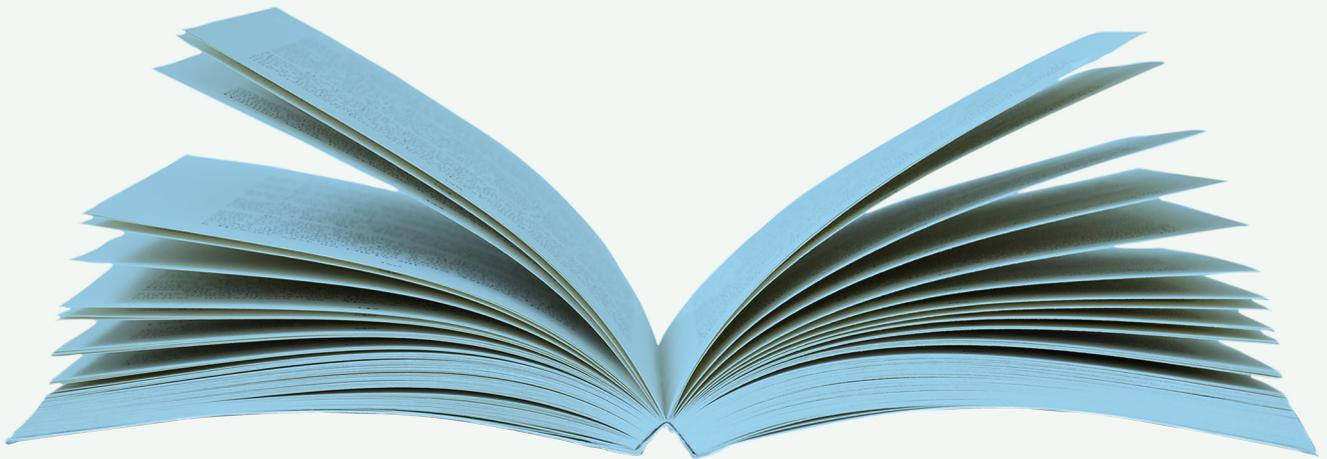




- *Free to use.*
自由編輯運用
- *Free to change.*
共享優質課本
- *Free to share.*

IT S234 Web Site Design (Free courseware)



香港公開大學
THE OPEN UNIVERSITY
OF HONG KONG



© The Open University of Hong Kong

Printing History

January 2015: First adopted in Open Textbooks



This work is licensed under a [Creative Commons-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/)

Original source: Free Courseware: IT S234 Web Site Design

<http://olefree.ouhk.edu.hk/course1126/its234.nsf>

Contents

Chapter 1 Overview of standard Web technologies	1
1.1 About this module	1
1.2 Introduction	2
1.3 Overview of standard Web technologies.....	2
1.3.1 Hypertext Markup Language (HTML).....	2
1.3.1.1 Activity 1	5
1.3.2 Cascading Style Sheets (CSS).....	6
1.3.2.1 Activity 2	10
1.3.2.2 Suggested answers to Activity 2	11
1.3.3 Extensible Markup Language (XML).....	11
1.3.3.1 Activity 3	13
1.3.3.2 Suggested answers to activity 3	14
1.3.3.3 More on Extensible Markup Language (XML).....	14
1.3.3.4 Activity 4	15
1.3.3.5 Suggested answers to Activity 4	15
1.3.4 XHTML.....	15
1.3.4.1 Activity 5	18
1.3.4.2 Suggested answers to Activity 5	19
1.3.4.3 Activity 6	19
1.3.4.4 Suggested answers to Activity 6	20
1.3.5 JavaScript.....	20
JavaScript and JScript	21
1.3.5.1 Activity 7	22
1.3.5.2 Suggested answers to Activity 7	23
1.3.6 Server-side scripting	23
1.3.6.1 Activity 8	24
1.3.6.2 Suggested answers to Activity 8	24
1.3.6.3 More on Server-side scripting	25
1.3.6.4 Activity 9	25
1.3.7 Web authoring tools	25
1.3.7.1 Code editors.....	26
1.3.7.1.1 Activity 10.....	27
1.3.7.2 Image editors.....	27
1.3.7.3 Web browsers.....	28
1.4 Self-test.....	28
Create an online résumé	28
1.5 References.....	29
1.6 Conclusion.....	29

Chapter 1

Overview of standard Web technologies

1.1 About this module



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

Welcome to this free courseware module 'Overview of standard Web technologies'!

This module is taken from the OUHK course *IT S234 Web Site Design* (http://www.ouhk.edu.hk/wcsprd/Satellite?pagename=OUHK/tcGenericPage2010&c=C_ETPU&cid=191154044800&lang=eng), a ten-credit, middle level course that is included in a number of programmes offered by the [School of Science and Technology \(http://www.ouhk.edu.hk/wcsprd/Satellite?pagename=OUHK/tcSubWeb&l=C_ST&lid=191133000200&lang=eng\)](http://www.ouhk.edu.hk/wcsprd/Satellite?pagename=OUHK/tcSubWeb&l=C_ST&lid=191133000200&lang=eng) of the OUHK. The course builds on the basic Internet skills taught in *IT S123* (http://www.ouhk.edu.hk/wcsprd/Satellite?pagename=OUHK/tcGenericPage2010&c=C_ETPU&cid=191154085200&lang=eng) to equip learners with the practical skills needed to design and manage websites for small-scale Web commerce and small organizations. In addition you will also become familiar with developing trends in the technology and how these are likely to influence website design.

IT S234 is mainly presented in printed format and comprises ten study units. Each unit contains study content, activities, self-tests, assigned readings, etc for students' self-learning. **This module (The materials for this module, taken from Unit 1 of the print-based course *IT S234*, have been specially adapted to make them more suitable for studying online. In addition to this topic on 'Overview of standard Web technologies', the original unit also includes the topics 'website review and evaluation', 'other design issues', 'establishing your web presence', 'Web browser' and 'the website design process'.)** retains most of these elements, so you can have a taste of what an OUHK course is like. Please note that no credits can be earned on completion of this module. If you would like to pursue it further, you are welcome to enrol in *IT S234 Web Site Design* (http://www.ouhk.edu.hk/wcsprd/Satellite?pagename=OUHK/tcGenericPage2010&c=C_ETPU&cid=191154044800&lang=eng).

