

# HTML Cookbook

A short Introduction to HTML for Beginners

© George Kneif (2006)

**This is a provisional and incomplete copy**

## Table of Contents

1 Introduction .....	3
1.1 How to use this manual .....	3
1.2 What is HTML? .....	3
1.3 What is CSS?.....	4
1.4 Character Sets, Unicode and HTML Entities.....	4
1.5 Screen Resolution.....	4
1.6 Colours .....	5
1.1.1 Colour Names.....	5
1.1.2 RGB Colour Values .....	5
1.1.3 Hexadecimal Code .....	5
1.7 Browser Incompatibility.....	5
1.8 Standards Organizations.....	6
2 Basic Tags .....	7
1.9 Basic formatting: Paragraphs and Line Breaks .....	8
1.10 Basic styles: Bold, Italics, Underline .....	8
1.11 Basic Attributes: 'align', 'style', 'class'.....	9
1.1.4 The 'align' attribute.....	9
1.1.5 The 'style' attribute .....	10
1.1.6 The 'class' attribute.....	10
1.12 The 'span' Tag .....	10
1.13 Comments.....	11
1.14 An Example of Chapter 2.....	11
3 Links.....	14
1.15 The 'a' tag .....	14
1.1.7 The 'href' attribute .....	14
1.1.8 The 'target' attribute.....	15
1.1.9 The 'class' attributes .....	15
1.16 Emails.....	15
1.17 An Example of Chapter 3.....	15
4 Pictures .....	18
1.18 The 'img' Tag.....	18
1.1.10 Specifying you Picture Source .....	18
1.1.11 Scaling your Picture .....	18
1.1.12 Picture Borders.....	18
1.1.13 Aligning your Picture and Text.....	19
1.1.14 The 'title' and 'alt' attributes .....	19
1.1.15 Padding between Picture and Text.....	19
1.19 Pictures as Links.....	19
1.20 Some Do-s and Don't-s .....	20
1.21 An Example of Chapter 4.....	20
5 Positioning Document Elements .....	23
1.22 The 'pre' tag .....	23

## 1 Introduction

This little manual is primarily intended for members of George's Circle who want to upload the contents of their webpages but do not know how to format their texts for the Internet. It is called *Cookbook*, because – like cookbooks' recipes – it gives practical advice without going into fine details or into the theories of cooking.

Nevertheless, even cookbooks assume that the reader and the potential cook knows quite a lot about cooking: that you have to make a fire (or turn on the gas or electricity) to cook; that you know how to measure weight, volume or temperature; that you know what the utensils are and how to use them; etc. So it is with the Internet. There is some background knowledge that you should know before we get into the itty-gritty of Internet 'cooking'.

The rest of this introduction explain some of the most important background ideas. You may safely ignore it if you know something about HTML.

### 1.1 How to use this manual

Normal reference books take a particular topic and discuss it exhaustively. We shall do it quite differently.

Each subsection contains a small *recipe*: a small explanation and then instructions on how to do certain things you will surely want to do. After you have read the section, follow the instructions by doing it. You learn best by trying and experimenting

This text is in 'Times New Roman'. Texts that you type will be in blue 'Courier New'. Text that is going to be displayed is in brown 'Arial'.

As preparation create a page with a blank template on your website. Then take a text file with enough text to fill about  $\frac{1}{2}$  –  $\frac{3}{4}$  of your page. The text should not have any other formatting than what you could achieve with a typewriter, i.e. spaces, newlines and tabs. You can use any text editor program to create text files, e.g. *Notepad* on *Windows* platforms or *Emacs* or *vi* on *Linux*. You can also write your text in MS Word but you must store it as a text file. You will be modifying that text file as we go along, upload it and see the effects of your changes.

### 1.2 What is HTML?

Exercise 1.1: Upload your text file then load your page. ■

You will have noticed that your browser has displayed the text *en bloc*, i.e. it has reduced all sequences of **white space**, i.e. space, tab, carriage return and line feed, to a single space. This is because it had expected an **HTML** (Hypertext Markup Language) document from the server and displayed it accordingly.

HTML was the first attempt to transmit text using only ASCII (American Standard Code for Information Interchange) characters, i.e. characters, whose code range is between 32 and 126, the

characters that were available on the original US keyboard. In an HTML document special sequence of the allowed characters, so called **tags**, tell the recipient how to interpret the text.

