

## Chapter 2 — Adding Images, Media, Color, and Font Effects

A Web page is rather flay and uninteresting when there are no pictures, graphics, and there are no colors. This chapter shows how to properly include images, sound files, and video files into your page. We will also use inline style to add color to elements and change fonts on our pages.

### Learning Outcomes for this Chapter

Upon completion of the reading of this chapter and assignments found at the end, a student should be able to:

- insert an image on a web page with alternate text for accessibility,
- recall, describe, and choose images in various formats based upon the specific features of the image,
- add audio and video playback to a Web page,
- differentiate between common audio formats and include multiple formats in a single tag,
- differentiate between common video formats and include multiple formats in a single tag,
- add color to the page and the content of the page using inline style,
- explain the concept of RGB colors and use the color name, hexadecimal form, and `rgb()` function in CSS,
- recall the 5 standard font families found in all Web browsers,
- change the font family and size on an element or on an entire Web page.

### Adding Images to Your Page

The `<img>` tag allows images to be placed into an HTML document.<sup>23</sup> It is categorized as a Phrasing Element, meaning that it is placed inside paragraphs and other elements of the document. It is also an empty tag and does not require a closing tag. The `<img>` tag **requires** two attributes: the `src` (short for source) attribute and the `alt` (short for alternate name) attribute inside of the tag. A properly written `<img>` will look like this:

```

```

In this example, the `src`, or the source of the image, is directing the web browser to the location of the image. The image `clouds.jpg` rests inside the same folder as the HTML document.

```

```

23 <https://html.spec.whatwg.org/multipage/embedded-content.html#the-img-element>



In the second example the image file is inside the "media" folder that is in the same folder as the HTML document. If the image does not exist in the specified path, the browser will display the text in the **alt** attribute.

In order to prevent long and complicated **src** URLs when you are starting out, it is recommended that you place your images and media in the same folder as your HTML (For a more advanced discussion of URLs, see Chapter 4). Image file names are **case-sensitive** on some servers, so it is important to type in the files exactly how they are in your folder.

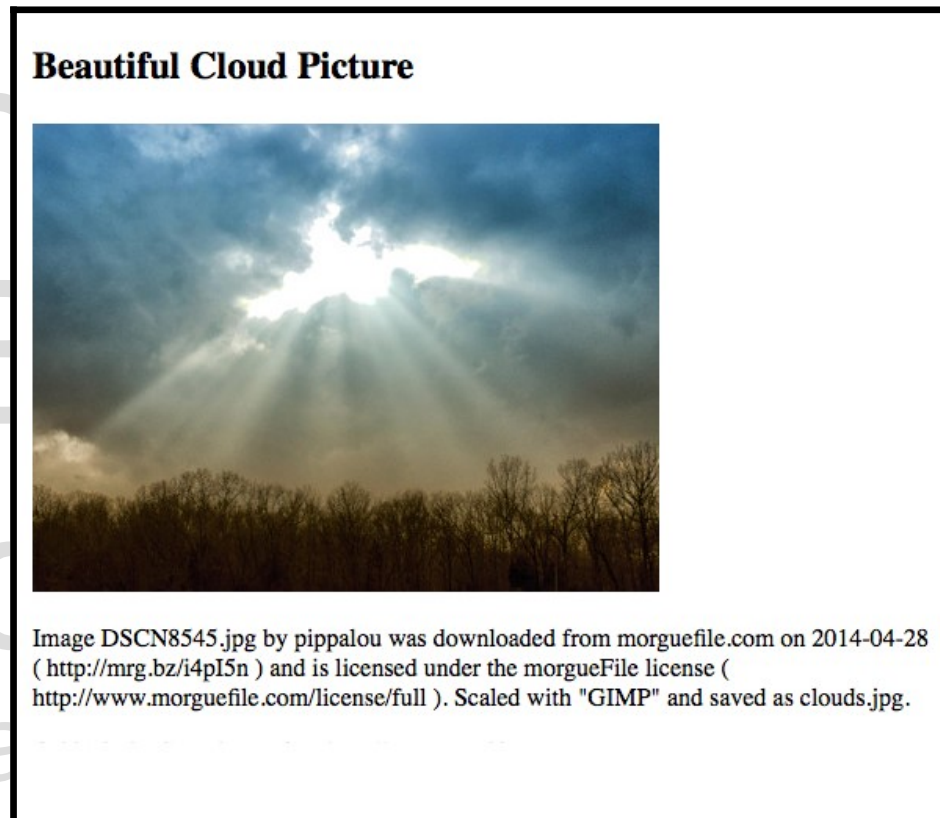
The **alt** attribute contains a brief text message that describes what the image is. It is displayed when the browser is unable to load the image or to describe the image to the visually impaired that are using a screen reader. The **alt** is **required** but if the image is not important to the content of the page the quotes may be left empty.