This module will take you about **six hours** to complete, including the time for completing the activities and self-tests (but not including the time for assigned readings).

Good luck, and enjoy your study!

1.2 Introduction



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

Websites are created using a set of standard technologies. Many of these technologies involve markup and programming languages for encoding page content and specifying the appearance and interactive behaviour of Web elements. In this module, you are introduced to the most common Web development technologies promoted by the [World Wide Web Consortium \(http://www.w3.org/\)](http://www.w3.org/). You will also be guided as to how you can incorporate some of these technologies into your own webpages without the need to learn the details of the associated language.

1.3 Overview of standard Web technologies



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

'If you're going to build a house — or a website — you need the right tools. If your house's foundation is good, the house will stand for years despite the changing world around it.' I saw this excellent analogy describing the standard Web technologies for building websites on the popular reference site, [A List Apart \(http://alistapart.com/article/ALAprimer\)](http://alistapart.com/article/ALAprimer).

We will take a tour of the standard languages and technologies commonly used to build websites today. Many of these standards are promoted by the World Wide Web Consortium (W3C), such as HTML, XML, XHTML, Cascading Style Sheets and standardized JavaScript. We have a greater chance of creating reliable and well-formed websites by observing and using these standards. Using these standards also helps ensure that our sites will keep working even when they are viewed in new browser versions or even new kinds of devices in the future.

World Wide Web Consortium (W3C)

The W3C is an international organization which oversees the development and promotion of the standard languages used by the people who make websites. The W3C also maintains the updated and detailed specifications of these standards online at the W3C website(www.w3c.org).

1.3.1 Hypertext Markup Language (HTML)



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

HTML is the universal language spoken on the Web. Any Web browser running on any hardware platform can read and interpret your documents if they're written in HTML. It follows that writing HTML is the most basic skill you need as a Web designer.

HTML is a markup language. For example, within a document, we use the `<h1>` and `</h1>` tags to surround or contain all the level one headings, `<p>` and `</p>` to contain paragraphs, `` and `` to contain a list, `` and `` to mark the items within a list, and so on. These are examples of **structural markup tags** (Structural markup tags describe the purpose of text without specifying how they should be displayed.).

markup language

You may have heard "markup language" before, but what does it mean, exactly? Well, when we apply markup (i.e. add HTML tags) to a document, we are essentially dividing the document into parts.

Figure 1.1 shows an example of structural markup tags and how the example displays in a Web browser.

```
<ul>
<li>Chocolate</li>
<li>Vanilla</li>
<li>Strawberry</li>
<li>Cookies and Cream</li>
</ul>
```

Fig. 1.1: An example of structural markup tags

HTML rendering result:

- Chocolate
- Vanilla
- Strawberry
- Cookies and Cream

On the other hand, HTML also contains markup tags that describe how the content is supposed to look rather than what part of the document it is. These types of tags are called presentation tags.

presentation tags

Examples of presentation tags are the `` and `` tags used to apply boldface to text, the `<i>` and `</i>` tags used to italicize text, the `
` tag used to insert a new line, the `` and `<emphaisis>` tags to emphasize text, and the `` and `` tags used to specify text styles, sizes and colour.

Figure 1.2 shows an example of presentation tag.

```
If I wanted this line to be displayed in <i>italics</i>
I would use italic tags.
```

Fig. 1.2: An example of presentation tag

HTML rendering result:

If I wanted this line to be displayed in *italics* I would use italic tags.

There is another set of tags that requires the browser to perform special functions associated with the element. These types of tags can be classified as processed and interactive markup.

processed and interactive markup

Examples of processed and interactive markup are the `<a>` and `` tags used to mark up hyperlinks, the `` tag used to include any inline images in the document, and the `<input>` tag used to define any form elements. The browser not only needs to know how to display such elements, it also needs to know how to process the interaction between the user and these elements.

Figure 1.3 shows an example of such markup.

```
Please refer to our
<a href='http://www.speedyoffice.com.hk'>website</a> for more details.
```

Fig. 1.3: An example of such markup

HTML rendering result:

Please refer to our [website](http://www.speedyoffice.com.hk/) for more details.

Note: The website that the above hyperlink refers to is fictitious. It doesn't exist in the real world.

There's an excellent tutorial at www.w3schools.com/html/default.asp in case you need more HTML practice. Now complete the following activity as a review of HTML.

HTML was invented by [Tim Berners-Lee](http://en.wikipedia.org/wiki/Tim_Berners-Lee) (http://en.wikipedia.org/wiki/Tim_Berners-Lee), who based it on **Standard Generalized Markup Language (SGML)** (SGML is an international standard for marking up text for presentation on a variety of physical devices). The basic idea of SGML is that the document's structure should be separate from its presentation.

If this is the case, you may be wondering why HTML includes presentation tags and attributes.

why HTML includes presentation tags and attributes

Well, there were no clear standards for the HTML language during its early years. Browser software companies added many non-standard features into the language, and these features eventually found their way into the standard specification. Most presentational markup elements have become deprecated under the HTML 4.01 specification. Deprecated tags will no longer be supported in current and future specifications of a markup language, so we need to avoid using such tags. The HTML 4.01 specification recommends using Cascading Style Sheets (CSS) for designing page layout and appearance instead.

1.3.1.1 Activity 1



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

creativecommons.org/licenses/by-sa/4.0/).

Download the Speedy Office Supplies case study Web site, which is available as one zipped file *u234case2.zip*.

[u234case2.zip](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1672/media/u234case2.zip) (<http://www.opentextbooks.org.hk/system/files/resource/1/1666/1672/media/u234case2.zip>)

Unzip all the files to your local drive. Browse the pages locally and spend some time getting familiar with the content and page layouts in the site.

Use HTML structural tags to do the following:

- Include an announcement regarding the weekly sale items on the home page (*index.html*).

You are given the details for the announcement in the file *sale_items.txt*.

[sale_items.txt](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1672/media/sale_items.txt) (http://www.opentextbooks.org.hk/system/files/resource/1/1666/1672/media/sale_items.txt)

A screen shot below showing how the announcement may be incorporated on the home page.

