# Chapter 1: Structure

-HTML describes the structure of pages.

-HTML **elements**: characters that live inside angled brackets. Elements are usually made up of 2 **tags**: an opening tag and a closing tag.

-HTML uses elements to describe the structure of pages.

-<html>: anything between it is HTML code.

-<body>: anything between it should be shown inside the main browser window.

-<h1>: main heading

-<p>: a paragraph of text

-<h2>: sub-heading

-**Attributes** tell us more about elements. Attributes provide additional information about the contents of an element. They appear on the opening tag of element and are made up of 2 parts: a name + value, separated by an equals sign.

-Body, Head, Title

+<body>: Everything inside this element is shown inside the main browser window

+<head>: contain information about the page. You will usually find <title> element inside

+<title>: shown in top of browser or on the tab.

-Looking at how other sites are built: View->Source

-Summary:  
+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 com 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 see, what they do, and where they can go.

# Chapter 2: Text

-When creating a web page, you add tags (markup) to the content of page.

+**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:

+HTML has 6 levels of headings: <h1>-<h6>

-Paragraph: Create a paragraph: <p>

-Bold and Italic: <b> <i>

-Superscript and subscript: <sup> <sub>

-White space collapsing: When the browser comes across two or more spaces next to each other, it only displays one space.

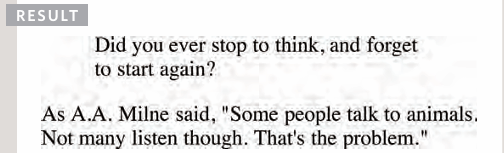
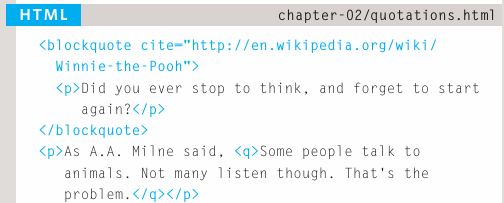
-Line breaks and horizontal rules: <br /> <hr />

-Visual editors and their code views: Content management systems and HTML editors like Dreamweaver usually have two views of the page: a visual editor and a code view.

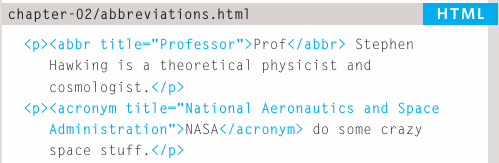
-**Semantic markup**: text elements that are not intended to affect the structure of web pags, but they do add extra information to pages.

+Strong and emphasis: <strong>: bold <em>:italic

+Quotations: <blockquote> longer quotes that take up an entire paragraph. <q> shorter quotes that sit within a paragraph.

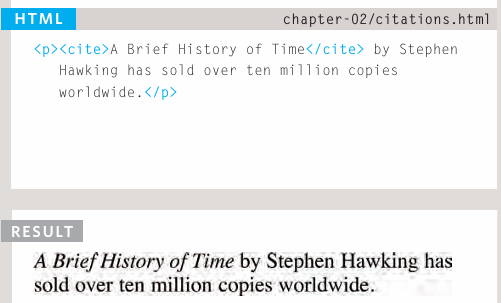


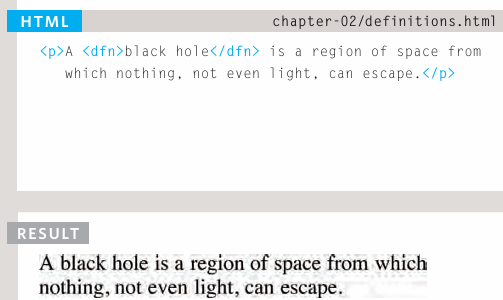
+Abbreviations & Acronyms: <abbr>





+Citations & Definition: <cite>: italics, <dfn>: italic or not change.

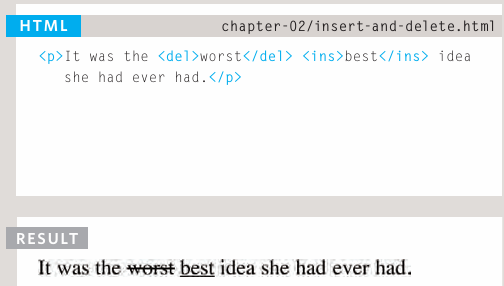


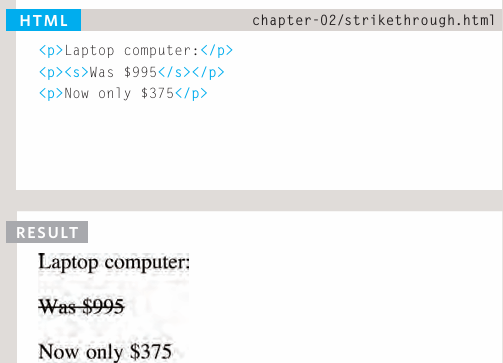


+Author details: <address>: italic



+Changes to content: <ins> <del> <s>





-Summary:

+HTML elements are used to describe the structure of page (headings, subheadings, paragraphs)

+They also provide semantic information (where emphasis should be placed, the definition of any acronyms used, when given text is a quotation)

# Chapter 3: Lists

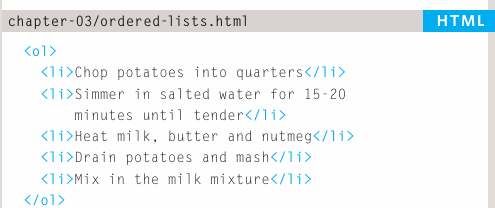
-There are lots of occasions when we need to use lists. HTML provides us with 3 different types:

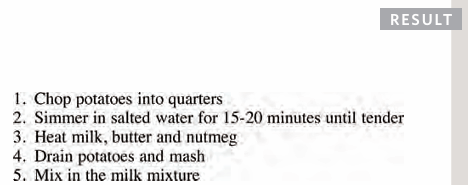
+Ordered lists: each item in the list is numbered.

+Unordered lists: begin with a bullet point.

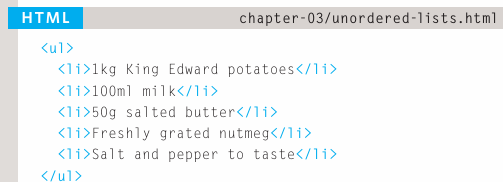
+Definition lists: set of term along with the definitions for each of those terms.

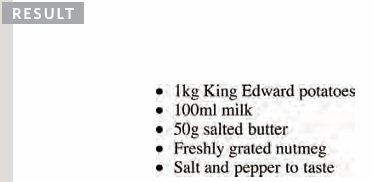
-Ordered lists: <ol> <li>



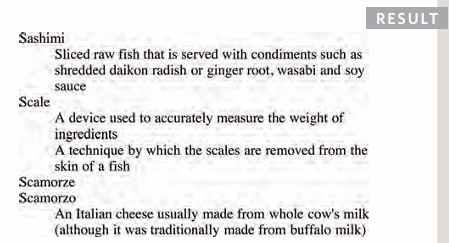
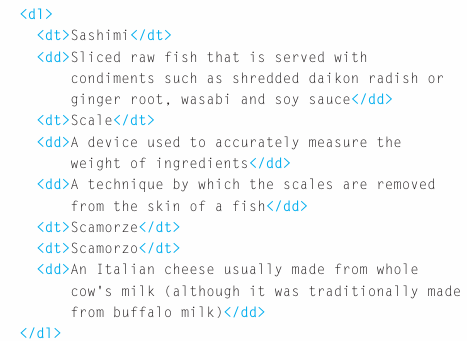


-Unordered lists: <ul> <li>



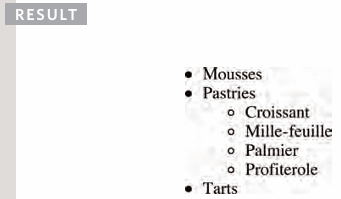


-Definition lists: <dl> <dt> <dd>



-Nested lists: You can put a 2nd list inside <li> element





-Summary:

+There are 3 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.

