Internet Programming

Lecture 01

Instructor:

Daniel Slack, P. Eng.

Applied Computer Science

University of Winnipeg

Outline

- Preliminaries
- Announcements
- Background

Preliminaries

Website

- Check the course website regularly http://courses.acs.uwinnipeg.ca/2909-001
  - Updates will be posted frequently
- Demos at http://courses.acs.uwinnipeg.ca/2909Demo

Course Textbook

- Main Texts (available online):
  - Eloquent JavaScript; 2nd Edition; Marijn Haverbeke; William Pollock, ISBN-10: 1-59327-584-6
  - JavaScript Design Patterns; Addy Osmani; O'Reilly; ISBN-10: 1-44933-181-5
- Supplementary Texts
  - Head First HTML5 Programming; Eric Freeman, Elisabeth Robson; O'Reilly, ISBN: 978-1-449-39054-9
  - P. Carey, HTML, XHTML, and Dynamic HTML, 4th Edition, Boston, Massachusetts, USA: Cengage, 2010.
  - Head First HTML with CSS & XHTML; Eric Freeman, Elisabeth Robson; O'Reilly, ISBN: 978-0-596-10197-8
  - jQuery Fundamentals; Rebecca Murphey; available at: http://jqfundamentals.com/
  - jQuery: Novice to Ninja; Earle Castledine, Craig Sharkie; available at: http://www.sitepoint.com/books/jquery1/samplechapters.php
  - Essential JavaScript & jQuery Design Patterns; Addy Osmani; available at: http://addyosmani.com/blog/essentialjsdesignpatterns/

Videos

- Douglas Crockford On Javascript
  - History of Javascript videos
  - https://www.youtube.com/playlist?list=PL7664379246A246CB
- JSConf Videos
  - Tons of videos on YouTube channel
  - Event loop - https://www.youtube.com/watch?v=8aGhZQkoFbQ

Syllabus

- The course syllabus can be downloaded from:

http://www.acs.uwinnipeg.ca/assets/CourseOutlines/Fall2016/ACS-2909-050.pdf

Lecture Format

- Lectures will consist of
  - Slides (available from the course website)
  - Problems and examples solved in class
  - Current developments in the field related to the course
    - Current developments in my field
    - Interesting technology news

Group Page for Class

- Students can post questions at the following forum

https://groups.google.com/forum/#!forum/uw_2016fall_2909

- I will answer all questions emailed to me on this forum
  - With identification removed.
- I will also post notifications
  - Assignments posted
  - Assignments marked
  - Exams marked

ACS User Accounts

- UserName: <WebAdvisorUserName>
- Password: <student number>ACS!
- Please contact Daniel Sheppard to resolve problems
- da.sheppard@uwinnipeg.ca
- Office: 3D19

Stylesheets

Applying a Style Sheet

- Three ways to apply a style to an HTML document:
  - Inline Styles
  - Embedded Style Sheet
  - External Style Sheet
- Each approach has its own advantages and disadvantages
- You will probably use some combination of these styles in developing your site

Using Inline Styles

- Style applied directly to an element using the following

<element style=“style1:value1; style2:value2; style3:value3;…”>

- style# are the names of the style properties
- value# are the values associated with each style property
- e.g. <h1 style=“text-align: center; color: red”>Sunny Acres</h1>
- Easy to interpret
  - Applied directly to the elements they affect
- Generally cumbersome
  - Have to type same commands for all your heading elements
  - Not consistent with the goal of CSS
    - Separate content from style

Using Embedded Style Sheets

- The power of style sheets becomes evident when you moving from away from content
- One approach is to collect all the styles used in the document in an embedded style sheet
  - Placed in the head section of the document
- Created using the style element

```
<style>
   style declarations
</style>
```

Using Embedded Style Sheets

- Style declarations are the declarations of the various styles to be applied to elements in the current document

```
selector {style1: value1; style2: value2; style3: value3 ... }
```

- e.g. All the h1 headings are centered and the text is red

```
<style>
   h1 {text-align: center; color: red}
</style>
```

- e.g. Apply the same style to more than one element

```
<style>
   h1, h2 {text-align: center; color: red}
</style>
```

Using External Style Sheets

- Embedded style sheet is limited to the page elements of the current document
- If you wanted the same style for whole site, the styles would have to be repeated in the head of each page
  - Cumbersome and error-prone
- Instead, you can place the style declarations in an external style sheet
  - Text file containing style declarations
- The file can then be linked to any or all pages in your site
  - Thus all pages have the same style

Using External Style Sheets

- The file name extension indicates the language
  - .css is used for CSS style sheets
- An external style sheet looks like an embedded style
  - Except the style declarations are not enclosed with <style> tags
- Previous example:

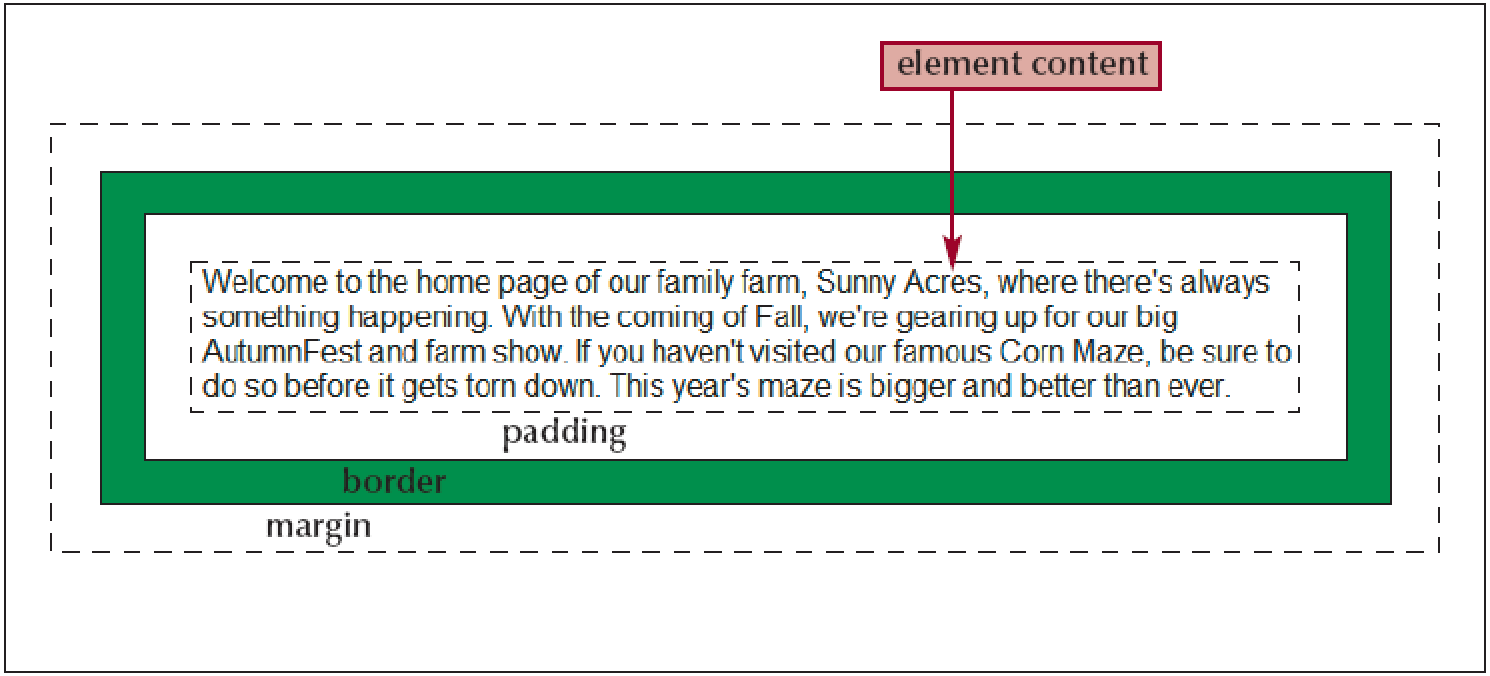
```
h1 {text-align: center; color: red}
```

- Add these style sheets to your html page using the following

```
<link href="./path/file.css" rel="stylesheet" />
```

Box Model

Working with the Box Model


element content
Welcome to the home page of our family farm, Sunny Acres, where there's always something happening. With the coming of Fall, we're gearing up for our big AutumnFest and farm show. If you haven't visited our famous Corn Maze, be sure to do so before it gets torn down. This year's maze is bigger and better than ever.
padding
border
margin