- ▶ Home
- ▶ Products
- ▶ Services
- ▶ About
- ▶ Contact Us

FREE DELIVERY for all orders above \$500

Flat fee of \$15 for all other orders. Same day or next day delivery available.

WEEKLY SALES OFFERS

1. Cadenza White, 5000 Sheets Premium Multi-Use Paper 8-1/2" x 11"
 - Item # CADE2747 , Style # 3P5806
 - For a limited time: Buy 3, Get 1 FREE

To take advantage of this FREE OFFER, use promotion code "CPIM" in your next order.

Special offer is limited to three per order.
2. QualiMark Brown Envelope with string 12" x 15"
 - Item # 20104084 , Style # QMBE9295
 - For a limited time: HKD 35.00

To take advantage of this SPECIAL OFFER, use promotion code "QMPK" in your next order.

Special offer is limited to ten per order.



1.3.2 Cascading Style Sheets (CSS)



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

As you've seen earlier, HTML is made up of tags that **define the structure of documents** (e.g. titles, headings, paragraphs, lists) as well as tags that **describe the way an element should be displayed in the Web browser** (e.g. fonts). Mixing up structural tags and presentation tags, the way HTML does, can lead to code that is difficult to understand and maintain.

Think of the work involved if you wanted to change the font colour to blue and size to '+2' for all headings in the HTML code shown in [Figure 1.4](#). As you can see, you would have to update the tags on all three headings separately.

Original
<pre><h3>Heading One</h3> <p>This text appears under Heading One.</p> <h3>Heading Two</h3> <p> This text appears under Heading Two.</p> <h3>Heading Three</h3> <p> This text appears under Heading Three.</p></pre>
Headings: change font colour to blue, size to "+2"
<pre><h3>Heading One</h3> <p>This text appears under Heading One.</p> <h3>Heading Two</h3> <p> This text appears under Heading Two.</p> <h3>Heading Three</h3> <p> This text appears under Heading Three.</p></pre>

Fig. 1.4: All headings in the HTML code

HTML rendering result:

Heading One

This text appears under Heading One.

Heading Two

This text appears under Heading Two.

Heading Three

This text appears under Heading Three.

Heading One

This text appears under Heading One.

Heading Two

This text appears under Heading Two.

Heading Three

This text appears under Heading Three.

Now think of all the work to be done if this update needs to be made over and over again on all the pages on your site! This is not just very tedious work, it's also very easy to miss one or two spots, leading to an inconsistent appearance of level-three headings on your site.

Now let's see what happens if we use style sheets instead. If you've worked with desktop publishing or word processing software, chances are you've used styles to format your document. Styles are formatting rules that you apply to the elements in your documents. [Figure 1.5](#) is the example in [Figure 1.4](#) using style sheets for formatting.

```
<html>
<head>
<title>CSS Example</title>
<style type='text/css'>
  h3 {font: large; color: green}
</style>
</head>
<body>
<h3>Heading One</h3>
<p>This text appears under Heading One.</p>
<h3>Heading Two</h3>
<p> This text appears under Heading Two.</p>
<h3>Heading Three</h3>
<p> This text appears under Heading Three.</p>
</body>
</html>
```

Fig. 1.5: Using style sheets for formatting

HTML rendering result:

Heading One

This text appears under Heading One.

Heading Two

This text appears under Heading Two.

Heading Three

This text appears under Heading Three.

The style sheet is surrounded by the `<style>` and `</style>` tags within the head of the HTML document (see [Figure 1.6](#)). The `h3` element is followed by a list of the formatting rules that will be applied to it, which would display all level three headings as large green text.

```

<head>
<title>CSS Example</title>
<style type='text/css'>
  h3 {font: x-large; color: blue}
</style>
</head>

```

Fig. 1.6: The <style> and <style> tags

Since all the formatting rules are confined to a single location, you can update the style for level three headings in one place only. No more mixing up your structural and your presentation tags in the body of the HTML page.

If you need to use the same style on multiple pages, you can even save the styles in a separate stylesheet file (called 'styles.css', for example) and have multiple webpages link to the same stylesheet, as Figure 1.7 shows.

```

<HEAD>
<TITLE>CSS Example</TITLE>
<link rel='stylesheet' type='text/css' href='styles.css' media='screen'>
</HEAD>

```

Fig. 1.7: Stylesheet file

Style sheets bring an explicit separation between the structure of a document and its presentation. To modify the presentation, the change can be restricted to the style sheet alone. Style sheets also offer greater control over fonts, colours, indents, margins, and the general look and layout of your webpages.

The following activity is meant to whet your appetite in using CSS without going into too much detail.

Aside from the appearance of elements, you probably noticed changing the style sheet used can also affect the positioning of the elements on a page.

1.3.2.1 Activity 2



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

We will view a demonstration of HTML and Cascading Style Sheets at work in this activity.

1. Visit the [CSS Zen Garden \(http://www.csszengarden.com/\)](http://www.csszengarden.com/) website.
This site was built to demonstrate what can be achieved through CSS design and through the separation of content and presentation. Note that this site was created using HTML for the content and CSS for the presentation.
2. Download the HTML and the CSS files for the page you are currently viewing by right-clicking on the given links. Open the CSS file in your text editor, such as Notepad.
Describe the styles used to format the following: body text (<body>), paragraph

text (<p>) and hyperlinks (<a:link>). Please note that you are not expected to understand what all the rules mean at this point, but browsing through the style sheet should give you an idea of how it is structured.

3. Now click on some of the designs listed on the right-hand menu bar. The same content will be reloaded, but the page formatting and style will change depending on the style sheet selected.

1.3.2.2 Suggested answers to Activity 2



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

Describe the styles used to format the following: body text (<body>), paragraph text (<p>) and hyperlinks (<a:link>) in the downloaded CSS file.

Your answer depends on the style sheet that you downloaded. I downloaded the style sheet used in the default homepage. Here are the formatting rules I saw:

- For body text — font: 8pt/16pt georgia; color: #555753; background: #fff url(blossoms.jpg) no-repeat bottom right; margin: 0px;
- For paragraph text — font: 8pt/16pt georgia; margin-top: 0px; text-align: justify;
- For hyperlinks — font-weight: bold; text-decoration: none; color: #B7A5DF;

1.3.3 Extensible Markup Language (XML)



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

HTML is the most successful markup language of all time. You can view the simplest HTML tags on virtually any device, from palmtops to mainframes. Given the success of HTML, why did the W3C create XML? To answer that question, take a look at [Figure 1.8](#).