Of course, HTML has evolved and it can do much more than just telling the browsers how to display text. Nevertheless it has reached its old age and a new version of HTML, **XHTML** (Extensible Hypertext Markup Language), has been defined. As HTML is simpler and it will be supported by the browsers for a long time, we shall concentrate our attention on HTML.

### 1.3 What is CSS?

Although HTML has defined some tags for styles, they were found to be inadequate. To correct these deficiencies CSS (Cascading Style Sheets) came to the rescue. CSS is a language in its own right, in which you can define complex styles and their relationships to HTML components, thus HTML can connect to such style definitions.

We shall refer to some CSS code and make use of some elementary CSS definitions provided by the website.

### 1.4 Character Sets, Unicode and HTML Entities

ASCII characters are encoded as bytes, however, only the lower seven bits were defined. Later it was extended to the full range by including all the characters used by west European languages, e.g. ä, ß, ç and ě. Thus the character set **Latin-1** was born.

Nowadays, when the Internet also transmits in all sorts of languages and scripts e.g. Arabic, Hebrew, Chinese, an 8-bit character set is not sufficient. In order to accommodate all the different characters the 16-bit Unicode character set has been created.

The Latin-2 character set is also an 8-bit code, used by some central and east European languages. For example the Latin-2 Hungarian character ő has a code of 245, which corresponds to the Latin-1 character ð. It is the responsibility of the browser to distinguish between the two. The Unicode characters are, however, different: ð = 245 but ő = 337.

Characters that you cannot type or those that are reserved by HTML can be specified by a numeric code or an HTML entity, i.e. a specified name assigned to that character. They all start with an ampersand (&) and terminated with a semicolon (;). For example the small ae ligature æ you can specify either as `&#230;` or `&aelig;` The character sequence `<b>` would be interpreted by HTML as 'bold characters follow'. If you wanted the three characters to appear exactly as you have typed them, you must write `&lt;b>`; You will find the numeric codes and HTML entities of the Latin-1 character set in the Appendix.

### 1.5 Screen Resolution

The information on your screen is displayed as tiny dots called **pixels**. The resolution of your screen is the number of pixels on your screen. The minimum resolution is 640 × 480 pixels. Most screens nowadays have a resolution at least 1024 × 768 pixels and if you have a good monitor with a large

screen you can have 1600 × 1200. Practically all screens are of an aspect ratio (i.e. horizontal size divided by vertical size) of 4:3.

We can use relative or absolute measurements when we want to define the size and position of an object. For displays you should always use pixels. This is a relative measurement because the pixel sizes – or rather the distance between two pixels – depends on the size and resolution of your screen. However, it ensures that your objects always look the same on all screens. For printing one uses absolute measurements: centimetres (cm) for position and points (pt) for character size.

## 1.6 Colours

There are three ways to specify colours.

### 1.1.1 Colour Names

A set of standard colour names have been specified for use in HTML, e.g. red, blue, moccasin, darkseagreen. You will find the standard and extended colour names in the Appendix.

### 1.1.2 RGB Colour Values

R stands for red, G for green and B for blue. Each of the three colours can take up any value between 0 and 255: 0 being no colour, 255 full colour. You can create all the colours with the combination of the three colours.

You do not use the RGB values directly in your HTML code, instead you combine them into a single 24-bit hexadecimal number.

### 1.1.3 Hexadecimal Code

In you HTML code you can specify colours by the code #rrggbb, where rr is the hexadecimal value for red, i.e. 0 – ff, similarly gg for green and bb for blue. Thus

black	=	#000000
blue	=	#0000ff
lime	=	#00ff00
red	=	#ff0000
aqua	=	#00ffff
fuchsia	=	#ff00ff
yellow	=	#ffff00
white	=	#ffffff
grey	=	#808080
maroon	=	#800080

## 1.7 Browser Incompatibility

Unfortunately the HTML specification is not precise. The browsers have a great deal of leeway on how to display, say, paragraph spacing. Also, some browsers have introduced their own facilities. *George's Circle* has been tested with five browsers and the result is a compromise. If you ask the Webmaster, he will check your pages with all available browsers to ensure that they display correctly.

