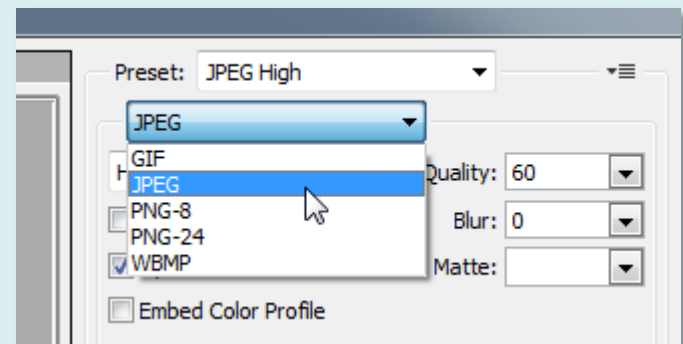


# Images and Colour with CSS

Webpage Design

# 4 Common Web Formats

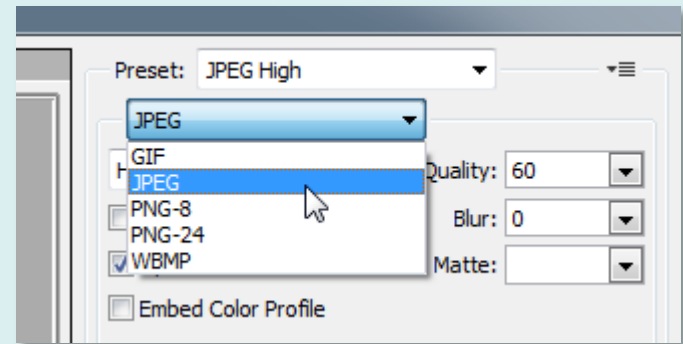
- JPEG (Joint Photographic Experts Group)
- GIF (Graphics Interchange Format)
- PNG-8 (Portable Network Graphics 8bit)
- PNG-24 (Portable Network Graphics 24bit)



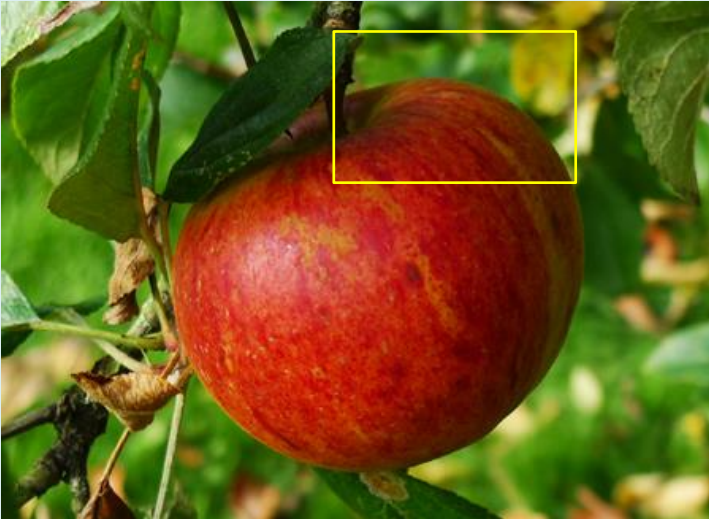
# Colour Depth

- JPEG      True Colour (24 bit)
- GIF      256 colours (8 bit)
- PNG-8    256 colours (8 bit)
- PNG-24   True Colour (24 bit)

“True” colour image formats like JPEG can display around 16 million colours (16,777,216 to be precise, that’s 256 red x 256 green x 256 blue). Since the human is capable of discerning fewer colours than this, true colour images are equivalent to what we see in the real world. 256 colour images are sometimes referred to as using an “indexed” colour palette.



# 256 Colour Dithering



File formats with 256 colour palettes (GIF & PNG-8) are not good for photographic images.

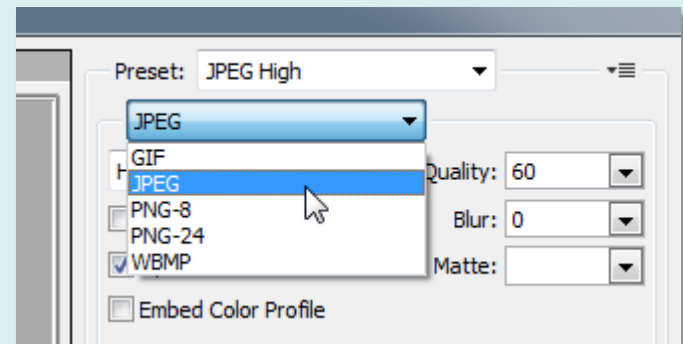
They use a process called “dithering” to simulate additional colours.

<http://en.wikipedia.org/wiki/Dither>

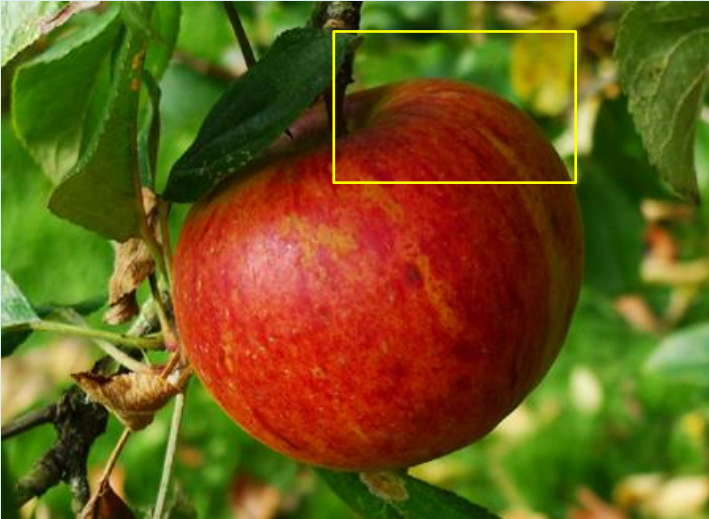
# Killer Feature

- JPEG      Variable compression
- GIF      Frame animation
- PNG-8      none (small file size?)
- PNG-24      Alpha transparency

You might wonder why we need 4 different file formats for web design work. Well, they all have different attributes and each one has a “killer” feature that the others don’t and this makes them useful in different situations.



# JPEG Compression



The JPEG file format is particularly good for photographs because it has a “true colour” palette and variable compression algorithm that can be used to control file size.

Over-compression can result in “artefacts” appearing in the image.