Here is an example of how your page will look with an `<img>` tag:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Chapter 2 - Image</title>
  </head>

  <body>
    <h2>Beautiful Cloud Picture</h2>
    
    <p>Image DSCN8545.jpg by pippalou was downloaded from morguefile.com
    on 2020-05-24 ( http://mrg.bz/i4pI5n ) and is licensed under the
    morgueFile license (http://www.morguefile.com/license/full ).
    Scaled with "GIMP" and saved as clouds.jpg.</p>
  </body>
</html>
```





*Illustration 4: Including an Image on Your Page*

Remember, if you do not have permission to use an image on your web site, you should not use it. Images are protected under International Laws called Copyright Laws. If you have permission it is always proper to give credit to the provider of your image. If you create the image (drawing or photograph) then you, as the author, may use the image on your Web pages.

## Image Types and Sizes

Though the extension of an image (.jpg, .jpeg, .gif, .png, etc.) may seem inconsequential, it is essential for images to be of the proper file type in order to ensure that a browser will display them properly. Here is a list of commonly supported file-types on most browsers for quick referencing:

- GIF — Has a wide range of colors (256 different colors per image from 16,777,216 (24-bit) possible colors) and has transparency. GIFs can also be created to show simple animations. GIFs are very commonly used for clip-art and logos. <sup>24</sup>

<sup>24</sup> [http://en.wikipedia.org/wiki/Graphics\\_Interchange\\_Format](http://en.wikipedia.org/wiki/Graphics_Interchange_Format)



- **JPG/JPEG** — The JPG format is used for photographic quality images. It features lossy compression, which means that files are made smaller by reducing image quality. For a large image this is typically not an issue, but if the image is scaled it may lead to a lower quality or pixelated appearance.<sup>25</sup>
- **PNG** — The PNG image type allows for more than 16 million colors and transparencies. It is compressed using a lossless method.<sup>26</sup>
- **SVG** — This file type uses vector-based graphics, meaning that it can be resized without losing any quality.<sup>27</sup>

There are other file-types, however support may be varied depending on the browser.

Large file sizes (especially with the JPG) result in better quality images, but cause long load times that can cause the user inconvenience. On the other hand, too low of a quality can result in a less than attractive webpage. There are other factors to consider when choosing a file type, such as transparency, so be sure to choose the right file type for your needs.

Later you will be shown how to scale an image with CSS to allow it to be shown smaller or larger than its size. It is recommended that you resize an image to the needed size before using it on your page. Shrinking a large image will require a longer load time and wasted data transmission, while enlarging an image will result in poor visual quality.

## Adding Sounds and Music to Your Page

The `<audio> . . . </audio>` tag allows sounds and music to be added to a webpage.<sup>28</sup> Unlike the `<img>` tag, `<audio> . . . </audio>` requires a closing tag. There are two general ways to add sound: 1) using the `src` attribute, or 2) using the `<source>` tag. Here is an example of the first.

```
<audio src="media/clang.mp3">Can't play the clang sound.</audio>
```

In the example above, `src` functions just as it does when inserting an image. The text between the `<audio> . . . </audio>` will appear to the user if the browser does not support the file type. This way of adding sound to a page does not allow for more than one file-type, limiting the number of users who will be able to hear your sound.

In order to achieve maximum browser compatibility, it is recommended that an `<audio> . . . </audio>` tag be written with multiple different formats of the same audio. It looks like:

25 <https://www.jpeg.org/index.html>

26 [http://en.wikipedia.org/wiki/Portable\\_Network\\_Graphics](http://en.wikipedia.org/wiki/Portable_Network_Graphics)

27 [http://en.wikipedia.org/wiki/Scalable\\_Vector\\_Graphics](http://en.wikipedia.org/wiki/Scalable_Vector_Graphics)

28 <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/audio>