## 1.8 Standards Organizations

The World Wide Web Consortium has the responsibility of standardizing any technology related to the Web; they manage the HTML, XML, and CSS standards, as well as related standards for document addressing on the Web. You should have a quick look at their website <http://www.w3.org> to see what is available. There you will find everything you always wanted to know but were afraid to ask.

The Internet Engineering Task Force (IETF) is responsible for defining and managing every aspect of Internet technology. Their website is at <http://www.ietf.org>.

There are also several newsgroups devoted to the Web, each part of the *comp.infosystems.www* hierarchy.

## 2 Basic Tags

The best way to explain tags is by examples.

### Example 2.1: Start tag only

If you write `<br>` in your text, the browser will insert a line break, e.g. if you write

```
This text<br>should go over two lines!
```

it will appear as

This text  
should go over two lines! ■

### Example 2.2: Start tag, text, end tag

If you write `<b>`, the following text will appear in **bold**. But how do you stop it being bold? By inserting the end tag `</b>`. For example, if you write

```
Words six to eight should <b>appear in bold</b>!
```

It will be displayed as

Words six to eight should **appear in bold!** ■

XHTML insists that all start tags should have a corresponding end tag. Thus you should write example 2.1 as

```
This text<br/>should go over two lines!
```

and not as you might expect as

```
This text<br></br>should go over two lines!
```

### Example 2.3: Start tag with attributes

A start tag may take attributes. An attribute consists of a key, followed by the equal sign, followed by the key value in double quotes. Some attributes consist of a key only. In HTML you may get away with writing the value without the double quotes, however, XHTML always demands them, therefore it is best to get used to always using them. For example, if you write your paragraph tag as

```
<p align="right">This paragraph is aligned to the right of the  
page.</p>
```

it will appear as



This paragraph is aligned to the right of the page.

The case of characters within a tag is irrelevant. Thus you could write `<br/>`, `<BR/>`, `<Br/>` or even `<bR/>`. Nevertheless, it is best to use the same scheme all through your text.


The text between a starting tag and the corresponding end tag may contain other tags, depending on the tags you use. For example you may write

```
<p align="right">This paragraph is aligned to the  
<b><i>right</i></b> of the page.</p>
```

which would produce

This paragraph is aligned to the *right* of the page.

displaying the word 'right' in bold italics. The tags must be properly nested, of course.

Exercise 2.1: Append the above examples to your text, upload it and see what happens. Then make some mistakes in the tags and see how your browser reacts. 

You may be surprised to see that your browser has ignored all the mistakes. It gave no error indication. However, this is no problem because if you do not see what you intended, you know that something is amiss and you can re-examine your code. It may be a problem, however, when you use a tag or attribute that is acceptable to one browser but not to another. Alas, that is HTML for you!

## 1.9 Basic formatting: Paragraphs and Line Breaks

To define a paragraph, use the paragraph tag `<p>`, e.g.

```
<p>This is my paragraph.</p>
```

The amount of space between two paragraphs depends on the browser.

Use the `<br/>` tag to start a new line. Sometimes it is more convenient to insert extra space between two lines by typing `<br/><br/>` rather than creating a new paragraph. Do not assume that the generated space is the same as that between two paragraphs!

## 1.10 Basic styles: Bold, Italics, Underline

To display a text in **bold**, you should put it within the tags `<b> ... </b>`. For example

This `<b>word</b>` is in bold.

would appear as

This **word** is in bold.

To display a text in *italics*, you should put it within the tags `<i> ... </i>`. For example

This `<i>word</i>` is in italics.

would appear

This *word* is in italics.

To display a text underlined, you should put it within the tags `<u> ... </u>`. For example

This `<u>word</u>` is underlined.

would appear

This word is underlined.

You can nest the style tags in any order. For example, if you type

This `<u><i><b>word</b></i></u>` is underlined bold italics.

would appear as

This ***word*** is underlined bold italics.

## 1.11 Basic Attributes: 'align', 'style', 'class'

You can use the three attributes with the `<p>` tag. Their effect extends to the entire paragraph.

Later you will see that these tags can be used for many other tags too.

### 1.1.4 The 'align' attribute

Browsers automatically align a paragraph to the left margin of the page. If you wish to change the alignment, you must use the 'align' attribute. It can take four different values:

```
align="left",  
align="right",
```

```
align="center" and  
align="justify".
```

Note the American spelling of centre! The 'justify' alignment aligns a full line to both the left and right margins of the page, adjusting the word separation if necessary.

