

Introduction to HTML

Max Design - Hands-on CSS

Lesson

TWO

Max Design - Hands-on CSS

What will we
cover in this
lesson?

Lesson 2: Content markup

 element

 element

<i> element

<a> element

File paths

 element

 element

 element

<dl> element

<abbr> element

<hr> element

<q> element

<blockquote> element

<address> element

<small> element

<sub> element

<sup> element

Open the
exercise folder

Open the folder called
“start” and then open the file
called “**lesson02.htm**” using
some sort of HTML editor.

Step 1:

`` element

The `` element
represents content that
requires **importance**.

Exercise

Add a `` element...

A code editor window with a light gray title bar and a vertical scrollbar on the right. The text is displayed in a monospaced font.

```
<p>
```

```
    Some strong text.
```

```
</p>
```

A second code editor window, identical in style to the first, showing a different code snippet.

```
<p>
```

```
    Some <strong>strong</strong> text.
```

```
</p>
```

Step 2:

`` element

The `` element represents content that requires **emphasis**.

Exercise

Add an `` element...

A code editor window with a light gray title bar and a vertical scrollbar on the right. The text is displayed in a monospaced font.

```
<p>
```

```
    Some emphasis text.
```

```
</p>
```

A second code editor window, identical in style to the first, showing a different code snippet.

```
<p>
```

```
    Some <em>emphasis</em> text.
```

```
</p>
```

Step 3:

<i> element

The <i> element represents a segment of text that is **typically italicised** (taxonomic terms, technical terms, language phrases etc) but may not need to be emphasised.

Exercise

Add an `<i>` element...

<p>

Common wombat(Vombatus ursinus)</i>

</p>

<p>

Common wombat <i>(Vombatus ursinus)</i>

</p>

Step 4:

<a> element

The **<a> element** is used to create hyperlinks.

```
<p>  
  A <a>link</a> .  
</p>
```

In order for the `<a>` element to become a link we need to add the **HREF attribute**.

```
<p>  
  A <a href="add url here">link</a>.  
</p>
```

The value of the HREF attribute **needs to be a URL** - a web address. We will cover this in more detail with “File paths” soon.

```
<p>  
  A <a href="add url here">link</a>.  
</p>
```

The `<a>` element can wrap around a single word, a line of text or a whole section of text. It can even be used to **wrap around images.**

```
<a href="url here">word</a>  
<a href="url here">... text here ...</a>  
<a href="url here">... image here ...</a>
```

Any content that is placed inside an `<a>` element that contains an HREF **will become “clickable”** - the user can click on this content to activate the link.

Opening a new
window

We can force a link to open a new window by adding a **TARGET attribute**. The value needs to be set to “_blank”.

```
<a href="url" target="_blank">a link to  
open a new window</a>
```

This is not considered a good practice. Users **should be allowed to decide for themselves** whether they want a link to open a new window or not.

Exercise

Let's add some links...

```
<p>
```

```
    A link.
```

```
</p>
```

```
<p>
```

```
    A link to open a new window.
```

```
</p>
```

```
<p>
```

```
    A <a href="http://google.com">link</a>.
```

```
</p>
```

```
<p>
```

```
    A link to <a href="http://google.com"
target="_blank" >open a new window</a>.
```

```
</p>
```

Note:

File paths

There are times when you want to link to a specific web page within your site or another site. **This is when file paths come into play.**



```
<a href="url here">Link</a>
```

1. Absolute paths

File paths can be **absolute**, which means they point to a full web address or files/folders within this address.

```
<a href="http://site.com/">Link</a>  
<a href="http://site.com/help.htm">Link</a>
```

2. Relative paths

File paths can be **relative**, which means they point to files or folders that are located “relative to the current document”.

The link below will look for a file called “help.htm” **in the same directory** as our document.

A screenshot of a code editor window with a light gray border and a white background. The window contains a single line of HTML code: `Link`. The text is color-coded: the opening tag `<a` is purple, the attribute `href="help.htm"` is green, and the closing tag `` is gray. The word `Link` is in a standard gray font. On the right side of the window, there is a vertical scrollbar with a blue handle and a small upward-pointing arrow at the top.

```
<a href="help.htm">Link</a>
```



parent folder



help.htm



our-document.htm

The link below will look for a file called “help.htm” **in one directory above** our document.

