

1

STRUCTURE

- ▶ Understanding structure
- ▶ Learning about markup
- ▶ Tags and elements

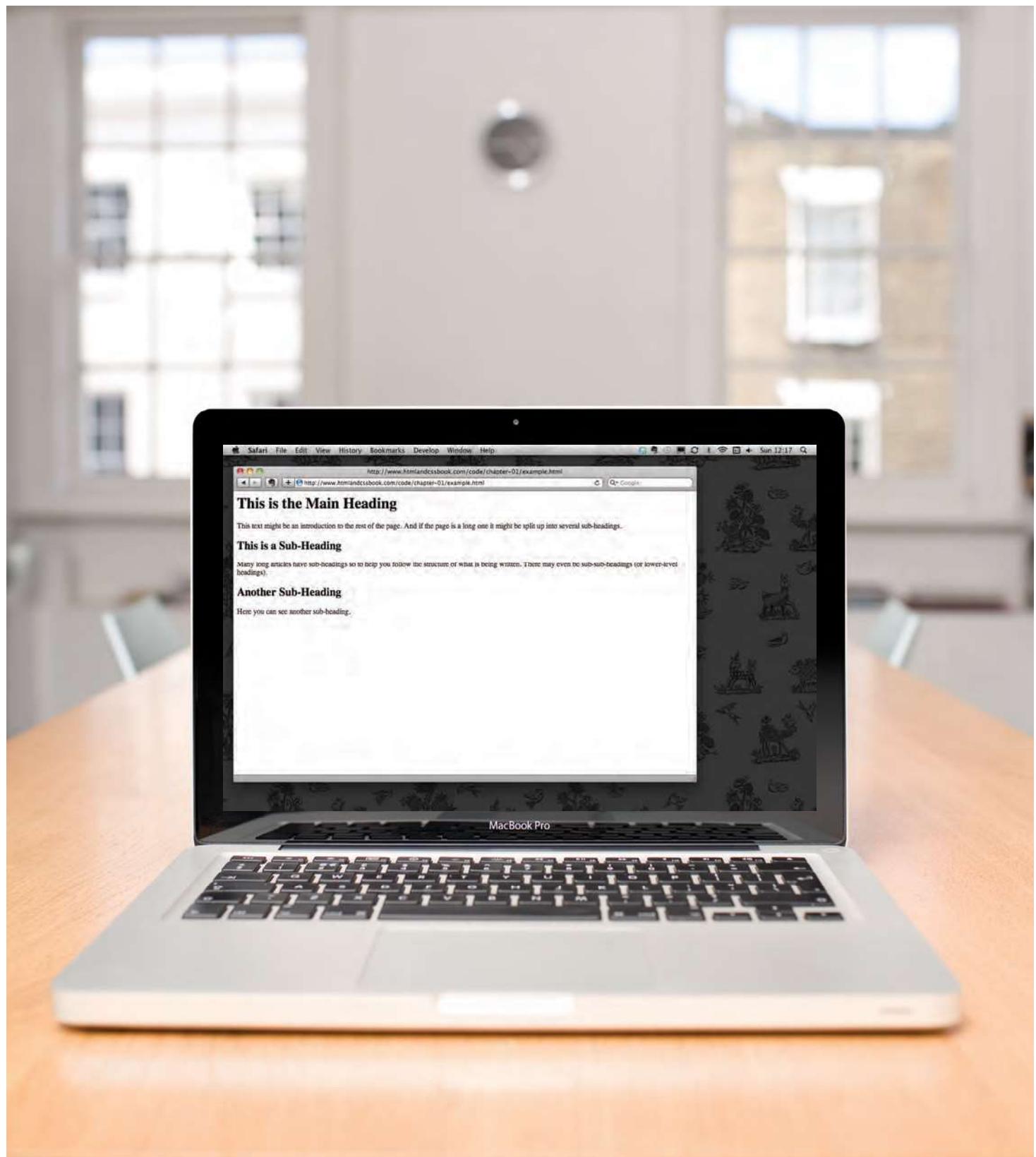


We come across all kinds of documents every day of our lives. Newspapers, insurance forms, shop catalogues... the list goes on.

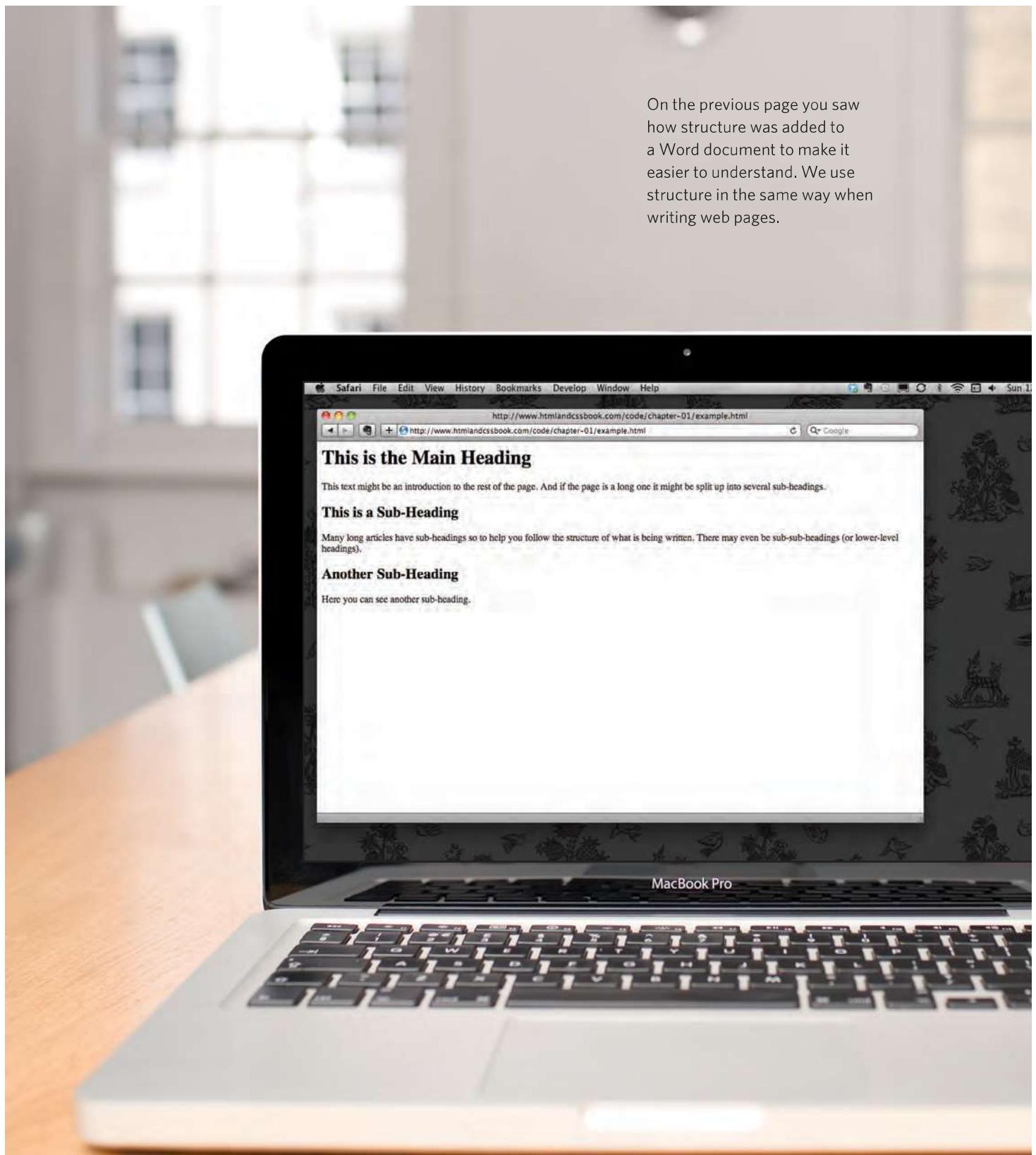
Many web pages act like electronic versions of these documents. For example, newspapers show the same stories in print as they do on websites; you can apply for insurance over the web; and stores have online catalogs and e-commerce facilities.

In all kinds of documents, structure is very important in helping readers to understand the messages you are trying to convey and to navigate around the document. So, in order to learn how to write web pages, it is very important to understand how to structure documents. In this chapter you will:

- See how HTML describes the structure of a web page
- Learn how tags or elements are added to your document
- Write your first web page



On the previous page you saw how structure was added to a Word document to make it easier to understand. We use structure in the same way when writing web pages.





HTML DESCRIBES THE STRUCTURE OF PAGES

In the browser window you can see a web page that features exactly the same content as the Word document you met on the page 18. To describe the structure of a web page, we add code to the words we want to appear on the page.

You can see the HTML code for this page below. Don't worry about what the code means yet. We start to look at it in more detail on the next page. Note that the HTML code is in blue, and the text you see on screen is in black.

```
<html>
  <body>
    <h1>This is the Main Heading</h1>
    <p>This text might be an introduction to the rest of
       the page. And if the page is a long one it might
       be split up into several sub-headings.<p>
    <h2>This is a Sub-Heading</h2>
    <p>Many long articles have sub-headings so to help
       you follow the structure of what is being written.
       There may even be sub-sub-headings (or lower-level
       headings).</p>
    <h2>Another Sub-Heading</h2>
    <p>Here you can see another sub-heading.</p>
  </body>
</html>
```

The HTML code (in blue) is made up of characters that live inside angled brackets — these are called HTML **elements**. Elements are usually made up of two **tags**: an opening tag and a closing tag. (The closing tag has an extra forward slash in it.) Each HTML element tells the browser something about the information that sits between its opening and closing tags.

HTML USES ELEMENTS TO DESCRIBE THE STRUCTURE OF PAGES

Let's look closer at the code from the last page.
There are several different elements. Each element has an opening tag and a closing tag.

CODE

The diagram illustrates the structure of an HTML document by highlighting nested elements with colored borders. A green border surrounds the entire document, starting with the opening tag <html>. Inside it, an orange border encloses the <body> tag. The content within the <body> tag is contained within a grey border. This grey border contains several nested elements: an <h1> tag, a <p> tag, another <h2> tag, a <p> tag, and a final <h2> tag. Each of these nested elements is also enclosed in its own grey border, showing the recursive nature of HTML structure. The code itself is presented in a monospaced font within these colored boxes.

```
<html>
  <body>
    <h1>This is the Main Heading</h1>
    <p>This text might be an introduction to the rest of the page. And if the page is a long one it might be split up into several sub-headings.</p>
    <h2>This is a Sub-Heading</h2>
    <p>Many long articles have sub-headings so to help you follow the structure of what is being written. There may even be sub-sub-headings (or lower-level headings).</p>
    <h2>Another Sub-Heading</h2>
    <p>Here you can see another sub-heading.</p>
  </body>
</html>
```

Tags act like containers. They tell you something about the information that lies between their opening and closing tags.

DESCRIPTION

The opening `<html>` tag indicates that anything between it and a closing `</html>` tag is HTML code.

The `<body>` tag indicates that anything between it and the closing `</body>` tag should be shown inside the main browser window.

Words between `<h1>` and `</h1>` are a main heading.

A paragraph of text appears between these `<p>` and `</p>` tags.

Words between `<h2>` and `</h2>` form a sub-heading.

Here is another paragraph between opening `<p>` and closing `</p>` tags.

Another sub-heading inside `<h2>` and `</h2>` tags.

Another paragraph inside `<p>` and `</p>` tags.

The closing `</body>` tag indicates the end of what should appear in the main browser window.

The closing `</html>` tag indicates that it is the end of the HTML code.

A CLOSER LOOK AT TAGS



The characters in the brackets indicate the tag's purpose.

For example, in the tags above the p stands for paragraph.

The closing tag has a forward slash after the the < symbol.



The terms "tag" and "element" are often used interchangeably.

Strictly speaking, however, an element comprises the opening

tag and the closing tag and any content that lies between them.

ATTRIBUTES TELL US MORE ABOUT ELEMENTS

Attributes provide additional information about the contents of an element. They appear on the opening tag of the element and are made up of two parts: a **name** and a **value**, separated by an equals sign.



The attribute **name** indicates what kind of extra information you are supplying about the element's content. It should be written in lowercase.

The **value** is the information or setting for the attribute. It should be placed in double quotes. Different attributes can have different values.

Here an attribute called `lang` is used to indicate the language used in this element. The value of this attribute on this page specifies it is in US English.

HTML5 allows you to use uppercase attribute names and omit the quotemarks, but this is not recommended.



The majority of attributes can only be used on certain elements, although a few attributes (such as lang) can appear on any element.

Most attribute values are either pre-defined or follow a stipulated format. We will look at the permitted values as we introduce each new attribute.

The value of the lang attribute is an abbreviated way of specifying which language is used inside the element that all browsers understand.

BODY, HEAD & TITLE

<body>

You met the <body> element in the first example we created. Everything inside this element is shown inside the main browser window.

<head>

Before the <body> element you will often see a <head> element. This contains information *about* the page (rather than information that is shown within the main part of the browser window that is highlighted in blue on the opposite page). You will usually find a <title> element inside the <head> element.

<title>

The contents of the <title> element are either shown in the top of the browser, above where you usually type in the URL of the page you want to visit, or on the tab for that page (if your browser uses tabs to allow you to view multiple pages at the same time).

/chapter-01/body-head-title.html

HTML

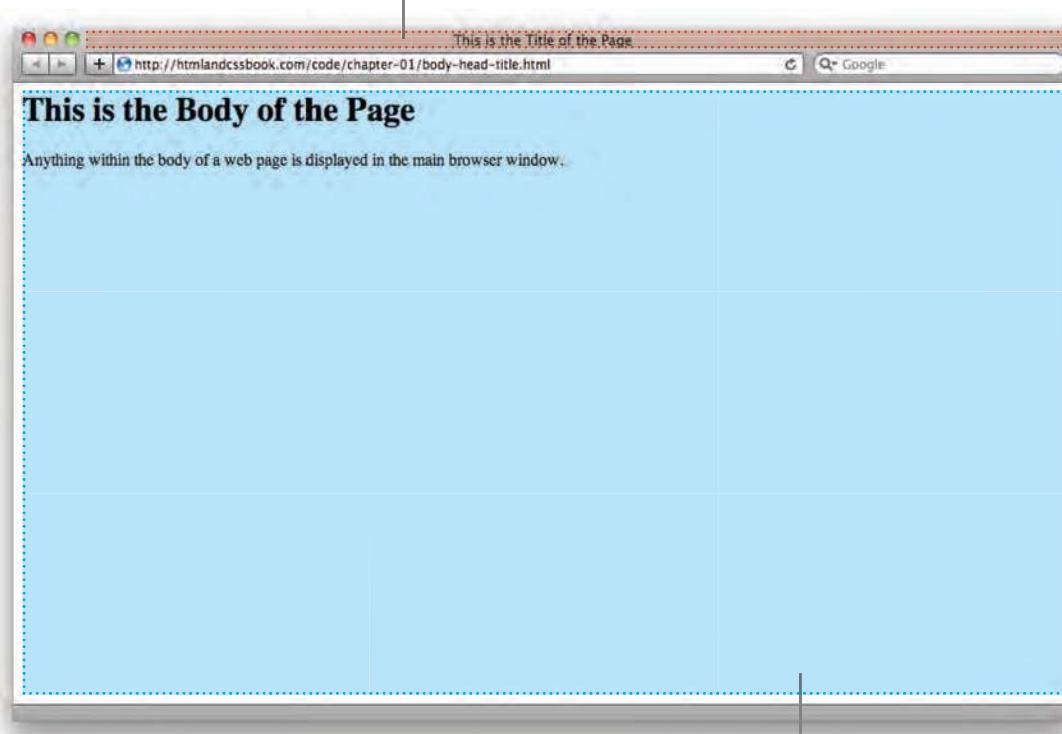
```
<html>
  <head>
    <title>This is the Title of the Page</title>
  </head>
  <body>
    <h1>This is the Body of the Page</h1>
    <p>Anything within the body of a web page is
       displayed in the main browser window.</p>
  </body>
</html>
```

RESULT

This is the Body of the Page

Anything within the body of a web page is displayed in the main browser window.

Anything written between the <title> tags will appear in the title bar (or tabs) at the top of the browser window, highlighted in orange here.



Anything written between the <body> tags will appear in the main browser window, highlighted in blue here.

You may know that HTML stands for HyperText Markup Language. The HyperText part refers to the fact that HTML allows you to create links that allow visitors to move from one

page to another quickly and easily. A markup language allows you to annotate text, and these annotations provide additional meaning to the contents of a document. If you think of a web

page, we add code around the original text we want to display and the browser then uses the code to display the page correctly. So the tags we add are the markup.

CREATING A WEB PAGE ON A PC

To create your first web page on a PC, start up Notepad. You can find this by going to:

Start

All Programs (or Programs)

Accessories

Notepad

You might also like to download a free editor called Notepad++ from notepad-plus-plus.org.



Type the code shown on the right.





3

Go to the File menu and select **Save as...**. You will need to save the file somewhere you can remember. If you like, you could create a folder for any examples that you try out from this book.

Save this file as **first-test.html**. Make sure that the **Save as type** drop down has **All Files** selected.



4

Start your web browser. Go to the **File** menu and select **Open**. Browse to the file that you just created, select it and click on the **Open** button. The result should look something like the screen shot to the left.

If it doesn't look like this, find the file you just created on your computer and make sure that it has the file extension **.html** (if it is **.txt** then you need to go back to Notepad and save the file again, but this time put quote marks around the name "first-test.html").

CREATING A WEB PAGE ON A MAC

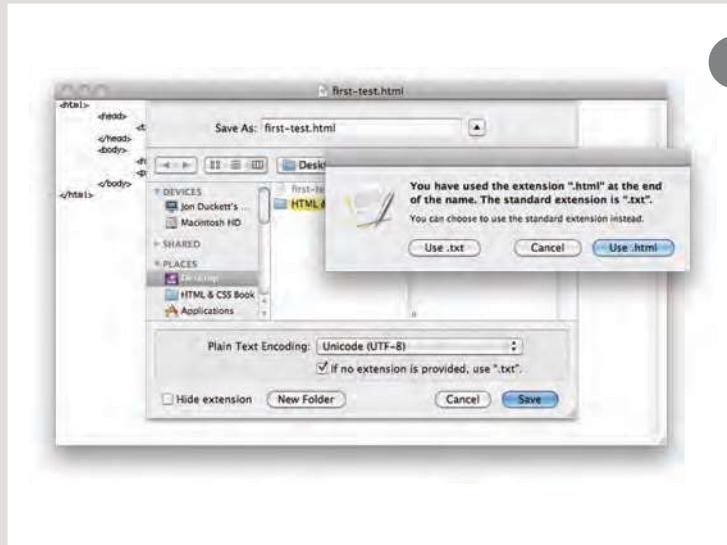
To create your first web page on a Mac, start up TextEdit. This should be in your **Applications** folder.

You might also like to download a free text editor for creating web pages called TextWrangler which is available from barebones.com.



Type the code shown on the right.





3

Now go to the **File** menu and select **Save as...**. You will need to save the file somewhere you can remember.

If you like, you could create a folder for any examples that you try out from this book. Save this file as `first-test.html`. You will probably see a window like the screen shot to the left.

You want to select the **Use .html** button.

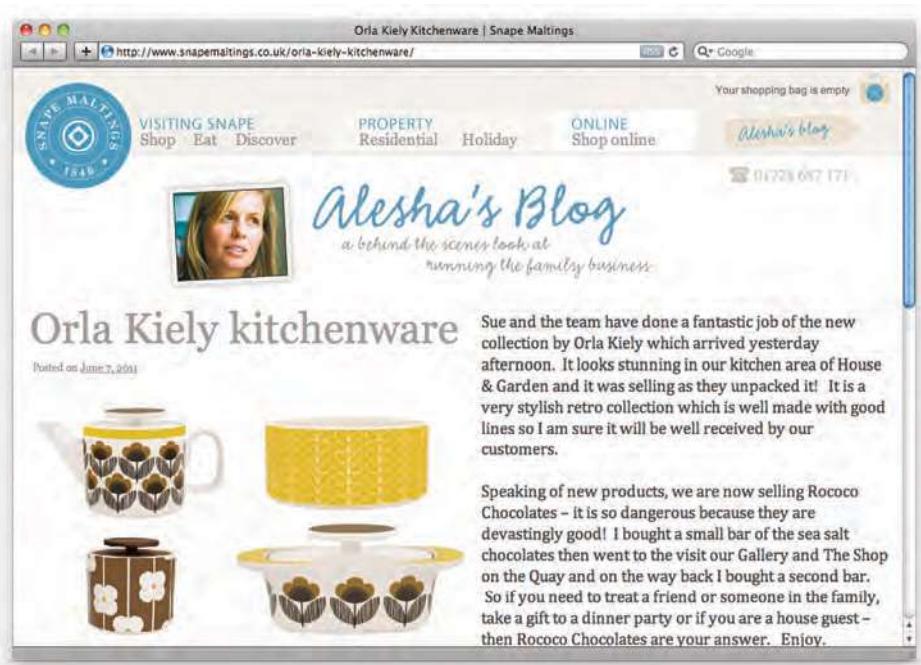


4

Next, start your web browser, go to the **File** menu, and select **Open**. You should browse to the file that you just created, select it and click on the **Open** button. The result should look like the screen shot to the left.

