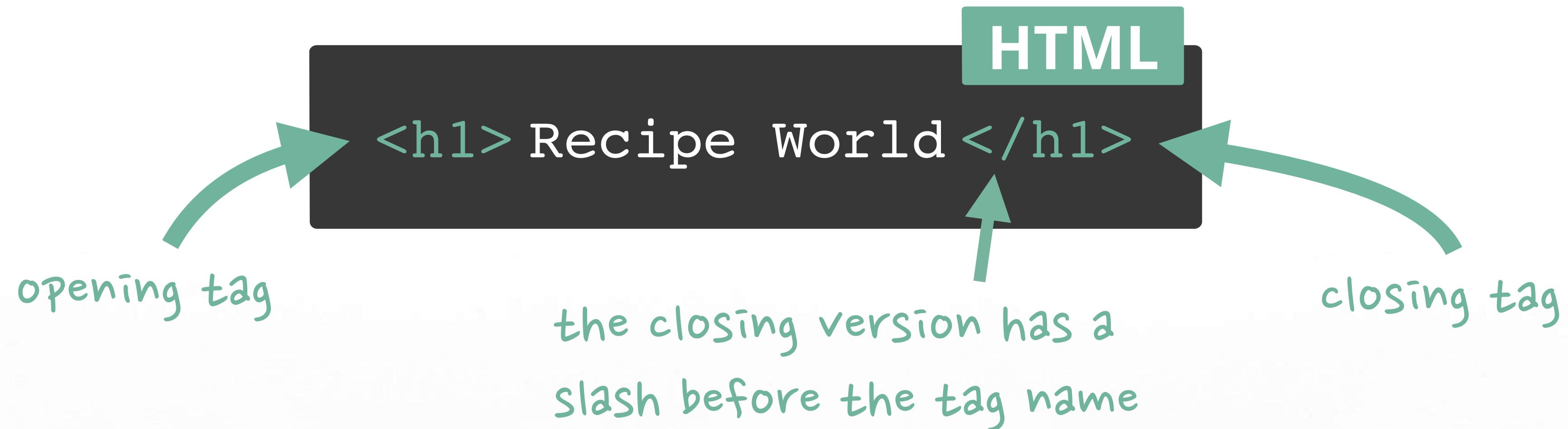
Section Name:

Directions



# Writing HTML

Most of the time, you'll put your content in between **HTML tags**, which have corresponding opening and closing versions.



**Recipe World**



# Use heading tags to define your content hierarchy

Higher heading numbers mean the content that appears between the headings is less important than lower numbers.

HTML

```
<h1>Recipe World</h1> ← More important  
<h2>Magic Cake</h2>  
<h3>Ingredients</h3>  
<h3>Directions</h3> ← Less important
```



Recipe World

Magic Cake

Ingredients

Directions

Generally speaking:

The **page title/company name** goes in the **<h1>**

The **page main subject** goes in the **<h2>**

**<h3>** through **<h6>** are used to organize other divisions of page content



# Use paragraph tags for non-heading text

< p > is called the paragraph tag.

HTML

```
<h1>Recipe World</h1>
<h2>Magic Cake</h2>
<p>Magic Cake is one of the tastiest...</p>
<h3>Ingredients</h3>
...
...
```



This paragraph  
describes Magic  
cake

Add paragraph content (where necessary) in between heading tags.

## Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.



# Use an unordered list to display a list of things

<ul> stands for unordered list.

Each list item needs to also be put inside of an <li> tag.

unordered list

```
<h3>Ingredients</h3>
<ul>
  <li>2 eggs</li>
  <li>1 pound of sugar</li>
  <li>3 sticks of butter</li>
</ul>
```

list items

HTML

This list of ingredients doesn't need to be in a specific order

## Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter



# Nesting tags

HTML tags don't have to always just contain text, they can contain other HTML tags.

HTML

```
<p>Magic Cake is one of...
<h3>Ingredients</h3>
<ul>
  <li>2 eggs</li>
  <li>1 pound of sugar</li>
  <li>3 sticks of butter</li>
</ul>
<h3>Directions</h3>
...
when you indent, it's easy to see that 3
<li> tags are nested inside of a <ul> tag
```

parent →

children

A tag that contains other tags is called the **parent**.

The tags contained in a parent tag are called **children**.



# Not indenting child tags makes HTML hard to read

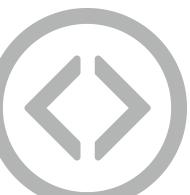


HTML

```
<h1>Recipe World</h1>
<h2>Magic Cake</h2>
<p>Magic Cake is one of...
<h3>Ingredients</h3>
<ul><li>2 eggs</li><li>1 pound of
sugar</li><li>3 sticks of butter</li>
</ul>
<h3>Directions</h3>
...

```

This is valid  
HTML, but it's  
really hard to  
read



# Use an ordered list to show list items in a certain order

If the content in your list does refer to steps to be followed, use an ordered list.

HTML

ordered list

list items

```
<h3>Ingredients</h3>
<ul>...
<h3>Directions</h3>
<ol>
    <li>Mix eggs, sugar, and butter in
        a large bowl.</li>
    <li>Spread into a non-stick dish</li>
    <li>Bake at 350
        <li>Let sit at room
            temperature for 20
            minutes</li>
    <li>Eat and enjoy!</li>
</ol>
```

these steps need to be  
followed in order or  
the cake will be ruined

## Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!



# Wrapping everything in the body

Any content that appears on a web page should be in between a <body> tag.

the parent of all visible content

```
<body>
  <h1>Recipe World</h1>
  <h2>Magic Cake</h2>
  <p>Magic Cake is one of...</p>
  <h3>Ingredients</h3>
  <ul>...</ul>
  <h3>Directions</h3>
  <ol>...</ol>
</body>
```

HTML

<body> doesn't display anything, but helps keep the page content organized.

## Recipe World

### Magic Cake

Magic Cake is one of the tastiest cakes the world have tried it and lived to tell about it, and these a

#### Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

#### Directions

1. Mix eggs, sugar, and butter in a large bowl
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!



# Adding a head tag

Non-visible stuff goes in the <head> tag.

HTML

```
<head>
</head> ← we'll discuss what goes in the
<body>    <head> later when we need it
          <h1>Recipe World</h1>
          ...
</body>
```

You'll eventually use the head tag to load other useful scripts, like CSS and JavaScript.



# Make everything a child of a single parent tag

---

All of your HTML goes inside of the <html> tag.

HTML

```
<html>
  <head>
  </head>
  <body>
    <h1>Recipe World</h1>
    ...
  </body>
</html>
```



Notice that we're indenting again so it's  
easy to see the parent/child relationship



# Setting the DOCTYPE to html

---

The DOCTYPE sets the HTML version.

HTML

```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
    <h1>Recipe World</h1>
    ...
  </body>
</html>
```

The browser can make better decisions about how to display your page when it knows which version of HTML you're using.



# The recipe page so far

## Recipe World

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

#### Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

#### Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!



# Adding links to our recipe site



## Recipe World

- [Home](#)
- [Recipes](#)
- [Suggest](#)

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.



#### Ingredients

- 2 eggs
- 1 pound sugar
- 3 sticks of butter

#### Directions

1. Mix eggs, sugar, and butter in a large bowl
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!

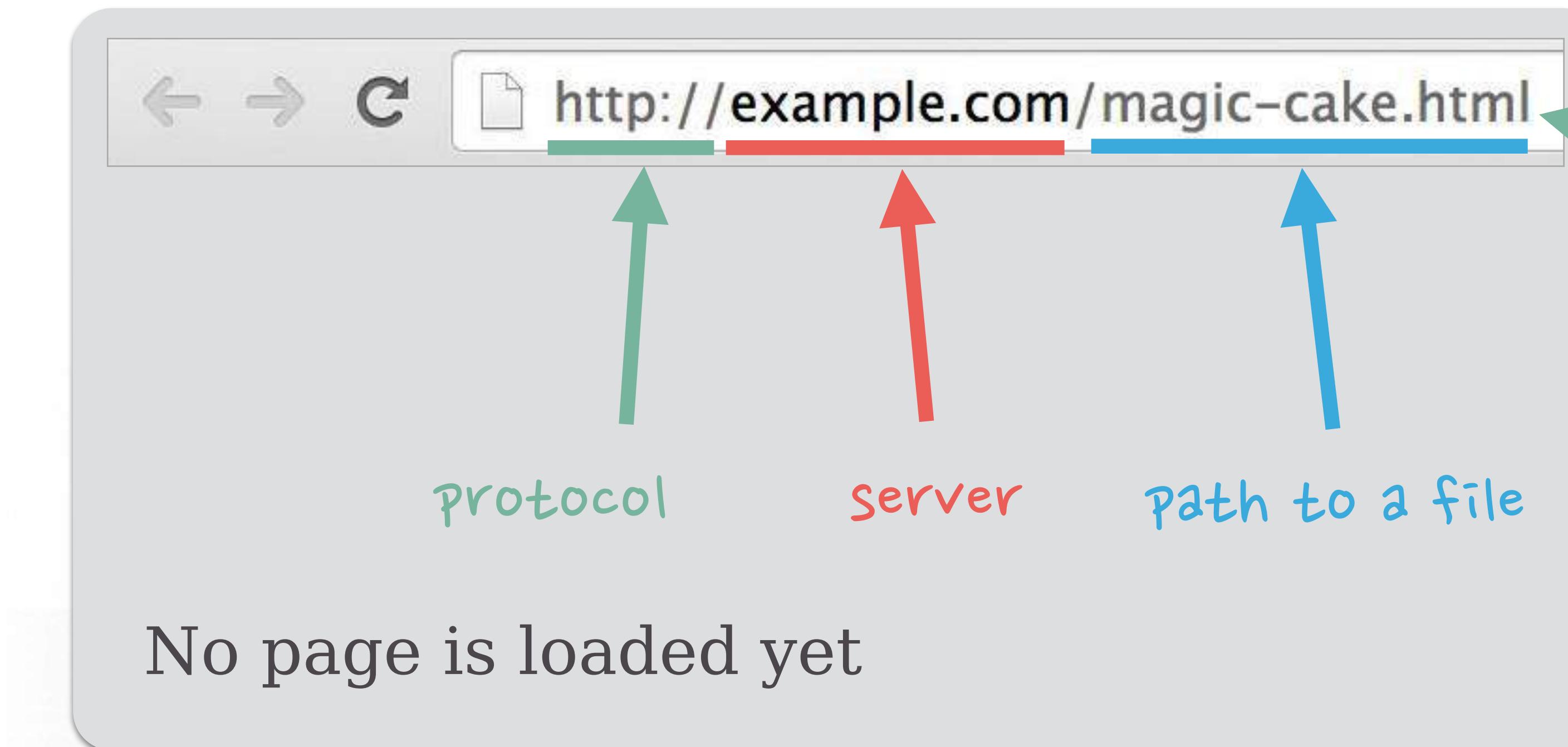


# What happens when you type a URL in the Browser

1

Enter a URL into the browser's address bar.

Browser



The URL contains a **protocol**, **server** and a **file** that is being requested.

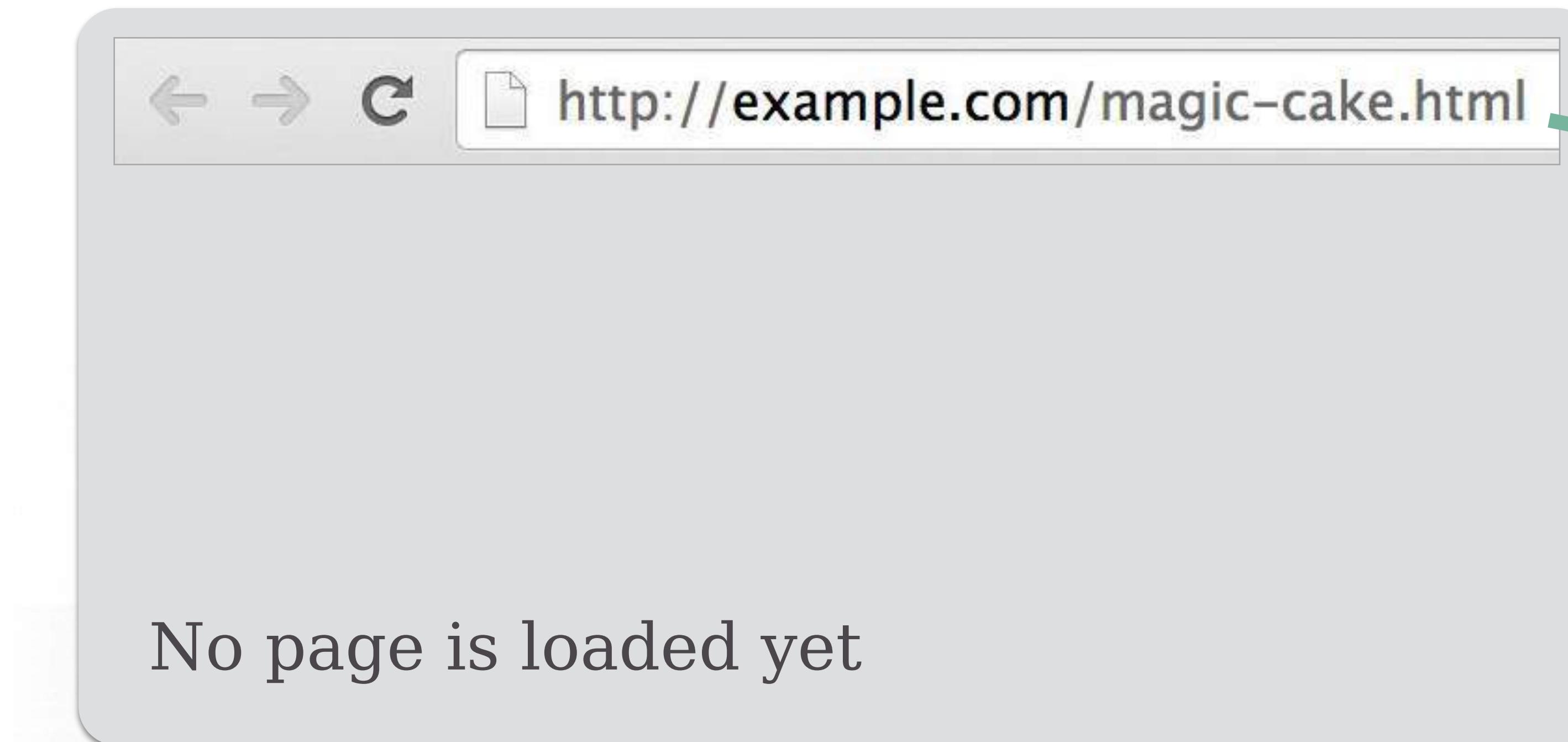


# What happens when you type a URL in the Browser

2

A request is sent to the server, and the server looks for the file that's being requested.

Browser



Server

example.com



# What happens when you type a URL in the Browser

- The server returns the file to the browser and the browser displays it.



# What happens when you click a link in the Browser

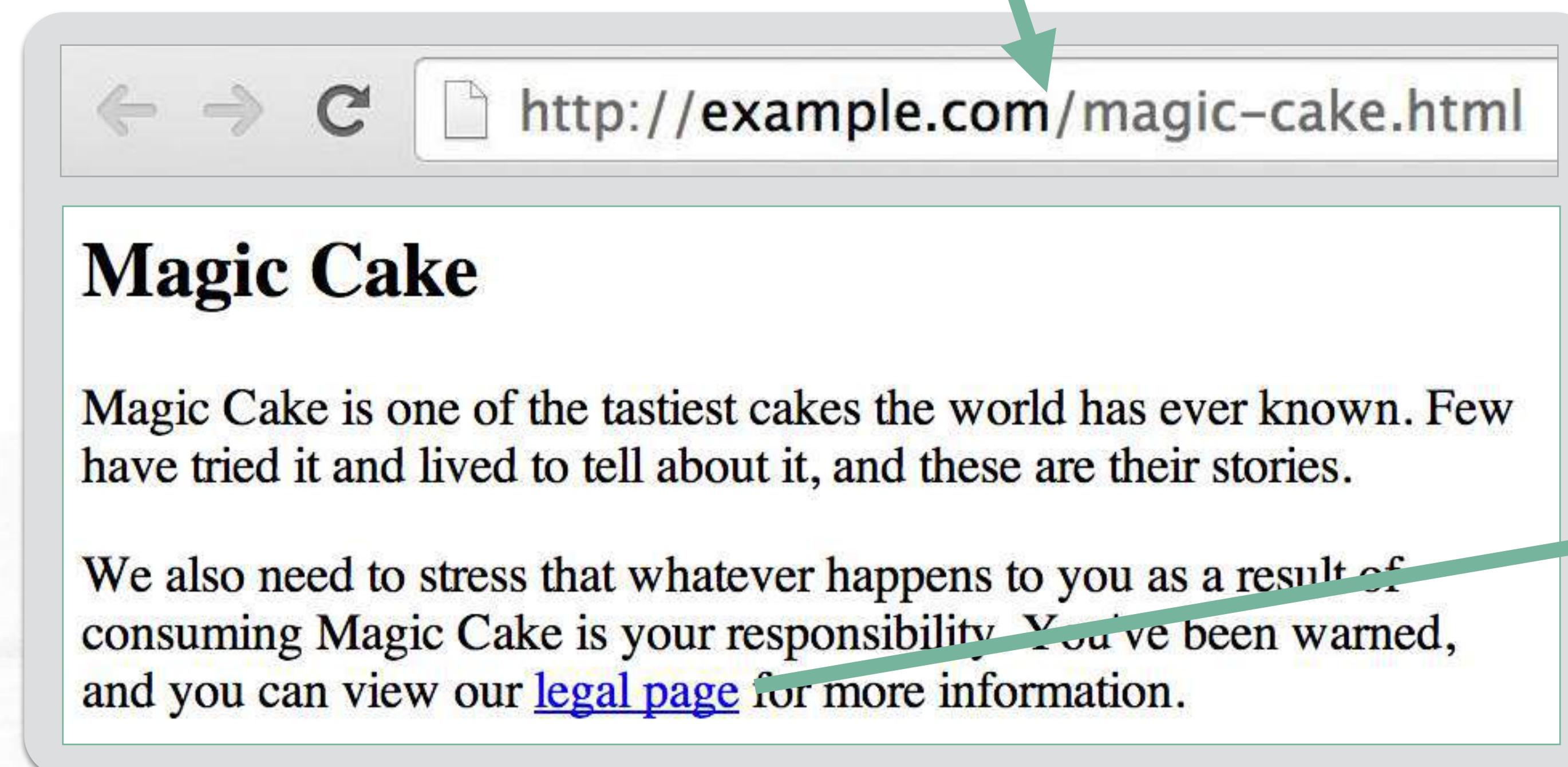
Clicking a link is like typing a URL in the address bar, only the URL is already pre-entered in HTML as part of link.

1

Click the link on the page, and a request for the file in that link is sent to the server.

Browser

current page containing the link



Server  
example.com



File

magic-cake.html



File

legal.html

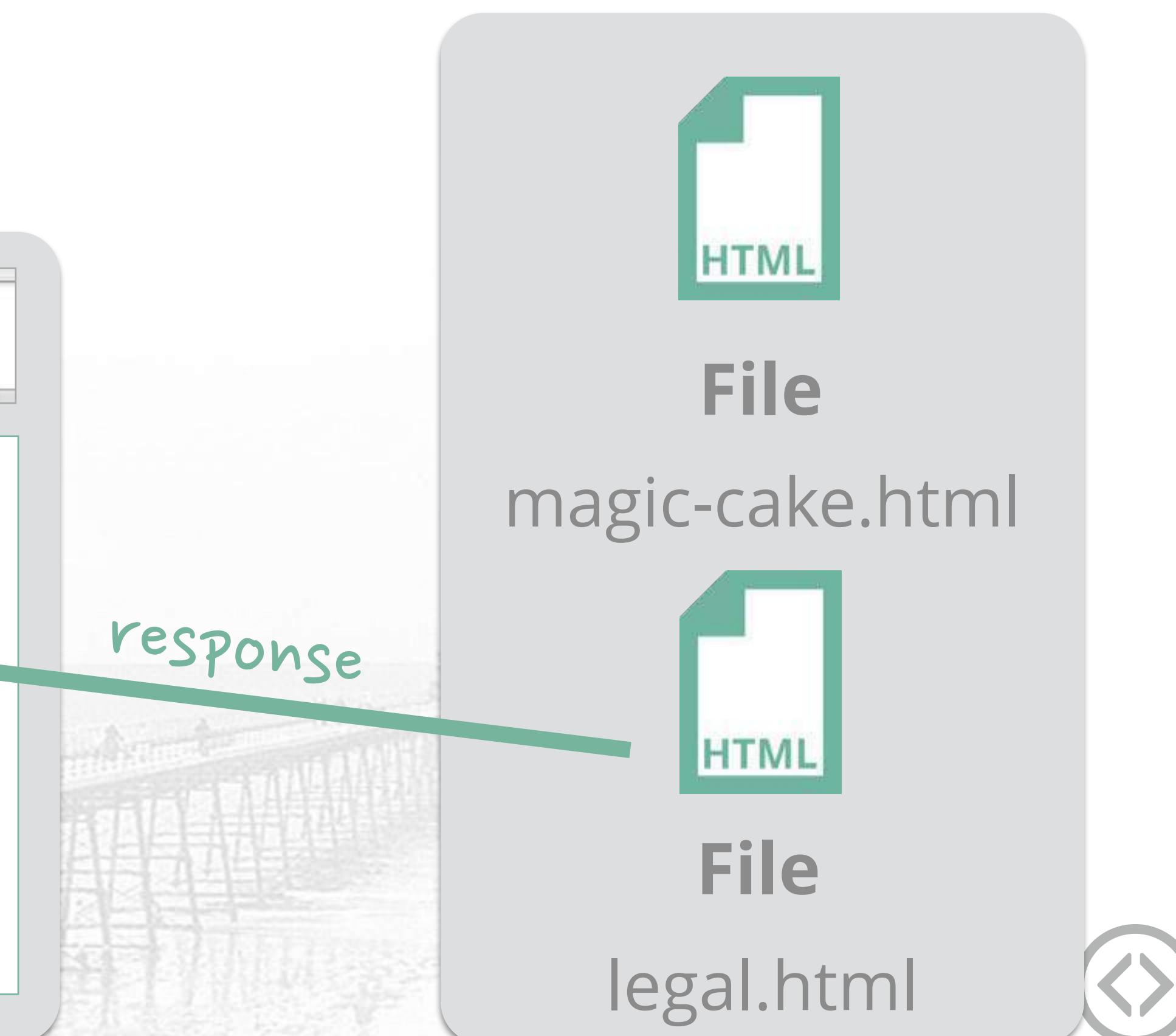
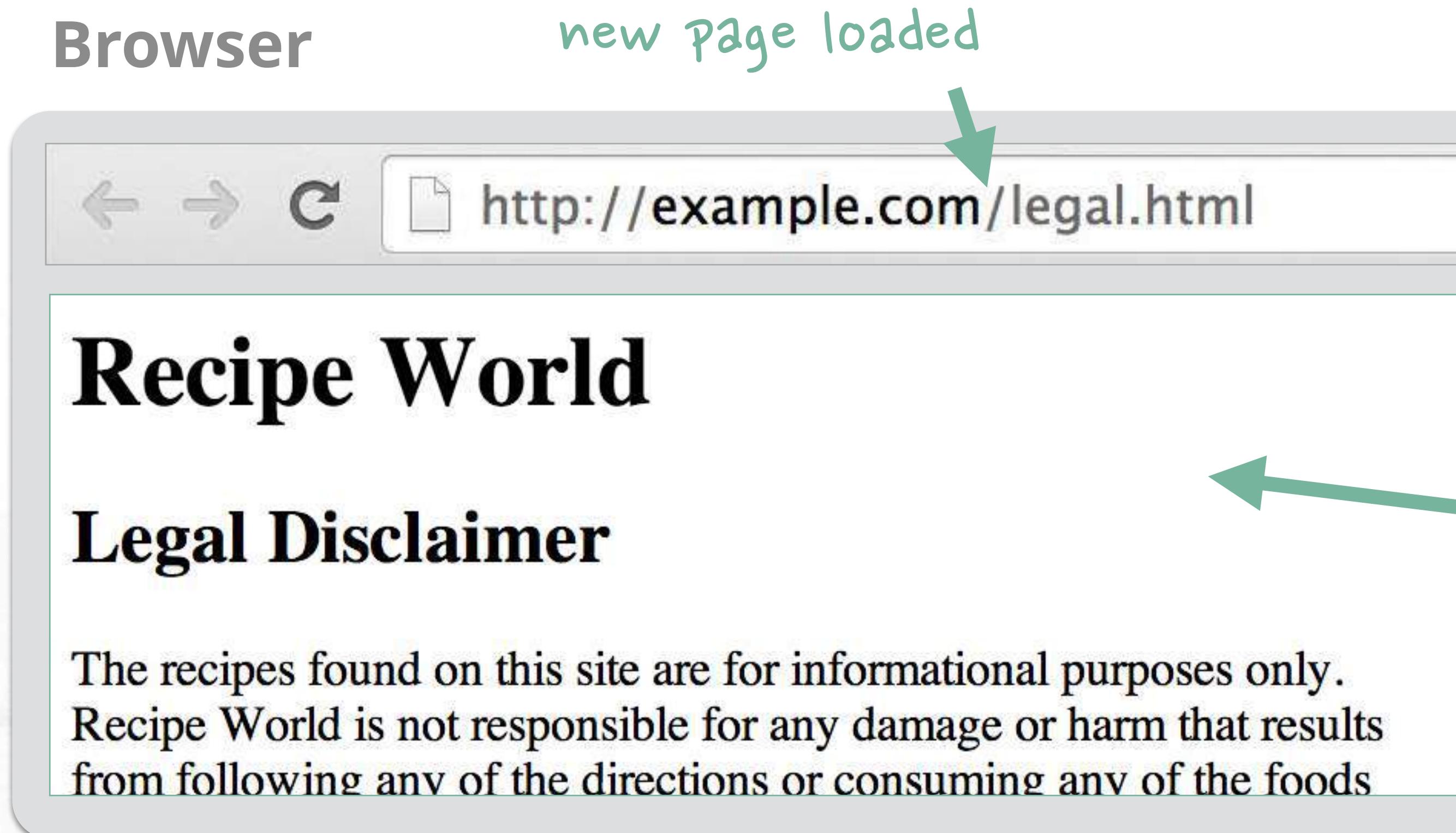


# What happens when you click a link in the Browser

2

The server returns the file to the browser and the browser replaces the current page with the new file.