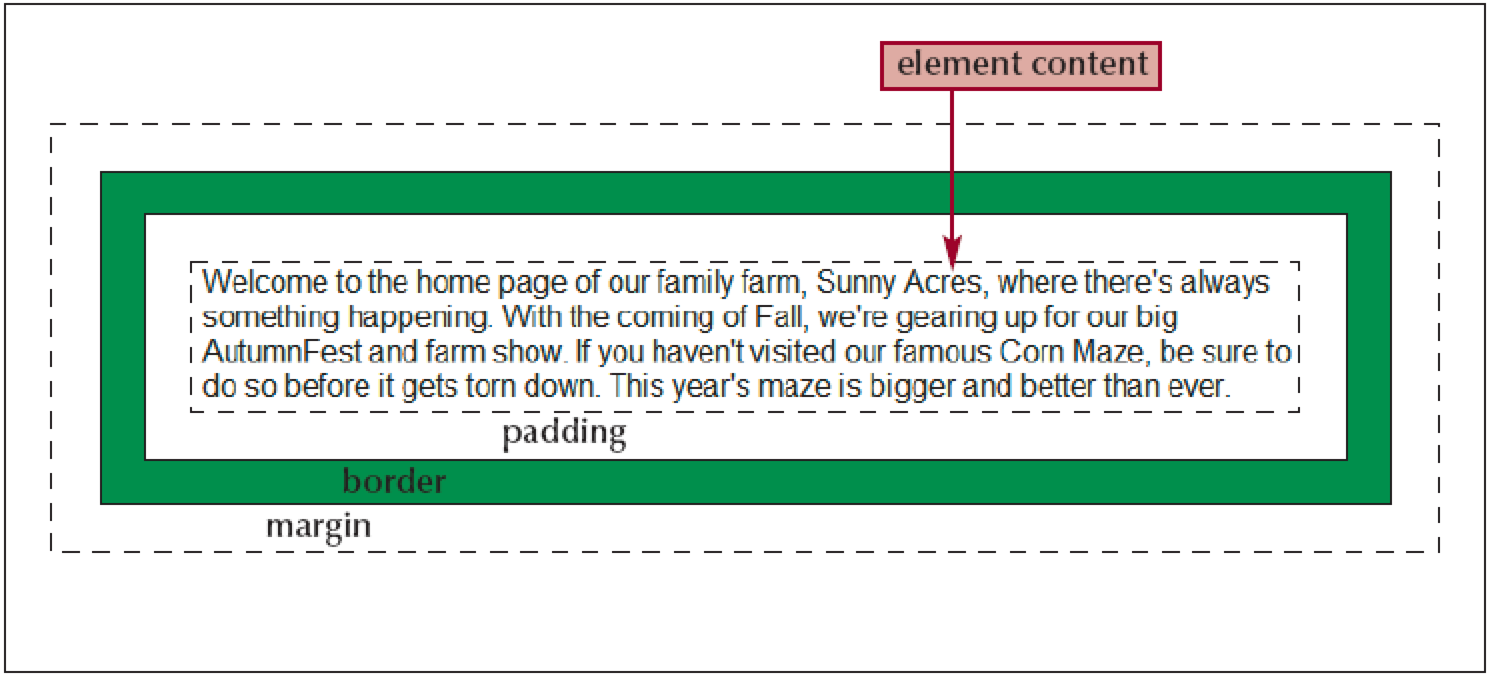
Position of Styles

- Box Model Styles have ‘positions’
  - top
  - bottom
  - left
  - right
- Applied to styles for those specific positions
- Ex.
  - margin-top
  - padding-bottom

Margin Styles

- Control your margins with the following four styles:
  - margin-top: length
  - margin-right: length
  - margin-bottom: length
  - margin-left: length
- Margin values can also be negative.
  - Creates an overlay effect
  - Browser to render one element on top of another
- You can also combine the four margin styles into a single style:
  - margin: top right bottom left

Working with the Box Model

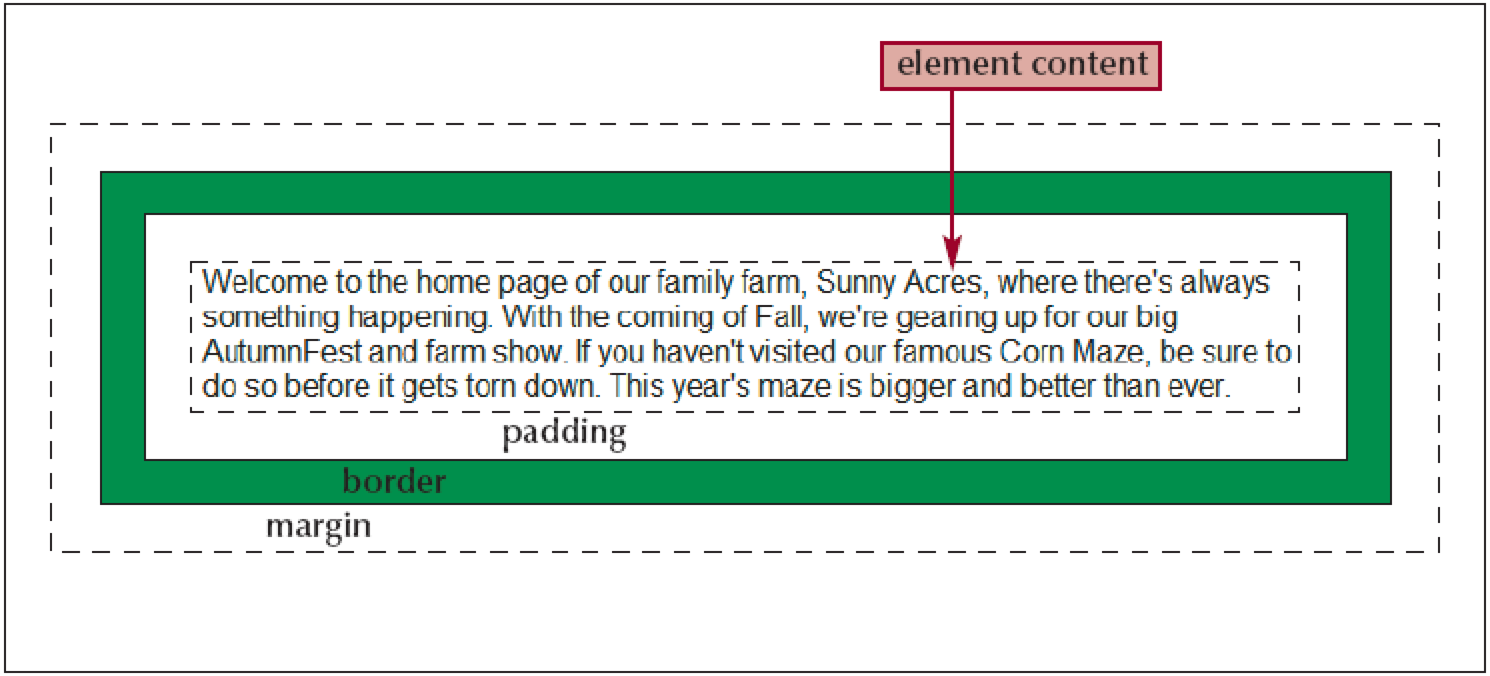


Welcome to the home page of our family farm, Sunny Acres, where there's always something happening. With the coming of Fall, we're gearing up for our big AutumnFest and farm show. If you haven't visited our famous Corn Maze, be sure to do so before it gets torn down. This year's maze is bigger and better than ever.