### 1.1.5 The 'style' attribute

The 'style' attribute is the connecting tag to CSS. Its value is some CSS code. We shall learn later the most important CSS codes. Here just a small illustration. The paragraph tag

```
<p style="text-align: justify; padding-left: 20px;">
```

would add an extra 20 pixel wide space before the justified paragraph text.

### 1.1.6 The 'class' attribute

As CSS style specifications may be long and complex, you can combine them into a *class*. Your templates offer a set of classes that you may use. These classes specify font type (sans-serif), weight (normal, bold) and orientation (normal, italics). The 'class' value names are self-explanatory:

- normal10px – normal24px: at every pixel in-between.
- italic10px – italic24px: at every pixel in-between.
- bold10px – bold24px: at every pixel in-between.
- bold26px – bold40px: at every even pixel in-between.
- bold44px – bold64px: at every fourth pixel in-between.
- bold italic10px – bolditalic24px: at every pixel in-between.
- bold italic26px – bolditalic40px: at every even pixel in-between.
- bold italic44px – bolditalic64px: at every fourth pixel in-between.

#### Example 2.4: Using classes

Suppose you want to make a bold underlined heading with big letters. One possibility would be:

```
<p align="center" class="bold32px"><u>Heading</u></p>
```



Sans-serif letters look better on display and serif ones on print.

Exercise 2.2: Style one of your paragraphs as example 2.4. Upload and look at the result. Experiment by changing the class value, alignment and style tags. ■

## 1.12 The 'span' Tag

Suppose you have set your paragraph's class to *normal13px* but then you would like to set part of your text to a different class. What can you do? The 'span' tag with your new class attribute changing the style of your text in the tag block without changing anything outside it. The following example makes this clear:

```
<p class="normal13px">One of the most useful tags is the <span class="bold16px">SPAN</span> tag because it can change the style of just a portion of your text.</p>
```

This would appear as

One of the most useful tags is the **SPAN** tag because it can change the style of just a portion of your text.

### 1.13 Comments

You may put comments with your text. They are not displayed; they are purely there to document your code. You start your comment with `<!--` and terminate it with `-->`. For example

```
<!-- This is a comment -->
```

### 1.14 An Example of Chapter 2

The following is the content from one of Ann's pages:

```
<!-- Display Content -->
<!-- Heading -->
<br/>
<p align="center" class="bold24px">Talks</p>

<!-- Subheading -->
<p class="bold18px">The Italian Influence on the Gardens of
Kent</p>

<!-- Text -->
<p style="text-align: justify; padding-left: 20px; padding-right:
20px;">
```

```
The Italian influence on the gardens in Kent can be found from the
16th century until the present day. Gardens can change greatly
over time and have often been used as a self-expression as new
owners have been eager to imprint their own personality on their
garden.
```

```
Find out how they have done this in Kent.
```

```
</p>
```

The displayed page is shown on the next page. Note that the template presets certain attributes: alignment (left), text type (Sans-Serif), text size (13px), text weight (normal), text colour (blue).

You need to specify only those attributes that you wish to change from their default values. Note, for example, that the subheading paragraph does not specify alignment and the text block paragraph does not specify class because they are happy using the default values. You will find the default values listed in the Help pages when you define your template for your page.



# Ann Kneif

<a href="#">Home Page</a>	<a href="#">Academia</a>	<a href="#">Talks</a>	<a href="#">Snippets</a>	<a href="#">Album</a>	<a href="#">Main</a>	
<a href="#">Leeds Castle</a>	<a href="#">Kent Gardens</a>	<a href="#">Glassmaking</a>	<a href="#">Bevin Boys</a>	<a href="#">WWII Women</a>	<a href="#">Previous</a>	<a href="#">Next</a>

## Talks

### The Italian Influence on the Gardens of Kent

The Italian influence on the gardens in Kent can be found from the 16th century until the present day. Gardens can change greatly over time and have often been used as a self-expression as new owners have been eager to imprint their own personality on their garden. Find out how they have done this in Kent.

