HTML Page Structure

Learning Outcomes

- Understand the structure of an **HTML Element**, and be able to recognize its variants.

- Be able to read and compose a **relative path**, and be able to distinguish it from an **absolute path**.

- Understand the implications of **nesting** of elements, and in particular be able to distinguish between correct and incorrect nesting.

- Be able to differentiate between **block** and **inline** elements
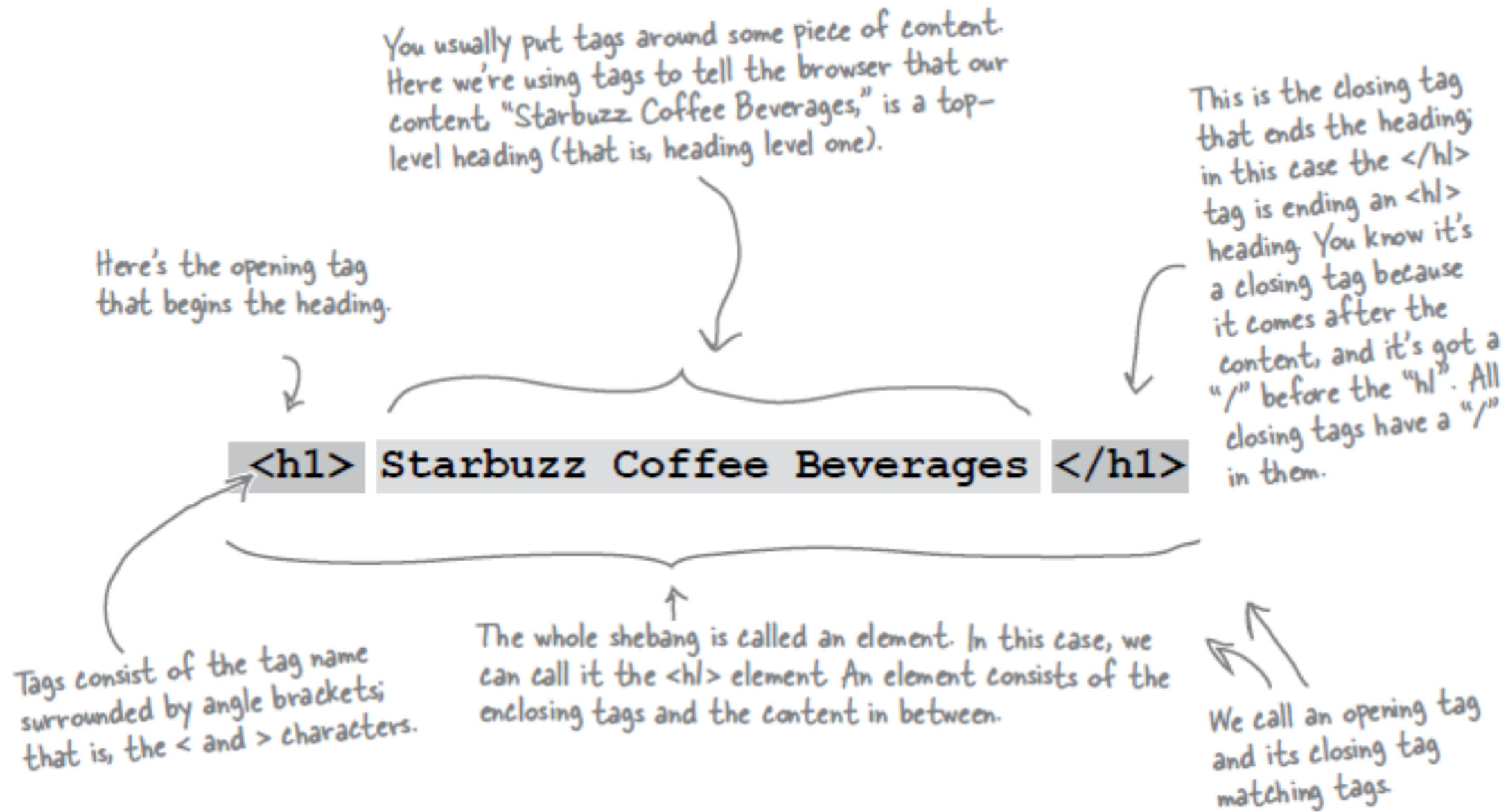
- Understand the general context of **wireframing**

Agenda

- Elements, Attributes, & Documents

- Linking

- Nesting

- Linebreak, Block & Inline Elements
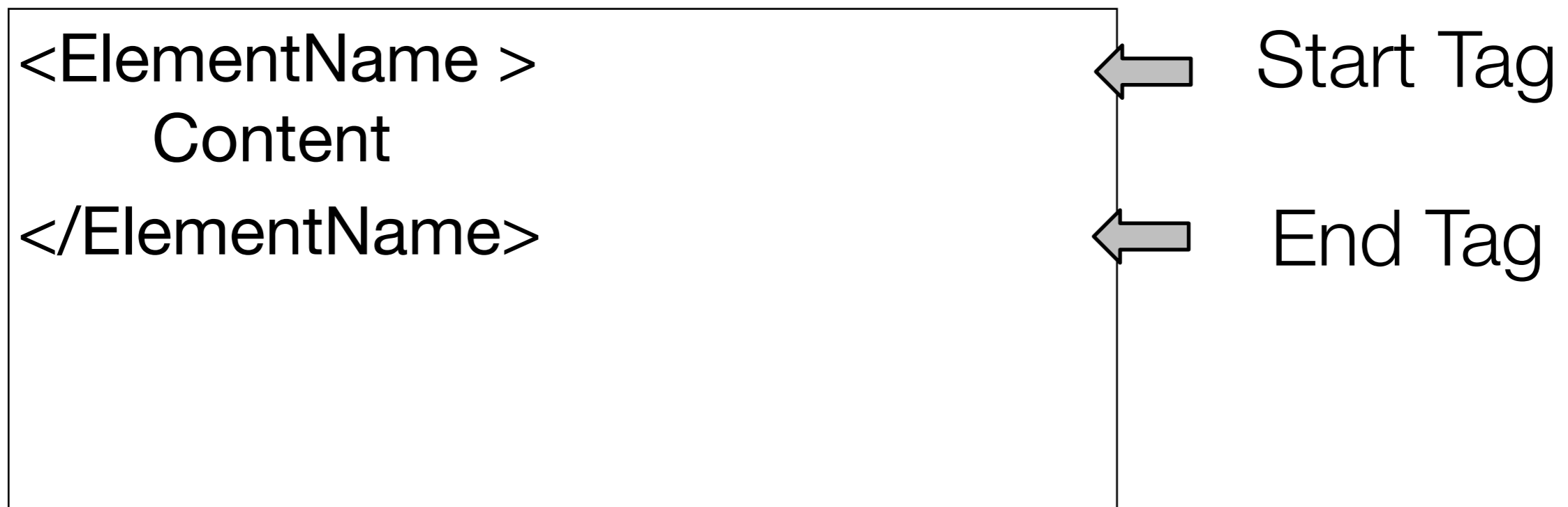
- Structuring a Page / Wireframing

Agenda

- Elements, Attributes, & Documents

- Linking

- Nesting

- Linebreak, Block & Inline Elements

- Structuring a Page / Wireframing

Components of an HTML Element

You usually put tags around some piece of content. Here we're using tags to tell the browser that our content, "Starbuzz Coffee Beverages," is a top-level heading (that is, heading level one).

