HTML Basics

HTML Basics

Essential characteristics of html code

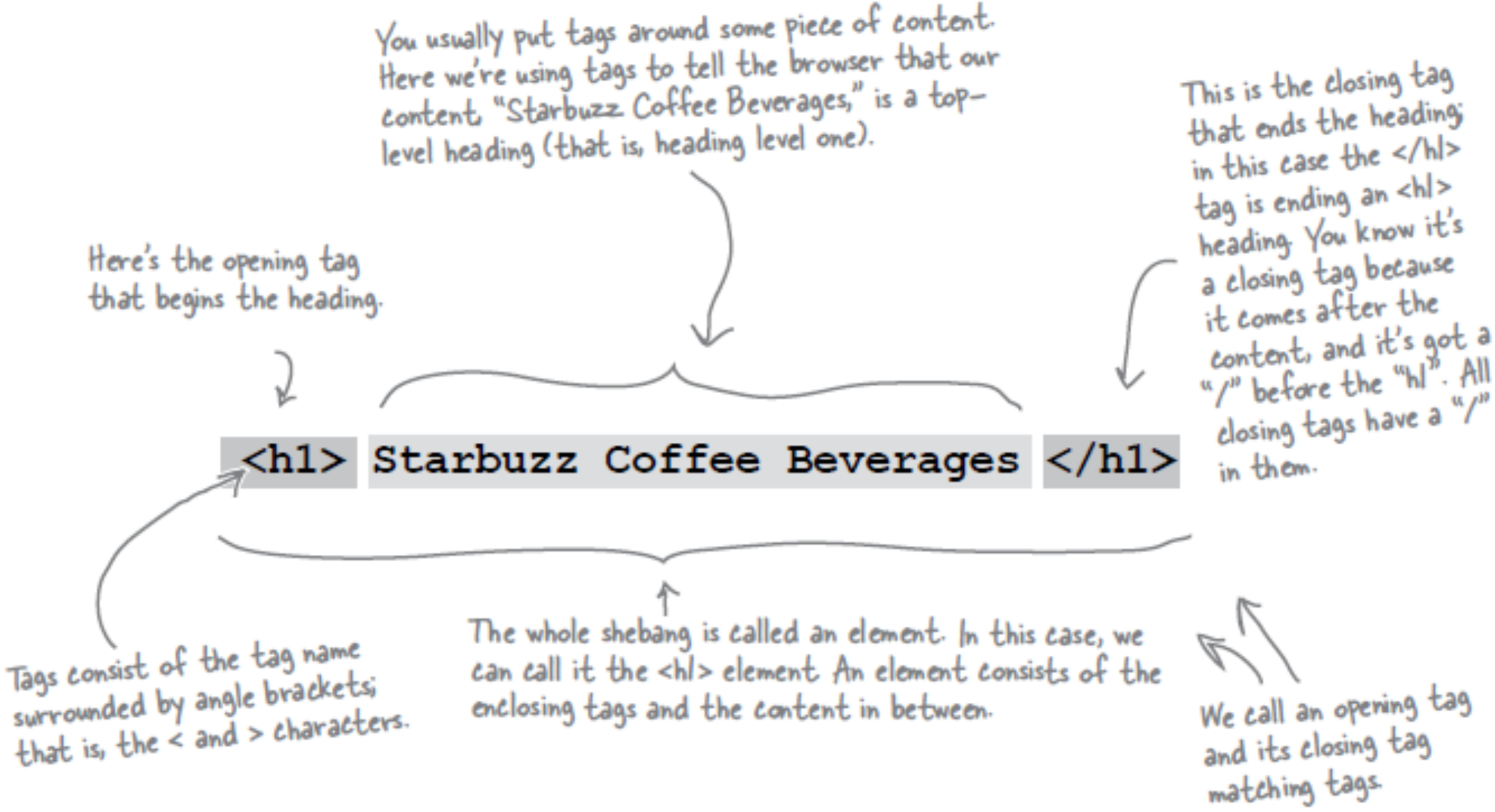
Learning Outcomes

- 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

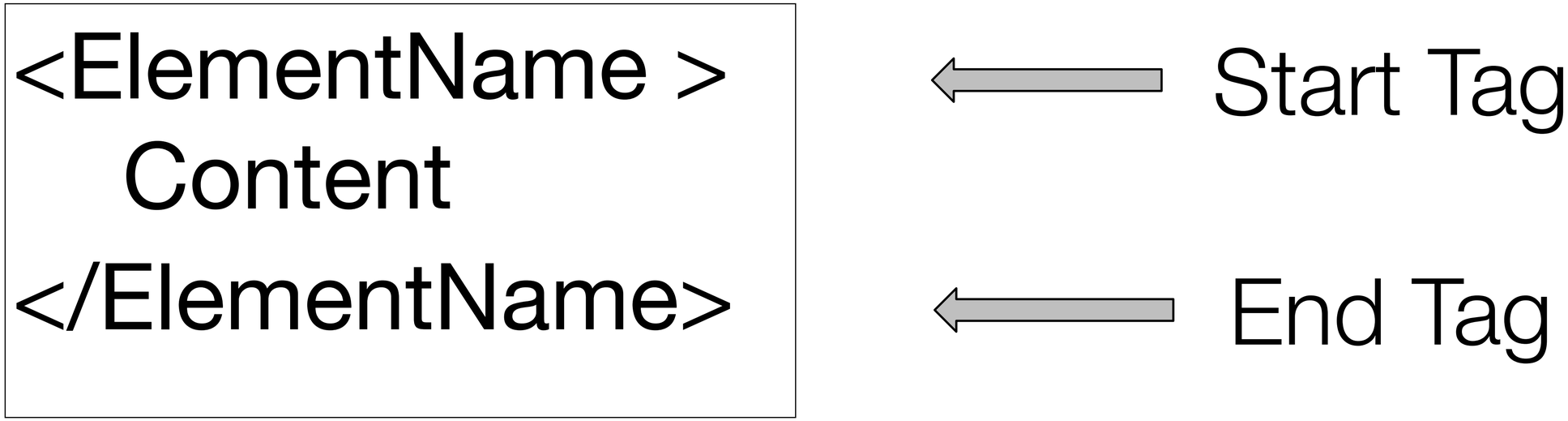
Agenda

- Elements, Attributes, & Documents
- Linking
- Nesting
- Line break, Block & Inline Elements

Components of an HTML Element



Components of an HTML Element



<title>

<title> My App Store </title>

ElementName: *<title>*

Content: *My App Store*

ElementName: *</title>*

```
<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 apps
  from the best publishers, at great
  prices.
</p>
```

ElementName:	* *
Content:	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 apps from the best publishers, at great prices.
ElementName:	* *