A screenshot of a code editor window with a light gray border and a white background. The window contains a single line of HTML code: `Link`. The text is color-coded: the opening tag `<a` is purple, the attribute `href="` is green, the path `../help.htm"` is green, and the closing tag `>Link` is gray. The window has a standard macOS-style title bar with a red, yellow, and green button on the right side.

```
<a href=" ../help.htm">Link</a>
```



help.htm



parent folder



our-document.htm

The link below will look for a file called “help.htm” **in a folder called “about”** which is in the same directory as our document.

A screenshot of a code editor window with a light gray title bar and a vertical scrollbar on the right. The editor contains a single line of HTML code: `Link`. The text is color-coded: the opening tag `<a` is purple, the attribute `href="about/help.htm"` is green, and the closing tag `>Link` is gray.

```
<a href="about/help.htm">Link</a>
```




parent folder



our-document.htm



about



help.htm

3. Site relative paths

File paths can be **relative to the root of the site**. They begin with a “/” which indicate that the path starts at the site root.

The link below will look for a file called “help.htm” **in the root of the site.**

A browser window with a grey title bar and a vertical scrollbar on the right. The main content area is white and displays an HTML anchor tag. The text is color-coded: the opening tag <a is grey, href= is purple, the path /help.htm is green, and the closing tag >Link is grey.

```
<a href="/help.htm">Link</a>
```



site root



help.htm



another folder



another folder



our-document.htm

4. fragment identifiers

File paths **can point to a specific place in a web page**, rather than the whole page itself.

These types of links are called “**fragment identifiers**” as they identify and target fragments of a web document.

We can do this in **two simple steps.**

1. identify the fragment
2. link to this fragment

Step 1: The fragment needs to be identified with an ID value. This can be placed on any relevant element.

```
<div id="page-fragment">  
    Some content  
</div>
```

Step 2: The HREF inside the `<a>` element must point to this fragment.

This can be achieved using the “#” **symbol** followed by the value of the ID.

```
<a href="#page-fragment">Link to page  
fragment</a>
```

... other content

```
<div id="page-fragment">  
    Some content  
</div>
```

top of the page

You can also use a fragment identifier to **link to the top of the current page.**

This can be achieved using the “#” **symbol on its own** for the HREF value.

A browser window with a light gray title bar and a vertical scrollbar on the right. The main content area is white and displays an HTML anchor tag. The text is color-coded: the opening tag <a is purple, href=" is purple, # is green, > is blue, Top of the page is black, is blue, and the closing tag > is purple.

```
<a href="#">Top of the page</a>
```

Exercise

Let's add some more links...


```
<p>
    A fragment link
</p>
<p>
    Page top
</p>
```

```
<p>
    A <a href="#intro">fragment link</a>
</p>
<p>
    <a href="#">Page top</a>
</p>
```

Step 5:

 element

To add an image, we use the ** element**. This element is a void element - so it has no closing tag.

A screenshot of a web browser window. The address bar is empty. The main content area displays the code in a blue, monospaced font. The browser's interface includes a title bar at the top, a scrollbar on the right, and a status bar at the bottom.

```
<img>
```

We will need to add **two attributes** to this element.

The first attribute is the SRC attribute - which tells the browser **where to find the image.**

```

```

The **file path rules** apply for SRC attribute - just like for the HREF attribute.

```
  
  
  
  

```

The second **attribute is the ALT attribute** - which provides a text alternative for devices that cannot load the image.

```

```

You can also add **WIDTH**
and HEIGHT attributes to
your images.

```

```


Exercise

Let's add an image...

```
<p>  
    [image here]  
</p>
```

```
<p>  
      
</p>
```

Step 6:

 element

The element represents an “**unordered list**” of **items**, where the order of the items is not important.

The element can contain one or more or **“list” elements.**

```
<ul>  
  <li>This is list item 1</li>  
  <li>This is list item 2</li>  
</ul>
```

Exercise

Let's add an unordered list...