This is the closing tag that ends the heading; in this case the </h1> tag is ending an <h1> heading. You know it's a closing tag because it comes after the content, and it's got a "/" before the "h1". All closing tags have a "/" in them.

Here's the opening tag that begins the heading.

`<h1>` Starbuzz Coffee Beverages `</h1>`

Tags consist of the tag name surrounded by angle brackets; that is, the < and > characters.

The whole shebang is called an element. In this case, we can call it the <h1> element. An element consists of the enclosing tags and the content in between.

We call an opening tag and its closing tag matching tags.

Components of an HTML Element

<ElementName >
   Content
</ElementName>

⬅ Start Tag

⬅ End Tag

<title>

<title> My App Store </title>

ElementName: *<title>*

Content: *My App Store*

ElementName: *</title>*

\<p>

\<p>
  600 million YouTube views later, Dude Perfect is back
  with their most epic game yet! Go BIGGER than ever
  hitting mind-blowing trick shots through tons of crazy
  levels!
\</p>

ElementName:	*\<p>*
Content:	600 million YouTube views later, Dude Perfect is back with their most epic game yet! Go BIGGER than ever hitting mind-blowing trick shots through tons of crazy levels!
ElementName:	*\</p>*

Attributes

- Attributes give you a way to specify additional information about an element.

SAFETY FIRST

Attributes are always written the same way: first comes the attribute name, followed by an equals sign, and then the attribute value surrounded in double quotes.

You may see some sloppy HTML on the Web that leaves off the double quotes, but don't get lazy yourself. Being sloppy can cause you a lot of problems down the road (as we'll see later in the book).

Do this (best practice)

`<a href="top10.html">Great Movies</a>`

attribute name
equals sign
double quote
attribute value
double quote

Not this

`<a href=top10.html>Great Movies</a>`

No double quotes around the attribute value

\<a\>

 App Store

ElementName:	*\<a\>*
AttributeName:	`href`
`AttributeValue:`	"apps.html"
Content:	*App Store*
ElementName:	*\</a\>*

ElementName:	*<img>*
AttributeName:	src
AttributeValue:	"../images/delete.jpg"
Content:	*empty*
ElementName:	*none*

Attributes

there are no
Dumb Questions

Q: **Can I just make up new attributes for an HTML element?**

A: Web browsers only know about a predefined set of attributes for each element. If you just made up attributes, then browsers wouldn't know what to do with them, and as you'll see later in the book, doing this will very likely get you into trouble. When a browser recognizes an element or an attribute, we like to say that it "supports" that element or attribute. You should only use attributes that you know are supported.

HTML Document Structure

- html
  - head
    - title
  - body
    - h1
    - ol
    - etc...

```html
<!DOCTYPE HTML>
<html>
  <head>
    <title>APP Store</title>
  </head>
  <body>
    <h1>Mobile Applications</h1>
    <ol>
      <li><a href="apps.html">Apps</a></li>
    </ol>
    <h2>Most Popular Apps</h2>
    <ul>
      <li><img src="images/lightening.jpg"/>Strike I</li>
      <li><img src="images/plane.png"/>Crash Landing</li>
    </ul>
    <h2>Recommended Apps</h2>
    <ul>
      <li><img src="images/warrior.jpg"/>Chop</li>
      <li><img src="images/xbox.jpg"/>XBox mania</li>
    </ul>
  </body>
</html>
```

Agenda

- Elements, Attributes, & Documents

- Linking

- Nesting

- Linebreak, Block & Inline Elements

- Structuring a Page / Wireframing

Linking
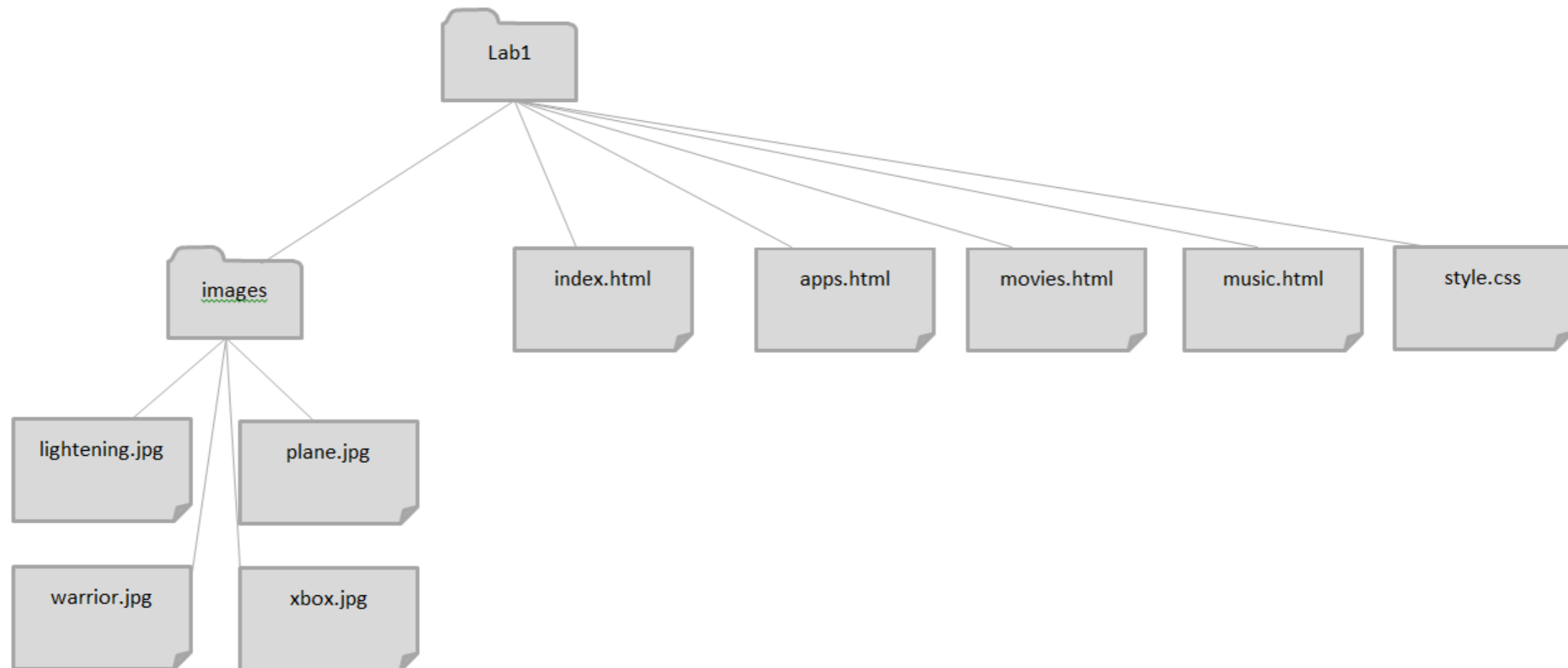
Linking to other pages or images