Server  
example.com



# How to make a link

Anything that appears between the opening and closing **a** tags is the part that will be a clickable link on the web page.

HTML

```
<p>...can view our <a>legal page</a> for more...</p>
```

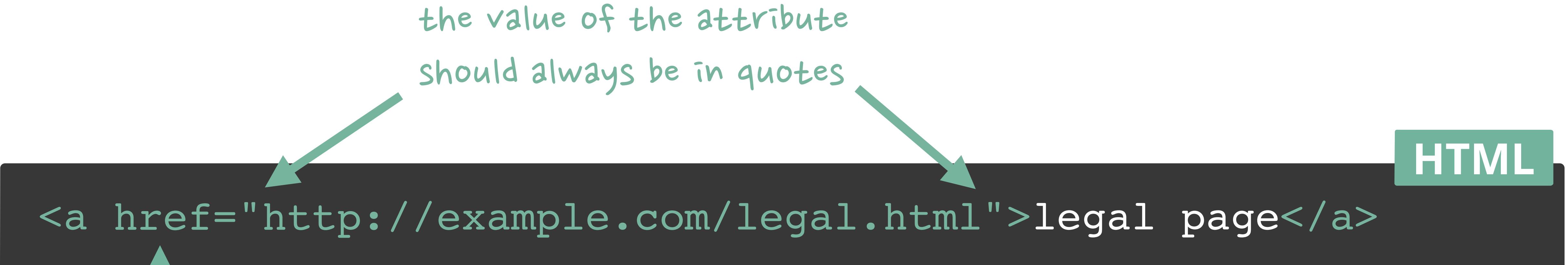
link text

**a** is short for **anchor**.



# How to make a link

The **href** attribute sets what page should open when the link is clicked.



HTML **attributes** let you add additional features or information to a tag beyond just the tag name and text content.



# How to make a link

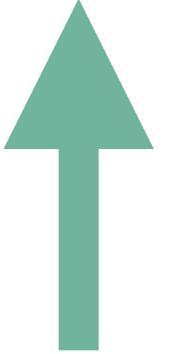
The **target** attribute sets where the page you set in the href should open when the link is clicked.

HTML

```
<a href="http://..." target="_blank"
```

HTML attribute

if you don't add the target attribute, the default is \_self

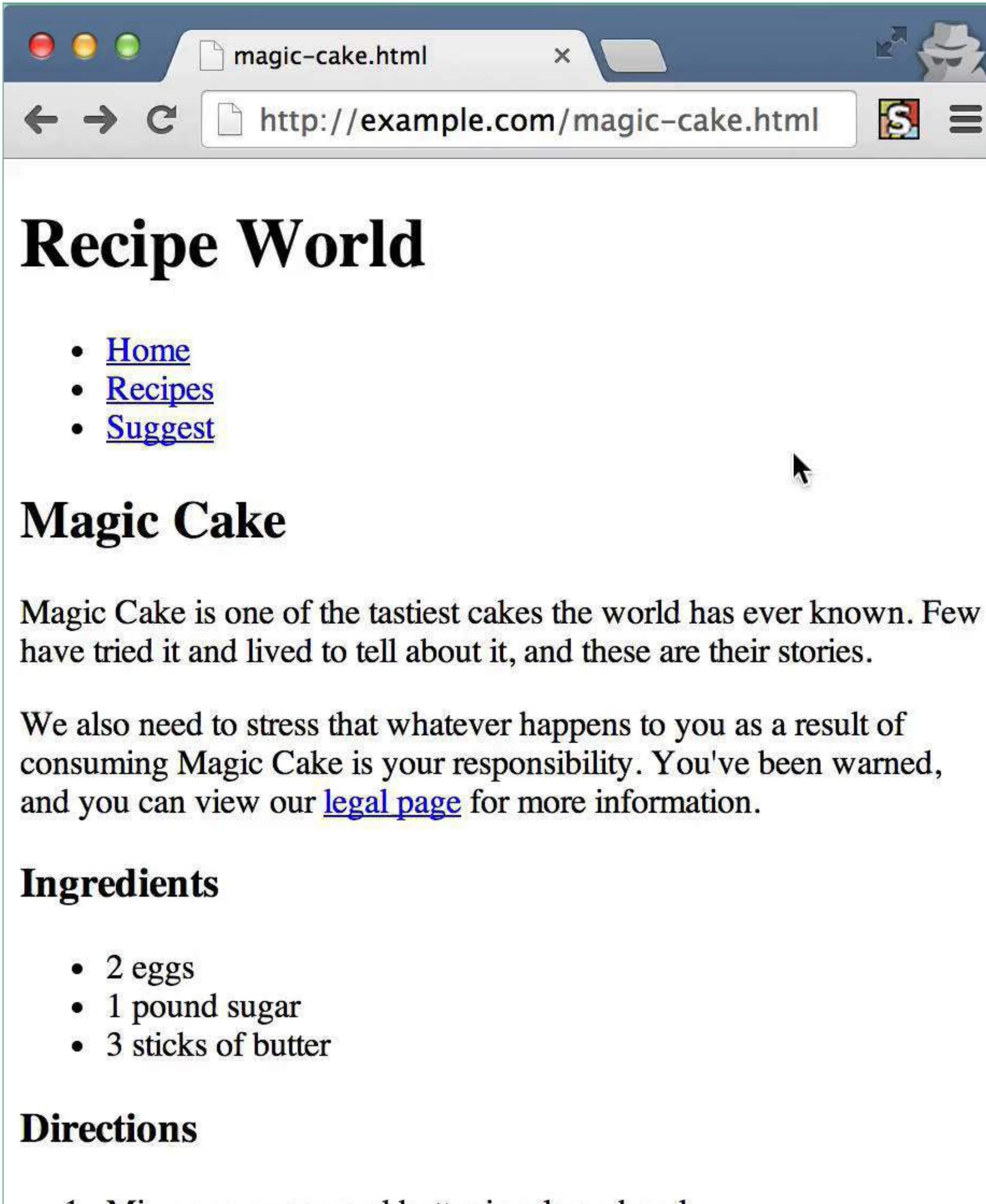


\_blank means *open this page in a new tab/window.*

\_self means *replace the currently open page in this tab/window.*



# Demo of a link opening in a new tab



A screenshot of a web browser window titled "magic-cake.html". The address bar shows "http://example.com/magic-cake.html". The page content is a recipe for "Magic Cake". It features a navigation menu with links to "Home", "Recipes", and "Suggest". Below the menu, the title "Magic Cake" is displayed. A paragraph describes the cake as one of the tastiest in the world. A warning message follows, stating that consuming the cake is the user's responsibility and linking to a "legal page". The "Directions" section is partially visible at the bottom.

• [Home](#)  
• [Recipes](#)  
• [Suggest](#)

## Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

### Ingredients

- 2 eggs
- 1 pound sugar
- 3 sticks of butter

### Directions



# Absolute vs. Relative paths

Absolute paths ask for a file from a specific server

HTML

```
<a href="http://example.com/legal.html">legal page</a>
```



*absolute always includes the protocol and server*

Relative paths ask for a file with no server specified.

HTML

```
<a href="legal.html">legal page</a>
```



*relative never includes the protocol and server*

Since no server name is included, the browser assumes you mean the same server that your page is on.



# Links using relative paths

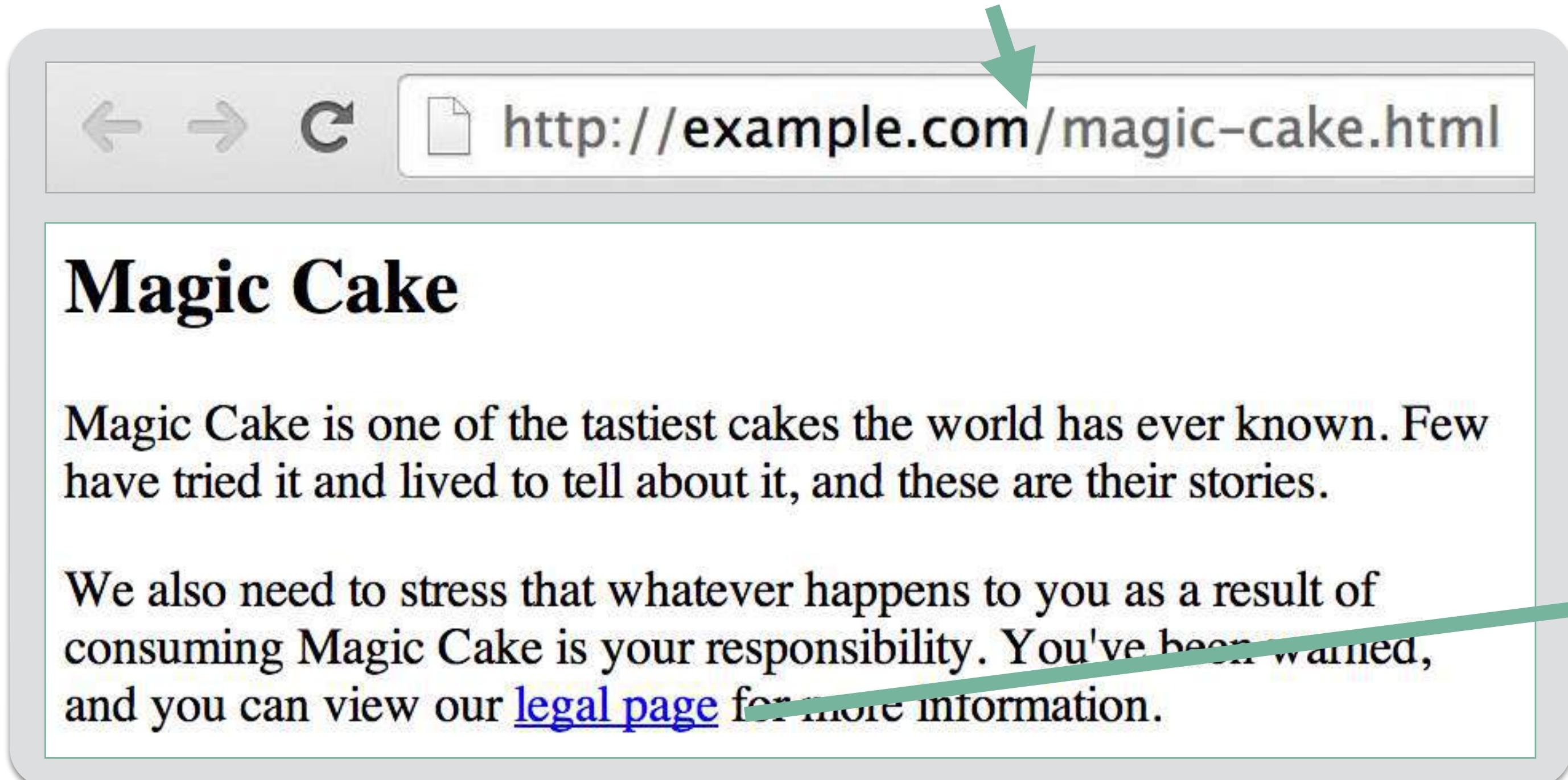
If no protocol/server is included, the browser assumes the protocol and server are the same as the page you're requesting from.

```
<a href="legal.html">legal page</a>
```

HTML

Browser

current page containing the link



Server  
example.com



File

magic-cake.html

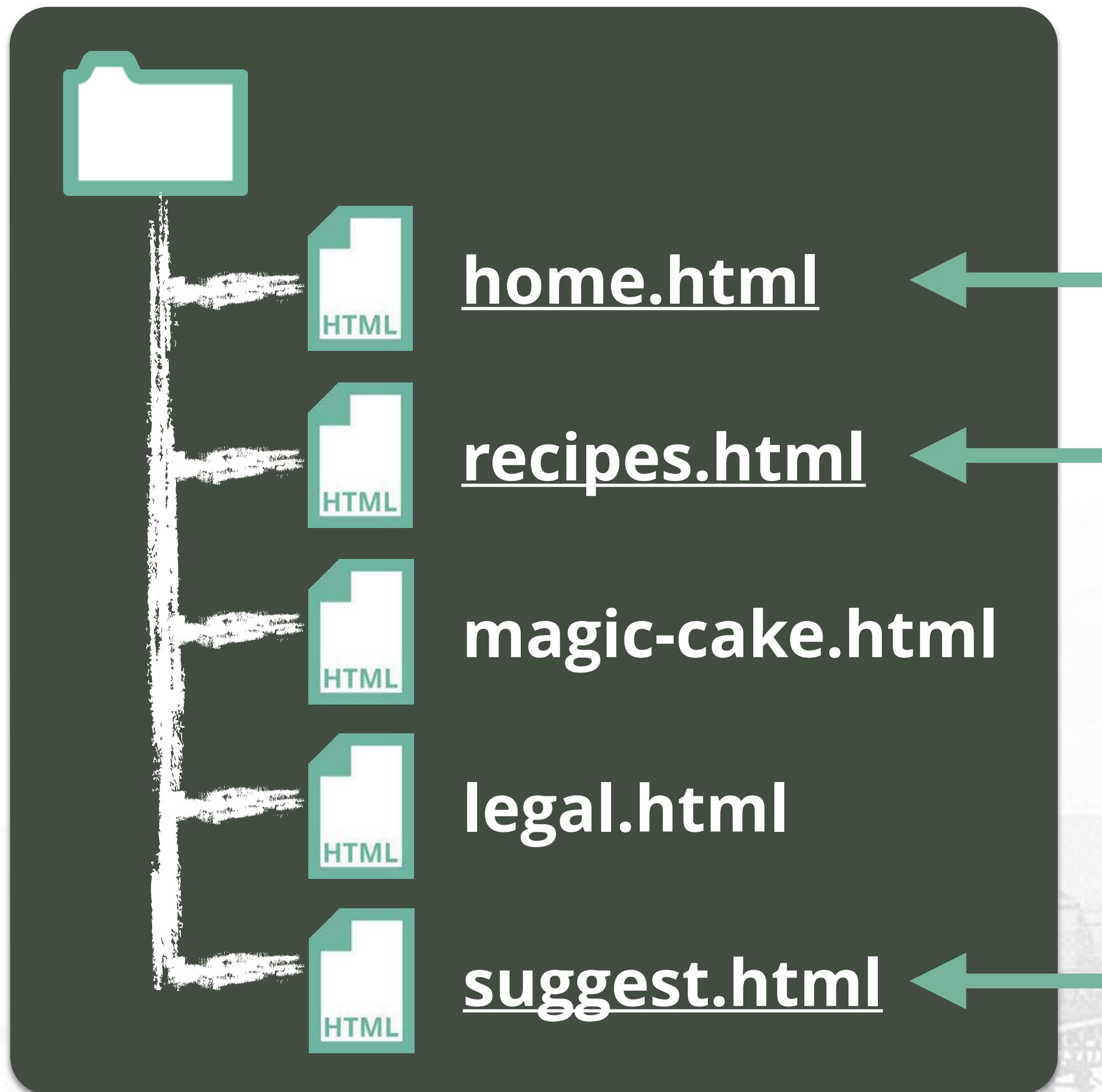


File

legal.html

# Figuring out the site navigation

## Recipe World server files



All of our HTML files are currently in our main site folder on the server.

These are the three main pages that should go in our site navigation.



# Adding site navigation with relative links

Relative links should almost always be used when linking to pages that are located on the same site.

add this block of HTML to all pages

HTML

```
<body>
  <h1>Recipe World</h1>
  <ul>
    <li><a href="home.html">Home</a></li>
    <li><a href="recipes.html">Recipes</a></li>
    <li><a href="suggest.html">Suggest</a></li>
  </ul>
  <h2>Magic Cake</h2>
  ...
</body>
```



# The Magic Cake recipe page with site navigation



## Recipe World

- [Home](#)
- [Recipes](#)
- [Suggest](#)

## Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.



### Ingredients

- 2 eggs
- 1 pound sugar
- 3 sticks of butter

### Directions

1. Mix eggs, sugar, and butter in a large bowl
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!



# Creating a link to another web site

First, find and copy the URL to the other site

The screenshot shows a web browser window with a blue header bar. The title bar says "magic-cake.html". The address bar contains the URL "http://example.com/magic-cake.html". The main content area of the browser displays a page titled "Recipe World". Below the title is a navigation menu with three items: "Home", "Recipes", and "Suggest", each preceded by a bullet point and underlined text. The next section is titled "Magic Cake" in a large, bold font. A paragraph of text follows, stating: "Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories." Another paragraph below it reads: "We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information." At the bottom of the page, there is a section titled "Ingredients" with a bulleted list: "2 eggs", "1 pound [sugar](#)", and "3 sticks of butter". A small circular icon with a double-headed arrow is located in the bottom right corner of the slide.

magic-cake.html

← → C http://example.com/magic-cake.html

S

## Recipe World

- [Home](#)
- [Recipes](#)
- [Suggest](#)

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

### Ingredients

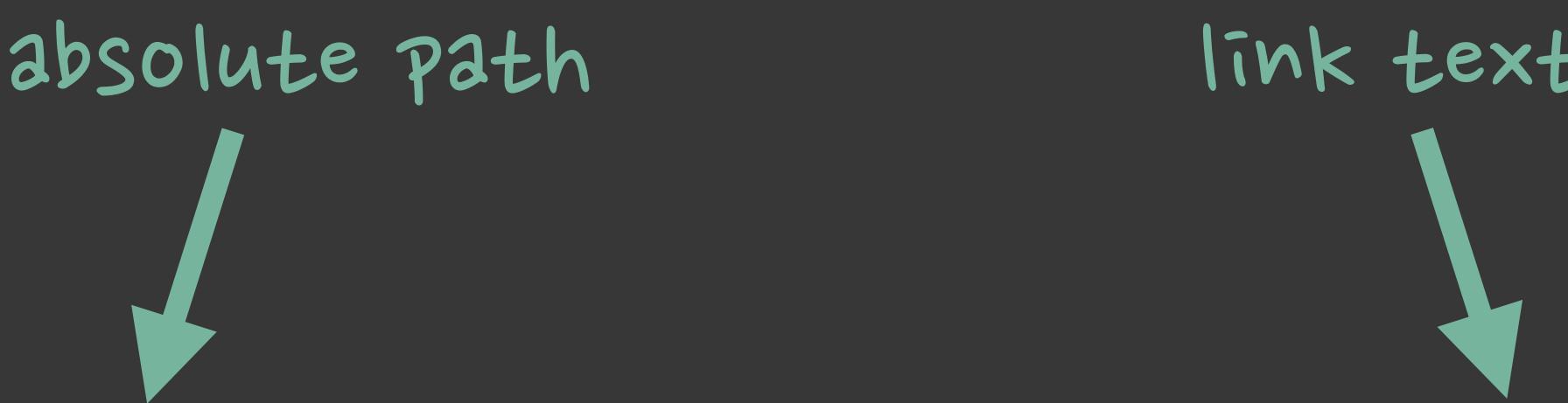
- 2 eggs
- 1 pound [sugar](#)
- 3 sticks of butter

# Creating a link to another web site

Absolute links need to always be used when linking to pages that are located on another site/server.

HTML

```
<h3>Ingredients</h3>
<ul>
  <li>2 eggs</li>
  <li>1 pound of
    <a href="http://www.amazon.com/Domino-...">sugar</a>
  </li>
  <li>3 sticks of butter</li>
</ul>
```





**LEVEL 2**

**CSS**

# The early days of style on the web

When the Web was brand new, there was no separation between HTML and the **presentation** of that HTML.



HTML X

```
<H3 COLOR=RED><CENTER>Ingredients</CENTER></H3>
```

caps-lock got stuck a lot back then too

HTML shouldn't determine how something **looks**,  
it should just determine how content is structured.



# Separating content from presentation

---

Thankfully, the authors of the web realized that, and created a way to make rules for how tags should look and put them in a separate place from the HTML.

HTML

```
<h3>Ingredients</h3>
```

CSS

*Make the **h3** tag green*

*Make the **h3** a bigger font*

*Make the **h3** have a border at the bottom*



# Cascading Style Sheets

CSS is code that changes the **appearance** of HTML.

HTML without CSS

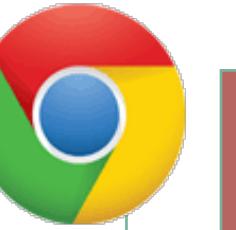
## Recipe World

- [Home](#)
- [Recipes](#)
- [Suggest](#)

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of



HTML with CSS

## Recipe World

[Home](#) [Recipes](#) [Suggest](#)

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

#### Ingredients

- 2 eggs
- 1 pound sugar



# Writing your first selector

Tags are selected by creating something called a **selector**.

HTML

```
<p>Magic Cake is one of the tastiest...</p>
```

the simplest selector is the **type selector**, and  
is just the tag name without <> brackets.

CSS

```
p {  
  text-decoration: underline;  
}
```

this is one of many properties  
that can be applied to selectors

## Magic Cake

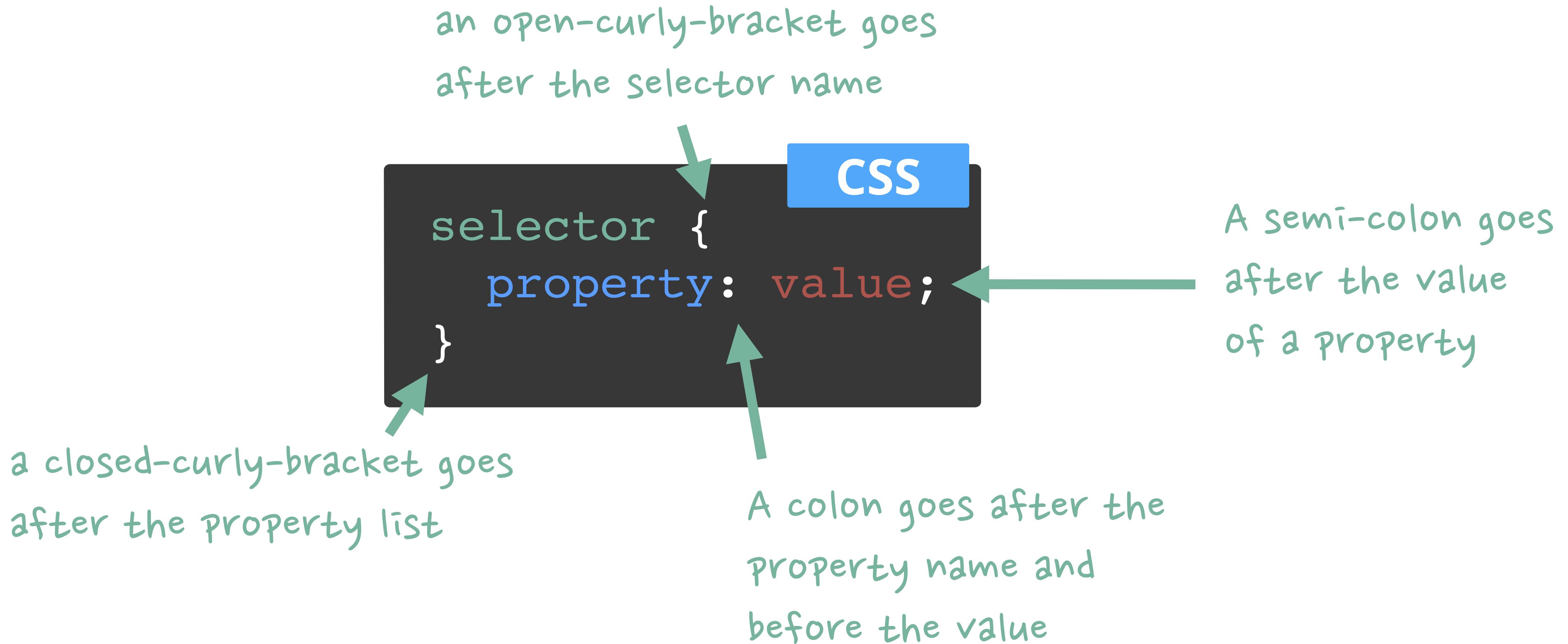
Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

each property can usually be set  
to one of a few different values



# The syntax of a selector

Selectors have a very specific syntax that needs to be followed or the CSS won't work.



# Using multiple properties in one selector

A single selector can change multiple properties.

HTML

```
<p>Magic Cake is one of the tastiest...</p>
```

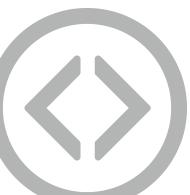
CSS

```
p {  
    text-decoration: underline;  
    color: red;  
}
```

multiple properties applied to a  
single selector

## Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.



# Selecting multiple tags with one selector

Selectors will select all matching tags on the page and apply properties.

## HTML

```
<h3>Ingredients</h3>
<ul>
  <li>2 eggs</li>
  <li>1 pound of sugar</li>
  <li>3 sticks of butter</li>
</ul>
```

## CSS

```
li {
  font-size: 24px;
}
```

## Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

### Ingredients

- 2 eggs
- 1 pound sugar
- 3 sticks of butter

bigger font size now  
for all `<li>` tags

### Directions

1. Mix eggs, sugar, and butter in a large bowl
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!



# Selecting only tags that are children of another tag

A **descendent selector** can be used to select tags only if they are children of another tag.

```
<h3>Ingredients</h3>
<ul>
  <li>2 eggs</li>
  <li>1 pound of sugar</li>
  <li>3 sticks of butter</li>
</ul>
```

HTML

select only li tags that are children of ul tags

```
ul li {
  font-size: 24px;
}
```

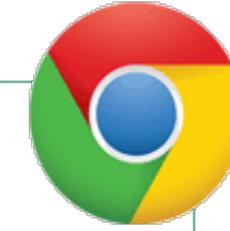
CSS

## Ingredients

- 2 eggs
  - 1 pound sugar
  - 3 sticks of butter
- only the <ul> list items are bigger now

## Directions

1. Mix eggs, sugar, and butter in a large bowl
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!



**TIP:** read the selector list from right-to-left. The left-most selector is the parent, with children following to the right.



# Selecting tags based on actions and conditions

A **pseudo-selector** is a modifier that can be added to a selector to select a tag only when a certain condition has occurred.