8.

```
<h3>Fist of Fury</h3>
<p>Starring:</p>
<li>Bruce Lee</li>
<li>Nora Miao</li>
<p>Directed by:</p>
<li>Wei Lo</li>
<p>Country:</p>
<li>Hong Kong</li>
<p>Year:</p>
<li>1972</li>
```

Fig. 1.8: Sample HTML code

HTML was designed to be user-friendly. Without any special software, you and I can view the code in [Figure 1.8](#) in a text editor and understand that its content refers to the movie *Fist of Fury* and its related information, such as the country of origin, year of release, leading actors and director.

While the tags in this document tell the browser that this document is made up of headings, paragraphs and lists, it doesn't tell the browser what the information is. Humans have the native intelligence to understand the meaning of this document, but computers don't. Now why would this be a problem? Well, the Web would be a lot more useful if computers could also extract information directly from webpages, without waiting for any human assistance to help them understand and process the contents.

XML is a markup language which allows you to create your own tags, which can then be used to produce self-describing documents. Unlike HTML, which is a language just for creating webpages, XML is a language that lets you *create other languages*, hence the term 'extensible' in its name. By designing your own customized markup language, you are free to create and attach your own markup tags to the data in your own documents.

[Figure 1.9](#) shows the same code fragment shown earlier. This time, the HTML tags have been replaced with XML tags that describe the meaning of the data instead. By using XML to describe the meaning of data found on webpages, you and I can make it easier for computers to process the information on these pages.

```
<?xml version='1.0' encoding='UTF-8' ?>
<movie>
<title>Fist of Fury</title>
<actorlist>
<actor>Bruce Lee</actor>
<actor>Nora Miao</actor>
</actorlist>
<director>Wei Lo</director>
<country>Hong Kong</country>
<release_year>1972</release_year>
</movie>
```

Fig. 1.9: Sample XML code

XML also acts as the foundation for Web protocols.

Web protocols

Web protocols that are based on XML include Scalable Vector Graphics (SVG), Synchronized Multimedia Integration Language (SMIL), Wireless Markup Language (WML) and Really

Simple Syndication (RSS). These are all protocols used by computer applications to transmit data over the Web.

You will now try viewing XML data within your Web browser in the next activity. Note that XML files end with the '.xml' file extension. Like HTML, they are also text-based, which means you can open and edit them using a regular text editor such as Notepad on Windows and *vi* (<http://en.wikipedia.org/wiki/Vi>) or *emacs* (<http://en.wikipedia.org/wiki/Emacs>) on UNIX.

1.3.3.1 Activity 3



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

The file *movie.xml* contains the sample XML document in this section.

[movie.xml](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1677/media/movie.xml) (<http://www.opentextbooks.org.hk/system/files/resource/1/1666/1677/media/movie.xml>)

View the *movie.xml* file using Internet Explorer 5.0 (and up) or Mozilla Firefox. Try clicking the hyphen or plus sign next to the `<movie>` and `<actorlist>` tags.

1. What happens to the tags when you click on them?

Save the files *movie.xml* and *movie.css* to your local drive by right-clicking their links.

[movie.xml](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1677/media/movie.xml) (<http://www.opentextbooks.org.hk/system/files/resource/1/1666/1677/media/movie.xml>)

[movie.css](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1677/media/movie.css) (<http://www.opentextbooks.org.hk/system/files/resource/1/1666/1677/media/movie.css>)

2. Open *movie.xml* in a text editor and type in the following code before the opening `<movie>` tag.

```
<?xml-stylesheet type=text/css href=movie.css?>
```

Make sure you type in the exact line that you see here, including all punctuation marks and white spaces.

Save the *movie.xml* file and view the file in your Web browser again.

3. How did the appearance of the XML document change after the reference to *movie.css* was added?

1.3.3.2 Suggested answers to activity 3



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

The XML document appears as a tree or hierarchical structure. You can expand and collapse the nested tags by clicking on the hyphen (-) and plus (+) signs next to their parent tags.

```
<?xml version="1.0" encoding="UTF-8" ?>
- <movie>
  <title>Fist of Fury</title>
  - <actorlist>
    <actor>Bruce Lee</actor>
    <actor>Nora Miao</actor>
  </actorlist>
  <director>Wei Lo</director>
  <country>Hong Kong</country>
  <release_year>1972</release_year>
</movie>
```

Here is movie.xml after the reference to movie.css is added to it. The browser formats the tags in the XML document according to the styles attached to that particular tag.



1.3.3.3 More on Extensible Markup Language (XML)



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

You can see from the previous activity that XML files can be viewed as a document tree or they can be formatted using Cascading Style Sheets (CSS). There's another technology called Extensible Style Sheets (XSL), which can also be used to change the appearance of XML files when displayed. XSL is an advanced topic that is beyond the scope of this module.

When editing XML documents, you must keep in mind the following rules:

1. The document begins with a processing instruction `<?xml ...?>`. This identifies the file as an XML document and states the encoding scheme used.
2. Opening tags must have closing tags.

3. Tags must be nested properly.
4. Values in a tag must be enclosed in quotation marks.

XML documents that observe the rules above are considered *well-formed*. It takes a while to get used to XML's strictness when you've been spoiled by HTML's leniency. Just remember that this newfound discipline will pay off in the form of pages that are less error-prone and easier to maintain.

1.3.3.4 Activity 4



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

Now let's introduce a few mistakes into the XML document in [Activity 3 \(Page 13\)](#):

- Change the lowercase `<movie>` and `</movie>` tags into uppercase (i.e. `<MOVIE>` and `</MOVIE>`).
- Remove the closing `</actorlist>` tag which matches the opening `<actorlist>` tag.

Make the changes in a text editor, save the `movie.xml` file and view it again in Internet Explorer.

What does the browser do with the updated XML document?