If it doesn't look like this, you might need to change one of the settings in TextEdit. Go to the TextEdit menu and select **Preferences**. Then on the preferences for **Open and Save**, tick the box that says **Ignore rich text commands in HTML files**. Now try to save the file again.

CODE IN A CONTENT MANAGEMENT SYSTEM



If you are working with a content management system, blogging platform, or e-commerce application, you will probably log into a special administration section of the website to control it. The tools provided in the administration sections of these sites usually allow you to edit parts of the page rather than the entire page, which means you will rarely see the `<html>`, `<head>`, or `<body>` elements.

Looking at the content management system on the opposite page, you have a box

that allows you to enter a title for the page, another box for the main article, a way to enter a publication date, and something to indicate which section of the site this page belongs in.

For an e-commerce store, you might have boxes that allow you to enter a title for the product, a description of the product, its price, and the quantity available.

That is because they use a single 'template' to control all of the pages for a section of the site. (For example, an e-commerce

system might use the same template to show all of their products.) The information you supply is placed into the templates.

The advantage of this approach is that people who do not know how to write web pages can add information to a website and it is also possible to change the presentation of something in the template, and it will automatically update every page that uses that template. If you imagine an e-commerce store with 1,000 items for sale, just



altering one template is a lot easier than changing the page for each individual product. In systems like this, when you have a large block of text that you can edit, such as a news article, blog entry or the description of a product in an e-commerce store, you will often see a text editor displayed.

Text editors usually have controls a little like those on your word processor, giving you different options to style text, add links or insert images. Behind the scenes these editors

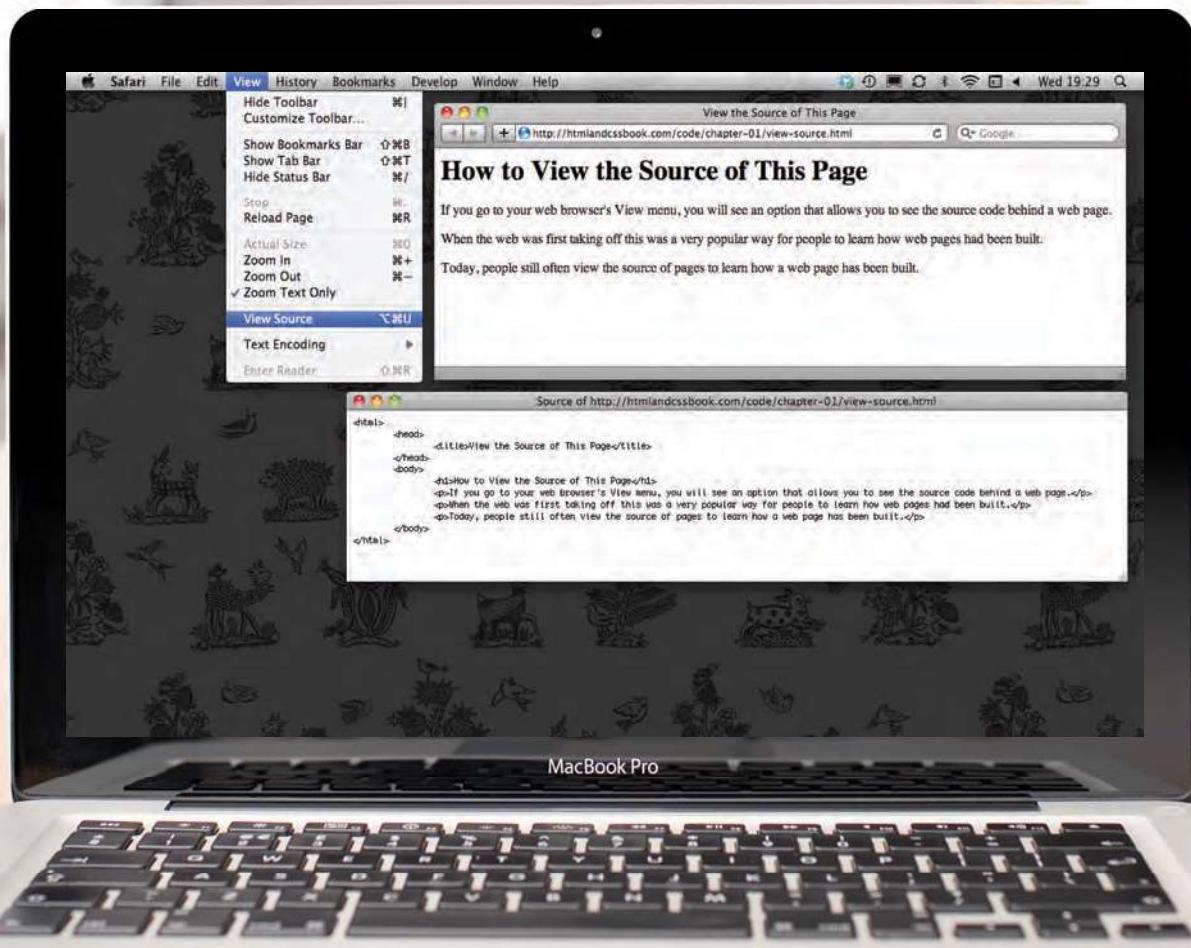
are adding HTML code to your text, just like the code you have seen earlier in this chapter. Many of these editors will have an option that allows you to see (and edit) the code that they produce.

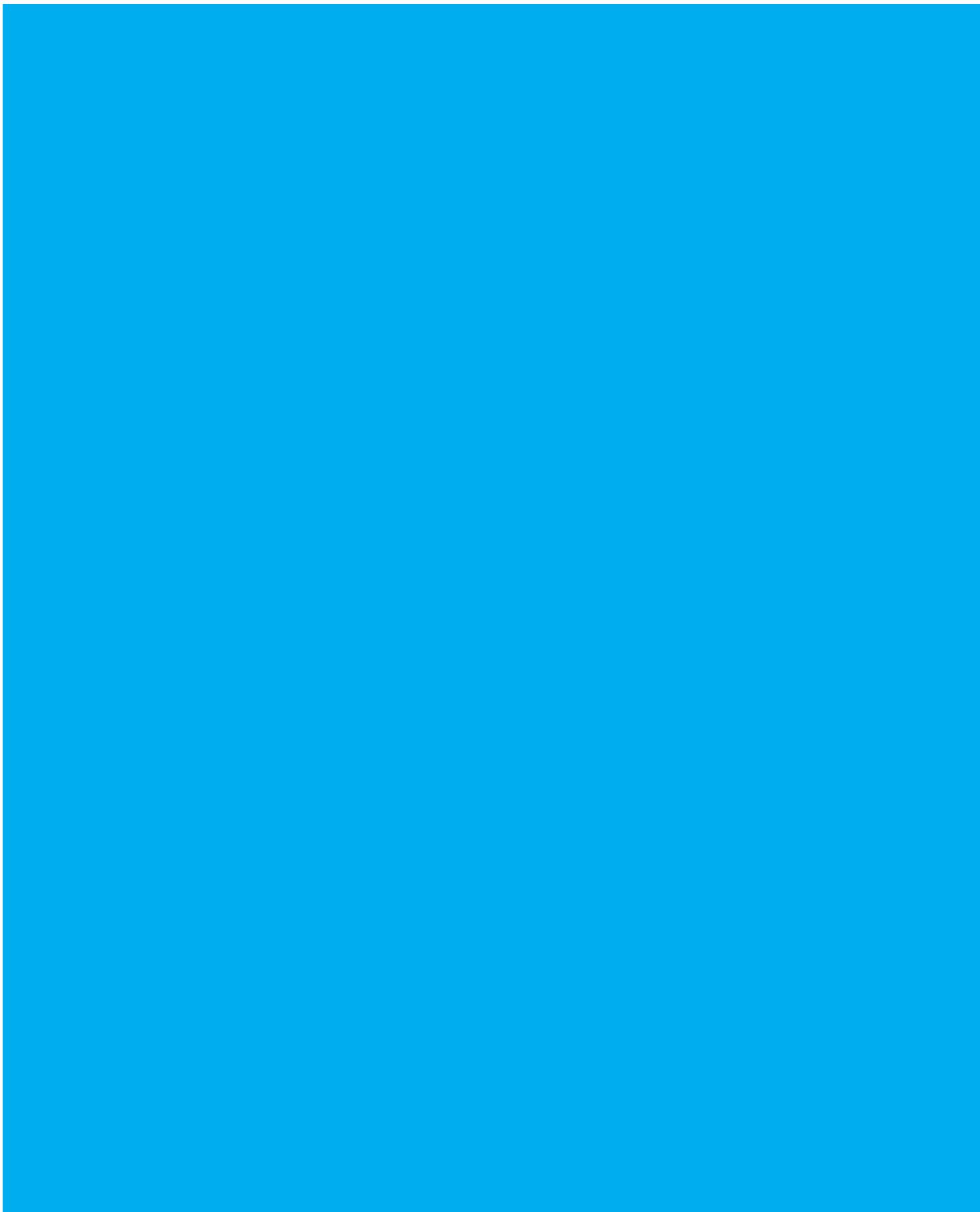
Once you know how to read and edit this code, you can take more control over these sections of your website.

In the example above, you can see that the text editor has a tab for Visual / HTML views of what the user enters. Other systems

might have a button (which often shows angle brackets) to indicate how to access the code.

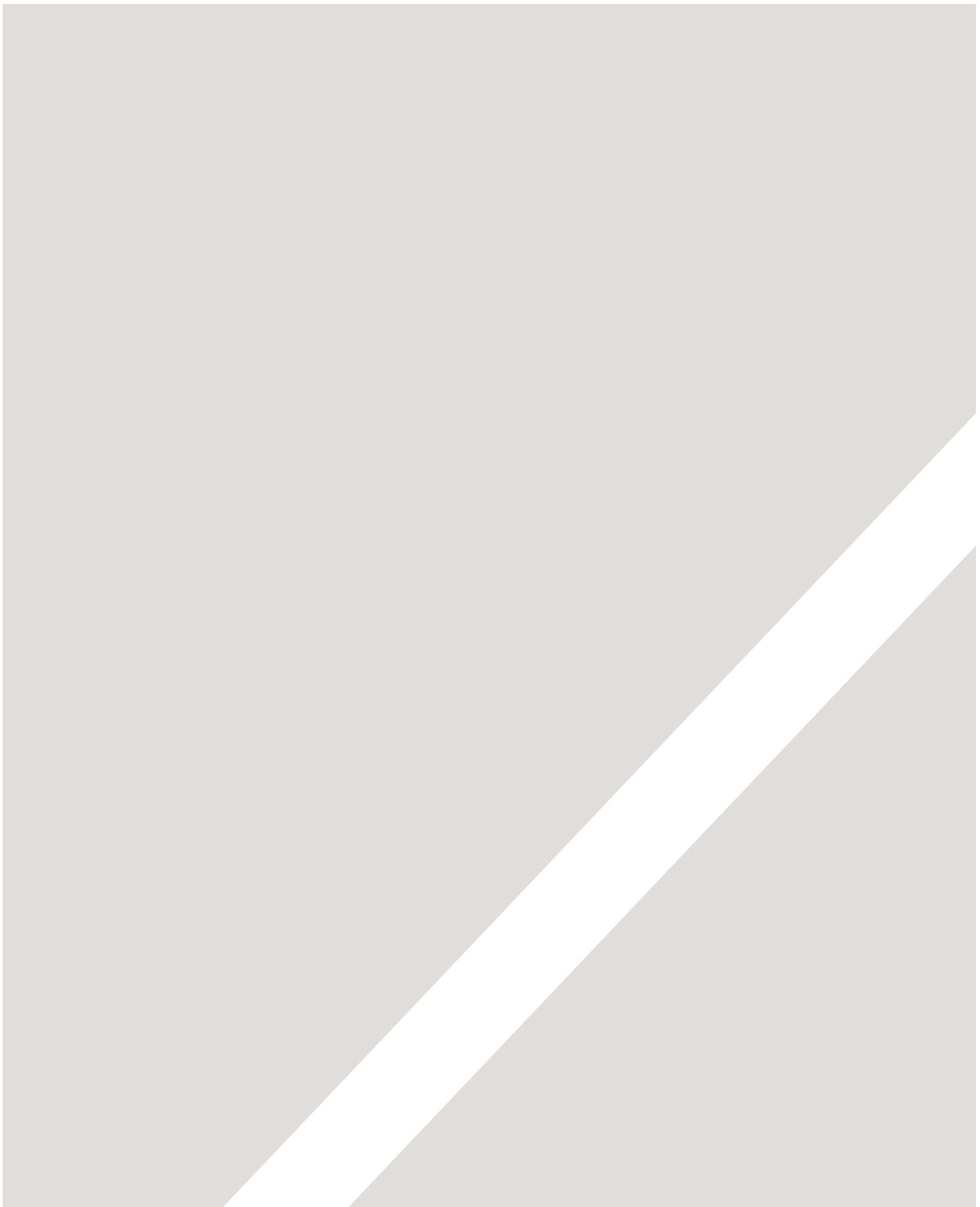
Some content management systems offer tools that also allow you to edit the template files. If you do try to edit template files you need to check the documentation for your CMS as they all differ from each other. You need to be careful when editing template files because if you delete the wrong piece of code or add something in the wrong place the site may stop working entirely.





SUMMARY STRUCTURE

- ▶ HTML pages are text documents.
- ▶ HTML uses tags (characters that sit inside angled brackets) to give the information they surround special meaning.
- ▶ Tags are often referred to as elements.
- ▶ Tags usually come in pairs. The opening tag denotes the start of a piece of content; the closing tag denotes the end.
- ▶ Opening tags can carry attributes, which tell us more about the content of that element.
- ▶ Attributes require a name and a value.
- ▶ To learn HTML you need to know what tags are available for you to use, what they do, and where they can go.



2

TEXT

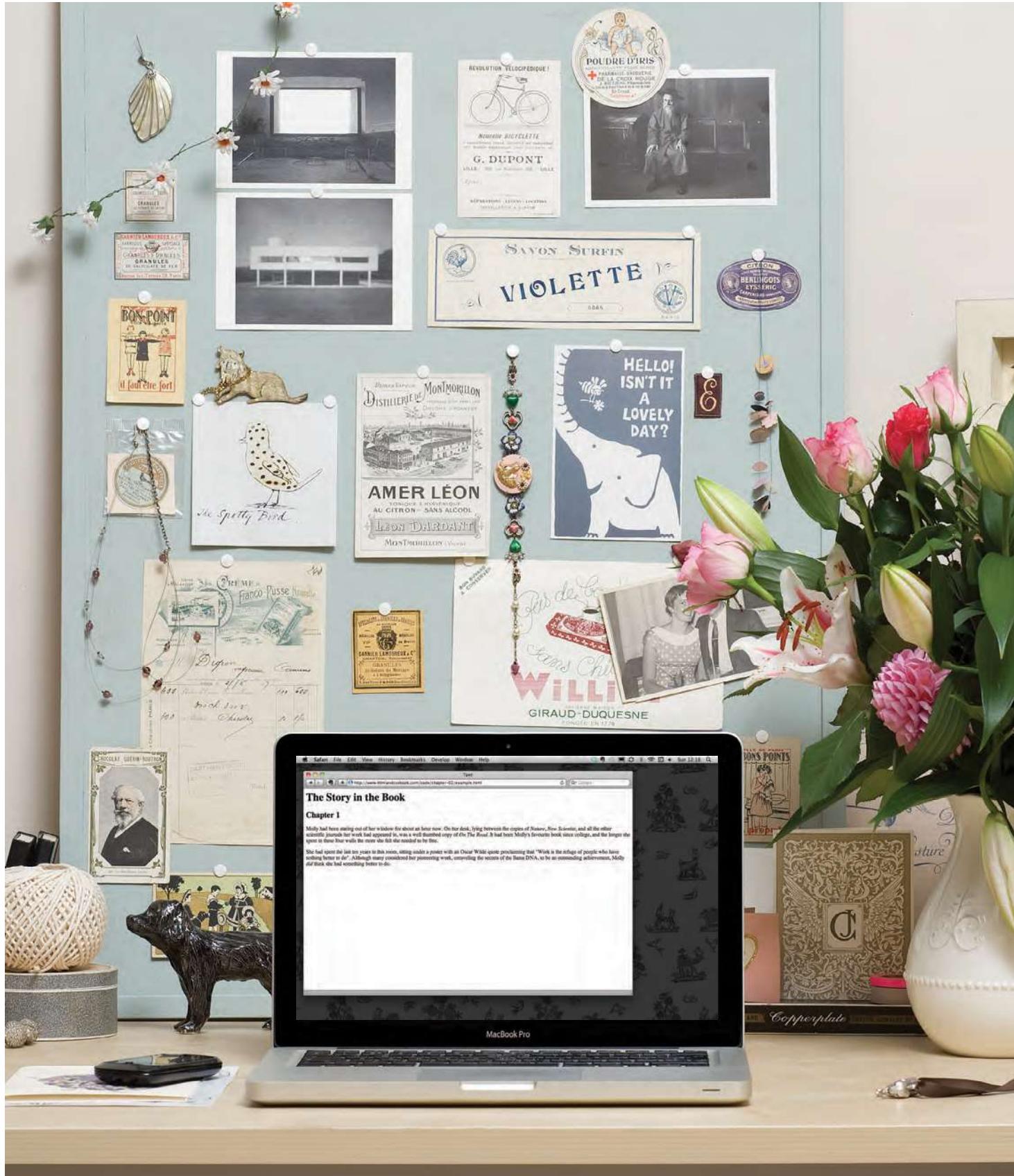
- ▶ Headings and paragraphs
- ▶ Bold, italic, emphasis
- ▶ Structural and semantic markup



When creating a web page, you add tags (known as markup) to the contents of the page. These tags provide extra meaning and allow browsers to show users the appropriate structure for the page.

In this chapter we focus on how to add markup to the text that appears on your pages. You will learn about:

- **Structural markup:** the elements that you can use to describe both headings and paragraphs
- **Semantic markup:** which provides extra information; such as where emphasis is placed in a sentence, that something you have written is a quotation (and who said it), the meaning of acronyms, and so on



HEADINGS

<h1>
<h2>
<h3>
<h4>
<h5>
<h6>

HTML has six "levels" of headings:

<h1> is used for main headings

<h2> is used for subheadings

If there are further sections under the subheadings then the <h3> element is used, and so on...

Browsers display the contents of headings at different sizes. The contents of an <h1> element is the largest, and the contents of an <h6> element is the smallest. The exact size at which each browser shows the headings can vary slightly. Users can also adjust the size of text in their browser. You will see how to control the size of text, its color, and the fonts used when we come to look at CSS.

chapter-02/headings.html

HTML

```
<h1>This is a Main Heading</h1>
<h2>This is a Level 2 Heading</h2>
<h3>This is a Level 3 Heading</h3>
<h4>This is a Level 4 Heading</h4>
<h5>This is a Level 5 Heading</h5>
<h6>This is a Level 6 Heading</h6>
```

This is a Main Heading

RESULT

This is a Level 2 Heading

This is a Level 3 Heading

This is a Level 4 Heading

This is a Level 5 Heading

This is a Level 6 Heading

PARAGRAPHS

HTML

chapter-02/paragraphs.html

```
<p>A paragraph consists of one or more sentences  
that form a self-contained unit of discourse. The  
start of a paragraph is indicated by a new  
line.</p>  
<p>Text is easier to understand when it is split up  
into units of text. For example, a book may have  
chapters. Chapters can have subheadings. Under  
each heading there will be one or more  
paragraphs.</p>
```

<p>

To create a paragraph, surround the words that make up the paragraph with an opening `<p>` tag and closing `</p>` tag.

By default, a browser will show each paragraph on a new line with some space between it and any subsequent paragraphs.

RESULT

A paragraph consists of one or more sentences that form a self-contained unit of discourse. The start of a paragraph is indicated by a new line.