- Creating links to other web pages and to image files can get confusing! If your link to a file or page is incorrect you get:

    Whoops we can't seem to find that page! (404 error)

Or

    No image shows and you have a broken link to an image

Links: Absolute vs Relative

Absolute

- Complete path to a file on the hard disk: e.g:

    C:/My Documents/webdevelopment/lab01/images/xbox.jpg

    C:My Documents/webdevelopment/lab01/movies.html

- Relative:

    ./images/xbox.jpg

    ../apps.html

    index.html

    Trace route from "current position" to the destination

    ".." means go up one level

    Directory name may prefix filename

Relative Link Examples

- If we are in "lab1" then "images/plane.png" is a relative link from the current folder (Lab1) to the images folder, and to the file "plane.png" in that folder

- If we are in the images folder, and we wish to link to one of the web pages, the link "../movies" means "go up one level, and then find "movies.html"

- Avoid absolute links!

```
<a href="movies.html">Movies</a>
```

```
<img src="./images/warrior.jpg">
```

Agenda

- Elements, Attributes, & Documents

- Linking

- Nesting

- Linebreak, Block & Inline Elements

- Structuring a Page / Wireframing

Nesting

- When we put one element inside another element, we call that nesting.

- We say, the <p>element is nested inside the <body>element.

- We put a <body>element inside an <html>element, a <p>element inside a <body>element etc.

<html>

<body>

<p>

<q>

<q> nested inside <p>,
nested inside <body>,
nested inside <html>.
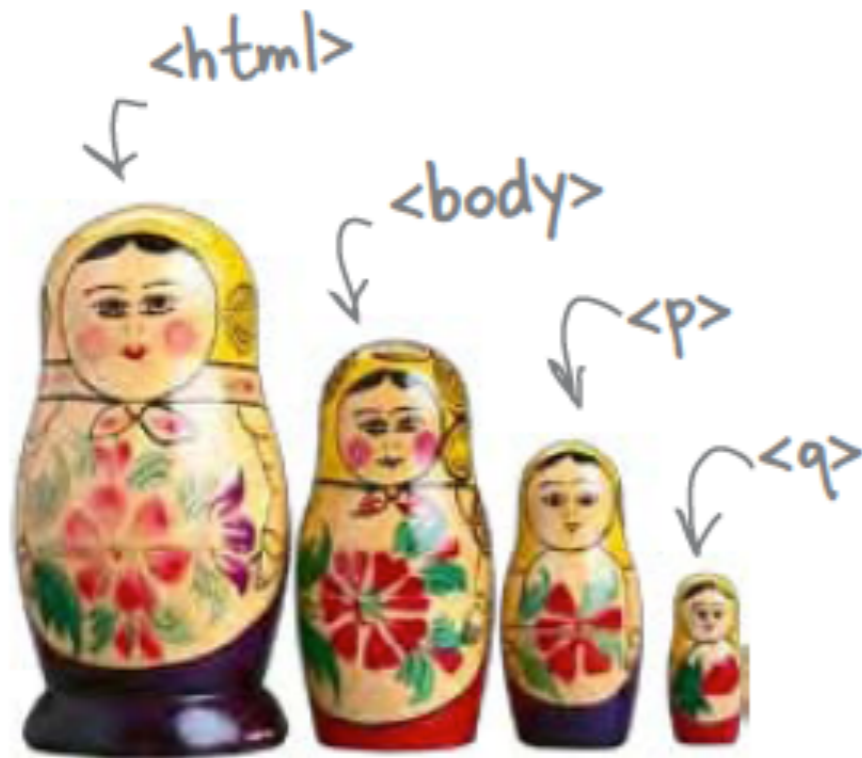
Nesting - Tree Structure

```html
<html>
  <head>
    <title>Musings</title>
  </head>
  <body>
    <p>
        To quote Buckaroo,
        <q>The only reason
           for time is so
           that everything
           doesn't happen
           at once.</q>
    </p>
  </body>
</html>
```

<html> is always the element at the root of the tree.

<html> has two nested elements: <head> and <body>. You can call them both "children" of <html>.

<body> is nested within the <html> element, so we say <body> is the "child" of <html>.

title> is nested within the head> element.

The parent of <q> is <p>, the parent of <p> is <body>, the parent of <body> is <html>.

html
head   body
title   p
         q
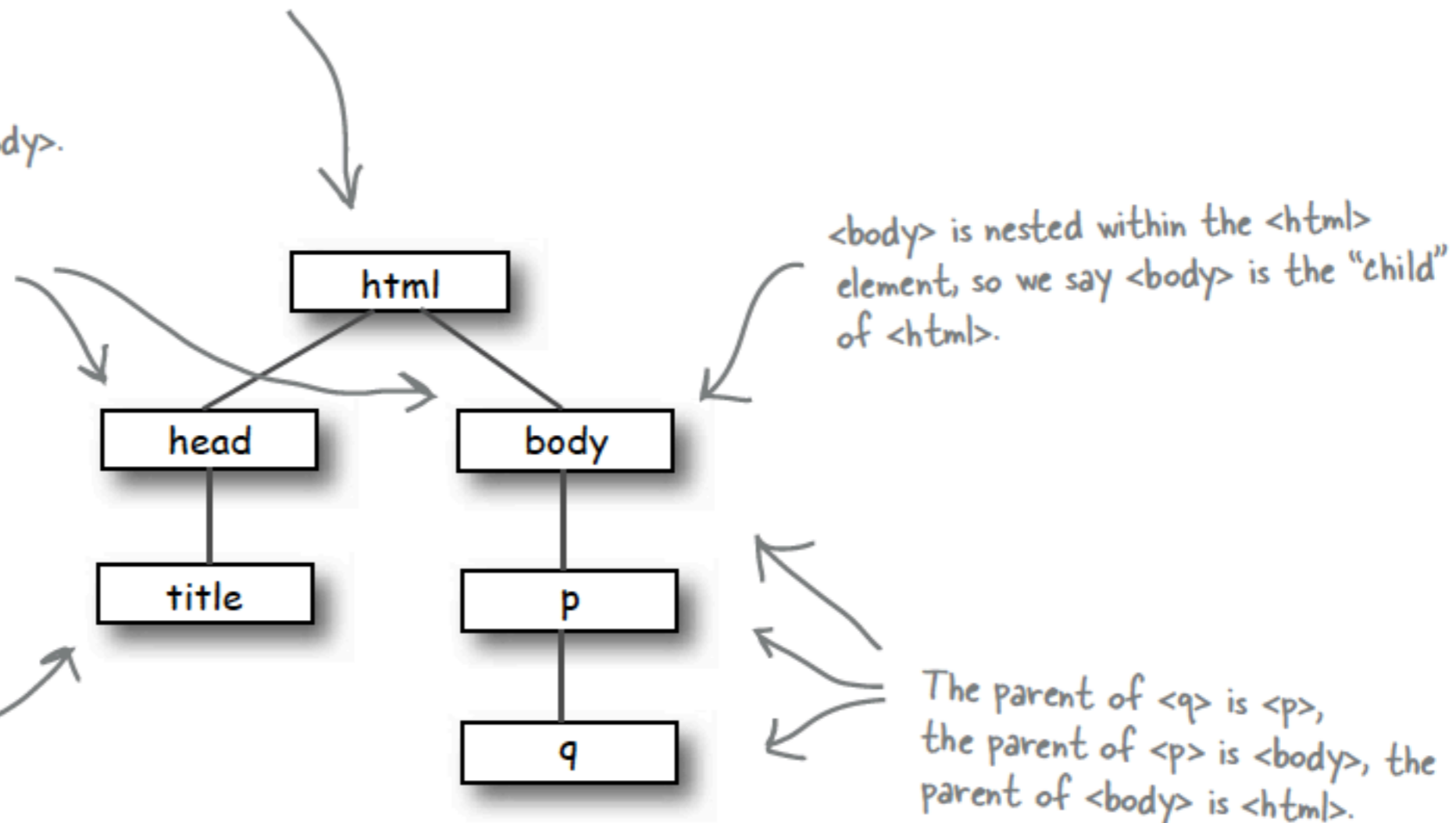
Nesting can be Incorrect!