<a href="#">Home Page</a>	<a href="#">Academia</a>	<a href="#">Talks</a>	<a href="#">Snippets</a>	<a href="#">Album</a>	<a href="#">Main</a>	
<a href="#">Leeds Castle</a>	<a href="#">Kent Gardens</a>	<a href="#">Glassmaking</a>	<a href="#">Bevin Boys</a>	<a href="#">WWII Women</a>	<a href="#">Previous</a>	<a href="#">Next</a>

Created: 18/01/2006                      Author: Ann Kneif                      Last upload: 18/01/2006

## 3 Links

The beauty of HTML is the ease with which one can navigate through a document spreading over several pages or load a disparate one in a new window for, say, illustration. One click with the mouse and the page is already there.

The main tag for creating links is the 'a' tag.

### 1.15 The 'a' tag

The general form of a simple 'a' tag is

```
<a class="..." href="..." target="...">Text</a>
```

The 'href' attribute is mandatory, the others are optional. The above code will output 'Text' only, specially marked by the browser, usually in blue and underlined. Text, of course, may be other things than just text, for example pictures, which we shall describe later.

#### 1.1.7 The 'href' attribute

The value of the 'href' attribute is the transmission protocol and URL of the page you wish to access. The URL may be absolute or relative. The transmission protocol with separator for us is always 'http://' (Hyper Text Transmission Protocol).

Using absolute address of the home page of George's Circle would be

```
<a href="http://www.kneif.info/main/Home-en.php"> or  
<a href="http://www.kneif.info">
```

In the second case the server recognizes that the address is a directory and searches for a file called index.html or index.php. Either of these two files would then load the home page.

Relative addresses can be relative to the server's or to your own root directory. The transmission protocol is omitted in relative addresses.

An address relative to the server always starts with a '/'. For example the main home page can be addressed as

```
<a href="/main/Home-en.php">
```

The same page can be accessed relative to your own root directory as

```
<a href="../main/Home-en.php">
```

where '..' specifies the directory above yours, i.e. the website's root directory.

To access a file in your own root directory you just specify the file name. If you want to load a file from a directory below yours, say a picture file called 'MyPic.jpg' in your 'media' directory, you would write

```
<a href="media/MyPic.jpg">
```

Note that if you hover over a link with the mouse, the target URL will be displayed in the windows status bar.

### 1.1.8 The 'target' attribute

Occasionally you may not want the new page to overwrite the current one but want to appear in a new window. For this you can use the 'target' attribute. E.g.

```
<a href="http://www.open.ac.uk/students" target="_blank">
```

would load the Open University students' home page in a new window.

### 1.1.9 The 'class' attributes

If you do not specify a class attribute in your 'a' tag the browser displays your link text in its standard 'livery', i.e. blue underlined. If you want to distinguish between an internal and an external link, use the class attribute with value 'external'. This will display your link in red. E.g.

```
<a class="external" href="http://www.open.ac.uk/students" target="_blank">The students' Home Page</a>
```

will be displayed as

[The students' Home Page](http://www.open.ac.uk/students)

## 1.16 Emails

If you wish that your readers send you an email, you can set up a special *mailto* link. Clicking this link the browser opens an email form for the client with your mailbox address already filled in. The *mailto* link is a special value of the 'href' attribute of the 'a' tag. For example the webmaster would write:

```
<a href="mailto:webmaster@kneif.info">Webmaster</a>
```

where [webmaster@kneif.info](mailto:webmaster@kneif.info) is the Webmaster's mailbox address and the link text 'Webmaster' will be displayed in the standard link colour.

## 1.17 An Example of Chapter 3

The following code is Ann's home page, which contains all the various link forms:

```
<!-- Display Content -->
<!-- Heading -->
<br>
<p align="center" class="bold24px">Home Page</p>

<!-- Subheading -->
<p class="bold18px">A brief Introduction</p>

<!-- Text -->
<p style="text-align: justify; padding-left: 20px; padding-right:
20px;">
I am George's soul mate.<br><br>

I have too many interests so I do not have enough time to pursue
any of them properly
- gardening, cooking, photography, reading are just a few.<br><br>

I work as a guide at <a class="external" href="http://www.leeds-
castle.com/" target=_blank>Leeds Castle</a> in Kent,
'the loveliest castle in the world'.<br><br>

I completed my PhD in May 2005.
You can see my academic career under <a href="academia1-
en.php">Academia</a>.<br><br>

I also give talks to local groups and societies.
See <a href="talks1-en.php">Talks</a> for further detail.<br><br>
</p>

<p align="left" class="bold18px">Contact</p>
<p style="text-align: justify; padding-left: 20px; padding-right:
20px;">
I may be contacted via e-mail at <a
href="mailto:ann@kneif.info">ann@kneif.info</a>.
</p>
```

Note that Ann uses `<br>` instead of `<br/>` and the target attribute is without inverted commas. This is fine in HTML but would be an error in XHTML.