# JPEG File Size vs. Image Quality



High compression  
Low quality  
Small file size (9KB)



Medium compression  
Medium quality  
Medium file size (11KB)



Low compression  
High quality  
Large file size (14KB)

Uses: Photographs on the Web

Features: 16 Million colours, lossy compression, small files

# GIF Animation

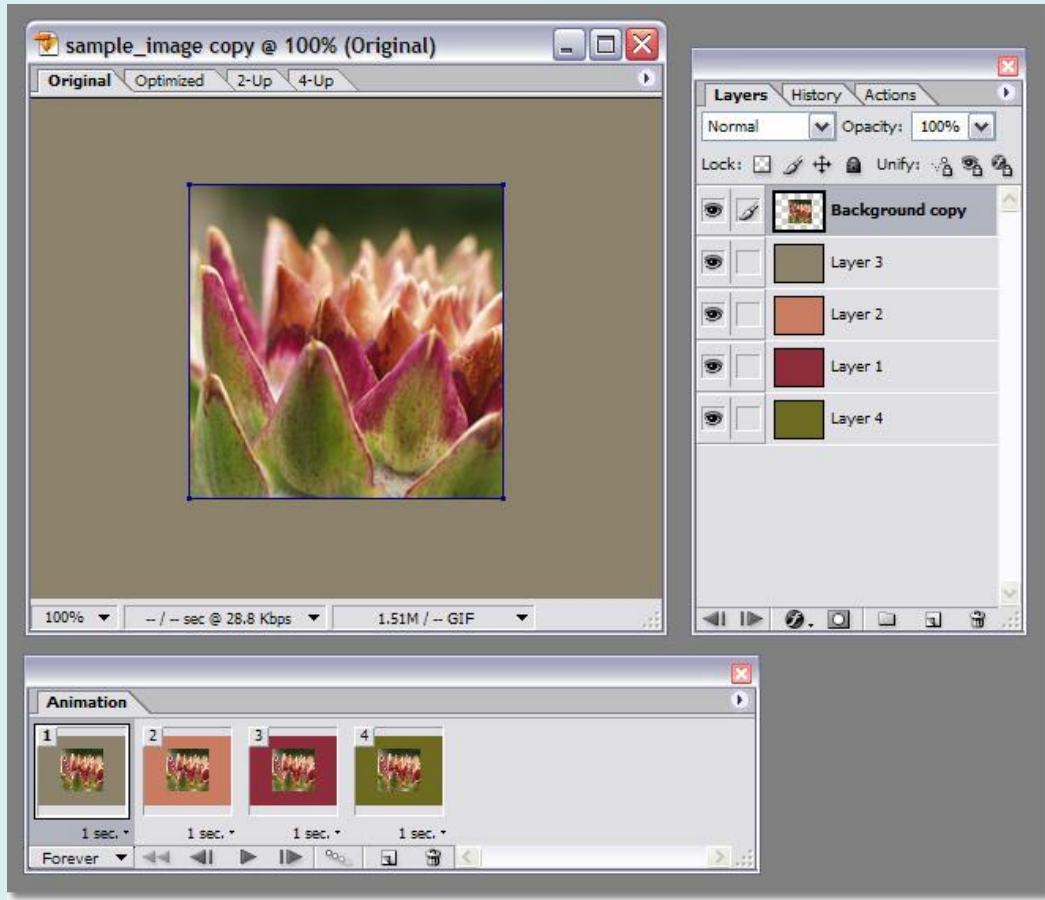


**GIF Animation**





# Creating GIF Animation



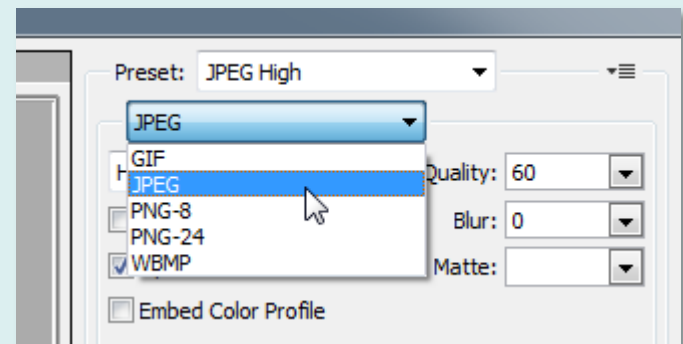
Photoshop CS3 and above can be used to build GIF animations. Using a sequence of different layer settings for each frame, a simple (or complex) frame-based animation can be created. The Photoshop Animation palette allows the time between each frame to be varied and it will even tween between different layer states.

# Animation Result

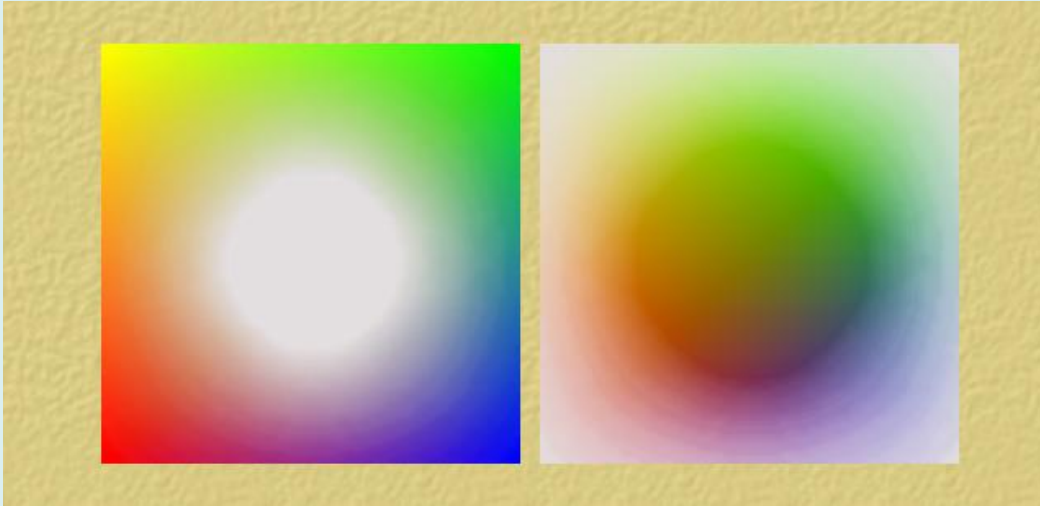


# Transparency

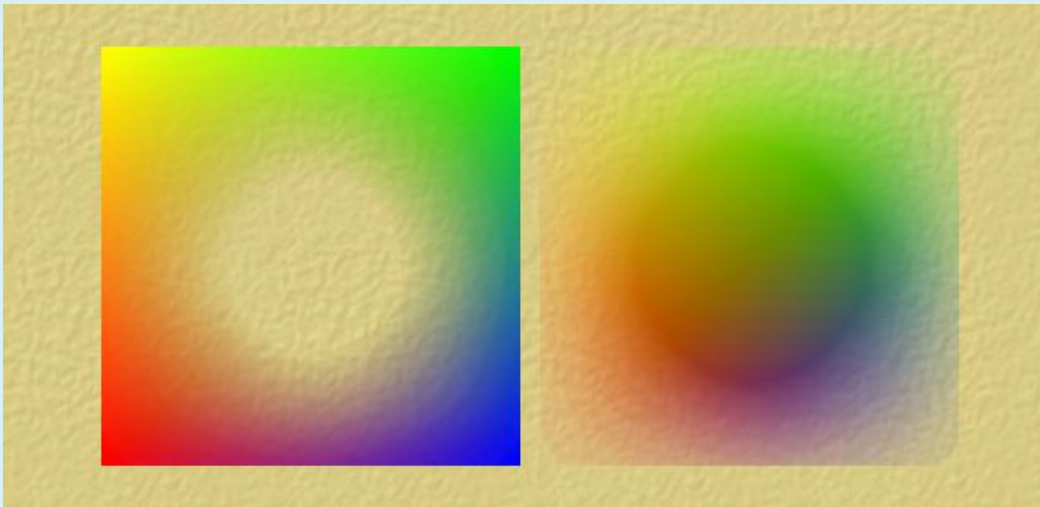
- JPEG      None
- GIF      1 colour
- PNG-8      1 colour
- PNG-24      Alpha channel (256 colour)