`<p>I'm so going to blog <em>this</em></p>`
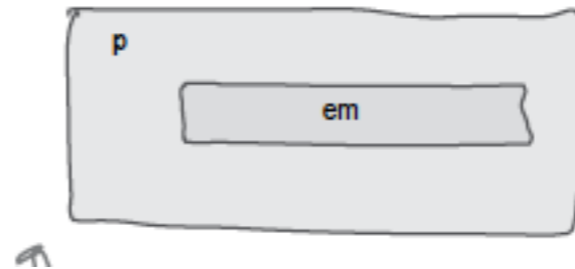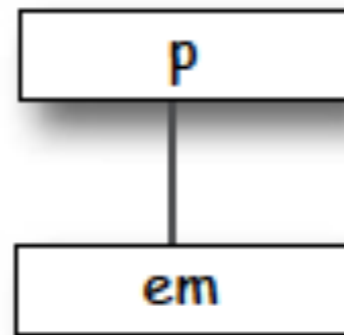
Good

`<p>I'm so going to blog <em>this</p></em>`

?

Bad

Agenda

- Elements, Attributes, & Documents

- Linking

- Nesting

- Linebreak, Block & Inline Elements

- Structuring a Page / Wireframing

Line breaks

there are no
Dumb Questions

Q: I think I know what a linebreak is; it's like hitting the carriage return on a typewriter or the Return key on a computer keyboard. Right?

A: Pretty much. A linebreak is literally a "break in the line," like this, and happens when you hit the Return key, or on some computers, the Enter key. You already know that linebreaks in HTML files don't show up visually when the browser displays a page, right? But now you've also seen that anytime you use a block element, the browser uses linebreaks to separate each "block."

Line Breaks

```
<h2>July 14, 2012</h2>
<p>
    I saw some Burma Shave style signs on the
    side of the road today:
</p>
<blockquote>
    Passing cars, <br>
    When you can't see, <br>
    May get you, <br>
    A glimpse, <br>
    Of eternity. <br>
</blockquote>
<p>
    I definitely won't be passing any cars.
</p>
```
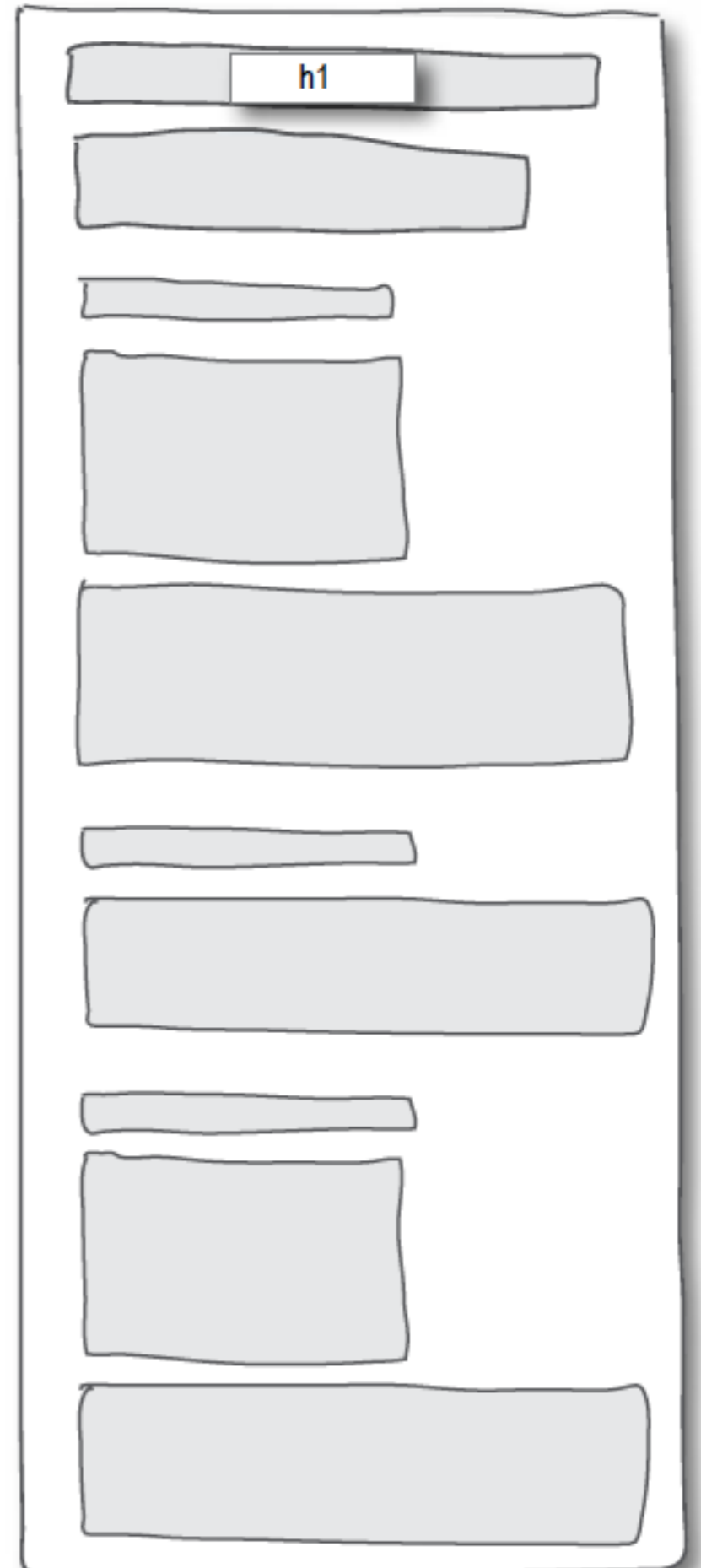
Here's the July 14th snippet from Tony's page.