<a>

```
<a href="apps.html"> App Store </a>
```

ElementName: *<a>*

AttributeName: `href`

`AttributeValue:` *“apps.html”*

Content: *App Store*

ElementName: **

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

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

<img src="../images/delete.jpg">

HTML Document Structure

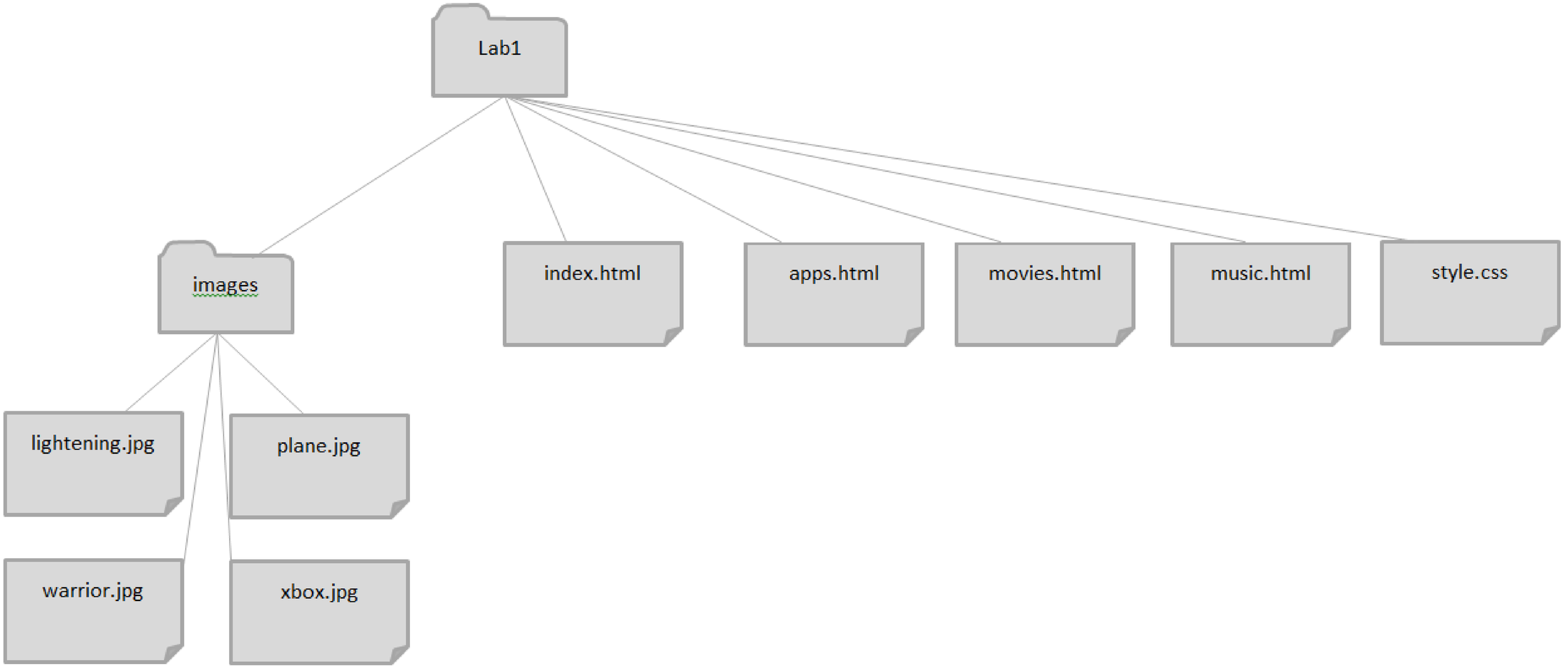
- html
  - head
    - title
  - body
    - h1
    - ol
    - etc...

```
<!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
- Line break, Block & Inline Elements

Linking


Lab1
images
index.html
apps.html
movies.html
music.html
style.css
lightening.jpg
plane.jpg
warrior.jpg
xbox.jpg