```
<audio>
  <source src="angels.mp3" type="audio/mpeg">
  <source src="angels.ogg" type="audio/ogg">
  Your browser does not support the audio element.
</audio>
```

In this example, the `<audio>...</audio>` tag contains several `<source>` tags to specify the location and type of the media file. This allows for multiple formats of the same audio. The **type** attribute in this example communicates to the browser what format of audio it is reading, and must match the **src** file's format.<sup>29</sup>

## Audio Attributes

There are several additional attributes that can be contained within the `<audio>...</audio>` tag that specify when the audio is played and at what volume. `<audio autoplay>` tells the browser to begin playing the sound automatically, and `<audio controls>` will place a controller on the webpage that the user can interact with. The **controls** attribute is highly recommended as it allows the user to pause the sound, restart it, and raise and lower the volume.

## Audio Formats

There are many different file types for sounds and music, however different browsers will have different levels of support for each one. Here is a quick reference list of the three most common formats.<sup>30</sup>

- **mp3** — Stands for MPEG Layer III audio. A very common audio compression format
- **wav** — Stands for Windows uncompressed audio format. Common sound file format made on computers that run the Windows operating system
- **ogg** — Free and open source container file holding Vorbis format compressed audio

Keep in mind that some browsers and systems may play some but not all the formats of audio. In order to ensure the maximum support on all systems, you may consider using an audio transcoding application to convert your file into as many formats as possible. There are many programs available to do this, like:

- Audacity<sup>31</sup> — Open source audio recorder, editor, and transcoder

29 <http://www.iana.org/assignments/media-types/media-types.xhtml>

30 [http://en.wikipedia.org/wiki/Audio\\_file\\_format](http://en.wikipedia.org/wiki/Audio_file_format)

31 <https://sourceforge.net/projects/audacity/>



- `fre:ac`<sup>32</sup> — An open source audio ripper encoder that was previously known as BonkEnc
- `MediaCoder`<sup>33</sup> — For computers using the Windows operating system

Here is an example page with an image and audio on it:

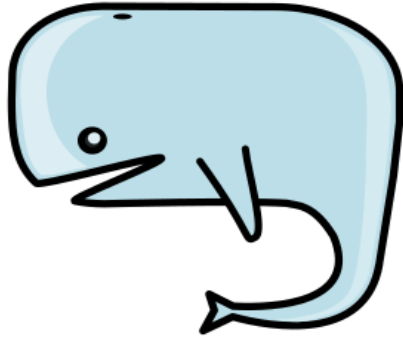
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Chapter 2 - Audio</title>
  </head>
  <body>
    <h2>A Brief Reading from Moby Dick by Herman Melville</h2>
    
    <p>Call me Ishmael. Some years ago- never mind how long precisely-
    having little or no money in my purse, and nothing particular to
    interest me on shore, I thought I would sail about a little and
    see the watery part of the world.</p>
    <audio controls>
      <source src="media/mobydick_callme.mp3" type="audio/mpeg">
      <source src="media/mobydick_callme.ogg" type="audio/ogg">
      Your browser does not support the audio element.
    </audio>
    <p>Text from
    https://americanliterature.com/author/herman-melville/book/moby-dick-or-the-
    whale/chapter-1-loomings and in the public domain. Audio clip taken from
    https://librivox.org/moby-dick-by-herman-melville and in the public domain.
    Image of a whale was taken from https://openclipart.org/detail/120/whale and
    in the public domain.</p>
  </body>
</html>
```

32 <https://www.freac.org/>


33 <http://www.mediacoderhq.com/>



**A Brief Reading from Moby Dick by Herman Melville**



Call me Ishmael. Some years ago- never mind how long precisely- having little or no money in my purse, and nothing particular to interest me on shore, I thought I would sail about a little and see the watery part of the world.



Text from <https://americanliterature.com/author/herman-melville/book/moby-dick-or-the-whale/chapter-1-loomings> and in the public domain. Audio clip taken from <https://librivox.org/moby-dick-by-herman-melville> and in the public domain. Image of a whale was taken from <https://openclipart.org/detail/120/whale> and in the public domain.