Add a <br> element to any line when you want to break the flow and insert a "linebreak."

Line Breaks

Chance, CO, Why, AZ and Truth or Consequences, NM.

July 14, 2012

I saw some Burma Shave style signs on the side of the road today:

> Passing cars,
> When you can't see,
> May get you,
> A glimpse,
> Of eternity.

I definitely won't be passing any cars.

June 2, 2012
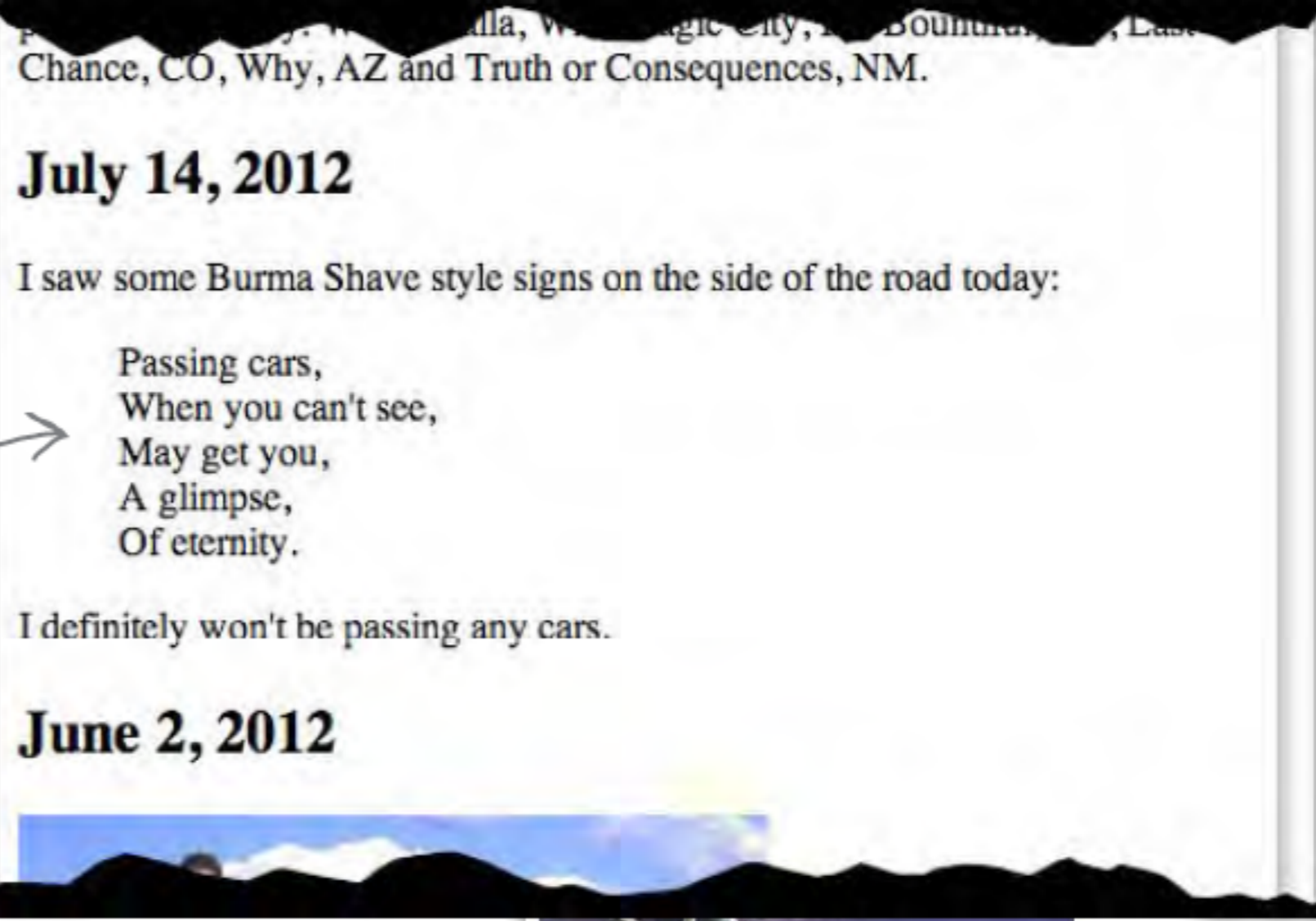
Each line now has a linebreak after it.

Line Breaks

- Break:
 </br>

- An Empty element: - sometimes shortened to:

  -
</br>

- or just

  -

- or

  -

Block vs Inline Elements

- Block elements are always displayed as if they have a line break before and after them

- inline elements appear "in line" within the flow of the text in your page.

```
  "Block elements
stand on their own;
 inline elements go
   with the flow."
```