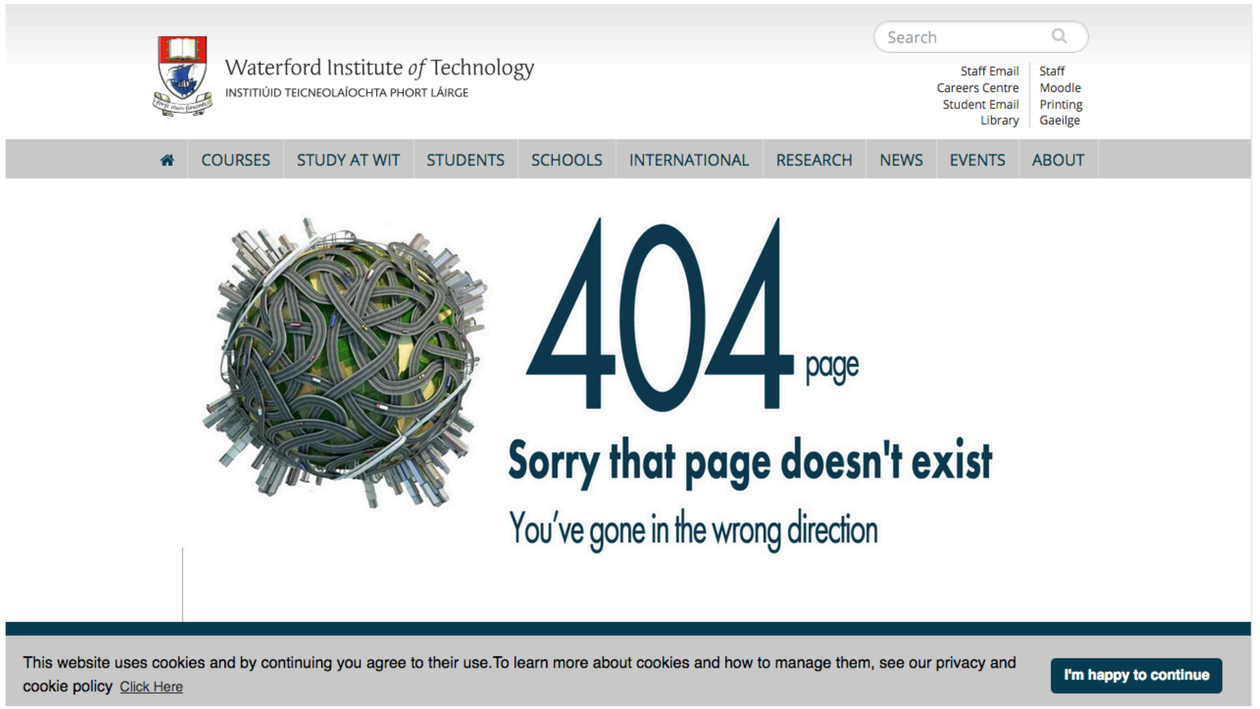
Linking to other pages or images

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:/web-development/lab-01/images/xbox.jpg

  c:/web-development/lab-01/index.html

Relative:

  ./images/xbox.jpg

  ../apps.html

  index.html

You can trace route from “current position” to the destination

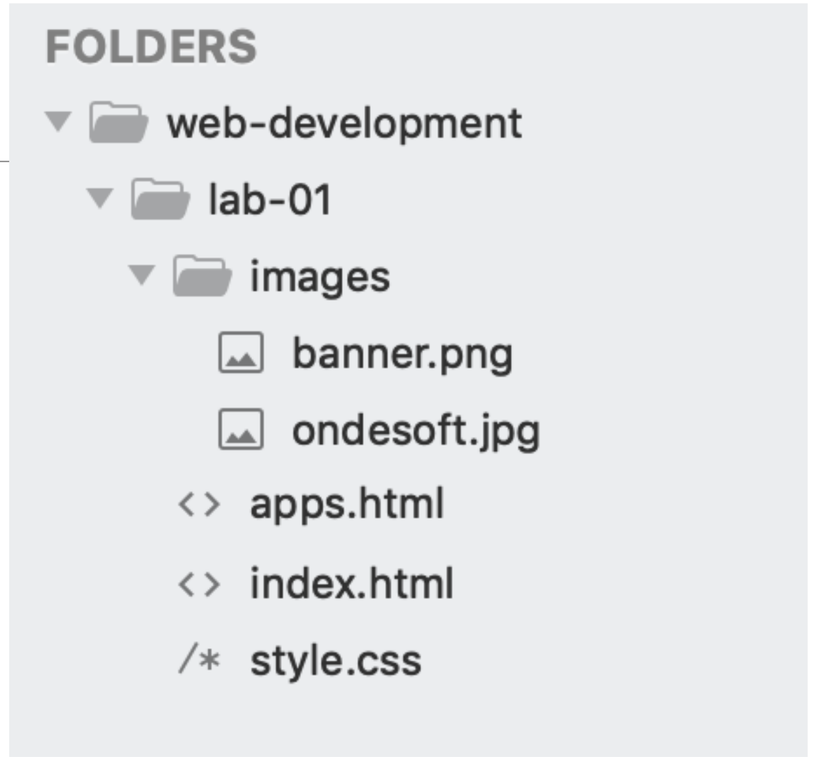
“..” means go up one level

Directory name may prefix filename

Relative Link Examples

- If we are in “lab-01” then “images/banner.png” is a relative link from the current folder (lab-01) to the images folder, and to the file “banner.png” in that folder

- Avoid absolute links!




