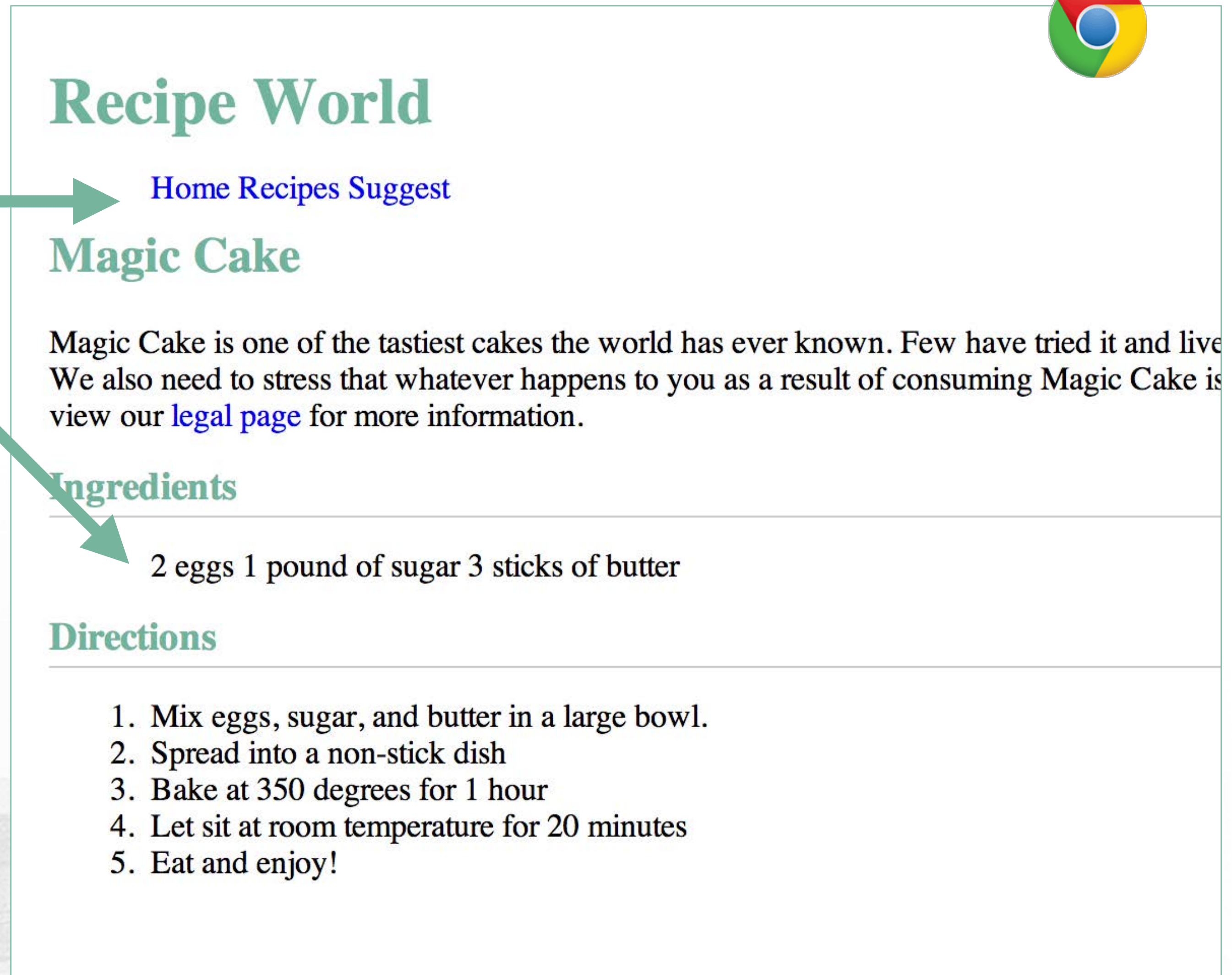# FRONT-END FOUNDATIONS

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** but <u>not</u> the other.

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!

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;
}
```

Recipe World

Home Recipes Suggest

Magic Cake

Magic Cake is one of the tastiest cakes the world has ev
We also need to stress that whatever happens to you as
view our legal page for more information.

Ingredients

2 eggs 1 pound of sugar 3 sticks of butter

*padding-left is gone from the nav *

*padding-left remains on the ingredients *

Using classes in descendent 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;
}
```

CSS

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 ev
We also need to stress that whatever happens to you as
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

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

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

Recipe World

still displayed inline → Home Recipes Suggest

Magic Cake

Magic Cake is one of the tastiest cakes the world has
We also need to stress that whatever happens to you a
view our legal page for more information.