Text is easier to understand when it is split up into units of text. For example, a book may have chapters. Chapters can have subheadings. Under each heading there will be one or more paragraphs.

BOLD & ITALIC

By enclosing words in the tags `` and `` we can make characters appear bold.

The `` element also represents a section of text that would be presented in a visually different way (for example key words in a paragraph) although the use of the `` element does not imply any additional meaning.

chapter-02/bold.html

HTML

```
<p>This is how we make a word appear bold.</b>
</p>
<p>Inside a product description you might see some
<b>key features</b> in bold.</p>
```

RESULT

This is how we make a word appear **bold.**

Inside a product description you might see some **key features** in bold.

<i>

By enclosing words in the tags `<i>` and `</i>` we can make characters appear italic.

The `<i>` element also represents a section of text that would be said in a different way from surrounding content — such as technical terms, names of ships, foreign words, thoughts, or other terms that would usually be italicized.

chapter-02/italic.html

HTML

```
<p>This is how we make a word appear italic</i>.
</p>
<p>It's a potato Solanum tuberosum.</p>
<p>Captain Cook sailed to Australia on the
<i>Endeavour</i>.</p>
```

RESULT

This is how we make a word appear *italic*.

It's a potato *Solanum tuberosum*.

Captain Cook sailed to Australia on the *Endeavour*.

SUPERSCRIPT & SUBSCRIPT

HTML

chapter-02/superscript-and-subscript.html

```
<p>On the 4<sup>th</sup> of September you will learn  
about E=MC<sup>2</sup>.</p>  
<p>The amount of CO<sub>2</sub> in the atmosphere  
grew by 2ppm in 2009<sub>1</sub>.</p>
```

RESULT

On the 4th of September you will learn about E=MC².

The amount of CO₂ in the atmosphere grew by 2ppm in 2009₁.

<sup>

The `<sup>` element is used to contain characters that should be superscript such as the suffixes of dates or mathematical concepts like raising a number to a power such as 2².

<sub>

The `<sub>` element is used to contain characters that should be subscript. It is commonly used with foot notes or chemical formulas such as H₂O.

WHITE SPACE

In order to make code easier to read, web page authors often add extra spaces or start some elements on new lines.

When the browser comes across two or more spaces next to each other, it only displays one space. Similarly if it comes across a line break, it treats that as a single space too. This is known as **white space collapsing**.

You will often see that web page authors take advantage of white space collapsing to indent their code in order to make it easier to follow.

chapter-02/white-space.html

HTML

```
<p>The moon is drifting away from Earth.</p>
<p>The moon      is drifting away from Earth.</p>
<p>The moon is drifting away from
Earth.</p>
```

RESULT

The moon is drifting away from Earth.

The moon is drifting away from Earth.

The moon is drifting away from Earth.

LINE BREAKS & HORIZONTAL RULES

HTML

chapter-02/line-breaks.html

```
<p>The Earth<br />gets one hundred tons heavier  
every day<br />due to falling space dust.</p>
```

RESULT

The Earth
gets one hundred tons heavier every day
due to falling space dust.

As you have already seen, the browser will automatically show each new paragraph or heading on a new line. But if you wanted to add a line break inside the middle of a paragraph you can use the line break tag `
`.

HTML

chapter-02/horizontal-rules.html

```
<p>Venus is the only planet that rotates  
clockwise.</p>  
<hr />  
<p>Jupiter is bigger than all the other planets  
combined.</p>
```

RESULT

Venus is the only planet that rotates clockwise.

Jupiter is bigger than all the other planets combined.

<hr />

To create a break between themes — such as a change of topic in a book or a new scene in a play — you can add a horizontal rule between sections using the `<hr />` tag.

There are a few elements that do not have any words between an opening and closing tag. They are known as **empty elements** and they are written differently.

An empty element usually has only one tag. Before the closing angled bracket of an empty element there will often be a space and a forward slash character. Some web page authors miss this out but it is a good habit to get into.

SEMANTIC MARKUP

There are some text elements that are not intended to affect the structure of your web pages, but they do add extra information to the pages — they are known as semantic markup.

In the rest of the chapter you will meet some more elements that will help you when you are adding text to web pages. For example, you are going to meet the `` element that allows you to indicate where emphasis should be placed on selected words and the `<blockquote>` element which indicates that a block of text is a quotation.

Browsers often display the contents of these elements in a different way. For example, the content of the `` element is shown in italics, and a `<blockquote>` is usually indented. But you should not use them to change the way that your text looks; their purpose is to describe the content of your web pages more accurately.

The reason for using these elements is that other programs, such as screen readers or search engines, can use this extra information. For example, the voice of a screen reader may add emphasis to the words inside the `` element, or a search engine might register that your page features a quote if you use the `<blockquote>` element.

STRONG & EMPHASIS

The use of the `` element indicates that its content has strong importance. For example, the words contained in this element might be said with strong emphasis.

By default, browsers will show the contents of a `` element in bold.

chapter-02/strong.html

HTML

```
<p><strong>Beware:</strong> Pickpockets operate in  
this area.</p>  
<p>This toy has many small pieces and is <strong>not  
suitable for children under five years old.  
</strong></p>
```

RESULT

Beware: Pickpockets operate in this area.

**This toy has many small pieces and is not
suitable for children under five years old.**

The `` element indicates emphasis that subtly changes the meaning of a sentence.

By default browsers will show the contents of an `` element in italic.

chapter-02/emphasis.html

HTML

```
<p>I <em>think</em> Ivy was the first.</p>  
<p>I think <em>Ivy</em> was the first.</p>  
<p>I think Ivy was the <em>first</em>. </p>
```

RESULT

I think Ivy was the first.

I think Ivy was the first.

*I think Ivy was the *first*.*

QUOTATIONS

HTML

chapter-02/quotations.html

```
<blockquote cite="http://en.wikipedia.org/wiki/  
Winnie-the-Pooh">  
  <p>Did you ever stop to think, and forget to start  
    again?</p>  
</blockquote>  
<p>As A.A. Milne said, <q>Some people talk to  
  animals. Not many listen though. That's the  
  problem.</q></p>
```

RESULT

Did you ever stop to think, and forget
to start again?

As A.A. Milne said, "Some people talk to animals.
Not many listen though. That's the problem."

There are two elements
commonly used for marking up
quotations:

<blockquote>

The `<blockquote>` element is
used for longer quotes that take
up an entire paragraph. Note
how the `<p>` element is still
used inside the `<blockquote>`
element.

Browsers tend to indent the
contents of the `<blockquote>`
element, however you should not
use this element just to indent a
piece of text — rather you should
achieve this effect using CSS.

<q>

The `<q>` element is used for
shorter quotes that sit within
a paragraph. Browsers are
supposed to put quotes around
the `<q>` element, however
Internet Explorer does not —
therefore many people avoid
using the `<q>` element.

Both elements may use the `cite`
attribute to indicate where the
quote is from. Its value should
be a URL that will have more
information about the source of
the quotation.

ABBREVIATIONS & ACRONYMS

<abbr>

If you use an abbreviation or an acronym, then the <abbr> element can be used. A title attribute on the opening tag is used to specify the full term.

In HTML 4 there was a separate <acronym> element for acronyms. To spell out the full form of the acronym, the title attribute was used (as with the <abbr> element above). HTML5 just uses the <abbr> element for both abbreviations and acronyms.

chapter-02/abbreviations.html

HTML

```
<p><abbr title="Professor">Prof</abbr> Stephen  
Hawking is a theoretical physicist and  
cosmologist.</p>  
<p><acronym title="National Aeronautics and Space  
Administration">NASA</acronym> do some crazy  
space stuff.</p>
```

RESULT

Prof Stephen Hawking is a theoretical physicist and cosmologist.
NASA do some crazy space stuff.

National Aeronautics and Space
Administration

CITATIONS & DEFINITIONS

HTML

chapter-02/citations.html

```
<p><cite>A Brief History of Time</cite> by Stephen  
Hawking has sold over ten million copies  
worldwide.</p>
```

RESULT

A Brief History of Time by Stephen Hawking has
sold over ten million copies worldwide.

<cite>

When you are referencing a piece of work such as a book, film or research paper, the `<cite>` element can be used to indicate where the citation is from.

In HTML5, `<cite>` should not really be used for a person's name — but it was allowed in HTML 4, so most people are likely to continue to use it.

Browsers will render the content of a `<cite>` element in italics.

HTML

chapter-02/definitions.html

```
<p>A <dfn>black hole</dfn> is a region of space from  
which nothing, not even light, can escape.</p>
```

RESULT

**A black hole is a region of space from which
nothing, not even light, can escape.**

<dfn>

The first time you explain some new terminology (perhaps an academic concept or some jargon) in a document, it is known as the defining instance of it.

The `<dfn>` element is used to indicate the defining instance of a new term.

Some browsers show the content of the `<dfn>` element in italics. Safari and Chrome do not change its appearance.

AUTHOR DETAILS

<address>

The <address> element has quite a specific use: to contain contact details for the author of the page.

It can contain a physical address, but it does not have to. For example, it may also contain a phone number or email address.

Browsers often display the content of the <address> element in italics.

You may also be interested in something called the hCard microformat for adding physical address information to your markup.

ONLINE EXTRA:

You can find out more about hCards on the website accompanying this book.

chapter-02/address.html

HTML

```
<address>
  <p><a href="mailto:homer@example.org">
    homer@example.org</a></p>
  <p>742 Evergreen Terrace, Springfield.</p>
</address>
```

homer@example.org

RESULT

742 Evergreen Terrace, Springfield.

CHANGES TO CONTENT

HTML

chapter-02/insert-and-delete.html

```
<p>It was the <del>worst</del> <ins>best</ins> idea  
she had ever had.</p>
```

RESULT

It was the worst best idea she had ever had.

**<ins>
**

The `<ins>` element can be used to show content that has been inserted into a document, while the `` element can show text that has been deleted from it.

The content of a `<ins>` element is usually underlined, while the content of a `` element usually has a line through it.

HTML

chapter-02/strikethrough.html

```
<p>Laptop computer:</p>  
<p><s>Was $995</s></p>  
<p>Now only $375</p>
```

RESULT

Laptop computer:

Was \$995

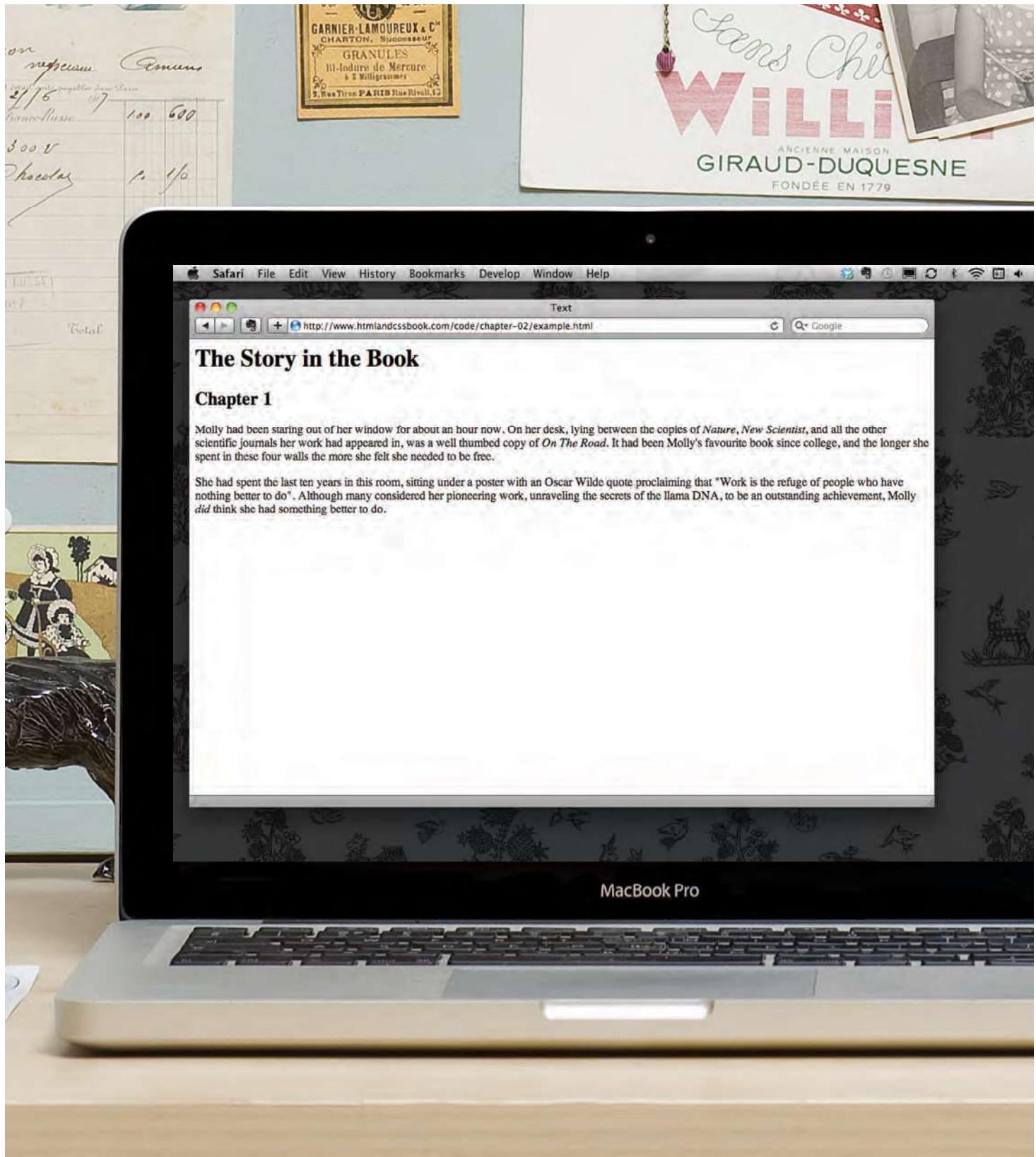
Now only \$375

<s>

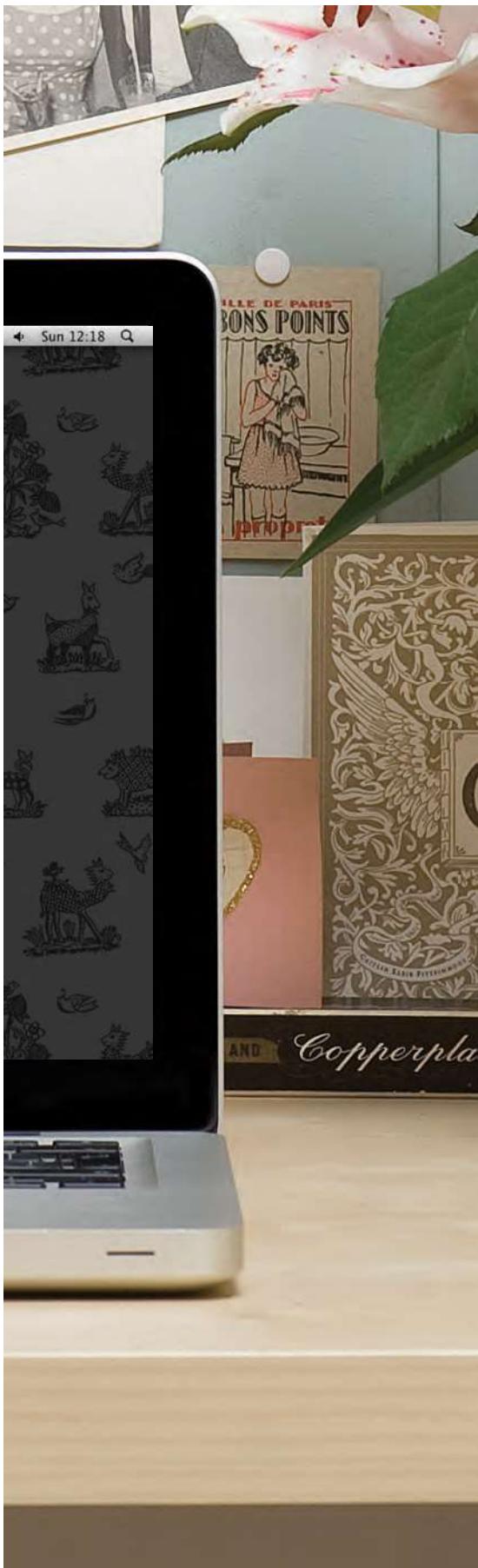
The `<s>` element indicates something that is no longer accurate or relevant (but that should not be deleted).

Visually the content of an `<s>` element will usually be displayed with a line through the center.

Older versions of HTML had a `<u>` element for content that was underlined, but this is being phased out.



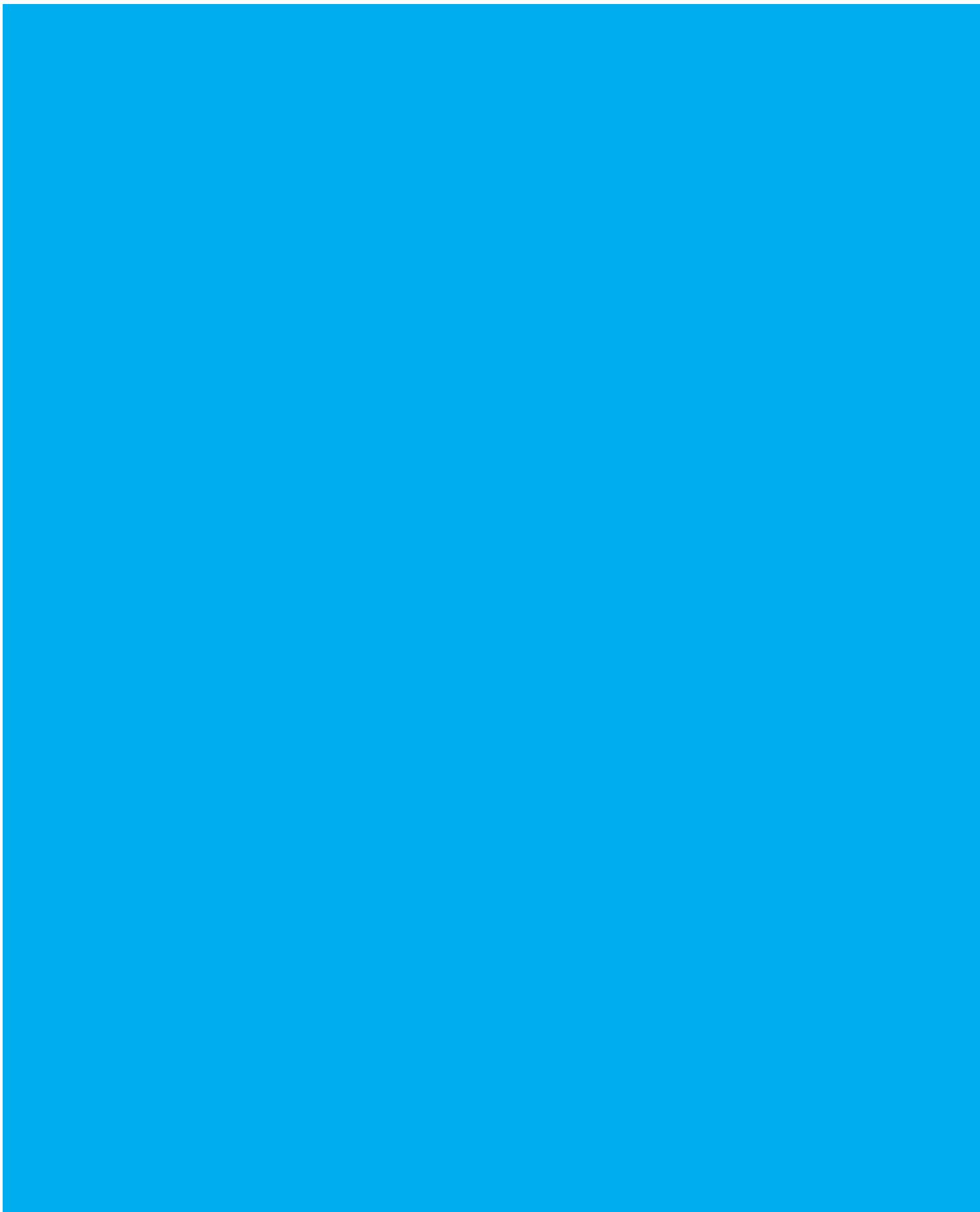
EXAMPLE TEXT



This is a very simple HTML page that demonstrates text markup.

