

Web Media

Chapter 7

Objectives

1. Digital Representation
2. Color Models
3. Image Concepts
4. File Formats
5. Audio and Video
6. HTML Canvas

Digital Representation of Images

Section 1 of 6

Digital Representation

Of Images

When you see text and images *on your desktop* monitor or your mobile screen, you are seeing many small squares of colored light called **pixels** that are arranged in a two-dimensional grid.

These same images and text *on the printed page* are not created from pixels, but from small overlapping dots usually called **halftones**

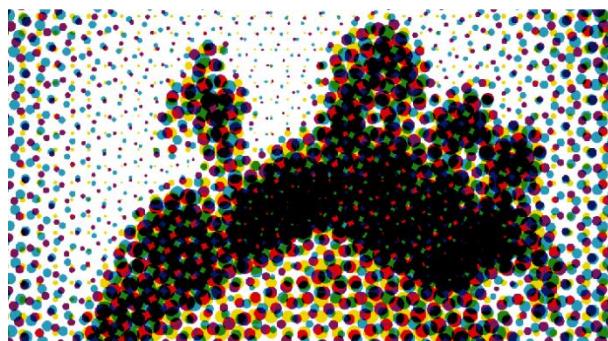
Digital Representation Of Images



Original photographic image



Output as pixels
(size exaggerated)



Output as halftones
(size exaggerated)

Digital Representation

Of Images

Digital representation ultimately refers to representing information as numbers

- text characters are digitally represented using standardized 8-bit (ASCII) or 16-bit (UNICODE) numbers
- Finite number of text characters, infinite pictures...

Digital Representation

Of Images

Two basic categories of image

1. Raster (a.k.a. bitmap)

the image is broken down into a two-dimensional grid of colored squares

2. Vector

composed of objects such as lines, circles, Bezier curves, and polygons

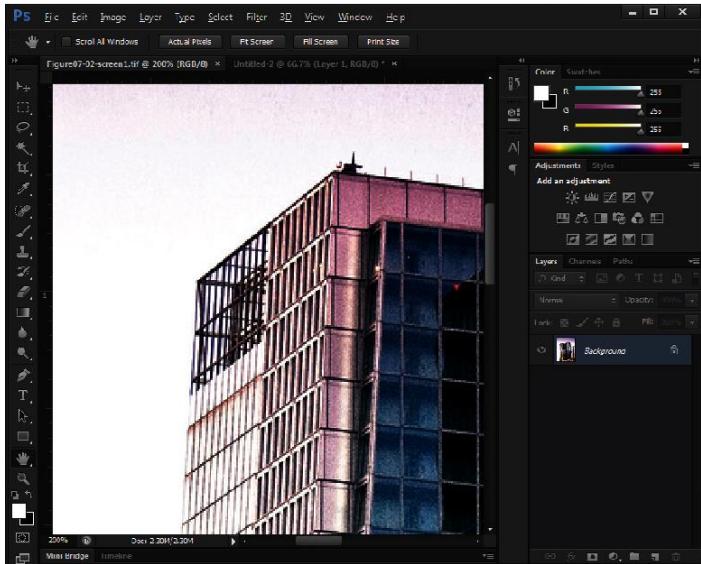
Raster Image

A close up

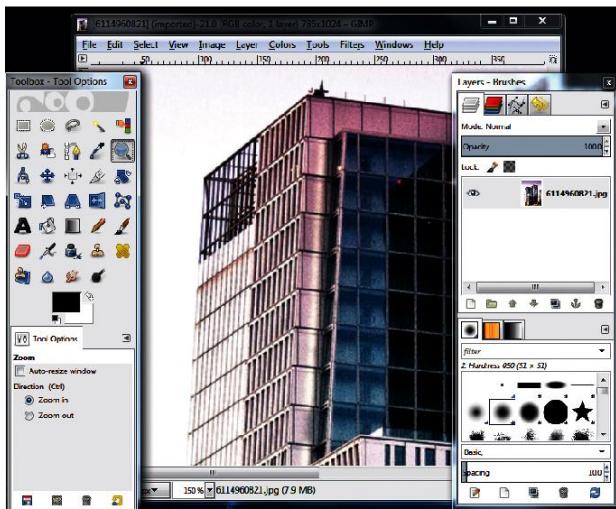


Raster Image

Raster Image Editors



Adobe Photoshop

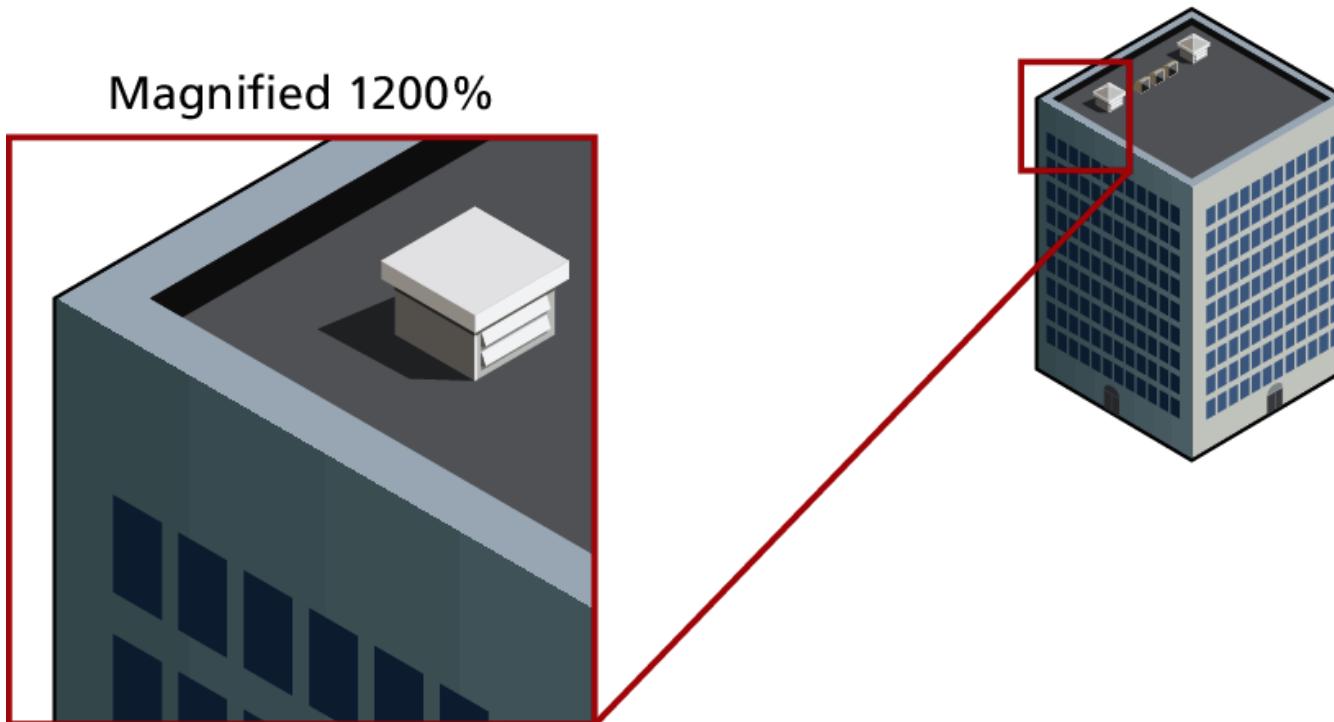


GIMP

Vector Images

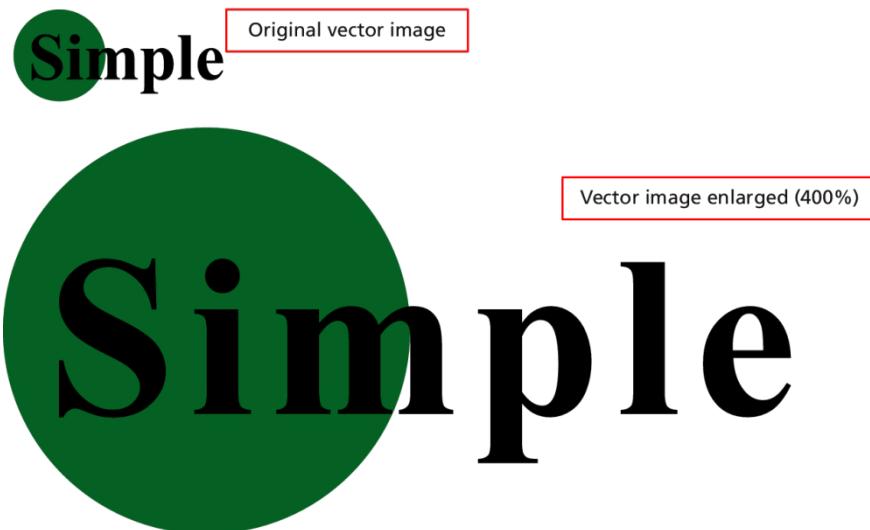
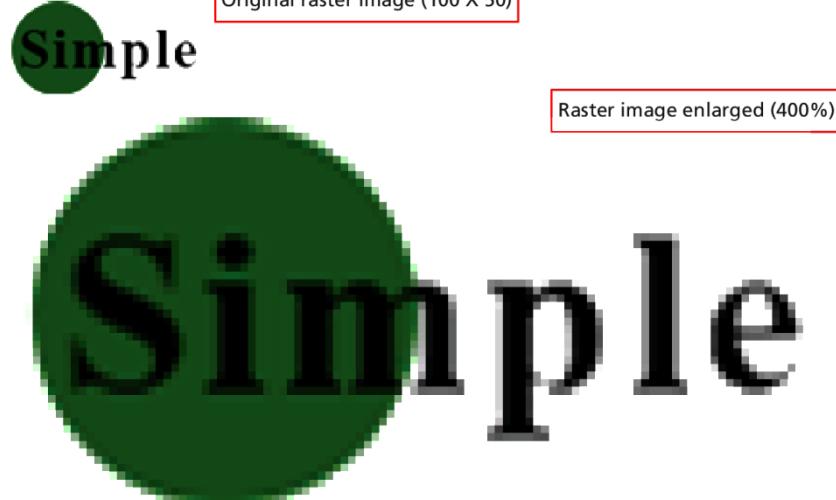
An Example

The main advantage of vector images is that they can be shrunken or enlarged without a loss of quality