Padding Styles

- Styles to set padding are similar to styles to set margins:
  - padding-top: value
  - padding-right: value
  - padding-bottom: value
  - padding-left: value
- You can also combine the four margin styles into a single style:
  - padding: top right bottom left

Working with the Box Model


element content
Welcome to the home page of our family farm, Sunny Acres, where there's always something happening. With the coming of Fall, we're gearing up for our big AutumnFest and farm show. If you haven't visited our famous Corn Maze, be sure to do so before it gets torn down. This year's maze is bigger and better than ever.
padding
border
margin

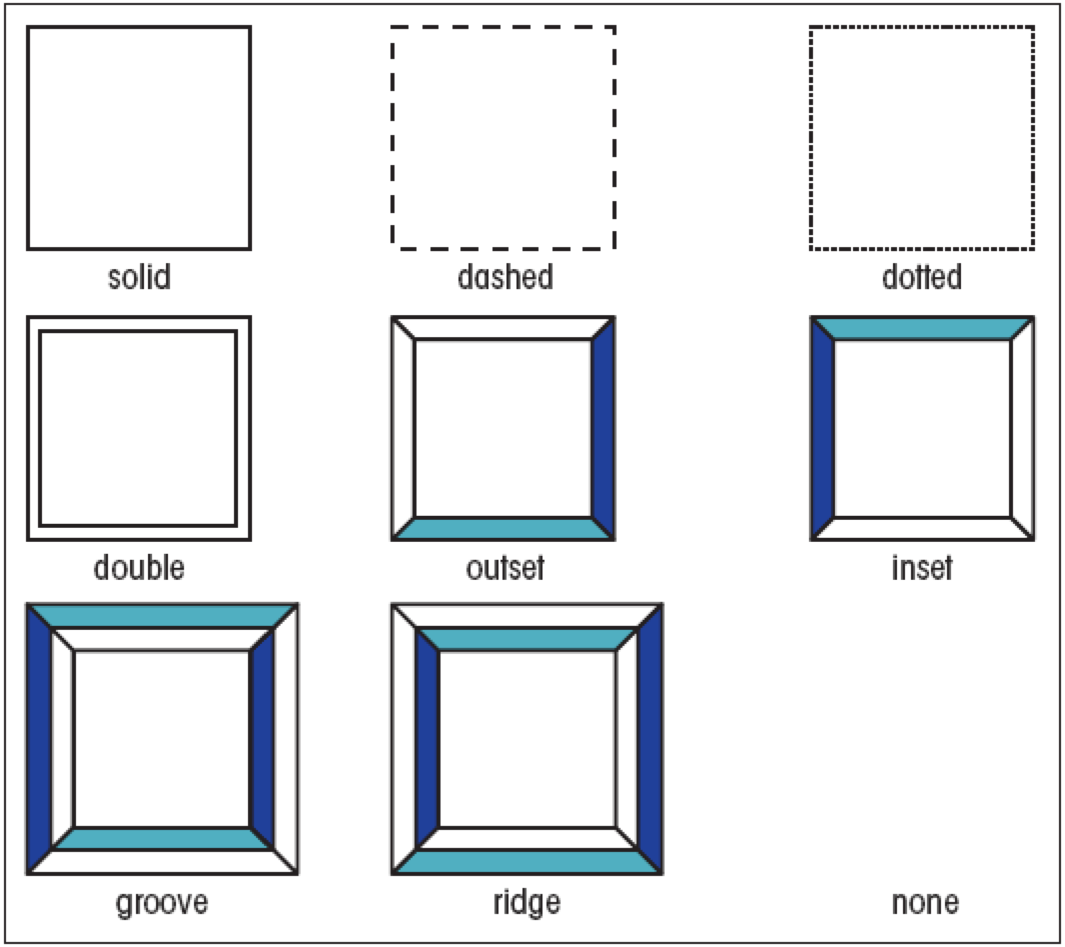
Border Styles

- border-top-width: *length*
- border-right-width: *length*
- border-bottom-width: *length*
- border-left-width: *length*
- border-width: *top right bottom left*
- border-top-color: *colour*
- border-right-color: *colour*
- border-bottom-color: *colour*
- border-left-color: *colour*
- border-color: *top right bottom left*

Border Styles

- border-top-style: *type*
- border-right-style: *type*
- border-bottom-style: *type*
- border-left-style: *type*
- border-style: *top right bottom left*
- border-radius: *radius*