*Illustration 5: Page with Audio and Text*

## Adding Videos to Your Page

The `<video>...</video>` tag functions very similar to the `<audio>...</audio>` tag.<sup>34</sup> The attributes `autoplay` and `controls` function in the same fashion. File formats are extremely important when you are inserting a video. There are two ways to write a `<video>...</video>` tag. The first is appropriate when the video has only one format, and is written like this:

```
<video src="videofile.mp4">Your browser does not support the video element</video>
```

For maximum compatibility you should use the multi-format `<video>...</video>` tag:

<sup>34</sup> <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/video>



```
<video width="640" height="496" controls>
  <source src="media/heart.ogv" type="video/ogg">
  <source src="media/heart.mp4" type="video/mp4">
  Your browser does not support the video element.
</video>
```

It is recommended that you include both the .mp4 and .ogg file formats in your HTML.<sup>35</sup> To achieve this you may need to use a program to transcode your video files. Here are some recommended programs:

- handbrake<sup>36</sup> — An open source program that can convert nearly any file format
- ffmpeg<sup>37</sup> — A free software project that can convert video file formats and much more

Here is an example page with a short video clip:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Chapter 2 - Video</title>
  </head>
  <body>
    <h2>Raising of the flag on Iwo Jima (1945)</h2>
    <video width="320" height="240" controls>
      <source src="media/CEP413.mp4" type="video/mp4">
      <source src="media/CEP413.ogv" type="video/ogg">
      Your browser does not support the video element.
    </video>
    <p>Video by the U.S. Government Office of War Information.</p>
    <p>Video from https://archive.org/details/CEP413 and in the public
domain.</p>
  </body>
</html>
```

35 [http://www.w3schools.com/html/html5\\_video.asp](http://www.w3schools.com/html/html5_video.asp)

36 <https://handbrake.fr/>

37 <http://www.ffmpeg.org>







*Illustration 6: Adding Video to Your Page*

## Adding Inline Style to Your Page

CSS, or Cascading Style Sheets, allow you to add color, fonts, and stylistic formatting to your page and site.<sup>38</sup> In general there are three ways to add style to a webpage: 1) inline style which is applied within a tag; 2) internal style which is applied directly to the HTML document using the `<style> . . . </style>` tag; and 3) external style which is imported from a separate CSS document.

External CSS is the proper way to style a professional webpage, but since we are just starting out it is useful to begin with the more simplified inline style. We will discuss internal and external style in depth in Chapter 5.

In order to add inline style to an HTML document, you will need to use the **style** attribute. The **style** attribute is written within a tag in the format:

```
<sometag style="property:value;">
```

<sup>38</sup> <http://www.w3.org/Style/CSS/Overview.en.html>



So, if you wanted to change the color of, say, an `<h1>...</h1>` tag, you would write it as `<h1 style="color:purple;">...</h1>`. As with any HTML code, it is important to make sure that it is written correctly with proper casing and quotations.

The most convenient way to add style to an entire page is to specify the `style` attribute within the body of the HTML document. For example, you may use the `style` attribute to make all of the text red. However, what if you want some text to be blue? CSS can be overwritten by taking advantage of the "child" and "parent" inheritance rules. The C in CSS stands for "cascading", meaning that it follows a hierarchical scale of importance. The farther down a line of CSS code is within a "parent" tag, the more precedence it takes. For example, observe this block of HTML:

```
<body style="color:red;">
  <p>This text is red</p>
  <p style="color:blue;">This text is blue</p>
</body>
```

In this example, the `<body style="color:red;">...</body>` is specifying that all text on the webpage will be red. However, because the "child" tag `<p style="color:blue;">...</p>` is inside the parent tag `</body>`, its style takes precedence, and will thus display that paragraph with blue text.

## Colors

There are various ways to set colors for a webpage. We have already seen colors being set by name, which is the most convenient way to add colors. There are 140 color names that can be set and viewed on an HTML page, and writing them is as easy as adding the `style="color:colorname;"` attribute to an HTML tag.<sup>39</sup> While this is an easy way to add colors, it limits you to using only the colors that are provided by the standard.

Another, more flexible way to add color to your webpage is to add a hexadecimal number rather than a color name. A hexadecimal number is written in the `#RRGGBB` format which represents three two digit base 16 numbers that describe the amount of red, green, and blue in the color. For example, `#FF0000` is the color red. There are 16,777,216 possible colors that can be written in this format provided you know the code, making this method much more versatile than using color names.

The third method of adding color is a bit easier to read than the hexadecimal number format. The `rgb()` style function works similarly to the hexadecimal code, only the red, green, and blue are

<sup>39</sup> <https://www.w3.org/wiki/CSS/Properties/color/keywords>



written as numbers that range from 0 to 255, with 0 being none.<sup>40</sup> For example, the color red would be written as `rgb(255, 0, 0)`, meaning it has the maximum amount of red and no green or blue. All 16,777,216 color combinations are possible using this method.

Here is an example of all three methods being used in an HTML document. In this example the background is being specified with the `background-color` style.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Chapter 2 - Colors</title>
  </head>
  <body style="background-color: #ffff00;">

    <h1 style="background-color: blue; color: white;">My Colorful
    Page</h1>
    <p style="color:rgb(255,100,50);">Opanesat tocusan dis re retapih
    vanohir! Der leminoy delisot ner ibopi menid goneg fo latato. Col
    tagaci oge loroqe idat secobe no. Necuti bacena naraset ceg
    ilebarop fecocief.</p>

  </body>
</html>
```