CSS

```
a {  
    text-decoration: none;  
}
```

removes the underline from all links on a page

```
a:hover {  
    text-decoration: underline;  
    color: darkred;  
}
```

adds an underline and changes the color of all  
links on a page ONLY when the mouse is over  
the link

## Recipe World

- Home
- Recipes
- Suggest



# Using pseudo-selectors to narrow selection criteria

The **:first-child** pseudo-selector can be applied to narrow the amount of child tags selected

**HTML**

```
<h3>Directions</h3>
<ol>
  <li>Mix eggs, sugar...</li>
  <li>Spread into a...</li>
  <li>Bake at 350...</li>
</ol>
```

select only the first li tag

**CSS**

```
ol li:first-child {
  color: red;
}
```

## Directions

1. Mix eggs, sugar, and butter in a large bowl
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!



# Where do we put all this stuff?

One place you can put CSS is in a **style** tag that's a child of the **head** tag.

HTML

```
<html>
  <head>
    <style type="text/css">
      a {
        color: red;
      }
    </style>
  </head>
  <body>
    ...
  </body>
</html>
```

The `type` attribute lets the browser know that CSS is coming

Your CSS selectors can be written in between the **style** opening and closing tags.



# Hexadecimal colors

A popular way to choose colors in CSS is to use hexadecimal numbers.

CSS

```
p {  
  color: red;  
  color: #FF0000;  
  
  color: black;  
  color: #000000;  
  
  color: yellow;  
  color: #FFFF00;  
}
```

Here's three keyword colors  
and their hexadecimal  
equivalents.

red

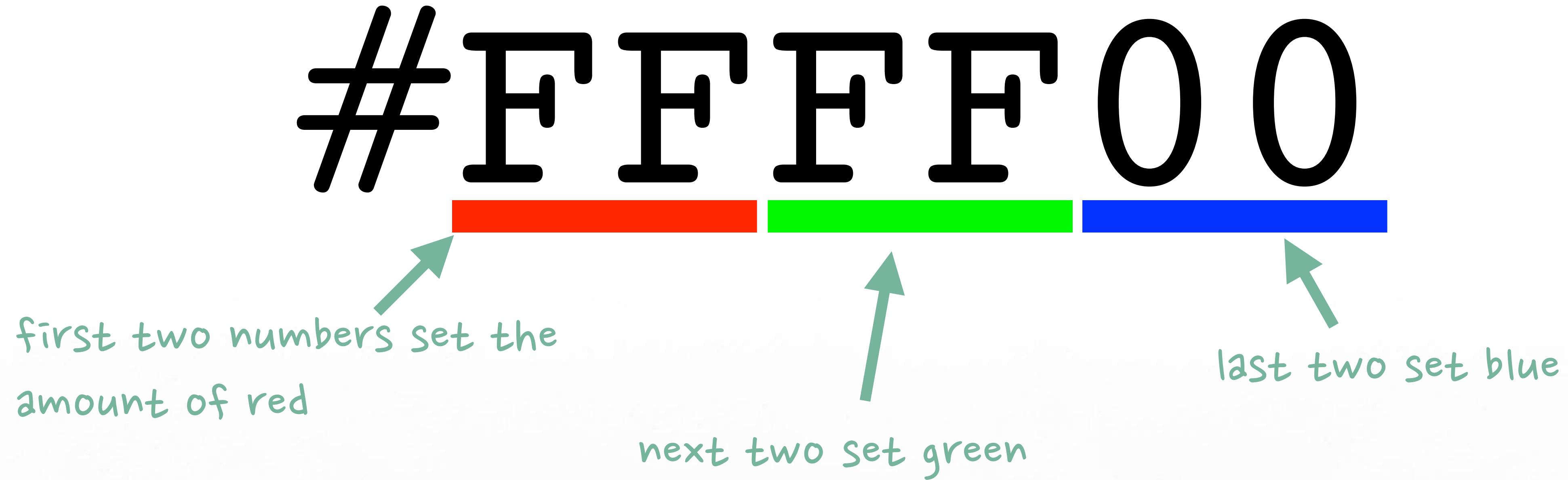
black

yellow



# Hexadecimal colors

Each hex color is actually three different parts.

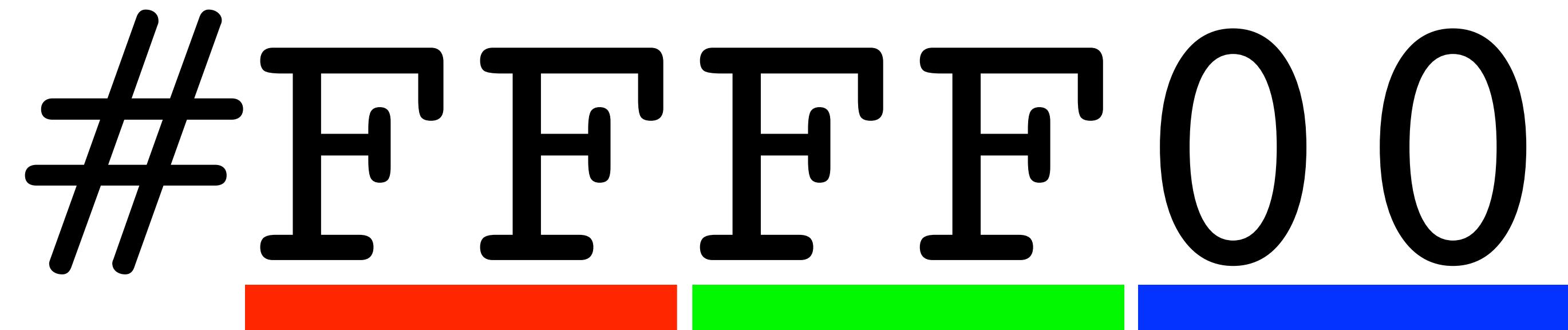


# Hexadecimal colors - Minimum and Maximum

---

Color values for red, green, and blue run from **0** - **255** decimal,  
or **00** - **FF** hexadecimal.

#**FFFF00**



The text '#FFFF00' is displayed in a large, bold, black font. The last two digits, '00', are underlined. Below the underline are three horizontal bars: a red bar, a green bar, and a blue bar. The red bar is at the bottom of its range (0), the green bar is at its maximum (255), and the blue bar is also at its maximum (255).

This color is maximum red, maximum green, and no blue.

That's equivalent to the color **Yellow**



# Breaking down decimal numbers

Decimal numbers run from 0-9.

76

(decimal)

multiply the left by 10

$$(7 \times 10)$$

multiply the right by 1

$$(6 \times 1)$$

Add them together, and you get 76 decimal



# Breaking down hexadecimal numbers

Since we're so used to reading decimal numbers, sometimes it makes sense to convert hexadecimal to decimal.

76

(hexadecimal)

multiply the left by 16

( 7   x   16 )

multiply the right by 1

( 6   x   1 )

Add them together, and you get **118** decimal



# Breaking down hexadecimal numbers

Since we're so used to reading decimal numbers, sometimes it makes sense to convert hexadecimal to decimal.

**FF**

(hexadecimal)



multiply the left by 16

$$(F \times 16)$$

multiply the right by 1

$$(F \times 1)$$

?

?

?

?

How do we multiply by a letter?!\$#!\$?



# Understanding letter-based hexadecimal numbers

Hexadecimal numbers run from 0-15, but use letters for numbers above 9.

$$A = 10$$

$$B = 11$$

$$C = 12$$

$$D = 13$$

$$E = 14$$

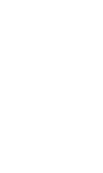
$$F = \underline{15}$$

FF



multiply the left by 16

$$(\underline{15} \times 16)$$



multiply the right by 1

$$(\underline{15} \times 1)$$



Add them together, and you get  $240 + 15$ , or **255** decimal



# Calculating the color from hexadecimal numbers

---

#7403AB



Red:  $(7 \times 16) + (4 \times 1) = 116$

Green:  $(0 \times 16) + (3 \times 1) = 3$

Blue:  $(10 \times 16) + (11 \times 1) = 171$

This color is some red, almost no green, and more blue than red.

That's a color that looks **Purple**



# Calculating the color from hexadecimal numbers

---

#FF00B3



Red:  $(15 \times 16) + (15 \times 1) = 255$

Green:  $(0 \times 16) + (0 \times 1) = 0$

Blue:  $(11 \times 16) + (3 \times 1) = 179$

This color is lots of red, no green, and a little less blue than red.

That's a color that looks **Hot Pink**



# Colors on the web

---

By using hexadecimal numbers to set colors, we get 256 possible combinations for each of the three color channels.

or...

$$256 \times 256 \times 256 = 16,777,216$$



# Our page after changing some styles with CSS

## Questions

How does our page know how much vertical and horizontal space to put between tags?

How can we control that spacing?

why this much space?

## Answer

The box model

## Recipe World

- [Home](#)
- [Recipes](#)
- [Suggest](#)

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known about it, and these are their stories.

We also need to stress that whatever happens to you as a result of this cake is your responsibility. You've been warned, and you can view our [legalese](#) if you like.

### Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter



# The Box

Every tag shown in the body is contained in an invisible rectangle that we'll call **the box**.

HTML

```
<p>Magic Cake is one of...</p>
<p>We also need to stress...</p>
```

These two paragraph tags are shown as two boxes stacked on top of each other.

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

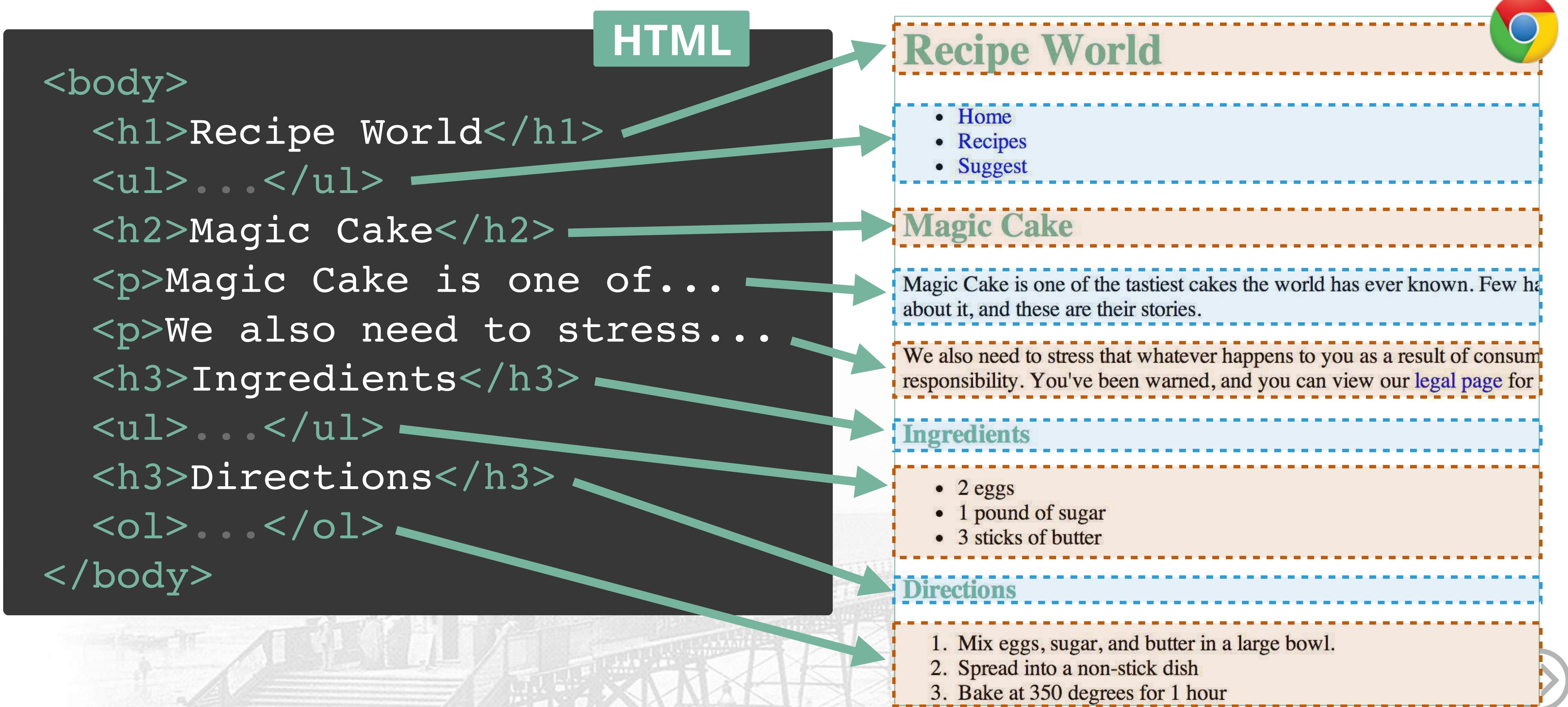


content  
boxes



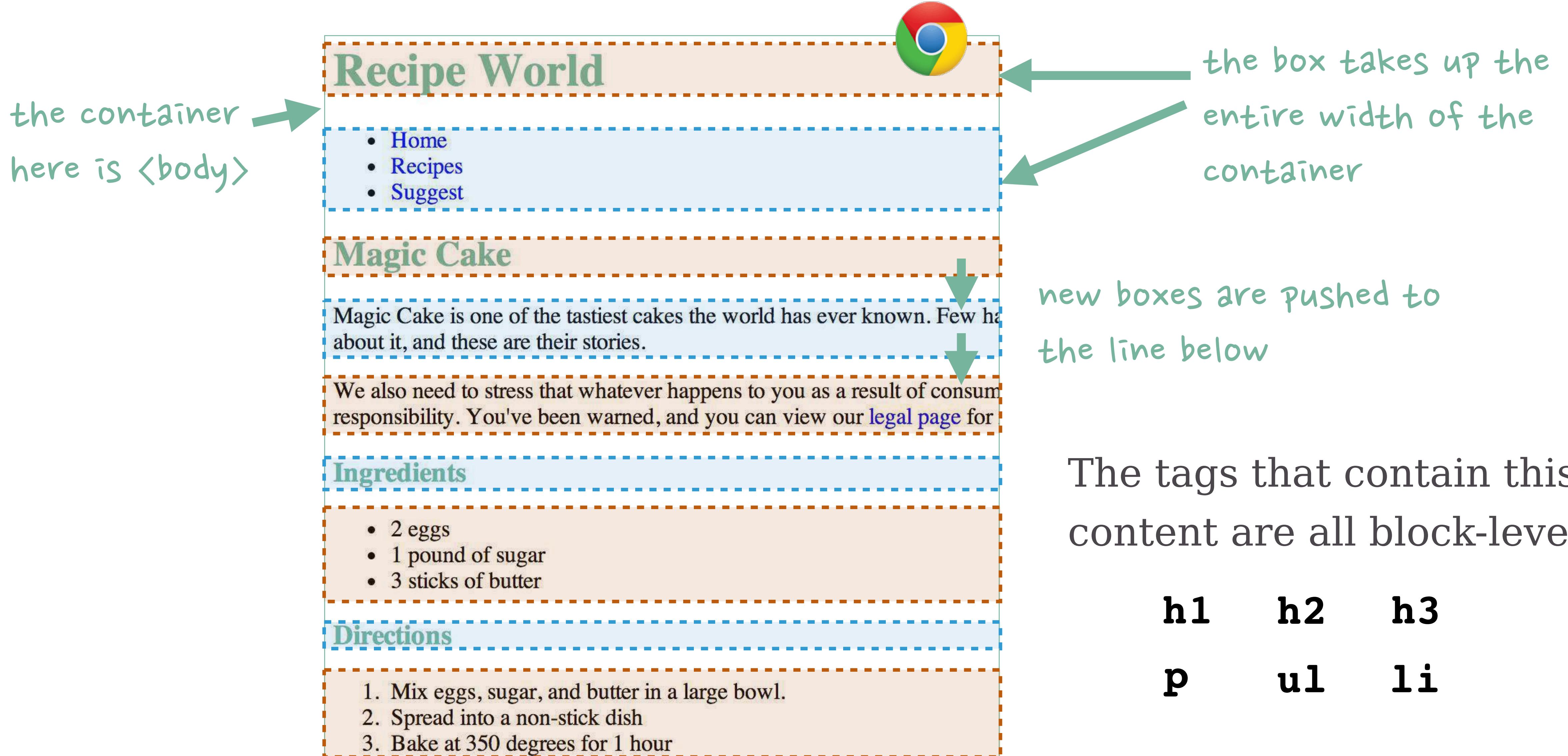
# The boxes on our page

Here are the boxes for the current version of the magic-cake.html page.



# Block-level tags

The content of **block-level** tags take up the entire width of the **container**.



# Inline-level tags

If a tag is not block-level, it's **inline-level**.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.



inline-level tags don't try to take  
up more width than they need



HTML

```
<p>...can view our <a href="legal.html">legal page</a>  
for more information</p>
```

Some common inline-level tags

**a      img**

**input    label**



# Turning block-level into inline-level

Sometimes, you'll want block-level tags to be inline-level.

**Example:** list items displaying horizontally instead of vertically

## block-level list items

Recipe World



- Home
- Recipes
- Suggest

the default display is block-level

## inline-level list items

Recipe World



Home Recipes Suggest

CSS

```
ul li {  
    display: inline;  
}
```



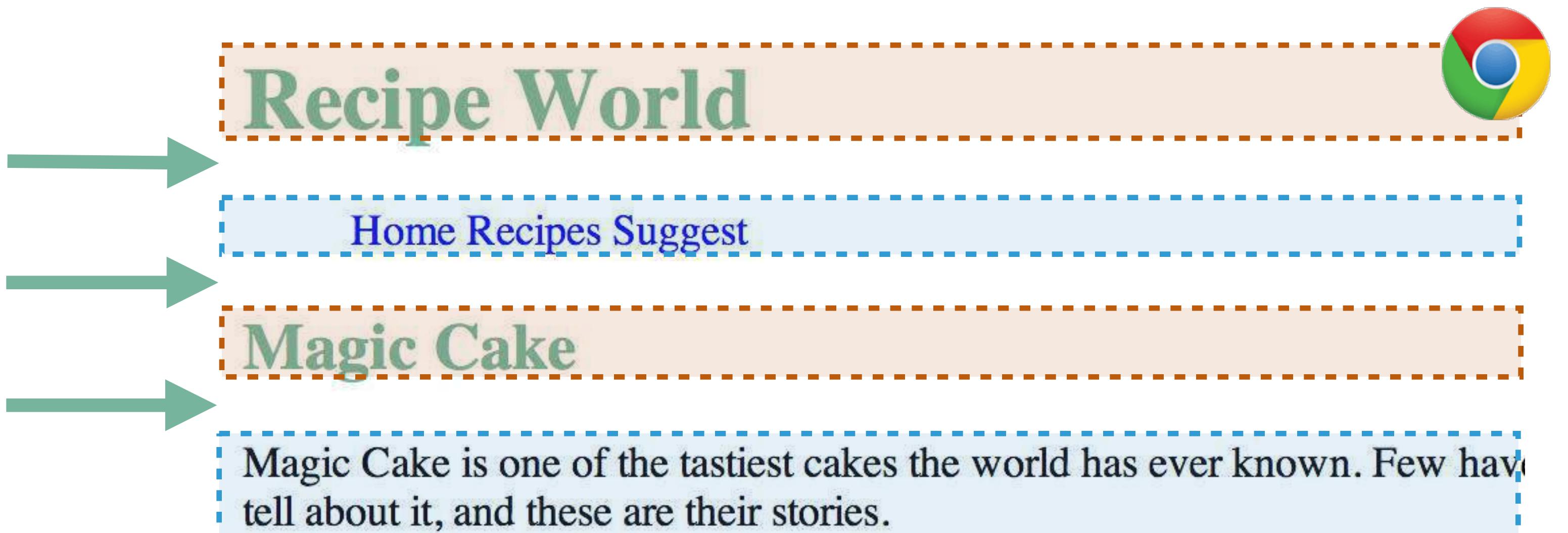
# Block-level vertical spacing

## What we know

- Each tag's content fits in an invisible box
- Each block-level tag's box takes up an entire line (horizontal space)

## Back to our problem

How does the browser  
know how much space  
goes between the boxes?



# The Box Model

---

The **box model** is a way to describe the borders and spacing in between the **boxes** of each tag.

- There are 4 parts of the box model:

1. Content area

The **content area** contains your actual content (text, images, etc.)



The content area will only take up as much vertical space as it needs to display the content inside.



# The Box Model

---

- There are 4 parts of the box model:

## 2. Padding

Padding is added to the **top**, **right**, **bottom**, or **left** of the content area.



# The Box Model

---

- There are 4 parts of the box model:

## 3. Border

Borders are added around the **top, right, bottom, or left** of the padding

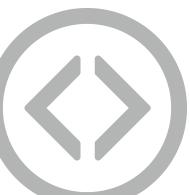


# The Box Model

- There are 4 parts of the box model:

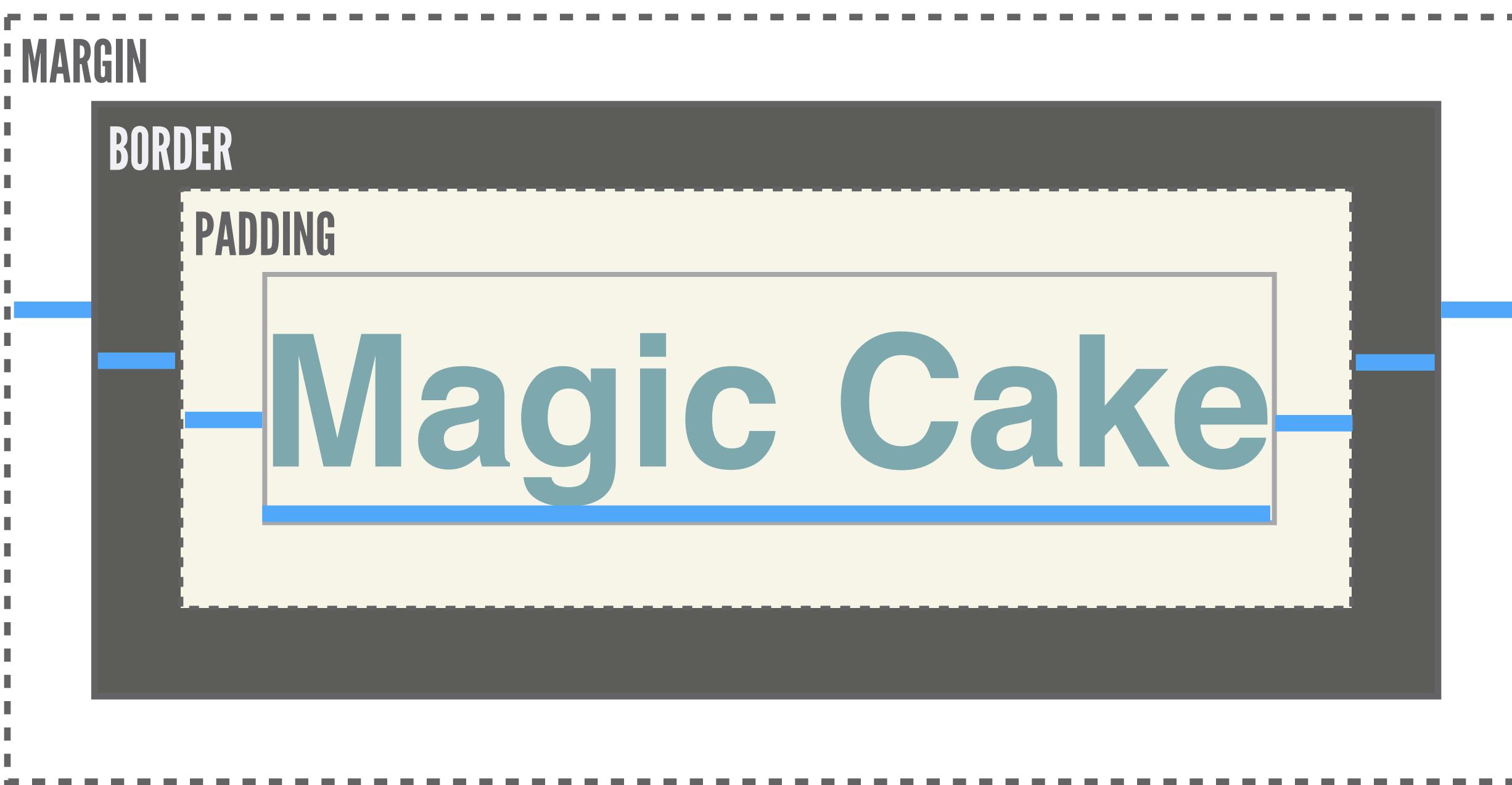
## 4. Margin

Margins are added to the **top**, **right**, **bottom**, or **left** of the border.



# How to calculate the size of the box

The full size of a box after these four properties have been set can be calculated like this:



- + content area width
  - + padding-left + padding-right
  - + border-left + border-right
  - + margin-left + margin-right
- = the full box width

use top, bottom, and  
height to calculate  
full box height



# Applying the box model properties

Apply padding to one side at a time...

CSS

```
h2 {  
  padding-top: 6px;  
  padding-right: 3px;  
  padding-bottom: 0;  
  padding-left: 0;  
}
```

If we want to put some space above and to the right of some text, one option is to adjust the padding.

PADDING

Magic Cake

...or all at once in a clockwise order

CSS

```
h2 {  
  padding: 6px 3px 0 0;  
}  
      top    right   bottom  left
```



# Applying the box model properties

You can apply borders all at once and not specify a side.

CSS

```
h2 {  
    border-width: 6px;  
    border-style: solid;  
    border-color: black;  
}
```

BORDER

PADDING

Magic Cake

Creating borders has a shortcut version, too.

CSS

```
h2 {  
    border: 6px solid black;  
}
```

width      style      color



# Applying the box model properties

You can also just pick one side and just add a border there.