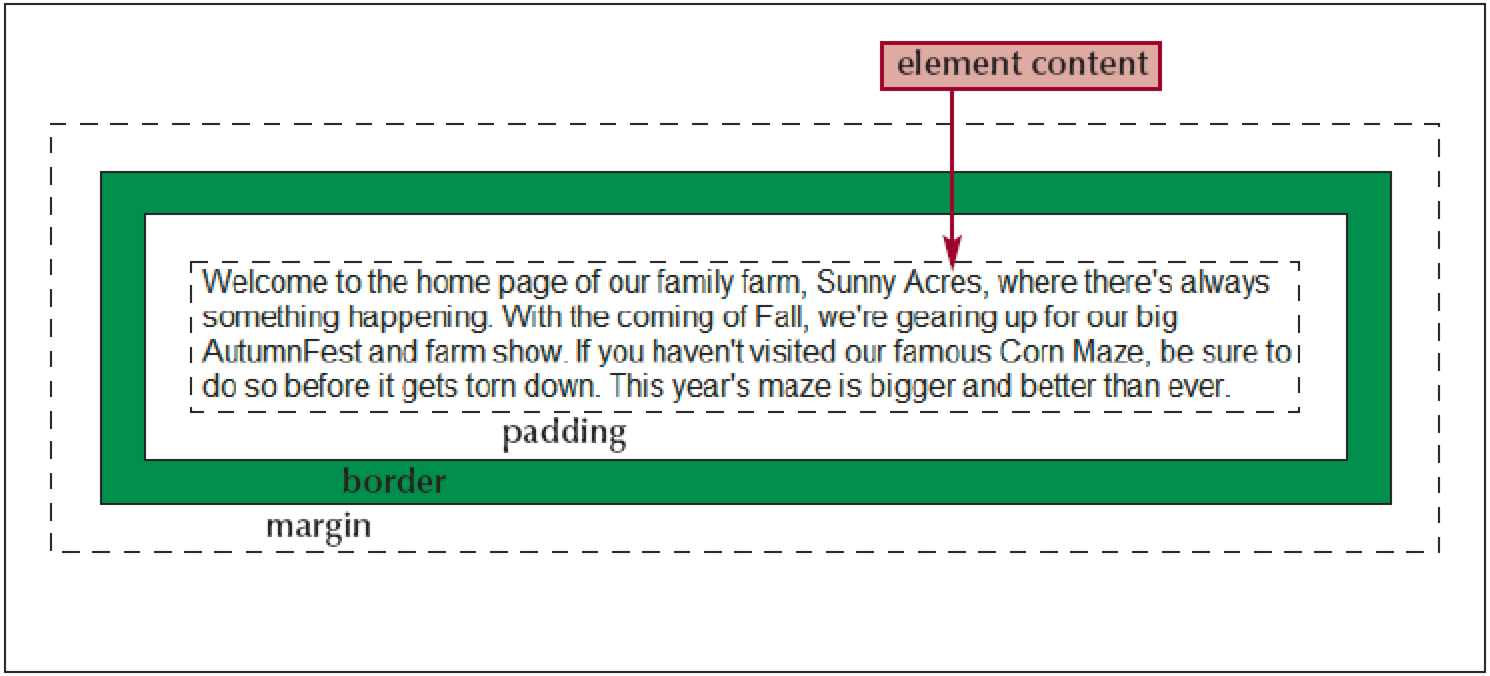
Border Styles



Border Styles

- All of the border styles can be combined into a single style
  - border-top: width style colour
  - border-right: width style colour
  - border-bottom: width style colour
  - border-left: width style colour
  - border: width style colour

Working with the Box Model


element content
Welcome to the home page of our family farm, Sunny Acres, where there's always something happening. With the coming of Fall, we're gearing up for our big AutumnFest and farm show. If you haven't visited our famous Corn Maze, be sure to do so before it gets torn down. This year's maze is bigger and better than ever.
padding
border
margin

Images

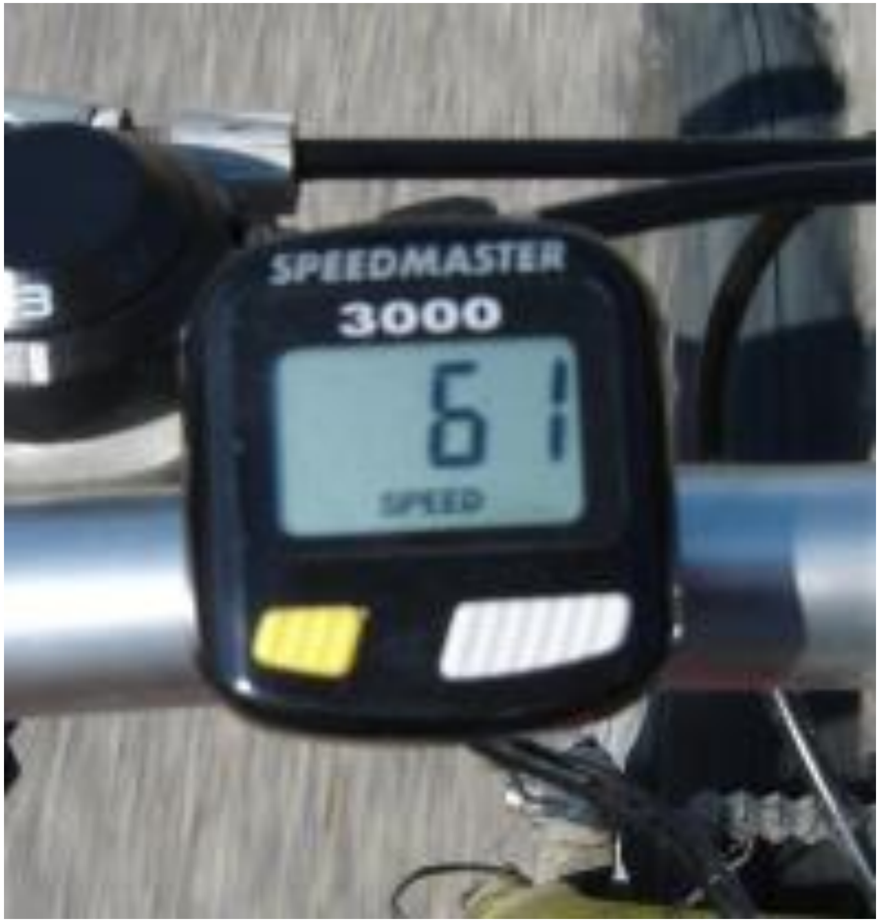
Working with Images

- Browsers support three graphic formats
  - GIF
  - JPEG
  - PNG
- Browsers may support other formats
  - Best to stick with formats above