```
<a href=“apps.html”>Movies</a>
```

```
<img src=“./images/banner.jpg”>
```

```
<img src=“c:/web-development/lab-01/images/banner.jpg”>
```

Agenda

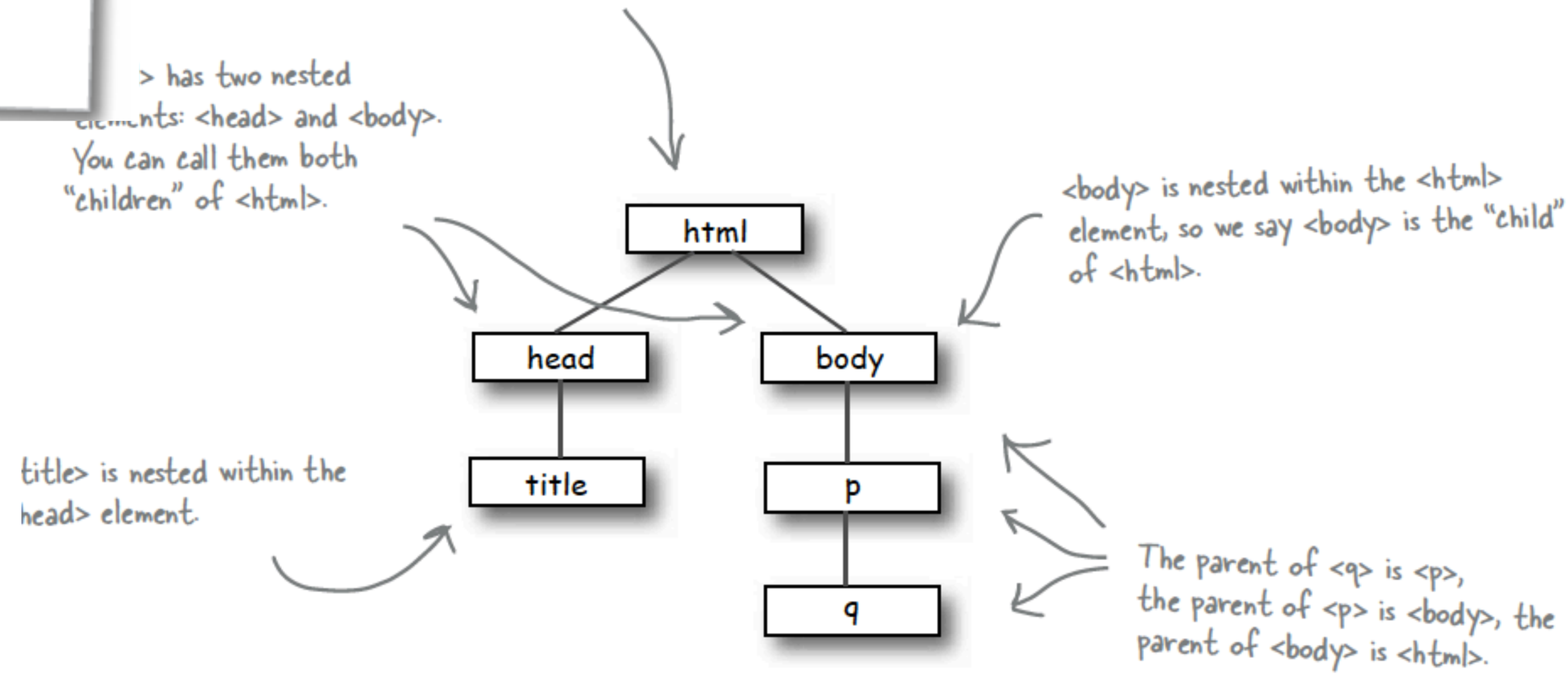
- Elements, Attributes, & Documents
- Linking
- Nesting
- Line break, Block & Inline Elements

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.

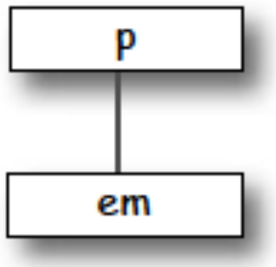
Nesting - Tree Structure

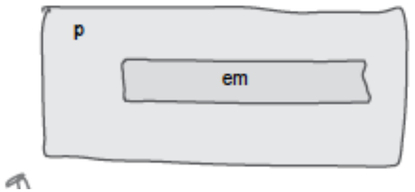
```
<html>
  <head>
    <title>Musings</title>
  </head>
  <body>
    <p>
      To quote Buckaroo,
      <q>The only reason