# PNG-24 Alpha Transparency



Alpha transparency in IE6



Alpha transparency in IE7 and above

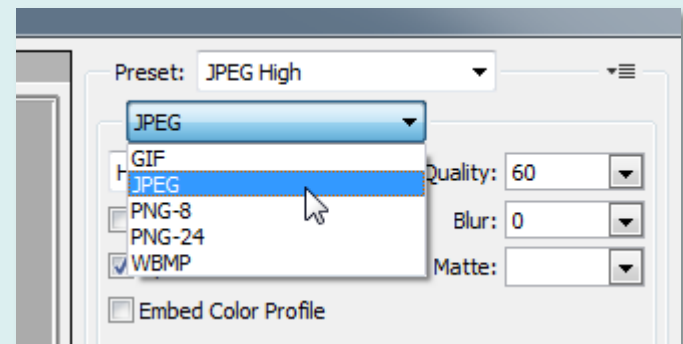
The use of PNG alpha transparency has been limited by browser support (IE6 and below do not support it).

Now that IE6 is a minority browser, it is possible to use this feature to great effect.

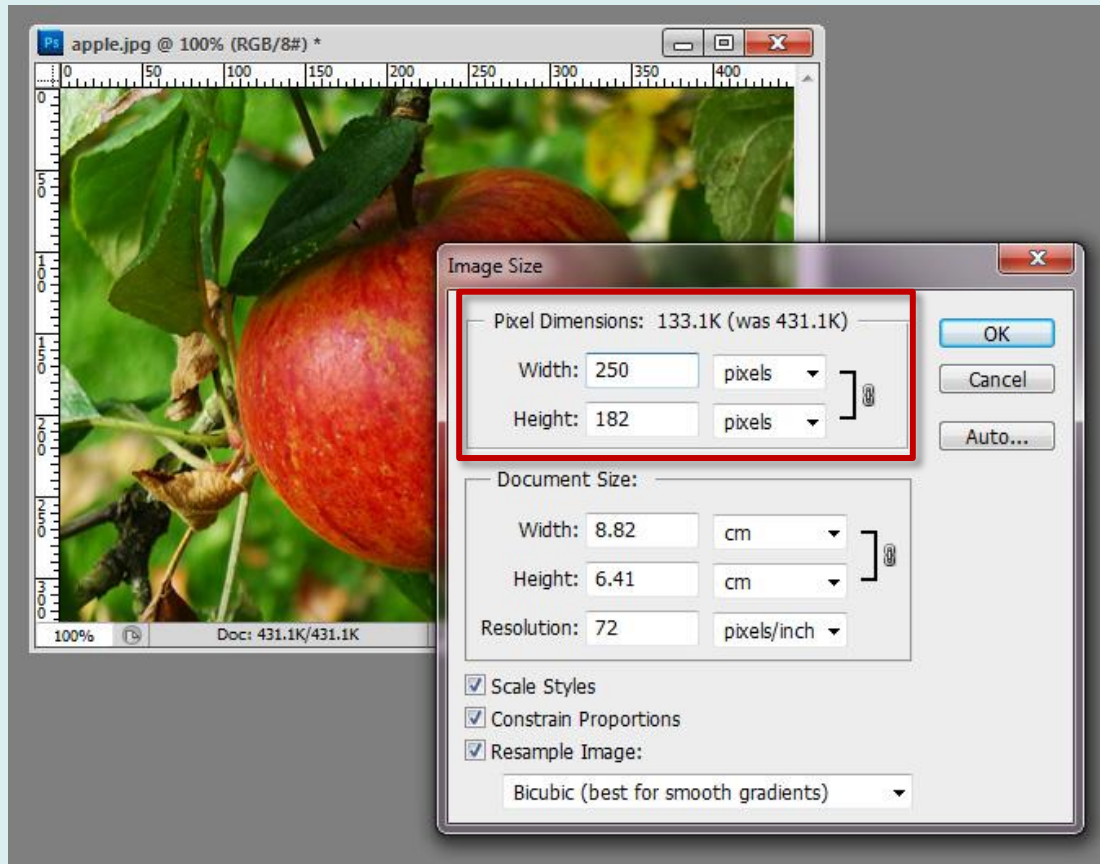
For example, a foreground image can be given a drop-shadow that will work with any background colour or image.

# Best used for...

- JPEG      Photographs
- GIF      Simple graphics or animation
- PNG-8      Simple graphics (often smaller than GIF)
- PNG-24      Complex graphics