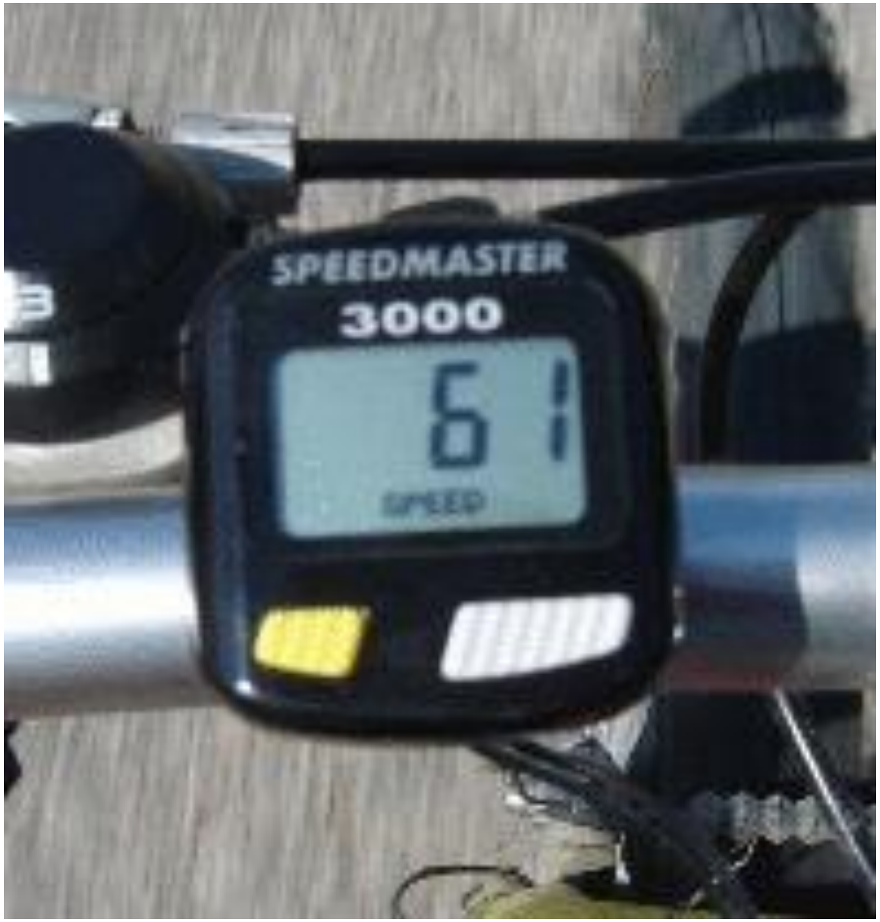
GIF Images

- Graphics Interchange Format
  - Only 256 colors
  - Clip art images, line art, logos, icons
  - Allows for transparent color
    - Used by spacers
  - Animated GIFs
  - Interlaced or non-interlaced