Illustration 7: My Colorful Page

Notice in the line `<h1 style="background-color: blue; color:`

<sup>40</sup> [http://www.w3schools.com/cssref/css\\_colors.asp](http://www.w3schools.com/cssref/css_colors.asp)



`white;">...</h1>` that multiple styles are being specified within one tag. Multiple styles in an attribute are separated by semi-colons ";". The last style in the list may or may not have a semi-colon but web developers often put it there.

## Fonts

Adding different fonts to your webpage works in much the same way as adding color. The **font-family** style is used to specify the name of the font or fonts. If multiple fonts are defined, the first one that is available is used. You should always list more than one font, as not all computers will have the same fonts loaded on them. If the user does not have a font that you have specified, the browser will follow the list that you provide in your CSS until it reaches one that it can load. In the event that it cannot find any of them, it will display the default font provided by the browser.

Fonts typically fall into five generic categories. All browsers will have predefined fonts for these five types:<sup>41</sup>

- serif — a typeface such as Times New Roman that has small lines attached to strokes. Sometimes they are difficult to read when small sizes are displayed.
- sans-serif — a typeface such as Arial that does not have serifs. Generally considered to look "clean" and modern.
- cursive — a typeface resembling script.
- fantasy — a typeface that is generally used for headers and for text that stands out. Smaller sizes are not recommended with these fonts.
- monospace — a typeface where each letter takes up the exact same amount of space

You should always specify one of the five generic font styles at the end of your `font-family`. This is to ensure that the browser will honor your font request, even if it does not have the specific font you asked for.

If a non-generic font has spaces in its name, you will need to put it in quotes inside the style. Be careful with quotes, if the style is inside double quotes you should use single quotes around the font name. It may look like this:

```
<body style="font-family: 'DejaVu Sans', fantasy">
```

Font names are case-insensitive.<sup>42</sup>

41 <https://www.w3.org/TR/css-fonts-3/#generic-font-families>

42 <https://www.w3.org/TR/css-fonts-3/#font-family-casing>



## Font Size

The `font-size` style is used to specify the height of the font that is displayed on a webpage.<sup>43</sup> There are several ways to specify the size of the font, but each method is written in the same `font-size: _____; format.`

A `font-size` can be specified directly by name, with the different possibilities being `xx-small`, `x-small`, `small`, `medium`, `large`, `x-large`, and `xx-large`. This is a quick and simple way to change the size of the font, but it sacrifices accuracy.

Font sizes can also be specified by "points", which are units that are 1/72 of an inch tall and have their origin back to old printing presses.<sup>44</sup> Your computer will try to guess how big to make this on the screen, making it inaccurate. This method is written as `font-size: 4pt;`. Similar to points are pixels (written as `font-size: 4px;`), and define the height by numbers of pixels. Both points and pixels are absolute measurements, and will not scale to fit a browser.