Vector vs Raster

Examples



Color models

Section [2](#) of 6

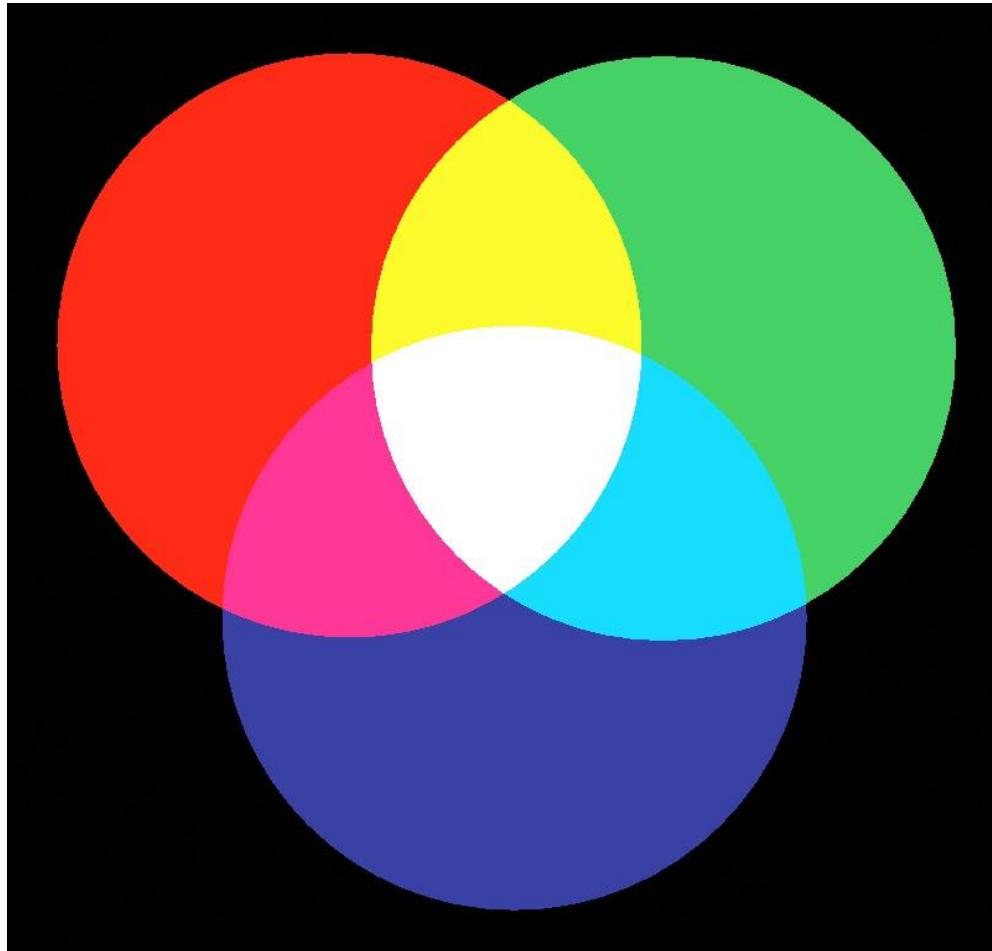
Multiple Models

There are a variety of different ways to specify color on the web.

- Named Colours
- **RGB** (Red Green and Blue)
- **CMYK** (Cyan-Magenta- Yellow-Key (or black))
- **HSL/HSB** (Hue, Saturation, Lightness/Brightness)

RGB

That shirt is a nice #33CA8F color.



RGB

How to represent

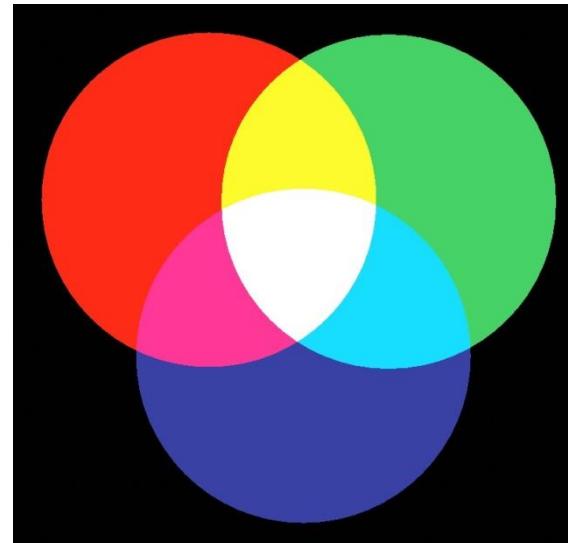
- 6 Digit Hexadecimal notation
 - #RRGGBB, where each digit 0-F hex
 - Example **#FF0000**
- CSS rgb function
 - Rgb(red, green, blue), values 0-255
 - Example **rgb(255, 0, 0);**

RGB

A Little color theory

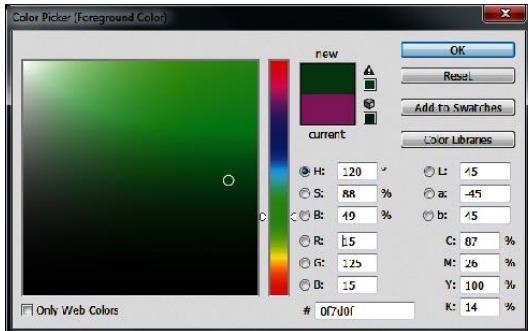
Because the RGB colors combine to create white, they are also called **additive colors**.

- The absence of colored light is black
- Adding all colors together creates white.

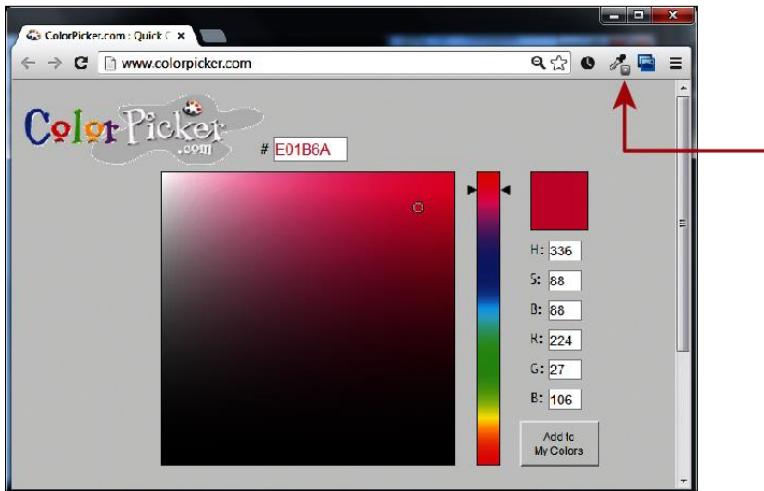


RGB

How to choose



Color Picker (Photoshop)



ColorZilla
(Chrome Extension)

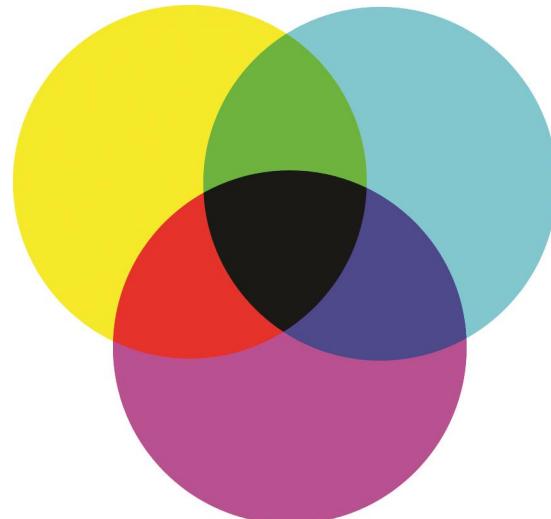
<http://www.colorpicker.com/>

CMYK

not every image will be displayed on an RGB device

In traditional color printing, color is created through overlapping cyan, magenta, yellow, and black dots that from a distance create the illusion of the combined color. For this a different color mode is necessary, namely

- **CMYK color model** for Cyan-Magenta Yellow-Key (or black).