GIF Images



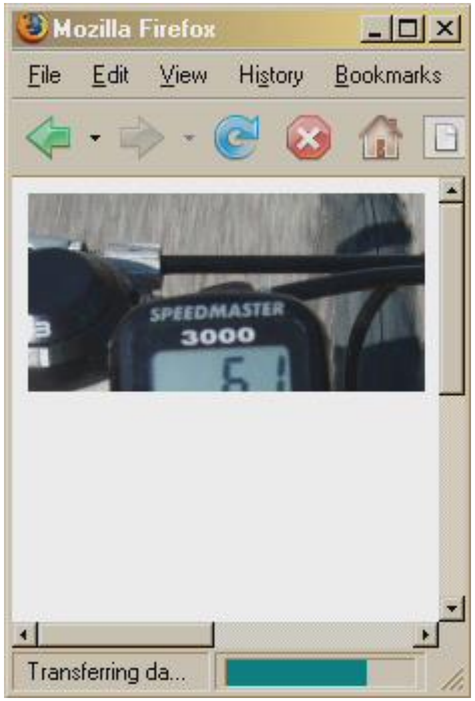
Jpeg – 22,116 colors

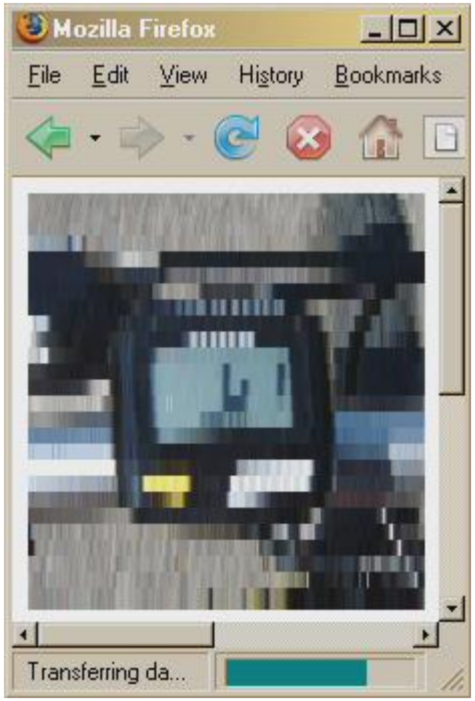


Gif – 256 colors

GIF Images

Non-interlaced

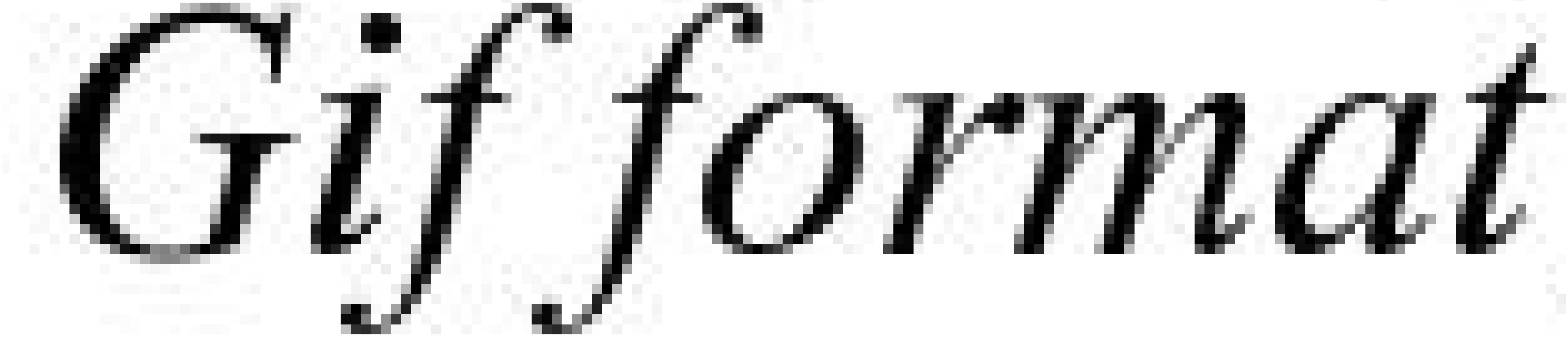




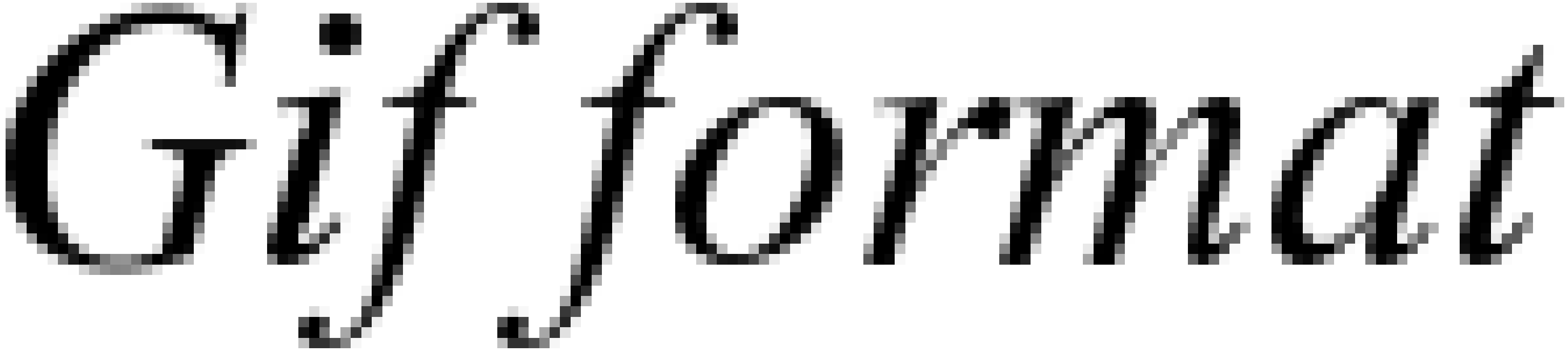
Interlaced

GIF Images

JPEG



GIF



JPEG Images

- Joint Photographic Experts Group
  - 16.7 millions of colors
  - Excellent for natural images
  - Compression rate can be adjusted
  - Progressive, nonprogressive

PNG Images

- PNG (Portable Network Graphics) has been gaining wider acceptance
- Includes:
  - Animation and transparency
  - 16.7 million colours
- Not all browsers support PNG images

Adding Image to HTML

<img src="*url*" alt="*some_text*">

Colours

Colours in HTML and CSS

- HTML/CSS are text-based language
  - Requires defining colours in textual terms
- Colours are defined in one of two ways:
  - By the colour value
  - By the colour name
- Colour value: a numerical expression that precisely describes a colour
- Provides:
  - More control
  - More choices