An `em` unit represents a unit of measure relative to the font size of the parent, or the tag that it is nested within. For example, `font-size: 2em;` will make the text 200% of the size of the parent text. The "%" unit of measurement is similar, but works based on the current size of the text. For example, if the text is currently as 14pt, adding `font-size: 200%;` will increase the font to 28pt. Both `em` and `%` can be scaled in size to the user's preference, making the font more flexible and cooperative with cascading.<sup>45</sup>

Here is an example of the `font-family` and `font-size` attributes at work:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Chapter 2 - Font</title>
  </head>
  <body style="font-family: Arial, Helvetica, sans-serif;">

    <h1>Font from Body</h1>
    <h2 style="font-family: 'Book Antiqua', Palatino, serif; font-size:
400%;">Serif override</h2>
    <p style="color:rgb(255,100,50);">Opasesat tocusan dis re retapih
vanohir! Der leminoy delisot ner ibopi menid goneg fo latato. Col
tagaci oge lороpe idat secobe no. Necuti bacena naraset ceg
ilebarop fecocief.</p>
```

43 <https://developer.mozilla.org/en-US/docs/Web/CSS/font-size>

44 [https://en.wikipedia.org/wiki/Point\\_\(typography\)](https://en.wikipedia.org/wiki/Point_(typography))

45 <https://www.w3.org/TR/css-fonts-3/#font-size-prop>



```
</body>  
</html>
```

## Font from Body

# Serif override

Opanesat tocusan dis re retapih vanohir! Der leminoy delisot ner ibopi  
menid goneg fo latato. Col tagaci oge lороpe idat secobe no. Necuti  
bacena naraset ceg ilebarop fecocief.

*Illustration 8: Changing Fonts*

Be sure to note in the above example how multiple style attributes can be contained in one tag, and how CSS hierarchy can be used to specify fonts in child tags.

## **Advanced Topics for Exploration**

There are ways to display the exact font that you want on your page without requiring the user to already own the font on their computer. Web Fonts can be used to download fonts you want, and are placed in external style sheets with the `@font-face` rule.<sup>46 47</sup>

## **Vocabulary**

- Uniform Resource Locator (URL)
- alt
- audio
- background-color
- color
- cursive
- encoding
- fantasy
- font
- font-family
- font-size
- generic font

46 [http://www.w3schools.com/css/css3\\_fonts.asp](http://www.w3schools.com/css/css3_fonts.asp)

47 <https://developer.mozilla.org/en-US/docs/Web/CSS/@font-face>



- gif
- img
- jpg/jpeg
- monospace
- phrasing element
- png
- point
- sans-serif
- serif
- source
- src
- style
- transcoding
- video

Free  
eBook  
Edition

Please support this work at  
<http://syw2l.org>

Free  
eBook  
Edition



## Exercises

### Word Search

```
b r g a n i r e c g i f - i o o g y
l a p o i n t s c o s o e v r L a o
p t c c c r r i t o g i - d i p g a
n f r k c n r f g y l n i r n d g o
g U o a g u n r o e l o e o c t e v
l n R n n r r e n n n e r y j i l o
m e t L t s o s c g t e s r c p g i
u o t - e i c u i c s - r s e a e t
c n n o o n f o n v g o f i e c l g
g L g o c f c a d d e c u a c r l t
a r j s s o i o n i - i t r m a i j
u a p u s p o - d t n c o e c i s f
d e g u o o a d l i a g o a a e l c
i o e a t i c c y L n s r l f t i y
o y e a s t m a e d g g y c o n y c
c n a i o c s g g s r l o g i r o z
r a e a s a n s - s e r i f n g a s
n i g r f o n t - s i z e j p r r v
```

URL, alt, audio, background-color, color, cursive, encoding, fantasy, font, font-family, font-size, generic, gif, img, jpeg, jpg, monospace, png, point, sans-serif, serif, source, src, style, transcoding, video

### Problems

1. Create a page about a hobby that you or a friend has. Include a level one heading, a paragraph describing the hobby, and a paragraph with your sources. Go to a site like <http://www.openclipart.org> and find an image to include on your page. Change the font on the 'h1' to customize the page. Validate your HTML using the validation service at <http://validator.w3.org/>.
2. Find a piece of music or sound effect on-line or create your own. Have a graphic on the page, a header, and a brief description of the music/sound. Change the header font and color. Change the page background color. Validate your HTML using the validation service at <http://validator.w3.org/>. Sites like <http://www.archive.org> and





<http://www.openclipart.org> are good resources.

3. Find a video on-line or create your own. Have a graphic on the page, a header, and a brief description of the video. Change the header font and color. Change the page background color. Validate your HTML using the validation service at <http://validator.w3.org/>. Sites like <http://www.archive.org> and <http://www.openclipart.org> are good resources.

Free  
eBook  
Edition

Please support this work at  
<http://syw2l.org>

Free  
eBook  
Edition