# Set your image size...

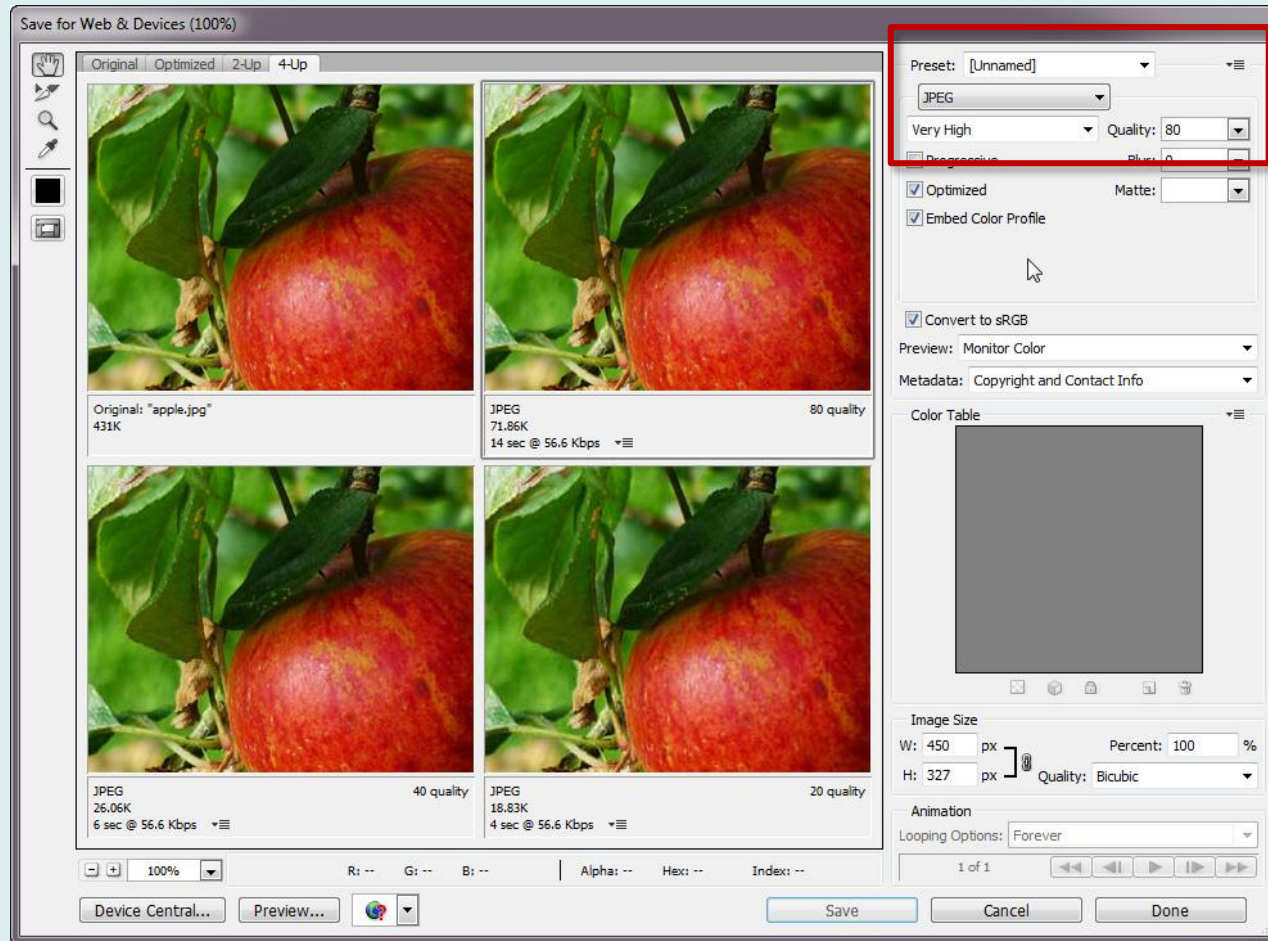


When working with web graphics, it's a good idea to set your Photoshop rulers to "pixels" ([Edit > Preferences > Units & Rulers...](#)) because that's the most convenient unit type for our work.

You can then resize your image ([Image > Image Size...](#)) by entering either the required width or height value in the dialogue box. By default, the aspect ratio is locked, so changing either the width or the height will result in a proportionally scaled image.



# Save for Web...



Photoshop has a special tool for helping us to save our images ready for the web. It offers a choice of all 4 formats of interest to us and it can give us previews of how the image will look once it is saved. It can also tell us how big each file will be and how long it will take to download.

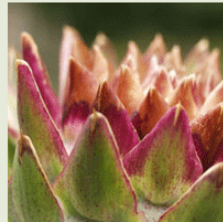
# Image Positioning with CSS



# CSS Floats

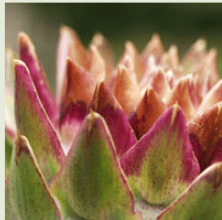
Mauris rhoncus, justo vitae imperdiet sodales, tellus diam ultrices est, non condimentum felis est quis elit. Maecenas dolor. Nulla facilisi. Quisque velit. Nulla facilisi. Sed sed velit. Nam interdum.

Suspendisse congue risus et lectus. Proin pellentesque porttitor nisl. Duis aliquam. In eu felis. Aliquam erat volutpat. Aliquam scelerisque volutpat leo. Aenean tempor sapien quis lorem. Nulla lorem metus, tincidunt at, porttitor ut, vestibulum ut, nunc. Etiam ac erat non tortor pellentesque bibendum. Ut sodales volutpat ligula. Maecenas vestibulum suscipit sem. Nunc tristique augue laoreet ipsum. Phasellus quis dolor. Fusce pellentesque. Nam mollis ullamcorper nunc. Nam viverra sodales nunc. Praesent vulputate neque vel orci consequat tincidunt.



Sed elit. Nulla sapien risus, pulvinar vel, porttitor at, lobortis non, leo. Vivamus a odio. Sed vulputate quam a nunc. Integer aliquam. Ut ac enim nec ligula venenatis pretium. Cras venenatis nisl aliquet neque. Maecenas nisl. Proin faucibus

odio nibh eu enim. Nullam posuere leo a lacus. Pellentesque odio. Maecenas tempus dui at purus tristique mattis. Mauris nisl ligula, commodo non, consectetur nec, semper in, sem.



Quisque a nisi a mauris dignissim ultricies. Ut ipsum purus, hendrerit ut, varius eu, fringilla sodales, magna. Cras quis ipsum in lectus volutpat egestas. Sed eget nibh. Morbi commodo dapibus nunc. Pellentesque convallis tempor erat. Ut commodo. Aliquam condimentum. Praesent lacus tortor, dapibus sit amet, congue eget, blandit sit amet, arcu. Fusce euismod ultricies urna. Sed bibendum lectus vel nulla. Vivamus faucibus.

Donec mollis, ante at pretium scelerisque, pede magna imperdiet leo, sed condimentum velit augue ac metus.

Aliquam iaculis nonummy quam. Morbi elementum quam in lectus. Vestibulum congue, lectus sed convallis malesuada, urna lectus rhoncus tellus, nec rutrum lectus magna a turpis. Suspendisse elementum sem a dui. Curabitur dictum mauris sit amet urna. Cras ornare. Donec sapien purus, vehicula eget, volutpat quis, ullamcorper

CSS floats are one of the most powerful tools we have to control the relative position of HTML elements within a page. You're going to be using them a lot for controlling page layouts but for now, let's consider their use for placing images within columns of text.

By default, images will just run inline with text (they are inline elements) but if we apply a CSS float property rule to them, we can control where they appear in relation to the surrounding text.

Floating an image to the right causes text to flow to the left and vice versa.

# Left Placement

odio nibh eu enim. Nullam posuere leo a lacus. Pellentesque odio. Maecenas tempus dui at purus tristique mattis. Mauris nisi ligula, commodo non, consectetur nec, semper in, sem.



Quisque a nisi a mauris dignissim ultricies. Ut ipsum purus, hendrerit ut, varius eu, fringilla sodales, magna. Cras quis ipsum in lectus volutpat egestas. Sed eget nibh. Morbi commodo dapibus nunc. Pellentesque convallis tempor erat. Ut commodo. Aliquam condimentum. Praesent lacus tortor, dapibus sit amet, congue eget, blandit sit amet, arcu. Fusce euismod ultricies urna. Sed bibendum lectus vel nulla. Vivamus faucibus.

Donec mollis, ante at pretium scelerisque, pede magna imperdiet leo, sed condimentum velit augue ac metus.