Unordered list item 1
Unordered list item 2
Unordered list item 3
Unordered list item 4

```
<ul>  
  <li>Unordered list item 1</li>  
  <li>Unordered list item 2</li>  
  <li>Unordered list item 3</li>  
  <li>Unordered list item 4</li>  
</ul>
```

Step 7:

 element

The element represents an **“ordered list” of items**, where the items have been intentionally ordered.

Just like the ``, the `` element can contain one or more **`` elements**.

```
<ol>  
  <li>This is list item 1</li>  
  <li>This is list item 2</li>  
</ol>
```

Exercise

Let's add an ordered list...

```
Ordered list item 1  
Ordered list item 2  
Ordered list item 3  
Ordered list item 4
```

```
<ol>  
  <li>Ordered list item 1</li>  
  <li>Ordered list item 2</li>  
  <li>Ordered list item 3</li>  
  <li>Ordered list item 4</li>  
</ol>
```

Step 8:

<dl> element

The <dl> represents a
“**description list**” consisting
of one or more name/value
groups.

Each group consists of one or more **names** (<dt>) and one or more **values** (<dd>).

```
<dl>
  <dt>This is name 1</dt>
    <dd>This is value 1</dd>
  <dt>This is name 2</dt>
    <dd>This is value 2</dd>
  <dt>This is name 3</dt>
</dl>
```

You can have **more than one name elements** (<dt>) assigned to one value element (<dd>)

```
<dl>
  <dt>This is name 1</dt>
  <dt>This is name 2</dt>
    <dd>This is value 1</dd>
</dl>
```


You can have **more than one value elements** (<dd>) assigned to one name element (<dt>)

```
<dl>
  <dt>This is name 1</dt>
    <dd>This is value 1</dd>
    <dd>This is value 2</dd>
</dl>
```

Exercise

Let's add a description list...

```
This is name 1  
This is value 1  
This is name 2  
This is value 2
```

```
<dl>  
  <dt>This is name 1</dt>  
    <dd>This is value 1</dd>  
  <dt>This is name 2</dt>  
    <dd>This is value 2</dd>  
</dl>
```

Step 9:

<abbr> element

The `<abbr>` element
represents an **abbreviation
or acronym**.

```
<p>  
  Learn <abbr>HTML</abbr> .  
</p>
```

The **title attribute** can be used to provide the expanded information, displayed as a tool tip.

```
<p>  
    Learn <abbr title="Hypertext Markup  
Language">HTML</abbr>.  
</p>
```

Exercise

Let's add an abbreviation...

```
<p>
```

```
    Learn HTML.
```

```
</p>
```

```
<p>
```

```
    Learn <abbr title="Hypertext Markup  
Language">HTML</abbr>.
```

```
</p>
```


Step 10:

`<hr>` element

The `<hr>` element represents a **block-level thematic break**. This element is a void element - so it has no closing tag.

A screenshot of a code editor window. The window has a light gray title bar and a vertical scrollbar on the right side. The text '<hr>' is displayed in the editor area in a blue monospace font.

```
<hr>
```

Exercise

Let's add a horizontal rule...

<hr>

Step 11:

<q> element

The `<q>` element represents some **phrasing content** quoted from another source.

A code editor window with a light gray border and a white background. It contains HTML code in blue text. The code is:

```
<p>  
    He said: <q>make my day</q>.  
</p>
```

```
<p>  
    He said: <q>make my day</q>.  
</p>
```

Exercise

Let's add a quote...

<p>

He said: Make my day.

</p>

<p>

He said: <q>Make my day</q>.

</p>

Step 12:

<blockquote>
element

The `<blockquote>` element represents a **section of content** that is quoted from another source.

```
<blockquote>  
  <p>A long quote</p>  
</blockquote>
```

Exercise

Let's add a blockquote...