1.3.3.5 Suggested answers to Activity 4



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

When I tried to view it on Internet Explorer, the browser was unable to display the invalid XML document. Instead, it detected the errors and showed the following message:

```
XML Parsing Error: mismatched tag. Expected: </actorlist>.
Location: file:///E:/ouhk/its234/mod_rtf/U1/activities/invalid_html.xml
Line Number 10, Column 3:</MOVIE>
--^
```

1.3.4 XHTML



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

HTML has contributed greatly to the growth of the Web because it is easy to learn and very **forgiving** (A webpage may look perfect in a Web browser even though its underlying HTML is incorrectly formed.) of errors and non-standard markup. Take a look at [Figure 1.10](#).

```
<p>If you load this HTML into a Web browser, it will  
<em>display correctly even though there are two  
mistakes in the code.</p></em>  
<p>Can you find the mistakes?
```

Fig. 1.10: Sample HTML containing errors and non-standard markup

HTML rendering result:

If you load this HTML into a Web browser, it will *display correctly even though there are two mistakes in the code.*

Can you find the mistakes?

HTML's tolerance for such inconsistencies has made it accessible to everyone, from professional Web developers to beginning Web designers. However, its flexibility comes with a price.

a price

HTML's flexibility comes with a price. The truth is that browsers need to perform additional processing in order to get around the errors that are left behind by Web developers who've grown used to undisciplined and sloppy coding. Browsers running on desktop computers may have the processing power necessary to do the extra work. However, we shouldn't forget less powerful devices, such as mobile phones, handheld computers, and Braille readers for the visually impaired. On top of that, Web browsers are being installed in more and more places, such as household appliances and automobiles. There is no way you can test each page you produce in all the possible devices that may be used to browse them.

Due to these limitations, the World Wide Web Consortium (W3C) now recommends using XHTML instead of HTML to write your webpages.

XHTML

XHTML is still the HTML that you know but written according to the rules of XML, which is much stricter. As a result, XHTML produces valid, well-formed webpages which can be viewed consistently regardless of the browser and hardware platform used.

The next activity aims to help you understand the difference between a webpage written in HTML and in XHTML. Please note that XHTML files also end with a .html extension.

Activity 5 (Page 18)

Suggested answers to Activity 5 (Page 19)

From the previous activity, you've seen that many of the HTML tags you know are still used in XHTML. You just have to be more consistent and disciplined in how you code these tags.

Now let's look at [Figure 1.11](#), which is a basic template for producing XHTML documents. Don't worry if you don't understand the details of each statement within the code. Everything will be explained in the next reading. In the meantime, just focus on the structure of the XHTML document.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
'http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd'>
<html xmlns='http://www.w3.org/1999/xhtml'>
<head>
<title>
</title>
</head>
<body>
</body>
</html>
```

Fig. 1.11: Basic template for producing XHTML documents

As you can see, the structure of an XHTML document is almost the same as an HTML document. It includes both the head and the body elements. The main difference you may have spotted would be the DOCTYPE declaration at the start of the document and the xmlns attribute of the html element.

DOCTYPE

The DOCTYPE tag tells the Web browser what version of XHTML is used. XHTML 1.0 has three versions — transitional, strict and frameset. Out of these versions, XHTML 1.0 Transitional works well in both old and new browsers.

The next reading describes the rules for writing valid XHTML. Many of the rules are derived from XML, since XHTML is just another application of XML. This online article was written for the staff of the New York Public Library but the introduction is well-written and can be applied by Web developers everywhere.

Reading 1: 'XHTML guidelines: New York Public Library Online Style Guide' (<http://legacy.www.nypl.org/styleguide/xhtml/guidelines.html>).

In the next activity, we will submit the XHTML webpage from Activity 5 to an [online validator](#) (An online validator verifies whether a webpage is coded according to the rules of writing well-formed XHTML).

Activity 6 (Page 19)

Suggested answers to Activity 6 (Page 20)

You may have realized by now that you have to be more consistent and disciplined in how you code your markup in XHTML than in HTML. To encourage you to stick with the recommended standards, the next reading describes what can be gained from authoring your pages using XHTML instead of HTML.

Reading 2: 'XHTML benefits: New York Public Library Online Style Guide (<http://legacy.www.nypl.org/styleguide/xhtml/benefits.html>)'.

As you can see, XHTML produces valid, well-formed webpages which can be viewed consistently and accurately no matter which platform is used — whether it's a desktop computer, wireless handheld device, mobile phone or Braille reader.

wireless handheld device, mobile phone or Braille reader

You may think that having your pages display properly on non-PC devices is too far off in the future for you to worry about right now, and perhaps you're right. In fact, browsers will probably have to support HTML pages the way they're currently written for a long, long time, because it's going to take a long, long while for folks to get around to converting their webpages into XHTML.

It's good to get started on the right foot with XHTML since it doesn't take a lot of effort once you get used to it. It also comes with the added bonus of producing pages that work more consistently on all browsers.

1.3.4.1 Activity 5



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

Open Internet Explorer.

View the files *night_scene1.html* and *night_scene2.html*.

[night_scene1.html](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1682/media/night_scene1.htm) (http://www.opentextbooks.org.hk/system/files/resource/1/1666/1682/media/night_scene1.htm)

[night_scene2.html](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1682/media/night_scene2.htm) (http://www.opentextbooks.org.hk/system/files/resource/1/1666/1682/media/night_scene2.htm)

night_scene1.html is written in HTML while *night_scene2.html* is written in XHTML.

1. Do you notice any difference in how the browser renders the *night_scene1.html* and *night_scene2.html* pages?

Now view the HTML source of both pages.

- Identify at least 3 differences in the source code.

1.3.4.2 Suggested answers to Activity 5



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

- There is no difference when `night_scene1.html` and `night_scene2.html` are viewed in a Web browser.
- Here are the differences between the source codes:

Night_scene1.html (HTML)	Night_scene2.html (XHTML)
HTML tags are in mixed uppercase and lowercase.	All HTML tags are in lowercase.
Not all attribute values are enclosed in double quotes.	All attribute values are enclosed in double quotes.
The IMG tag does not have a forward slash (/) in front of the closing bracket following it.	The IMG tag has a forward slash (/) in front of the closing bracket following it.
The starting HTML tag does not have any attributes.	The HTML tag has an attribute called <code>xmlns</code> .

1.3.4.3 Activity 6



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

- Download the XHTML webpage from Activity 5 to your hard disk. This would be either `night_scene1.html` or `night_scene2.html`.
- Visit the W3C's online validation service at validator.w3.org (<http://validator.w3.org/>).
- Validate the XHTML webpage using the Validate by File Upload option. Your page will be validated according to the DOCTYPE declaration used. What are the results?
- Now let's change the DOCTYPE declaration on the webpage. Using any text editor, change the DOCTYPE to XHTML 1.0 Strict instead, as follows:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" 'http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd' >
```

Note that the XHTML 1.0 Strict version does not allow any presentation tags in your HTML document. It requires that CSS should be used instead of HTML for formatting and layout purposes.