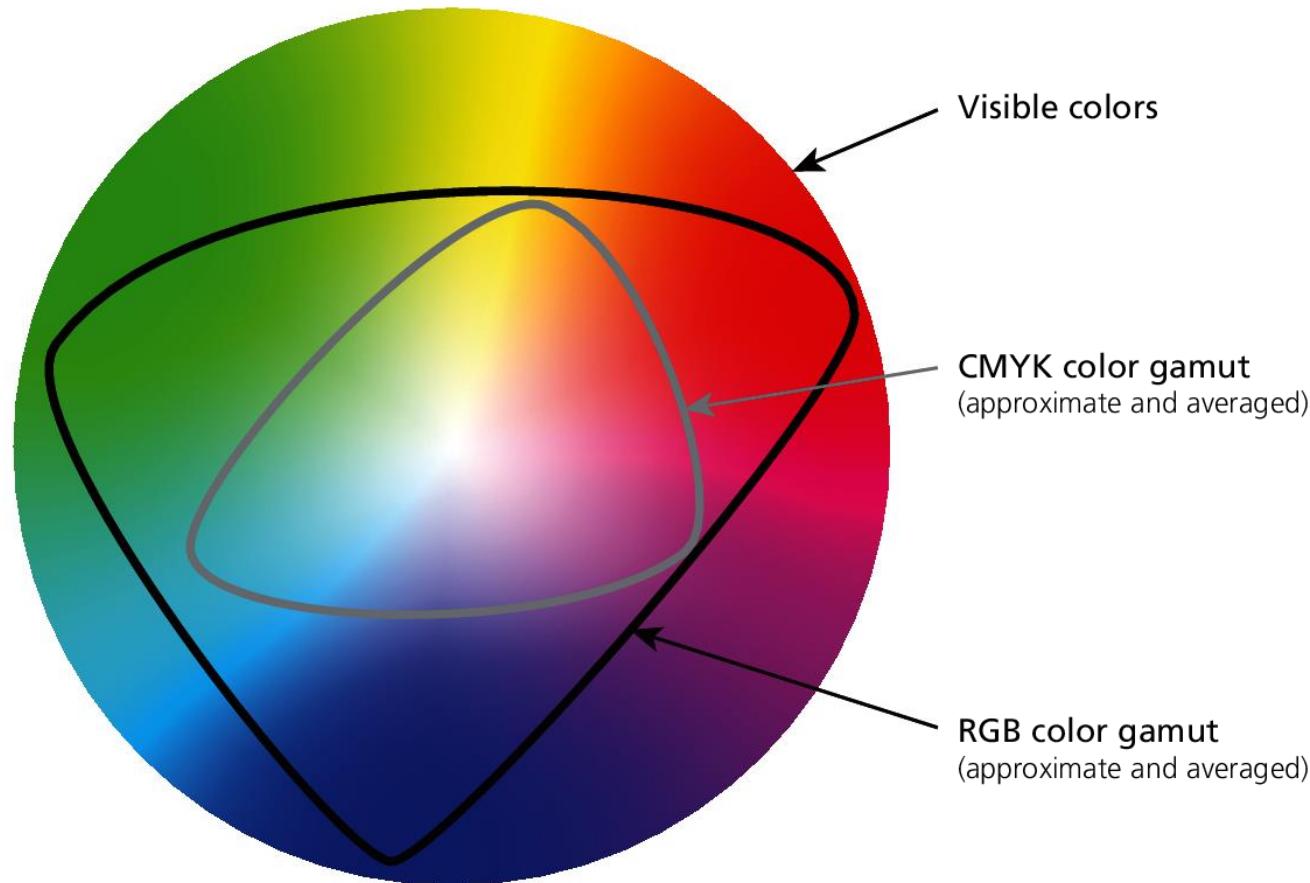
CMYK

For printers



Gamut

The range of colors that a color system can display or print



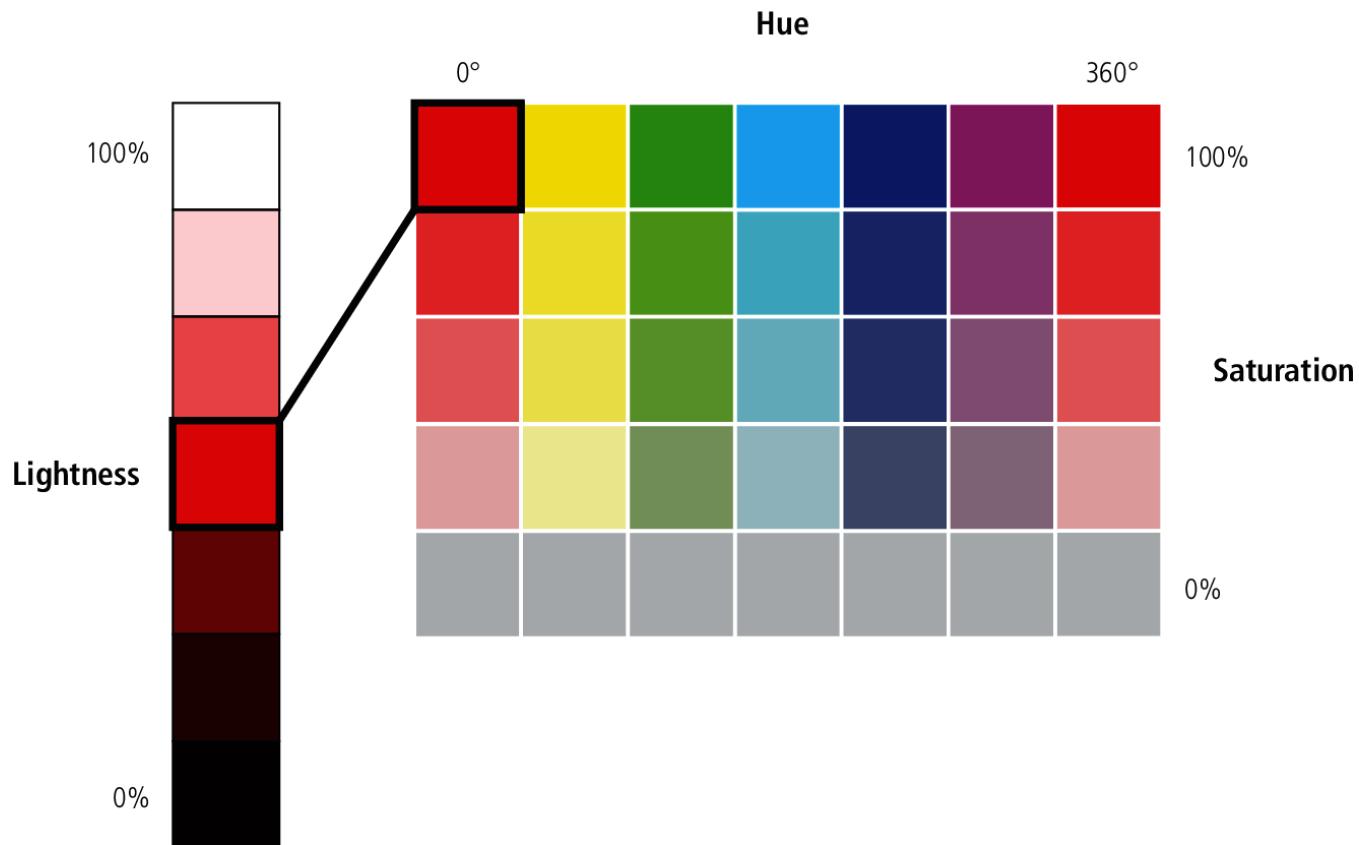
HSL/HSB

Hue, Saturation, Brightness(HSB) or Lightness(HSL)

is more closely aligned to the way we generally talk about color. It breaks a color down into three components:

- **hue** (what we generally refer to as color)
- **saturation** (the intensity or strength of a color)
- **Lightness/brightness** (that is, the relative lightness or darkness of a color)

HSL/HSB



HSL

CSS3 Support

CSS3 has support for the HSL model using the notation:

hsl(hhh, ss%, bb%)

- **hue** is an angle between 0 and 360 (think of hue as a circle)
- **saturation** is a percentage between 0 and 100, where 0% is completely desaturated (grey) while 100% is fully saturated.
- **Brightness/lightness** is a percentage between 0 and 100, with 0 percent being pure dark (black), and 100 percent being pure bright (white)

Opacity

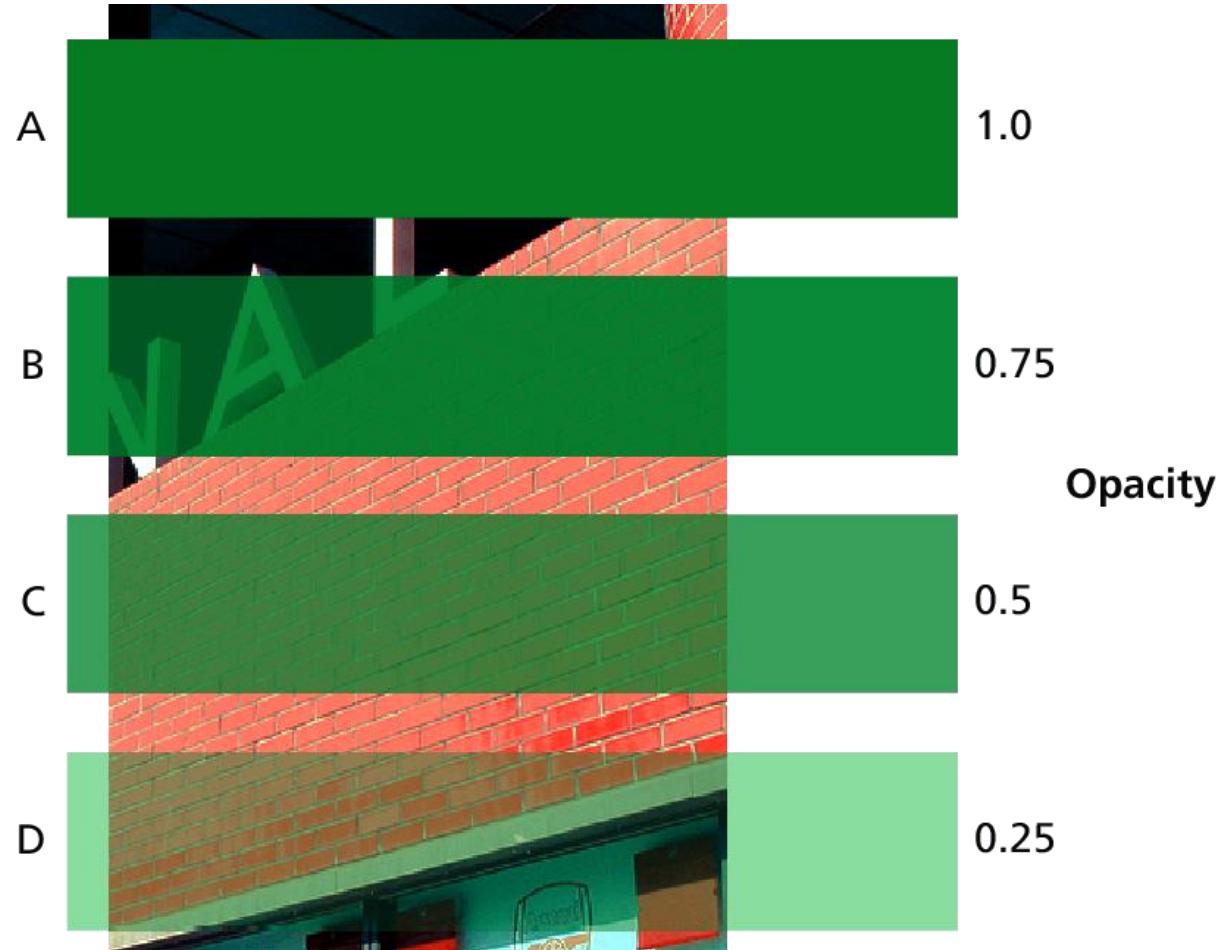
I can see right through you

Opacity is the degree of transparency in the color. This value is also referred to as **alpha transparency**.

- Opacity is typically a percentage value between 0 and 100 (or between 0 and 1.0)
 - 0 means no opacity (transparent)
 - 100 means that the element is fully opaque (no transparency).
- You can also add opacity values to a color specification using the `rgba()` or `hsla()`

Opacity

Transparency



Opacity

In CSS

```
.rectangleA {  
    background-color: rgb(0, 255, 0);  
}
```

A diagram illustrating the components of an RGB color value. The code shows 'background-color: **rgb(0, 255, 0)**'. Three vertical lines descend from the closing parenthesis of 'rgb' to three labels: 'red' at the top, 'blue' in the middle, and 'green' at the bottom.

```
.rectangleB {  
    background-color: green;  
    opacity: 0.75;  
}
```

```
.rectangleC {  
    background-color: rgba(0, 255, 0, 0.50);  
}
```

A diagram illustrating the components of an RGBA color value. The code shows 'background-color: **rgba(0, 255, 0, 0.50)**'. Four vertical lines descend from the closing parenthesis of 'rgba': 'opacity' at the top, followed by 'background-color' (which is shared with the previous diagram), 'hue' (the first parameter of 'hsla'), and 'luminosity' (the second parameter of 'hsla').

```
.rectangleD {  
    background-color: hsla(120, 100%, 50%, 0.25);  
}
```

A diagram illustrating the components of an HSLA color value. The code shows 'background-color: **hsla(120, 100%, 50%, 0.25)**'. Six vertical lines descend from the closing parenthesis of 'hsla': 'hue' (the first parameter), 'saturation' (the second parameter), 'luminosity' (the third parameter), and 'opacity' (the fourth parameter).