Ingredients

displaying as a block-level tag again (the default) →
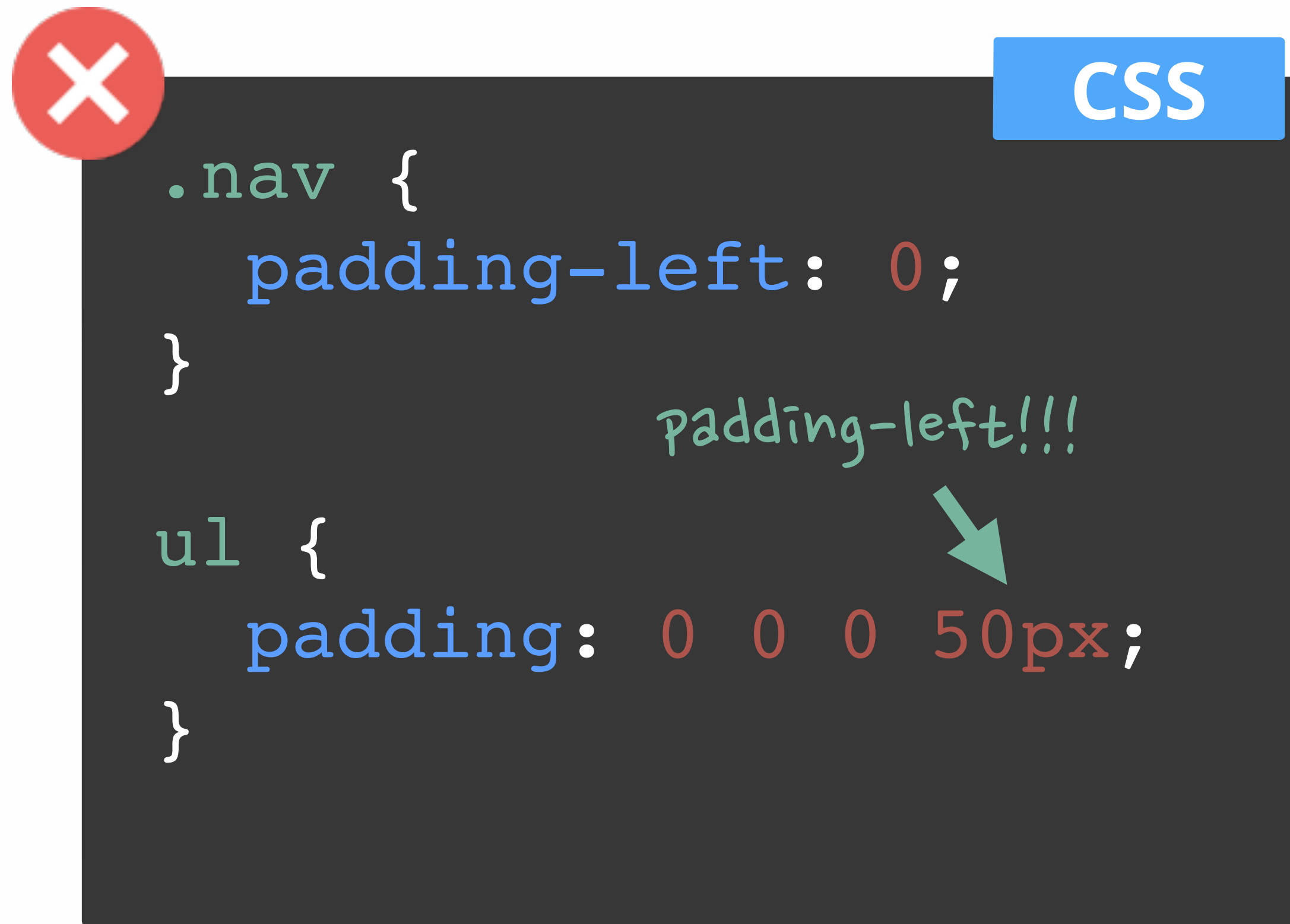- 2 eggs
- 1 pound of sugar
- 3 sticks of butter

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;
}

                    padding-left!!!

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

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 <u>after</u> 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**.