A code editor window with a light gray title bar and a vertical scrollbar on the right. The text is in a monospaced font.

```
<p>
```

```
    Long quotation here.
```

```
</p>
```

A second code editor window, identical in style to the first, showing nested HTML tags.

```
<blockquote>
```

```
    <p>
```

```
        Long quotation here.
```

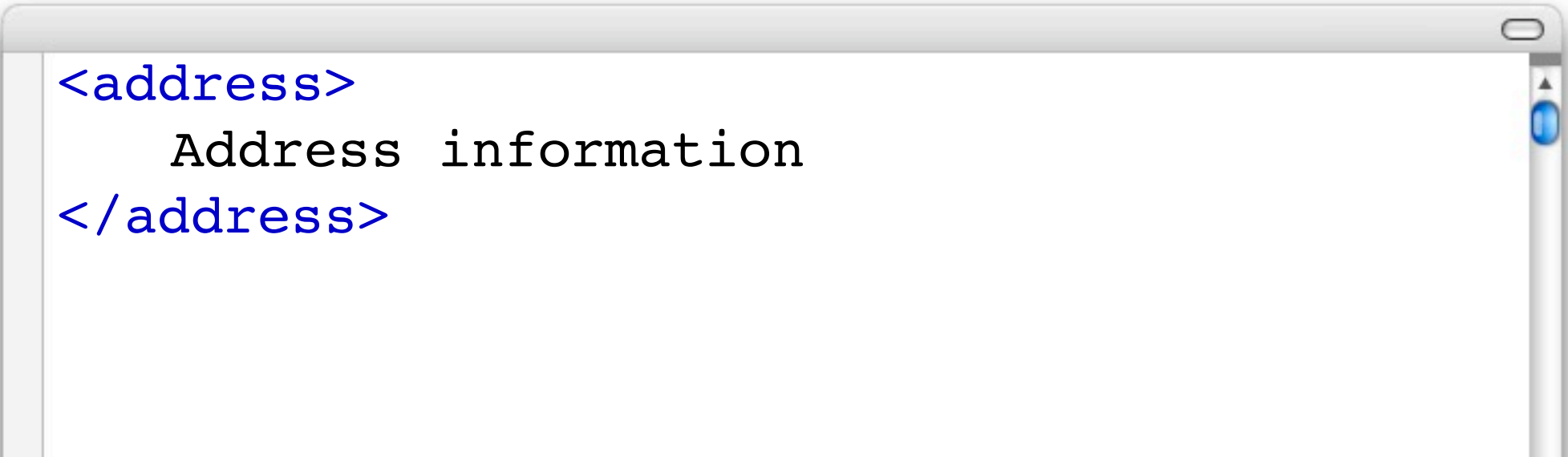
```
    </p>
```

```
</blockquote>
```

Step 13:

<address> element

The address element is for **contact information for a document** or part of a document.



```
<address>  
    Address information  
</address>
```

The <address> element **must not be used for general address information**, unless those addresses are in fact the relevant contact information for the document.

Exercise

Let's add an address...

This page was created by John Smith

Email: john@me.com

<address>
This page was created by John Smith

Email: john@me.com </
address>

Step 14:

<small> element

The <small> element represents “**small print**” such as legal disclaimers and caveats.

```
<p>  
    <small>Copyright Site Name 2012</small>  
</p>
```

The <small> element is **not used for presentational purposes** - ie. to make the text look smaller.

Exercise

Let's add a small element...

A code editor window with a light gray title bar and a vertical scrollbar on the right. The text is in a monospaced font.

```
<p>
```

```
    Copyright Site Name 2012
```

```
</p>
```

A code editor window with a light gray title bar and a vertical scrollbar on the right. The text is in a monospaced font.

```
<p>
```

```
    <small>Copyright Site Name 2012</small>
```

```
</p>
```

Step 15:

<sub> element

The `<sub>` element is used for information that is a **subscript**.

```
<p>
```

```
    Drink more H<sub>2</sub>O.
```

```
</p>
```


Exercise

Let's add a subscript...

<p>

Drink more H₂O.

</p>

<p>

Drink more H₂O.

</p>

Step 16:

<sup> element

The `<sup>` element is used for information that is a **superscript**.

```
<p>  
  A footnote<sup>[ 1 ]</sup>  
</p>
```

Exercise

Let's add a superscript...

<p>

A footnote[1]

</p>

<p>

A footnote^[1]

</p>



Russ Weakley

Max Design

Site: maxdesign.com.au

Twitter: twitter.com/russmaxdesign

Slideshare: slideshare.net/maxdesign

Linkedin: linkedin.com/in/russweakley