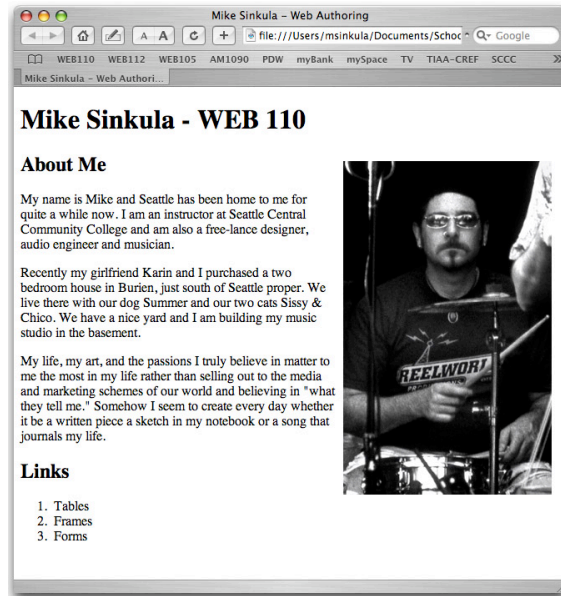**Week Four:** Lists, Links & URLs

Lists

- (X)HTML specifications contain special codes for creating lists of items
- there area few different types of list you can create: *numbered*, *bulleted* and *definition*
- lists can be very useful in calling out important material and navigational items

Ordered Lists

- The first type of list we will take a look at is the ordered list:


```
<!-- Begin Links -->
<h2>Links</h2>
 <ol>
    <li>Tables</li>
    <li>Frames</li>
    <li>Forms</li>
 </ol>
<!-- End Links -->
```
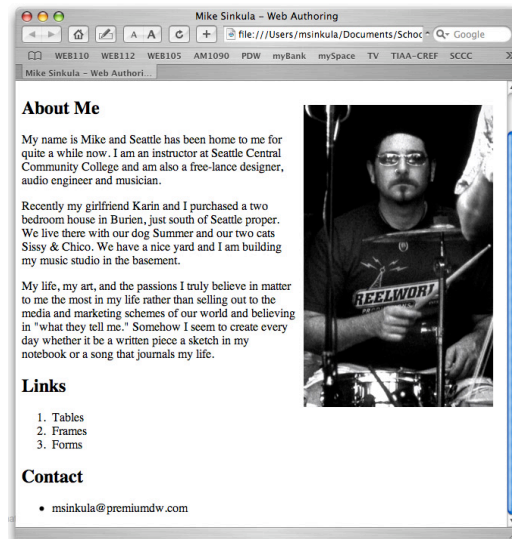
where `<ol>` is the tag for an *ordered* list and `<li>` is the tag for the list *item* inside of the ordered list

- by creating an *ordered* list, we are specifying that out list items will be numbered in order of appearance in the code
- you will also notice that the ordered list appears in the browser with a default indent – eventually you will learn how to control this indent to your liking using style sheets

Unordered Lists

- The next type of list we will look at is the unordered list:

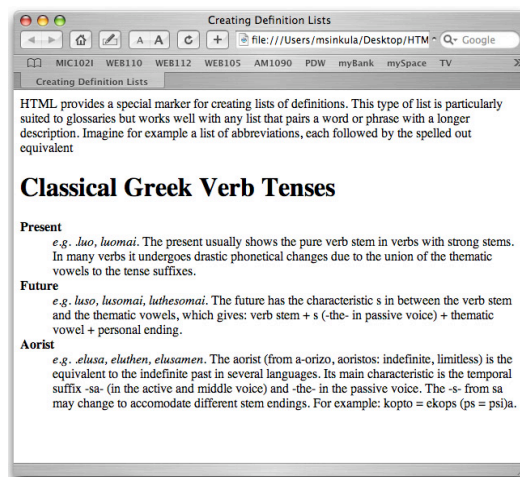```
<!-- Begin Contact -->
<h2>Contact</h2>

    msinkula@premiumdw.com

<!-- End Contact -->
```

where `<ul>` is the tag to create an unordered list and (again) the `<li>` tag is the list item inside the unordered list

- the default type of an unordered list is to create a solid round shaped bullet for the list items – eventually you will see how to change the appearance of, hide or create your own image for the bullet using style sheets

**Definition Lists**

- We will not be using definition lists in this part of our assignments, but let's take a look anyhow:

```
<dl>
  <dt>Present</dt>
     <dd>e.g. .luo, luomai...</dd>
  <dt>Future</dt>
     <dd>e.g. luso, lusomai...</dd>
  <dt>Aorist</dt>
     <dd>e.g. .elusa, eluthen, elusamen...</dd>
</dl>
```
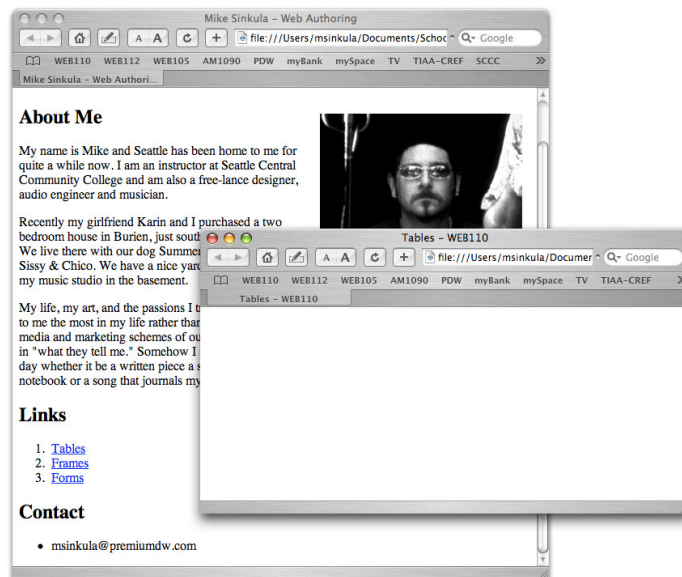
Where `<dl>` starts off our definition list, `<dt>` gets used as the tag to present the definition term and `<dl>` is that tag that gets used for our definition list item

## Links & URLs

- Links are the distinguishing feature of the web
- Links allow you to navigate from page to page, view pictures, open or download a media file, etc.
- **Links** have three fundamental parts to make them work:

    1. **Label** – the actual text of graphic item the user clicks on to activate the link
    2. **Destination** – the specified place that the user ends up when they activate the link
    3. **Target** – how the destination will be displayed in the browser

### Relative Page Links

- For our list items that we have created under our links headline we will be using *links* via an ANCHOR that will open these pages in a new browser window:



```
<!-- Begin Links -->
<h2>Links</h2>
<ol>
     <li><a href="tables.html" target="_blank">Tables</a></li>
     <li><a href="frameset.html" target="_blank">Frames</a></li>
     <li><a href="form.html" target="_blank">Forms</a></li>
</ol>
<!-- End Links -->
```

where we use the `<a>` tag to create the (relative) *link* and use the `href` *attribute* with the *value* of `page.html` to specify the *destination* of the link

the *destination* of `page1.html` will also open in a new browser window because we have used the `target` *attribute* with the *value* of `_blank`

NOTE: in this case I have opened a new page without content

Absolute Page Links

- We will not be using absolute page links in this part of our assignments, but let's take a look anyhow:

```
<!-- Begin Resource Links -->
<ul>
    <li><a href="http://www.cookwood.com/html5ed/examples/"
    target="_blank">HTML for the World Wide Web &gt;&gt;</a>
    </li>
</ul>
<!-- End Resource Links -->
```

where we still use the `<a>` tag to create the (absolute) *link* and use the `href` *attribute*, but, with the *value* of `http://www.doamain.com/folder/page.html` to specify the *destination* of the *link* to a page that sits on another server out there in cyber space

Specific Anchor Links

- Sometimes you will need to send a link to a spot on the page you are on, especially if the page has quite a bit of content
- For example, on the page below there are several text links near the top of the page that when clicked go to a specific paragraph on the page:


```
<p>Read our <a href="#corporations">Appeal to Corporations and
Investors</a> for more information.</p>
```
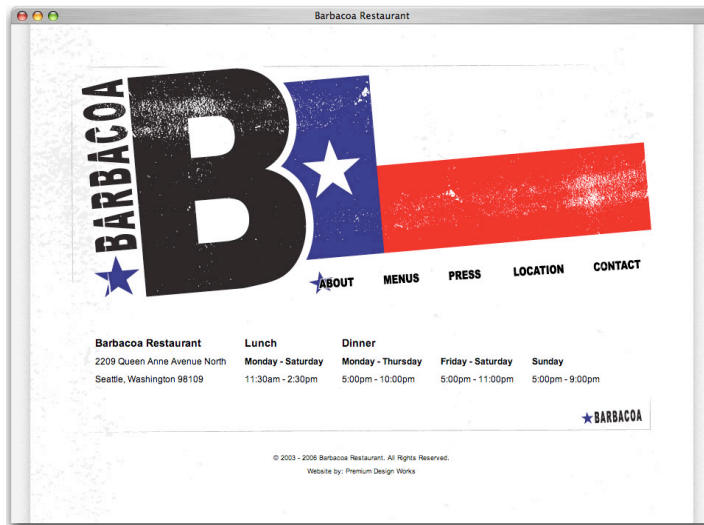
Where inside the `<a>` tag we need to specify the `href` value using the `#` sign and a name – in this case `#corporations`

```
<h1><a name="corporations"></a>Appeal to Corporations and Investors</h1>
```

once the `<a>` tag is clicked the page will scroll down to the `name` attribute with the value of `corporations`

Image Links

- Sometimes you will also want to use an image as a link instead of using text – this is useful in a menu that you want to use a specific typeface:


```
<li><a href="about.php"><img name="texHeader_03"
src="images/texHeader_03.gif" border="0" alt="About"></a></li>
```
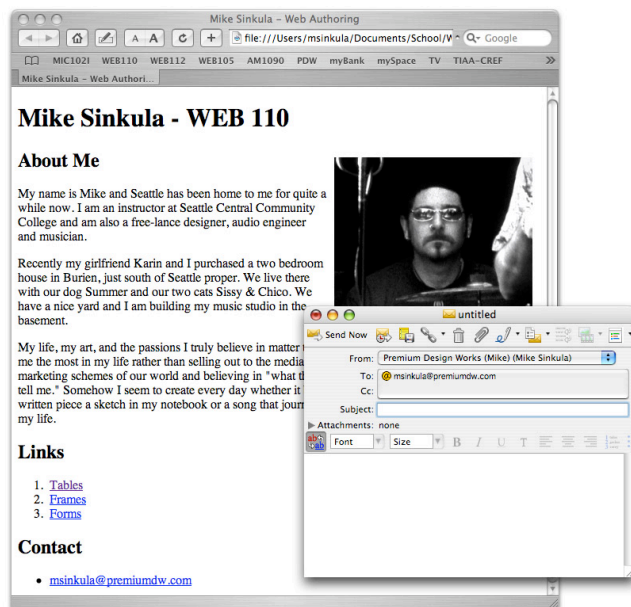
where the `<img>` tag is inside of the `<a>` tag and, just like a text link, we use the `href` *attribute* with the *value* of `about.php`

Email Links

- Instead of using a form to send an email via a web page, it is also possible to simply create a link that will open up in an email application program:

```
<!-- Begin Contact -->
<h2>Contact</h2>
<ul>
<li><a href="mailto:msinkula@premiumdw.com">msinkula@premiumdw.com</a></li>
</ul>
<!-- End Contact -->
```

where we are using a similar text link with the `<a>` tag, but in this case the `href` *attribute* gets a *value* written with a `mailto:` appendix just before the email address of `msinkula@premiumdw.com`

Assignment: "Page" Links
Reading: Chapters 13 & 7

```
<!-- Begin Contact -->
<h2>Contact</h2>
<li><a href="mailto:msinkula@premiumdw.com">msinkula@premiumdw.com
```