Aliquam iaculis nonummy quam. Morbi elementum

quam in lectus. Vestibulum congue, lectus sed convallis malesuada, urna lectus rhoncus tellus, nec rutrum lectus magna a turpis. Suspendisse elementum sem a dui. Curabitur dictum mauris sit amet urna. Cras ornare. Donec sapien purus, vehicula eget, volutpat quis, ullamcorper

```

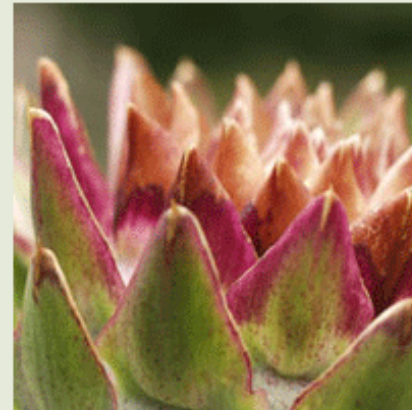
```

```
img.left {  
  float: left;  
  margin: 7px 20px 10px 0;  
}
```

# Right Placement

Mauris rhoncus, justo vitae imperdiet sodales, tellus diam ultrices est, non condimentum felis est quis elit. Maecenas dolor. Nulla facilisi. Quisque velit. Nulla facilisi. Sed sed velit. Nam interdum.

Suspendisse congue risus et lectus. Proin pellentesque porttitor nisl. Duis aliquam. In eu felis. Aliquam erat volutpat. Aliquam scelerisque volutpat leo. Aenean tempor sapien quis lorem. Nulla lorem metus, tincidunt at, porttitor ut, vestibulum ut, nunc. Etiam ac erat non tortor pellentesque bibendum. Ut sodales volutpat ligula. Maecenas vestibulum suscipit sem. Nunc tristique augue laoreet ipsum. Phasellus quis dolor. Fusce pellentesque. Nam mollis ullamcorper nunc. Nam viverra sodales nunc. Praesent vulputate neque vel orci consequat tincidunt.



Sed elit. Nulla sapien risus, pulvinar vel, porttitor at, lobortis non, leo. Vivamus a odio. Sed vulputate quam a nunc. Integer aliquam. Ut ac enim nec ligula venenatis pretium. Cras venenatis nisl aliquet neque. Maecenas nisi. Proin faucibus

```

```

```
img.right {  
    float: right;  
    margin: 7px 0 10px 20px;  
}
```

<http://css.maxdesign.com.au/floatutorial/tutorial0101.htm>

# Centre Placement

libero odio ultrices felis, non sagittis lacus orci porta diam. Curabitur eu ipsum. Aenean nisl quam, hendrerit non, auctor a, semper eget, sapien. Proin sit amet dolor at leo ultrices vestibulum.



Ut tellus nisl, egestas eget, facilisis ac, consequat ac, leo. Etiam iaculis, quam hendrerit aliquet consequat, mauris lacus feugiat eros, sit amet sagittis felis nibh ac lectus. Proin et eros. Nulla erat nulla, aliquam at, rutrum et, fermentum ut, auque. Donec tempus magna id felis. Etiam

```

```

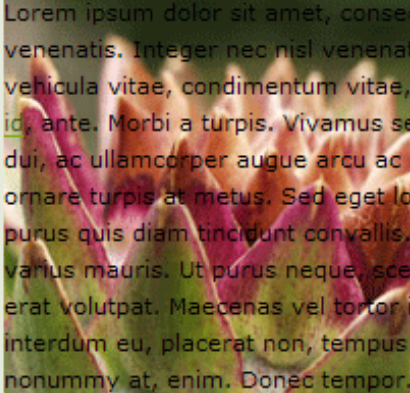
```
img.center {  
  display: block;  
  margin: 10px auto;  
}
```

Centring an image is a bit different, we can't float centre. We need two declarations, one to change the image behaviour from *inline* to *block-level* and another to make the left and right margins equal using the *auto* value.

# Background Images

# Background

id, velit. Curabitur euismod, massa sed lobortis facilisis, libero odio ultrices felis, non sagittis lacus orci porta diam. Curabitur eu ipsum. Aenean nisl quam, hendrerit non, auctor a, semper eget, sapien. Proin sit amet dolor at leo ultrices vestibulum.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla in ligula in risus fermentum venenatis. Integer nec nisl venenatis nisl ultrices auctor. Proin arcu dui, bibendum sit amet, vehicula vitae, condimentum vitae, lorem. Mauris enim est, vehicula eu, molestie ut, vulputate id, ante. Morbi a turpis. Vivamus semper, pede et lacinia accumsan, dolor lacus fermentum dui, ac ullamcorper augue arcu ac est. Quisque eget odio. Curabitur molestie leo a urna. Morbi ornare turpis at metus. Sed eget lorem at augue blandit rutrum. Nulla quis lectus. Maecenas a purus quis diam tincidunt convallis. Proin in nisi in sem rutrum dapibus. Sed pellentesque varius mauris. Ut purus neque, scelerisque vitae, varius vitae, fermentum ac, nulla. Aliquam erat volutpat. Maecenas vel tortor in dui tristique sagittis. Sed sed quam. Donec elit eros, interdum eu, placerat non, tempus et, nisl. In justo massa, aliquet ut, ullamcorper id, nonummy at, enim. Donec tempor. Sed congue. Donec in mauris. Vestibulum vestibulum euismod elit. Praesent ac risus. Nam tristique. Sed feugiat risus vel felis. Sed libero. Praesent nulla leo, mattis et, faucibus fermentum, venenatis sit amet, libero. Curabitur porta. Nam congue, turpis nec condimentum facilisis, risus quam eleifend dolor, ac fermentum massa dui in ipsum.

The CSS background property can be used to display an image in the background of an element.

Obviously, this should be used with caution because background images often make a design look too fussy or make the overlying text difficult to read.

```
<p class="background">Lorem ipsum dolor... </p>
```

```
p.background {  
    background: url(artichoke.jpg) no-repeat;  
}
```

In this example, an image is shown in the background of a paragraph element.



# Background Repeat

id, velit. Curabitur euismod, massa sed lobortis facilisis, libero odio ultrices felis, non sagittis lacus orci porta diam. Curabitur eu ipsum. Aenean nisl quam, hendrerit non, auctor a, semper eget, sapien. Proin sit amet dolor at leo ultrices vestibulum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla in ligula in risus fermentum venenatis. Integer nec nisl venenatis nisl ultrices auctor. Proin arcu dui, bibendum sit amet, vehicula vitae, condimentum vitae, lorem. Mauris enim est, vehicula eu, molestie ut, vulputate id, ante. Morbi a turpis. Vivamus semper, pede et lacinia accumsan, dolor lacus fermentum dui, ac ullamcorper augue arcu ac est. Quisque eget odio. Curabitur molestie leo a urna. Morbi ornare turpis at metus. Sed eget lorem at augue blandit rutrum. Nulla quis lectus. Maecenas a purus quis diam tincidunt convallis. Proin in nisi in sem rutrum dapibus. Sed pellentesque varius mauris. Ut purus neque, scelerisque vitae, varius vitae, fermentum ac, nulla. Aliquam erat volutpat. Maecenas vel tortor in dui tristique sagittis. Sed sed quam. Donec elit eros, interdum eu, placerat non, tempus et, nisl. In justo massa, aliquet ut, ullamcorper id, nonummy at, enim. Donec tempor. Sed congue. Donec in mauris. Vestibulum vestibulum euismod elit. Praesent ac risus. Nam tristique. Sed feugiat risus vel felis. Sed libero. Praesent nulla leo, mattis et, faucibus fermentum, venenatis sit amet, libero. Curabitur porta. Nam congue, turpis nec condimentum facilisis, risus quam eleifend dolor, ac fermentum massa dui in ipsum.