CSS

```
h2 {  
    border-bottom: 6px solid black;  
}  
width style color
```

PADDING

## Magic Cake

BORDER

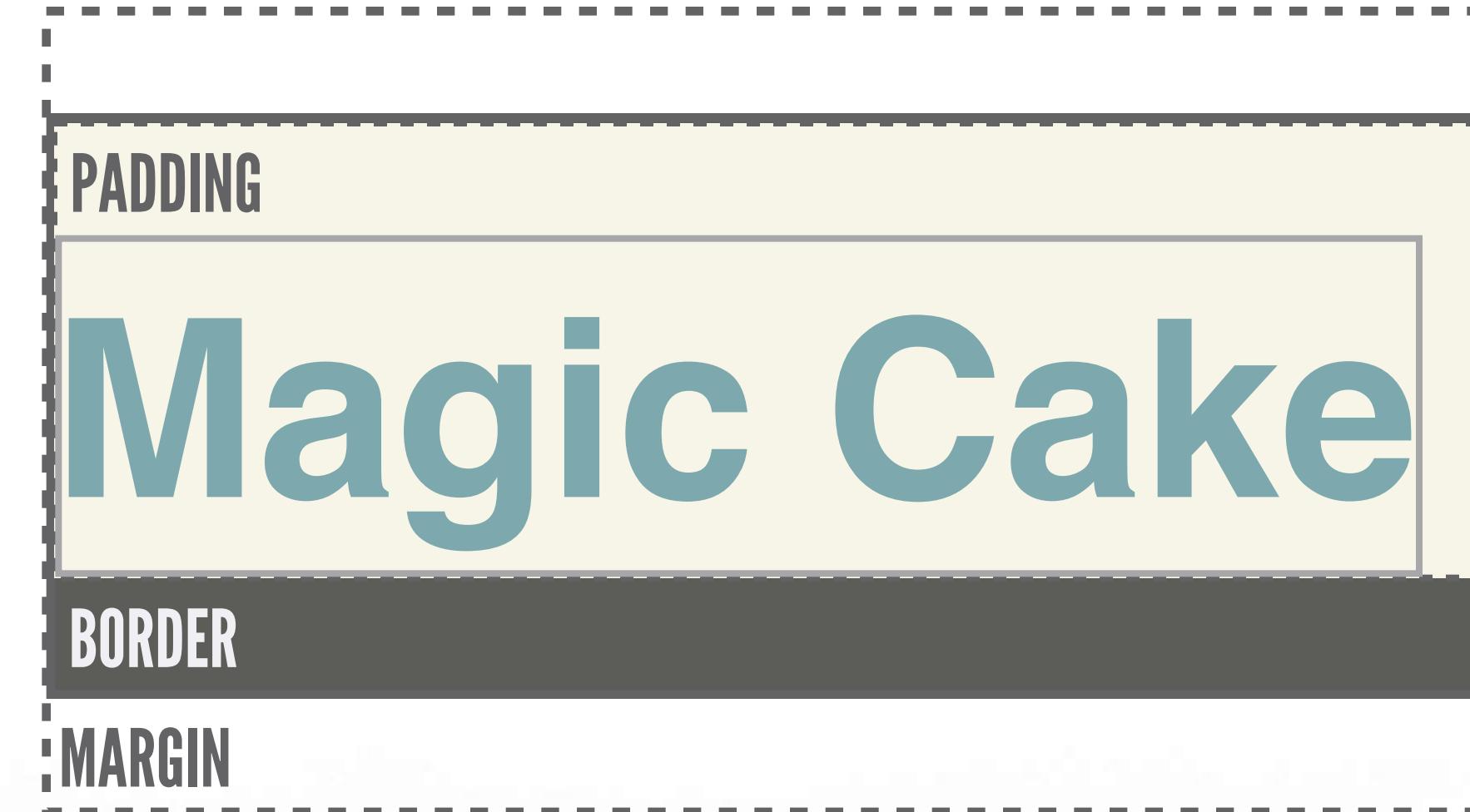


# Applying the box model properties

Add margins the same way you add padding...

CSS

```
h2 {  
    margin-top: 6px;  
    margin-right: 0;  
    margin-bottom: 6px;  
    margin-left: 0;  
}
```



...or use the same shortcut syntax as padding.

CSS

```
h2 {  
    margin: 6px 0 6px 0;  
}
```

top right bottom left



# When should you use padding?

Padding is used to control the size of a box *without* adjusting the size of the content inside the box.

CSS

```
h2 {  
    padding: 0 0 0 0;  
}
```

Home Recipes Suggest

Magic Cake

Magic Cake is one of the tastiest cakes tell about it, and these are their stories.

CSS

```
h2 {  
    padding: 10px 0 10px 0;  
}
```

Home Recipes Suggest

Magic Cake

Magic Cake is one of the tastiest cakes tell about it, and these are their stories.



# When should you use margin?

Margin is used to control the space between boxes.

CSS

```
h2 {  
    margin: 20px 0 20px 0;  
}
```

CSS

```
h2 {  
    margin: 4px 0 4px 0;  
}
```

Home Recipes Suggest

Magic Cake

Magic Cake is one of the tastiest cakes tell about it, and these are their stories.

Home Recipes Suggest

Magic Cake

Magic Cake is one of the tastiest cakes tell about it, and these are their stories.



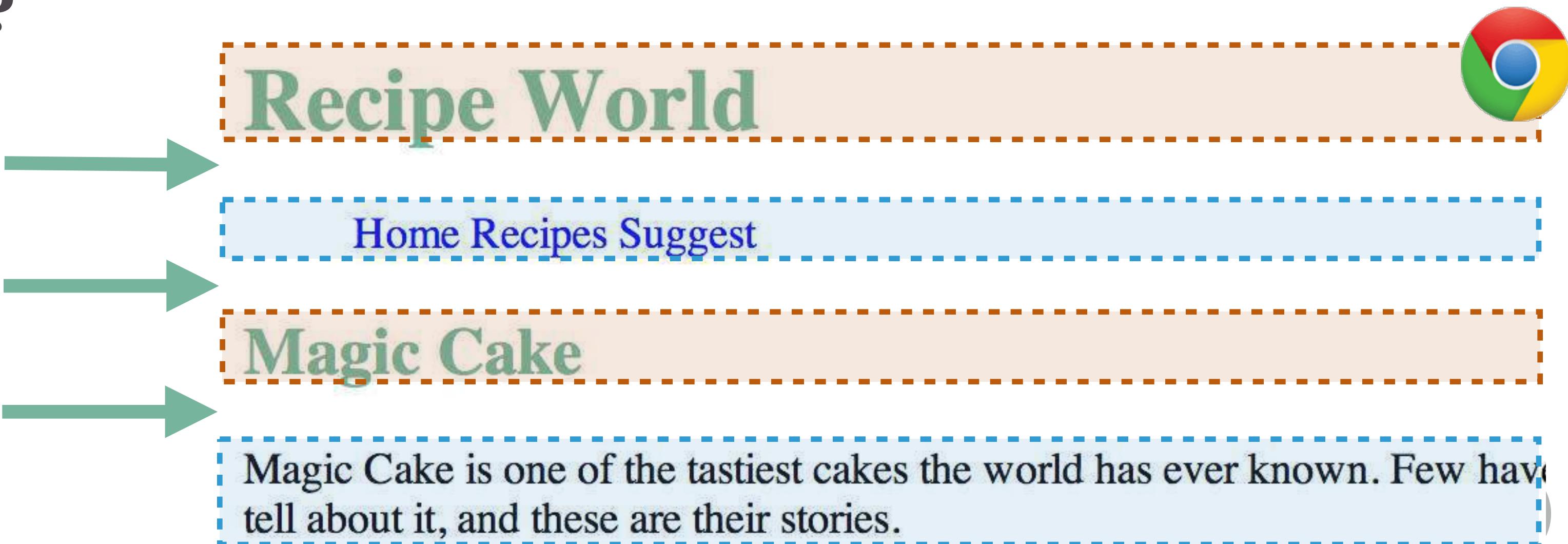
# Summary of our new box model knowledge

## What we now know:

- Each tag's content fits in an invisible box
- Each block-level tag's box takes up an entire line
- Padding can be used to adjust spacing within a container
- Margin can be used to adjust spacing between containers

## Why is there spacing here?

We didn't set any margins or padding?!@!#?#



# Default browser styles

Browsers actually have a default stylesheet for when no custom styles are set.

## Default box styles left on

### Recipe World

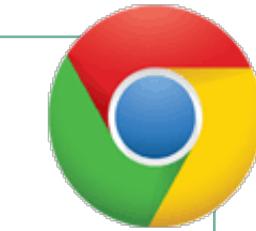
Home Recipes Suggest

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few of these are their stories.

We also need to stress that whatever happens to you as a result of consuming this cake, it's your own responsibility. You've been warned, and you can view our [legal page](#) for more information.

### Ingredients



## Default box styles turned off

### Recipe World

Home Recipes Suggest

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few of these are their stories.

We also need to stress that whatever happens to you as a result of consuming this cake, it's your own responsibility. You've been warned, and you can view our [legal page](#) for more information.

### Ingredients

2 eggs 1 pound of sugar 3 sticks of butter

### Directions

Mix eggs, sugar, and butter in a large bowl.

Spread into a non-stick dish

Bake at 350 degrees for 1 hour

Let sit at room temperature for 20 minutes

Eat and enjoy!



# Resetting default browser styles

Add this to the very top of your CSS file to reset default styles.

include each tag that's in your  
HTML for that page

CSS

```
html, body, h1, h2,  
h3, p, ol, ul, li, a {  
    padding: 0;  
    border: 0;  
    margin: 0;  
}
```

Now, all of the default box properties are reset, and we need to set the box model properties ourselves.

Default box styles turned off

Recipe World

Home Recipes Suggest

Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known about it, and these are their stories.

We also need to stress that whatever happens to you as a result of responsibility. You've been warned, and you can view our [legal](#)

Ingredients

2 eggs 1 pound of sugar 3 sticks of butter

Directions

Mix eggs, sugar, and butter in a large bowl.

Spread into a non-stick dish

Bake at 350 degrees for 1 hour

Let sit at room temperature for 20 minutes

Eat and enjoy!



# A Box Model process - Start with the body

The order that you apply the box model properties is personal preference, but here's one approach:

Start from the highest parent element

This is often <body>

```
CSS  
body {  
  padding: 20px 20px 20px 20px;  
}
```

This padding has the effect of pushing all of the children away from the edges of the <body>.

this padding is applied to every side of the body



## Recipe World

Home Recipes Suggest

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. I to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of c your responsibility. You've been warned, and you can view our leg information.

#### Ingredients

2 eggs 1 pound of sugar 3 sticks of butter

#### Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!



# A Box Model process - Next, look at headings

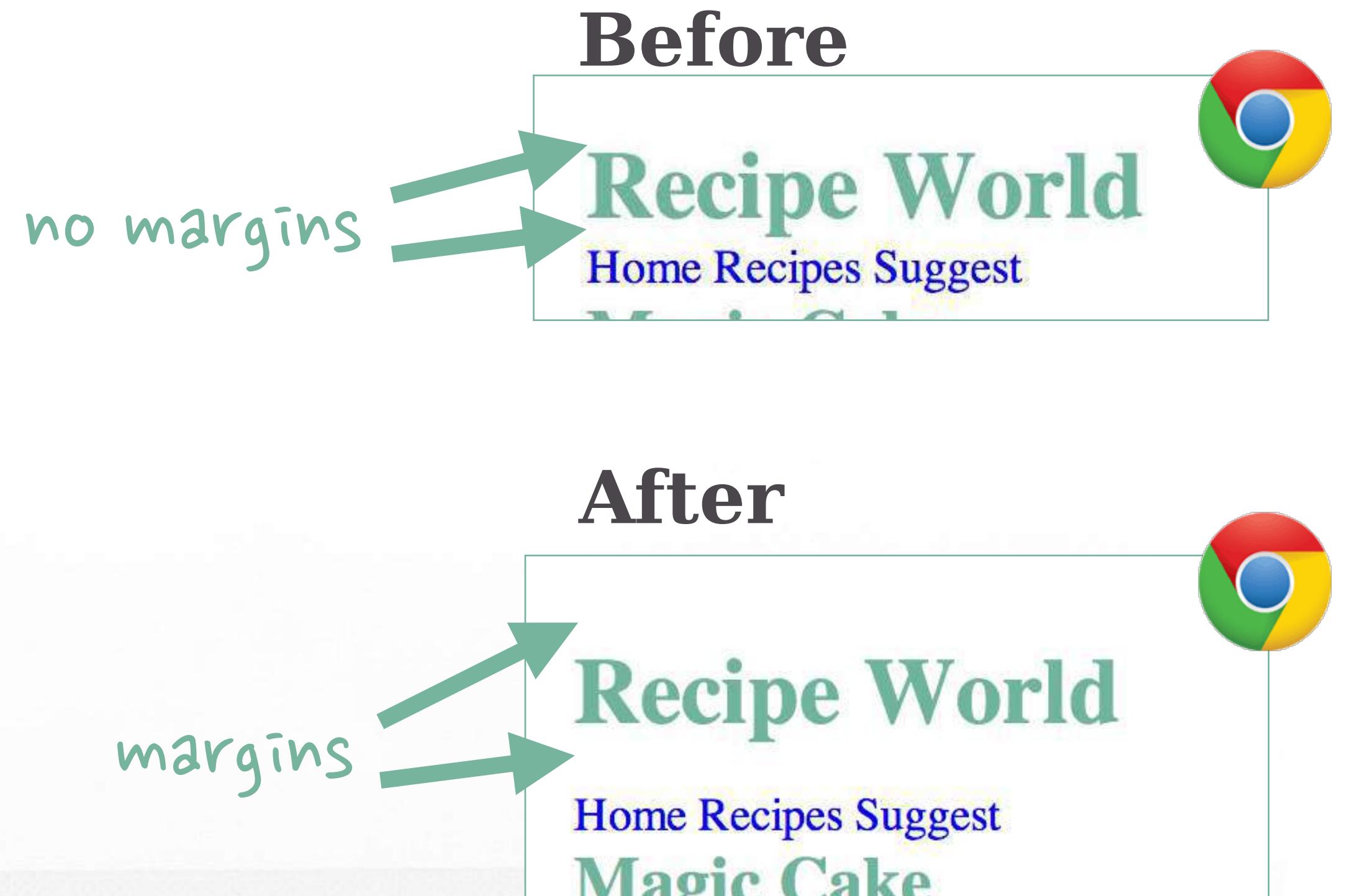
Next, focus on the heading tags

<h1>

CSS

```
h1 {  
    margin: 10px 0 15px 0;  
}
```

Use margin to put vertical space between tags.



# A Box Model process - Next, look at headings

Next, focus on the heading tags

<h2>

CSS

```
h2 {  
    margin: 10px 0 20px 0;  
}
```

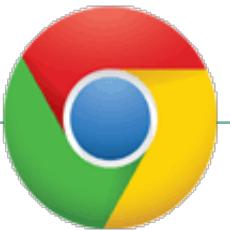
no margins

Before

Home Recipes Suggest

**Magic Cake**

Magic Cake is one of the tastiest cakes the world has to tell about it, and these are their stories.  
We also need to stress that whatever happens to you



margins

After

Home Recipes Suggest

**Magic Cake**

Magic Cake is one of the tastiest cakes the world has to tell about it, and these are their stories.



# A Box Model process - Next, look at headings

Next, focus on the heading tags

<h3>

CSS

```
h3 {  
  margin: 15px 0 15px 0;  
}
```

no margins

**Before**

Magic Cake is one of the tastiest cakes to tell about it, and these are their stories. We also need to stress that whatever happens, your responsibility. You've been warned, and here's some information.

**Ingredients**

2 eggs 1 pound of sugar 3 sticks of butter



**After**

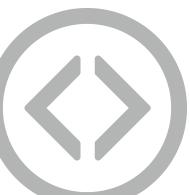
Magic Cake is one of the tastiest cakes to tell about it, and these are their stories. We also need to stress that whatever happens, your responsibility. You've been warned, and here's some information.



margins

**Ingredients**

2 eggs 1 pound of sugar 3 sticks of butter



# A Box Model process - borders

After your margins are in place, think about where borders might be necessary.

CSS

```
h3 {  
  margin: 15px 0 15px 0;  
  border-bottom: 1px solid #CCCCCC;  
}
```

this bottom border visually  
separates the sections

We also need to stress that whatever happens, it's your responsibility. You've been warned, and you have information.

## Ingredients

2 eggs 1 pound of sugar 3 sticks of butter

## Directions



# A Box Model process - padding

Use padding to put more space between the content and border.

CSS

```
h3 {  
  margin: 15px 0 15px 0;  
  border-bottom: 1px solid #CCCCCC;  
  padding-bottom: 3px;  
}
```

Before



After



now we've got a little  
more space between  
the content and border



# A Box Model process - padding

Padding is also used to adjust list and list item indentation.

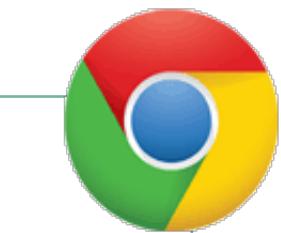
## Before

### Ingredients

2 eggs 1 pound of sugar 3 sticks of butter

### Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!



## After

### Ingredients

→ 2 eggs 1 pound of sugar 3 sticks of butter

### Directions

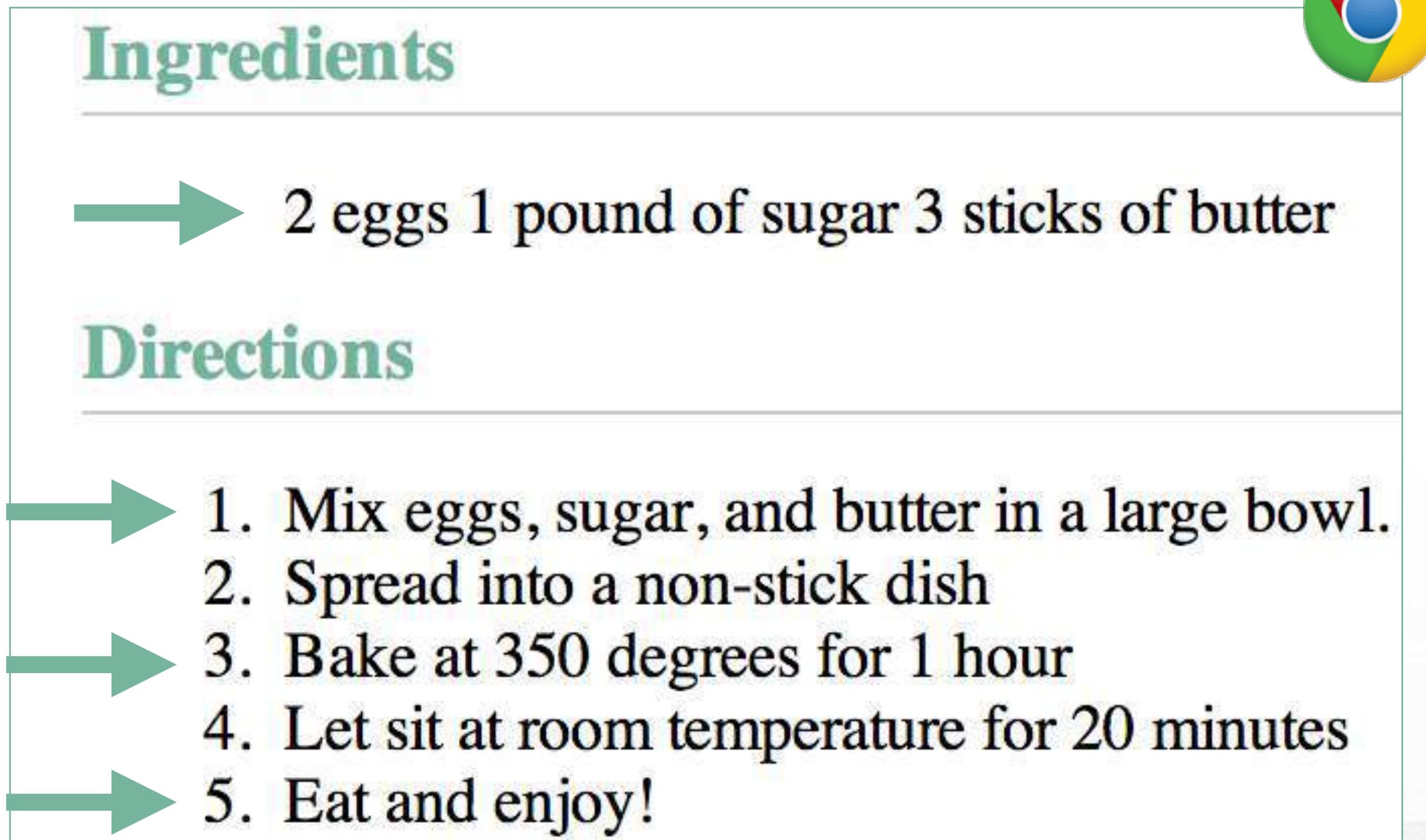
- 1. Mix eggs, sugar, and butter in a large bowl.
- 2. Spread into a non-stick dish
- 3. Bake at 350 degrees for 1 hour
- 4. Let sit at room temperature for 20 minutes
- 5. Eat and enjoy!



# A Box Model process - padding

Padding is also used to adjust list and list item indentation.

After



The screenshot shows a web browser window displaying a recipe card. At the top left is a green "Ingredients" section containing a list of items with green arrows pointing to them. Below it is a "Directions" section containing a numbered list of steps with green arrows pointing to them. The browser's address bar and other UI elements are visible at the top, and a blurred background image of a roller coaster is visible behind the card.

**Ingredients**

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

**Directions**

- 1. Mix eggs, sugar, and butter in a large bowl.
- 2. Spread into a non-stick dish
- 3. Bake at 350 degrees for 1 hour
- 4. Let sit at room temperature for 20 minutes
- 5. Eat and enjoy!

CSS

```
ul {  
    padding: 0 0 0 50px;  
}  
  
ol {  
    padding: 0 0 0 50px;  
}
```

left padding



# Result after adjusting box model properties

Here's where our page stands right now:

The screenshot shows a web browser window with a green header bar containing the text "Result after adjusting box model properties". Below the header is a navigation bar with links "Home", "Recipes", and "Suggest". The main content area features a large title "Recipe World" and a sub-section "Magic Cake". A detailed description of the cake follows, along with sections for "Ingredients" and "Directions". The "Directions" section contains a numbered list of five steps. The entire content is framed by a light gray border.

**Recipe World**

[Home](#) [Recipes](#) [Suggest](#)

## Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and live. We also need to stress that whatever happens to you as a result of consuming Magic Cake is view our [legal page](#) for more information.

### Ingredients

2 eggs 1 pound of sugar 3 sticks of butter

### Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!



# Fixing page style problems

## PROBLEM

The `<ul>` padding is being applied to the navigation and ingredients.

We need a way to add padding to **one** `<ul>` but not the other.



The screenshot shows a web browser window with a light gray background. At the top right is the Google Chrome logo. The main content area displays a website titled "Recipe World". Below the title is a horizontal menu bar with three items: "Home", "Recipes", and "Suggest". A large green arrow points from the "PROBLEM" section on the left towards this menu. The main content area features a section titled "Magic Cake" with a sub-section "Ingredients" below it. The "Ingredients" section contains the text "2 eggs 1 pound of sugar 3 sticks of butter". Below this is a section titled "Directions" which lists five numbered steps: 1. Mix eggs, sugar, and butter in a large bowl. 2. Spread into a non-stick dish. 3. Bake at 350 degrees for 1 hour. 4. Let sit at room temperature for 20 minutes. 5. Eat and enjoy!



# Using classes to differentiate between tags

Add a **class** attribute to an HTML tag so that you can access it directly in CSS...

HTML

```
<ul class="nav"><li>...</li></ul>
```

... then access this class in CSS by putting a period in front of the class name.

CSS

```
.nav {  
  padding-left: 0;  
}
```

padding-left is gone from the nav <ul>

padding-left remains on the ingredients <ul>

## Recipe World

[Home](#) [Recipes](#) [Suggest](#)

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever seen. We also need to stress that whatever happens to you as you view our [legal page](#) for more information.

#### Ingredients

2 eggs 1 pound of sugar 3 sticks of butter



# Using classes in descendant selectors

Classes can be used interchangeably with tags, so the way descendant selectors work doesn't change.

HTML

```
<ul class="nav">
  <li><a href="home.html">Home</a></li>
  <li><a href="recipes.html">Recipes</a></li>
  <li><a href="suggest.html">Suggest</a></li>
</ul>
```

CSS

```
.nav a {
  color: #0000af;
  text-decoration: none;
}
```

This selector says *find all **a** tags that are children of **any tag** with a **class of “nav”***, and set the color to blue and turn off the underline.



# Using classes to differentiate between tags

**Problem:** We don't want the ingredients list to display inline

our current rule for list items

CSS

```
ul li {  
    display: inline;  
}
```

should be the default block-level, not inline

## Recipe World

Home Recipes Suggest

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever seen.  
We also need to stress that whatever happens to you as you view our [legal page](#) for more information.

#### Ingredients

2 eggs 1 pound of sugar 3 sticks of butter



# Using classes to differentiate between tags

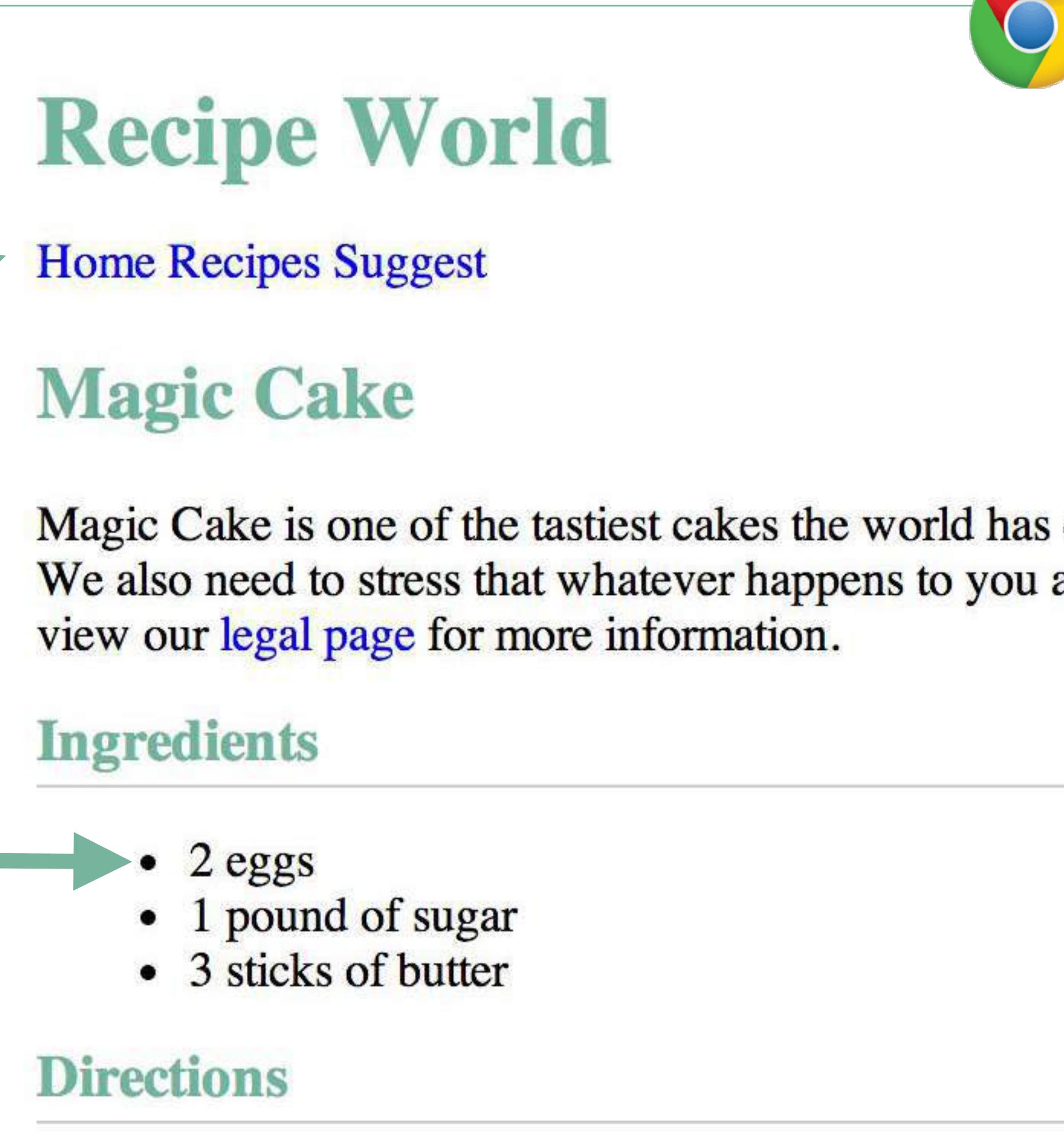
Classes can be used interchangeably with tags, so the way descendant selectors work doesn't change.