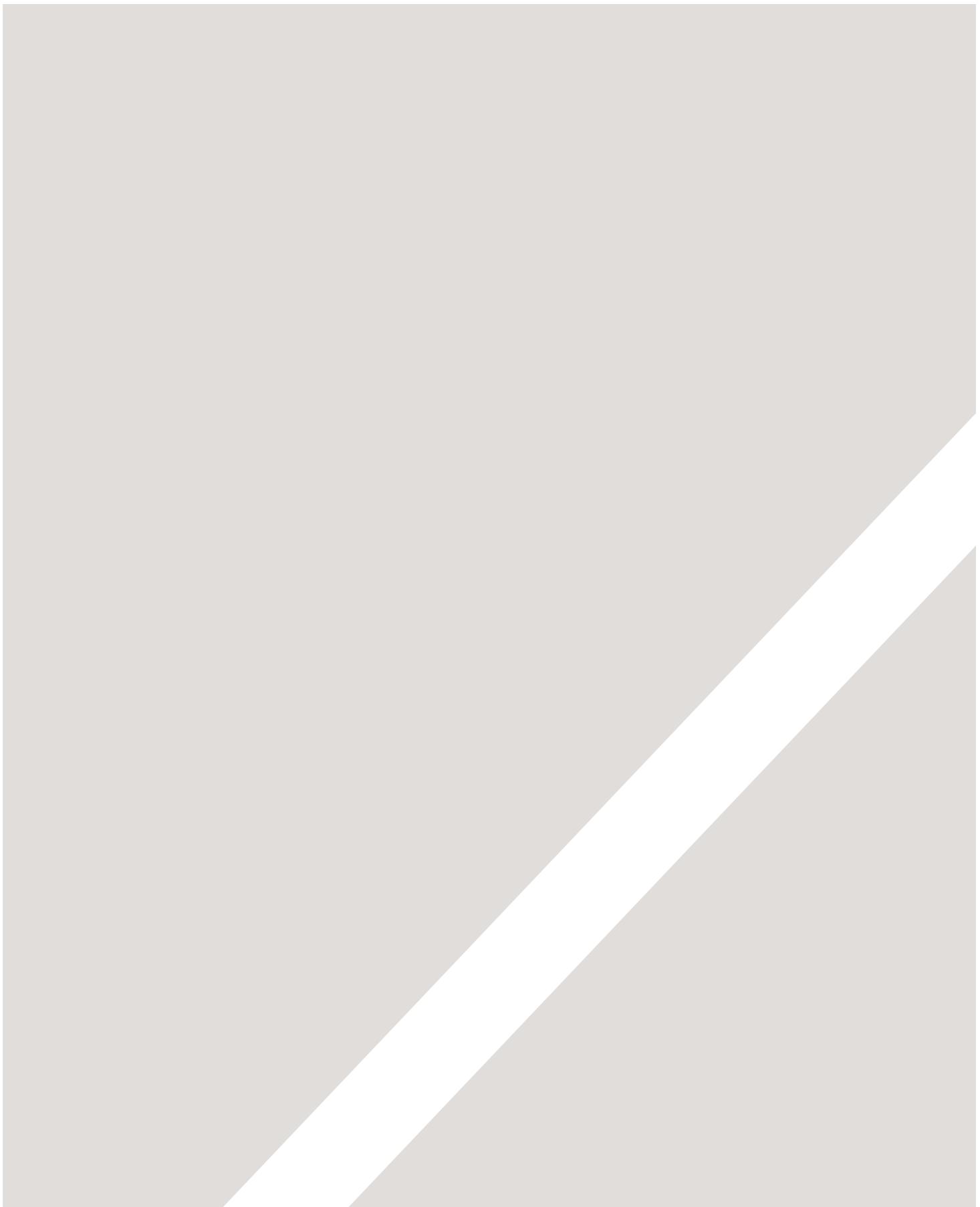
Structural markup includes elements such as `<h1>`, `<h2>`, and `<p>`. Semantic information is carried in elements such as `<cite>` and ``.

```
<html>
  <head>
    <title>Text</title>
  </head>
  <body>
    <h1>The Story in the Book</h1>
    <h2>Chapter 1</h2>
    <p>Molly had been staring out of her window for about
       an hour now. On her desk, lying between the copies
       of <i>Nature</i>, <i>New Scientist</i>, and all
       the other scientific journals her work had
       appeared in, was a well thumbed copy of <cite>On
       The Road</cite>. It had been Molly's favorite book
       since college, and the longer she spent in these
       four walls the more she felt she needed to be
       free.</p>
    <p>She had spent the last ten years in this room,
       sitting under a poster with an Oscar Wilde quote
       proclaiming that <q>Work is the refuge of
       people who have nothing better to do</q>. Although
       many considered her pioneering work, unraveling
       the secrets of the llama <abbr
       title="Deoxyribonucleic acid">DNA</abbr>, to be an
       outstanding achievement, Molly <em>did</em> think
       she had something better to do.</p>
  </body>
</html>
```



SUMMARY TEXT

- ▶ HTML elements are used to describe the structure of the page (e.g. headings, subheadings, paragraphs).
- ▶ They also provide semantic information (e.g. where emphasis should be placed, the definition of any acronyms used, when given text is a quotation).



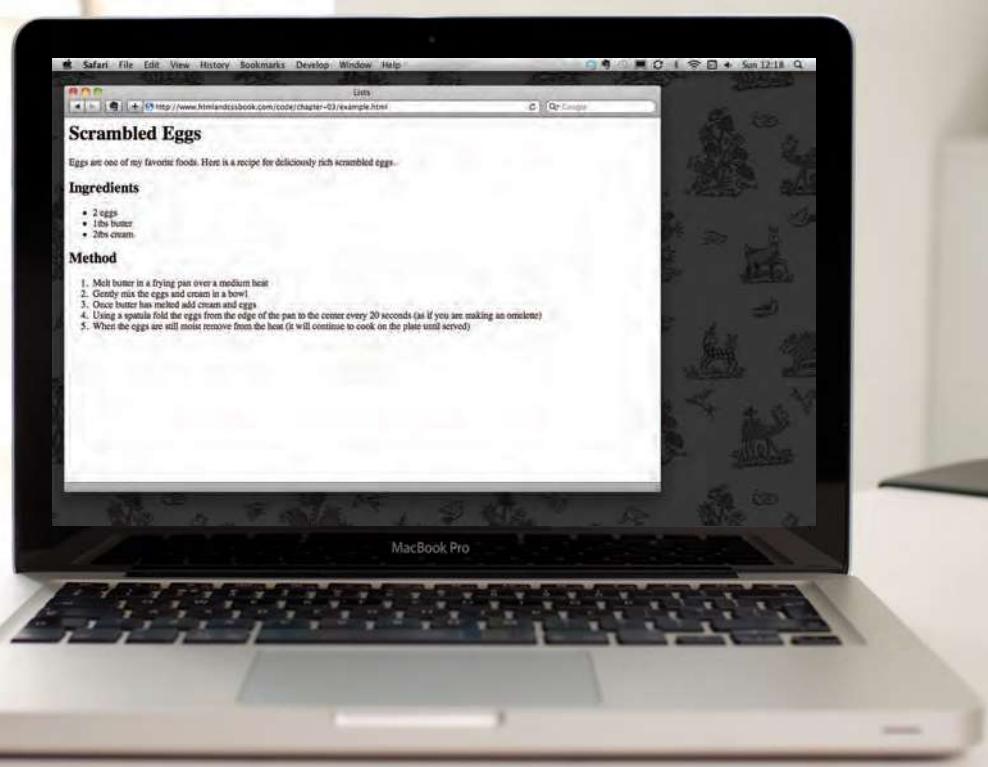
3

LISTS

- ▶ Numbered lists
- ▶ Bullet lists
- ▶ Definition lists

There are lots of occasions when we need to use lists. HTML provides us with three different types:

- **Ordered lists** are lists where each item in the list is numbered. For example, the list might be a set of steps for a recipe that must be performed in order, or a legal contract where each point needs to be identified by a section number.
- **Unordered lists** are lists that begin with a bullet point (rather than characters that indicate order).
- **Definition lists** are made up of a set of terms along with the definitions for each of those terms.



ORDERED LISTS

The ordered list is created with the element.

Each item in the list is placed between an opening tag and a closing tag. (The li stands for list item.)

Browsers indent lists by default.

Sometimes you may see a type attribute used with the element to specify the type of numbering (numbers, letters, roman numerals and so on). It is better to use the CSS list-style-type property covered on pages 333-335.

chapter-03/ordered-lists.html

HTML

```
<ol>
  <li>Chop potatoes into quarters</li>
  <li>Simmer in salted water for 15-20
      minutes until tender</li>
  <li>Heat milk, butter and nutmeg</li>
  <li>Drain potatoes and mash</li>
  <li>Mix in the milk mixture</li>
</ol>
```

RESULT

1. Chop potatoes into quarters
2. Simmer in salted water for 15-20 minutes until tender
3. Heat milk, butter and nutmeg
4. Drain potatoes and mash
5. Mix in the milk mixture

UNORDERED LISTS

HTML

chapter-03/unordered-lists.html

```
<ul>
  <li>1kg King Edward potatoes</li>
  <li>100ml milk</li>
  <li>50g salted butter</li>
  <li>Freshly grated nutmeg</li>
  <li>Salt and pepper to taste</li>
</ul>
```

RESULT

- 1kg King Edward potatoes
- 100ml milk
- 50g salted butter
- Freshly grated nutmeg
- Salt and pepper to taste

The unordered list is created with the `` element.

Each item in the list is placed between an opening `` tag and a closing `` tag. (The `li` stands for list item.)

Browsers indent lists by default.

Sometimes you may see a type attribute used with the `` element to specify the type of bullet point (circles, squares, diamonds and so on). It is better to use the CSS `list-style-type` property covered on pages 333-335.

DEFINITION LISTS

<dl>

The definition list is created with the <dl> element and usually consists of a series of terms and their definitions.

Inside the <dl> element you will usually see pairs of <dt> and <dd> elements.

<dt>

This is used to contain the term being defined (the definition term).

<dd>

This is used to contain the definition.

Sometimes you might see a list where there are two terms used for the same definition or two different definitions for the same term.

chapter-03/definition-lists.html

HTML

```
<dl>
  <dt>Sashimi</dt>
  <dd>Sliced raw fish that is served with
       condiments such as shredded daikon radish or
       ginger root, wasabi and soy sauce</dd>
  <dt>Scale</dt>
  <dd>A device used to accurately measure the
       weight of ingredients</dd>
  <dd>A technique by which the scales are removed
       from the skin of a fish</dd>
  <dt>Scamorze</dt>
  <dt>Scamorzo</dt>
  <dd>An Italian cheese usually made from whole
       cow's milk (although it was traditionally made
       from buffalo milk)</dd>
</dl>
```

RESULT

Sashimi
Sliced raw fish that is served with condiments such as shredded daikon radish or ginger root, wasabi and soy sauce
Scale
A device used to accurately measure the weight of ingredients
A technique by which the scales are removed from the skin of a fish
Scamorze
Scamorzo
An Italian cheese usually made from whole cow's milk (although it was traditionally made from buffalo milk)

NESTED LISTS

HTML

chapter-03/nested-lists.html

```
<ul>
  <li>Mousses</li>
  <li>Pastries
    <ul>
      <li>Croissant</li>
      <li>Mille-feuille</li>
      <li>Palmier</li>
      <li>Profiterole</li>
    </ul>
  </li>
  <li>Tarts</li>
</ul>
```

You can put a second list inside an `` element to create a sub-list or nested list.

Browsers display nested lists indented further than the parent list. In nested unordered lists, the browser will usually change the style of the bullet point too.

RESULT

- Mousses
- Pastries
 - Croissant
 - Mille-feuille
 - Palmier
 - Profiterole
- Tarts

A close-up photograph of a MacBook Pro laptop screen displaying a web page. The screen shows a title 'Scrambled Eggs', a short introduction, a section for 'Ingredients' with a bulleted list, and a section for 'Method' with a numbered list of steps. The URL in the address bar is <http://www.htmllandcssbook.com/code/chapter-03/example.html>. The laptop is resting on a light-colored surface, and the background is slightly blurred.

Safari File Edit View History Bookmarks Develop Window Help

Lists

http://www.htmllandcssbook.com/code/chapter-03/example.html Google

Scrambled Eggs

Eggs are one of my favorite foods. Here is a recipe for deliciously rich scrambled eggs.

Ingredients

- 2 eggs
- 1tbs butter
- 2tbs cream

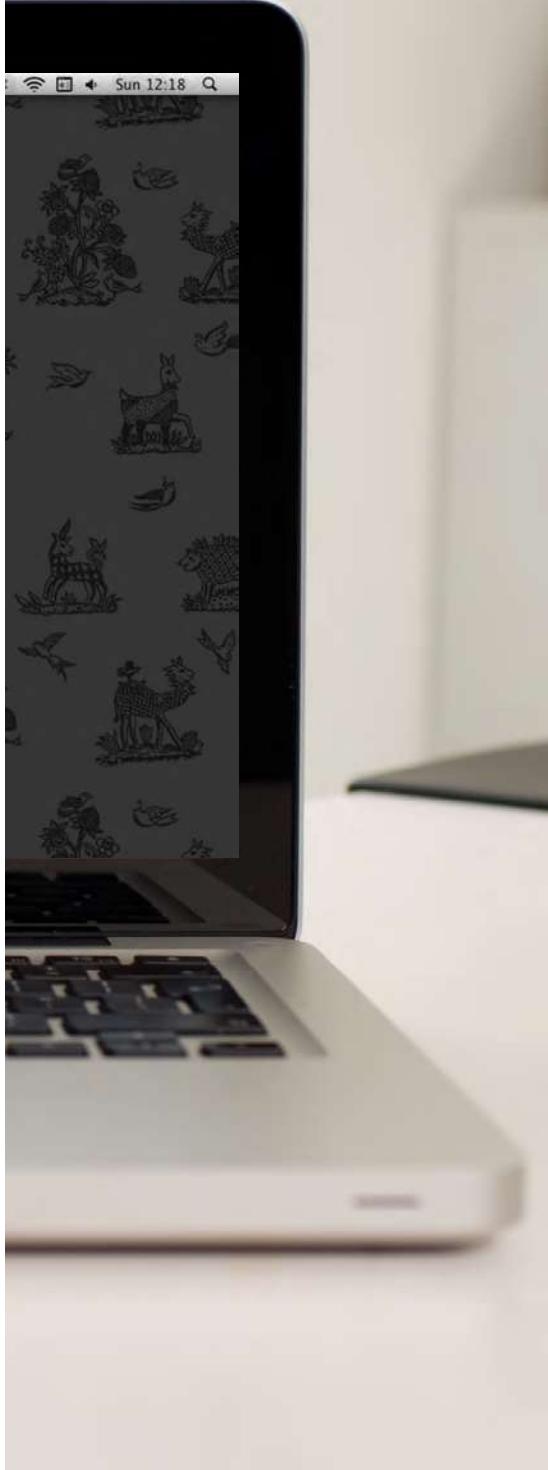
Method

1. Melt butter in a frying pan over a medium heat
2. Gently mix the eggs and cream in a bowl
3. Once butter has melted add cream and eggs
4. Using a spatula fold the eggs from the edge of the pan to the center every 20 seconds (as if you are making an omelette)
5. When the eggs are still moist remove from the heat (it will continue to cook on the plate until served)

MacBook Pro

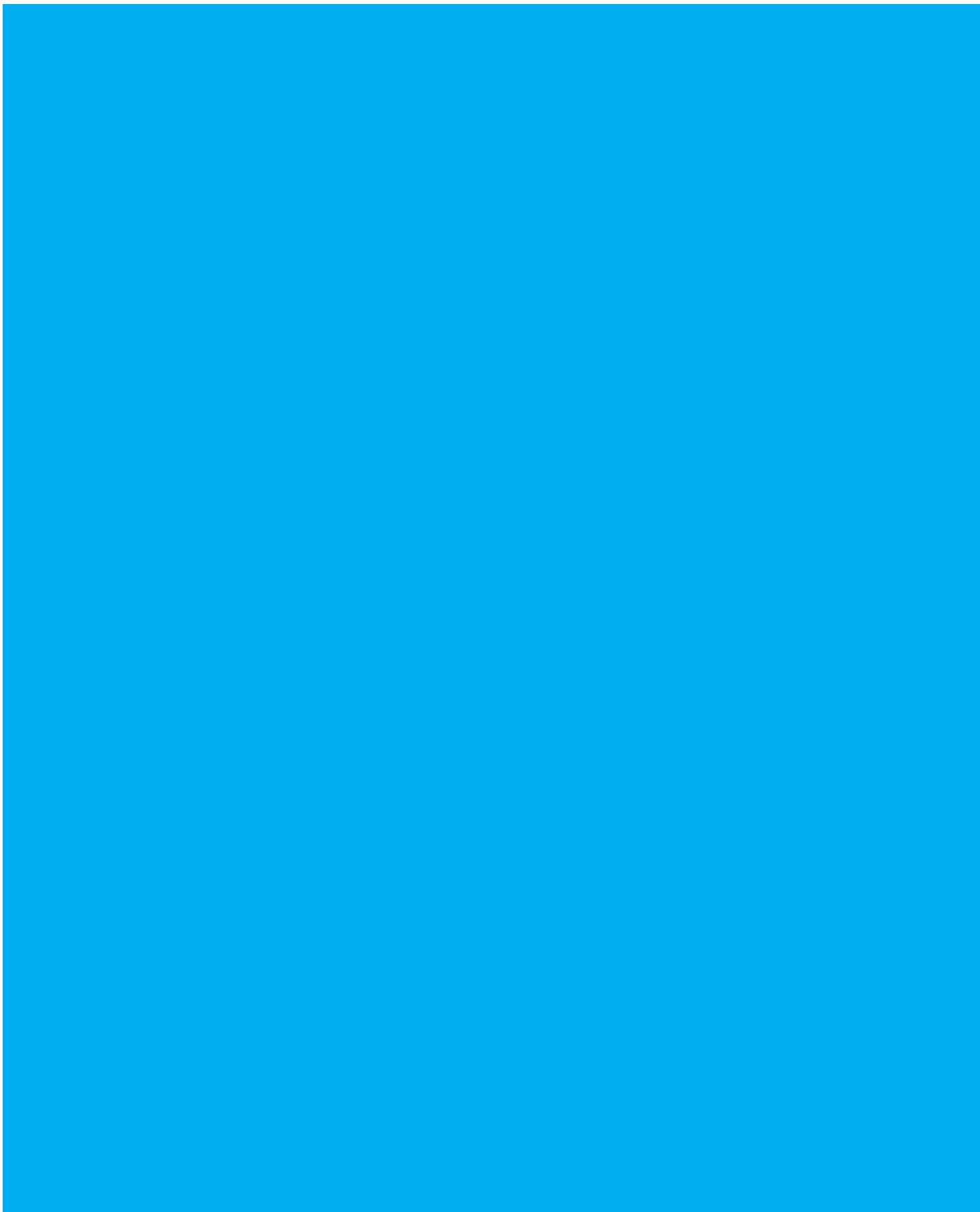
EXAMPLE

LISTS



Here you can see a main heading followed by an introductory paragraph. An unordered list is used to outline the ingredients and an ordered list is used to describe the steps.