# Chapter 4: Links

-Links are the defining feature of web because they allow you to move from one page to another.

-Types of links:

+Links from one websites 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.

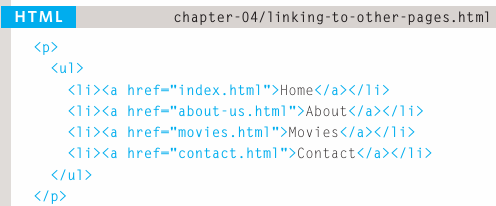
+Links that start up your email program and address a new email to someone.

-Linking to other sites: When you link to a different website, the value of href attribute will be the full web address for the site, which is **absolute URL**

+URL (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 page. An absolute URL starts with domain name + path to page.



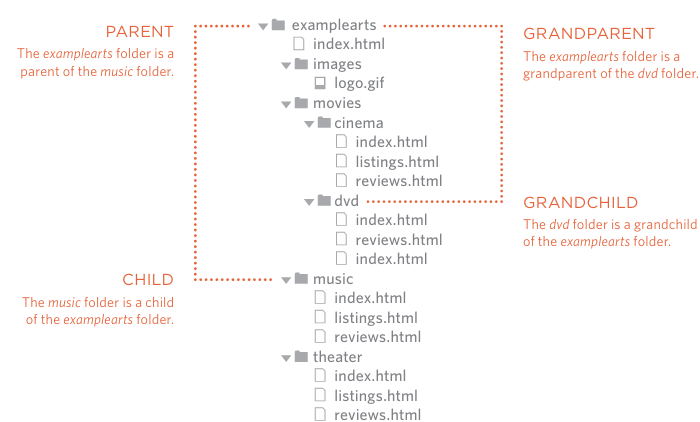
-Linking to other pages on the same site: Relative URL. If all the pages of site are in the same folder, the value of href attribute is just the name of file. If you have different page of a site in different folders, you can use more complex syntax to indicate where the page is in relation to the current page.



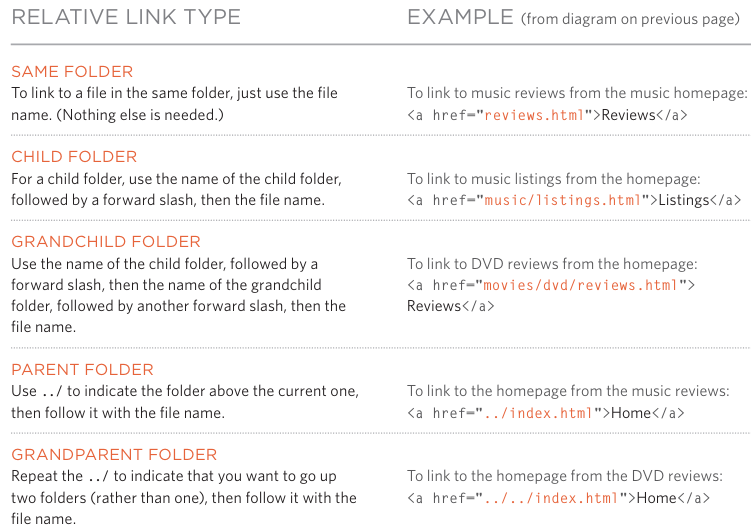
+Relative URLs: link to other pages within the same site. These are a shorthand version of absolute URLs because you don’t need to specify the domain name.

-Directory structure:

+Structure: The top-level folder is **root folder**.



-Relative Urls:



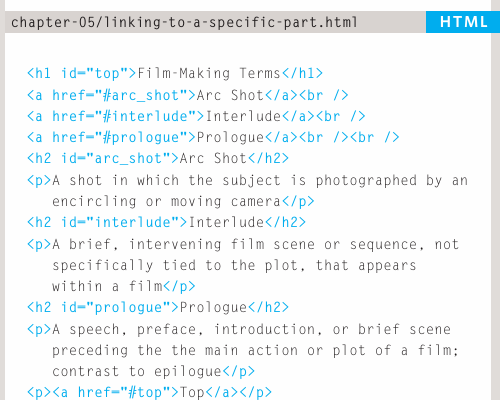
-Email links: mailto:



-Opening links in a new window: target



-Linking to a specific part of the same page:



-Linking to a specific part of another page: Example to link to the bottom of homepage of website: 

-Summary:

+Links are created using <a> element

+The <a> element uses 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 “emailto” field.

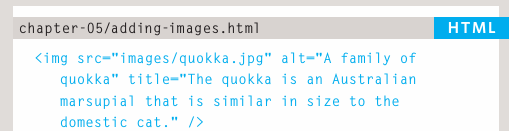
+You can use id attribute to target elements within a page that can be linked to.

# Chapter 5: Images

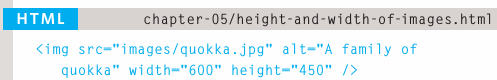
-Choosing images for your site

-Storing images on your site.

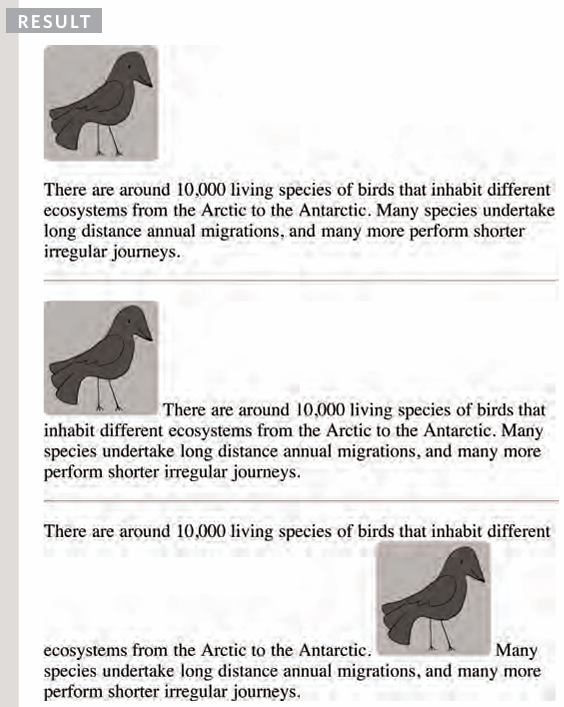
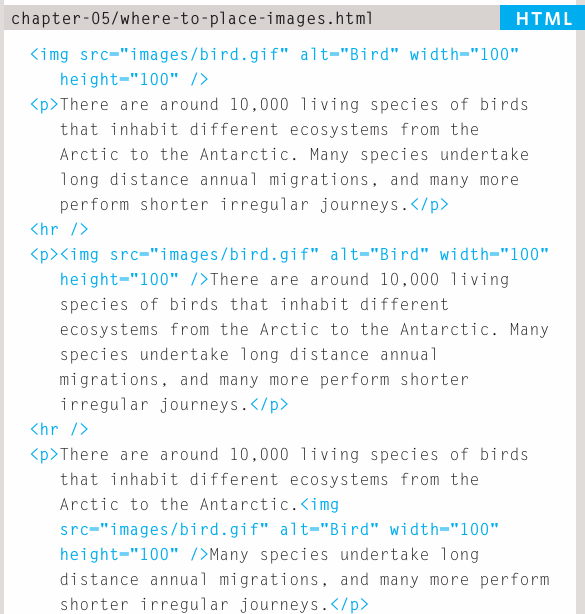
-Adding images: <img>



-Height and Width of images:



-Where to place images in code: before a paragraph, inside the start of a paragraph, in the middle of paragraph:

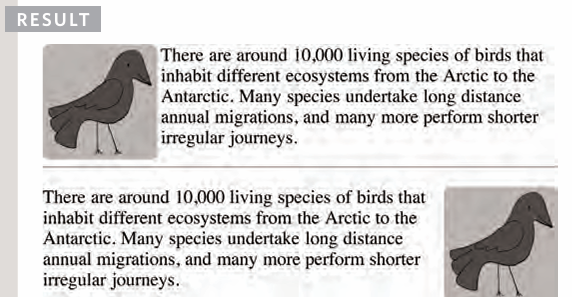
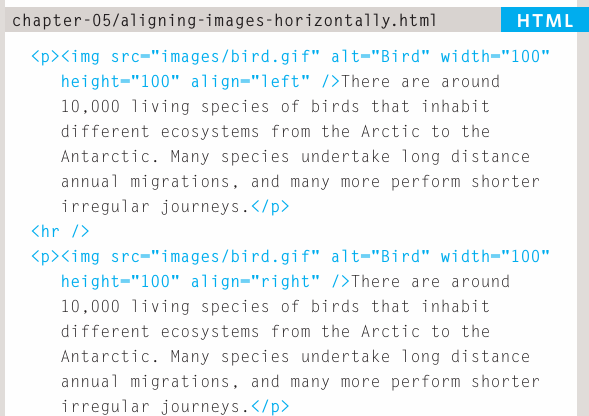


+Block elements always appear on a new line: <h1><p>

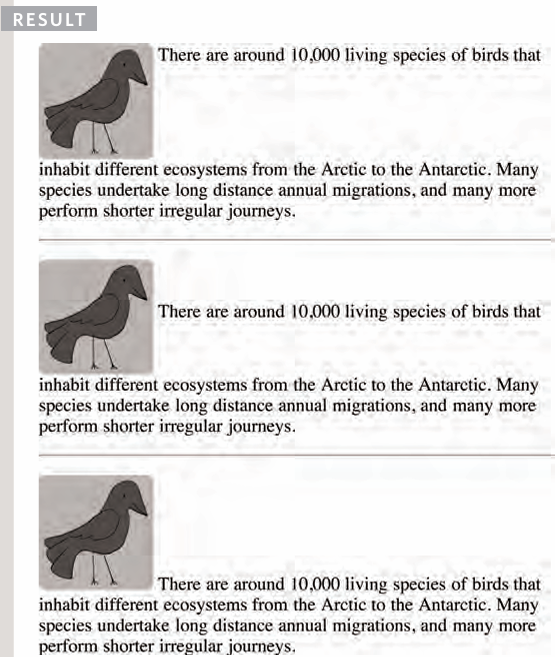
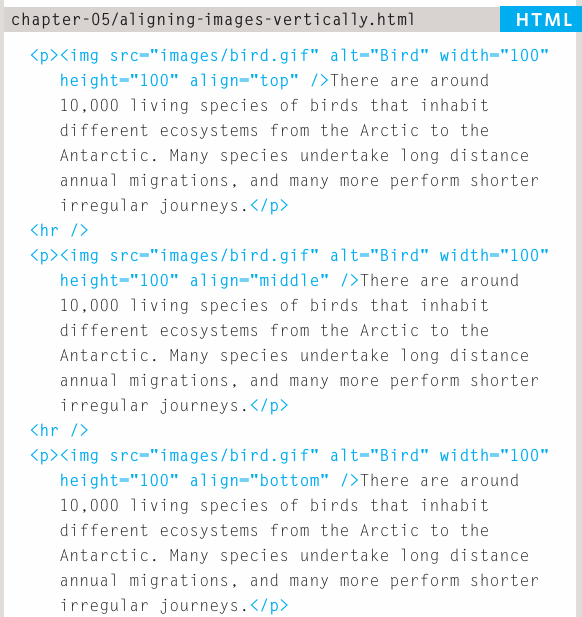
+Inline elements sit within a block level element and don’t start on a new line: <b> <em> <img>

-Old code:

+Align images horizontally: align + left/right



+Align images vertically: to/middle/bottom



-3 rules for creating images:

+Save images in the right format: jpeg, gif, png

+Save images at the right size

+Use the correct resolution

-Tools to edit and save images: Adobe Photoshop

-Image formats:

+jpeg: have many different colors in picture

+gif/png: few colors or large areas of same color

-Image dimensions: The images you use on website should be saved at the save with and weight that you want them to appear on page

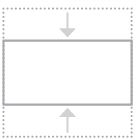
+reducing image size: quicker to download



+increasing image size: look blurry or blocky

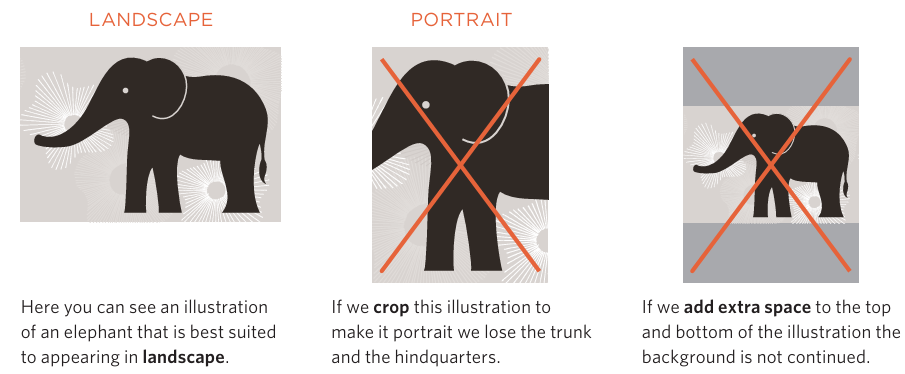


+changing shape: only some images can be cropped and still make sense



-Cropping images: When cropping images it’s important not to lose valuable information. It’s best to source images that are the correct shape if possible



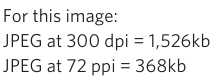


-Image Resolution

+Images created for web should be saved at a resolution of 72 ppi. The higher resolution of image, the larger the size of file

+JPGs, GIFs, PNGs belong to a type of image format **bitmap**: made up of lots of miniature squares. The resolution of images is the number of squares that fit within 1inch x 1nch square area

+Images appearing on computer screens are made up of squares called pixels. The web browsers on most desktop computers display images at a resolution of **72 pixels per inch** (ppi). Images in print materials (books, magazines) are made up of tiny circle called dots. These image printed at a resolution of 300 dots per inch (dpi)



+Computer displays are capped at a resolution of 72 ppi, using images on web with higher resolution won’t result in better image quality.

-Vector images

+Created in programs like Adobe Illustrator.

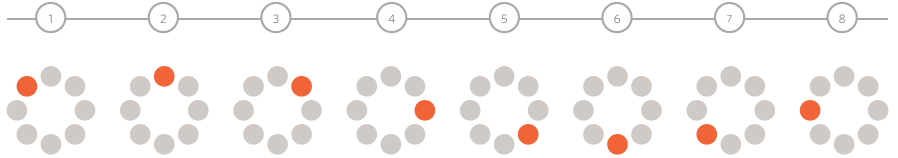
+Image is a line drawing (logo, illustration, diagram).

+Vector images are created by placing points on a grid, and drawing lines between those points. A color can then be added to fill in the lines -> increase dimensions of image without affecting the quality

+SVG are format used to display vector images on web.

-Animated gifs

+Show several frames of an image in sequence and therefore can be used to create simple animations

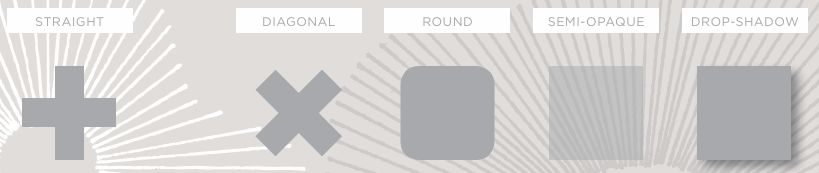


+Each extra frame of image increases the size of file.

+GIFs are only suitable for simple illustrations.