5. Save the changes to the file and validate the updated webpage using the Validate by File Upload option. What are the results this time?

1.3.4.4 Suggested answers to Activity 6



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

The validator recognizes the page '*night_scene2.html*' as valid XHTML 1.0 Transitional but not as valid XHTML 1.0 Strict. Some of the errors that prevent it from being validated as XHTML 1.0 Strict are:

- HTML attributes for formatting the page, such as bgcolor (for body tag), size, color and face (for font tag), align, height, width, border (for img tag) are used in the document, but these have been deprecated in XHTML 1.0 Strict.
- HTML tags for formatting the page, such as the font tag, are used in the document, but this tag has been deprecated in XHTML 1.0 Strict.

1.3.5 JavaScript



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

HTML gives you a certain degree of interactivity through hyperlinks. JavaScript takes this interactivity a whole lot further by making things happen as a result of your actions.

as a result of your actions

For example, you could display a second browser window pop-up with a welcome message when a page is initially loaded. Or you could navigate to a specific page within your site when a visitor makes a selection from a drop-down list. Or you could display a message in an alert window when a visitor clicks on a submit button in your form.

JavaScript is a lightweight programming language that is used to write the code that makes all this happen. Along with displaying a webpage, the browser is also responsible for interpreting and executing the JavaScript code that is included on it.

code

The code can be embedded right inside an HTML page or it can be found in an external file (with a '.js': extension) that the webpage links to, similar to the way we linked webpages to style sheets.

The video below shows how JavaScript integrates with a webpage.



Click this link to watch the video:

<http://www.opentextbooks.org.hk/system/files/resource/1/1666/1686/media/How%20JavaScript%20works.mp4>

JavaScript was created by Netscape (<http://en.wikipedia.org/wiki/Netscape>) and Sun Microsystems (http://en.wikipedia.org/wiki/Sun_Microsystems) at the height of the browser wars in the mid-1990s. Microsoft (<http://www.microsoft.com/en-us/default.aspx>) quickly followed suit by introducing its own flavour of JavaScript, called Jscript, which only worked in Internet Explorer.

JavaScript and JScript

JavaScript and JScript were roughly compatible but were not interoperable. This meant that a developer might have to write two versions of a script so that the same action could be seen on both browsers. This led to a lot of extra work and frustration for developers, as you can imagine!

Netscape eventually submitted the JavaScript specification to [ECMA International](http://www.ecma-international.org/) (<http://www.ecma-international.org/>) for standardization. **ECMAScript (Due to the popularity of JavaScript, hardly anyone ever refers to its standard version as 'ECMAScript'. We will continue to use the term 'JavaScript' in this module.)** is the name of the standardized scripting language derived from JavaScript and Jscript.

You won't need to learn how to write JavaScript in this module, but to modify and adapt existing JavaScript code (which can be downloaded from several sites on the Web) for your own use, with due respect to copyright, of course.

1.3.5.1 Activity 7



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

In this activity, we will make our button respond to our mouse click by adding JavaScript to our page.

Save the files *plain_button.html*, *popup.js* and *hello.html* to the same folder on your local drive.

[plain_button.html](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1687/media/plain_button.htm) (http://www.opentextbooks.org.hk/system/files/resource/1/1666/1687/media/plain_button.htm)

[popup.js](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1687/media/popup.js.txt) (<http://www.opentextbooks.org.hk/system/files/resource/1/1666/1687/media/popup.js.txt>)

[hello.html](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1687/media/hello.htm) (<http://www.opentextbooks.org.hk/system/files/resource/1/1666/1687/media/hello.htm>)

Open the file *plain_button.html* in your browser.

Click the button that says TALK TO ME.

1. What happens when you click the button TALK TO ME?

Now open up the file in a text editor, such as Notepad. You will see an HTML tag called `<input>` in the body of the document. This is the tag that displays the button.

Add a new attribute to this tag called `onClick`

This will cause the pop-up window to display the contents of *hello.html* when the button is clicked.

```
onClick="javascript:popUp('hello.html')"
```

When you have entered this new attribute, the complete line should look like this:

```
<input type=button value="TALK TO ME"
onClick="javascript:popUp('hello.html')">
```

Open the file *popup.js* and copy everything in it to *plain_button.html*.

This is the JavaScript code that will be triggered when the button is clicked.

Place all the code between the `<HEAD>` and `</HEAD>` tags in *plain_button.html*.

View the page in your browser and click the TALK TO ME button again.

2. After changing the code what happens when you click the TALK TO ME button?
3. What are the starting and ending HTML tags that enclose the JavaScript.

1.3.5.2 Suggested answers to Activity 7



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

- When the button that says 'TALK TO ME' is clicked, nothing happens.
- After adding the JavaScript code, the button displays a new browser window when it's clicked. The message shown in the new browser window is 'Hello, I'm glad you dropped by!'
- The tags that enclose the JavaScript code are `<SCRIPT>` and `</SCRIPT>`.