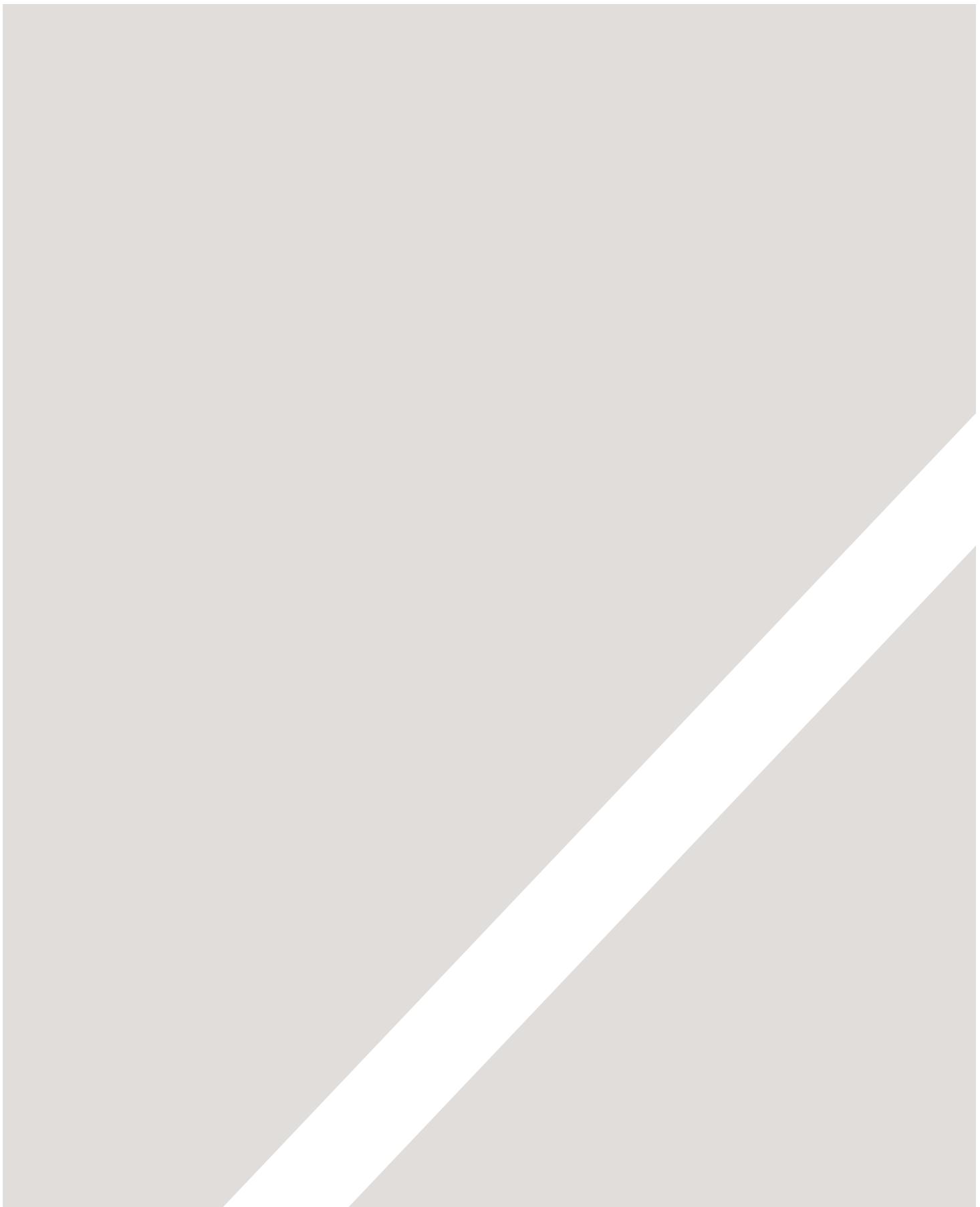
```
<html>
  <head>
    <title>Lists</title>
  </head>
  <body>
    <h1>Scrambled Eggs</h1>
    <p>Eggs are one of my favourite foods. Here is a recipe for deliciously rich scrambled eggs.</p>
    <h2>Ingredients</h2>
    <ul>
      <li>2 eggs</li>
      <li>1tbs butter</li>
      <li>2tbs cream</li>
    </ul>
    <h2>Method</h2>
    <ol>
      <li>Melt butter in a frying pan over a medium heat</li>
      <li>Gently mix the eggs and cream in a bowl</li>
      <li>Once butter has melted add cream and eggs</li>
      <li>Using a spatula fold the eggs from the edge of the pan to the center every 20 seconds (as if you are making an omelette)</li>
      <li>When the eggs are still moist remove from the heat (it will continue to cook on the plate until served)</li>
    </ol>
  </body>
</html>
```



SUMMARY

LISTS

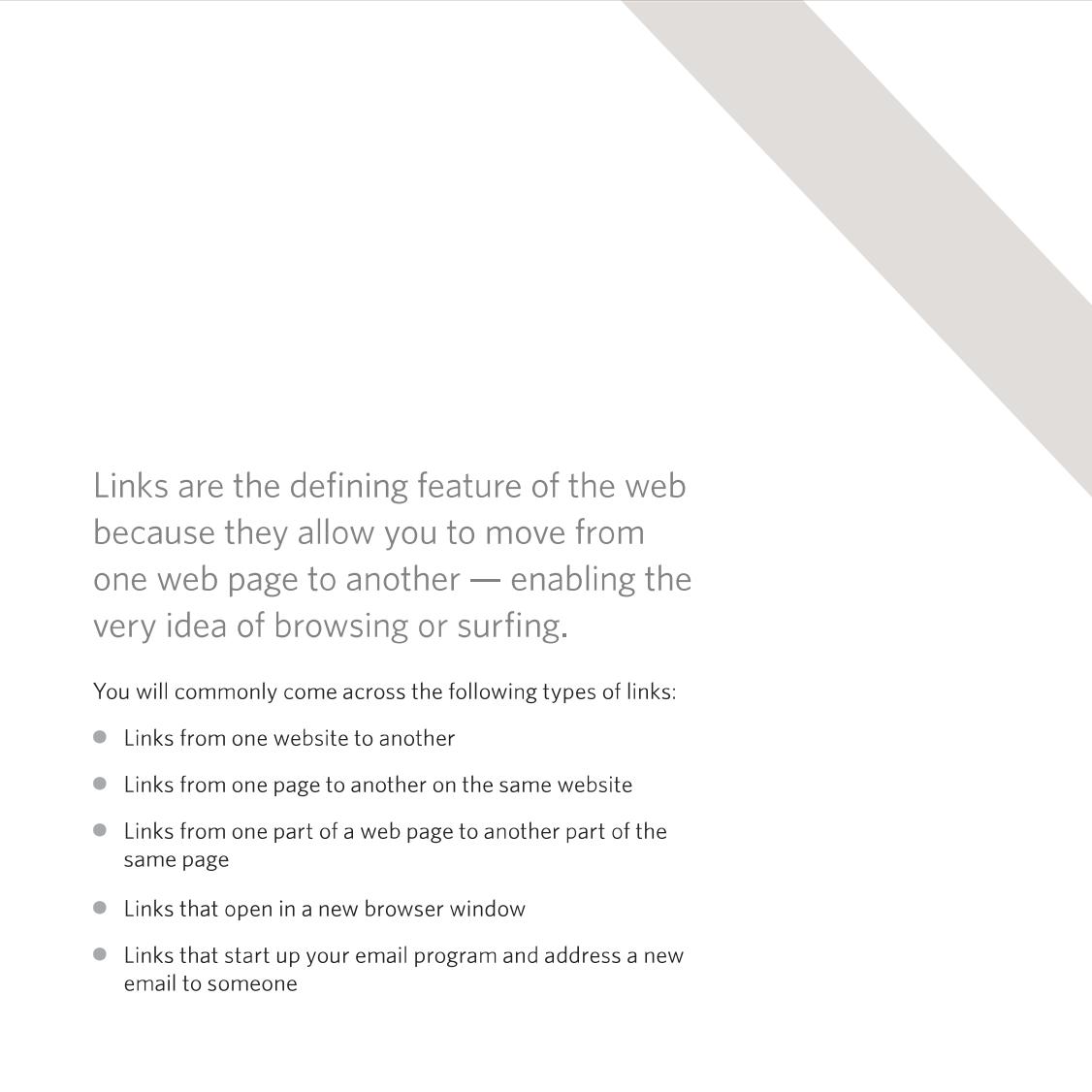
- ▶ There are three types of HTML lists: ordered, unordered, and definition.
- ▶ Ordered lists use numbers.
- ▶ Unordered lists use bullets.
- ▶ Definition lists are used to define terminology.
- ▶ Lists can be nested inside one another.



4

LINKS

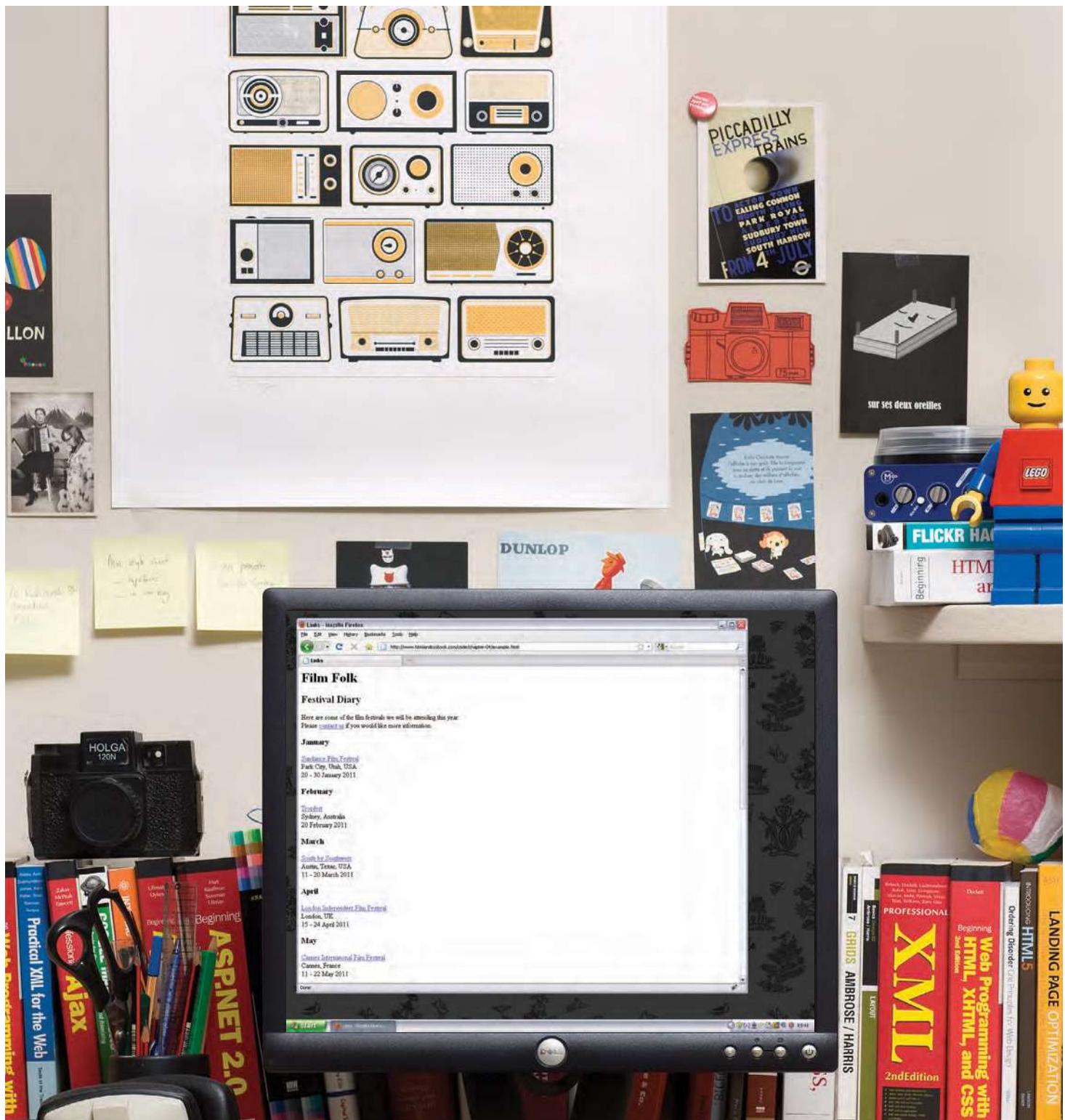
- ▶ Creating links between pages
- ▶ Linking to other sites
- ▶ Email links



Links are the defining feature of the web because they allow you to move from one web page to another — enabling the very idea of browsing or surfing.

You will commonly come across the following types of links:

- Links from one website to another
- Links from one page to another on the same website
- Links from one part of a web page to another part of the same page
- Links that open in a new browser window
- Links that start up your email program and address a new email to someone



WRITING LINKS

Links are created using the `<a>` element. Users can click on anything between the opening `<a>` tag and the closing `` tag. You specify which page you want to link to using the `href` attribute.

```
<a href="http://www.imdb.com">IMDB</a>
```

The diagram illustrates the structure of an anchor tag (`<a>`). It consists of four main parts:

- THIS IS THE PAGE THE LINK TAKES YOU TO**: Refers to the value of the `href` attribute, "http://www.imdb.com".
- THIS IS THE TEXT THE USER CLICKS ON**: Refers to the word "IMDB" which is the content between the opening and closing tags.
- OPENING LINK TAG**: The start of the tag, indicated by the left bracket of a horizontal brace under the `<a` tag.
- CLOSING LINK TAG**: The end of the tag, indicated by the right bracket of a horizontal brace under the closing `>` symbol.

The text between the opening <a> tag and closing tag is known as link text. Where possible, your link text should explain where visitors will be taken if they click on it (rather than just saying "click here"). Below you can see the link to IMDB that was created on the previous page.

Many people navigate websites by scanning the text for links. Clear link text can help visitors find what they want. This will give them a more positive impression of your site and may encourage them to visit it for longer. (It also helps people using screen reader software.)

To write good link text, you can think of words people might use when searching for the page that you are linking to. (For example, rather than write "places to stay" you could use something more specific such as "hotels in New York.")



LINKING TO OTHER SITES

<a>

Links are created using the `<a>` element which has an attribute called `href`. The value of the `href` attribute is the page that you want people to go to when they click on the link.

Users can click on anything that appears between the opening `<a>` tag and the closing `` tag and will be taken to the page specified in the `href` attribute.

When you link to a different website, the value of the `href` attribute will be the full web address for the site, which is known as an **absolute URL**.

Browsers show links in blue with an underline by default.

chapter-04/linking-to-other-sites.html

HTML

```
<p>Movie Reviews:<br/><ul><li><a href="http://www.empireonline.com">Empire</a></li><li><a href="http://www.metacritic.com">Metacritic</a></li><li><a href="http://www.rottentomatoes.com">Rotten Tomatoes</a></li><li><a href="http://www.variety.com">Variety</a></li></ul></p>
```

RESULT

Movie Reviews:

- [Empire](http://www.empireonline.com)
- [Metacritic](http://www.metacritic.com)
- [Rotten Tomatoes](http://www.rottentomatoes.com)
- [Variety](http://www.variety.com)

ABSOLUTE URLs

URL stands for Uniform Resource Locator. Every web page has its own URL. This is the web address that you would type into a browser if you wanted to visit that specific page.

An absolute URL starts with the domain name for that site, and can be followed by the path to a specific page. If no page is specified, the site will display the homepage.

LINKING TO OTHER PAGES ON THE SAME SITE

HTML

chapter-04/linking-to-other-pages.html

```
<p>
<ul>
<li><a href="index.html">Home</a></li>
<li><a href="about-us.html">About</a></li>
<li><a href="movies.html">Movies</a></li>
<li><a href="contact.html">Contact</a></li>
</ul>
</p>
```

RESULT

- [Home](#)
- [About](#)
- [Movies](#)
- [Contact](#)

<a>

When you are linking to other pages within the same site, you do not need to specify the domain name in the URL. You can use a shorthand known as a **relative URL**.

If all the pages of the site are in the same folder, then the value of the `href` attribute is just the name of the file.

If you have different pages of a site in different folders, then you can use a slightly more complex syntax to indicate where the page is in relation to the current page. You will learn more about these on the pages 81-84.

If you look at the download code for each chapter, you will see that the `index.html` file contains links that use relative URLs.

RELATIVE URLs

When linking to other pages within the same site, you can use relative URLs. These are like a shorthand version of absolute URLs because you do not need to specify the domain name.

We will take a closer look at relative URLs on pages 83-84 as there are several helpful shortcuts you can use to write links to other pages on your own website.

Relative URLs help when building a site on your computer because you can create links between pages without having to set up your domain name or hosting.

RELATIVE LINK TYPE

EXAMPLE (from diagram on previous page)

SAME FOLDER

To link to a file in the same folder, just use the file name. (Nothing else is needed.)

To link to music reviews from the music homepage:
`Reviews`

CHILD FOLDER

For a child folder, use the name of the child folder, followed by a forward slash, then the file name.

To link to music listings from the homepage:
`Listings`

GRANDCHILD FOLDER

Use the name of the child folder, followed by a forward slash, then the name of the grandchild folder, followed by another forward slash, then the file name.

To link to DVD reviews from the homepage:
`Reviews`

PARENT FOLDER

Use `../` to indicate the folder above the current one, then follow it with the file name.

To link to the homepage from the music reviews:
`Home`

GRANDPARENT FOLDER

Repeat the `..../` to indicate that you want to go up two folders (rather than one), then follow it with the file name.

To link to the homepage from the DVD reviews:
`Home`

When a website is live (that is, uploaded to a web server) you may see a couple of other techniques used that do not work when the files are on your local computer.

For example, you may see the name of a child folder without the name of a file. In this case the web server will usually try to show the homepage for that section.

A forward slash will return the homepage for the entire site, and a forward slash followed by a file name will return that file providing it is in the root directory.

OPENING LINKS IN A NEW WINDOW

HTML

chapter-04/opening-links-in-a-new-window.html

```
<a href="http://www.imdb.com" target="_blank">  
Internet Movie Database</a> (opens in new window)
```

RESULT

[Internet Movie Database](http://www.imdb.com) (opens in new window)

target

If you want a link to open in a new window, you can use the `target` attribute on the opening `a` tag. The value of this attribute should be `_blank`.

One of the most common reasons a web page author might want a link to be opened in a new window is if it points to another website. In such cases, they hope the user will return to the window containing their site after finishing looking at the other one.

Generally you should avoid opening links in a new window, but if you do, it is considered good practice to inform users that the link will open a new window before they click on it.

LINKING TO A SPECIFIC PART OF THE SAME PAGE

At the top of a long page you might want to add a list of contents that links to the corresponding sections lower down. Or you might want to add a link from part way down the page back to the top of it to save users from having to scroll back to the top.

Before you can link to a specific part of a page, you need to identify the points in the page that the link will go to. You do this using the `id` attribute (which can be used on every HTML element). You can see that the `<h1>` and `<h2>` elements in this example have been given `id` attributes that identify those sections of the page.

The value of the `id` attribute should start with a letter or an underscore (not a number or any other character) and, on a single page, no two `id` attributes should have the same value.

To link to an element that uses an `id` attribute you use the `<a>` element again, but the value of the `href` attribute starts with the `#` symbol, followed by the value of the `id` attribute of the element you want to link to. In this example, `` links to the `<h1>` element at the top of the page whose `id` attribute has a value of `top`.

chapter-05/linking-to-a-specific-part.html

HTML

```
<h1 id="top">Film-Making Terms</h1>
<a href="#arc_shot">Arc Shot</a><br />
<a href="#interlude">Interlude</a><br />
<a href="#prologue">Prologue</a><br /><br />
<h2 id="arc_shot">Arc Shot</h2>
<p>A shot in which the subject is photographed by an encircling or moving camera</p>
<h2 id="interlude">Interlude</h2>
<p>A brief, intervening film scene or sequence, not specifically tied to the plot, that appears within a film</p>
<h2 id="prologue">Prologue</h2>
<p>A speech, preface, introduction, or brief scene preceding the the main action or plot of a film; contrast to epilogue</p>
<p><a href="#top">Top</a></p>
```

LINKING TO A SPECIFIC PART OF ANOTHER PAGE

RESULT

Film-Making Terms

[Arc Shot](#)
[Interlude](#)
[Prologue](#)

Arc Shot

A shot in which the subject is photographed by an encircling or moving camera

Interlude

A brief, intervening film scene or sequence, not specifically tied to the plot, that appears within a film

Prologue

A speech, preface, introduction, or brief scene preceding the main action or plot of a film; contrast to epilogue

[Top](#)

If you want to link to a specific part of a different page (whether on your own site or a different website) you can use a similar technique.

As long as the page you are linking to has id attributes that identify specific parts of the page, you can simply add the same syntax to the end of the link for that page.

Therefore, the href attribute will contain the address for the page (either an absolute URL or a relative URL), followed by the # symbol, followed by the value of the id attribute that is used on the element you are linking to.