CSS

```
ul li {  
    display: inline;  
}  
  
.nav li {  
    display: inline;  
}
```

displaying as a  
block-level tag  
again (the  
default)

still displayed  
inline



The screenshot shows a web browser window displaying a recipe page titled "Recipe World". The page includes a navigation bar with "Home", "Recipes", "Suggest", and a search bar. The main content features a section for "Magic Cake" with a description, ingredients, and directions. A green arrow points from the explanatory text on the left to the "display: inline;" rule in the CSS code block, and another green arrow points from the "display: inline;" rule to the "Magic Cake" section of the page.

## Recipe World

Home Recipes Suggest

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever seen. We also need to stress that whatever happens to you after you've eaten it, don't worry, we'll help you get back on track. Just view our [legal page](#) for more information.

#### Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

#### Directions

# Writing class and type selectors in the right order

When you're just using type selectors, the order doesn't really matter.

When you start combining them with class selectors - the order matters.

 CSS

```
.nav {  
  padding-left: 0;  
}  
  
ul {  
  padding: 0 0 0 50px;  
}
```

padding-left!!!



Padding for **.nav** before **ul** rule  
**padding-left: 0;**

Padding for **.nav** after **ul** rule  
**padding-left: 50px;**

This **ul** padding overrides the **.nav** padding  
because it comes after the **.nav** in the stylesheet.



# Writing class and type selectors in the right order

Generally speaking, first declare the broadest rules with type selectors, and then get more specific with class selectors.

A diagram illustrating CSS selector ordering. A dark grey rectangular box contains CSS code. In the top right corner is a blue button-like shape with the word "CSS". On the left side of the box, there is a green circle with a white checkmark. A green curved arrow points from this checkmark towards the "ul" selector. Inside the box, the CSS code is:

```
ul {           broader
  padding: 0 0 0 50px;
}

.nav {        more specific
  padding-left: 0;
}
```

Padding for **.nav**  
before **.nav** rule  
**padding-left: 50px;**

Padding after **.nav** rule  
**padding-left: 0;**

flip them so the  
ul comes first,  
and then the .nav

This way, the broader rules **cascade** down and become defaults,  
and more specific rules can change the defaults.



# Creating a page that lists all recipes

Our recipes.html page will have a list of links to individual recipe pages.

## recipes.html

```
<html>
  <head>
  </head>
  <body>
    <h1>Recipe World</h1>
    <ul class="nav">
      <li><a href="home.html">Home</a></li>
      <li><a href="recipes.html">Recipes</a></li>
      <li><a href="suggest.html">Suggest</a></li>
    </ul>
    <h2>Recipes</h2>
  </body>
</html>
```

HTML

right now, this page just has  
the main site navigation and an  
h2 describing the page content



# Creating the list of recipes

The list of recipes can be an unordered list where each list item contains a few other block-level tags and a link to the individual recipe page.

## recipes.html

HTML

```
<h2>Recipes</h2>
<ul>
  <li>
    <h3><a href="magic-cake.html">Magic Cake</a></h3>
    <p>This is a dangerously delicious cake.</p>
  </li>
  <li>...</li> ← copy this general structure for each recipe
  <li>...</li>
  <li>...</li>
</ul>
```



# The recipes page has a problem

## Problem:

None of our styles are in recipes.html because we put our CSS in the **<head>** of magic-cake.html.

## Solution:

We need to put our CSS in a separate file and link that file to each of our HTML pages.

## Recipe World

- [Home](#)
- [Recipes](#)
- [Suggest](#)

### Recipes

- **Magic Cake**

This is a dangerously delicious cake.

- **Tacosaurus Mex**

It's always taco night somewhere!

- **Florida Omelette**

Omelette you in on a little secret - this is one tasty egg.

- **Mystery-Wich**

It's a sandwich. That's all we can really say right now.



all of our CSS  
rules are missing



# Move the CSS into a single file

Here we've created a single file called main.css so we have one place to put all of our CSS.

**magic-cake.html**

```
<html>
  <head>
    <style type="text/css">
      a {
        color: red;
      }
      all other styles
    </style>
  </head>
  ...
</html>
```

**HTML**

**main.css**

**CSS**

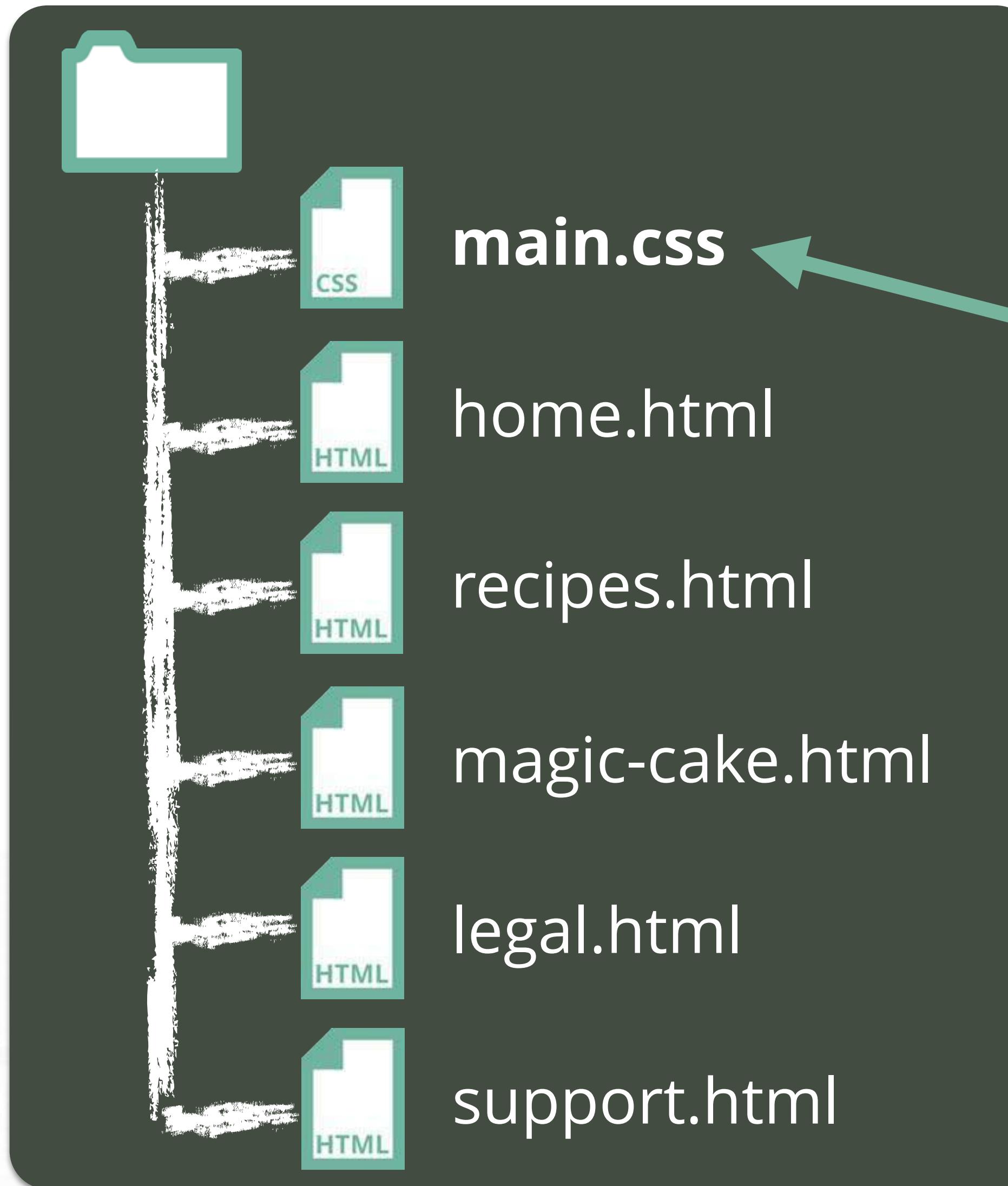


Move all of the selectors and rules into the CSS file, but **not** the HTML `<style>` tag.



# Visualizing our site files

## Recipe World server files



**main.css** will hold all of our CSS and be linked in the **<head>** of every page.

main.css

```
a {  
    color: red;  
}  
all other styles
```

CSS



# Where do you write CSS?

While CSS can be written inside of a `<style>` tag in an HTML file, most of the time it's written in a separate file and connected to the HTML file with a `link` tag.

## recipes.html

```
<html>    HTML  
    <head>  
        <link>  
    </head>  
    <body>  
        ...  
    </body>  
</html>
```

The link tag should be a child of the head tag

The link tag is an **empty** tag,  
meaning it is written with only an  
opening tag and has no closing tag.



# Where do you write CSS?

Since the tag is empty, the way you use it is by setting different **attributes**.

## recipes.html

```
<html>
  <head>
    <link type="text/css">
  </head>
  <body>
    ...
  </body>
</html>
```

HTML



this is called a **MIME type**

The **type** attribute set to **text/css** lets the browser know we're going to be loading a CSS file.



# Where do you write CSS?

Since the tag is empty, the way you use it is by setting different **attributes**.

## recipes.html

```
<html>
  <head>
    <link type="text/css" rel="stylesheet">
  </head>
  <body>
    ...
  </body>
</html>
```

HTML

The **rel** attribute is short for **relationship**.

Setting **rel** to **stylesheet** tells the HTML that it should use the linked file to determine page styles.



# Where do you write CSS?

Since the tag is empty, the way you use it is by setting different **attributes**.

**recipes.html**

```
<html>
  <head>
    <link type="text/css" rel="stylesheet" href="main.css">
  </head>
  <body>
    ...
  </body>
</html>
```

HTML



this is a relative path, so we know here that main.css should be in the same folder as our HTML files

The **href** attribute works just like it does with the **<a>** tag.



# Before-and-after linking the CSS file

Now recipes.html is using the general styles we've already created

## Before linking main.css

### Recipe World

- Home
- Recipes
- Suggest

### Recipes

- **Magic Cake**

This is a dangerously delicious cake.

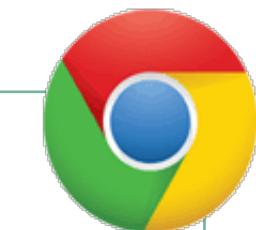
- **Tacosaurus Mex**

It's always taco night somewhere!

- **Florida Omelette**

Omelette you in on a little secret - this is one tasty egg.

- **Mystery-Wich**



## After linking main.css

### Recipe World

[Home](#) [Recipes](#) [Suggest](#)

### Recipes

- **Magic Cake**

This is a dangerously delicious cake.

- **Tacosaurus Mex**

It's always taco night somewhere!

- **Florida Omelette**

Omelette you in on a little secret - this is one tasty egg.

- **Mystery-Wich**



# Dealing with similar styles on multiple pages

Often using the exact same styles on all pages doesn't look right

## recipes.html

HTML

```
<h2>Recipes</h2>
<ul>
  <li>
    <h3><a href="magic-cake...>
      <p>This is a dangerously...
    </li>
    <li>...</li>
    <li>...</li>
    <li>...</li>
  </ul>
```

## Recipe World

Home Recipes Suggest

### Recipes

- **Magic Cake**

This is a dangerously delicious cake.

- **Tacosaurus Mex**

It's always taco night somewhere!

- **Florida Omelette**

Omelette you in on a little secret - this is one tasty egg.

- **Mystery-Wich**

It's a sandwich. That's all we can really say right now.

The **ul**, **ul li**, **h3**, and **p** styles are cascading down from our broader selectors.



# A proposal for changing the recipes page styles

no list-style-type  
for the <ul>

border around the <li>

left padding between  
the li border and  
content area

The screenshot shows a web page titled "Recipes". Below the title is a list of four items, each enclosed in a pink-bordered box. The items are:

- Magic Cake**  
This is a dangerously delicious cake.
- Tacosaurus Mex**  
It's always taco night somewhere!
- Florida Omelette**  
Omelette you in on a little secret - this is one tasty egg.
- Mystery-wich**  
It's a sandwich. That's all we can really say.

no border under the h3

our green heading color  
used for the <a> tag



# Dealing with similar styles on multiple pages

First, we'll give the `recipes.html` `<ul>` a unique class.

## recipes.html

HTML

```
<h2>Recipes</h2>
<ul class="recipes">
  <li>
    <h3><a href="magic-cake.html">Magic Cake</a></h3>
    <p>This is a dangerously delicious cake.</p>
  </li>
  <li>...</li>
  <li>...</li>
  <li>...</li>
</ul>
```



# Dealing with similar styles on multiple pages

Then, write some more specific CSS for `.recipes` and any children.

existing ul styles

CSS

```
ul {  
    padding: 0 0 0 50px;  
}
```

remember, `.recipes` is a specific `<ul>`

CSS

```
.recipes {  
    list-style-type: none;  
    padding: 0;  
}  
  
            ↗  
            resets all sides to 0
```

## Recipes

### Magic Cake

This is a dangerously delicious cake.

### Tacosaurus Mex

It's always taco night somewhere!

### Florida Omelette

Omelette you in on a little secret - this is one tasty egg.

### Mystery-Wich

It's a sandwich. That's all we can really say right now.



# Dealing with similar styles on multiple pages

Then, write some more specific CSS for `.recipes` and any children.

existing `ul li` styles

CSS

```
ul li {  
}  
}
```

the `.recipes` `<li>`'s will inherit the margins  
and add a new border and padding-left

CSS

```
.recipes li {  
    border: 1px solid #b56663;  
    padding-left: 15px;  
}
```

## Recipes

### Magic Cake

This is a dangerously delicious cake.

### Tacosaurus Mex

It's always taco night somewhere!

### Florida Omelette

Omelette you in on a little secret - this is one tasty egg.

### Mystery-Wich

It's a sandwich. That's all we can really say right now.



# Dealing with similar styles on multiple pages

Then, write some more specific CSS for `.recipes` and any children.

existing h3 styles

```
h3 {  
    margin: 15px 0 15px 0;  
    border-bottom: 1px solid #cccccc;  
    padding-bottom: 3px;  
}
```

CSS

the `.recipes h3` will inherit all of the h3 properties, but remove the bottom border

```
.recipes h3 {  
    border-bottom: 0;  
}
```

CSS

Recipes

**Magic Cake**

This is a dangerously delicious cake.

**Tacosaurus Mex**

It's always taco night somewhere!

**Florida Omelette**

Omelette you in on a little secret - this is one tasty egg.

**Mystery-Wich**

It's a sandwich. That's all we can really say right now.



# Dealing with similar styles on multiple pages

Then, write some more specific CSS for `.recipes` and any children.

We don't currently have a general CSS rule for `a`, so we can just add our specific version.

CSS

```
.recipes a {  
    color: #7facaa;  
    text-decoration: none;  
}
```

## Recipes

### Magic Cake

This is a dangerously delicious cake.

### Tacosaurus Mex

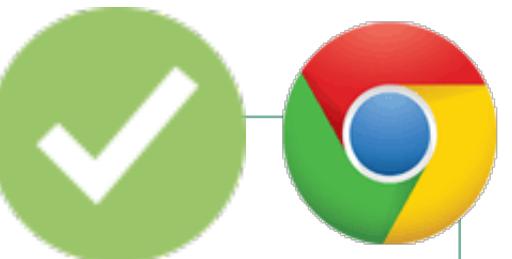
It's always taco night somewhere!

### Florida Omelette

Omelette you in on a little secret - this is one tasty egg.

### Mystery-Wich

It's a sandwich. That's all we can really say right now.



# Organizing the page layout

In our current layout, we've got a clear separation between the header and main content.



header

## Recipe World

[Home](#) [Recipes](#) [Suggest](#)

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever seen. We also need to stress that whatever happens to you as a result, please don't worry, just come back here and view our [legal page](#) for more information.

#### Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

#### Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Sprinkle cinnamon on top.

main content



# The div tag

div is short for **division**, and it's a block-level tag that's a generic way to group related content into sections on a page.

HTML

```
<body>
  <div>    header
    <h1>Recipe World</h1>
    <ul class="nav">...
  </div>
  <div>    main content
    <h2>Magic Cake</h2>
    ...
  </div>
</body>
```



# Use classes to describe what divs contain

A common way to differentiate between <div> tags is to add a class.

HTML

```
<body>
  <div class="header">
    <h1>Recipe World</h1>
    <ul class="nav">...
  </div>
  <div class="main-content">
    <h2>Magic Cake</h2>
    ...
  </div>
</body>
```

let's build this style

Recipe World

Home Recipes Suggest

Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and live. We also need to stress that whatever happens to you as a result of consuming Magic Cake i view our [legal page](#) for more information.

Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

Directions

1. Mix eggs, sugar, and butter in a large bowl.



Classes will let us style each <div> differently



# Use classes to describe what divs contain

A common way to differentiate between <div> tags is to add a class.

let the divs handle padding  
now and not the body

**HTML**

```
<body>
  <div class="header">
    <h1>Recipe World</h1>
    <ul class="nav">...
  </div>
  <div class="main-content">
    <h2>Magic Cake</h2>
    ...
  </div>
</body>
```

**CSS**

```
body {
  padding: 0 0 0 0;
}
.header {
  padding: 10px 10px 10px 10px;
}
.main-content {
  padding: 20px 20px 20px 20px;
  border: 1px solid #dddddd;
  margin: 30px 0 0 0;
}
```

# Giving the header a separate style

Let's make our header styles look different so we can easily see that the **header** is different from the **main-content**.

CSS

```
.header {  
    padding: 10px 10px 10px 10px;  
    background-color: #b56663;  
}  
.header a {  
    color: #ffffff;  
}  
h1 {  
    color: #ffffff;  
}
```

## Recipe World

[Home](#) [Recipes](#) [Suggest](#)

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and live to tell the tale. We also need to stress that whatever happens to you as a result of consuming Magic Cake is your own fault. Please view our [legal page](#) for more information.

#### Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

#### Directions

1. Mix eggs, sugar, and butter in a large bowl.



# Adjusting the size of divs

The width and height of any tag's content area can be changed in CSS.

So far, we've just wanted our tags to take up 100% of the available width, but let's make our **main-content** div a fixed size.

CSS

```
.main-content {  
    width: 500px;  
    padding: 20px;  
    border: 1px solid #dddddd;  
    margin: 30px 0 0 0;  
}
```

## Recipe World

Home Recipes Suggest

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories. We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

#### Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

#### Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish



only 500px wide now!



# Centering content

There's two main ways to center things, but the CSS you write depends on what you're centering.

If: you want to center an entire block-level tag

And: it is fixed-width

Then: set the left and right margins to auto

CSS

```
.main-content {  
    width: 500px;  
    padding: 20px;  
    border: 1px solid #dddddd;  
    margin: 30px auto 0 auto;  
}
```



The screenshot shows a web page titled "Recipe World" with a navigation bar for Home, Recipes, and Suggest. Below the header is a large white box containing a recipe for "Magic Cake". The title "Magic Cake" is centered within this box. To the right of the white box, there is a red sidebar with the Google Chrome logo at the top. A green arrow points from the word "centered!" to the centered title "Magic Cake".

**Recipe World**

Home Recipes Suggest

**Magic Cake**

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories. We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

**Ingredients**

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

**Directions**

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!

# Centering with margin auto

CSS

```
margin: 30px auto 0 auto;
```

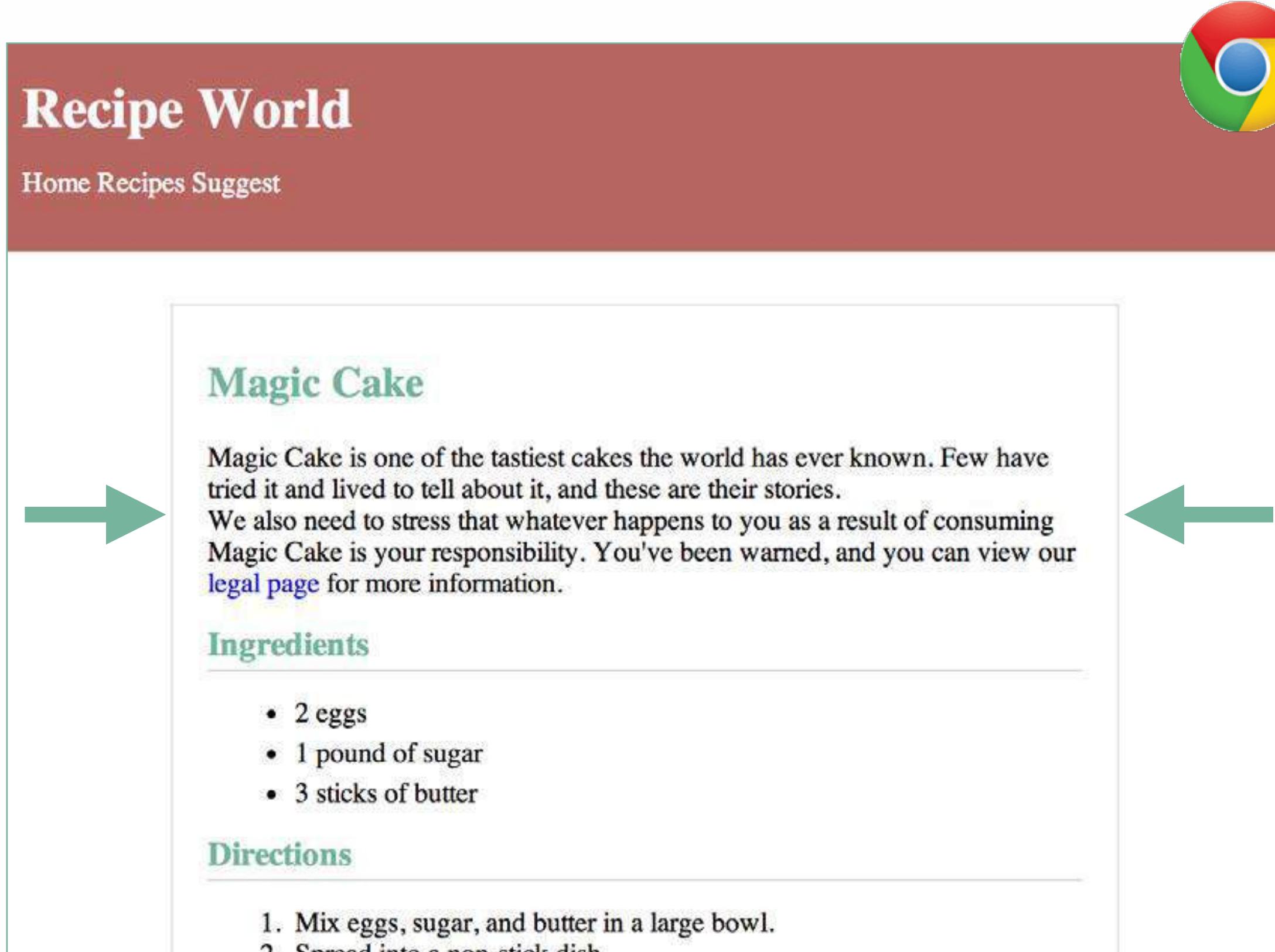


right



left

**auto** here means *take as much space as you can.*



The screenshot shows a web browser window with a red header bar containing the text "Recipe World", "Home", "Recipes", and "Suggest". The main content area has a white background and a light gray border. At the top of this area, there is a green arrow pointing to the left and a green arrow pointing to the right, both positioned above the "Magic Cake" title. The title "Magic Cake" is in bold green text. Below it, a paragraph of text discusses the cake's history and a warning about its effects. Underneath the text, there is a section titled "Ingredients" with a list of three items: "2 eggs", "1 pound of sugar", and "3 sticks of butter". Another green arrow points to the left below the "Ingredients" section. At the bottom of the content area, there is a section titled "Directions" with two numbered steps: "1. Mix eggs, sugar, and butter in a large bowl." and "2. Spread into a non-stick dish.". A green arrow points to the left below the "Directions" section. In the top right corner of the browser window, there is a small circular icon with the Google Chrome logo.



# Centering content

There's two main ways to center things, but the CSS you write depends on what you're centering.