Color Relationships

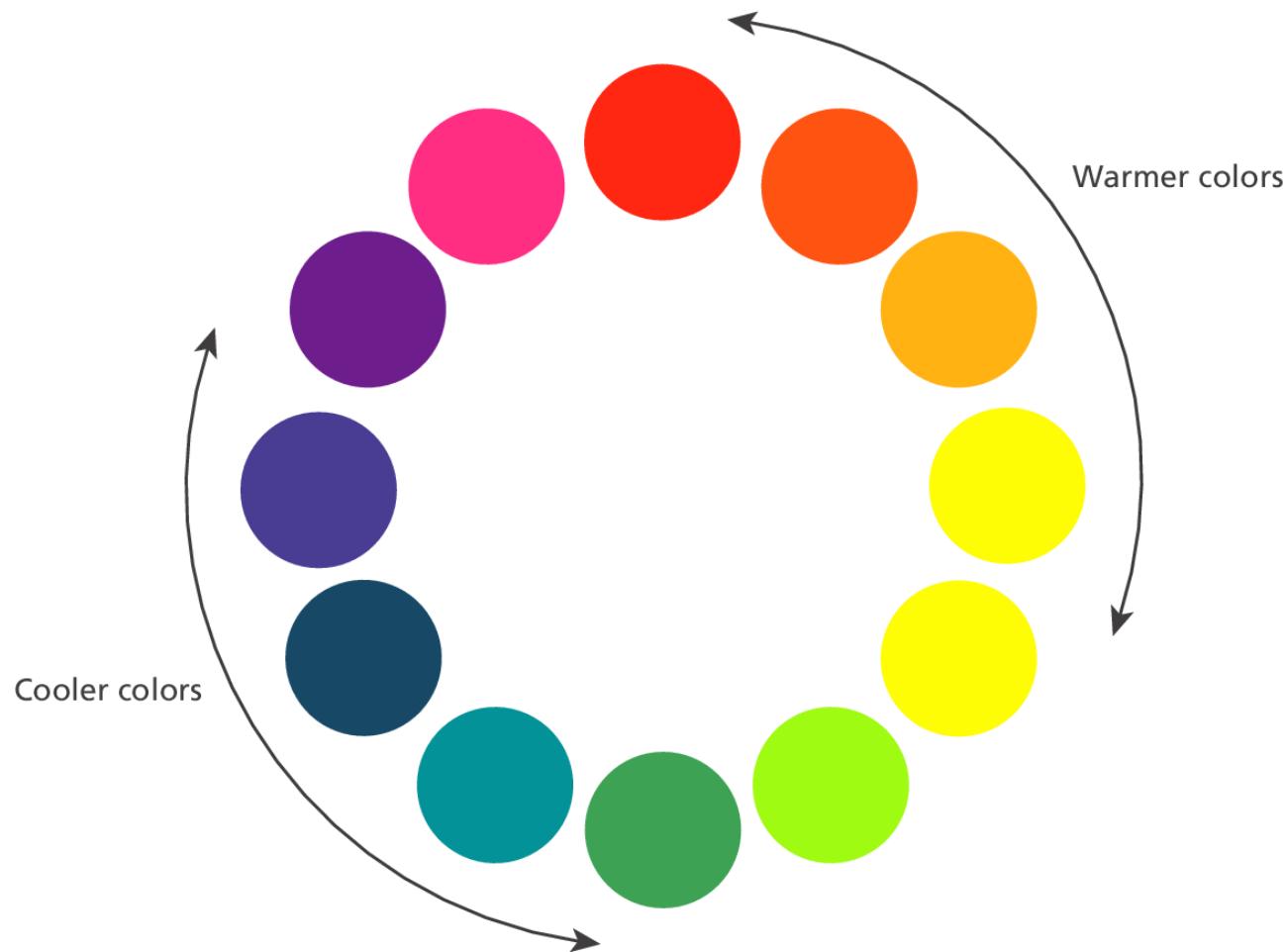
But that is a nice #33CA8F shirt

Humans do not dispassionately register a color's hue, saturation, and brightness. Instead we see colors in relationship to other colors.

- the way we perceive a color changes based on the other colors that are in close proximity.
- colors can evoke certain emotions and impressions, many of which are culturally determined
- Artists often use the color wheel, and more sophisticated mechanisms, to help understand and work with color

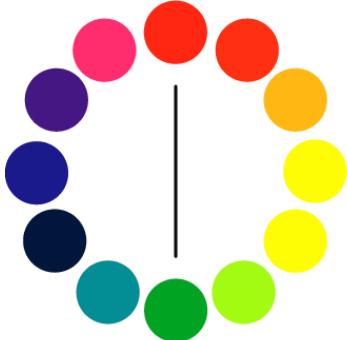
Color Relationships

Color Wheel



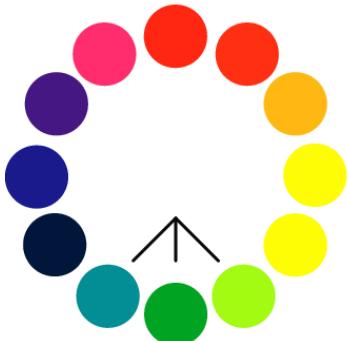
Color Relationships

Ways of picking color



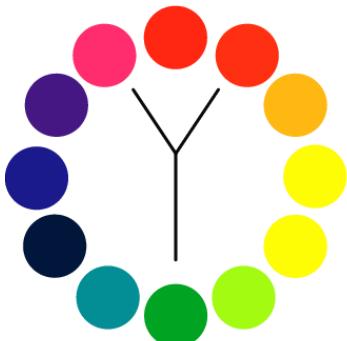
Complementary

These are color pairs that are on opposite ends of the color wheel. Complementary colors are highly contrasting and are believed to create a vibrant look. This scheme looks best when you place a warm color against a cool color.



Analogous

These are colors that are adjacent to one another on the color wheel. Since they lack contrast, they match well and create serene and harmonious designs. One color can be used as a dominant color while others are used to enrich the scheme.

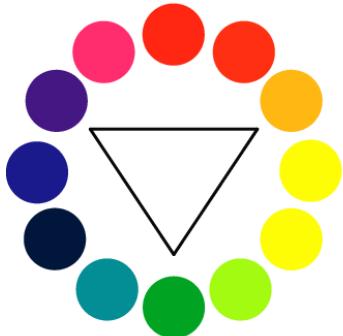


Split Complementary

It uses a primary color and the two colors on each side of its complementary color. This provides contrast but without the strong tension of the complementary scheme as well as providing some of the harmonies of an analogous scheme.

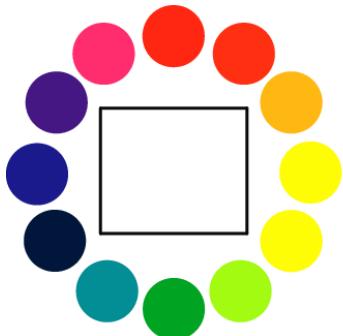
Color Relationships

A few more



Triad

Uses three colors on the color wheel in an equilateral triangle. Tends to be quite vibrant, gives a strong visual contrast but still retains a harmony among the colors. Works best if one color is dominant and the two others are used as accent colors.



Tetradic (Rectangular)

Also called a double complement, since it combines two sets of complementary colors. This rich scheme can be hard to harmonize if all four hues are used in equal amounts, so only one or two of the four colors should be dominant.

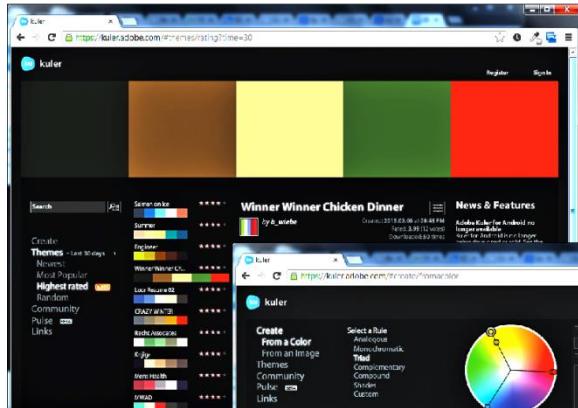
Color Relationships

A few online tools



<http://www.colorschemedesigner.com>

Allows you to construct themes based on different color relationships, and then see previews of sample websites with the colors in the scheme.



<http://kuler.adobe.com>

Also allows you to construct themes based on different color relationships. Also lets you browse and use color schemes put together and voted on by the Kuler community.



Color Relationships

Some Conclusions

The point here is that the colors you use in a website should not be chosen at random, but should work together in some manner.

.

- Programmers are not always the best judges of good color combinations.
- Sometimes you will have a visual designer who will handle these decisions.
- But for smaller projects, you may need to make those decisions yourself. Use the Tools!!!

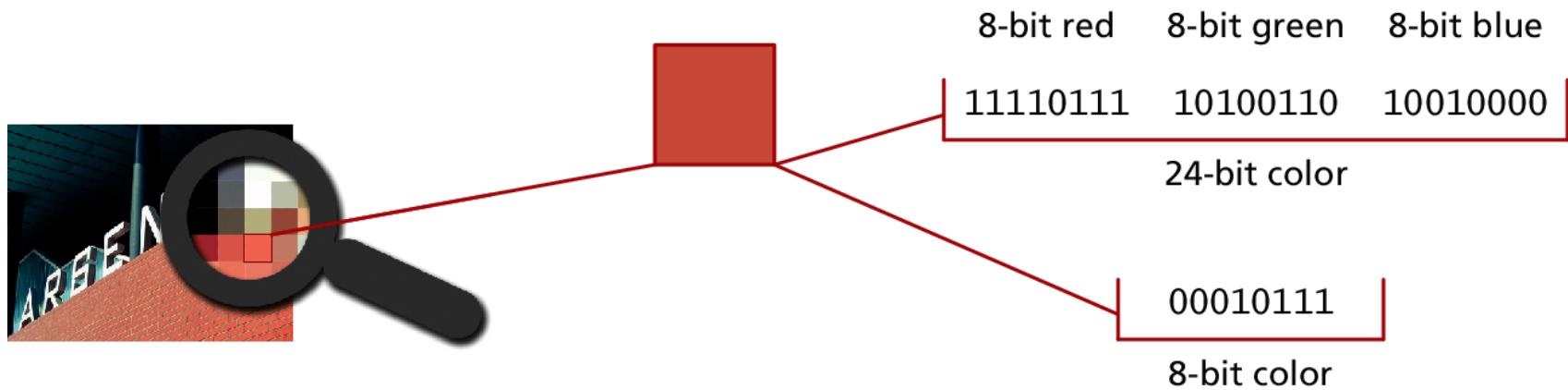
Image Concepts

Section 3 of 6

Color Depth

Color depth refers to the maximum number of possible colors that an image can contain.