-Transparency (see-through): select one of 2 formats

+Transparent GIF: transparent part of image has straight edges and 100% transparent (not semi-opaque)

+PNG: transparent part of image has diagonal or rounded edges or if you want a semi-opaque transparency or a drop-shadow

-HTML5: Figure and figure caption

+<figure>: contain images and caption so that the two are associated. You can have more than one image inside <figure> as long as they all share same caption

+<figcaption>: allow web page authors to add a caption to an image.



-Summary:

+<image> 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 web page and in the appropriate format.

+Photographs are best saved as JPEGs; illustrations or logos that uses flat colors are better saved as GIFs.

# Chap 6: Tables

-A table represents information in a grid format. 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.

-Basic table structure

+<table>: create table

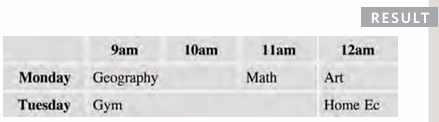
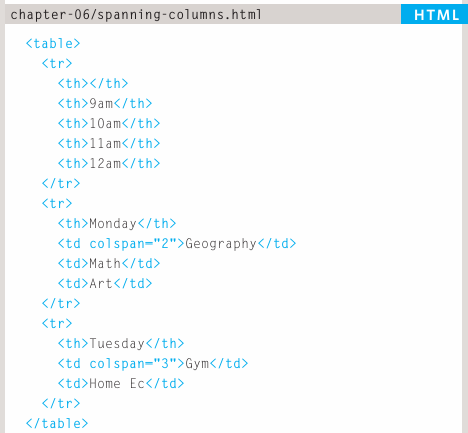
+<tr>: start of each row

+<td>: each cell of table

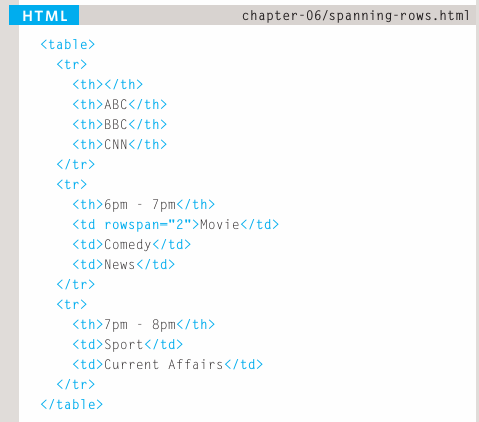


-Table heading: <th>

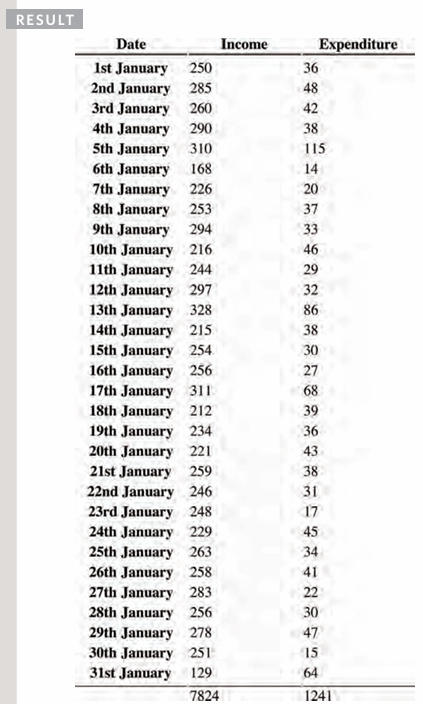
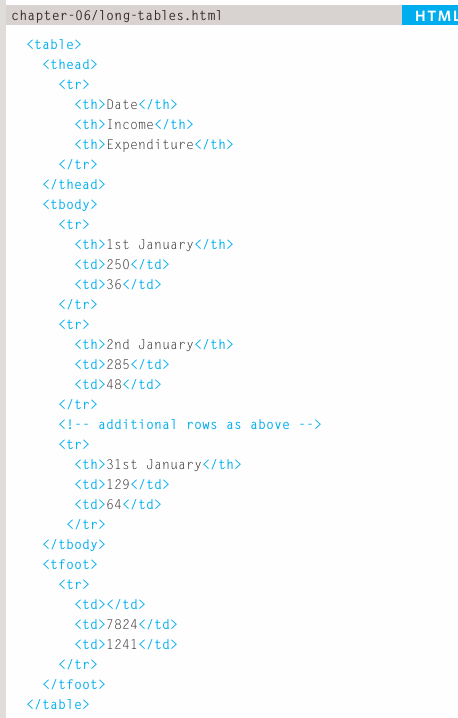
-Spanning columns:



-Spanning rows:



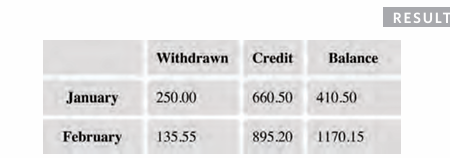
-Long tables: <thead> <tbody> <tfoot>



-Old code:

+width: how wide that table should be

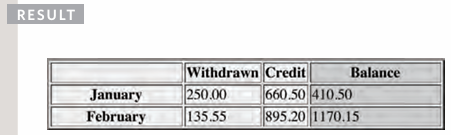
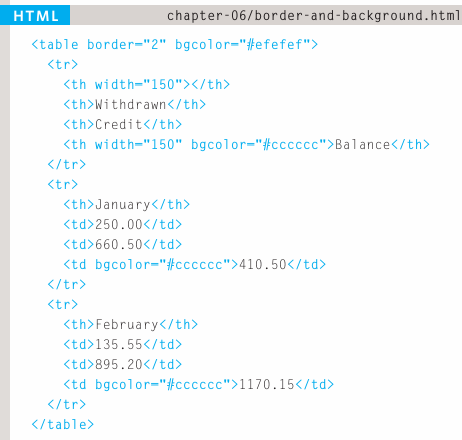
+cellpading: add space inside each cell of table, cellspacing: space between each cell of table.



-Old code: Border & background

+border: the width of the border in pixels (table+td)

+bgcolor: indicate background colors of table or individual table cells.



-Summary:

+The <table> element is used to add tables to a web page.

+A table drawn out row by row. Each row is created with <tr> element.

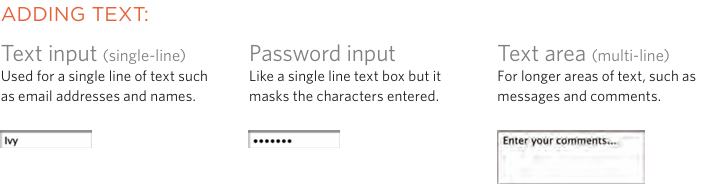
+Inside each row there are a number of cells represented by <td> element (or <th> if it is a header)

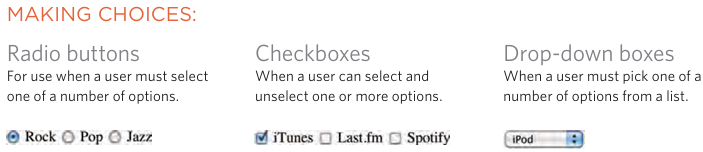
+You can make cells of a table span more than one row or column using rowspan and colspan attributes

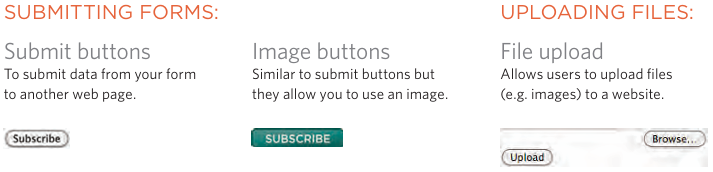
+For long tables you an split the table into a <thead>, <tbody>, <tfoot>