Examples

- Block - h1, h2, p, blockquote
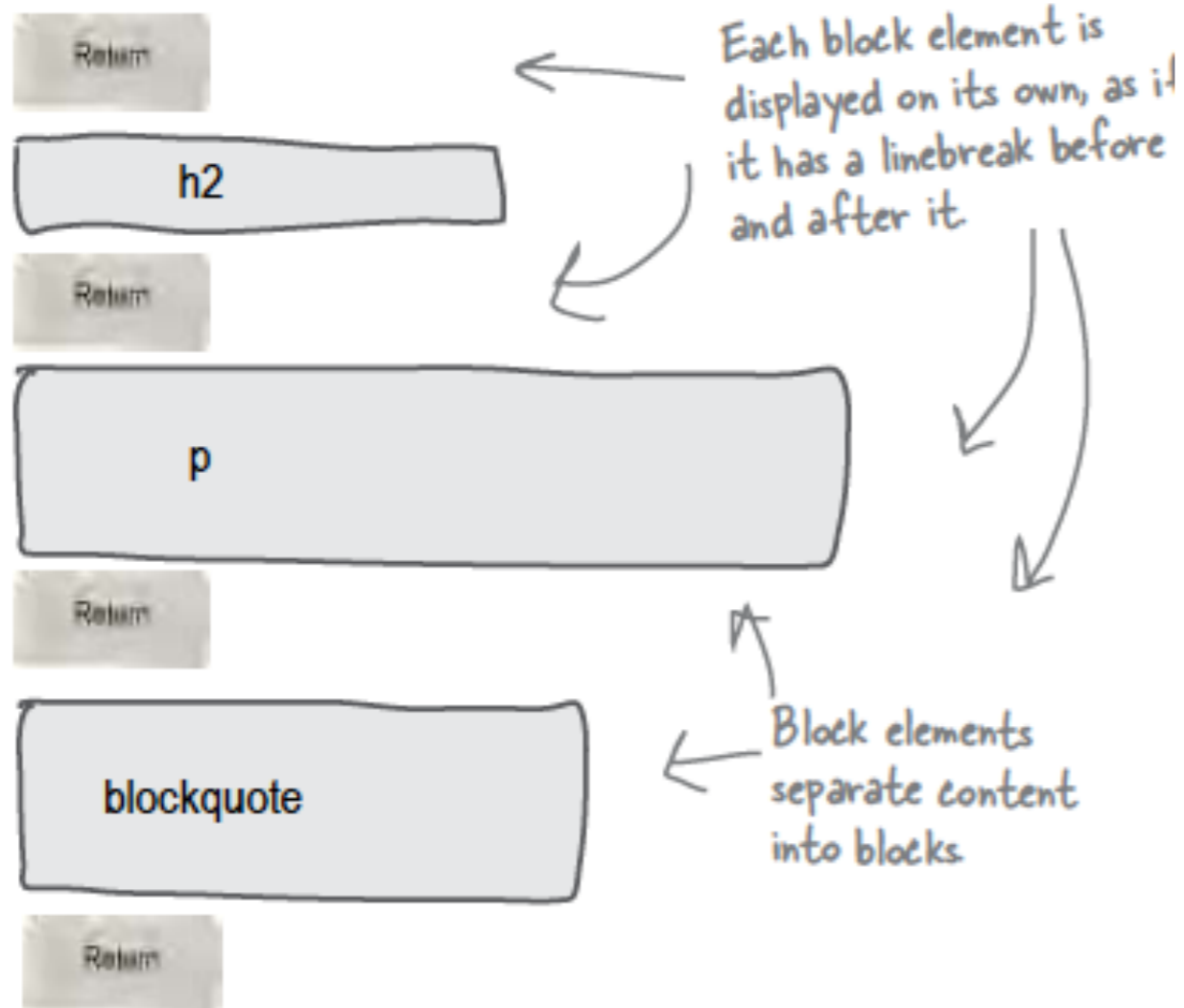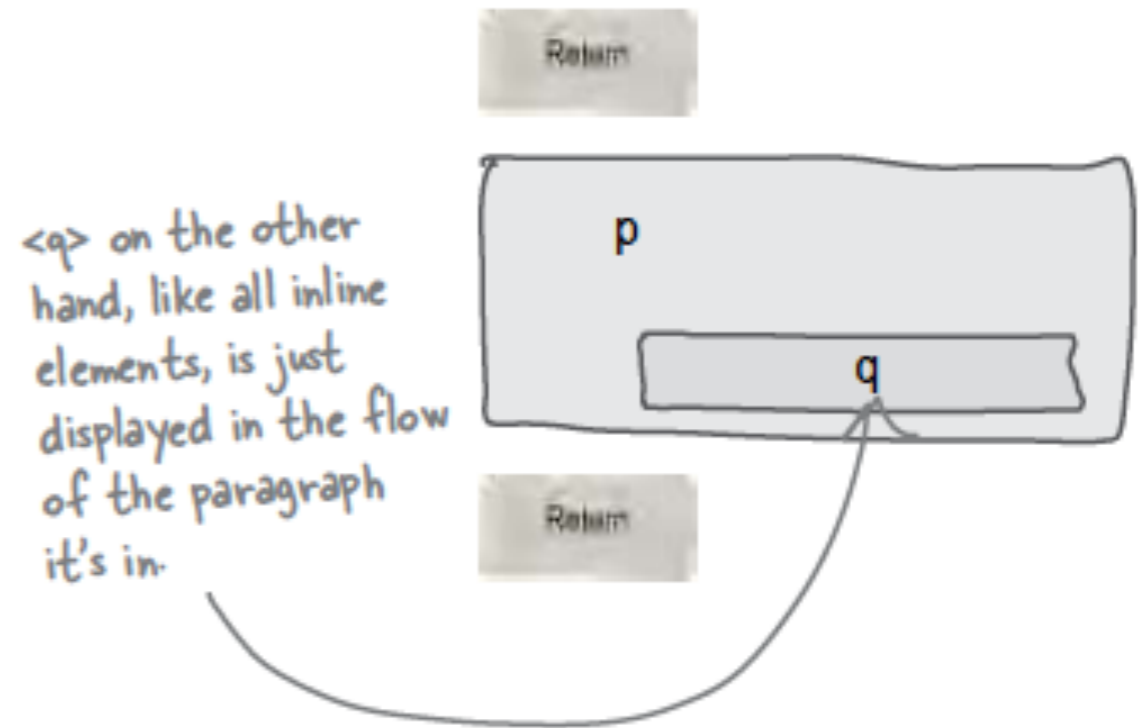
Each block element is displayed on its own, as if it has a linebreak before and after it

Block elements separate content into blocks

- Inline - a, em, q

<q> on the other hand, like all inline elements, is just displayed in the flow of the paragraph it's in.

Agenda

- Elements, Attributes, & Documents

- Linking

- Nesting

- Linebreak, Block & Inline Elements

- Structuring a Page / Wireframing

Planning a Document (1)

- Visualizing End Result

Planning a Document (2)

- Identify Block Structure

h1

Planning a Document (3)

- Building blocks

  - Don't forget \<html\> \<head\> and \<body\>

- Translate to html

  - Build content

```html
<html>
  <head>
    <title>My Trip Around the USA on a Segway</title>
  </head>
  <body>

    <h1>Segway'n USA</h1>
    <p>
        Documenting my trip around the US on my very own Segway!
    </p>

    <h2>August 20, 2012</h2>
    <img src="images/segway2.jpg">
    <p>
      Well I made it 1200 miles already, and I passed
      through some interesting places on the way: Walla Walla,
      WA, Magic City, ID, Bountiful, UT, Last Chance, CO,
      Why, AZ and Truth or Consequences, NM.
    </p>

    <h2>July 14, 2012</h2>
    <p>
      I saw some Burma Shave style signs on the side of the
      road today: "Passing cars, When you can't see, May get
      you, A glimpse, Of eternity." I definitely won't be passing
      any cars.
    </p>

    <h2>June 2, 2012</h2>
    <img src="images/segway1.jpg">
    <p>
      My first day of the trip! I can't believe I finally got
      everything packed and ready to go. Because I'm on a Segway,
      I wasn't able to bring a whole lot with me: cell phone, iPod,
      digital camera, and a protein bar. Just the essentials. As
      Lao Tzu would have said, "A journey of a thousand miles begins
      with one Segway."
    </p>

  </body>
</html>
```