For raster images, this value is determined by the number of bits used to represent the color or tone information for each pixel in the image



Color Depth

# Bits / Pixel	Description
8 bits or less	Sometimes referred to as indexed color. No more than 2^8 or 256 colors can be represented. Using 7 bits per pixel would allow only 128 colors, 6 bits per pixel would allow only 64 colors, 5 bits = 32 colors, 4 bits=16 colors, 3 bits = 8 colors, 2 bits = 4 colors, and 1 bit = 2 colors.
24 bits	Also called true color. 16.8 million colors can be represented. Eight bits each are used for red, green, and blue information.
32 bits	Same as 24 bit, but 8 bits of alpha transparency information is added.
48 bits	16 bits per red, green, and blue. While not supported in browsers, these deep color image depths are supported by specialized photo editing software.

Color Depth

Your output device has limits too

Image color depth is not the same thing as device color depth, which refers to the number of simultaneous colors a device can actually display.

- Most home and business-class LCD monitors are in fact often only 18-bit display devices (262,144 colors)
- LCD monitors that can display true 24-bit color are more expensive

Dithering

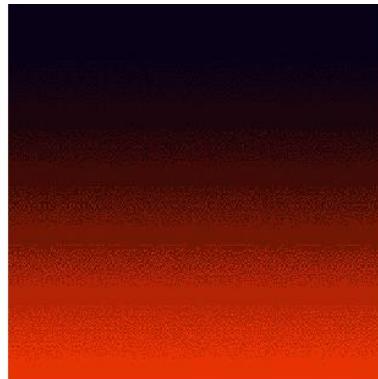
Monitors limited to less than true color create the illusion of more colors by **dithering** the available colors in a diffuse pattern of pixels



24-bit color



8-bit color



5-bit color

Notice the banding due to the dithering.

Image size

How big is this raster?

image size refers to how many pixels it contains,

- Usually pixels wide x pixels high

Notice that you do not use real-world measurement units. The size of an image on-screen is determined by

- the pixel dimensions of the image,
- the monitor size, and
- the computer's display resolution,

only one of which is at the control of the web designer

Image resize

Interpolate

Image size refers to how many pixels it contains,

- Usually pixels wide x pixels high

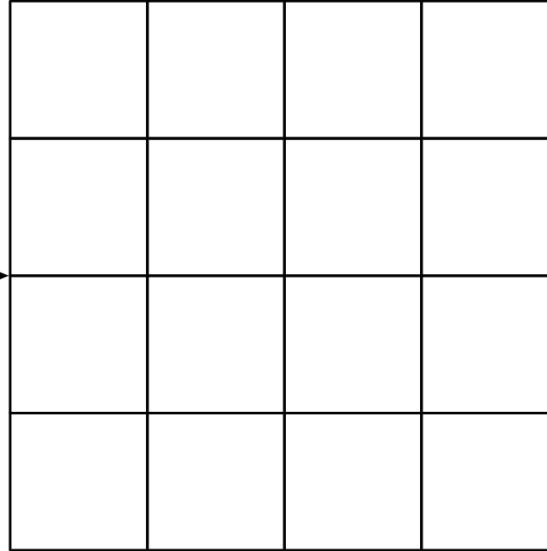
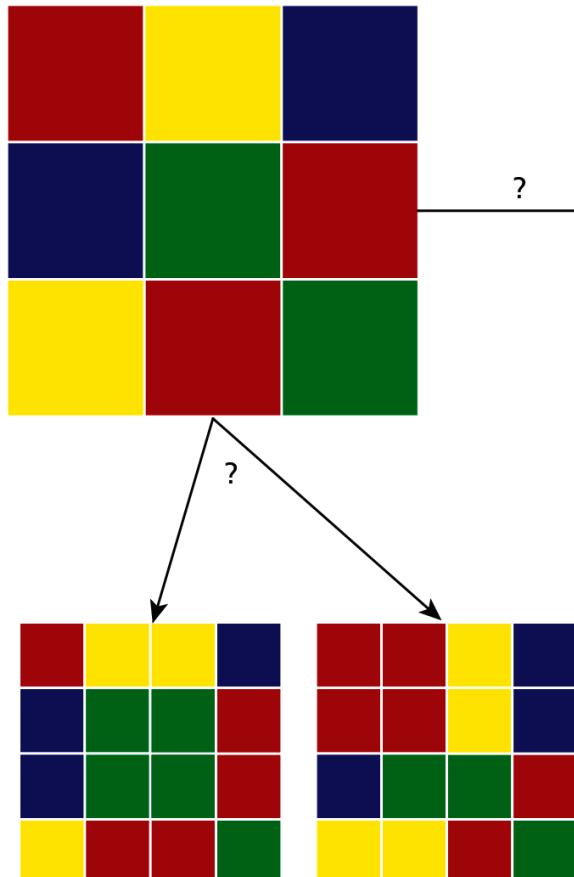
Notice that you do not use real-world measurement units. The size of an image on-screen is determined by

- the pixel dimensions of the image,
- the monitor size, and
- the computer's display resolution,

only one of which is at the control of the web designer

Interpolate

Add or remove pixels due to resizing



If we enlarge the 3x3 image on the left and make it a 4x4 image, what color should each square be?

There is no optimal interpolation solution to the problem of enlarging raster images.

Certain algorithms work better for certain types of images.

Interpolate

Whenever you resize (either larger or smaller) a raster image, the program doing the resizing must **interpolate**

- Making an image larger degrades the image much more than making it smaller
- increasing the size just a small percentage (say 10–20%) may likely result in completely satisfactory results

Enlarging and Reducing



Enlarging a small image a substantial amount will noticeably reduce its quality.



Decreasing the size of an image does reduce the quality as well, but it is not nearly as noticeable.

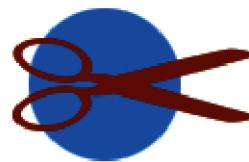
The browser is fast

But not the best looking, resize in photo editors

The best way to change the size of a nonphotographic original is to make the change in the program that created it.

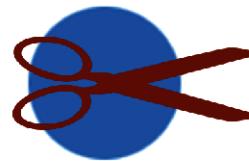


Original (200 x 50)



Scissor and Circle
This is one of those small but witty taglines

Enlarged in browser via
``



Scissor and Circle
This is one of those small but witty taglines

Enlarged original (600 x 150)

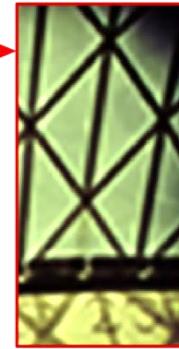
By enlarging the artwork in the program that it was originally created in (i.e., by increasing/decreasing the font and object sizes), the quality is maintained.

The browser is fast

But not the best looking, resize in photo editors



Enlarged using
bicubic interpolation
in Photoshop



Enlarged using
nearest neighbor interpolation
in browser



Display Resolution

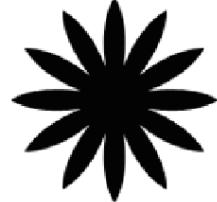
The **display resolution** refers to how many pixels a device can display.

Like image size, it is expressed in terms of the number of pixels horizontally by the number of pixels vertically. Some common display resolutions include:

- 1920 x 1600 px
- 1280 x 1024 px
- 1024 x 768 px
- 320 x 480 px

Display Resolution

Vs monitor size



Title Here
Subtitle Here

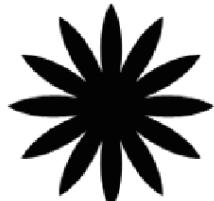
800 x 600 monitor



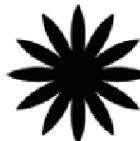
Title Here
Subtitle Here

1600 x 1200 monitor

Effect of monitor size



Title Here
Subtitle Here



Title Here
Subtitle Here



Title Here
Subtitle Here

iPhone

15" monitor

High Density complication

With new high-density displays (such as iPad retina displays) if they used a one-to-one mapping between the pixels in an image to the pixels on the screen, images would be too small.

These devices use something called a **reference pixel**.

For instance, the iPhone 4 has an actual physical display resolution of 640 x 960 px, yet at the browser, from a reference pixel perspective, it claims it has a display resolution of 320 x 480 px.

File Formats

Section 4 of 6

File formats

Each with it's own strengths

There a number of alternative file formats for images in web development.

- JPEG
- GIF
- PNG
- SVG

JPG

JPEG (Joint Photographic Experts Group) or JPG is a 24-bit, true-color file format that is ideal for photographic images. It uses a sophisticated compression scheme that can dramatically reduce the file size.



Original = 931K



JPG Quality 100 = 335K



JPG Quality 60 = 136K



JPG Quality 30 = 77K



JPG Quality 10 = 52K

JPG

JPG is, however, a **lossy compression** scheme, meaning that it reduces the file size by eliminating pixel information with each save. At the highest levels of compression, you will begin to see blotches and noise (also referred to as **artifacts**) appear at edges and in areas of flat color



Notice the noise artifacts at high contrast areas and in areas of flat color.

JPG

More artifacts



Original



Saved as jpeg

Notice the noise and the artifacts!

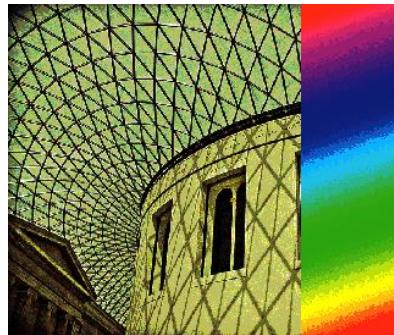
GIF

The **GIF** (Graphic Interchange Format) file was the first image format supported by the earliest web browsers.

GIF is an 8-bit or less format, meaning that it can contain no more than 256 colors!