1.3.6 Server-side scripting



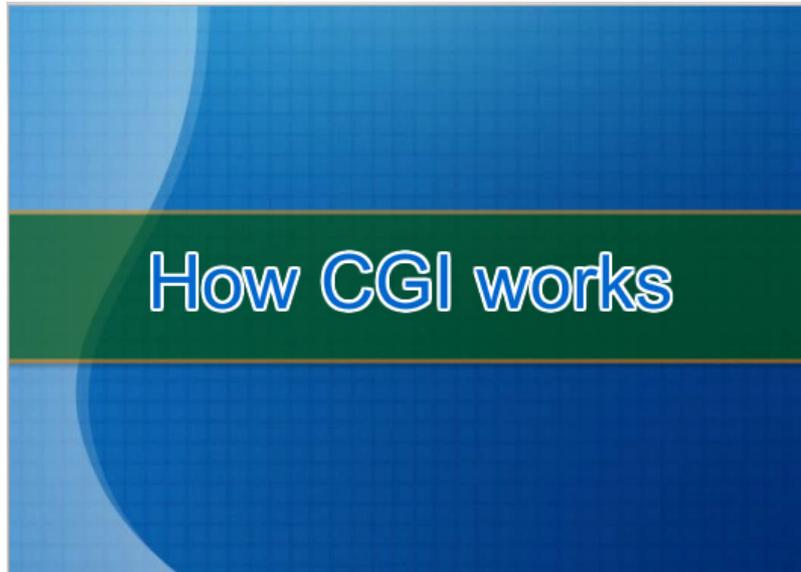
Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

All the technologies that we've previously seen, such as HTML, JavaScript and Cascading Style Sheets, are technologies that run inside your browser. Once a webpage has been loaded, along with any accompanying JavaScript or style sheet code, you are good and ready to go.

But what if you need to access a resource that is beyond the limits of your browser? You may want to search a website's catalogue for the latest and greatest digital camera from Sony, for example. In order to do this, you have to **submit** your search request via the webpage and send it to the Web server by clicking the Submit button.

Web servers are programmed to serve files, whether it's an HTML document, a style sheet or an image file. They are not designed to handle application-specific processes, such as shopping carts, online banking, or search engines. When the Web server receives a request such as the above catalogue search criteria, for example, it needs to hand it off to the program or application that has been written specifically to handle such requests.

The Common Gateway Interface, or CGI, allows Web servers to communicate with other programs running on the server so that these programs can process these user requests and send back the answers, usually in the form of a dynamically constructed webpage (as opposed to a static HTML document that is just residing on your Web server's disk). The following video (adapted from [Gundavaram 1996 \(Gundavaram, S \(1996\) CGI Programming on the World Wide Web, O'Reilly & Associates.\)](#)) illustrates how this works.



Click this link to watch the video:

<http://www.opentextbooks.org.hk/system/files/resource/1/1666/1689/media/Server-side%20scripting.mp4>

CGI, in effect, allows the functionality of your Web server to be extended, so that it doesn't just serve files, but is also capable of running applications in response to user requests and then giving them back a response. This is what makes the Web truly interactive.

1.3.6.1 Activity 8



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

Visit [Eliza \(http://www-ai.ijs.si/eliza/eliza.html\)](http://www-ai.ijs.si/eliza/eliza.html) and 'talk' to her about any topic you want.

View the HTML code behind the webpage.

1. Can you identify the Web address of the server-side script that runs behind Eliza?
2. What inputs are passed to this script? (Hint: the inputs can all be found in the form.)

1.3.6.2 Suggested answers to Activity 8



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

1. The server-side script that processes this form is found at http://www-ai.ijs.si/eliza-cgi-bin/eliza_script
2. There are two inputs sent to the server-side script for processing:
 - the message that is typed into the textbox;
 - and the Submit button.

1.3.6.3 More on Server-side scripting



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

Perl (<http://www.perl.org/>) is often mentioned in the same breath as CGI, but they are not one and the same.

they are not one and the same

CGI is the protocol which defines how application software can receive data and send responses back to a Web server. Any programming language can be used to generate webpages through the CGI protocol, although Perl is one of the most popular.

Due to the performance limitations of CGI, more efficient technologies such as [Active Server Pages \(ASP\)](http://msdn.microsoft.com/en-us/library/aa286483.aspx) (<http://msdn.microsoft.com/en-us/library/aa286483.aspx>), [Java Server Pages \(JSP\)](http://www.oracle.com/technetwork/java/javaee/jsp/index.html) (<http://www.oracle.com/technetwork/java/javaee/jsp/index.html>) and [PHP](http://php.net/) (<http://php.net/>) are commonly used nowadays.

1.3.6.4 Activity 9



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

See what sorts of scripts are available for download from:

- http://cgi.resourceindex.com/Programs_and_Scripts/Perl/
- http://php.resourceindex.com/Complete_Scripts/

Do you find anything that might be useful for your own projects?

1.3.7 Web authoring tools



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

A website is made up of a combination of **various types of computer files** (Some are **HTML documents**. Others are **image files**. Yet others are **multimedia files like Flash animations, MP3 audio and Acrobat documents**.) Unfortunately, no single software package gives you the tools to do everything from designing your pages to creating images and multimedia to previewing and testing your site.

In this section, we'll look at a checklist of the tools that most Web designers will use throughout the Web development process. These tools can be classified as:

- code editors;
- image editors;
- Web browsers (for viewing and testing your sites).

1.3.7.1 Code editors



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

A code editor is software that enables you to build webpages. It includes tools for organizing your pages into a site and then publishing your site on the World Wide Web. It serves as the centre of production for your site, where you will add text, images and multimedia files to your page layouts, then connect these individual pages into an entire website. Code editors can be classified as text-based HTML editors and What You See Is What You Get (WYSIWYG, pronounced wizzywig) editors.

text-based HTML editors

Text-based editors require HTML knowledge in order to use them, so they appeal to more intermediate and expert users who want to be in full control of the HTML behind their pages. However, it doesn't mean that you'll be entering all the code yourself from scratch. These editors come with assorted buttons, drop-down menus and short-cut keys to save you from typing all the HTML tags yourself. Examples of text-based editors are UltraEdit (URL: <http://www.ultraedit.com>) and Bare Bones BBEEdit (URL: <http://www.barebones.com/products/bbedit>).

What You See Is What You Get (WYSIWYG, pronounced wizzywig) editors

Visually-oriented people like designers and graphic artists prefer WYSIWYG editors because they simply draw their webpage layouts on the screen, add the text, connect the links, and position the images and multimedia files, much like drawing on a piece of paper. The editor translates the design into HTML code behind the scenes. You also have the option to view and make changes to the HTML code directly as you gain more experience. An example of commercial WYSIWYG editors is Adobe Dreamweaver (URL: <http://www.adobe.com/products/dreamweaver.html>).