# Chap 7: Forms

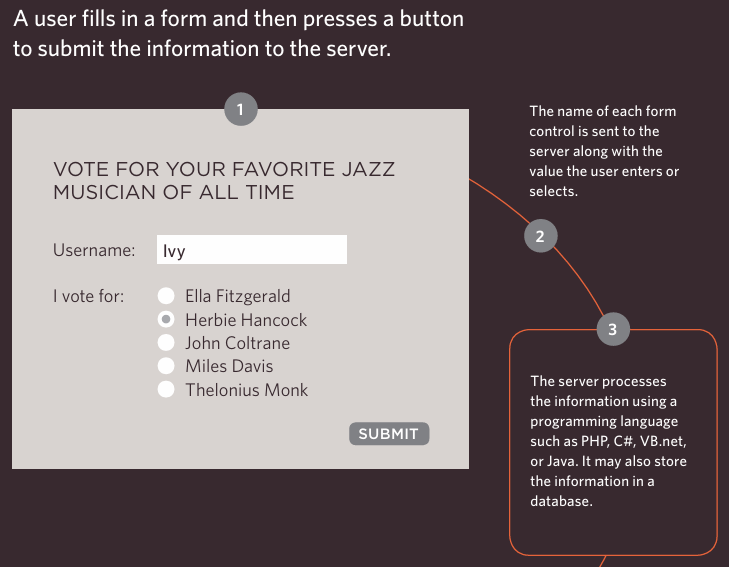
-Form controls:

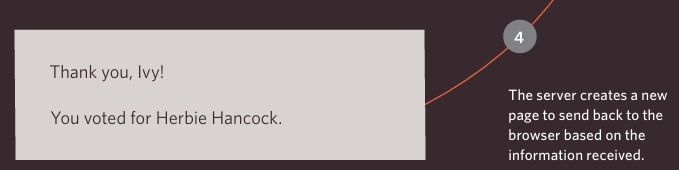


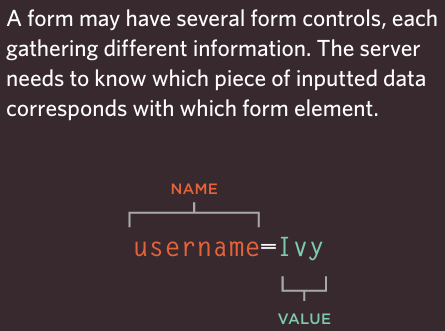


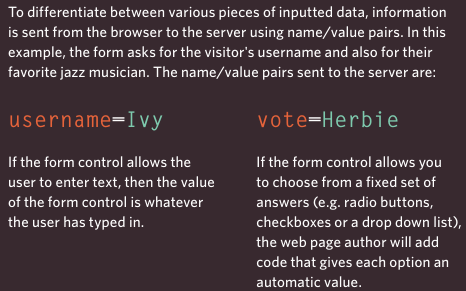


-How forms work:

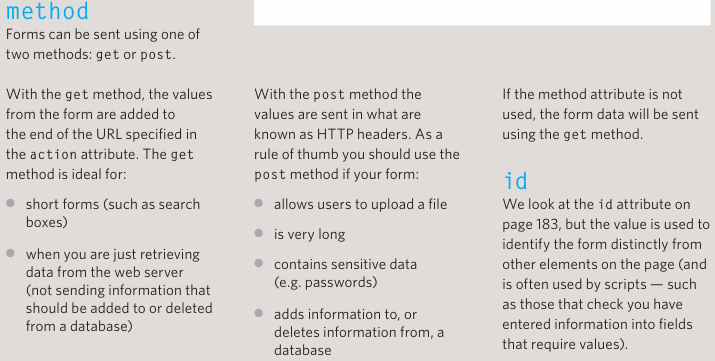








-Form structure: <form> <action> <method>



-Text input:

+<input>: create different form controles.

+type=”text”: determine what kind of input

+name: identify the form control and is sent along with information they enter to server

+size: indicate the width of text input

+maxlength: limit the number of characters a user enters



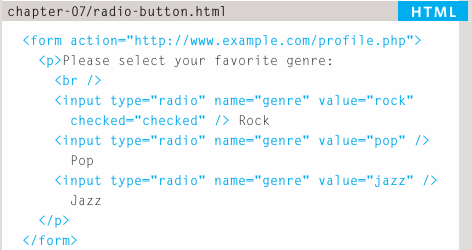
-Password input:



-Text area: create a multi-line text input.



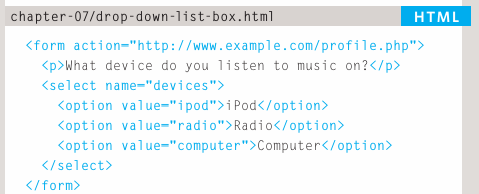
-Radio button: type=”radio”, value, checked



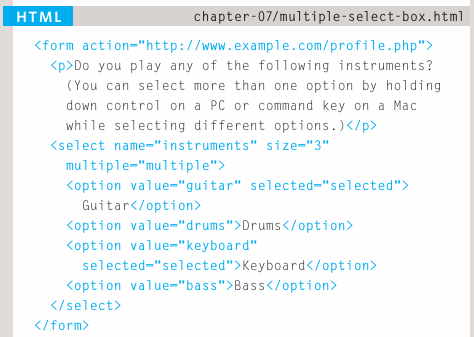
-Checkbox:



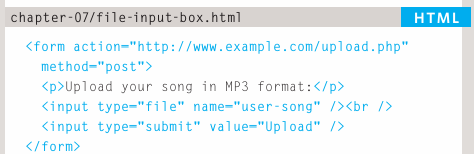
-Drop down list box: <select name> <option value selected>



-Multiple select box: <select size multiple>



-File input box:



-Submit cotton



+Image button:



-Button & Hidden controls

+<button> allow user more control over how their button appear, allow other elements to appear inside the button.

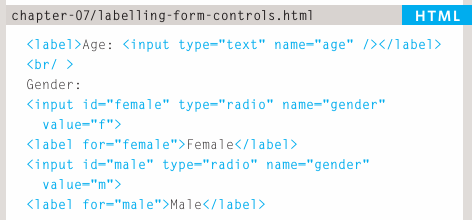
+type=”hidden”: not shown. They allow coder to add values to forms that users can’t see.



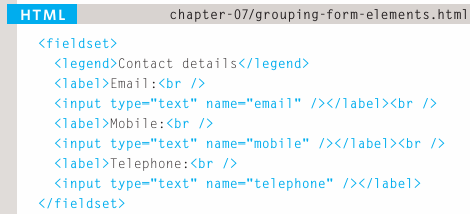
-Labelling form controls:

+<label> wrap text description and form input + kept separate from form control and use for attribute to indicate which form control it is a label for.

+for: states which form control the label belongs to.



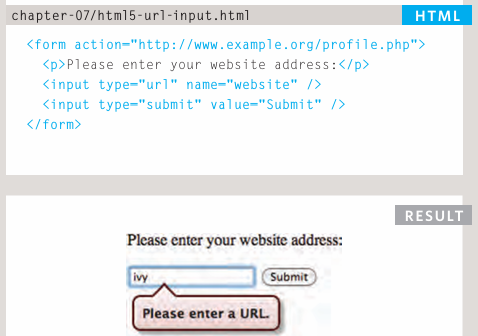
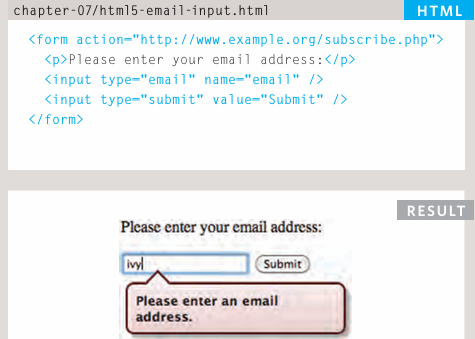
-Grouping form elements: <fieldset>+<legend>



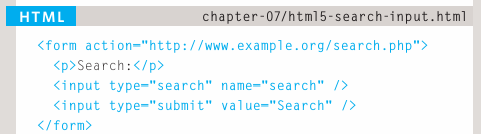
-HTML5: Data input



-HTML5: Email & URL input



-HTML5: Search input



-Summary:

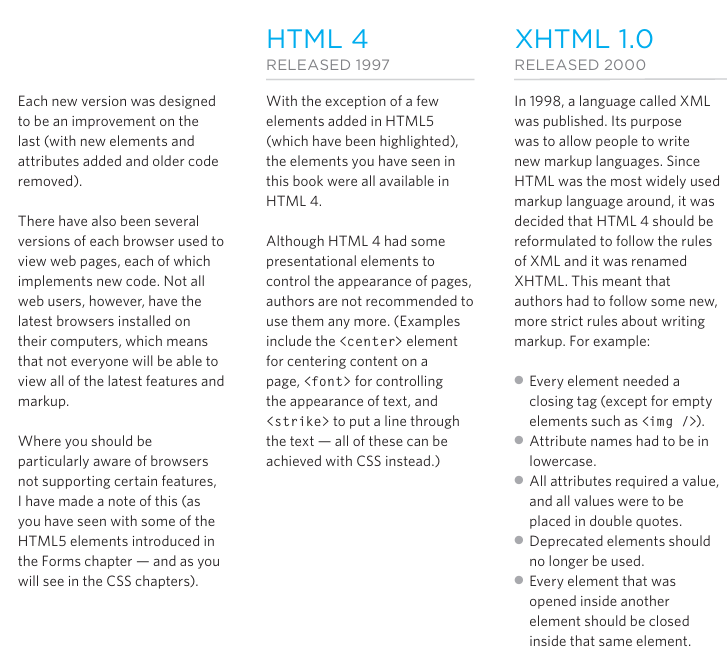
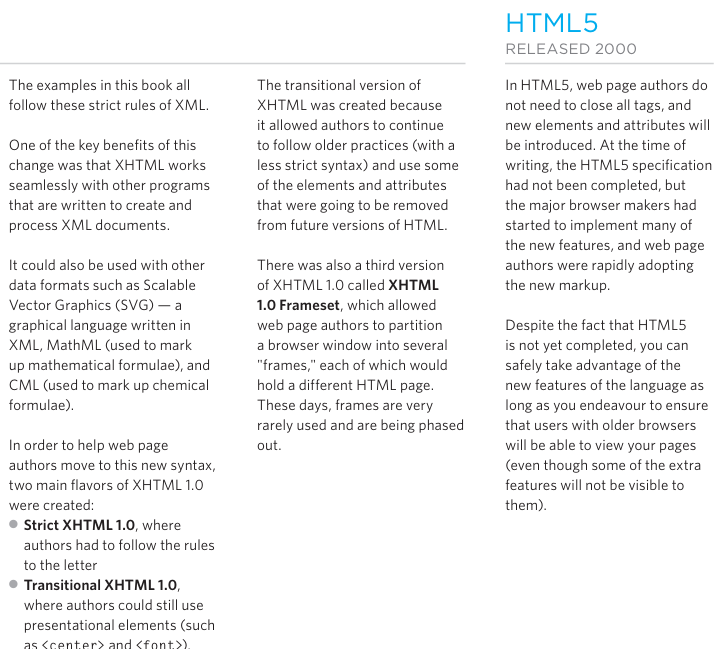
+Whenever you want to collect information from visitors you will need a form, which lives inside <form> element

+Information from a form is sent in name/value pairs

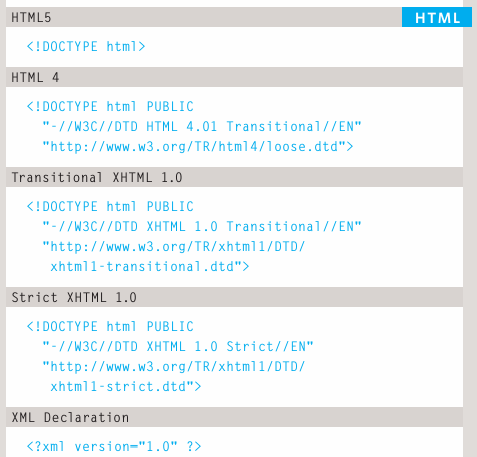
+Each form control is given a name, and the text the user types in or the values of options they select are sent to server.  
+HTML5 introduces new form elements which make it easier for visitors to fill in forms

# Chap 8: Extra Markup

-The evolution of HTML

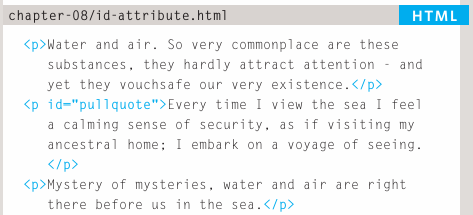
  


-DOCTYPES: tell a browser which version of HTML the page is using

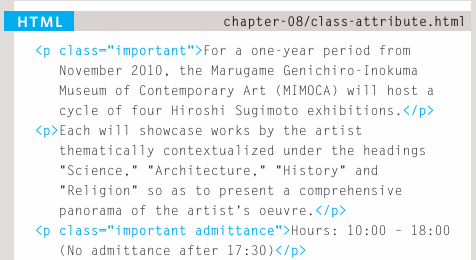


-Comments in HTML: <!-- -->

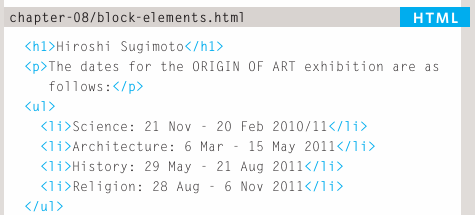
-id attribute: uniquely identify that element from other elements on page. It allows to style it differently than any other instance of same element on page+allow the script to work with that particular element.



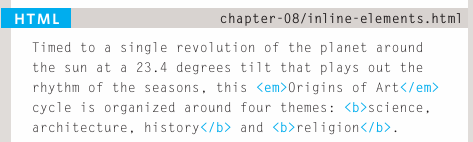
-class attribute: identify several elements as being different from other elements on page.



-block element: elements always appear to start on a new line in browser window: <h1> <p> <ul> <li>

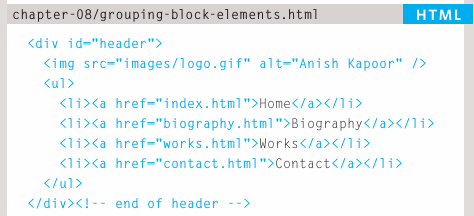


-Inline elements: appear to continue on same line as their neighbouring elements: <a> <b> <em> <img>

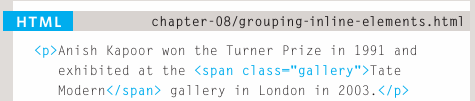


-Grouping text & elements in a block:

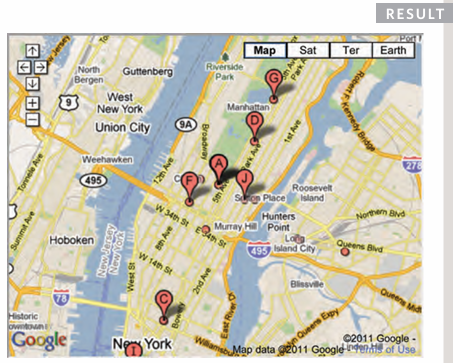
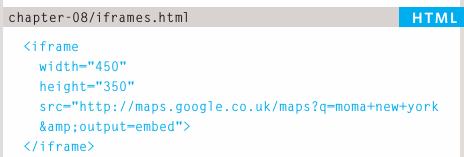
+<div>: group a set of elements together in one block-level box.



-Grouping text & elements inline: <span>

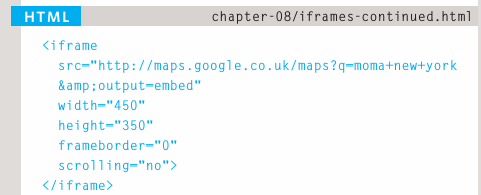


-iframe: a little window cut into the page to see another page: <iframe src height>



+scrolling + frameborder: not supported in HTML5

+seamless: in HTML5



-Information about your pages:

+<meta>: inside <head>, contains information about that web page.