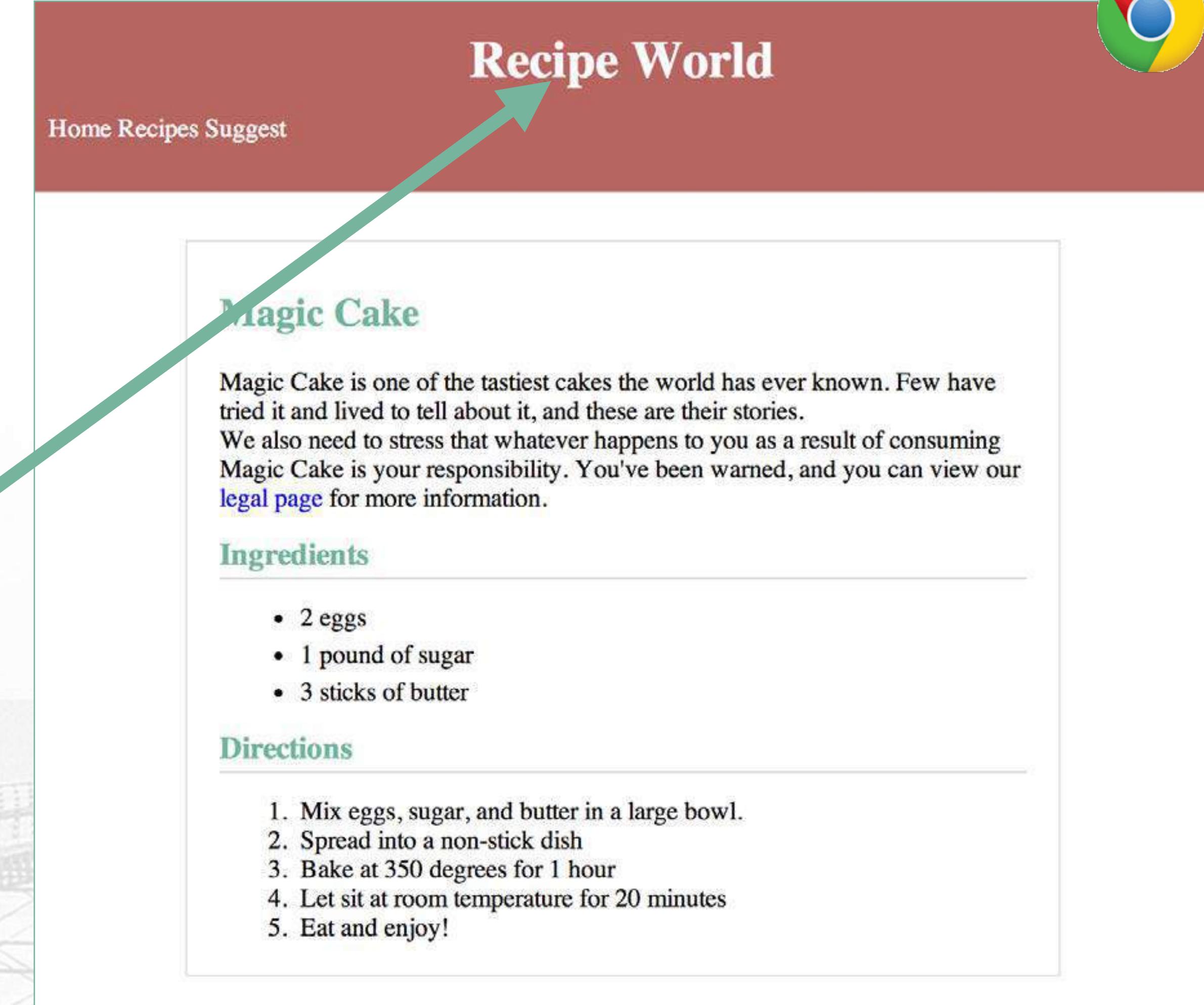
If: you want to center children  
inside a block-level tag

Then: set the **text-align** of the  
children to **center**

CSS

```
h1 {  
    color: #ffffff;  
    margin: 0 0 15px 0;  
    text-align: center;  
}
```

centered!



Home Recipes Suggest

# Recipe World

## Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories. We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

### Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

### Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!

# Centering content

There's two main ways to center things, but the CSS you write depends on what you're centering.

CSS

```
.nav {  
padding-left: 0;  
margin: 5px 0 20px 0;  
text-align: center;  
}  
  
h2 {  
margin: 10px 0 20px 0;  
text-align: center;  
}
```

centered!

Recipe World

Home Recipes Suggest



Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories. We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

## Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

## Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!

# A note about fixed widths

---

If you've been reading about or making websites, you might be wondering why we're using fixed widths instead of relative widths.

Fixed widths are great for learning how page layout works.



# LEVEL 4 IMAGES



# The three types of webpage images

---

Most images on webpages fall into one of these three categories

- Content Images
- Layout Images
- User Interface Images



# Content images

Content images are any images that are just as necessary to the page as all of the text.

Digital Cameras Best Sellers



Canon EOS Rebel T3 12.2 MP CMOS...  
★★★★★ (758)  
\$449.00 \$299.00

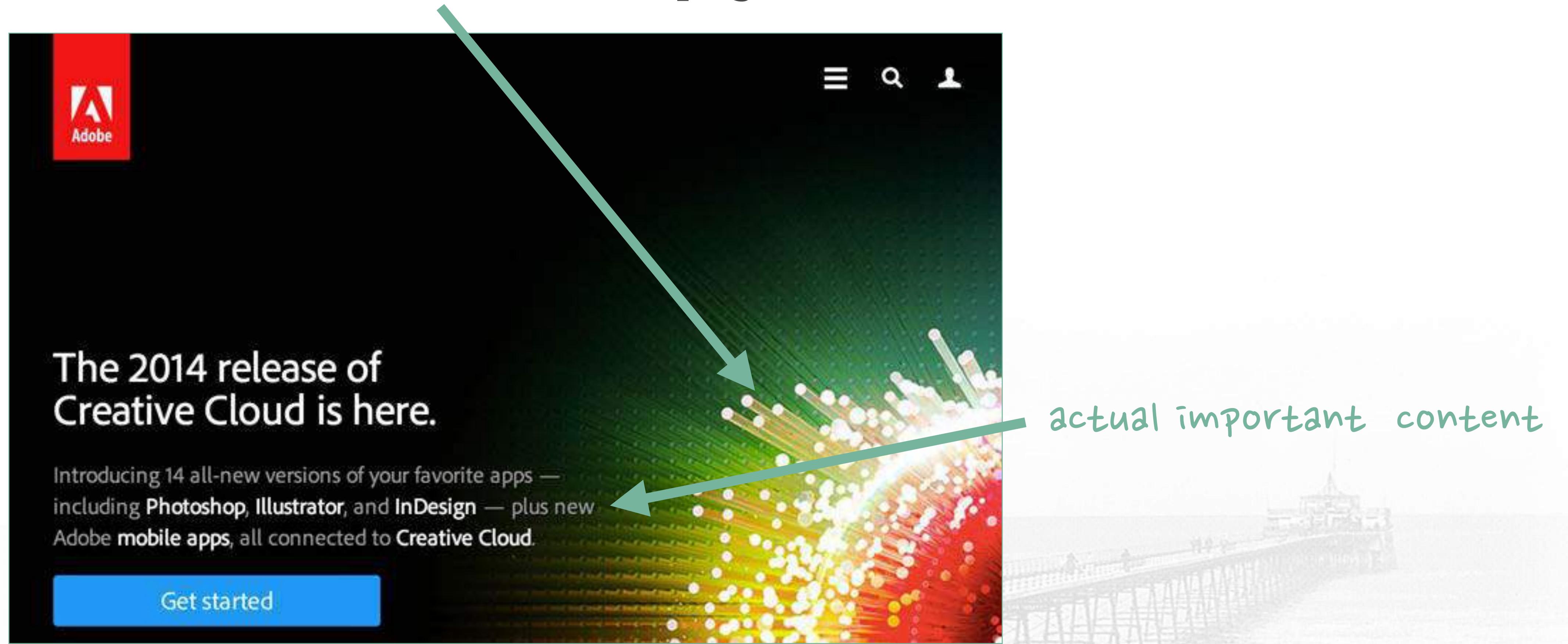
Canon PowerShot SX510 HS 12.1 MP CMOS...  
★★★★★ (319)  
\$249.00 \$199.00

[See all best sellers in Digital Cameras](#)



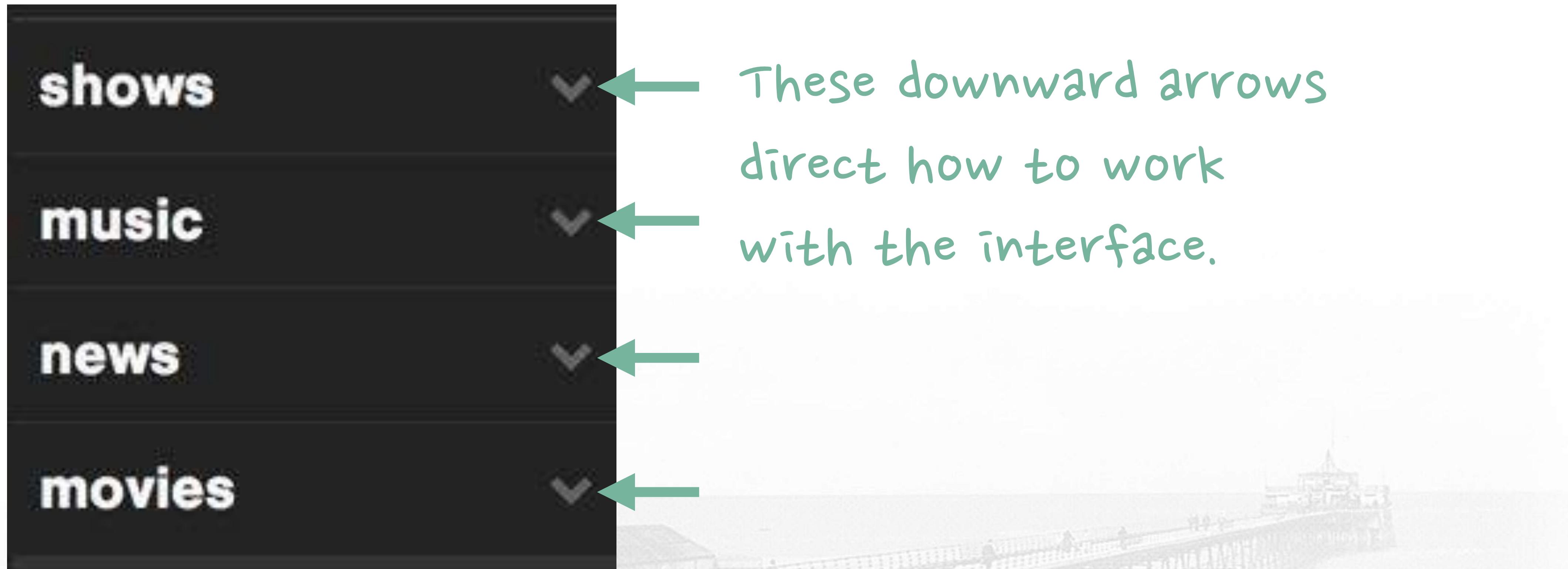
# Layout images

These are in the background, and are not necessary to understand the content of the page.



# User Interface images

These assist with the interface of the webpage, but are not required to understand the content of the page.



# Creating content images

---

Content images are created in HTML with the **<img>** tag.

```
<img>
```



<img> is an **empty tag**.

```
<img></img>
```



remember, that means it just  
has an opening tag (no closing)



# Creating content images

Content images are created in HTML with the `<img>` tag.

**magic-cake.html**

HTML

```

```

relative path

Recipe World server files



main site folder

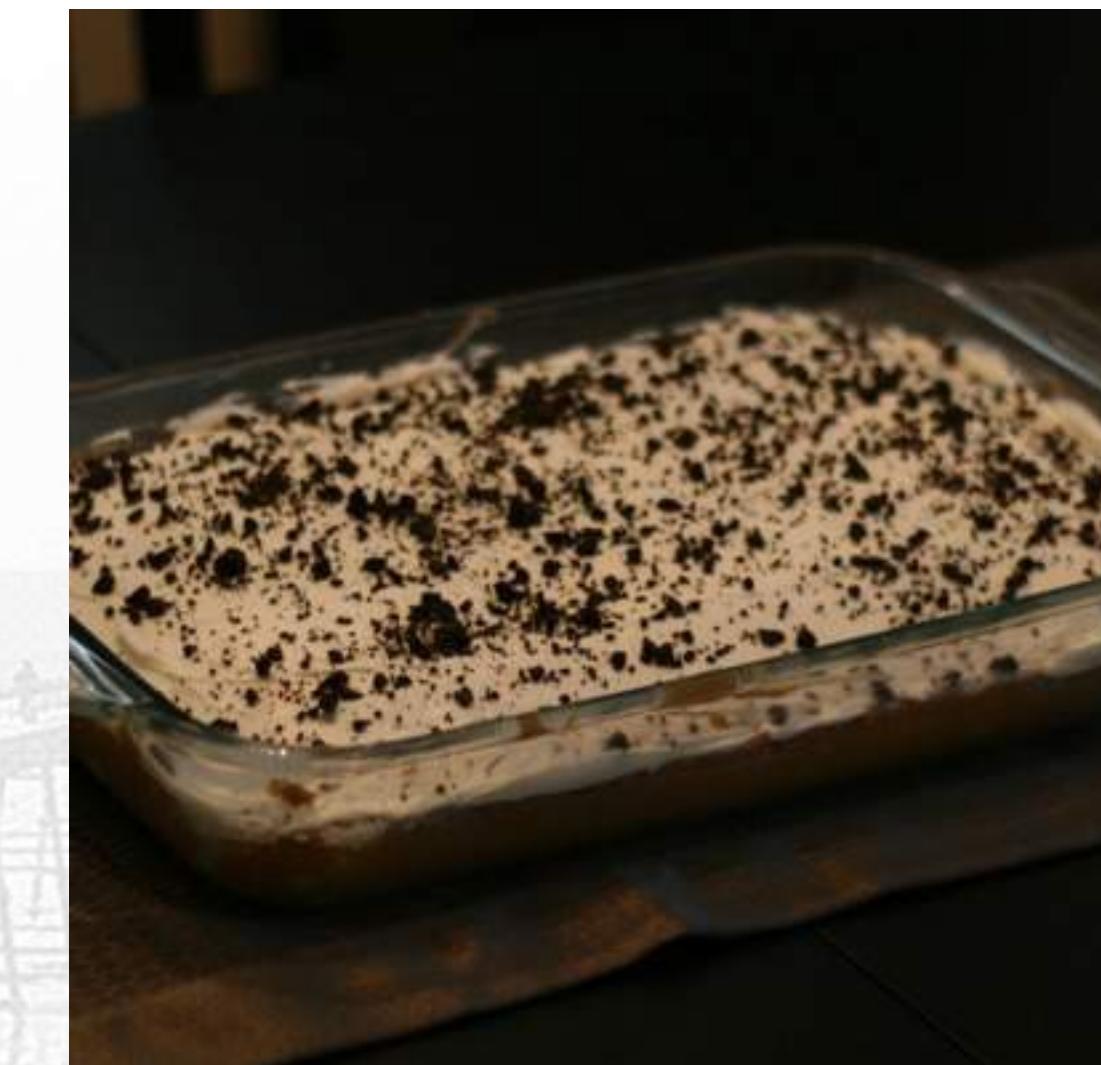


magic-cake.html



cake.png

Instead of open/close tags, add an **src attribute** to set which image is used.



# Putting images in their own folder

It makes sense to organize your images so they aren't in the main site folder.

**magic-cake.html**

**HTML**

```

```

Recipe World server files



**main site folder**



**magic-cake.html**

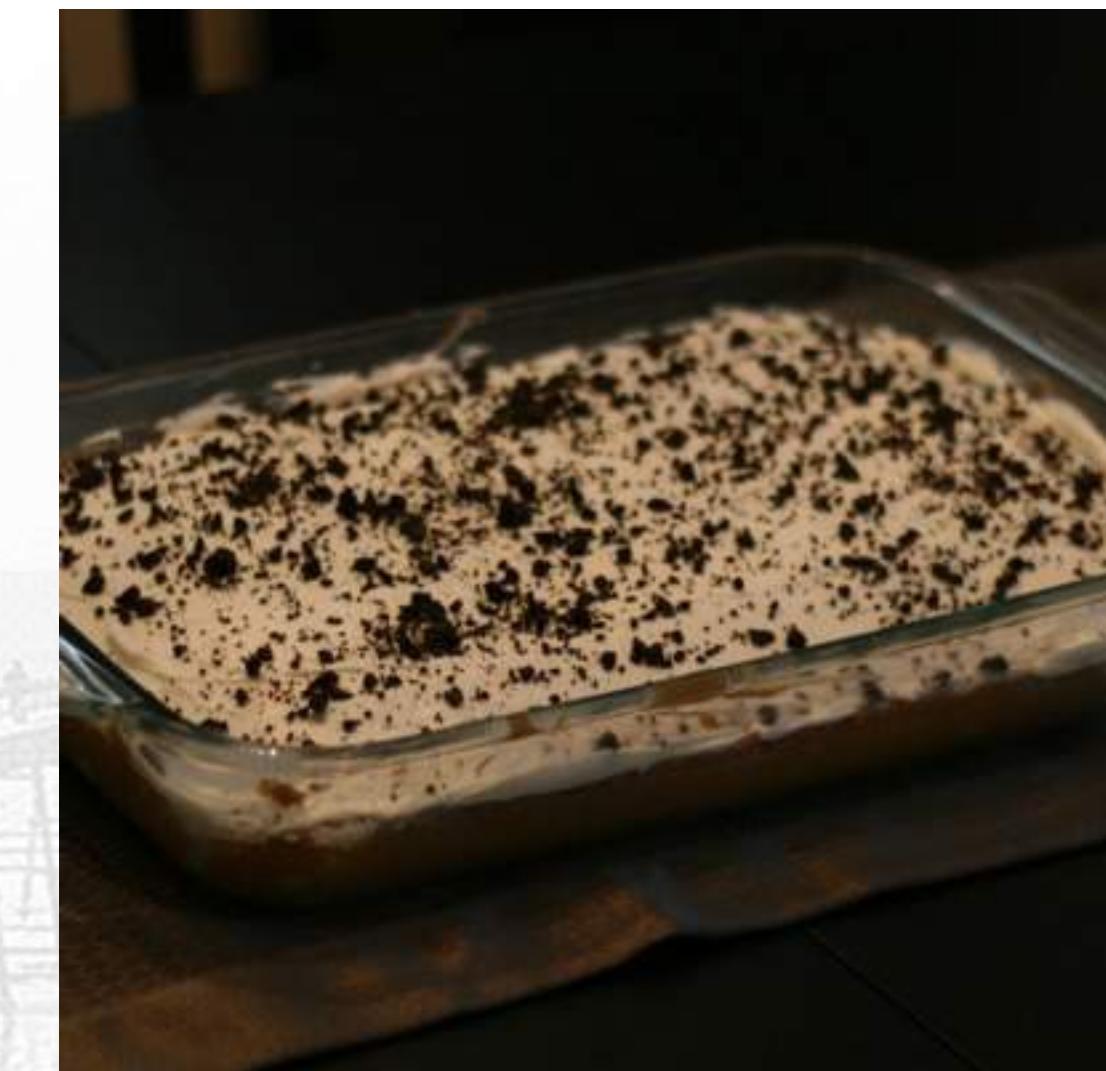


**images**



**cake.png**

now all of your  
images can go in  
the images folder



# How images are loaded

Recipe World server files



Browser requests magic-cake.html

The magic-cake HTML is loaded in the browser

```

```

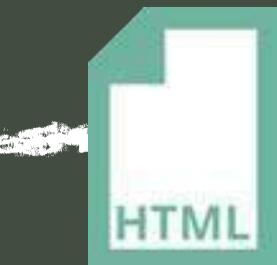
Browser finds this html

Browser requests images/cake.png

cake.png contents are returned and displayed



main site folder



magic-cake.html



images



cake.png



# Adding alt attributes to content images

---

The alt attribute should contain text that describes the image or the purpose of the image.

**magic-cake.html**

**HTML**

```

```

## Why?? Accessibility

Not everyone who views your page will be able to view the images.

Screen readers rely on alt tags to describe images to visitors that might not be able to view images.



# Laying out content images

`<img>` is an inline-level tag, but they are often put inside block-level tags.

This is similar to `<a>` tags, which almost always appear as children of block-level tags.

`magic-cake.html`

HTML

```
<h3>Your Photos</h3>
<ul class="photos">
  <li></li>
  <li></li>
  <li></li>
</ul>
```

`<li>` is a block-level tag

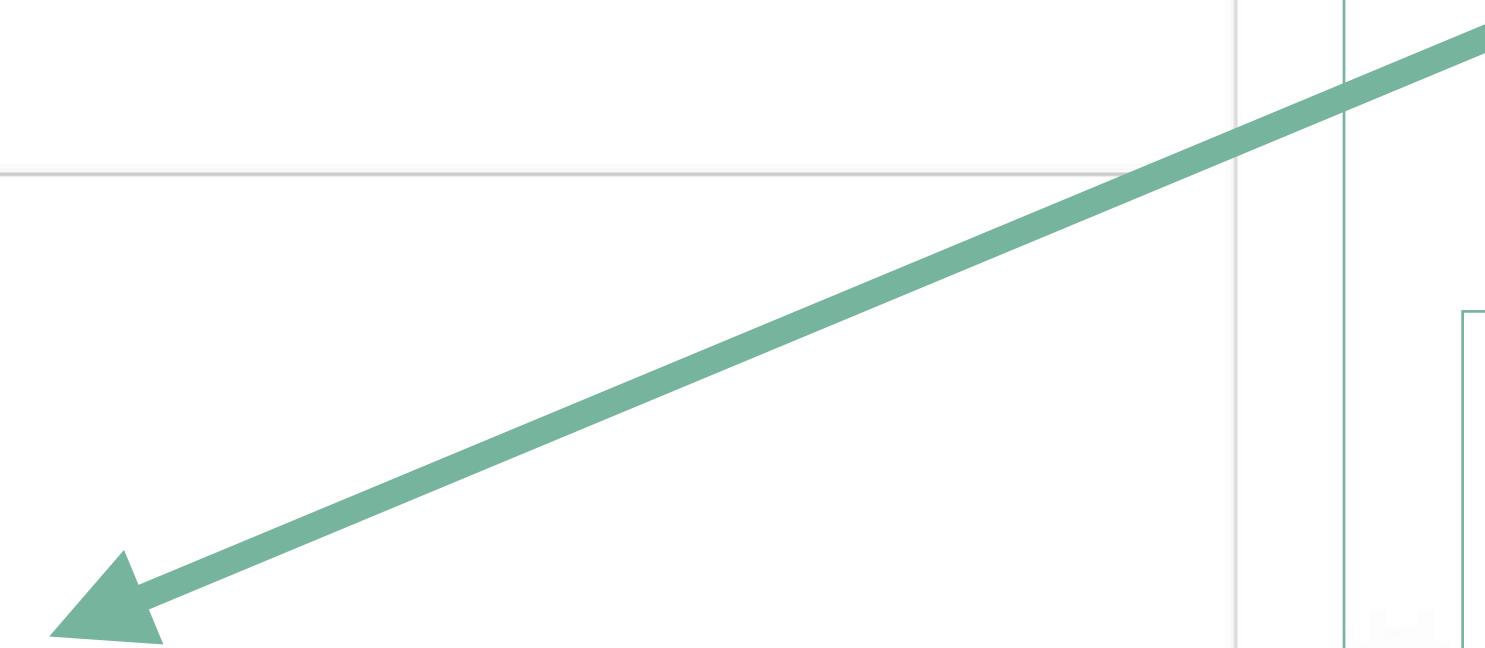


# Laying out content images

## Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!

## Your Photos



## Problem

We want these photos to display side-by-side, like this

## Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!

## Your Photos



# Laying out content images

magic-cake.html

HTML

```
<h3>Your Photos</h3>
<ul class="photos">
  <li><img...>
</ul>
```



CSS

```
.photos {
  list-style-type: none;
  padding: 0;
}
.photos li {
  display: inline;
  padding-left: 11px;
}
```

remove bullets  
and padding

make the images show  
horizontally with a  
little space in between

## Directions

1. Mix eggs, sugar, and butter in a large bowl.
2. Spread into a non-stick dish
3. Bake at 350 degrees for 1 hour
4. Let sit at room temperature for 20 minutes
5. Eat and enjoy!

## Your Photos



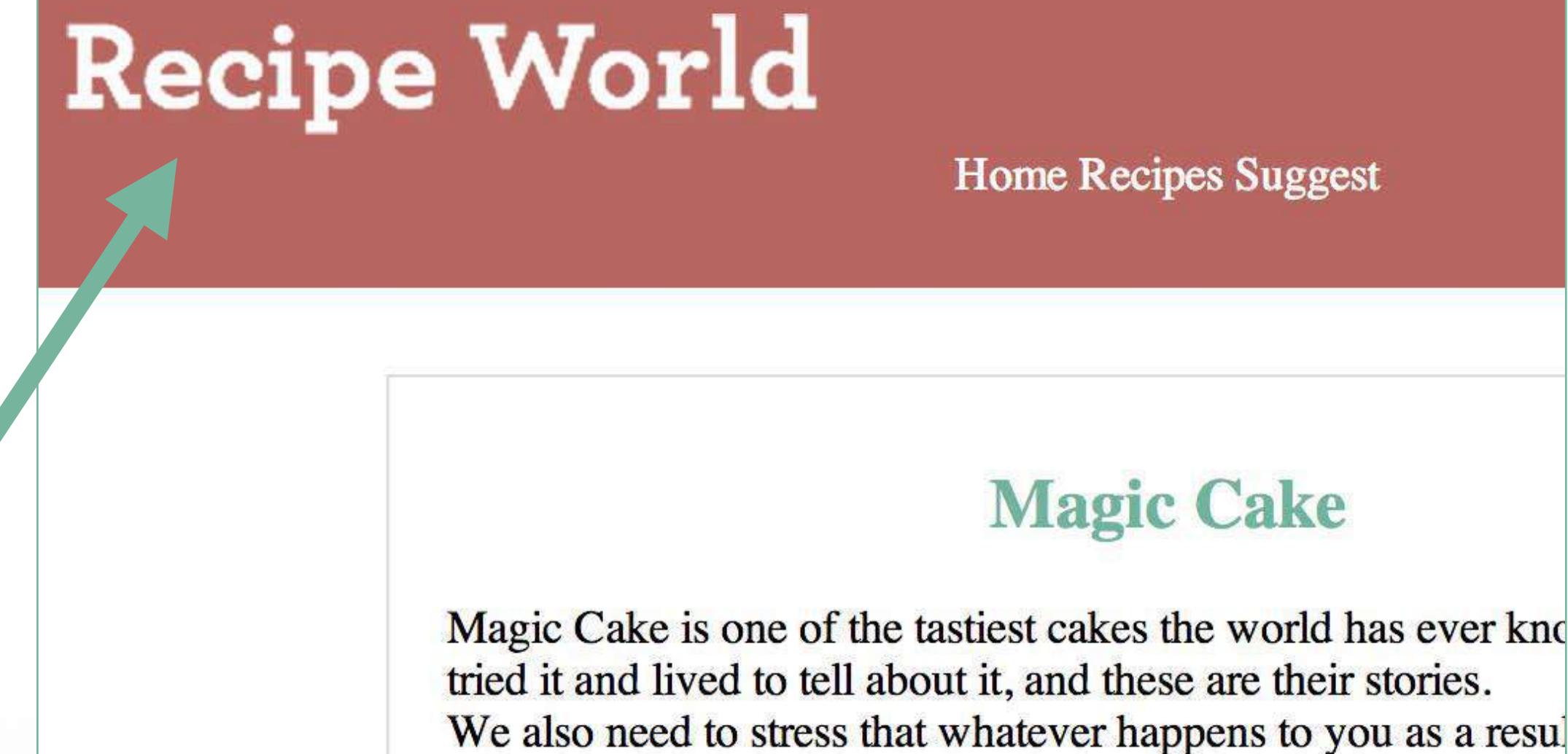
# Using an image as a site/logo mark

We can replace the `<h1>` with an `<img>` tag.