Ann's home page is displayed on the following page.

English ▾

# Ann Kneif

<a href="#">Home Page</a>	<a href="#">Academia</a>	<a href="#">Talks</a>	<a href="#">Snippets</a>	<a href="#">Album</a>	<a href="#">Main</a>
---------------------------	--------------------------	-----------------------	--------------------------	-----------------------	----------------------

## Home Page

### A brief Introduction

I am George's soul mate.

I have too many interests so I do not have enough time to pursue any of them properly – gardening, cooking, photography, reading are just a few.

I work as a guide at [Leeds Castle](#) in Kent, 'the loveliest castle in the world'.

I completed my PhD in May 2005. You can see my academic career under [Academia](#).

I also give talks to local groups and societies. See [Talks](#) for further detail.

### Contact

I may be contacted via e-mail at [ann@kneif.info](mailto:ann@kneif.info).

<a href="#">Home Page</a>	<a href="#">Academia</a>	<a href="#">Talks</a>	<a href="#">Snippets</a>	<a href="#">Album</a>	<a href="#">Main</a>
---------------------------	--------------------------	-----------------------	--------------------------	-----------------------	----------------------

Created: 17/01/2006

Author: Ann Kneif

Last upload: 17/01/2006

## 4 Pictures

Pictures are integral and a very important part of web pages. In this chapter we are looking at how you can insert pictures in your pages.

Consult your image processing program to find out how pictures are stored in files and the various file types. Currently templates accept GIF, JPEG, JPG, PNG and TIFF picture files only. You, of course, can load any file in your HTML code.

Use your *media* subdirectory for storing and retrieving your pictures.

### 1.18 The 'img' Tag

Pictures are embedded in your page via the 'img' tag. It has many attributes, of which only the 'src' attribute is mandatory. It is a stand-alone tag, so you should always terminate it with '/>'.

#### 1.1.10 Specifying you Picture Source

You use the 'src' attribute to specify the address of your picture file. It is similar to the 'href' attribute of the 'a' tag. Its value is the URL of a picture file. E.g. to insert your picture *MyPic* stored in file *MyPic.jpg* into your paragraph you should write

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

The above code will load your picture aligned to the left of the paragraph and let the *Text* flow around it.

#### 1.1.11 Scaling your Picture

You may use the browser to scale your picture by specifying the required size with the 'width' and 'height' attributes. The attributes values are in pixels. E.g. if you have a picture of size 1600 × 1200 pixel, you may reduce its dimensions tenfold by the following code:

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

You should be careful that you keep the aspect ratio, otherwise the browser will distort your picture by stretching one of the dimensions.

#### 1.1.12 Picture Borders

You may place a border around your pictures with the 'border' attribute. The value of the attributes is the thickness of the border in pixels. E.g.

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

### 1.1.13 Aligning your Picture and Text

You can use the `align` attribute with the `'img'` tag. It can take on five values: *left*, *right*, *top*, *middle* and *bottom*. The left and right values flow any subsequent text around the picture, which is moved to the corresponding margin; the remaining three align the image vertically with respect to the surrounding text.

If you want part of your text to flow with the picture but the rest to be placed below the picture you can use the `'br'` tag with the *clear* attribute.

The following code places pictures against both margins, lets part of the text run down the middle and the rest below the pictures:

```
<p>


Text1<br clear/>Text2
</p>
```

### 1.1.14 The 'title' and 'alt' attributes

Both attribute values should be the title of your picture. The `'title'` value is used to display a floating tab text when you hover above the picture with the mouse, the `alt` value is displayed in a picture place holder if the browser cannot load the picture for some reason or while the picture is being loaded.

### 1.1.15 Padding between Picture and Text

Browsers do not leave much space between the picture and text. However you may control this with the `'hspace'` and `'vspace'` attributes. Their values are in pixels. The following code would leave a 10 pixel gap between the picture and the text on the right and a 15 pixel gap between the picture and the overflowing text below it.