+ name=”description” contains a description of page. It used by search engines to understand what page is about and should be a maximum of 155 characters. Sometimes it’s also displayed in search engine results

+name=”keywords”: contains a list of comma-separated words that user might search on to find the page.

+name=”robot”: indicate whether search engines should add this page to search result or not. noindex: the page shouldn’t added. nofollow: search engine should add this page in their results but not any pages that it links to.

+http-equiv

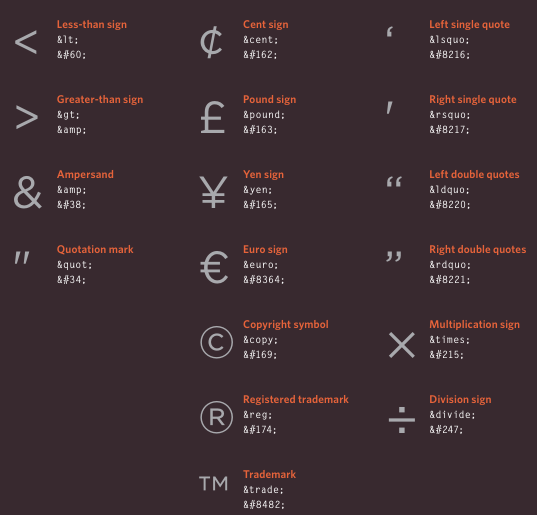
author: defines the author of web page.

pragma: prevent the browser from caching the page (storing it locally to save time downloading it on subsequent visit)

expires: indicate when the page should expire (no longer be cached). The data must be specified in format



-Escape characters:



-Summary

+DOCTYPES tell browsers which version of HTML

+Add comments in <!-- -->

+id and class attributes allow you to identify particular elements

+<div> and <span> elements allow you to group block-level and inline elements together

+<iframes> cut windows into web pages through which other pages can be displayed

+<meta> supplies all kinds of information about page

+Escape characters include special characters: <, >,…

# Chap 9: Flash, Video and Audio

-Flash is a very popular technology to add animations, video and audio to websites.

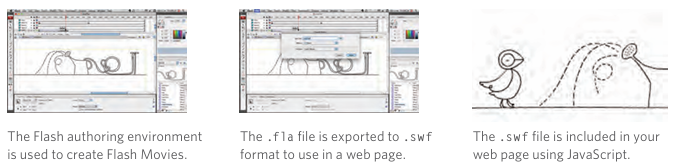
-How flash works:

+Purchase Flash authoring environment from Adobe or companies that offer Flash animations and slideshows, video and audio

+Create .fla file as Flash file. To use this file on web page, it has to be saved in SWF (.swf)

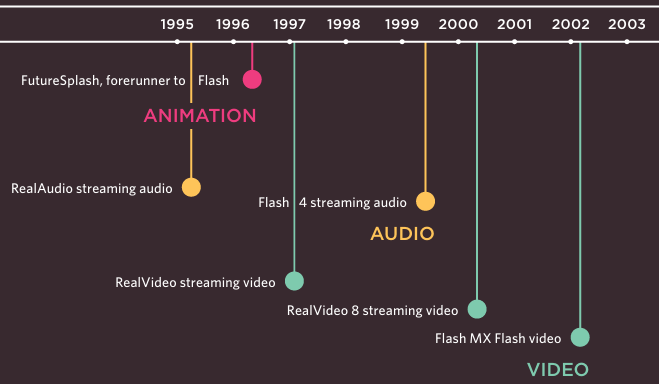
+When exporting movie into SWF format, Flash creates code used to embed the Flash movie in page. This code used HTML <object> and <embed> tags. Now it’s more common to use JS

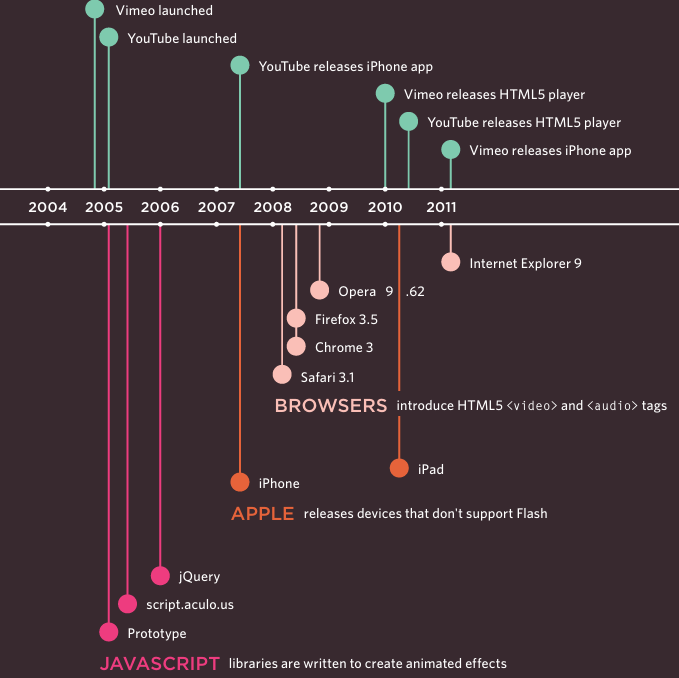
+To view Flash, browsers need to use a plugin (an extra piece of software that runs in browser) Flash Player.



-Use of Flash: Since 2005, fewer websites are written in Flash or even use elements of Flash in their pages.

-Timeline: flash, video & audio



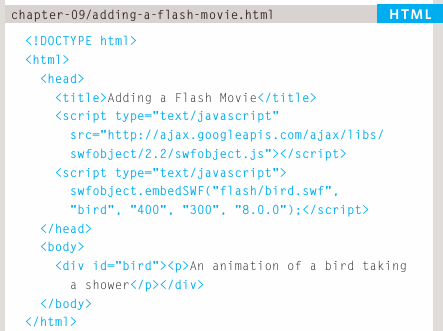


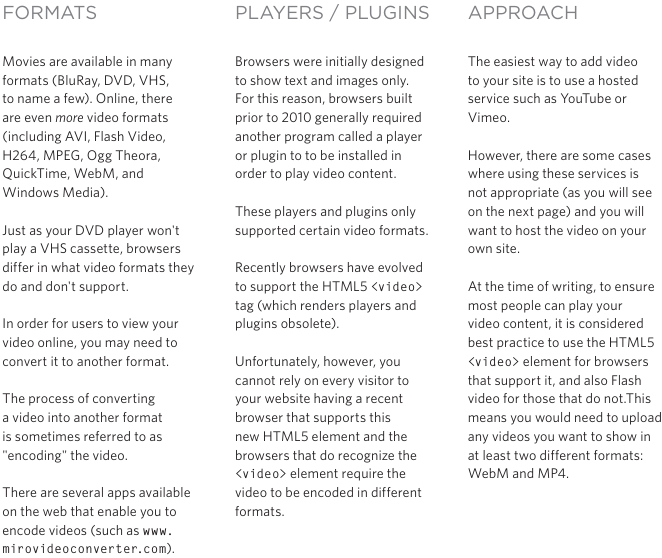
-Adding a flash movie to your web page:

+The most popular way of adding Flash into web page is using JavaScript. There are several scripts that allow to do this without an in-deph understanding of JS

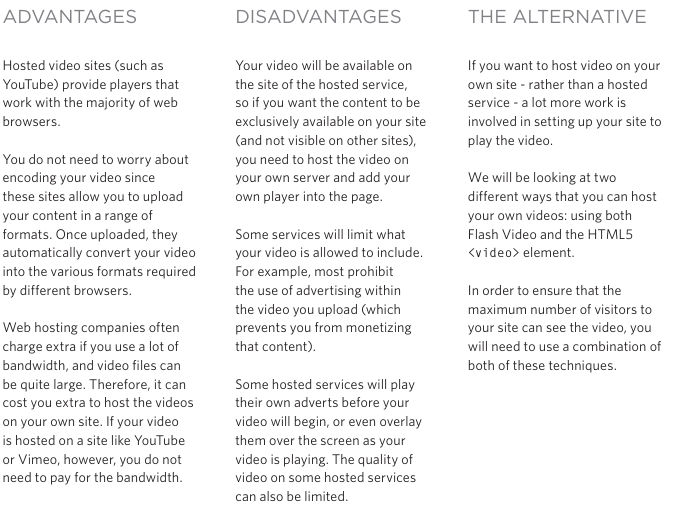
+The script we will be looking at is SWFObject. You can obtain a copy of it from Google.