h1
p
h2
img
p
h2
p
h2
img
p
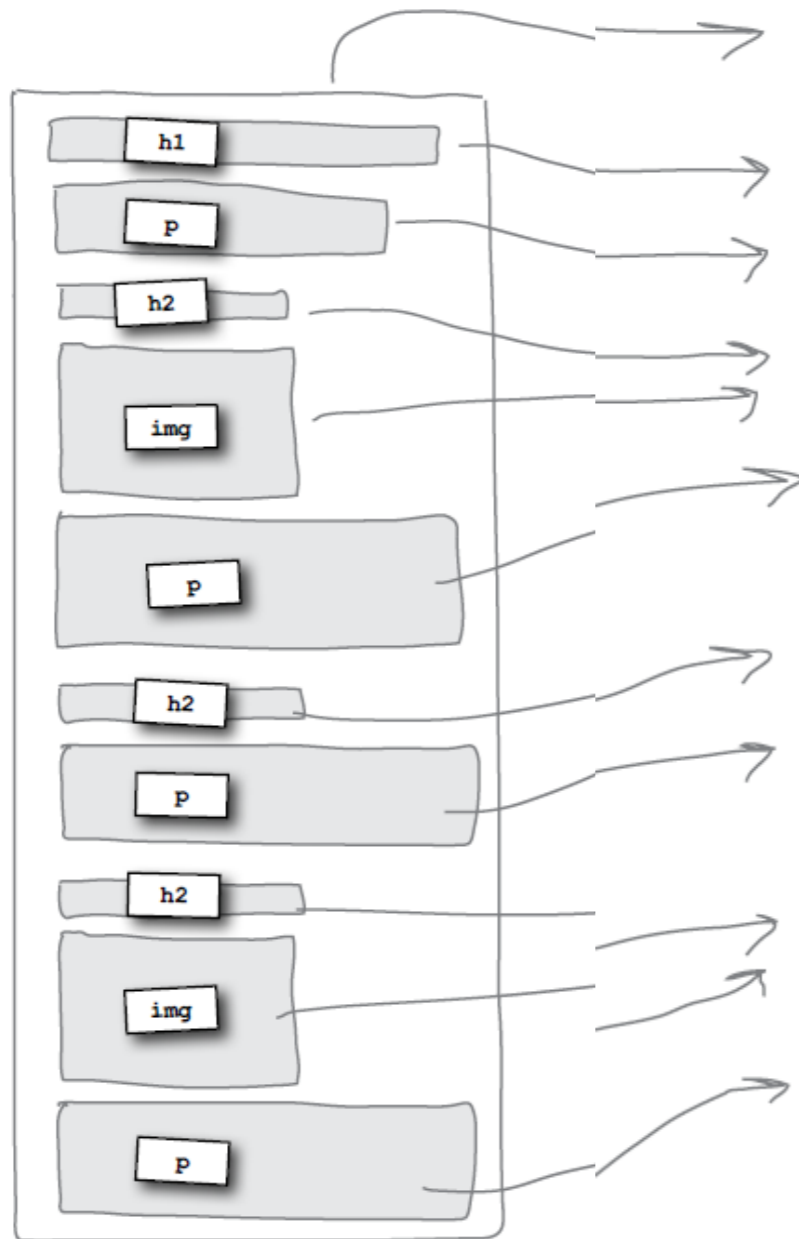
Wire framing

- A website wireframe is a basic [visual guide](#) used in [interface design](#) to suggest the structure of a [website](#) and relationships between its pages.

- A webpage wireframe is a similar illustration of the [layout](#) of fundamental elements in the [interface](#). Typically, wireframes are completed before any artwork is developed.
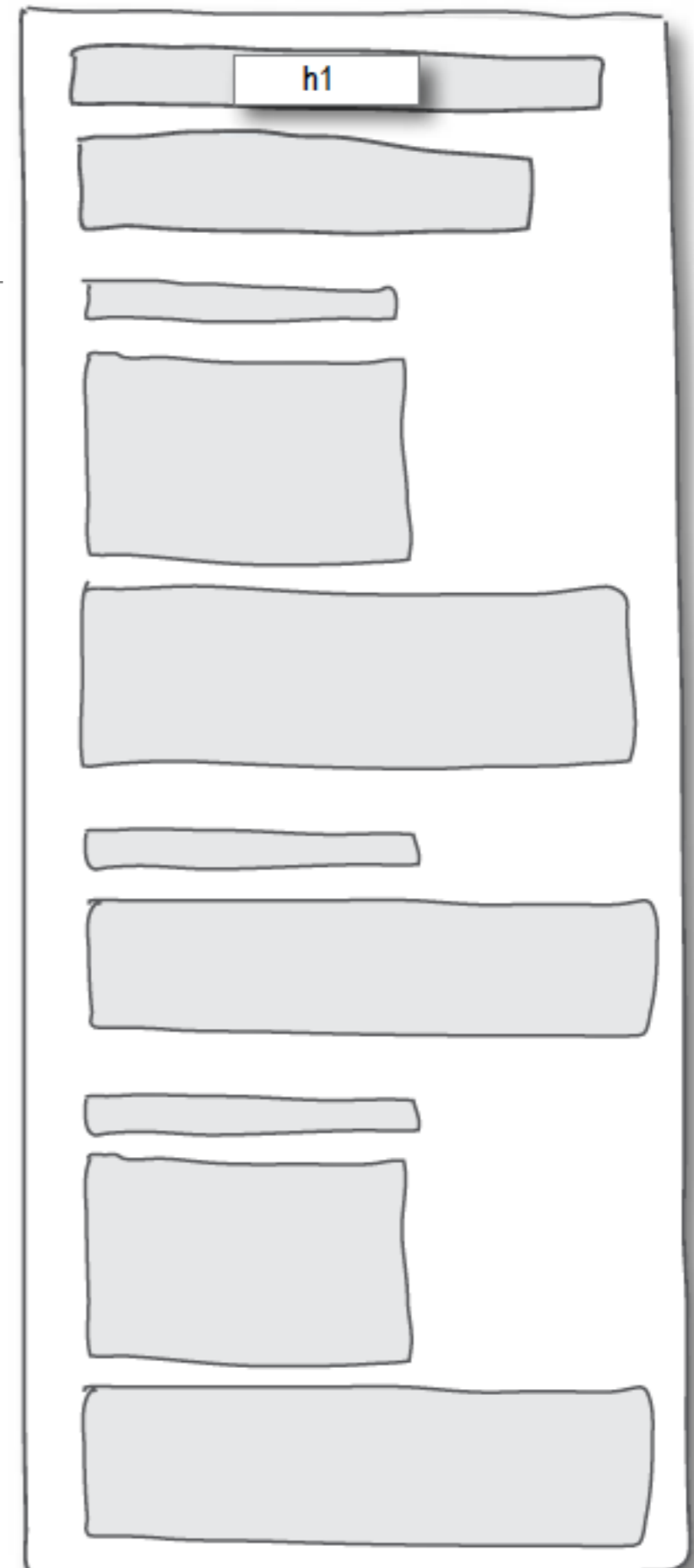
Agenda

- Elements, Attributes, & Documents

- Linking

- Nesting

- Linebreak, Block & Inline Elements

- Structuring a Page / Wireframing

Learning Outcomes

- Understand the structure of an ***HTML Element***, and be able to recognize its variants.