By default, all background images will repeat both vertically and horizontally (the X and Y directions) but we can control this behaviour using the *repeat* value.

In this example, the image repeats only in the horizontal direction.

```
<p class="background">Lorem ipsum dolor... </p>
```

```
p.background {  
  background: url(artichoke.jpg) repeat-x;  
}
```

# Background Position

id, velit. Curabitur euismod, massa sed lobortis facilisis, libero odio ultrices felis, non sagittis lacus orci porta diam. Curabitur eu ipsum. Aenean nisl quam, hendrerit non, auctor a, semper eget, sapien. Proin sit amet dolor at leo ultrices vestibulum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla in ligula in risus fermentum venenatis. Integer nec nisl venenatis nisl ultrices auctor. Proin arcu dui, bibendum sit amet, vehicula vitae, condimentum vitae, lorem. Mauris enim est, vehicula eu, molestie ut, vulputate id, ante. Morbi a turpis. Vivamus semper, pede et lacinia accumsan, dolor lacus fermentum dui, ac ullamcorper augue arcu ac est. Quisque eget odio. Curabitur molestie leo a urna. Morbi ornare turpis at metus. Sed eget lorem at augue blandit rutrum. Nulla quis lectus. Maecenas a purus quis diam tincidunt convallis. Proin in nisi in sem rutrum dapibus. Sed pellentesque varius mauris. Ut purus neque, scelerisque vitae, varius vitae, fermentum ac, nulla. Aliquam erat volutpat. Maecenas vel tortor in dui tristique sagittis. Sed sed quam. Donec elit eros, interdum eu, placerat non, tempus et, nisl. In justo massa, aliquet ut, ullamcorper id, nonummy at, enim. Donec tempor. Sed congue. Donec in mauris. Vestibulum vestibulum euismod elit. Praesent ac risus. Nam tristique. Sed feugiat risus vel felis. Sed libero. Praesent nulla leo, mattis et, faucibus fermentum, venenatis sit amet, libero. Curabitur porta. Nam congue, turpis nec condimentum facilisis, risus quam eleifend dolor, ac fermentum massa dui in ipsum.

A single background image placed in the middle of the parent element. The position values (50% 50%) places the image half-way down and half-way across the element, effectively centring the image.

```
<p class="background">Lorem ipsum dolor... </p>
```

```
p.background {  
  background: url(artichoke.jpg) no-repeat 50% 50%;  
}
```



# Background Image & Colour

id, velit. Curabitur euismod, massa sed lobortis facilisis, libero odio ultrices felis, non sagittis lacus orci porta diam. Curabitur eu ipsum. Aenean nisl quam, hendrerit non, auctor a, semper eget, sapien. Proin sit amet dolor at leo ultrices vestibulum.

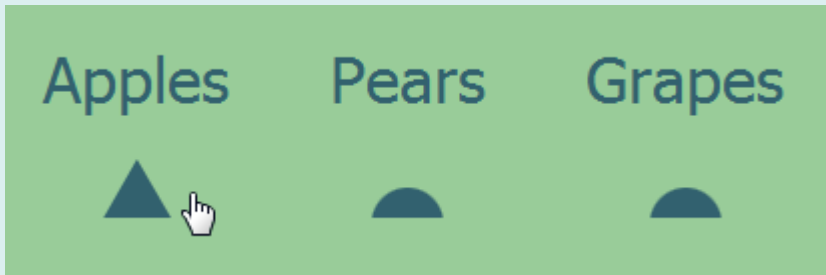
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla in ligula in risus fermentum venenatis. Integer nec nisl venenatis nisl ultrices auctor. Proin arcu dui, bibendum sit amet, vehicula vitae, condimentum vitae, lorem. Mauris enim est, vehicula eu, molestie ut, vulputate id, ante. Morbi a turpis. Vivamus semper, pede et lacinia accumsan, dolor lacus fermentum dui, ac ullamcorper augue arcu ac est. Quisque eget odio. Curabitur molestie leo a urna. Morbi ornare turpis at metus. Sed eget lorem at augue blandit rutrum. Nulla quis lectus. Maecenas a purus quis diam tincidunt convallis. Proin in nisi in sem rutrum dapibus. Sed pellentesque varius mauris. Ut purus neque, scelerisque vitae, varius vitae, fermentum ac, nulla. Aliquam erat volutpat. Maecenas vel tortor in dui tristique sagittis. Sed sed quam. Donec elit eros, interdum eu, placerat non, tempus et, nisl. In justo massa, aliquet ut, ullamcorper id, nonummy at, enim. Donec tempor. Sed sagittis. Donec in mauris. Vestibulum vestibulum euismod elit. Praesent ac risus. Nam tristique. Sed feugiat risus vel felis. Sed libero. Praesent nulla leo, mattis et, faucibus fermentum, venenatis sit amet, libero. Curabitur porta. Nam congue, turpis nec condimentum facilisis, risus quam eleifend dolor, ac fermentum massa dui in ipsum.

The background property allows us to have both a background image and a background colour on the same element.

```
<p class="background">Lorem ipsum dolor... </p>
```

```
p.background {  
  background: #D0D5C6 url(artichoke.jpg) no-repeat 50% 50%;  
}
```

# CSS Rollovers



```
a {  
    background-image:url(semi-circle.png);  
    background-position:50% 100%;  
    background-repeat:no-repeat;  
}  
a:hover {  
    background-image:url(triangle.png);  
}
```

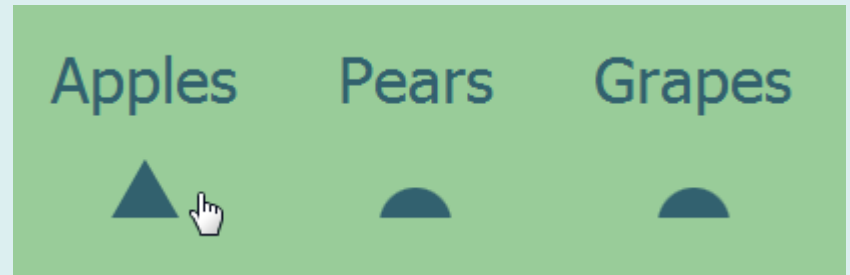
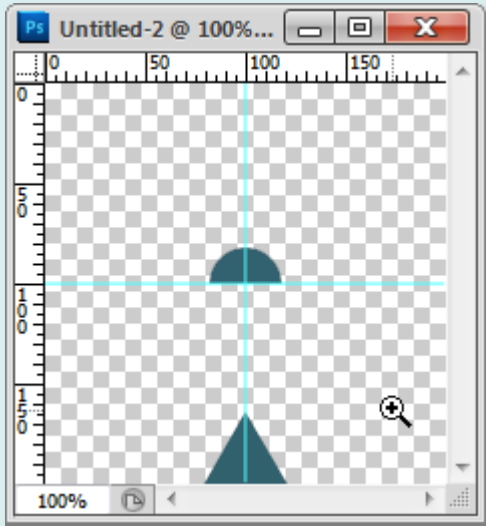
The pseudo-selectors of the anchor element can be used to switch one background image for another.