        for time is so
        that everything
        doesn't happen
        at once.</q>
    </p>
  </body>
</html>
```



Nesting can be Incorrect!

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

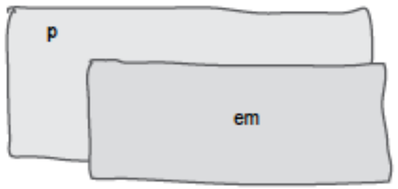




Good

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

?



Bad

Agenda

- Elements, Attributes, & Documents
- Linking
- Nesting
- Line break, Block & Inline Elements

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

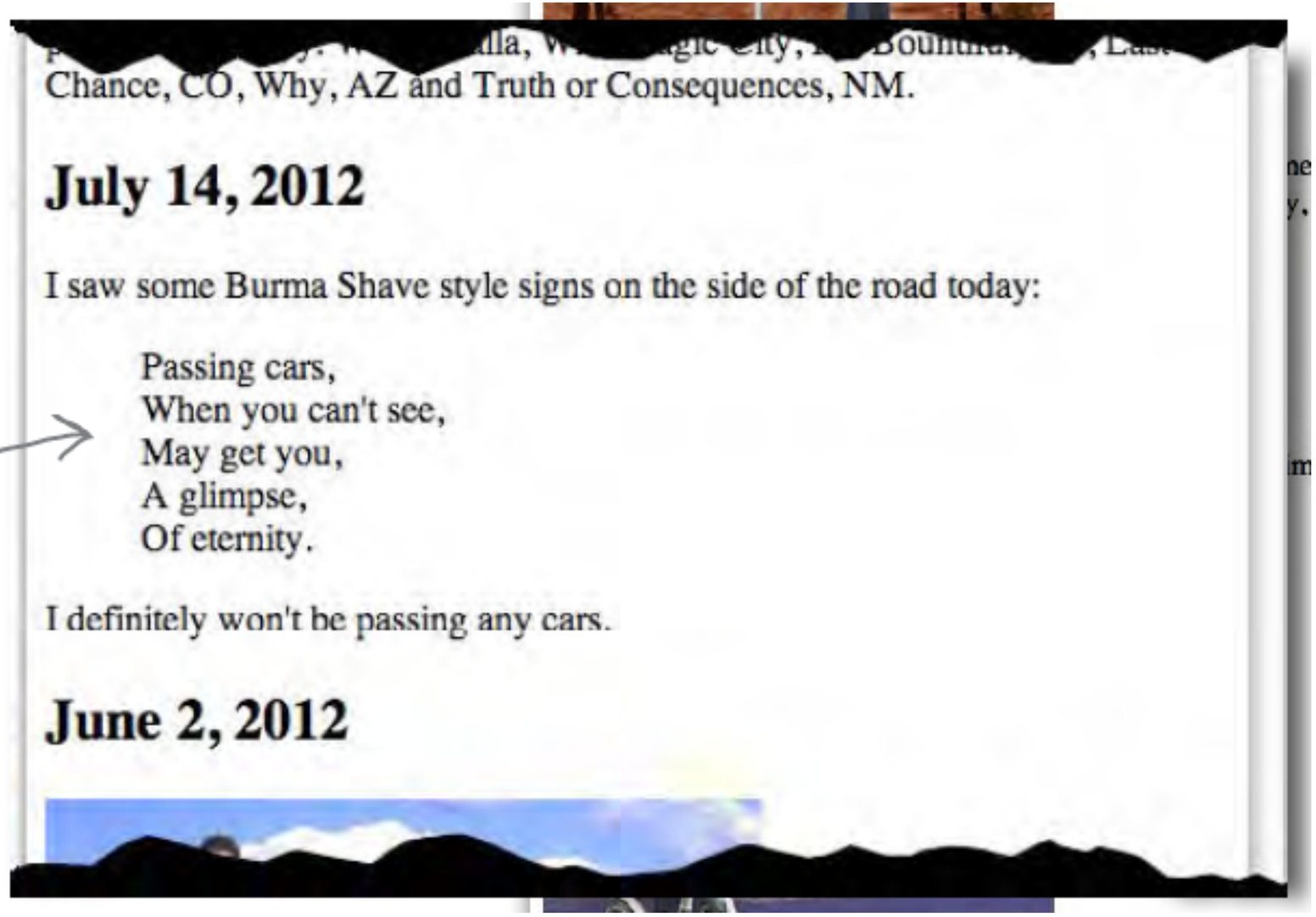
Here's the July 14th snippet from Tony's page.