For example, to link to the bottom of the homepage of the website that accompanies this book, you would write:

```
<a href="http://www.htmlandcssbook.com/#bottom">
```

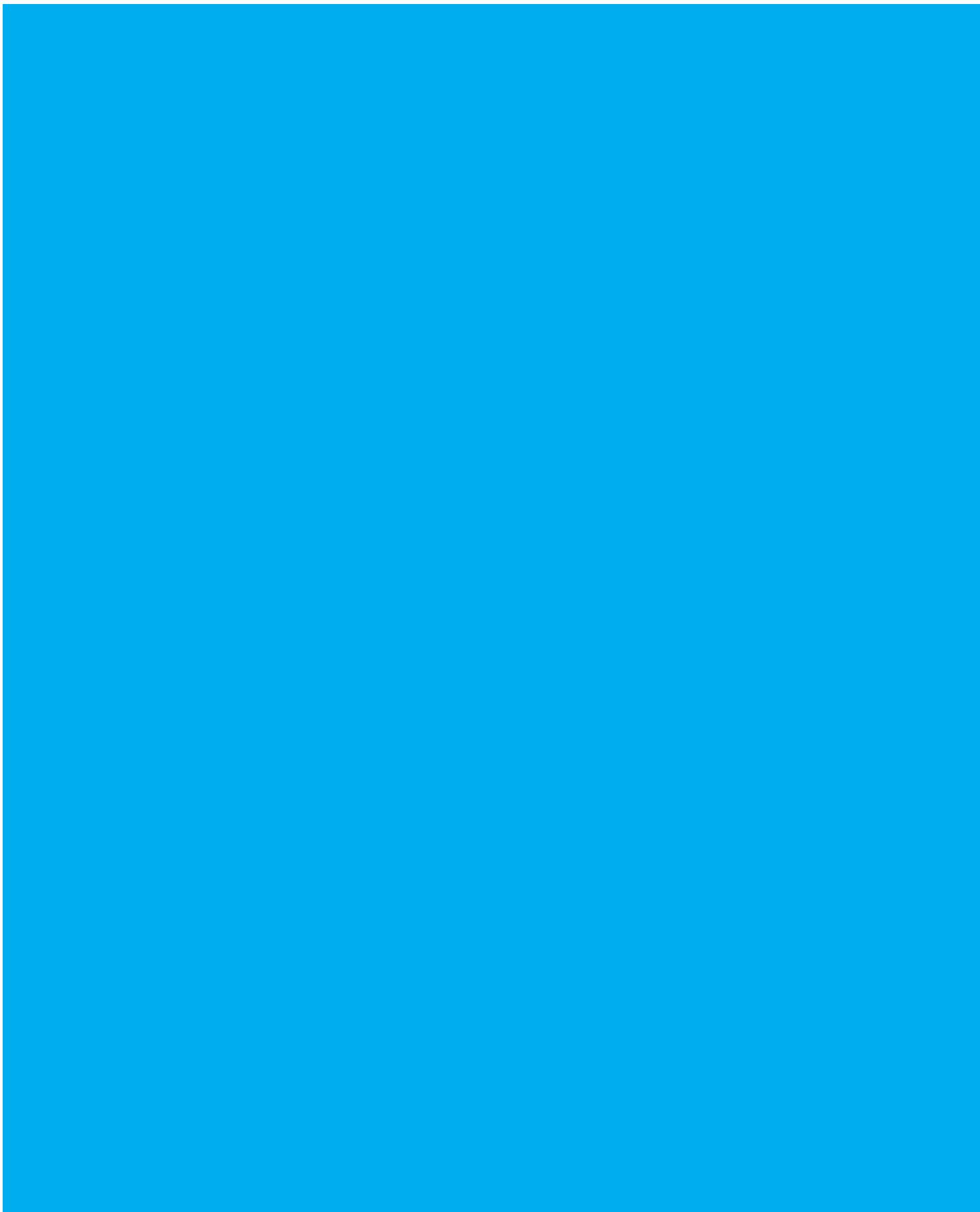
EXAMPLE LINKS

This example is of a web page about film.

The `<h1>` element is used with an `id` attribute at the top of the page so that a link can be added to take readers from the bottom of the page to the top. There is an email link to allow readers to contact the author of the web page. There are also a number of links to qualified URLs. These link to various film festivals. Below this list is a link to a relative URL which is an "about" page that lives in the same directory.

```
<html>
  <head>
    <title>Links</title>
  </head>
  <body>
    <h1 id="top">Film Folk</h1>
    <h2>Festival Diary</h2>
    <p>Here are some of the film festivals we
       will be attending this year.<br />Please
       <a href="mailto:filmfolk@example.org">
       contact us</a> if you would like more
       information.</p>
    <h3>January</h3>
    <p><a href="http://www.sundance.org">
       Sundance Film Festival</a><br />
       Park City, Utah, USA<br />
       20 - 30 January 2011</p>
    <h3>February</h3>
    <p><a href="http://www.tropfest.com">
       Tropfest</a><br />
       Sydney, Australia<br />
       20 February 2011</p>
    <!-- additional content -->
    <p><a href="about.html">About Film Folk</a></p>
    <p><a href="#top">Top of page</a></p>
  </body>
</html>
```

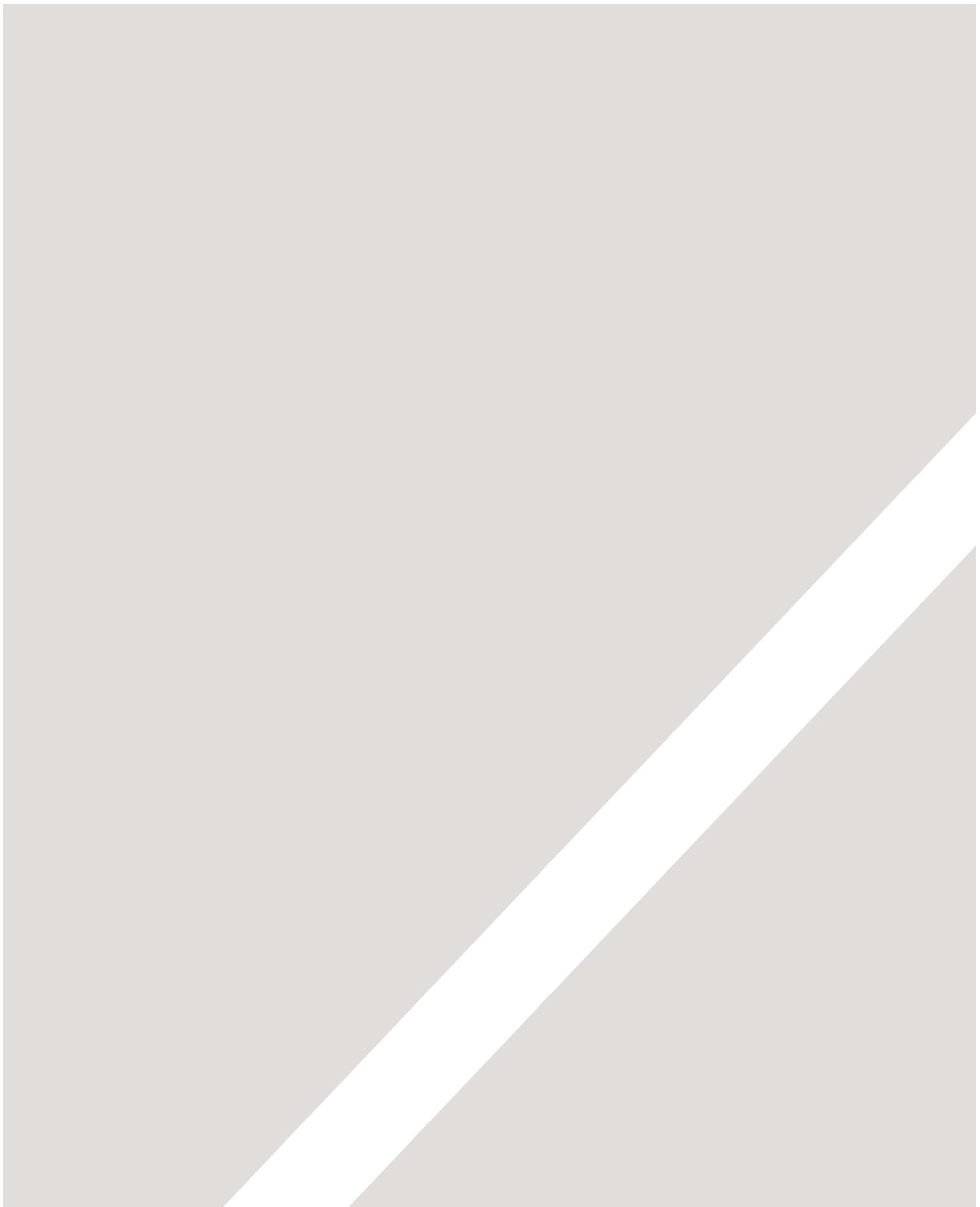




SUMMARY

LINKS

- ▶ Links are created using the `<a>` element.
- ▶ The `<a>` element uses the `href` attribute to indicate the page you are linking to.
- ▶ If you are linking to a page within your own site, it is best to use relative links rather than qualified URLs.
- ▶ You can create links to open email programs with an email address in the "to" field.
- ▶ You can use the `id` attribute to target elements within a page that can be linked to.



5

IMAGES

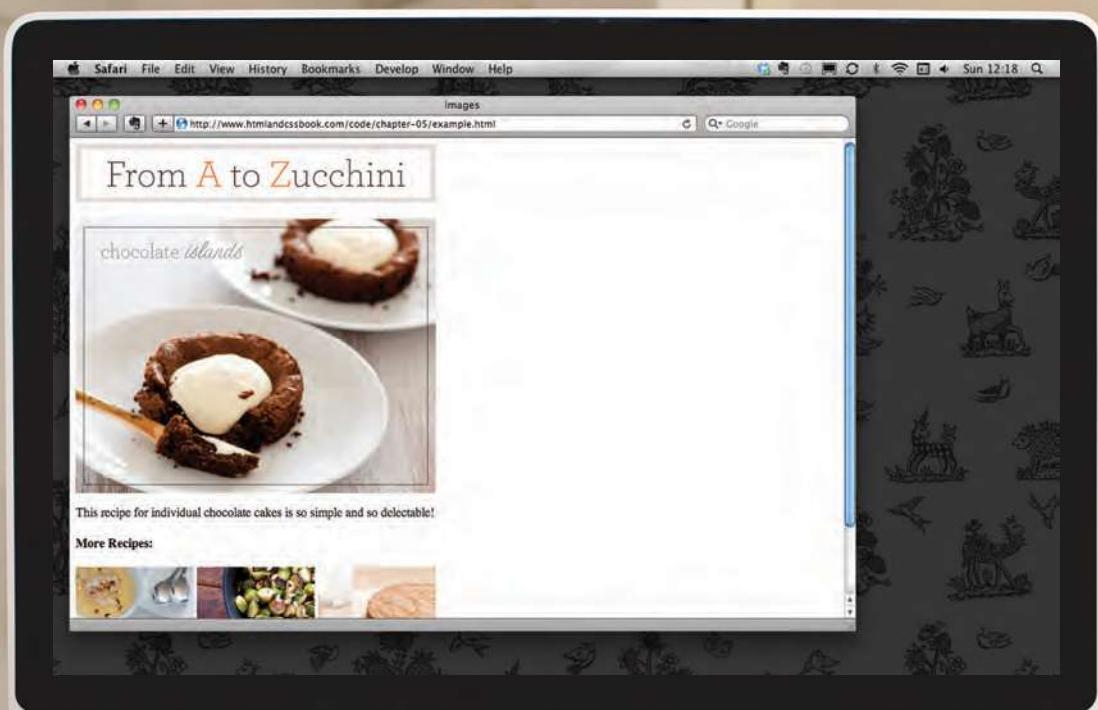
- ▶ How to add images to pages
- ▶ Choosing the right format
- ▶ Optimizing images for the web

There are many reasons why you might want to add an image to a web page: you might want to include a logo, photograph, illustration, diagram, or chart.

There are several things to consider when selecting and preparing images for your site, but taking time to get them right will make it look more attractive and professional. In this chapter you will learn how to:

- Include an image in your web pages using HTML
- Pick which image format to use
- Show an image at the right size
- Optimize an image for use on the web to make pages load faster

You can also use CSS to include images in your pages using the `background-image` property, which you will meet on pages 413-420.



ADDING IMAGES

To add an image into the page you need to use an element. This is an empty element (which means there is no closing tag). It must carry the following two attributes:

src

This tells the browser where it can find the image file. This will usually be a relative URL pointing to an image on your own site. (Here you can see that the images are in a child folder called **images** — relative URLs were covered on pages 83-84).

alt

This provides a text description of the image which describes the image if you cannot see it.

title

You can also use the title attribute with the element to provide additional information about the image. Most browsers will display the content of this attribute in a tooltip when the user hovers over the image.

chapter-05/adding-images.html

HTML

```

```

RESULT



The text used in the alt attribute is often referred to as **alt text**. It should give an accurate description of the image content so it can be understood by screen reader software (used by people with visual impairments) and search engines.

If the image is just to make a page look more attractive (and it has no meaning, such as a graphic dividing line), then the alt attribute should still be used but the quotes should be left empty.

HEIGHT & WIDTH OF IMAGES

HTML

[chapter-05/height-and-width-of-images.html](#)

```

```

RESULT



You will also often see an `` element use two other attributes that specify its size:

height

This specifies the height of the image in pixels.

width

This specifies the width of the image in pixels.

Images often take longer to load than the HTML code that makes up the rest of the page. It is, therefore, a good idea to specify the size of the image so that the browser can render the rest of the text on the page while leaving the right amount of space for the image that is still loading.

The size of images is increasingly being specified using CSS rather than HTML — see pages 409-410 for more information about this.

WHERE TO PLACE IMAGES IN YOUR CODE

Where an image is placed in the code will affect how it is displayed. Here are three examples of image placement that produce different results:

1: BEFORE A PARAGRAPH

The paragraph starts on a new line after the image.

2: INSIDE THE START OF A PARAGRAPH

The first row of text aligns with the bottom of the image.

3: IN THE MIDDLE OF A PARAGRAPH

The image is placed between the words of the paragraph that it appears in.

chapter-05/where-to-place-images.html

HTML

```

<p>There are around 10,000 living species of birds
    that inhabit different ecosystems from the
    Arctic to the Antarctic. Many species undertake
    long distance annual migrations, and many more
    perform shorter irregular journeys.</p>
<hr />
<p>There are around 10,000 living
    species of birds that inhabit different
    ecosystems from the Arctic to the Antarctic. Many
    species undertake long distance annual
    migrations, and many more perform shorter
    irregular journeys.</p>
<hr />
<p>There are around 10,000 living species of birds
    that inhabit different ecosystems from the
    Arctic to the Antarctic.Many species undertake long
    distance annual migrations, and many more perform
    shorter irregular journeys.</p>
```

RESULT



There are around 10,000 living species of birds that inhabit different ecosystems from the Arctic to the Antarctic. Many species undertake long distance annual migrations, and many more perform shorter irregular journeys.



There are around 10,000 living species of birds that inhabit different ecosystems from the Arctic to the Antarctic. Many species undertake long distance annual migrations, and many more perform shorter irregular journeys.

There are around 10,000 living species of birds that inhabit different



ecosystems from the Arctic to the Antarctic. Many species undertake long distance annual migrations, and many more perform shorter irregular journeys.

Where you place the image in the code is important because browsers show HTML elements in one of two ways:

Block elements always appear on a new line. Examples of block elements include the `<h1>` and `<p>` elements.

If the `` is followed by a block level element (such as a paragraph) then the block level element will sit on a new line after the image as shown in the first example on this page.

Inline elements sit within a block level element and do not start on a new line. Examples of inline elements include the ``, ``, and `` elements.

If the `` element is inside a block level element, any text or other inline elements will flow around the image as shown in the second and third examples on this page.

Block and inline elements are discussed in greater depth on pages 185-186.

OLD CODE: ALIGNING IMAGES HORIZONTALLY

align

The align attribute was commonly used to indicate how the other parts of a page should flow around an image. It has been removed from HTML5 and new websites should use CSS to control the alignment of images (as you will see on pages 411-412).

I have discussed it here because you are likely to come across it if you look at older code, and because some visual editors still insert this attribute when you indicate how an image should be aligned.

The align attribute can take these horizontal values:

left

This aligns the image to the left (allowing text to flow around its right-hand side).

right

This aligns the image to the right (allowing text to flow around its left-hand side).

chapter-05/aligning-images-horizontally.html

HTML

```
<p>There are around 10,000 living species of birds that inhabit different ecosystems from the Arctic to the Antarctic. Many species undertake long distance annual migrations, and many more perform shorter irregular journeys.</p>
<hr />
<p>There are around 10,000 living species of birds that inhabit different ecosystems from the Arctic to the Antarctic. Many species undertake long distance annual migrations, and many more perform shorter irregular journeys.</p>
```

RESULT



There are around 10,000 living species of birds that inhabit different ecosystems from the Arctic to the Antarctic. Many species undertake long distance annual migrations, and many more perform shorter irregular journeys.

There are around 10,000 living species of birds that inhabit different ecosystems from the Arctic to the Antarctic. Many species undertake long distance annual migrations, and many more perform shorter irregular journeys.



This looks a lot neater than having one line of text next to the image (as shown on the previous example).

When you give the align attribute a value of left, the image is placed on the left and text flows around it.

When you give the align attribute a value of right, the image is placed on the right and the text flows around it.

When text flows right up to the edge of an image it can make it harder to read. You will learn how to add a gap between text and images on pages 313-314 using the CSS padding and margin properties.

OLD CODE: ALIGNING IMAGES VERTICALLY

As you saw on the last page, the align attribute is no longer used in HTML5, but it is covered here because you may see it used in older websites and it is still used in the code created by some visual editors.

You can see how to use CSS to achieve the same effects on pages 285-286.

There are three values that the align attribute can take that control how the image should align vertically with the text that surrounds it:

top

This aligns the first line of the surrounding text with the top of the image.

middle

This aligns the first line of the surrounding text with the middle of the image.

bottom

This aligns the first line of the surrounding text with the bottom of the image.

chapter-05/aligning-images-vertically.html

HTML

```
<p>There are around
       10,000 living species of birds that inhabit
       different ecosystems from the Arctic to the
       Antarctic. Many species undertake long distance
       annual migrations, and many more perform shorter
       irregular journeys.</p>
<hr />
<p>There are around
       10,000 living species of birds that inhabit
       different ecosystems from the Arctic to the
       Antarctic. Many species undertake long distance
       annual migrations, and many more perform shorter
       irregular journeys.</p>
<hr />
<p>There are around
       10,000 living species of birds that inhabit
       different ecosystems from the Arctic to the
       Antarctic. Many species undertake long distance
       annual migrations, and many more perform shorter
       irregular journeys.</p>
```

RESULT



There are around 10,000 living species of birds that

inhabit different ecosystems from the Arctic to the Antarctic. Many species undertake long distance annual migrations, and many more perform shorter irregular journeys.



There are around 10,000 living species of birds that

inhabit different ecosystems from the Arctic to the Antarctic. Many species undertake long distance annual migrations, and many more perform shorter irregular journeys.



There are around 10,000 living species of birds that

inhabit different ecosystems from the Arctic to the Antarctic. Many species undertake long distance annual migrations, and many more perform shorter irregular journeys.

The value of `top` places the first line of text near the top of the image and subsequent lines of text appear under the image.

The value of `middle` places the first line of text near the vertical middle of the image and subsequent lines of text appear under the image.

The value of `bottom` places the first line of text near the bottom of the image and subsequent lines of text under the image.

When text flows right up to the edge of an image it can make it harder to read. You will learn how to add a gap between text and images on pages 313-314 using the CSS padding and margin properties.

If you would like all of the text to wrap around the image (rather than just one line of text), you should use the CSS `float` property discussed on pages 370-372.

In older code, you may see the `align` attribute used with the values `left` or `right` to achieve the same effect (as described on the previous page), although its use is no longer recommended.

THREE RULES FOR CREATING IMAGES

There are three rules to remember when you are creating images for your website which are summarized below. We go into greater detail on each topic over the next nine pages.

1

SAVE IMAGES IN THE RIGHT FORMAT

Websites mainly use images in jpeg, gif, or png format. If you choose the wrong image format then your image might not look as sharp as it should and can make the web page slower to load.

2

SAVE IMAGES AT THE RIGHT SIZE

You should save the image at the same width and height it will appear on the website. If the image is smaller than the width or height that you have specified, the image can be distorted and stretched. If the image is larger than the width and height if you have specified, the image will take longer to display on the page.