- Great for Images with flat-bands of color, or with limited number of colors
- Not very good for photographic images

GIF



GIF = 181K



JPEG = 104K



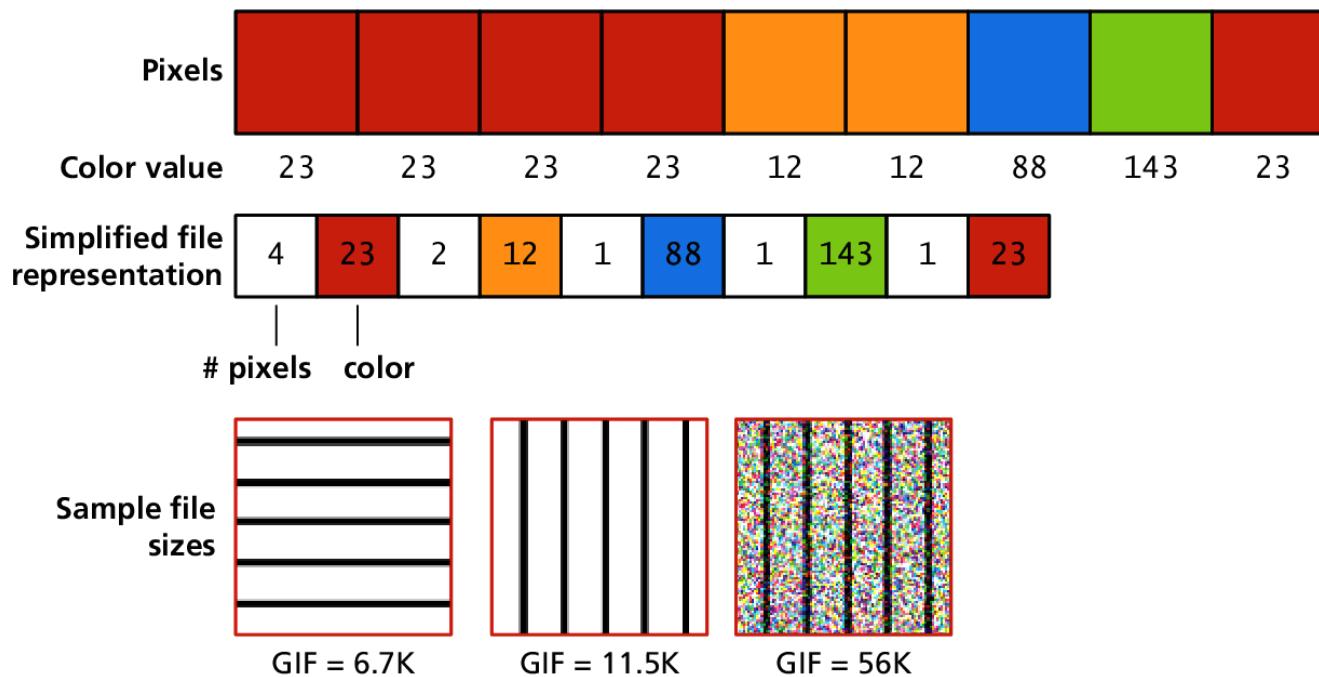
GIF = 23K



JPEG = 40K

GIF

The compression system is called **run-length compression** (also called **LZW compression**).



GIF

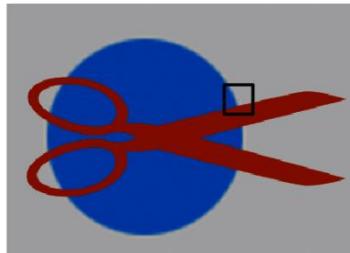
The GIF file format uses indexed color, meaning that an image will have 256 or fewer colors.

You might be wondering which 256 (or fewer) colors?

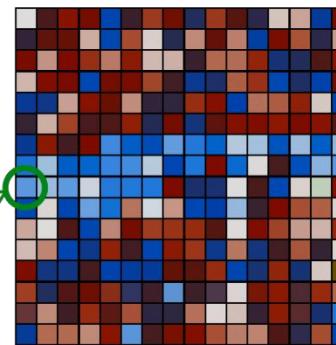
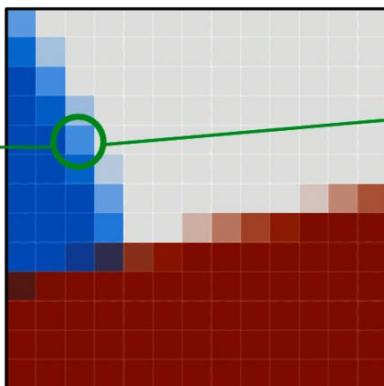
- Index color files dedicate 8 bits (or fewer) to each color pixel in the image.
- Those 8 or fewer bits for each pixel reference a color that is described in a **color palette**

GIF

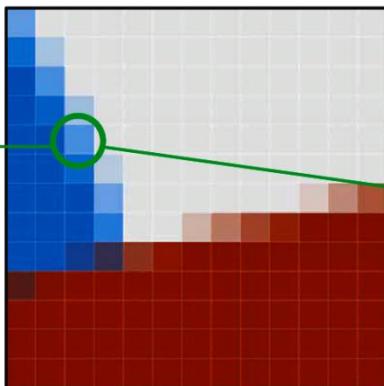
Color Palettes



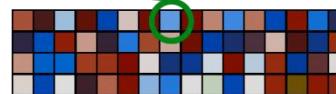
256-color palette = 8 bits per pixel
file size = (100000 pixels x 8) / 8 = 10K



Position 128 in palette
color definition = 00000001 00000111 11111010



Position 7 in palette
color definition = 00000001 00000111 11111010

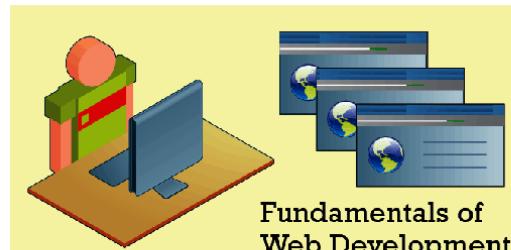


64-color palette = 6 bits per pixel
file size = (100000 pixels x 6) / 8 = 7.5K

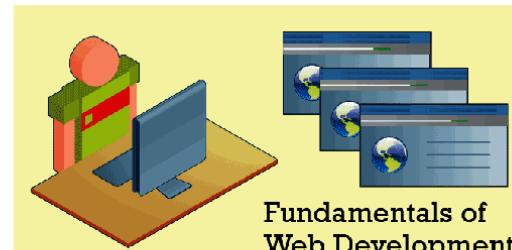
GIF

Optimizing

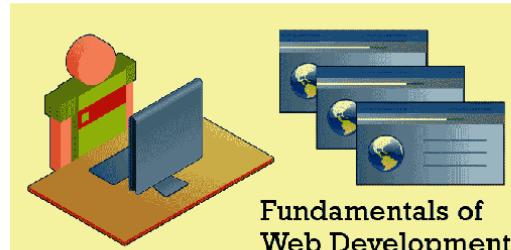
Optimizing GIF images is thus a trade-off between trying to reduce the size of the file as much as possible while at the same time maintaining the image's quality.



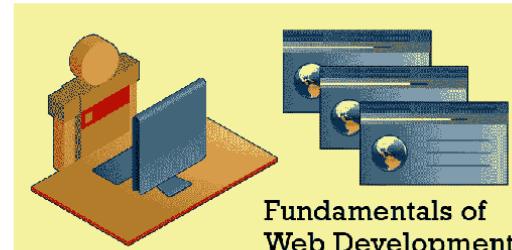
256 colors (8 bits / pixel) = 29 K



64 colors (6 bits / pixel) = 24 K



16 colors (4 bits / pixel) = 19 K

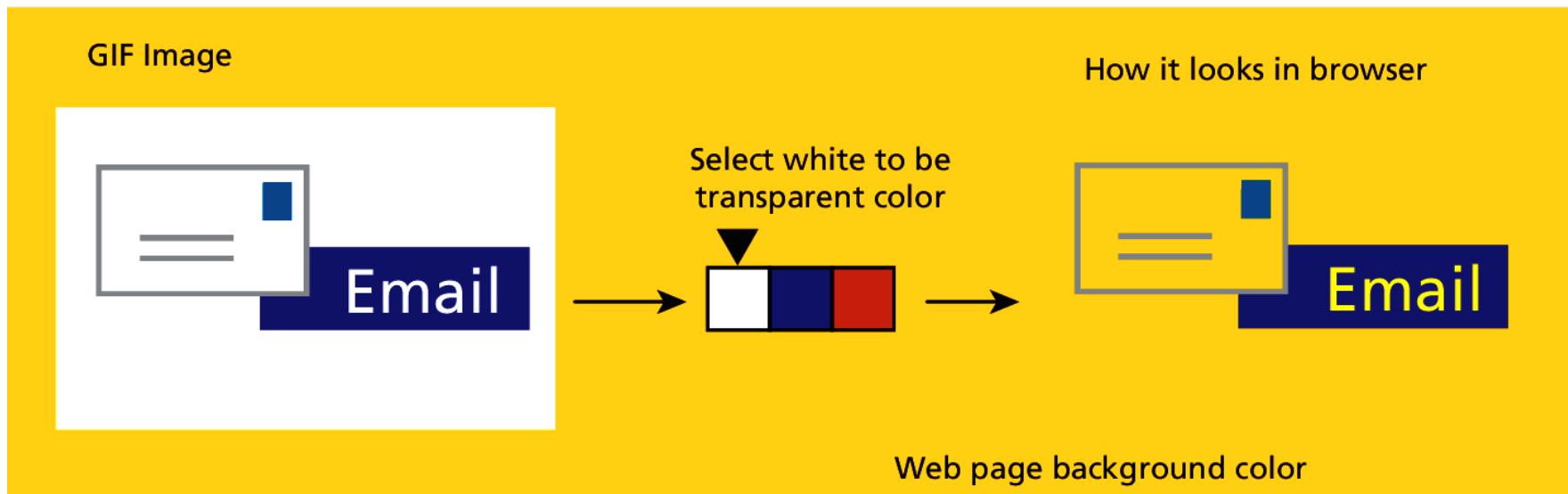


8 colors (3 bits / pixel) = 17 K

GIF

Transparency

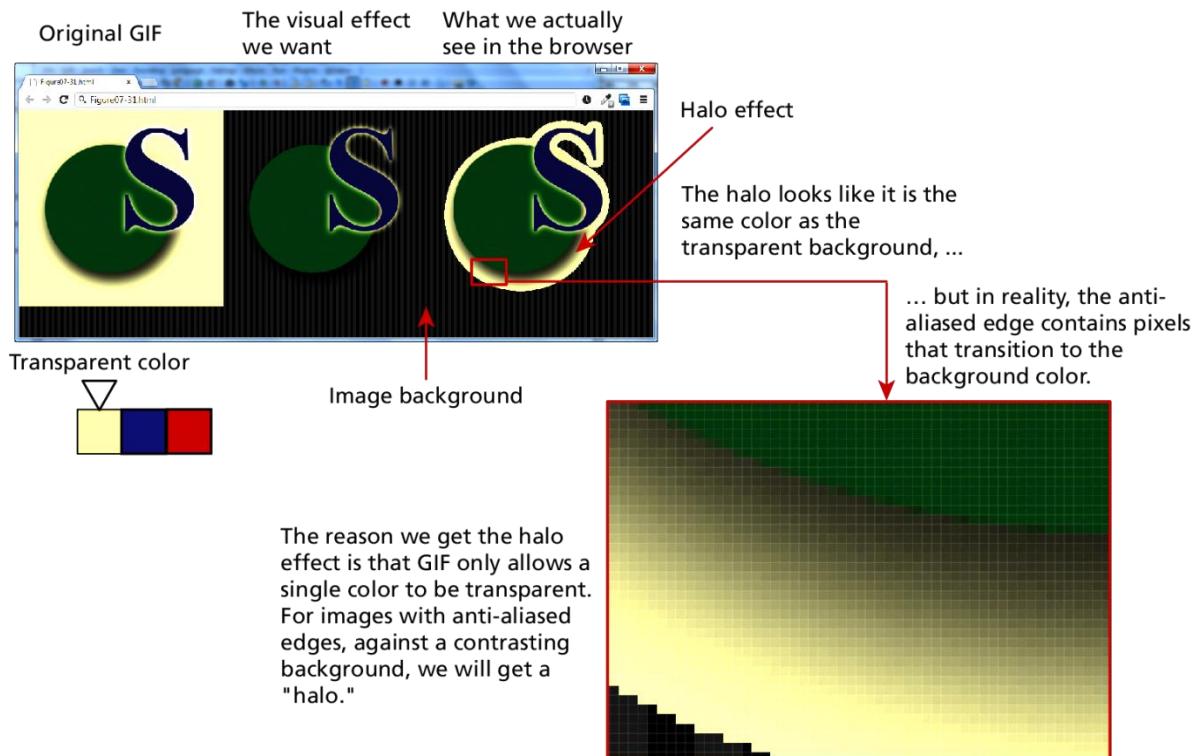
One of the colors in the color lookup table (i.e., the palette) of the GIF can be transparent



GIF

Transparency Trouble

Because GIF has only 1-bit transparency, transparent GIF files can also be disappointing when the graphic contains anti-alias ed edges with pixels of multiple colors



GIF

Animation

Animations are created by having multiple frames, with each frame the equivalent of a separate GIF image.