Most office suites nowadays can also generate HTML documents from a word processor or spreadsheet editor. For example, you can build your CV in Microsoft Word and save it as a webpage. The resulting HTML is often less than ideal in terms of readability and conformance to standards, but is an excellent way for non-technical beginners to start publishing on the Web.

In the next activity, you will see for yourself the wide range of code editors available on the market.

wide range

There are editors which are suitable for beginners, while some are targeted towards experienced users. Editors can also vary in price and features offered. Most of them offer free trial downloads which you can try for a limited period of time. Some of them may be open-source, which means they come for free although customer support might be limited to non-existent. You may want to download trial versions of various programs and decide for yourself which one works best for your needs.

1.3.7.1.1 Activity 10

Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

Visit CNet's software download site at: <http://www.download.com>.

Type 'web page editor' as the keyword in the Search box, and then press <Enter> to begin your search for trial software that can be downloaded.

You can further refine your software search by selecting an operating system, a licensing term (free or free to try), and a file size.

You might also like to visit the websites of Adobe (<http://www.adobe.com>) and Microsoft (<http://www.microsoft.com>) to check out their offerings directly. You could also use the keyword 'web page editor' or 'open source web page editor' on Google to see what may turn up.

1.3.7.2 Image editors

Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

Now, let me turn you over to the next reading for a discussion on image editors, multimedia editors and plug-ins.

Reading 3: 'How to choose Web software' (page 3 of 4) (<http://www.webreference.com/programming/web/3.html>).

Note: This reading only includes the section titled 'Choosing an image editor'.

You may want to note that there are a few open source image editors available. The most full-featured and supported open source application is the [GNU Image Manipulation Program \(GIMP\) \(http://www.gimp.org/\)](http://www.gimp.org/). This free software is a great alternative to the more expensive [Photoshop \(http://www.adobe.com/products/photoshop.html\)](http://www.adobe.com/products/photoshop.html) or even [Fireworks \(https://creative.adobe.com/products/fireworks\)](https://creative.adobe.com/products/fireworks).

1.3.7.3 Web browsers



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

Lastly, you will need to test the visual appearance and functionality of your websites in different browser software and browser versions. The most popular browsers in the market nowadays are [Internet Explorer](http://windows.microsoft.com/en-hk/internet-explorer/download-ie) (<http://windows.microsoft.com/en-hk/internet-explorer/download-ie>), [Mozilla Firefox](http://mozilla.com.tw/firefox/new/) (<http://mozilla.com.tw/firefox/new/>), [Safari](http://www.apple.com/safari/) (<http://www.apple.com/safari/>), and [Opera](http://www.opera.com/zh-tw) (<http://www.opera.com/zh-tw>).

Congratulations! You have now demonstrated the basic Web design skills that you will build on in the future.

1.4 Self-test



Available under [Creative Commons-ShareAlike 4.0 International License](http://creativecommons.org/licenses/by-sa/4.0/) (<http://creativecommons.org/licenses/by-sa/4.0/>).

Create an online résumé

You will now create an online résumé or CV to demonstrate your understanding of the basic technologies for building webpages. You may use HTML or XHTML to build the page, and you will be provided with an external CSS file that you can link to.

Create your online résumé using a text editor, such as Notepad.

The contents of this résumé are geared towards a Web designer position. The text and content can be downloaded here:

 [self_test_resume.zip](http://www.opentextbooks.org.hk/system/files/resource/1/1666/1694/media/self_test_resume.zip) (http://www.opentextbooks.org.hk/system/files/resource/1/1666/1694/media/self_test_resume.zip)

The download also includes a CSS file that can be used to format and lay out your résumé.

You should include the following elements in your webpage:

- titles (h1, h2)
- headings (h3, h4, h5, h6)
- bullet points (ul, ol, li)
- links to favourite Web design sites, together with appropriate labels for each link
- email link to yourself
- link to the top of the page labelled 'Back to Top'
- optional elements: photo, portfolio section containing links to design work, recommendations or awards, etc.
- link to the external stylesheet 'resume.css' in the <head> section of your document. The style sheet is included in the download above.

Congratulations! You have now demonstrated the basic Web design skills that you will build on in the future.

1.5 References



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

Below are the resources referred to or cited by the developer(s) of the original unit:

About.com: <http://www.about.com>

A List Apart: <http://www.alistapart.com>

Darlington, K (2005) *Effective Website Development: Tools and Techniques*, Addison Wesley.

Gundavaram, S (1996) *CGI Programming on the World Wide Web*, O'Reilly & Associates.

Nielsen, J (2006) 'Response time overview', <http://www.useit.com/papers/responsetime.html>

Schafer, S M (2005) *Web Standards: Programmer's Reference*, Wrox Press.

The Web Standards Project: <http://www.webstandards.org>

W3 Schools: <http://www.w3schools.com>

Web Reference: <http://www.webreference.com>

Web Style Guide: <http://www.webstyleguide.com>

Wikipedia: <http://www.wikipedia.org>

Willard, W (2001) *Web Design: A Beginner's Guide*, Osborne/McGraw Hill.

World Wide Web Consortium: <http://www.w3c.org>

1.6 Conclusion



Available under [Creative Commons-ShareAlike 4.0 International License \(http://creativecommons.org/licenses/by-sa/4.0/\)](http://creativecommons.org/licenses/by-sa/4.0/).

We went on a tour of the most common Web technologies, languages and development tools in this module.

Designers who primarily focus on the front-end must know HTML or XHTML for coding and marking up webpages, Cascading Style Sheets for formatting and layout, and JavaScript for implementing interactive features running on the browser. For those who want to branch out to developing dynamic webpages, a knowledge of how HTML integrates with server-side technologies such as CGI, PHP or PERL is required.

Code editors and image editors are a must-have for building and producing websites and graphics.

If you would like to learn more on this subject, you are welcome to enrol in *IT S234 Web Site Design* (http://www.ouhk.edu.hk/wcsprd/Satellite?pagename=OUHK/tcGenericPage2010&c=C_ETPU&cid=191154044800&lang=eng) offered by the School of Science and Technology (http://www.ouhk.edu.hk/wcsprd/Satellite?pagename=OUHK/tcSubWeb&l=C_ST&lid=191133000200&lang=eng) of the OUHK.