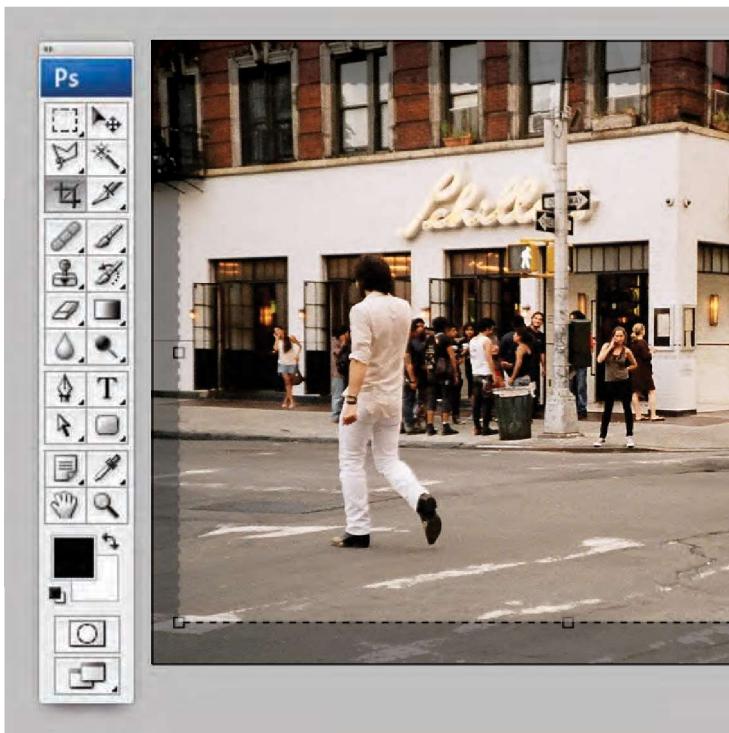
3

USE THE CORRECT RESOLUTION

Computer screens are made up of dots known as pixels. Images used on the web are also made up of tiny dots. Resolution refers to the number of dots per inch, and most computer screens only show web pages at 72 pixels per inch. So saving images at a higher resolution results in images that are larger than necessary and take longer to download.

TOOLS TO EDIT & SAVE IMAGES

There are several tools you can use to edit and save images to ensure that they are the right size, format, and resolution.



The most popular tool amongst web professionals is **Adobe Photoshop**. (In fact, professional web designers often use this software to design entire sites.) The full version of Photoshop is expensive, but there is a cheaper version called Photoshop Elements which would suit the needs of most beginners.

OTHER SOFTWARE

Adobe Fireworks
Pixelmator
PaintShop Pro
Paint.net

ONLINE EDITORS

www.photoshop.com
www.pixlr.com
www.splashup.com
www.ipiccy.com

ONLINE EXTRA

Watch videos that demonstrate how to resize images and save them in the correct format using both of these applications.

IMAGE FORMATS: JPEG

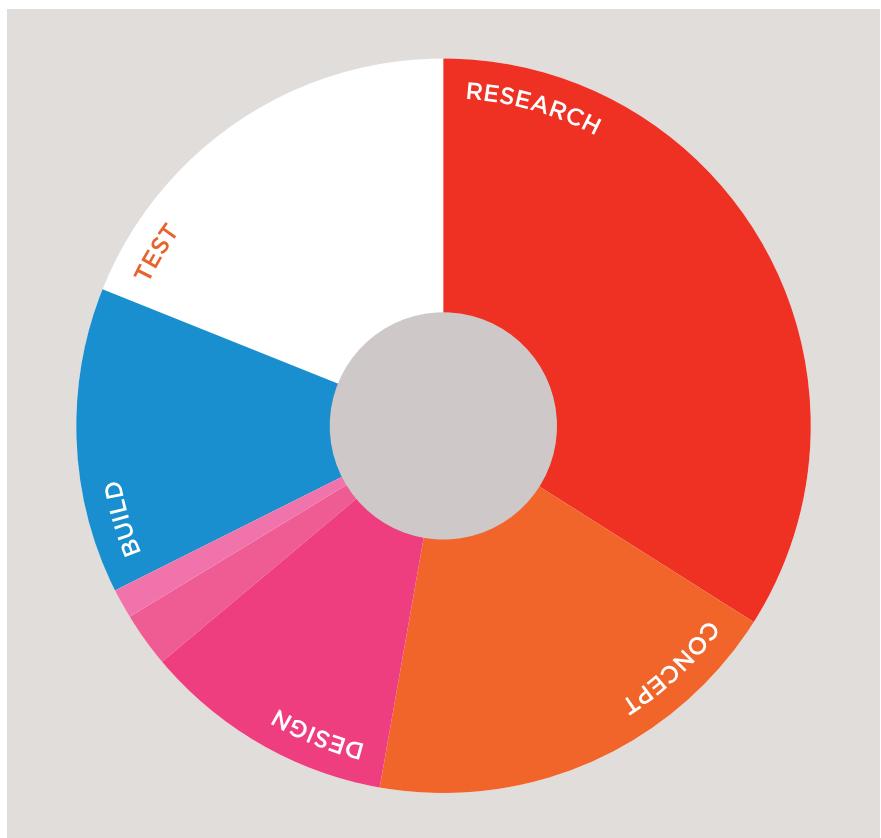
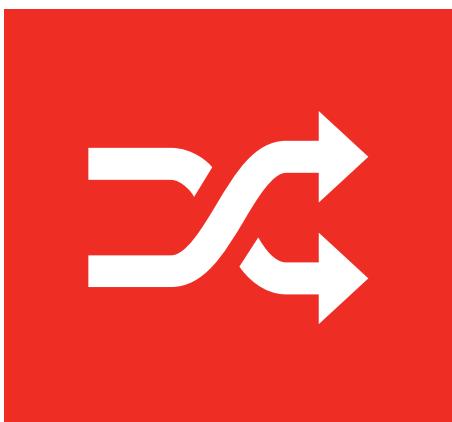


Whenever you have many different colors in a picture you should use a JPEG. A photograph that features snow or an overcast sky might look like it has large areas that are just white or gray, but the picture is usually made up of many different colors that are subtly different.

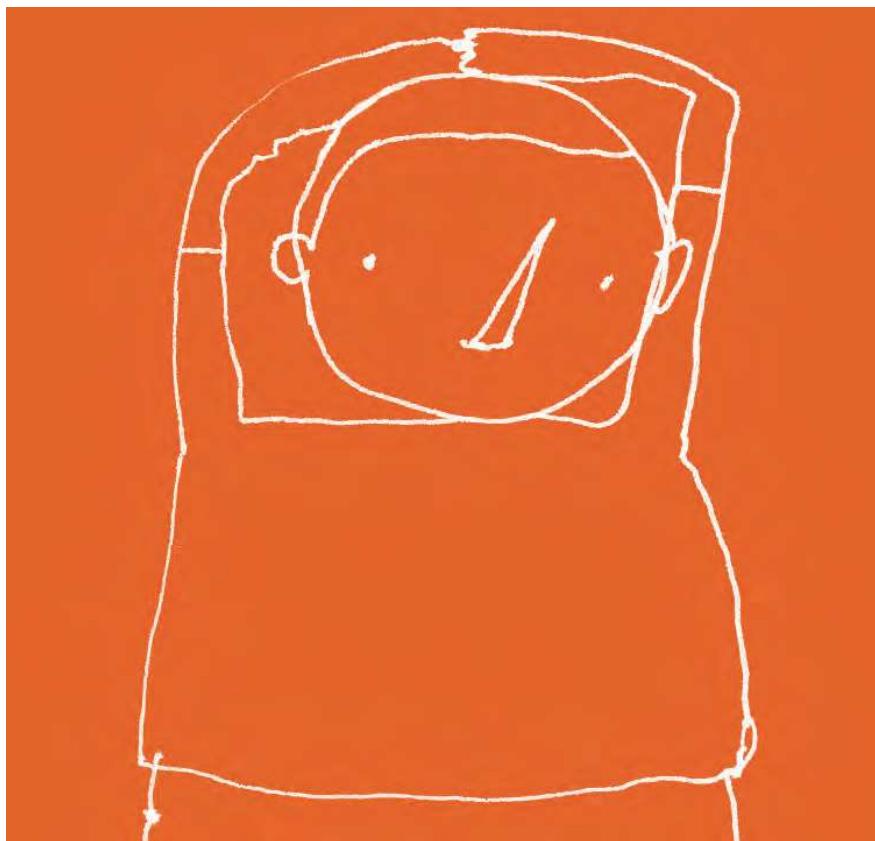




IMAGE FORMATS: GIF



Use GIF or PNG format when saving images with few colors or large areas of the same color.



When a picture has an area that is filled with exactly the same color, it is known as flat color. Logos, illustrations, and diagrams often use flat colors. (Note that photographs of snow, sky, or grass are not flat colors, they are made up of many subtly different shades of the same color and are not as suited to GIF or PNG format.)

EXAMINING IMAGES ON THE WEB

CHECKING THE SIZE OF IMAGES

If you are updating a website, you might need to check the size of an existing image before creating a new one to replace it. This can be achieved by right-clicking on the image and making a selection from the pop-up menu that appears. (Mac users will need to hold down the control key and click rather than right-click.)



DOWNLOADING IMAGES

If you want to download images from a website, you can do so by accessing the same pop-up menu. (Please remember however that all images online are subject to copyright and require explicit permission to reuse.)



On the left you can see how to check the size of images and how to download them using Safari. Below is a brief overview of what to select in the pop-up menu to perform these functions in various browsers.

CHROME

Size: **Open Image in New Tab**

Size appears in new tab

Download: **Save Image As**

FIREFOX

Size: **View Image Info**

Size appears in pop-up window

Download: **Save Image As**

INTERNET EXPLORER

Size: **Properties**

Size appears in pop-up window

Download: **Save Image**

SAFARI

Size: **Open Image in New Tab**

Size appears in title bar

Download: **Save Image As**

HTML5: FIGURE AND FIGURE CAPTION

HTML

chapter-05/figure-and-figure-caption.html

```
<figure>
  
  <br />
  <figcaption>Sea otters hold hands when they
    sleep so they don't drift away from each
    other.</figcaption>
</figure>
```

RESULT



Sea otters hold hands when they sleep so they don't drift away from each other.

<figure>

Images often come with captions. HTML5 has introduced a new `<figure>` element to contain images and their caption so that the two are associated.

You can have more than one image inside the `<figure>` element as long as they all share the same caption.

<figcaption>

The `<figcaption>` element has been added to HTML5 in order to allow web page authors to add a caption to an image.

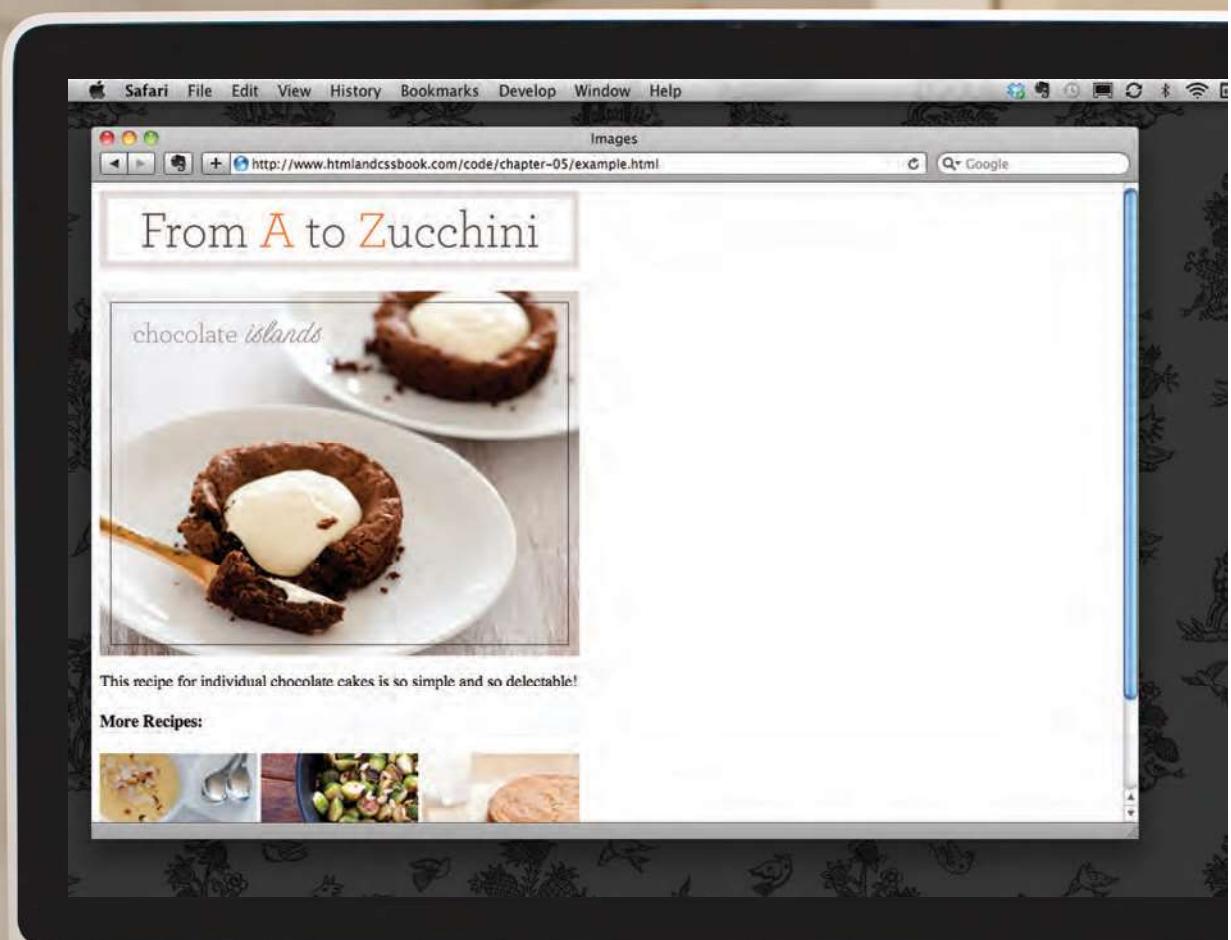
Before these elements were created there was no way to associate an `` element with its caption.

Older browsers that do not understand HTML5 elements simply ignore the new elements and display the content of them.

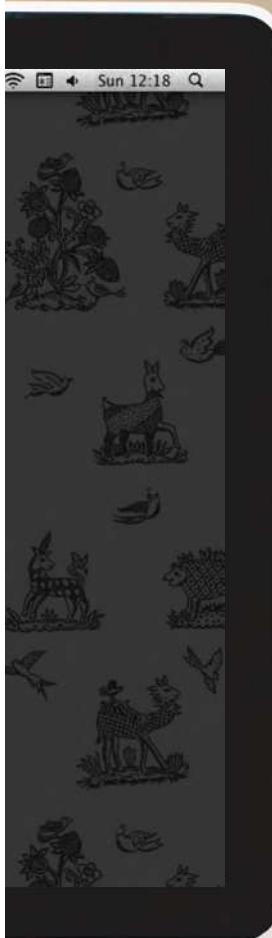
In this example, the logo is a GIF because it uses flat colors, while the photographs are JPEGs. The main photo is placed inside the HTML5 `<figure>` element and has its own caption.

The `alt` attribute on each image provides a description for those using screen readers and the `title` attribute provides additional information. (This is shown in the tooltip.)

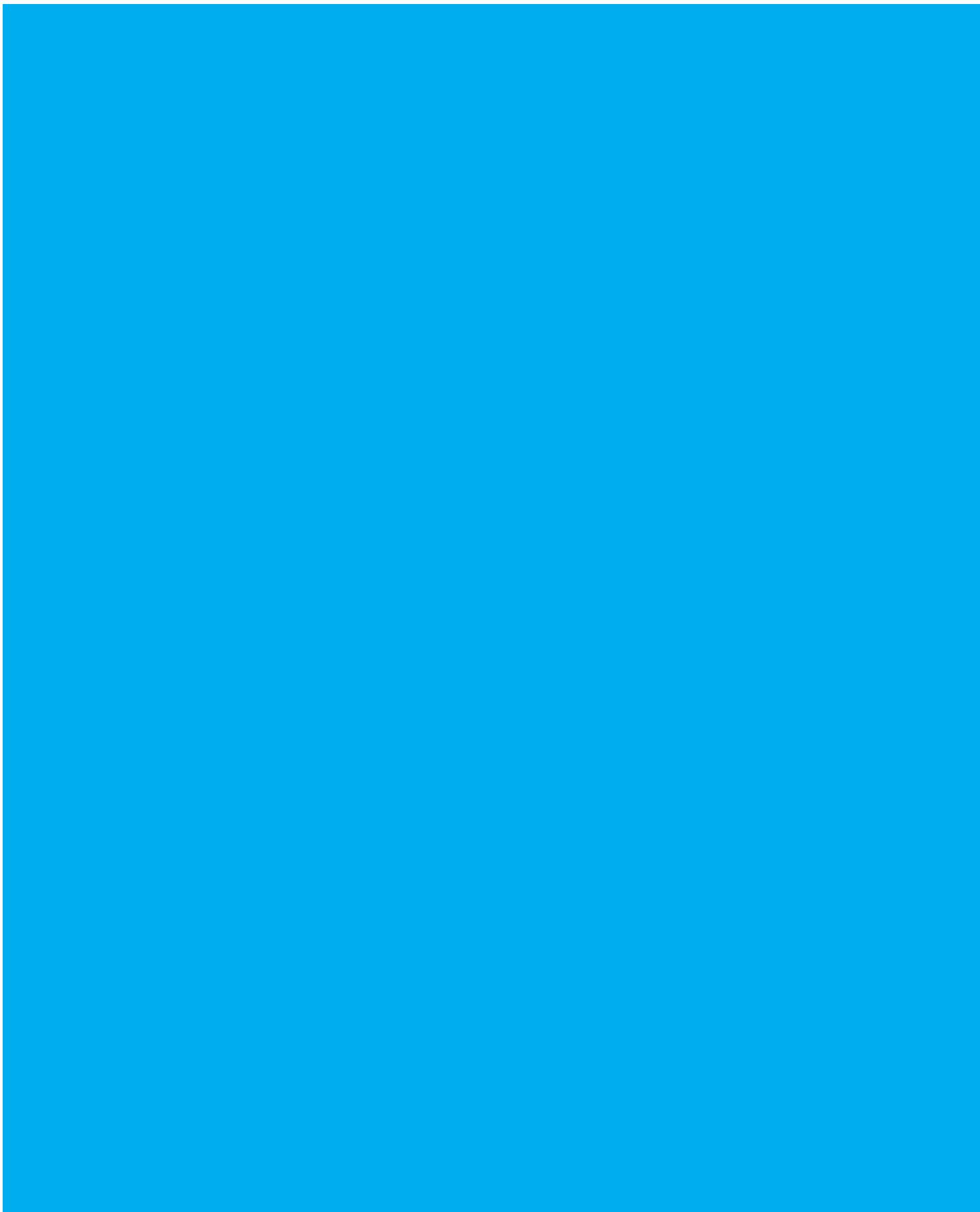
This example does not use the `height`, `width`, or `align` attributes as these are being phased out and you are encouraged to use CSS properties instead.



EXAMPLE IMAGES



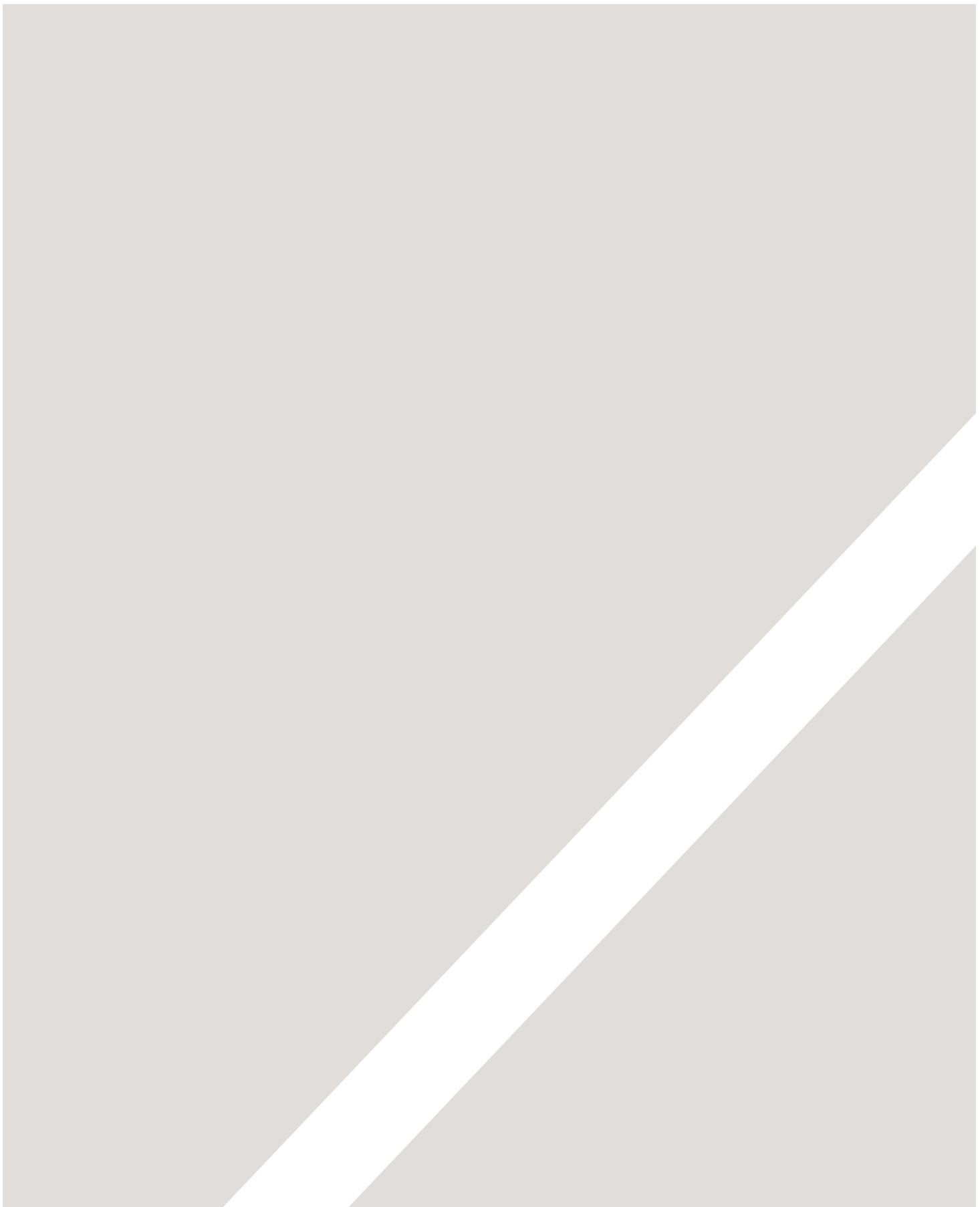
```
<html>
  <head>
    <title>Images</title>
  </head>
  <body>
    <h1>
      
    </h1>
    <figure>
      
      <p>
        <figcaption>
          This recipe for individual chocolate
          cakes is so simple and so delectable!
        </figcaption>
      </p>
    </figure>
    <h4>More Recipes:</h4>
    <p>
      
      
      
    </p>
  </body>
</html>
```



SUMMARY

IMAGES

- ▶ The `` element is used to add images to a web page.
- ▶ You must always specify a `src` attribute to indicate the source of an image and an `alt` attribute to describe the content of an image.
- ▶ You should save images at the size you will be using them on the web page and in the appropriate format.
- ▶ Photographs are best saved as JPEGs; illustrations or logos that use flat colors are better saved as GIFs.