In this example, the semi-circle.png image is swapped with the triangle.png image when the user mouses over the anchor element.

Before CSS, this action could only be achieved using javascript.

However, there is an even more efficient way to achieve the same result using just a single image file...

# Sprites



```
a {  
    background-image:url(sprite.png);  
    background-position:50% 8%;  
    background-repeat:no-repeat;  
}  
a:hover {  
    background-position:50% 94%;  
}
```

The effect is achieved by changing the background position of the image to show the required sprite.

Sprites have 2 key benefits:

1. They require only one call to the server = efficient.
2. They are effectively “preloaded” so there is no lag on hover.

Colour

# Colour Clash!

# Colour Harmony

# High Contrast

# Low Contrast

Contrast can be used to reinforce information hierarchy

# Colour contrast for accessibility

## Colour Contrast Check

Date created: January 11, 2005  
Date last modified: December 6, 2009

Foreground Colour:	Background Colour:	Results
<div>#B2C5AB</div>	<div>#1B3333</div>	<div>This is example text. Some of it bolded. Some of it italicized.</div>
Red: <input type="text"/>	Red: <input type="text"/>	Brightness Difference: ( $\geq 125$ ) <input type="text" value="144.53"/>
Green: <input type="text"/>	Green: <input type="text"/>	Colour Difference: ( $\geq 500$ ) <input type="text" value="417"/>
Blue: <input type="text"/>	Blue: <input type="text"/>	Are colours compliant? <input type="text" value="sort of"/>
Hue (°): <input type="text"/>	Hue (°): <input type="text"/>	Contrast Ratio <input type="text" value="7.31"/>
Saturation (%): <input type="text"/>	Saturation (%): <input type="text"/>	WCAG 2 AA Compliant <input type="text" value="YES"/>
Value (%): <input type="text"/>	Value (%): <input type="text"/>	WCAG 2 AA Compliant (18pt+) <input type="text" value="YES"/>
		WCAG 2 AAA Compliant <input type="text" value="YES"/>
		WCAG 2 AAA Compliant (18pt+) <input type="text" value="YES"/>

## About the Colour Contrast Check

The Colour Contrast Check Tool allows to specify a foreground and a background colour and determine if they provide enough of a contrast "when viewed by someone having color deficits or when viewed on a black and white screen"[W3C](#).

The tool will indicate that the colours pass the test if both the colour difference and the brightness difference exceed their threshold. It will indicate that it sort of passes if only one of the two values exceed their threshold. And finally, it'll fail to pass if neither value exceeds its threshold.

The tool will also indicate if the colours pass the newer [WCAG 2.0 contrast ratio formula](#). The WCAG 2.0 formula differentiates between text smaller than 18pt text larger than 18pt (or text that is bold and larger than 14pt). For AA compliance, text should have a ratio of at least 4.5:1 (larger text, at least 3:1). For AAA compliance, text should have a ratio of at least 7:1 (larger text, at least 4.5:1).

You can enter a three character value (eg: 036) and it'll automatically convert it to it's six character version.



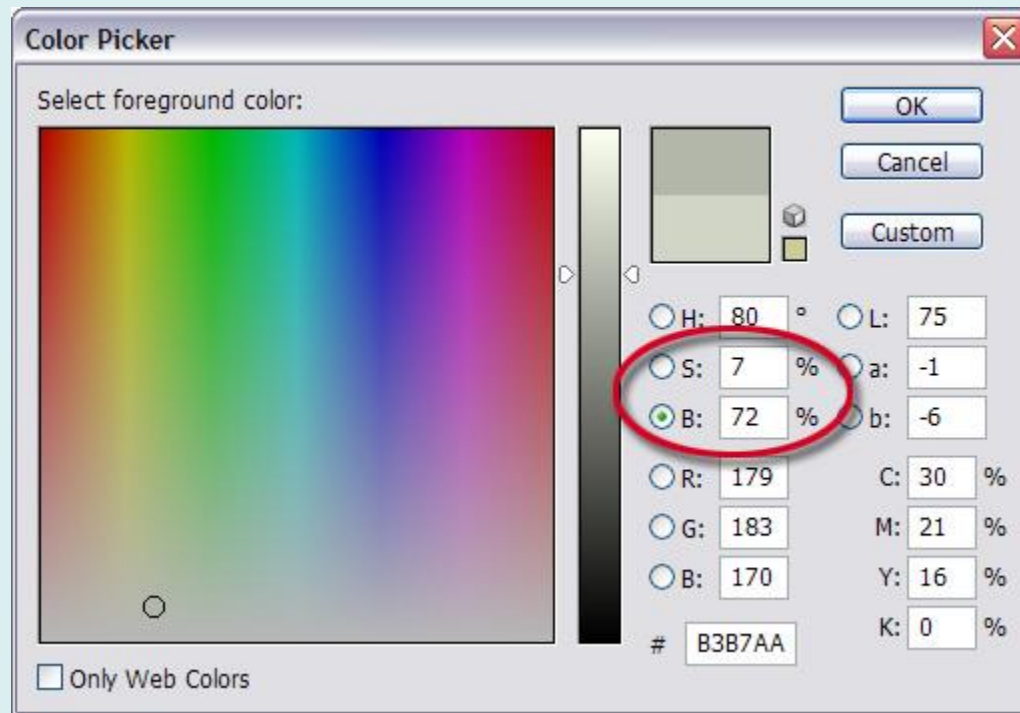
# Colour and CSS

- Colours can be specified in one of 3 ways:
  - Colour Name\*: `AliceBlue`
  - Hexadecimal Value: `#F0F8FF`
  - RGB Value: `rgb(240,248,255)`
- The hex value is most commonly used
- Hex values can be abbreviated: `#D9A` is the same as `#DD99AA`
- \*Not all colour names are W3C valid
- CSS3 brings new options for specifying colours (RGBA, HSL and HSLA)

[http://www.w3schools.com/css/css\\_colors.asp](http://www.w3schools.com/css/css_colors.asp)

[http://www.w3schools.com/css/css\\_colornames.asp](http://www.w3schools.com/css/css_colornames.asp)

# Monochromatic Schemes



Monochromatic colour schemes can be created using just the Saturation and Brightness sliders on the Photoshop colour picker.