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

```
CSS
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;
```

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
<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>
```

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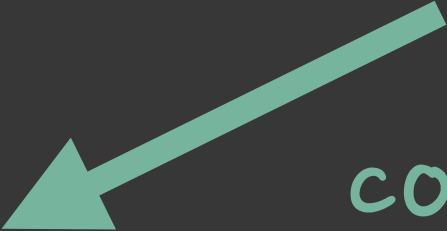
Creating the list of recipes

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>
  <li>...</li>
  <li>...</li>
</ul>
```

copy this general structure for each recipe

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

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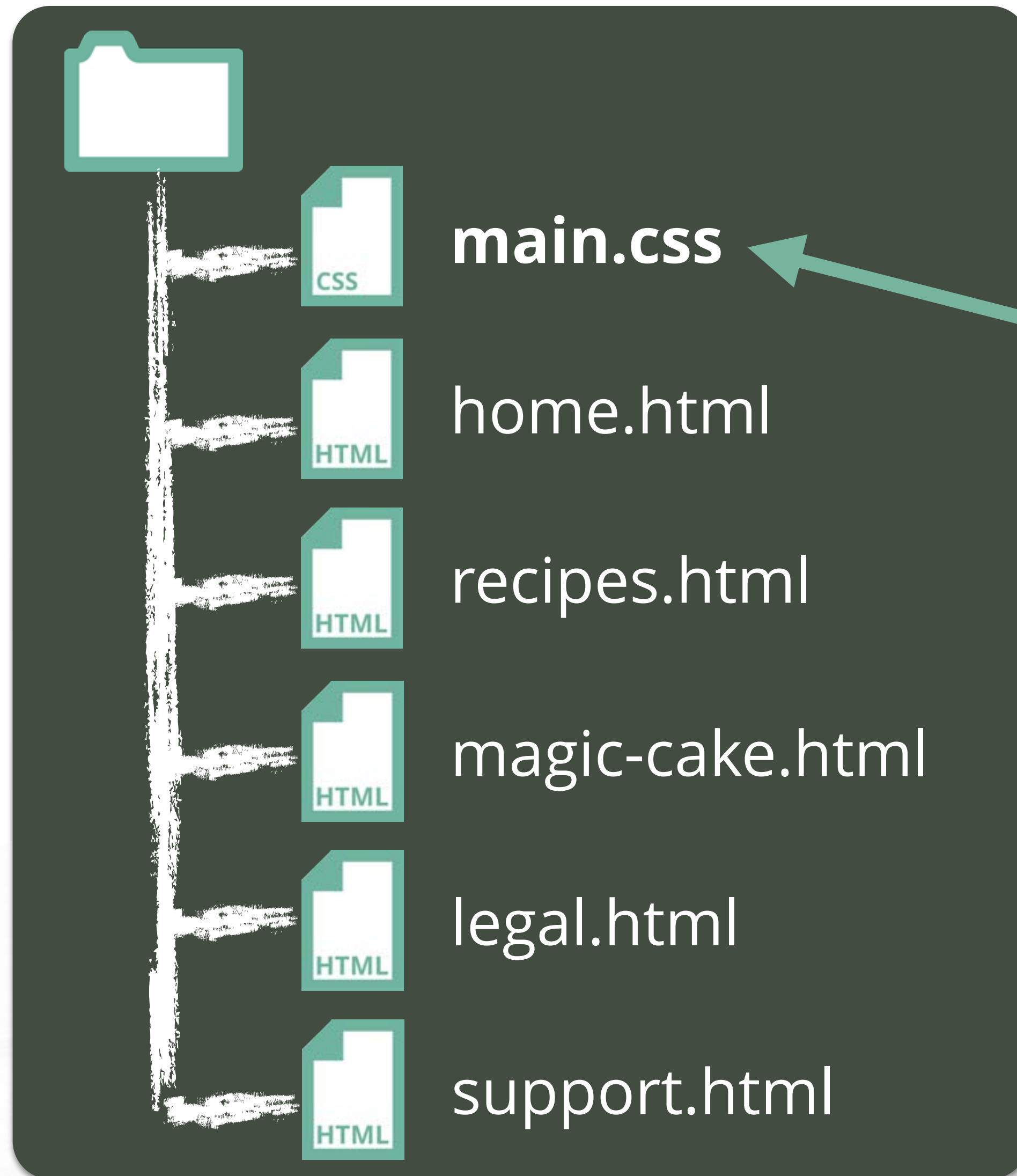
main.css

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** ←

📄 home.html

📄 recipes.html

📄 magic-cake.html

📄 legal.html

📄 support.html

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

main.css

CSS

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

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>
    <head>
        <link>
    </head>
    <body>

        . . .

    </body>
</html>
```

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.

```
<link></link>
```

Where do you write CSS?

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

recipes.html

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

    . . .

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

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

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

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.

A proposal for changing the recipes page styles

no list-style-type for the

border around the

left padding between the li border and content area

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.

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;
}
```

CSS

remember, .recipes is a specific

```css
.recipes {
    list-style-type: none;
    padding: 0;
}
```

CSS

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 '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

CSS

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

the .recipes h3 will inherit all of the h3

properties, but remove the bottom border

CSS

```
.recipes h3 {
    border-bottom: 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.

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

CSS

```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.