6

TABLES

- ▶ How to create tables
- ▶ What information suits tables
- ▶ How to represent complex data in tables

There are several types of information that need to be displayed in a grid or table. For example: sports results, stock reports, train timetables.

When representing information in a table, you need to think in terms of a grid made up of rows and columns (a bit like a spreadsheet). In this chapter you will learn how to:

- Use the four key elements for creating tables
- Represent complex data using tables
- Add captions to tables



Adult fares	Pay as you go	
	Peak*	Off-Peak**
Zones 1 only	E2.00	E1.50
Zones 1-2	E2.20	E1.70
Zones 1-3	E2.80	E2.00
Zones 1-4	E3.30	E2.30
Zones 1-5	E4.20	E2.60
Zones 1-6	E5.20	E3.20
Zones 2, 3, 4, 5 or 6	E1.50	E1.30
Zones 2-3, 3-4, 4-5, or 5-6	E1.90	E1.50
Zones 2-4, 3-5 or 4-6	E2.30	E1.70
Zones 2-5 or 3-6	E2.90	E2.00
Zones 2-6	E3.40	E2.20

*Peak Oyster single fares apply from 0630 to 0930 and from 1600 to 1900 Monday to Friday (excluding public holidays).

**Off-Peak Oyster single fare applies at all other times.

Using Oyster on National Rail

Pay as you go and Travelcard season tickets on Oyster can be used on all National Rail services in London in the zones paid for except on:

- Express
- Direct between Heathrow and Central London
- Direct between Central London and the South Coast

No time restriction
When you need to make a claim, of course we hope your pet gets better quickly. But if they don't, with M&S Pet Insurance there's no need to worry. Because there is no time limit on treatment. As long as you renew your policy and we can continue to offer cover, your vet's fees will be covered up to the limit stated. For full details of your cover have a look at your policy booklet. Keep our claims helpline number safe in case you need it, 0800 080 8750.

Ask about the 5% discount offered on premiums for additional pets

Benefits

Veterinary fees for illness and injury

- Definitive care

- Prescription food

- Complementary treatments

- Vet callouts

Limit for each illness/injury

Admission for missing pet

Rewards

tokens or missing pet

Emergency boarding (pet minder)

From injury

Healthcare

WHAT'S A TABLE?

A table represents information in a grid format. Examples of tables include financial reports, TV schedules, and sports results.

Grids allow us to understand complex data by referencing information on two axes.

Each block in the grid is referred to as a **table cell**. In HTML a table is written out row by row.

The screenshot shows the Reuters homepage with a focus on the 'Commodities' section. The top navigation bar includes links for U.S., News & Markets, Sectors & Industries, Analysis & Opinion, Register, Sign In, and a search bar. The main content area features a large grid for the Thomson Reuters/Jeffries CRB Index (TR/J CRB). The grid includes columns for Change (-3.36), Open (360.92), High (361.19), Low (357.99), and Times (04/18 14:58). Below this is a section for Commodity Futures, listing categories like Energy, Metals, Grains, Oilseeds, Softs, and Livestock. Another grid for the Thomson Reuters Equal Weight Continuous Commodities Index (CCI) shows data for Hogs, Oil, and Crude Oil. To the right, there's a 'MARKETS' sidebar with sections for Market Indices (DOW, S&P 500, NASDAQ, TR US INDEX), Currencies (EUR/USD, GBP/USD, USD/JPY), and Commodities (GOLD). Each of these sidebar sections also contains a grid of data rows.

BASIC TABLE STRUCTURE

<table>

The <table> element is used to create a table. The contents of the table are written out row by row.

<tr>

You indicate the start of each row using the opening <tr> tag. (The tr stands for table row.)

It is followed by one or more <td> elements (one for each cell in that row).

At the end of the row you use a closing </tr> tag.

<td>

Each cell of a table is represented using a <td> element. (The td stands for table data.)

At the end of each cell you use a closing </td> tag.

Some browsers automatically draw lines around the table and/or the individual cells. You will learn how to control the borders of tables using CSS on pages 309-312 and 337-340.

chapter-06/basic-table-structure.html

HTML

```
<table>
  <tr>
    <td>15</td>
    <td>15</td>
    <td>30</td>
  </tr>
  <tr>
    <td>45</td>
    <td>60</td>
    <td>45</td>
  </tr>
  <tr>
    <td>60</td>
    <td>90</td>
    <td>90</td>
  </tr>
</table>
```

RESULT

15	15	30
45	60	45
60	90	90

TABLE HEADINGS

HTML

chapter-06/table-headings.html

```
<table>
  <tr>
    <th></th>
    <th scope="col">Saturday</th>
    <th scope="col">Sunday</th>
  </tr>
  <tr>
    <th scope="row">Tickets sold:</th>
    <td>120</td>
    <td>135</td>
  </tr>
  <tr>
    <th scope="row">Total sales:</th>
    <td>$600</td>
    <td>$675</td>
  </tr>
</table>
```

RESULT

Saturday	Sunday
Tickets sold: 120	135
Total sales: \$600	\$675

<th>

The `<th>` element is used just like the `<td>` element but its purpose is to represent the heading for either a column or a row. (The `th` stands for table heading.)

Even if a cell has no content, you should still use a `<td>` or `<th>` element to represent the presence of an empty cell otherwise the table will not render correctly. (The first cell in the first row of this example shows an empty cell.)

Using `<th>` elements for headings helps people who use screen readers, improves the ability for search engines to index your pages, and also enables you to control the appearance of tables better when you start to use CSS.

You can use the `scope` attribute on the `<th>` element to indicate whether it is a heading for a column or a row. It can take the values: `row` to indicate a heading for a row or `col` to indicate a heading for a column.

Browsers usually display the content of a `<th>` element in bold and in the middle of the cell.

SPANNING COLUMNS

Sometimes you may need the entries in a table to stretch across more than one column.

The `colspan` attribute can be used on a `<th>` or `<td>` element and indicates how many columns that cell should run across.

In the example on the right you can see a timetable with five columns; the first column contains the heading for that row (the day), the remaining four represent one hour time slots.

If you look at the table cell that contains the words 'Geography' you will see that the value of the `colspan` attribute is 2, which indicates that the cell should run across two columns. In the third row, 'Gym' runs across three columns.

You can see that the second and third rows have fewer `<td>` elements than there are columns. This is because, when a cell extends across more than one column, the `<td>` or `<th>` cells that would have been in the place of the wider cells are not included in the code.

I added some CSS styles to this example so that you can see how the cells span more than one column. You will learn how to do this on pages 250, 337-340.

chapter-06/spanning-columns.html

HTML

```
<table>
  <tr>
    <th></th>
    <th>9am</th>
    <th>10am</th>
    <th>11am</th>
    <th>12am</th>
  </tr>
  <tr>
    <th>Monday</th>
    <td colspan="2">Geography</td>
    <td>Math</td>
    <td>Art</td>
  </tr>
  <tr>
    <th>Tuesday</th>
    <td colspan="3">Gym</td>
    <td>Home Ec</td>
  </tr>
</table>
```

RESULT

	9am	10am	11am	12am
Monday	Geography		Math	Art
Tuesday	Gym			Home Ec

SPANNING ROWS

HTML

chapter-06/spanning-rows.html

```
<table>
  <tr>
    <th></th>
    <th>ABC</th>
    <th>BBC</th>
    <th>CNN</th>
  </tr>
  <tr>
    <th>6pm - 7pm</th>
    <td rowspan="2">Movie</td>
    <td>Comedy</td>
    <td>News</td>
  </tr>
  <tr>
    <th>7pm - 8pm</th>
    <td>Sport</td>
    <td>Current Affairs</td>
  </tr>
</table>
```

RESULT

	ABC	BBC	CNN
6pm - 7pm	Movie	Comedy	News
7pm - 8pm		Sport	Current Affairs

You may also need entries in a table to stretch down across more than one row.

The `rowspan` attribute can be used on a `<th>` or `<td>` element to indicate how many rows a cell should span down the table.

In the example on the left you can see that ABC is showing a movie from 6pm - 8pm, whereas the BBC and CNN channels are both showing two programs during this time period (each of which lasts one hour).

If you look at the last `<tr>` element, it only contains three elements even though there are four columns in the result below. This is because the movie in the `<tr>` element above it uses the `rowspan` attribute to stretch down and take over the cell below.

I have added some CSS styles to this example so that you can see how the cells span more than one row. You will learn how to apply these CSS styles to tables on pages 250, 337-340.

LONG TABLES

There are three elements that help distinguish between the main content of the table and the first and last rows (which can contain different content).

These elements help people who use screen readers and also allow you to style these sections in a different manner than the rest of the table (as you will see when you learn about CSS).

<thead>

The headings of the table should sit inside the <thead> element.

<tbody>

The body should sit inside the <tbody> element.

<tfoot>

The footer belongs inside the <tfoot> element.

By default, browsers rarely treat the content of these elements any differently than other elements however designers often use CSS styles to change their appearance.

chapter-06/long-tables.html

HTML

```
<table>
  <thead>
    <tr>
      <th>Date</th>
      <th>Income</th>
      <th>Expenditure</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th>1st January</th>
      <td>250</td>
      <td>36</td>
    </tr>
    <tr>
      <th>2nd January</th>
      <td>285</td>
      <td>48</td>
    </tr>
    <!-- additional rows as above -->
    <tr>
      <th>31st January</th>
      <td>129</td>
      <td>64</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td></td>
      <td>7824</td>
      <td>1241</td>
    </tr>
  </tfoot>
</table>
```

RESULT

Date	Income	Expenditure
1st January	250	36
2nd January	285	48
3rd January	260	42
4th January	290	38
5th January	310	115
6th January	168	14
7th January	226	20
8th January	253	37
9th January	294	33
10th January	216	46
11th January	244	29
12th January	297	32
13th January	328	86
14th January	215	38
15th January	254	30
16th January	256	27
17th January	311	68
18th January	212	39
19th January	234	36
20th January	221	43
21st January	259	38
22nd January	246	31
23rd January	248	17
24th January	229	45
25th January	263	34
26th January	258	41
27th January	283	22
28th January	256	30
29th January	278	47
30th January	251	15
31st January	129	64
	7824	1241

Some of the HTML editors that come in content management systems offer tools to help draw tables. If the first row of your table only contains `<th>` elements then you may find that the editor inserts a `<thead>` element automatically.

Part of the reason for having separate `<thead>` and `<tfoot>` elements is so that, if you have a table that is taller than the screen (or, if printed, longer than one page) then the browser can keep the header and footer visible whilst the contents of the table scroll. This is intended to make it easier for users to see which column the data is in (however this functionality is not implemented by default in any current browser).

I have added some CSS styles to this example so that you can see the contents of the `<thead>` and `<tfoot>` being treated differently than the rest of the rows. You will learn how to apply these CSS styles to tables on pages 309-312 and 337-340.

OLD CODE: WIDTH & SPACING

There are some outdated attributes which you should not use on new websites. You may, however, come across some of them when looking at older code, so I will mention them here. All of these attributes have been replaced by the use of CSS.

The `width` attribute was used on the opening `<table>` tag to indicate how wide that table should be and on some opening `<th>` and `<td>` tags to specify the width of individual cells. The value of this attribute is the width of the table or cell in pixels.

The columns in a table need to form a straight line, so you often only see the `width` attribute on the first row (and all subsequent rows would use that setting).

The opening `<table>` tag could also use the `cellpadding` attribute to add space inside each cell of the table, and the `cellspacing` attribute to create space between each cell of the table. The values for these attributes were given in pixels.

I added CSS styles to this example so that you can see the width of the table cells more clearly. If you want to control the width or spacing of tables and cells you should use CSS as shown on pages 303, 337-340.

chapter-06/width-and-spacing.html

HTML

```
<table width="400" cellpadding="10" cellspacing="5">
  <tr>
    <th width="150"></th>
    <th>Withdrawn</th>
    <th>Credit</th>
    <th width="150">Balance</th>
  </tr>
  <tr>
    <th>January</th>
    <td>250.00</td>
    <td>660.50</td>
    <td>410.50</td>
  </tr>
  <tr>
    <th>February</th>
    <td>135.55</td>
    <td>895.20</td>
    <td>1170.15</td>
  </tr>
</table>
```

RESULT

	Withdrawn	Credit	Balance
January	250.00	660.50	410.50
February	135.55	895.20	1170.15

OLD CODE: BORDER & BACKGROUND

HTML

chapter-06/border-and-background.html

```
<table border="2" bgcolor="#efefef">
  <tr>
    <th width="150"></th>
    <th>Withdrawn</th>
    <th>Credit</th>
    <th width="150" bgcolor="#cccccc">Balance</th>
  </tr>
  <tr>
    <th>January</th>
    <td>250.00</td>
    <td>660.50</td>
    <td bgcolor="#cccccc">410.50</td>
  </tr>
  <tr>
    <th>February</th>
    <td>135.55</td>
    <td>895.20</td>
    <td bgcolor="#cccccc">1170.15</td>
  </tr>
</table>
```

RESULT

	Withdrawn	Credit	Balance
January	250.00	660.50	410.50
February	135.55	895.20	1170.15

The border attribute was used on both the `<table>` and `<td>` elements to indicate the width of the border in pixels.

The `bgcolor` attribute was used to indicate background colors of either the entire table or individual table cells. The value is usually a hex code (which we discuss on pages 249-252).

This example uses the HTML border and `bgcolor` attributes. No CSS attributes were utilized in this example.

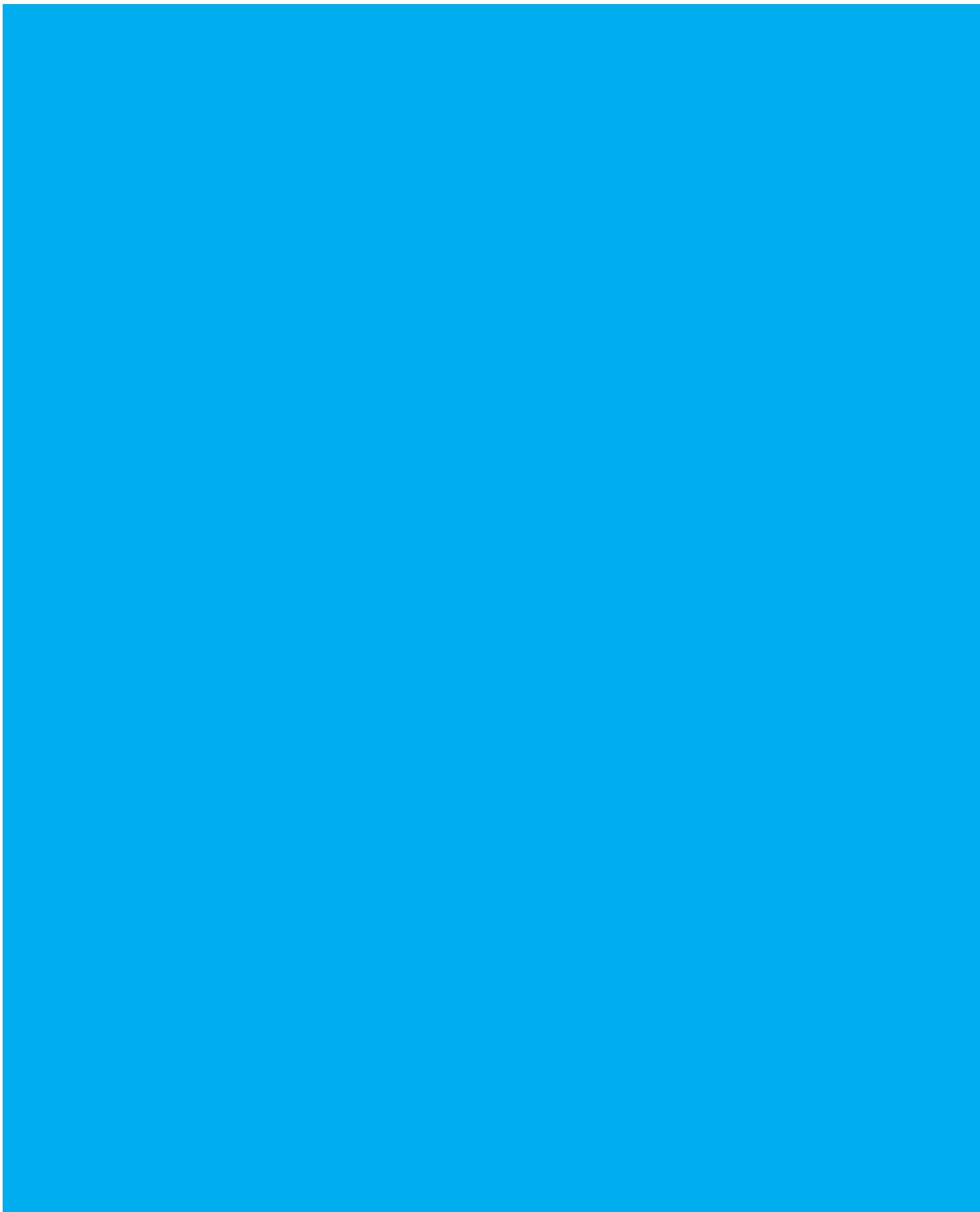
When building a new website you should use CSS to control the appearance of the table rather than these attributes. They are only covered here because you may come across them if you look at the code of older websites.

EXAMPLE

TABLES

```
<html>
  <head>
    <title>Tables</title>
  </head>
  <body>
    <table>
      <thead>
        <tr>
          <th></th>
          <th scope="col">Home starter hosting</th>
          <th scope="col">Premium business hosting</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <th scope="row">Disk space</th>
          <td>250mb</td>
          <td>1gb</td>
        </tr>
        <tr>
          <th scope="row">Bandwidth</th>
          <td>5gb per month</td>
          <td>50gb per month</td>
        </tr>
        <!-- more rows like the two above here -->
      </tbody>
      <tfoot>
        <tr>
          <td></td>
          <td colspan="2">Sign up now and save 10%!</td>
        </tr>
      </tfoot>
    </table>
  </body>
</html>
```





SUMMARY TABLES

- ▶ The `<table>` element is used to add tables to a web page.
- ▶ A table is drawn out row by row. Each row is created with the `<tr>` element.
- ▶ Inside each row there are a number of cells represented by the `<td>` element (or `<th>` if it is a header).
- ▶ You can make cells of a table span more than one row or column using the `rowspan` and `colspan` attributes.
- ▶ For long tables you can split the table into a `<thead>`, `<tbody>`, and `<tfoot>`.