GIF animations were *de rigueur* back in the middle 1990s, but are now mainly used only for advertisements or for creating retro-web experiences

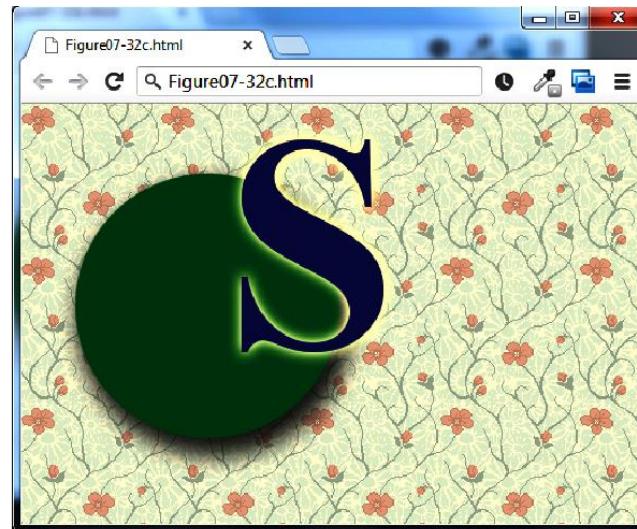
PNG

The **PNG** (Portable Network Graphics) format is a more recent format. Its main features are:

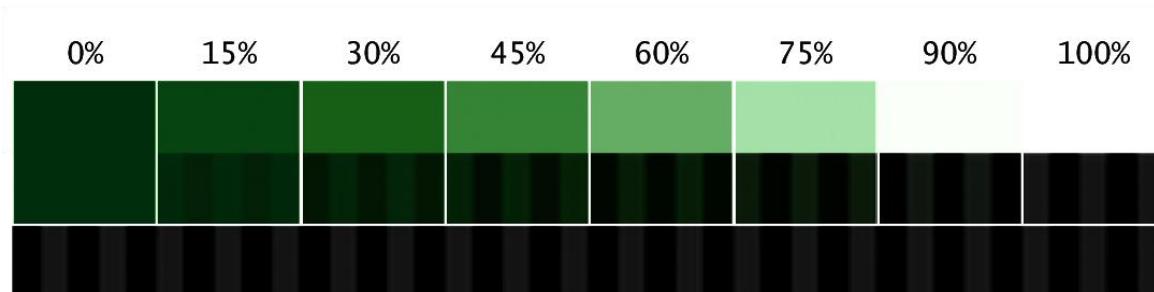
- Lossless compression.
- 8-bit (or 1-bit, 2-bit, and 4-bit) indexed color as well as full 24-bit true color (higher color depths are supported as well).
- **From 1 to 8 bits of transparency**
 - This means that pixels can become progressively more and more transparent along an image's anti-aliased edges, eliminating the transparency halo of GIF images

PNG

Better transparency than GIF.



PNG format with 256 levels of transparency



Transition showing six levels of transparency

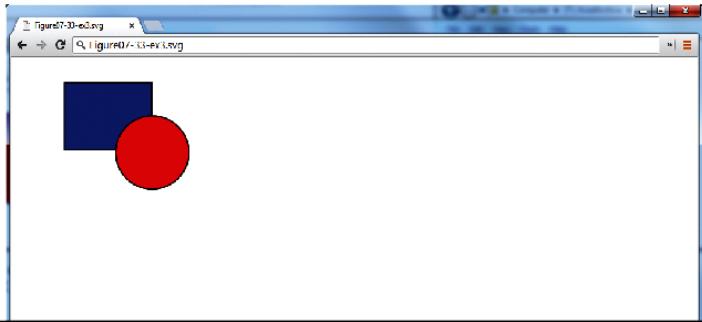
SVG

The **SVG** (Scalable Vector Graphics) file format is a vector format , and now has reasonably solid browser support on the desktop

- Like all vector formats, SVG graphics do not lose quality when enlarged or reduced
- SVG is an open-source standard, and the files are actually XML files
- SVG files end up being part of the HTML document, thus they can be manipulated by JavaScript

SVG

Example



```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.0//EN" "http://www.w3.org/TR/2001/REC-SVG-20010904/DTD/svg10.dtd">
<svg ...>
  <g id="layer1">
    <rect
      style="fill:#0000ff;fill-rule:evenodd;stroke:#000000;stroke-width:1px;stroke-linecap:butt;stroke-linejoin:miter;stroke-opacity:1"
      id="rect4324"           width="66.666664"           height="50.833332"           x="40.833332"           y="19.671911" />
    <path sodipodi:type="arc"
      style="fill:#ff0000;fill-rule:evenodd;stroke:#000000;stroke-width:1px;stroke-linecap:butt;stroke-linejoin:miter;stroke-opacity:1"
      id="path4326"           sodipodi:cx="107.91666"           sodipodi:cy="72.588577"           sodipodi:rx="27.916666"           sodipodi:ry="27.916666"
      d="m 135.83332,72.588577 a 27.916666,27.916666 0 1 1 -55.83333,0 27.916666,27.916666 0 1 1 55.83333,0 z" />
  </g>
</svg>
```

Other Formats

There are many other file formats for graphical information, but because they cannot be viewed by browsers, we are not interested in them as web developers.

Audio and VIDEO

Section 5 of 6

Audio and Video

Until HTML5, adding audio or video to a web page typically required making use of additional, often proprietary, plug-ins to the browser

- Perhaps the most common way of adding audio and video support until recently was through Adobe Flash
- it is possible now with HTML5 to add these media features in HTML without the involvement of a plug-in

Media Concepts

If you thought that image file formats were confusing...

There are a *lot* of different audio and video formats, many with odd and unfamiliar names like OGG and H.264.

Two concepts are essential to understanding media formats

1. Media encoding
2. Container Formats

Media Encoding

AKA Media Compression

Media is encoded using compression/decompression software, usually referred to as a **codec** (for **compression/decompression**)

For web-based **video**:

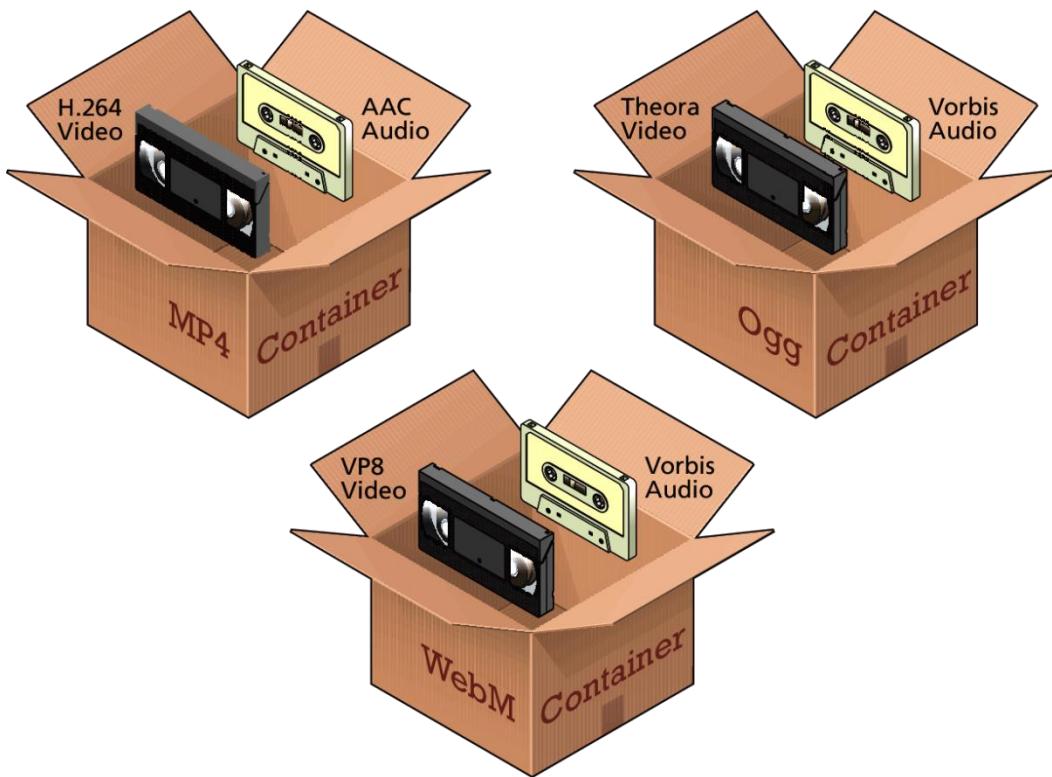
- H.264
- Theora
- VP8

For web-based **audio**:

- MP3
- AAC
- Vorbis

Container Formats

The container format specifies how that information is stored in a file, and how the different information within it is synchronized



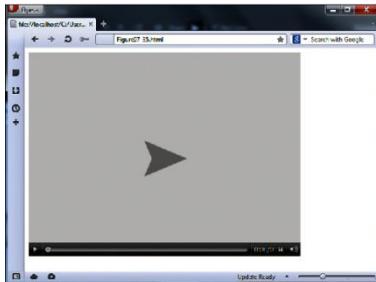
Browser Video Support

- **MP4 container with H.264 Video and AAC Audio.** This combination is generally referred to as **MPEG-4** and has the **.mp4** or **.m4v** file extension.
H.264 is a powerful video codec, but because it is patented and because the browser manufacturer must pay a licensing fee to decode it, not all browsers support it.
- **WebM container with VP8 video and Vorbis audio.** This combination was created by Google to be open-source and royalty free. Files using this combination usually have the **.webm** file extension.
- **Ogg container with Theora video and Vorbis audio.** Like the previous combination, this one is open-source and royalty free. Files using this combination usually have the **.ogv** file extension.