`magic-cake.html`

```
<div class="header">  
  <h1>Recipe World</h1>  
  <ul class="nav">...
```

HTML



```
<div class="header">  
    
  <ul class="nav">...
```

HTML



# Centering an image inside a block-level tag

<img> is an inline-level tag so it can't be centered with text-align.

Instead, set just that image to **display: block** and center it with the **margin: auto** approach.

## magic-cake.html

HTML

```
<div class="header">
  
  <ul class="nav">...
```

CSS

```
.header img {
  display: block;
  margin: 0 auto 0 auto;
}
```

Recipe World

Home Recipes Suggest

Magic Cake



Magic Cake is one of the tastiest cakes the world has ever known. Few have



# Creating layout images

Layout images are created in CSS with the **background** property.

The image shows a website layout. At the top is a red header bar containing the title "Recipe World" in large white font, and "Home Recipes Suggest" in smaller white font below it. The main content area has a white background with a faint gray grid pattern. The title "Magic Cake" is centered in large green font. Below the title is a paragraph of text: "Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories. We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information." Underneath this is a section titled "Ingredients" in green font, followed by a horizontal line and a bulleted list of ingredients: "• 2 eggs", "• 1 pound of sugar", and "• 3 sticks of butter". In the bottom right corner of the slide, there is a small circular icon with a double-headed arrow symbol.

## Recipe World

Home Recipes Suggest

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories. We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

#### Ingredients

- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

# Understanding the background property

There's several different properties that relate to background images.

**background-color** works just like the color property.

CSS

```
body {  
    background-color: #5f5f5f;  
}
```

## Recipe World

Home Recipes Suggest

### TIP

You should always set a background color in case the background image fails to load.

### Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories. We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

#### Ingredients

- 2 eggs

# Understanding the background property

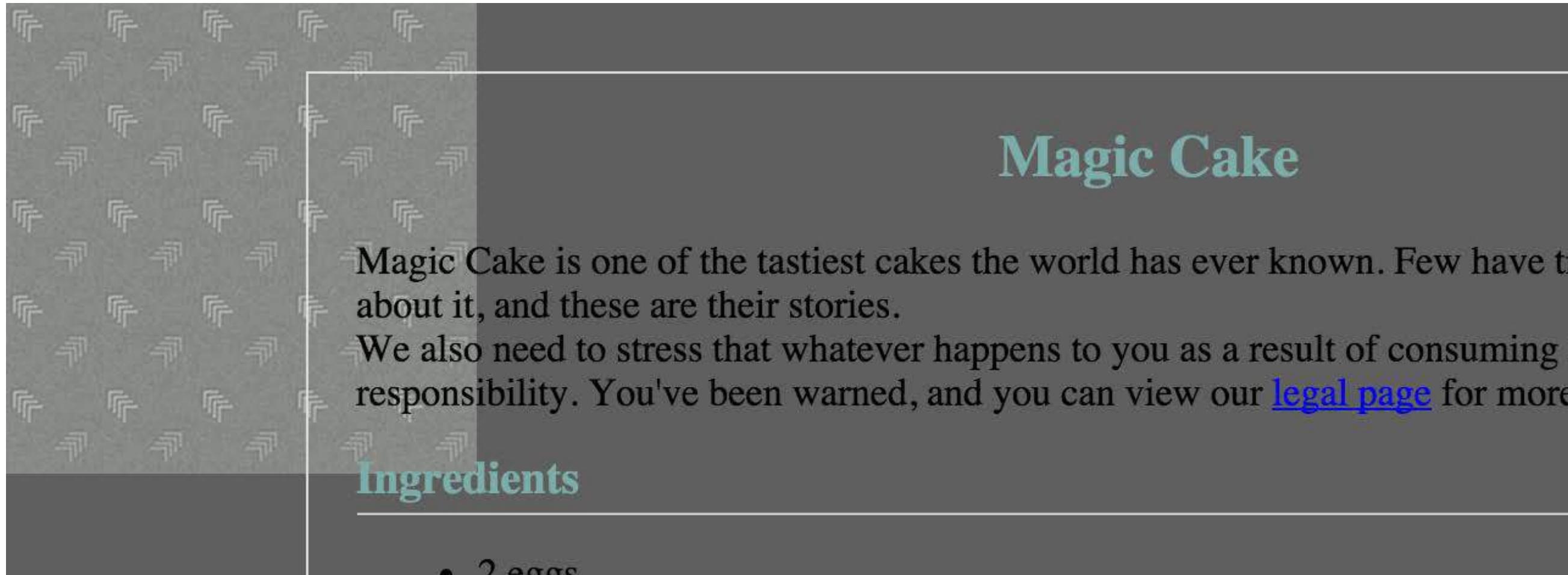
**background-image** can point to a relative or absolute path.

CSS

```
body {  
    background-color: #5f5f5f;  
    background-image: url( images/gobbler.png );  
}
```

the path has to be inside this url() code

notice that quotes  
don't need to be  
around the path



Check out [subtlepatterns.com](http://subtlepatterns.com)  
for this and other patterns!



# Understanding the background property

**background-position** can be used to position the image in a container.

CSS

```
body {  
    background-color: #5f5f5f;  
    background-image: url(images/gobbler.png);  
    background-position: top left;  
}
```



the first word can  
be **top**, **center**, or  
**bottom**

the second word  
can be **left**, **center**,  
or **right**



# Understanding the background property

**background-position** can be used to position the image in a container.

CSS

```
body {  
    background-color: #5f5f5f;  
    background-image: url(images/gobbler.png);  
    background-position: center right;  
}
```

this shows the image vertically  
centered and pushed to the right

## Magic Cake

iest cakes the world has ever known. Few have tried it and lived to tell stories.  
whatever happens to you as a result of consuming Magic Cake is your own business, and you can view our [legal page](#) for more information.

butter in a large bowl  
sk dish  
for 1 hour



# Understanding the background property

**background-repeat** can be used to repeat tiled images.

CSS

```
body {  
    background-color: #5f5f5f;  
    background-image: url(images/gobbler.png);  
    background-position: top left;  
    background-repeat: repeat;  
}
```

## Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

### Ingredients

- 2 eggs
- 1 pound sugar



# Understanding the background property

**background-repeat** can be used to repeat tiled images.

CSS

```
body {  
    background-color: #5f5f5f;  
    background-image: url(images/gobbler.png);  
    background-position: top left;  
    background-repeat: repeat-y;  
}
```



## Other values

- |           |                               |
|-----------|-------------------------------|
| repeat-x  | tile the image horizontally   |
| repeat-y  | tile the image vertically     |
| no-repeat | don't tile or repeat anything |



# Understanding the background property

All four of those can be shortened into a single **background** property.

```
body {  
    background-color: #5f5f5f;  
    background-image: url(images/gobbler.png);  
    background-position: top left;  
    background-repeat: repeat;  
}
```

CSS

```
body {  
    background: #5f5f5f url(images/gobbler.png) top left repeat;  
}
```

CSS

color

image

position

repeat



# Creating different tiled backgrounds

We can set different backgrounds for different containers.

Here, we're using one repeated background for the body, another for the main-content, and another for the divs on the recipes page.

CSS

```
body {  
  background: #000000 url(images/stardust.png) top left repeat;  
}  
.main-content {  
  background: #ffffff url(images/crossword.png) top left repeat;  
}  
.recipes li {  
  background: #ffffff url(images/li-bg.png) top left repeat;  
}
```

# Result of using three different layered backgrounds

The screenshot illustrates a website layout with three distinct background layers:

- body background**: The top layer is a solid reddish-brown color.
- .main-content background**: The middle layer is a dark gray color.
- .recipes li background**: The bottom layer is a white background with a light gray diagonal hatching pattern.

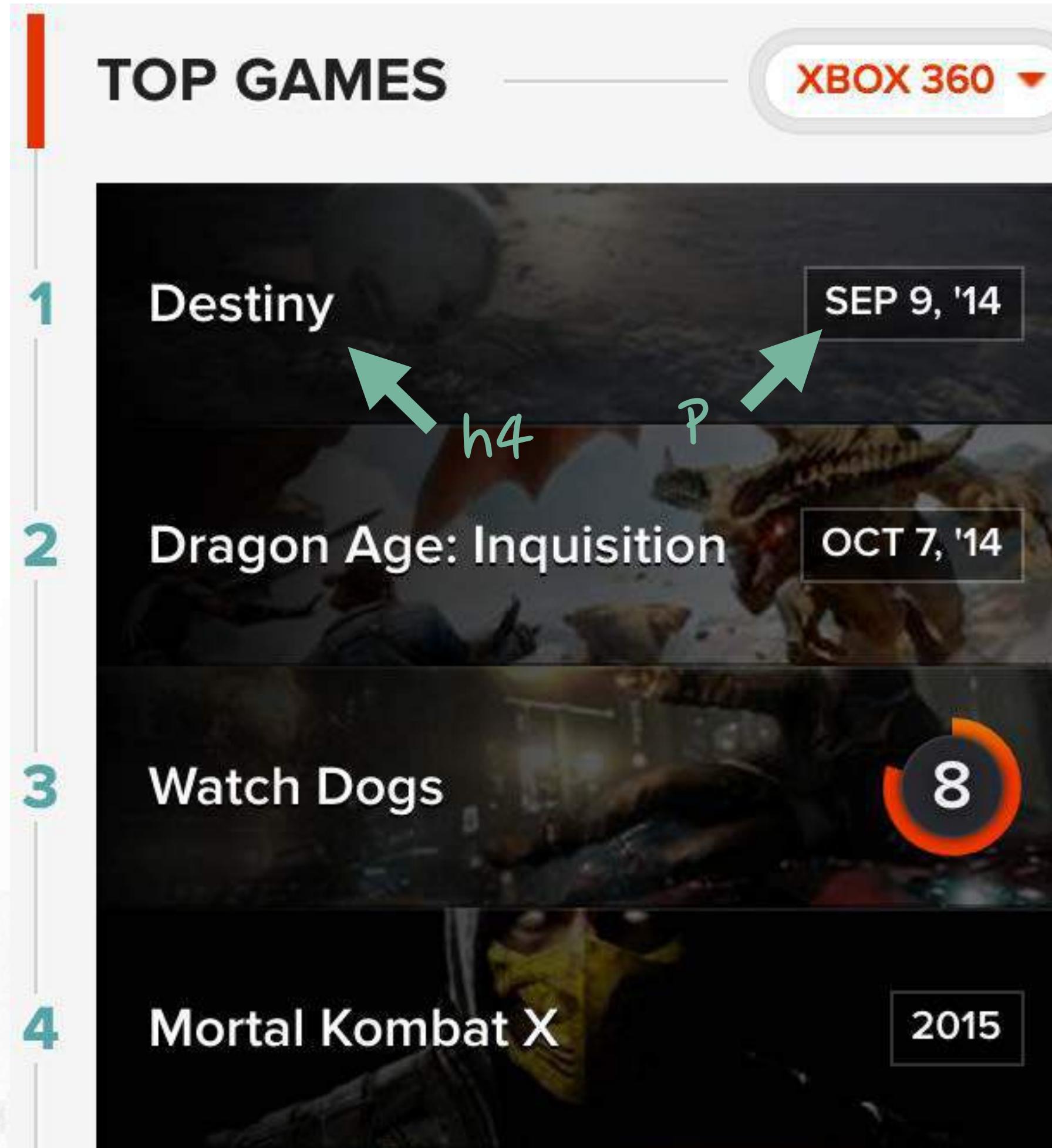
The website title "Recipe World" is displayed in large white font on the reddish-brown background. Below the title, there are navigation links: "Home", "Recipes", and "Suggest". The main content area has a heading "Recipes" in green. Three recipe cards are listed:

- Magic Cake**: Description: "This is a dangerously delicious cake."
- Tacosaurus Mex**: Description: "It's always taco night somewhere!"
- Florida Omelette**: Description: "Omelette you in on a little secret - this is one tasty egg."

Green arrows point from the labels to their respective background layers. A small circular icon with a double arrow symbol is located in the bottom right corner.

# Writing on top of background images

Some sites like to write text over images.



This list on [gamespot.com](http://gamespot.com) has text for the game name and release date appearing over a background image from each game.



# Writing on top of background images

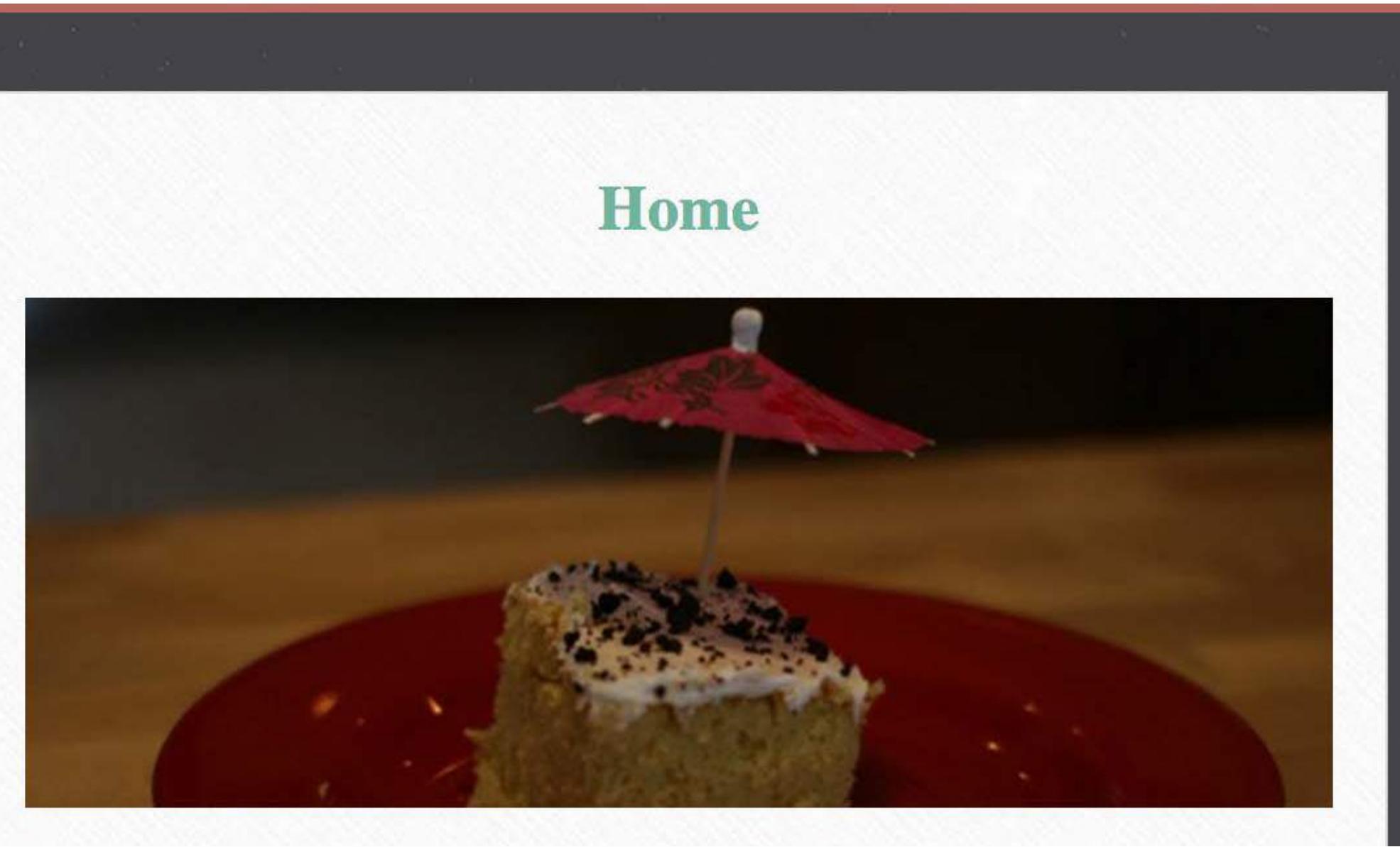
Some sites like to write text over images.

One way to do this is to create an empty **div** container that completely contains a background image, and put text tags inside of it.

home.html

HTML

```
<div class="main-content">
  <h2>Home</h2>
  <div class="featured-image">
  </div>
</div>
```

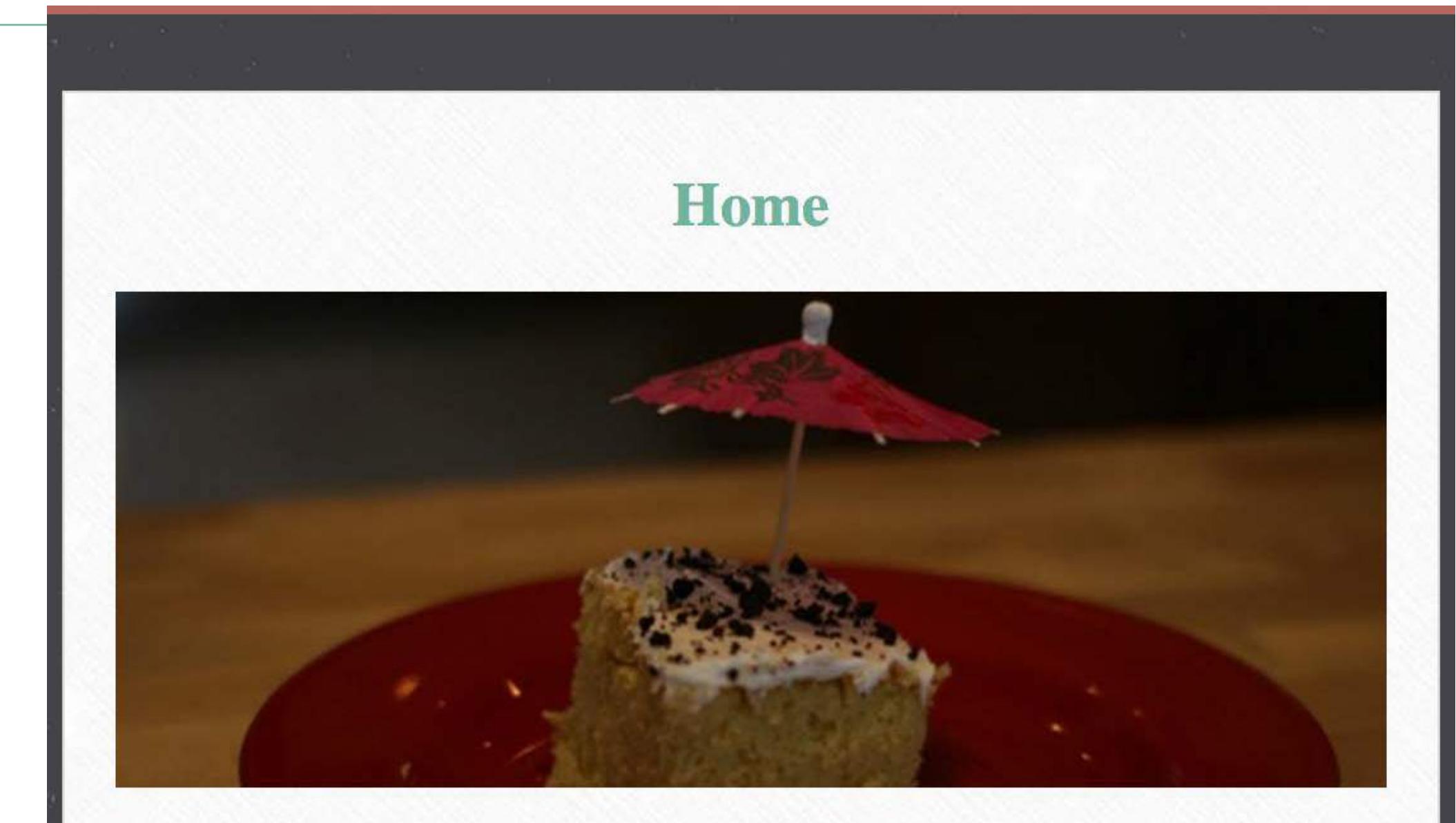


# First, add the background image to the container

home.html

```
<div class="main-content">
  <h2>Home</h2>
  <div class="featured-image">
  </div>
</div>
```

HTML



```
.featured-image {
  width: 630px;
  height: 246px;
  background: #ffffff url(images/featured-cake.png)
              top left no-repeat;
}
```

CSS

these should be equal to the  
width/height of the image



# Then, add a tag that shows text as a child of that container

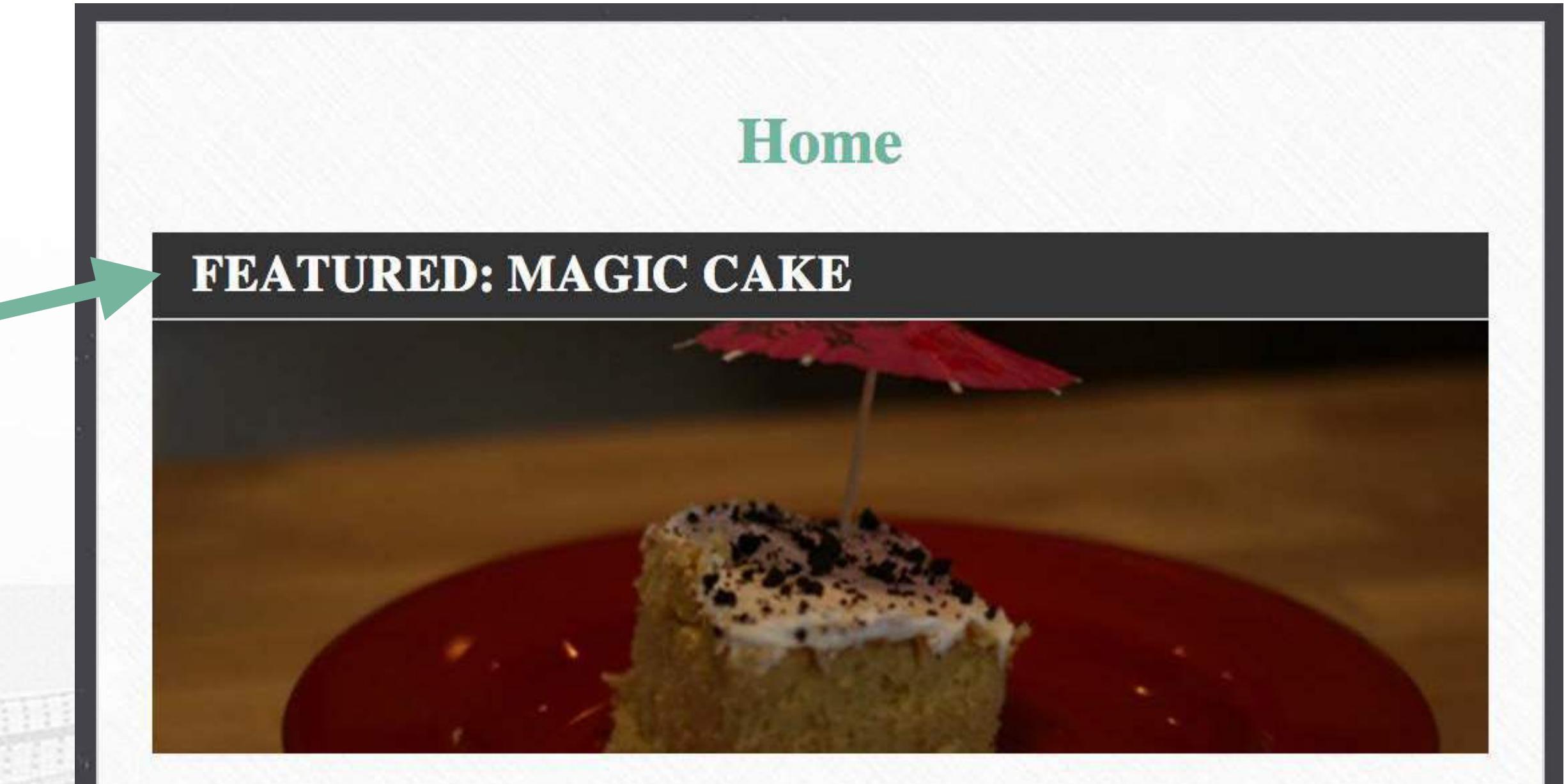
home.html

```
<div class="featured-image">  
  <h3>Featured: Magic Cake</h3>  
</div>
```

HTML

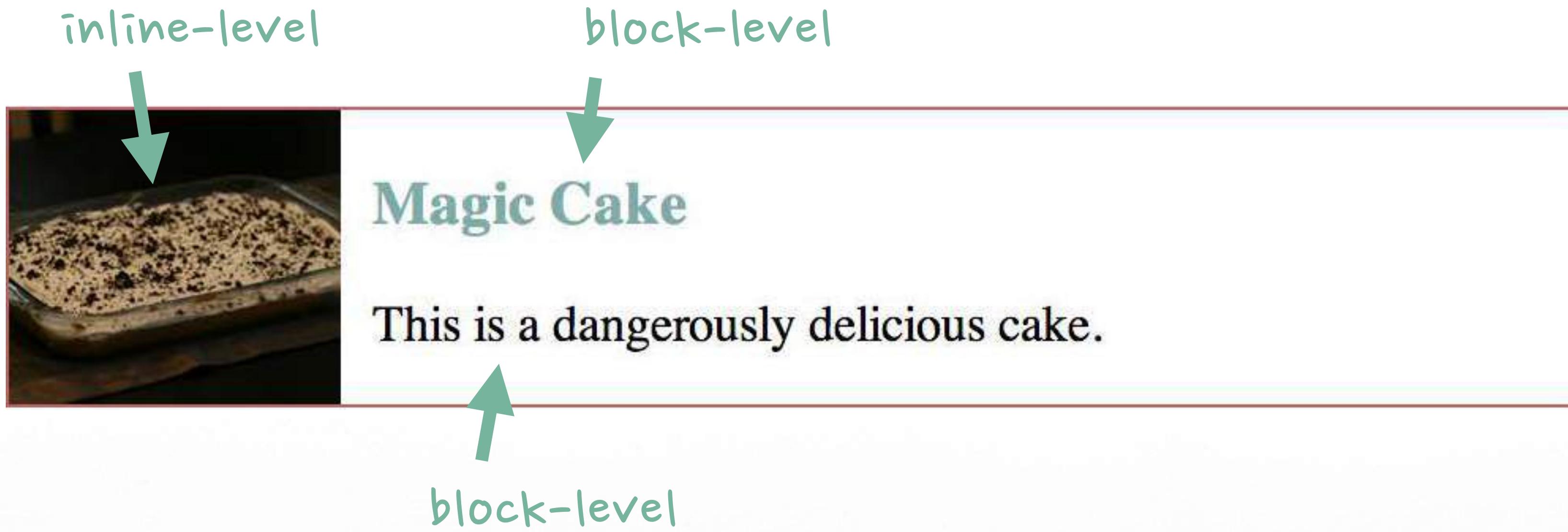
```
.featured-image h3 {  
  margin: 0;  
  background-color: #333333;  
  color: #ffffff;  
  padding: 5px 0 5px 15px;  
  text-transform: uppercase;  
}
```

CSS



# Displaying an image to the side of block-level tags

We know that tags are either block or inline.