+One advantage is allow browsers to show alternative content for users whose browsers aren’t capable of showing Flash.

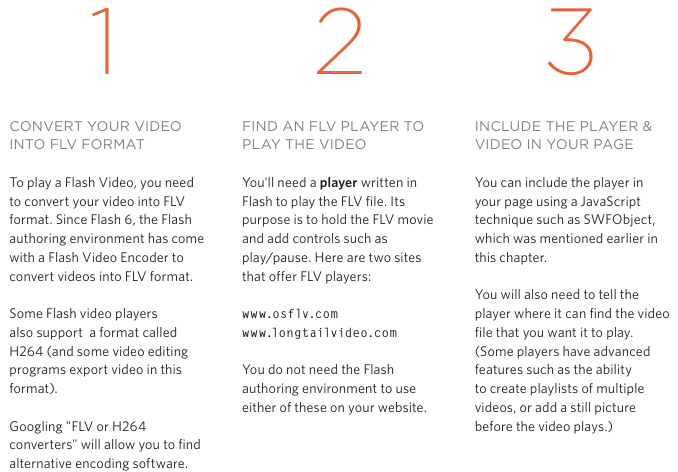
  
-Understanding video formats and players: There are 2 key issues to add video to your site: file formats and video players/plugins



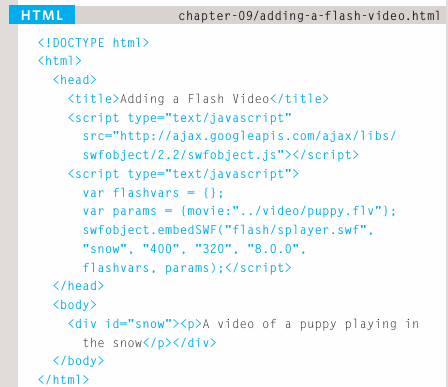
-Using hosted video services: The easiest way to add a video to site is to upload the video to a site (Youbute/Vimeo) and use features provided on their site to embed the video in your page.



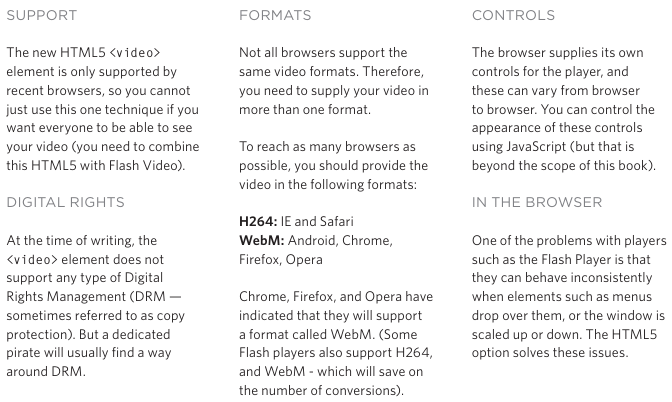
-Preparing a flash video for your site



-Adding a flash video to your page:



-HTML5: Preparing video for pages



-HTML5: Adding video to your pages: <video> has attributes to control video playback

+src: specify the path to the video

+poster: specify an image to show while the video is downloading or until the user tells the video to play.

+width, height: specify the size of player in pixels

+controls: indicates that the browser should supply its own controls for playback.

+autoplay: indicate the file should play automatically

+loop: indicate the video start playing again once it has ended

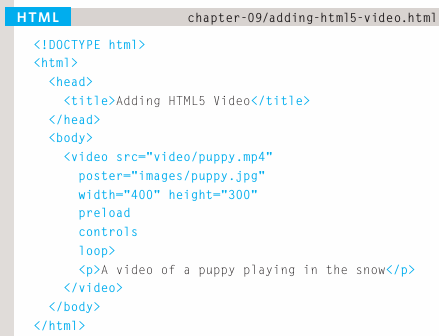
+preload: tell the browser what to do when page loads. It can have one of 3 values:

none: browser should not load video until user presses play

auto: browser should download the video when page loads

metadata: browser should just collect information like size, first frame, track list, duration.

+In HTML5, don’t need to supply values for all attributes like controls, autoplay, loop attributes. These attributes are like on/off switches.

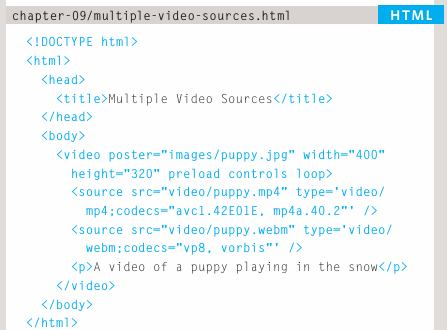


-HTML5: Multiple video sources: <source> specify the location of file to be placed (it replaced src). You can also use multiple <source> elements to specify that the video is available in different formats.

+src: specify the path to video

+type: tell the browser what format the video is. Otherwise, it will download some of video to see if it can play the file.

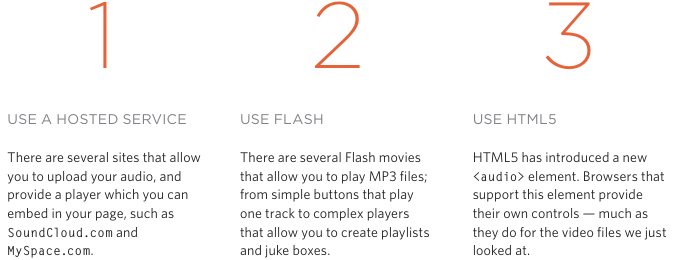
+codecs: encode the video is supplied within type attribute. Note the use of single quote, as well as double quotes in type attribute.



-HTML5: Combining flash & HTML5 video

+You may choose to offer HTML5 as 1st option, and flash video as a fallback for people whose browser doesn’t support HTML5 videol

-Adding audio to web pages



-Adding a flash mp3 player:



-HTML5: adding html5 audio to pages: <audio> include audio files in pages

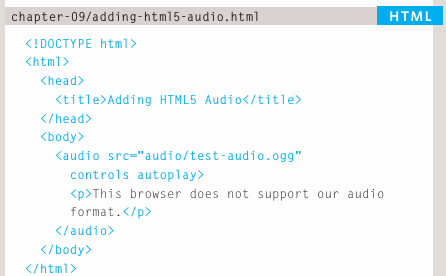
+src: specify path to audio file

+controls: indicate whether the player should display controls.

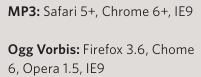
+autoplay:

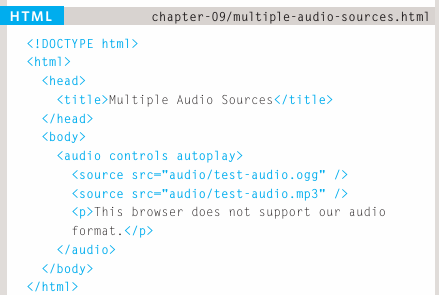
+preload

+loop



-HTML5: Multiple audio source

Different browsers support different formats for audio files: 



-Summary:

+Flash allows you to add animations, video and audio to the web.

+Flash is not supported on iPhone or iPad.

+HTML5 introduces <video> and <audio> elements for adding video and audio to web pages, but these are only supported in the latest browsers.

+Browsers that support HTML5 elements don’t all support the same video and audio formats, so you need to supply your files in different formats to ensure that everyone can see/hear them.