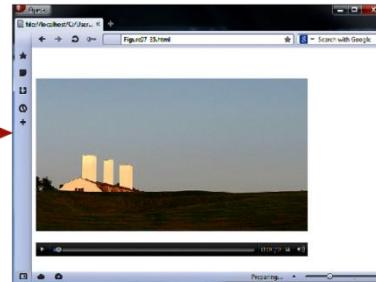
Browser Video Support

Type	IE	Chrome	FireFox	Safari	Opera	Android
MP4+H.264+AAC	Y	Y	N	Y	N	N
WebM+VP8+Vorbis	N	Y	Y	N	Y	Y
Ogg+Theora+Vorbis	N	Y	Y	N	Y	N

Using the <video> element



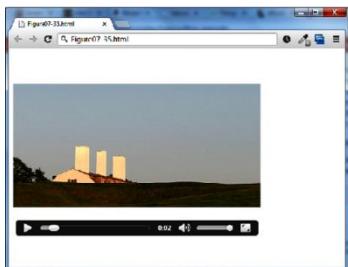
Showing poster image before playback



After playback begins (Opera)

```
<video id="video" poster="preview.png" controls width="480" height="360">
  <source src="sample.mp4" type='video/mp4; codecs="avc1.42E01E, mp4a.40.2"'>
  <source src="sample.webm" type='video/webm; codecs="vp8, vorbis"'>
  <source src="sample.ogv" type='video/ogg; codecs="theora, vorbis"'>

  <!--Use Flash if above video formats not supported -->
  <object width="480" height="360"
    type="application/x-shockwaveflash" data="sample.swf">
    <param name="movie" value="sample.swf">
    <param name="flashvars"
      value="controlbar=over&amp;image=preview.png&amp;file=sample.mp4">
    
  </object>
</video>
```



Chrome



Firefox



Internet Explorer

Browser Audio Support

- **MP3.** Both a container format and a codec. It is patented and requires browser manufacturers to pay licensing fees. Usually has the **.mp3** file extension.
- **WAV.** Also a container and a codec. Usually has the **.wav** file extension.
- **OGG.** Container with Vorbis audio. Open-source. Usually has the **.ogg** file extension.
- **Web.** Container with Vorbis audio. Open-source. Usually has the **.webm** file extension.
- **MP4.** Container with AAC audio. Also requires licensing. Usually has the **.m4a** file extension.

Browser Audio Support

Not all browsers are equal

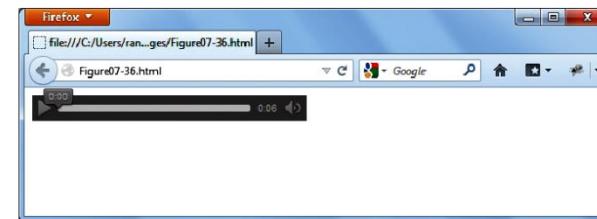
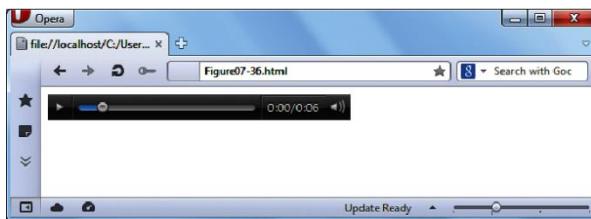
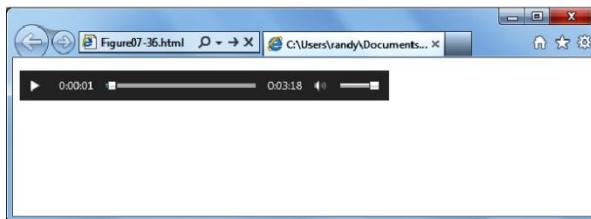
Type	IE	Chrome	FireFox	Safari	Opera	Android
MP3	Y	Y	Partial	Y	N	Y
WAV	N	Y	Y	Y	Y	Y
OGG+Vorbis	N	Y	Y	N	Y	Y
WebM+Vorbis	N	Y	Y	N	Y	Y
MP4+AAC	Y	Y	Partial	Y	N	Y

Using the <audio> element

Not all browsers are equal



```
<audio id="example" controls preload="auto">
  <source src="example.ogg" type="audio/ogg">
  <source src="example.wav" type="audio/webm">
  <source src="example.webm" type="audio/webm">
  <p>Browser doesn't support the audio control</p>
</audio>
```



HTML5 Canvas

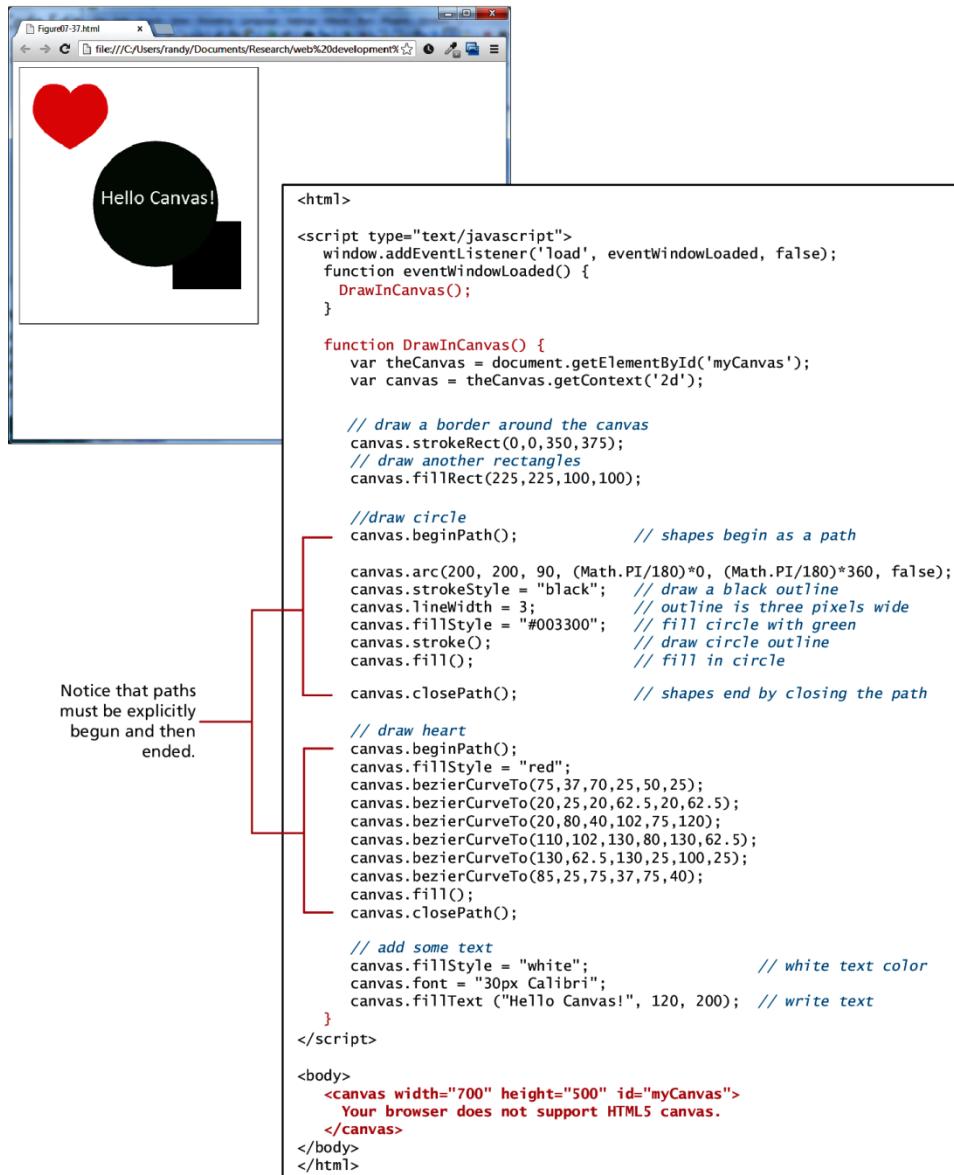
Section 6 of 6

HTML5 Canvas

The <canvas> element is often compared to the Flash environment, since like Flash the <canvas> element can be used to create animations, games, and other forms of interactivity

- There are a variety of specialized JavaScript libraries such as KineticJS, EaselJS, and Fabric.js to aid in the process of creating <canvas> and JavaScript-based sites

HTML5 Canvas



```
<html>
<script type="text/javascript">
window.addEventListener('load', eventWindowLoaded, false);
function eventWindowLoaded() {
    DrawInCanvas();
}

function DrawInCanvas() {
    var theCanvas = document.getElementById('myCanvas');
    var canvas = theCanvas.getContext('2d');

    // draw a border around the canvas
    canvas.strokeRect(0,0,350,375);
    // draw another rectangles
    canvas.fillRect(225,225,100,100);

    //draw circle
    canvas.beginPath();           // shapes begin as a path
    canvas.arc(200, 200, 90, (Math.PI/180)*0, (Math.PI/180)*360, false);
    canvas.strokeStyle = "black"; // draw a black outline
    canvas.lineWidth = 3;        // outline is three pixels wide
    canvas.fillStyle = "#003300"; // fill circle with green
    canvas.stroke();             // draw circle outline
    canvas.fill();               // fill in circle

    canvas.closePath();          // shapes end by closing the path

    // draw heart
    canvas.beginPath();
    canvas.fillStyle = "red";
    canvas.bezierCurveTo(75,37,70,25,50,25);
    canvas.bezierCurveTo(20,25,20,62.5,20,62.5);
    canvas.bezierCurveTo(20,80,40,102,75,120);
    canvas.bezierCurveTo(110,102,130,80,130,62.5);
    canvas.bezierCurveTo(130,62.5,130,25,100,25);
    canvas.bezierCurveTo(85,25,75,37,75,40);
    canvas.fill();
    canvas.closePath();

    // add some text
    canvas.fillStyle = "white";           // white text color
    canvas.font = "30px Calibri";
    canvas.fillText ("Hello Canvas!", 120, 200); // write text
}
</script>

<body>
<canvas width="700" height="500" id="myCanvas">
Your browser does not support HTML5 canvas.
</canvas>
</body>
</html>
```

Notice that paths must be explicitly begun and then ended.

What You've Learned

1. Digital Representation
2. Color Models
3. Image Concepts
4. File Formats
5. Audio and Video
6. HTML Canvas