- Be able to read and compose a ***relative path***, and be able to distinguish it form an ***absolute path***.

- Understand the implications of ***nesting*** of elements, and in particular be able to distinguish between correct and incorrect nesting.

- Be able to differentiate between ***block*** and ***inline*** elements

- Understand the general context of ***wireframing***

Labs

1: Introducing HTML   Lab-1 HTML Structure   01   02   03   04   Exercises

HTML pages

We are going to create a web site that is based on the Google Play site or the Apple app store. It will present mobile apps, music, and movies to the user.

Choose New File from the File menu, next choose Save from the file menu and name it `apps.html`

You should now have the following folder and file structure:

```
FOLDERS
  ▼ 📁 web-development
    ▼ 📁 lab01
      ▼ 📁 images
          🖾 banner.png
          🖾 ondesoft.jpg
        <> apps.html
        <> index.html
```
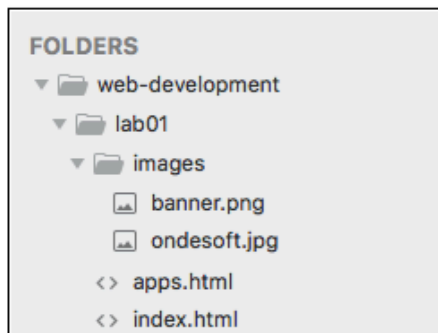
Next you will write some html code. Copy and paste in the following code:

index.html page content

```html
<!DOCTYPE HTML>
<html>
  <head>
    <title>Bundle APP Store</title>
  </head>
  <body>
    <img src="images/banner.png"/>
    <h1>Welcome to the App Bundle Store</h1>
    <p>
      This store brings you great app bundles week after week. We select the best power
      user apps from a broad range of suppliers and combine them into great deals. These are the highest quality
    </p>
    <p>
      Whether you are interested in gaming or graphics design, software development or media production - we have
    </p>
      <h2>Favourites</h2>
    <ul>
      <li>Hype by Tumult</li>
      <li>Webstorm by Idea</li>
      <li>Sublime, by sublimetext.com</li>
      <li>Desktop Utility by Sweet Productions</li>
```

38

Typical Lab:

- Screen shots to show outcome
- Some html to type
- More screen shots to
- show expected outcome
- Reflect as you go along

First Project

Consider creating a folder structure for your labs - starting with a folder called web-development

Open Sublime and choose Open from the File menu, navigate to the web-development folder you have just c

In sublime click on the web-development folder, right click - and select New Folder. Enter the folder name in the sidebar:
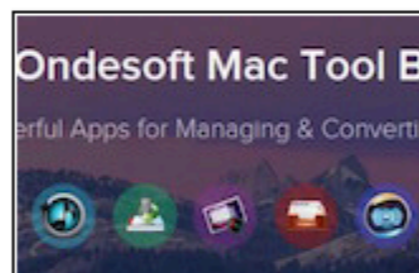
```
FOLDERS
▼ 📁 web-development
  ▶ 📁 lab01
```

Chose New File from the File menu, a new blank page should appear in the sublime window. Click Save on the F it into the lab01 folder. You should see the file name appear in the left panel in sublime.

```
▼ WebDevelopment
  ▼ Lab1
      index.html
```

Now in File Explorer create another folder in Lab1 that is called images

```
FOLDERS
▼ 📁 web-development
  ▼ 📁 lab01
    ▶ 📁 images
    <> index.html
```
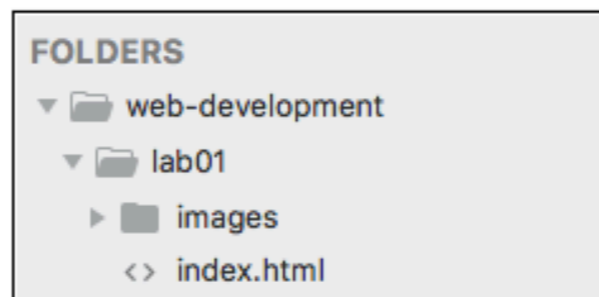
Next you are to save the following images into the images folder:

Ondesoft Mac Tool B
erful Apps for Managing & Converti

Lab-1.1

- Introduce you to the tools we will use during the web development

- Introduce you to creating, editing, saving and displaying a simple web page.

Lab 1.1

Create a WebDevelopment folder
to contain a new Lab folder
each week.

Create a simple static web site,
and be able to manage it within
the Sublime environment

Familiar with the following HTML
elements:
                <html> <head> <title>
                <link> <body> <h1> <h2>
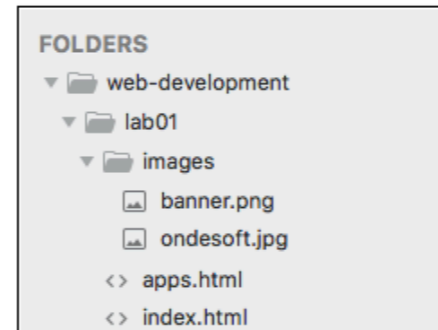   <p> <a> <img> and

HTML pages

We are going to create a web site that is based on the Google Play site or the Apple app store. It will present mobile apps, mu

Choose New File from the File menu, next choose Save from the file menu and name it `apps.html`

You should now have the following folder and file structure:

```
FOLDERS
  ▼ 📁 web-development
    ▼ 📁 lab01
      ▼ 📁 images
          🖼 banner.png
          🖼 ondesoft.jpg
      <> apps.html
      <> index.html
```

Next you will write some html code. Copy and paste in the following code:

index.html page content

```html
<!DOCTYPE HTML>
<html>
  <head>
    <title>Bundle APP Store</title>
  </head>
  <body>
    <img src="images/banner.png"/>
    <h1>Welcome to the App Bundle Store</h1>
    <p>
      This store brings you great app bundles week after week. We select the best power
      user apps from a broad range of suppliers and combine them into great deals. Thes
    </p>
    <p>
      Whether you are interested in gaming or graphics design, software development or
    </p>
      <h2>Favourites</h2>
    <ul>
      <li>Hype by Tumult</li>
      <li>Webstorm by Idea</li>
      <li>Sublime, by sublimetext.com</li>
      <li>Desktop Utility by Sweet Productions</li>
```