```
<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>
```

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

Line Breaks

Each line now has a linebreak after it.



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

Line Breaks

- Break:
 </br>
- An Empty element: - sometimes shortened to:
  -
</br>
- or just
  -
- or
  -

Block vs Inline Elements

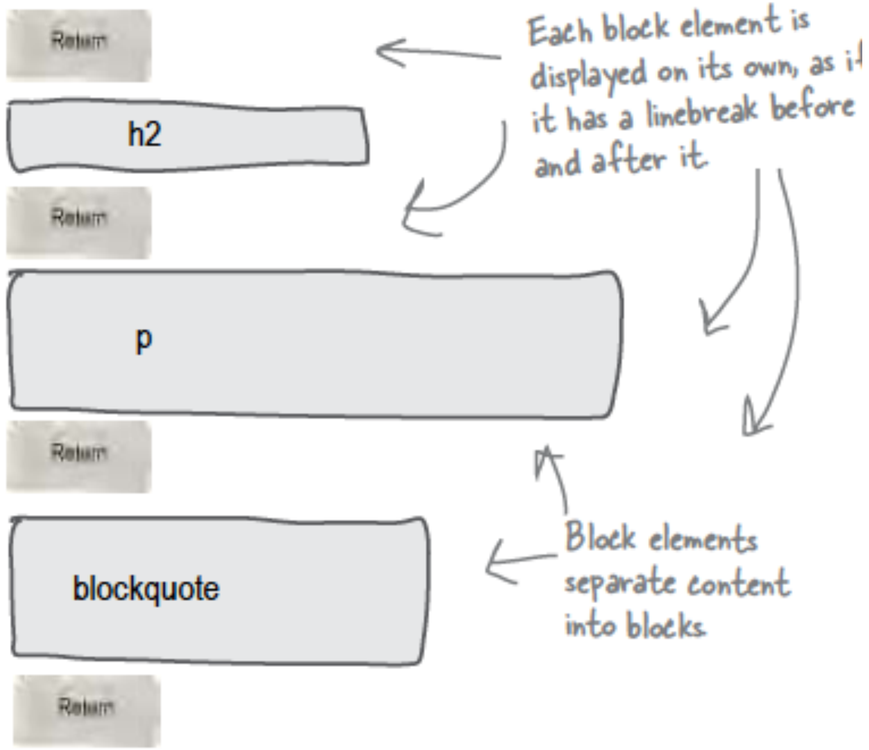
- Block elements are always displayed as if they have a line break before and after them

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

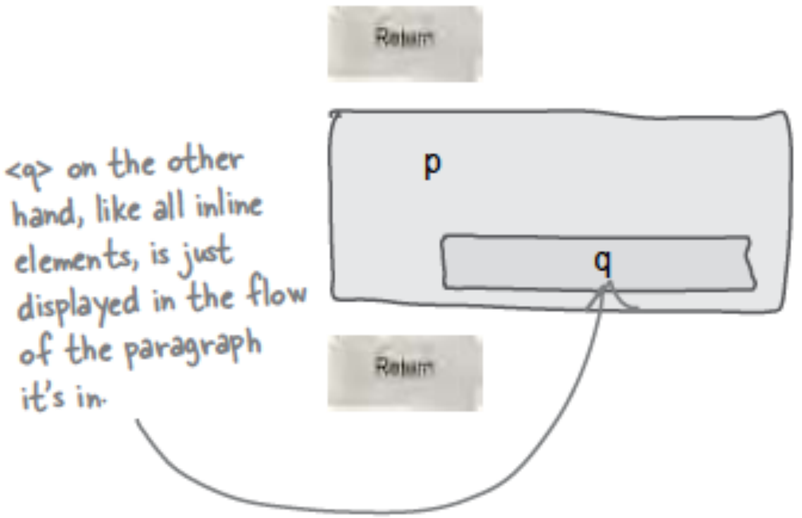
“Block elements stand on their own; inline elements go with the flow.”

Examples

- Block - h1, h2, p, blockquote



- Inline - a, em, q



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

Labs

HTML Fundamentals | Lab-01: HTML Structure | 01 | 02 | 03 | 04 | 05 | Exercise

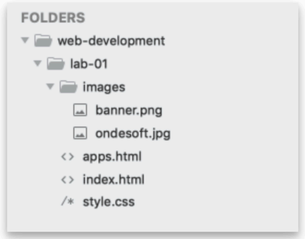
Steps

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, to create another new file called `apps.html`

You should now have the following folder and file structure visible in Sublime:



Instructions for each step

Make sure your sublime matches the above precisely.

Next you will write some html code into the two html files.

Copy and paste in the following code into each of the files:

index.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 apps
      from the best publishers, at great prices.
    </p>
    <p>
      Whether you are interested in gaming or graphics design, software development or media production
      - we have the bundle for you. Each <a href="apps.html">app bundle</a> is designed
      to compliment the others, delivering you an exciting take on a scene.
    </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>
    </ul>
  </body>
</html>
```