This is an image to left of multiple block-level tags.

How can we make this?



# Floating images

You might try setting two divs to inline.

left div set to inline



right div set to inline



**Magic Cake**

This is a dangerously delicious cake.



# Floating images

```
<ul class="recipes">
  <li>
    <div>
      
    </div>
    <div>
      <h3><a href="...">Magic Cake</a></h3>
      <p>...</p>
    </div>
  </li>
</ul>
```

HTML



magic-cake.html

Setting these two divs side-by-side with **display:inline** seems like it would work.

```
.recipes div {
  display: inline;
}
```

CSS



# Floating images

Turns out, we can't just flip things to inline and then put block-level tags inside of them.

Time to try something else.

left div set  
to inline →



right div set  
to inline →

Magic Cake

This is a dangerously delicious cake.



# Floating images

Put the `img`, `h3`, and `p` all in one `li` and **float** the `img` left.

`magic-cake.html`

```
<ul class="recipes">
  <li>
    
    <h3><a href="...">Magic Cake</a></h3>
    <p>...</p>
  </li>
</ul>
```

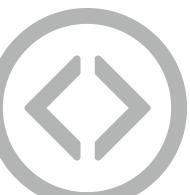
HTML



CSS

```
.recipes img {
  float: left;
}
```

This makes the image take up some of the left space that the **h3** and **p** boxes would normally eat up as block-level tags.



# Floating images

Success! We've also added some right padding so the image doesn't touch right up against the text.

CSS

```
.recipes img {  
  float: left;  
  padding-right: 10px;  
}
```



## Magic Cake

This is a dangerously delicious cake.



# LEVEL 5

# FONTS AND FORMS



# Fonts on the Web

Fonts and font styles can be controlled with CSS.

The screenshot shows a web page with the following elements annotated:

- bold, large** → **Learn By Doing**: A large, bold heading.
- regular, medium** → **No setup. No hassle. Just learning.**: A sentence describing the service.
- uppercase** → **VIEW OUR COURSES**: A button with uppercase text.
- regular, normal** → **Code School teaches web technologies in the comfort of your browser with video lessons, coding challenges, and screencasts.**: A descriptive paragraph.
- smaller** → **Learn where to start!**: A link with smaller text.
- uppercase** → **LEARN: GIT BACKBONE.JS SASS RAILS JQUERY**: A row of buttons with uppercase text.



# Our current page's fonts

Since we haven't set a font, the browser is just picking a default.

The screenshot shows a web page with a dark gray header and footer. The main content area has a light gray background with a subtle texture. At the top, the title "Magic Cake" is displayed in a large, bold, teal-colored font. Below the title, there is a paragraph of text in a black sans-serif font. Further down, there is another paragraph followed by a section titled "Ingredients" with a bulleted list of ingredients. Finally, there is a section titled "Directions" with a numbered list of steps. The overall layout is clean and modern, despite the lack of a specified font.

## Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

### Ingredients

- 2 eggs
- 1 pound sugar
- 3 sticks of butter

### Directions

- 1 Mix eggs, sugar, and butter in a large bowl.



# Changing the font with CSS

The **font-family** property is used to set the ideal font and a few fallback options.

CSS

```
body {  
  font-family: Helvetica, Arial, sans-serif;  
}
```



The ideal font

The browser will  
try to load the  
ideal font first.



# Changing the font with CSS

The fallback fonts are used if the ideal font isn't available.

```
body {  
  font-family: Helvetica, Arial, "Times New Roman", sans-serif;  
}
```

use quotes if fonts are multi-word

CSS

Fallback fonts

Not all fonts are installed in all browsers!!!



# Our page after changing the font

Here's magic-cake.html after changing the main font to Helvetica.

The screenshot shows a web page titled "Magic Cake". The title is centered at the top in a large, bold, teal-colored font. Below the title is a paragraph of text in a smaller, black sans-serif font. Further down is another paragraph, also in black font. A section titled "Ingredients" is followed by a bulleted list of three items. A section titled "Directions" is partially visible at the bottom. The entire page is framed by a dark gray border.

## Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

### Ingredients

- 2 eggs
- 1 pound sugar
- 3 sticks of butter

### Directions

1 Mix eggs, sugar, and butter in a large bowl



# Browser defaults - the font edition

Just like with the box model properties, the browser has default font styles - and you should reset these too!!

These weights  
and sizes are  
the browser  
defaults.

**Magic Cake**

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

**Ingredients**

- 2 eggs
- 1 pound sugar
- 3 sticks of butter

**Directions**

1. Mix eggs, sugar, and butter in a large bowl.

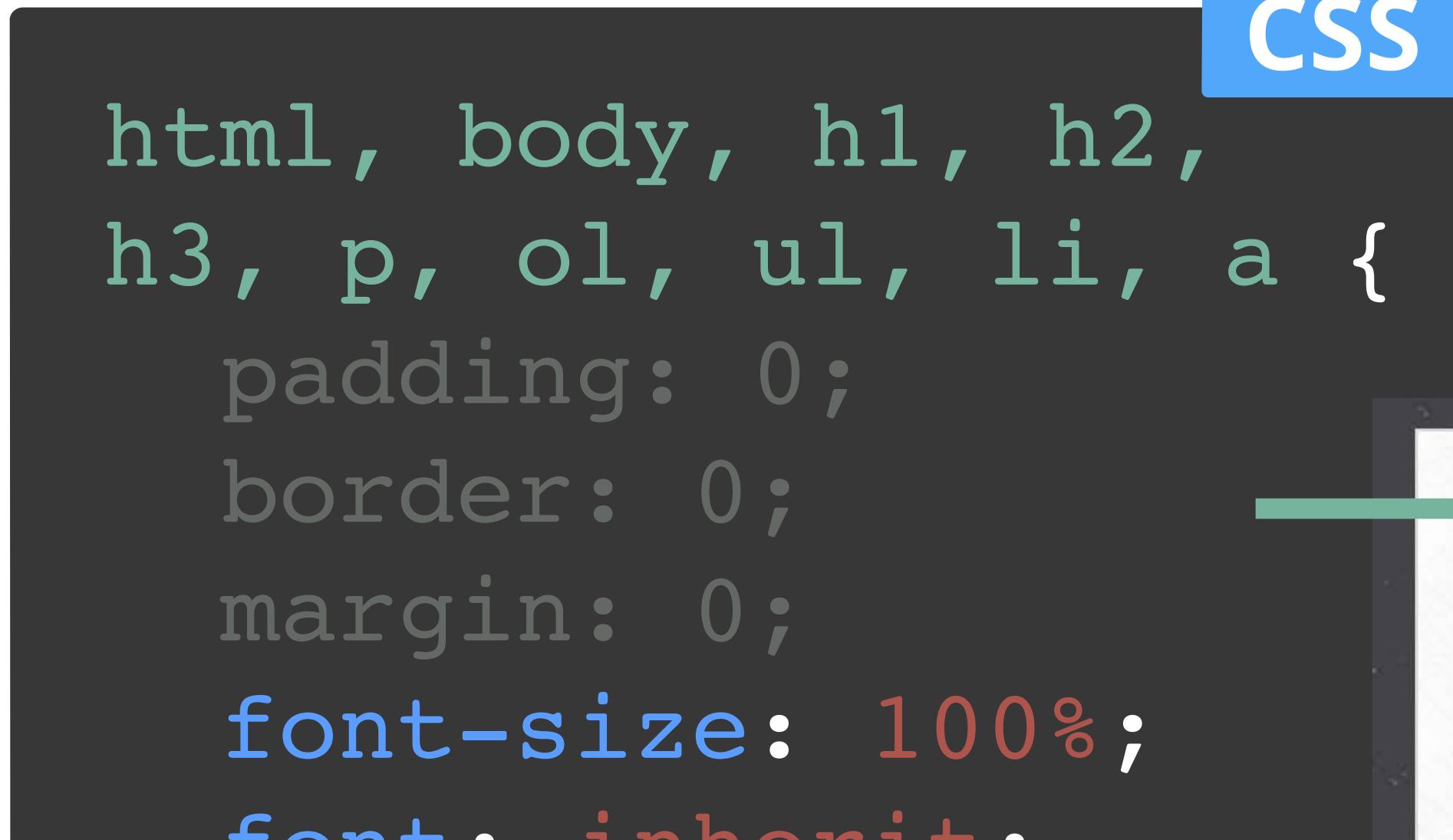


# Resetting default browser font styles

Now the font weights and sizes won't follow an arbitrary default.

CSS

```
html, body, h1, h2,  
h3, p, ol, ul, li, a {  
  padding: 0;  
  border: 0;  
  margin: 0;  
  font-size: 100%;  
  font: inherit;  
}
```



Magic Cake

Magic Cake is one of the tastiest cakes the world has ever known. Few have tried it and lived to tell about it, and these are their stories.

We also need to stress that whatever happens to you as a result of consuming Magic Cake is your responsibility. You've been warned, and you can view our [legal page](#) for more information.

## Ingredients

- 2 eggs
- 1 pound sugar
- 3 sticks of butter

## Directions

- 1 Mix eggs, sugar, and butter in a large bowl

# Changing font sizes

The size can be set with the **font-size** property.

CSS

```
h2 {  
    color: #7facaa;  
    margin: 0 0 20px 0;  
    text-align: center;  
    font-size: 26px;  
}
```



Magic Cake  
the tastiest cakes the world has ever known. Fe  
nd these are their stories.

## Pixels aren't the only size options

percentages

ems

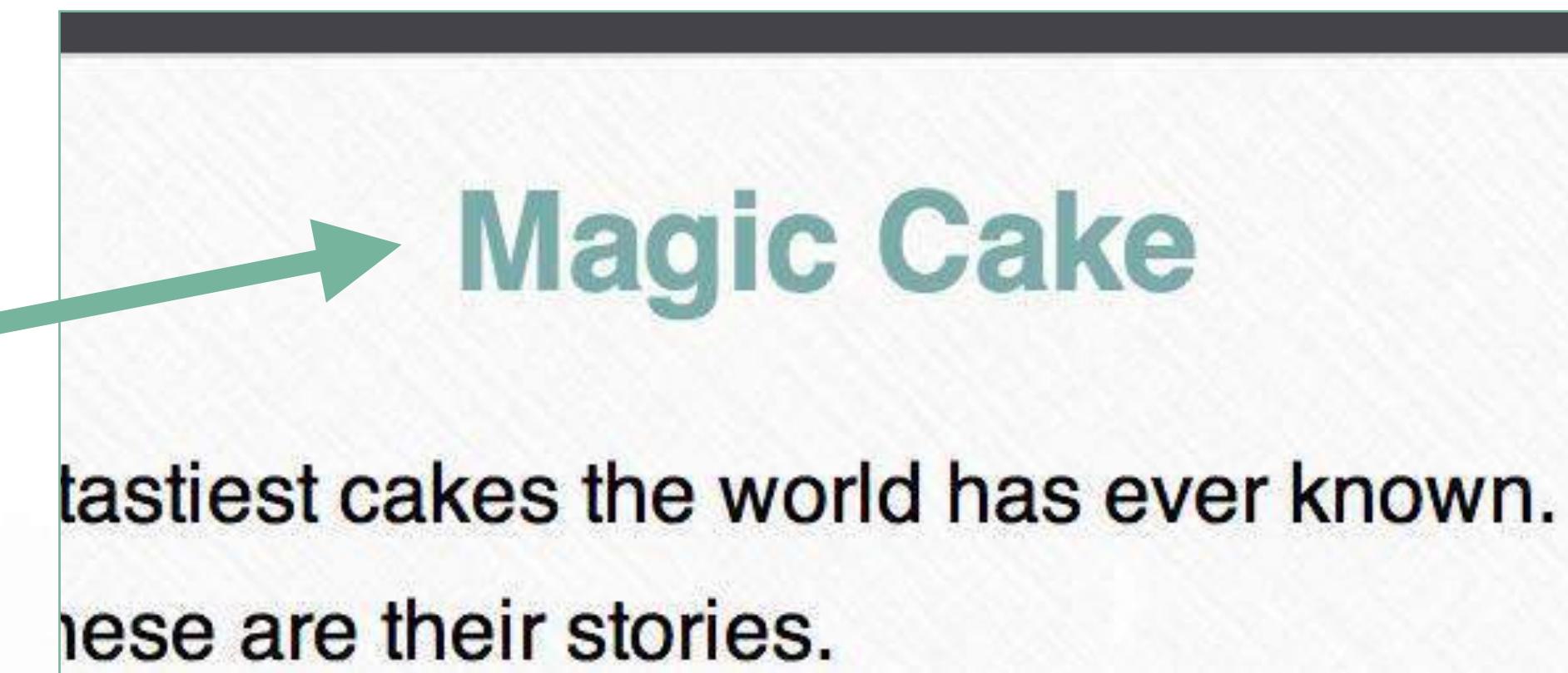


# Changing font thickness

The thickness can be set with the **font-weight** property.

CSS

```
h2 {  
    color: #7facaa;  
    margin: 0 0 20px 0;  
    text-align: center;  
    font-size: 26px;  
    font-weight: bold;  
}
```



## Weight values

normal

lighter

bold

bolder

100, 200... 900



# Changing other font properties

Here's a few other font properties:

CSS

```
.footer p {  
    color: #aaaaaa;  
    text-align: center;  
    font-weight: bold;  
    font-size: 12px;  
    font-style: italic;  
    text-transform: uppercase;  
}
```

BEFORE

© 2014 - Recipe World

AFTER

© 2014 - RECIPE WORLD

HTML

```
<div class="footer">  
    <p>&copy; 2014 - Recipe World</p>  
</div>
```

snazzy way to show a copyright symbol



# Adjusting the line height

**line-height** is like margin for each line in a box.

You can adjust the vertical spacing between different text tags by changing the margin.

margin

Margin can't adjust the spacing between lines in a single box.

Magic Cake is one of the tastiest cakes the world has ever seen. We've all lived to tell about it, and these are their stories.

We also need to stress that whatever happens, it's your responsibility. You've been warned, and you've been given all the information.



# Adjusting the line height

line-height is like margin for each line in a box.

CSS

```
.main-content p {  
  line-height: 16px;  
}
```

Magic Cake is one of the tastiest cakes  
lived to tell about it, and these are their

We also need to stress that whatever has  
is your responsibility. You've been warned.  
information.

CSS

```
.main-content p {  
  line-height: 26px;  
}
```

Magic Cake is one of the tastiest cakes  
lived to tell about it, and these are their

We also need to stress that whatever has  
is your responsibility. You've been warned.  
information.



# Web Forms

Forms are a way for a web page to get input from a user.

The diagram illustrates a "Sign In" form with the following components and labels:

- form**: Points to the overall form structure.
- label**: Points to the label "EMAIL OR USERNAME" positioned above the first input field.
- input**: Points to the text input field for entering an email or username.
- submit**: Points to the large blue "SIGN IN" button at the bottom of the form.

Other visible elements include:

- "Create Free Account" link in the top right corner.
- "PASSWORD" label and input field below the first one.
- "Forgot Your Password?" link below the password input.
- "Sign in with:" section at the bottom featuring social media icons for GitHub, Facebook, and Google.

Forms usually contain things like labels, inputs of several different types, text areas, and a submit button.



# Recipe World's form

## Disclaimer

Making forms actually process user input requires server-side code.

We can still design how the form will look in the browser now.

form

The diagram illustrates a web form titled "Suggest a recipe". It features several input fields: "Recipe Name" (text input), "Ingredients" (text area), "Your Name" (text input), "Your Email" (text input), and a checkbox for "Get Newsletter?". A submit button at the bottom is labeled "Submit the recipe". Green arrows point from the labels "label", "textarea", "input", and "submit" to their corresponding HTML elements in the form. A large green arrow points from the word "form" to the entire structure.

```
<form>
  <label>Recipe Name</label>
  <input type="text">
  <label>Ingredients</label>
  <textarea></textarea>
  <label>Your Name</label>
  <input type="text">
  <label>Your Email</label>
  <input type="text">
  <label>Get Newsletter?</label>
  <input type="checkbox">
  <input type="button" value="Submit the recipe">
</form>
```

# The HTML for a simple form

This HTML will produce the form shown below (*but some extra attributes would need to send the form data to the server*).

**suggest.html**

HTML

```
<form>
  <label>Recipe Name</label>
  <input type="text">
  <input type="submit" value="Click to Submit">
</form>
```

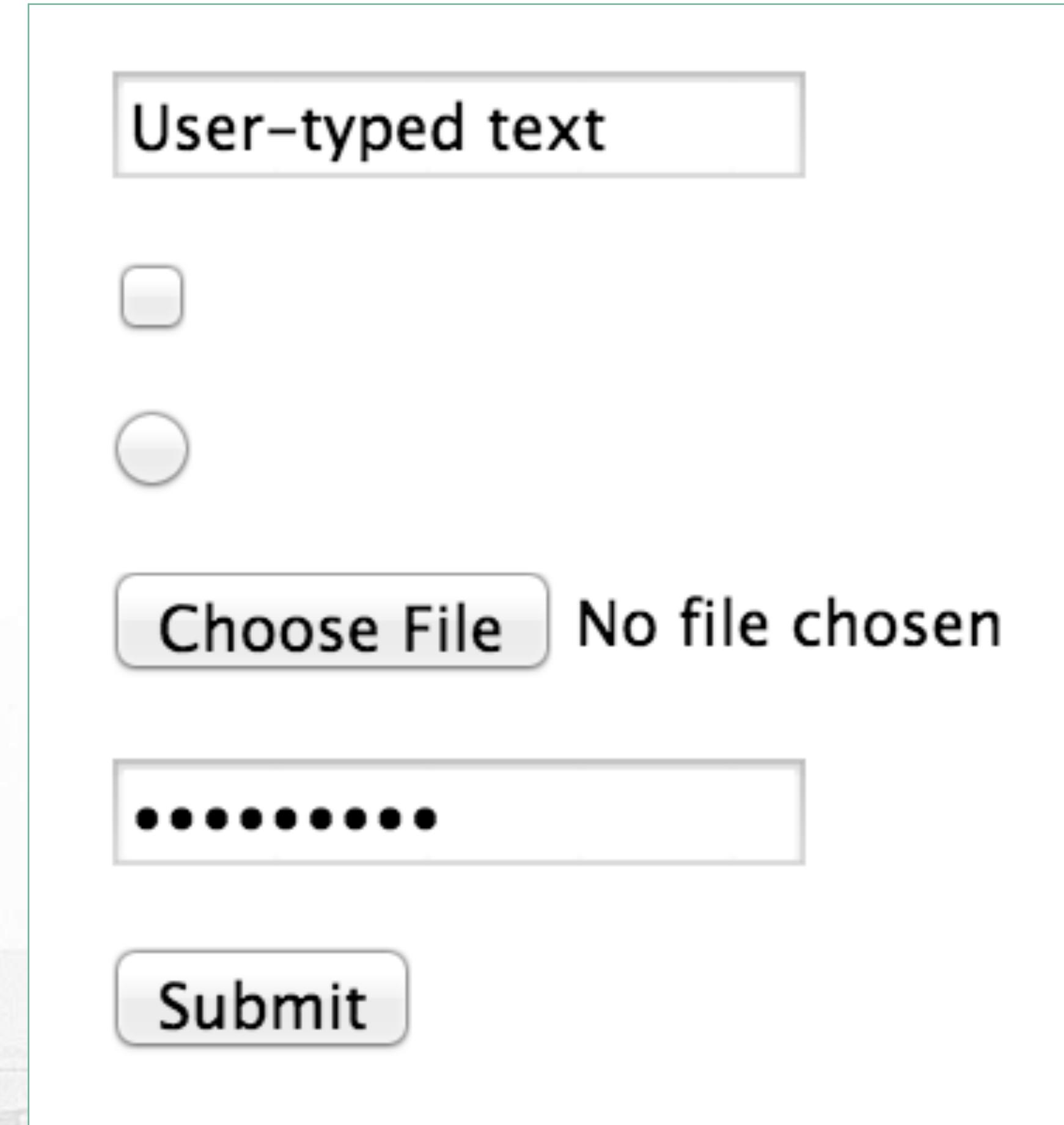
The diagram illustrates the correspondence between the HTML code and the resulting user interface. It features a light gray rectangular box containing a form. Inside the box, the text "Recipe Name" is followed by a text input field and a submit button labeled "Click to Submit". Three green arrows point from the labels "label", "input", and "submit" to their respective counterparts in the form interface. The "label" arrow points to the text "Recipe Name". The "input" arrow points to the text input field. The "submit" arrow points to the "Click to Submit" button.



# Common form input types

The **type** attribute sets the kind of input field that will display.

```
<input type="text">  
  
<input type="checkbox">  
  
<input type="radio">  
  
<input type="file">  
  
<input type="password">  
  
<input type="submit">
```



User-typed text

**Choose File** No file chosen

.....

**Submit**



# The **for** and **id** attributes

The value of the **for** attribute in the label should be the same as the value of the **id** attribute in an input field to associate the label and input.

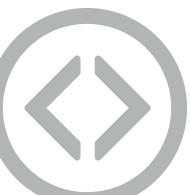
Each for/id pair has to be unique on the page.

HTML

```
<form>
  <label for="recipe-name">Recipe Name</label>
  <input type="text" id="recipe-name">
  <input type="submit" value="Click to Submit">
</form>
```

Recipe Name

Click to Submit



# Using a textarea instead of an input

input tags should be used for short or single-line user input.

The **textarea** tag is used for multi-line user input.

HTML

```
<form>          textareas also don't need to define a type  
    <label for="ingredients">Ingredients</label>  
    <textarea id="ingredients"></textarea>  
    <input type="submit" value="Click to Submit">  
</form>
```

A screenshot of a web browser showing a form. On the left, the word "Ingredients" is followed by a large text area. To the right of the text area is a button labeled "Click to Submit".

Unlike inputs, **textareas** need an opening and closing tag.



# Styling forms

labels and inputs are inline-level tags, but it usually makes sense to display one on top of the other like block-level instead of side-by-side.

Before

Recipe Name  Click to Submit

CSS

```
label, input {  
    display: block;  
}
```

After

Recipe Name

Click to Submit



# Styling forms

Once you've got both tags displaying as block-level, you can style them just like you would any other block-level tag.

CSS

```
label, input {  
    display: block;  
}  
  
label {  
    margin-bottom: 10px;  
}  
  
input {  
    width: 500px;  
    margin-bottom: 25px;  
}
```

Recipe Name

Your Name

Your Email

Click to Submit



# Styling the submit button separately

Since the submit button is technically an input tag, our input selector properties are affecting the way it is displayed.

**Attribute selectors** are a way to style a tag based on one of its attributes.

CSS

```
input {  
    width: 500px;  
    margin-bottom: 25px;  
}  
input[type=submit] {  
    width: 120px;  
    font-size: 30px;  
}
```

**Before**

Your Email

Click to Submit

**After**

Your Email

Click to Submit



# Styling inputs

The container around an input is actually just a border, so you can style it with the **border** property.

CSS

```
input[type=text] {  
  border: 2px solid #7facaa;  
}
```

Recipe Name

Your Name



# Styling inputs

To adjust the height of the input, style the text inside.

CSS

```
input[type=text] {  
  border: 2px solid #7facaa;  
  font-size: 24px;  
}
```

Recipe Name

Magic Cake

Your Name

Jon Friskics



# Styling inputs

And, since the green line is a border, use padding to put some space between the border and the text.

CSS

```
input[type=text] {  
  border: 2px solid #7facaa;  
  font-size: 24px;  
  padding: 7px;  
}
```

Recipe Name

Magic Cake

Your Name

Jon Friskics



# Styling textareas

You can set a width and height for **textareas**, but otherwise the styles behave similar to regular inputs.

CSS

```
textarea {  
    width: 500px;  
    height: 400px;  
    padding: 7px;  
    border: 2px solid #7facaa;  
    margin-bottom: 25px;  
    font-size: 24px;  
}
```

Ingredients



# Creating a separate style for a checkbox input

It looks awkward if the newsletter **label** and **input** are on separate lines since the checkbox is so small, so we can use attribute selectors again to make just this input and label display inline.

```
<form>  
  <label for="newsletter">Get Newsletter?</label>  
  <input type="checkbox" id="newsletter">  
  ...  
</form>
```

HTML

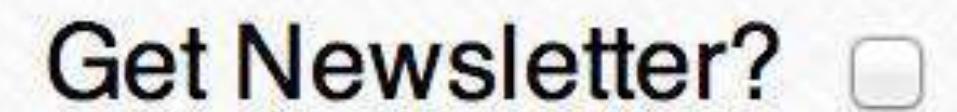
Before



```
input[type=checkbox], label[for=newsletter] {  
  display: inline;  
}
```

CSS

After



## Suggest a recipe

# The final form

Recipe Name

Ingredients

Your Name

Your Email

Get Newsletter?

[Submit the recipe](#)