FRONT-END FOUNDATIONS

Organizing the page layout

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

Recipe World

Home Recipes Suggest

header

Magic Cake

Magic Cake is one of the tastiest cakes the world has eve
We also need to stress that whatever happens to you as a
view our legal page for more information.

main

content

Ingredients

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

Directions

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

The div tag

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

```
<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.

let's build this style

HTML

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

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 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.

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

```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.

centered!

If: you want to center an <u>entire</u> 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;
}
```

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.*

Centering content

There's two main ways to center things, but the CSS you write depends on what you're centering.
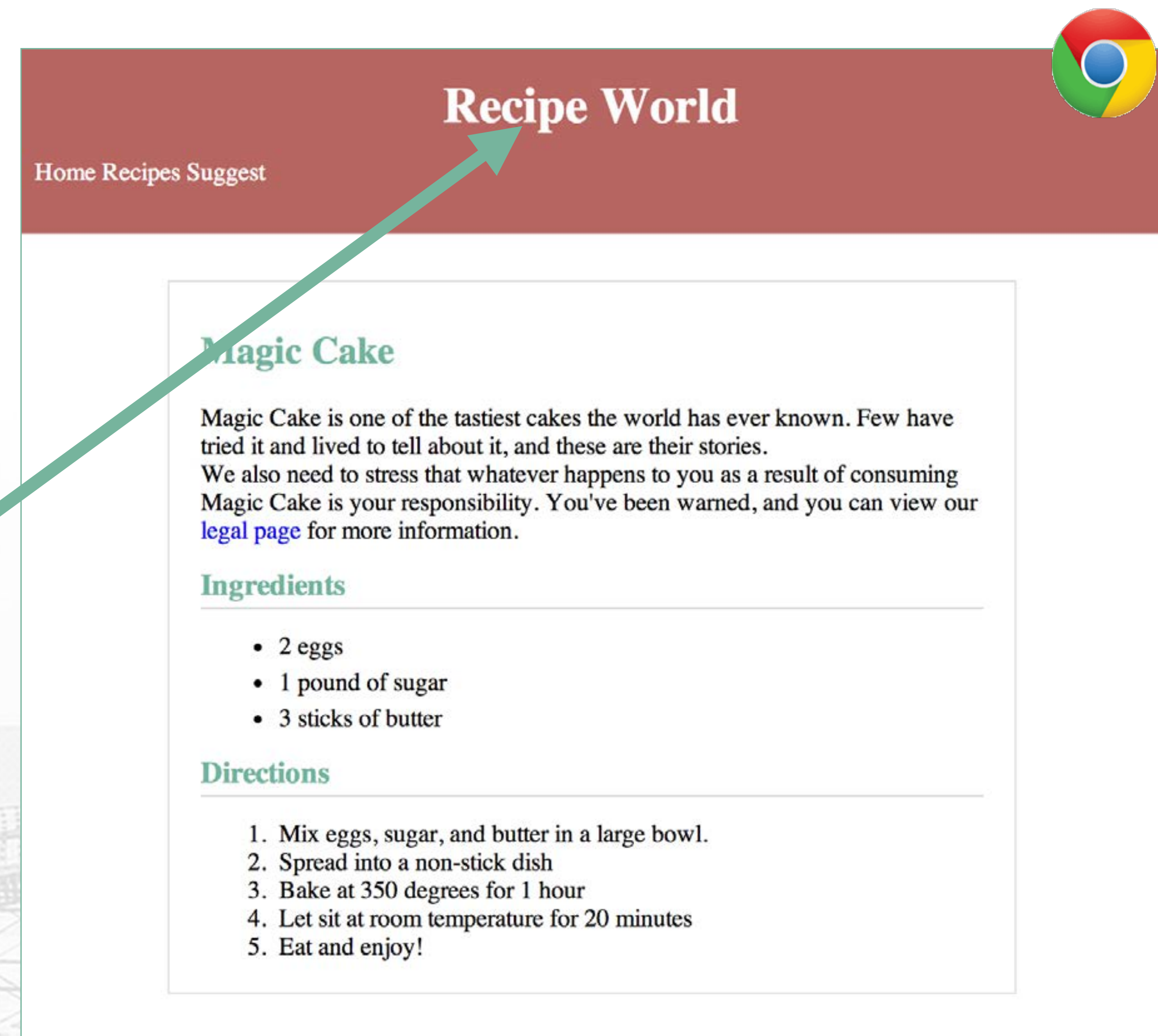
If: you want to center children <u>inside</u> a block-level tag

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

CSS

```css
h1 {
  color: #ffffff;
  margin: 0 0 15px 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!

Centering content

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

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, and after this course, you can continue on to our **Journey Into Mobile** course where you'll learn how to convert your fixed widths to relative ones!