apps.html

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Bundle APP Store</title>
  </head>
  <body>
    <img src="images/banner.png">
    <h3>Freebie</h3>
    <p>
      Stacksocial just published its so called Free Ondesoft Mac Tool Bundle, which contains 5 apps
      from Onesoft. The bundle is worth $146 will be probably available only a couple of days so you'd
      better hurry up to get it.
    </p>
    <p>
```

Typical Lab Step:

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

All you need to do is right-click on the "lab-01" folder in sublime, and select "New Folder" from the context menu - name the folder "images"

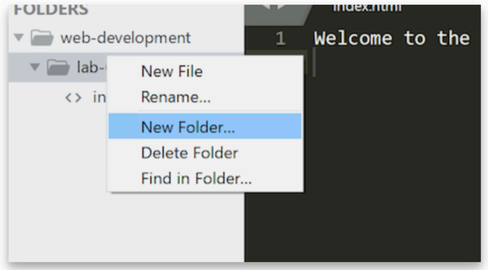
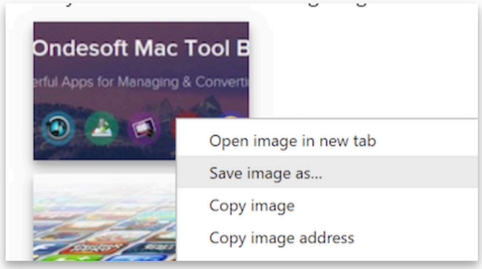


Image Library

Below are two images:

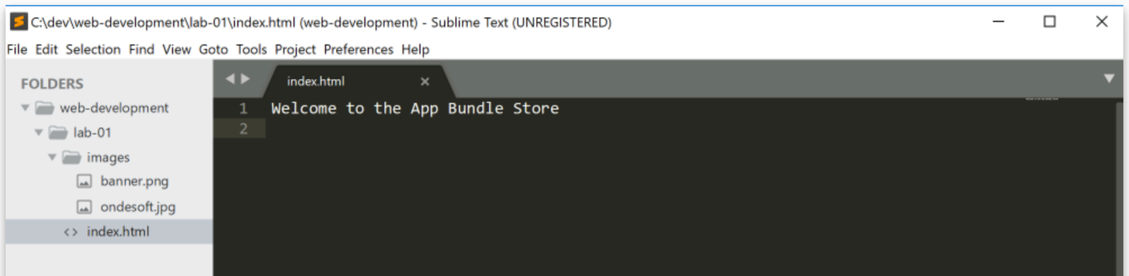
Right click on each one of these now (in the browser you are reading this on) and select "Save Image As".



It brings up a dialog inviting you to locate the folder, and name the file, you are about to save.

Your task now is to locate the "web-development/lab-01/images" folder you have just created, and save the files in there. The files will have a default name, which you can keep.

If all of the above goes as expected, you your sublime window should look like this:



Exercises

- To assist on reflection and independent learning.
- Not essential you attempt all of them (yet)
- Select a few that interest you or that you have time to work on

Exercises

Solution So Far..

This is an archive (zip file) of a solution to the lab so far:

- https://github.com/wit-hdip-comp-sci-2019/bundle-store/releases/tag/lab.01.end

The exercises below can be attempted in any order. You may not have time to complete them all. They are indented to promote some experimentation on your part, and need not be submitted to the course web. If you have having issues - post on the slack "web-development" channel, or contact your tutors directly via Slack Direct Messaging.

Exercise 1: Zip Archive

Download the "Source Code" from the above link. This is a zipped archive of the completed lab. See if you can uncompress (unzip) the archive. This can usually be accomplished just be double clicking it. The unzipped archive will be contained in a folder called "bundle-store". Now, relocate (drag/drop perhaps) this unzipped archive to somewhere else on your workstation. Perhaps consider creating a new folder called "solutions" inside the "web-development" folder, and place the solution in there.

Open the solution project, both in Sublime and a browser.

Exercise 2: Index.html

Just to get used to the editor, create a few more new apps in the 'apps.html', perhaps you can locate content from some web source (including images). For instance here:

- https://mac-bundles.com

Try to identify and replicate the way the code is indented in the existing content.

Exercise 3: New pages

Incorporate content into a new `directions.html` page. It can contain any content you think would be useful. Incorporate a link from index to this page.

Exercise 4: Link the CSS file to all pages

Currently you may not have the CSS file 'linked' to app.html or the `directions.html` page. i.e. only index.html is styled with the css rules you have defined. See if you can link the other pages now.

Exercise 4: HTML Reference

If you are finding initial contact with HTML a little challenging - this is an excellent starter resource to review:

- https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/HTML_basics

It is short read.

Exercise 5: Lists

In this lab you used the element `<ul>` in the index page.

Investigate the use of `<ol>` , find out the differences between these elements here:

- https://developer.mozilla.org/en-US/docs/Learn/CSS/Styling_text/Styling_lists