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

## 1.19 Pictures as Links

You can use pictures as links just by replacing the text in an `'a'` tag block with an `'img'` tag. Normally you use it either for loading a new page or for load a larger picture.

The following code loads a small picture with the page then when you click on the picture a larger version of it is displayed in a separate window:

```
<a href="media/leeds-with-swans.jpg" target="_blank">
```

```

</a>
```

If you use a picture for a link, browsers tend to put a border around the picture and colour it with the link colours in order to indicate that the picture is a link. You may suppress the border by setting it to 0 as above. In that case you should indicate somehow that the picture is a link.

In the next example the picture is used to load Ann's home page in the same window:

```
<a href="/Ann/index.php">

</a>
```

## 1.20 Some Do-s and Don't-s

- Do not forget that the content is more important than the pictures, even if we accept the adage that a picture says more than a hundred words. Even in an album, where pictures are the content, without some explanation pictures mean very little.
- Do not use 'bmp' files: they are big, use up lots of resources and take long to download. If you have any, convert them to 'jpg' before uploading them.
- The best strategy is to create two versions of your pictures: a small one to be downloaded with your page and a bigger one to be displayed in a separate window when the small picture is clicked.
- Conform to the standard resolution of 72 pixels per inch and do not make any picture larger than about  $\frac{3}{4}$  of the average screen.
- Your page should not force the viewer to scroll horizontally.

## 1.21 An Example of Chapter 4

The following code is Ann's page about Leeds Castle:

```
<!-- Display Content -->
<!-- Heading -->
<br>
<p align="center" class="bold24px">Talks</p>

<!-- Subheading -->
<p align="left" class="bold18px">Leeds Castle</p>

<!-- Text -->
```

```
<p style="text-align:justify; padding-left: 20px; padding-right:
20px;">
The castle has evolved over time with each owner leaving their own
mark.
The oldest remaining part of the present castle is the 12th
century Norman cellar.
Royalty owned the castle from the time of Edward I (1278)
until Edward VI (1552) after which it passed into private hands.
The last owner was Lady Baillie, who died in 1974. It is now owned
by a private trust.<br><br>
<a href="media/leeds-with-swans.jpg" target=_blank>

</a>
<b>Leeds Castle</b><br><br>
<span class="normal11px">
Leeds Castle, near Maidstone, Kent - photo by Ann Kneif.
<br><br><br><br><br><br><br>
</span>
<span class="italic11px" style="color: #0066cc">
(Click picture to see a larger version!)
</span>
</p>
```

Ann's page about Leeds Castle is displayed on the following page.



**Ann Kneif**

English ▾

<a href="#">Home Page</a>	<a href="#">Academia</a>	<a href="#">Talks</a>	<a href="#">Snippets</a>	<a href="#">Album</a>	<a href="#">Main</a>
<a href="#">Leeds Castle</a>	<a href="#">Kent Gardens</a>	<a href="#">Glassmaking</a>	<a href="#">Bevin Boys</a>	<a href="#">WWII Women</a>	<a href="#">Next</a>

## Talks

### Leeds Castle

The castle has evolved over time with each owner leaving their own mark. The oldest remaining part of the present castle is the 12th century Norman cellar. Royalty owned the castle from the time of Edward I (1278) until Edward VI (1552) after which it passed into private hands. The last owner was Lady Baillie, who died in 1974. It is now owned by a private trust.



#### Leeds Castle

Leeds Castle, near Maidstone, Kent - photo by Ann Kneif.

*(Click picture to see a larger version!)*

<a href="#">Home Page</a>	<a href="#">Academia</a>	<a href="#">Talks</a>	<a href="#">Snippets</a>	<a href="#">Album</a>	<a href="#">Main</a>
<a href="#">Leeds Castle</a>	<a href="#">Kent Gardens</a>	<a href="#">Glassmaking</a>	<a href="#">Bevin Boys</a>	<a href="#">WWII Women</a>	<a href="#">Next</a>

Created: 17/01/2006

Author: Ann Kneif

Last upload: 17/01/2006

## 5 Positioning Document Elements

Without intervention the browser flows document elements together positioned sequentially through the display. Occasionally however you may want to take some elements out of the flow and position it arbitrarily on your page. We cover three ways you may do this: the 'pre' tag, tables and the CSS position property in combination with the HTML 'div' tag..

### 1.22 The 'pre' tag