# Colour from Images



Monochromatic text colour variant: Brightness and Saturation

# Colour from Images



Monochromatic text colour variant: Brightness

# Colour from Images



Monochromatic text colour variant: Brightness

# Colour from Images



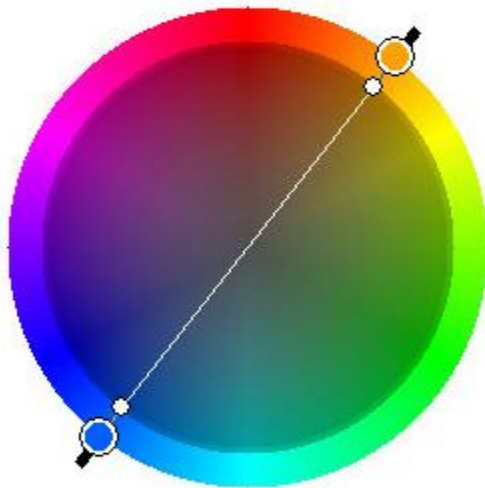
Monochromatic text colour variant: Brightness and Saturation

# Colour Theory

## What is color theory?

Color Theory is a set of principles used to create harmonious color combinations. Color relationships can be visually represented with a color wheel — the color spectrum wrapped onto a circle.

The color wheel is a visual representation of color theory:



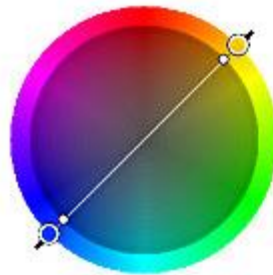
According to color theory, harmonious color combinations use any two colors opposite each other on the color wheel, any three colors equally spaced around the color wheel forming a triangle, or any four colors forming a rectangle (actually, two pairs of colors opposite each other). The harmonious color combinations are called color schemes – sometimes the term 'color harmonies' is also used. Color schemes remain harmonious regardless of the rotation angle.

# Other Colour Schemes



## Analogous Color Scheme

The **analogous color scheme** uses colors that are adjacent to each other on the color wheel. One color is used as a dominant color while others are used to enrich the scheme. The analogous scheme is similar to the monochromatic, but offers more nuances.



## Complementary Color Scheme

The **complementary color scheme** consists of two colors that are opposite each other on the color wheel. This scheme looks best when you place a warm color against a cool color, for example, red versus green-blue. This scheme is intrinsically high-contrast.

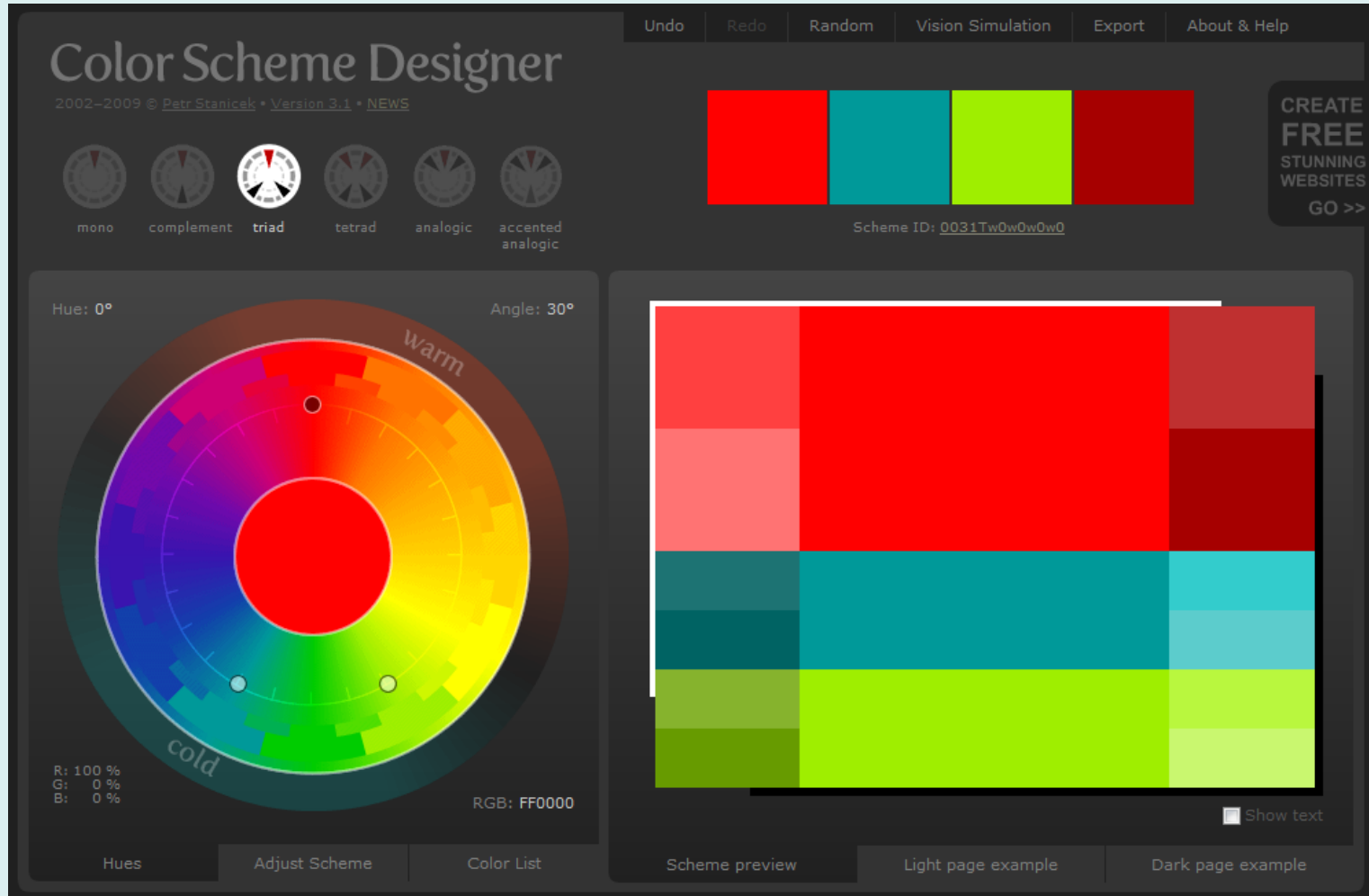


## Triadic Color Scheme

The **triadic color scheme** uses three colors equally spaced around the color wheel. This scheme is popular among artists because it offers strong visual contrast while retaining harmony and color richness. The triadic scheme is not as contrasting as the complementary scheme, but it looks more balanced and harmonious.



# Colour Scheme Tools



# Colour Scheme Tools

## Color Palette Generator

Enter the URL of an image to get a color palette that matches the image. This is useful for coming up with a website color palette that matches a key image a client wants to work with.

URL of image:



dull	vibrant
 #335500	 #33aa00
 #443311	 #886600
 #bb3311	 #ff1100
 #559922	 #77ff00
 #aaaa55	 #eedd77

If you like this color palette generator, you might like [ColorHunter.com](http://www.colorhunter.com)

THE END