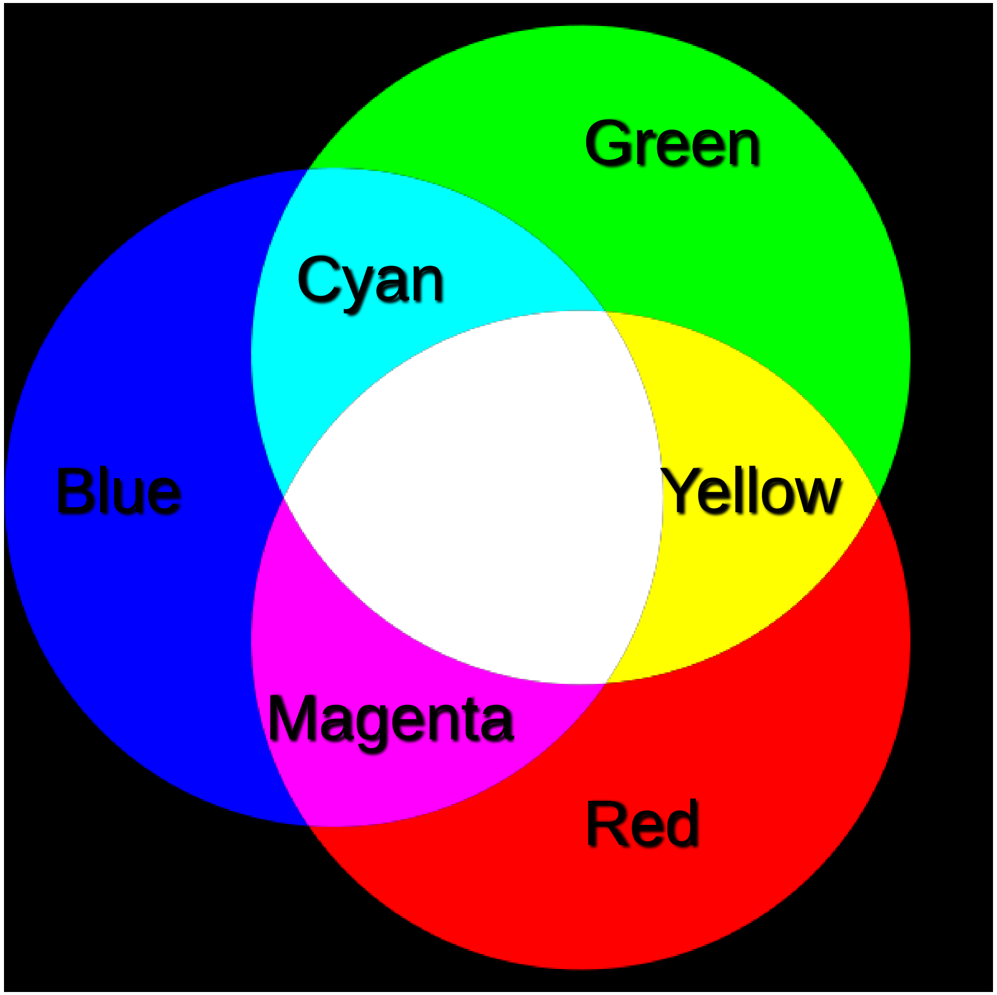
Colours in HTML and CSS

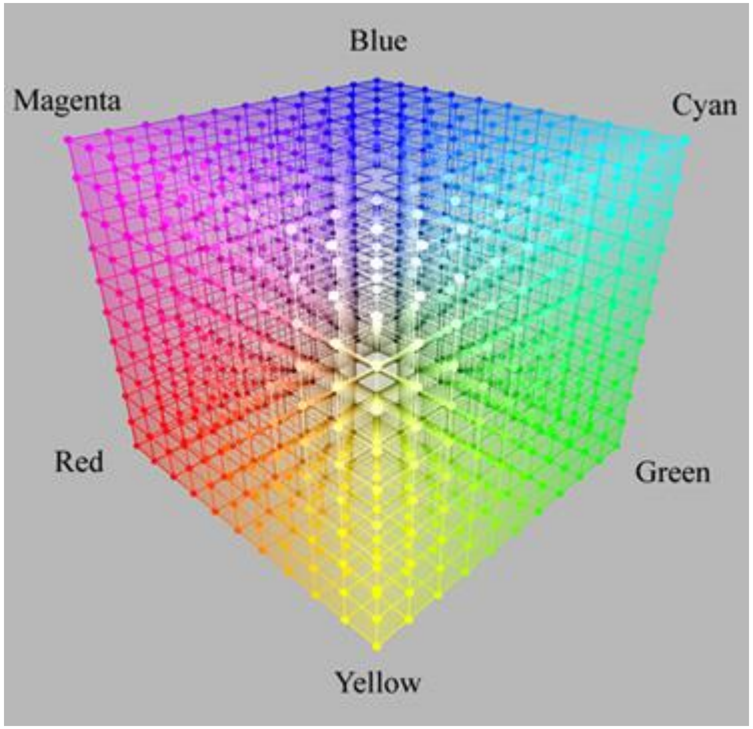
- White light is made up of three primary colours
- Any colour can be thought of as a combination of three primary colours:
  - red, green, and blue
- By varying the intensity of each primary colour, you can create almost any colour and any shade of colour
- Allows a computer monitor to combine pixels of red, green, and blue to create the array of colours you see on your screen

Colours in HTML and CSS

- Web browsers define colour mathematically
- Each colour is represented by a triplet of numbers,
  - Called an RGB triplet
- Colours are defined based on the strength of its red, green, and blue components
- The intensity of each of colour (RGB) is assigned a number from 0 (absence of colour) to 255 (highest intensity)
  - Allows for 255^3 (16.7 million) distinct colours
  - More colours than human eye can distinguish

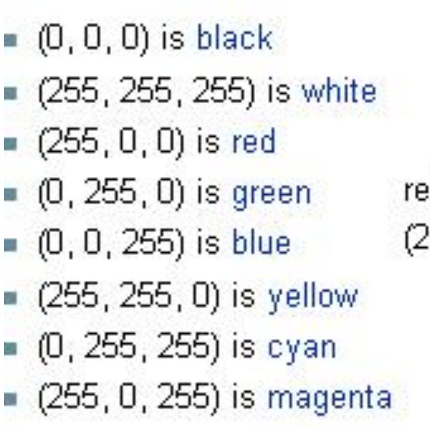
Primary colour model for light



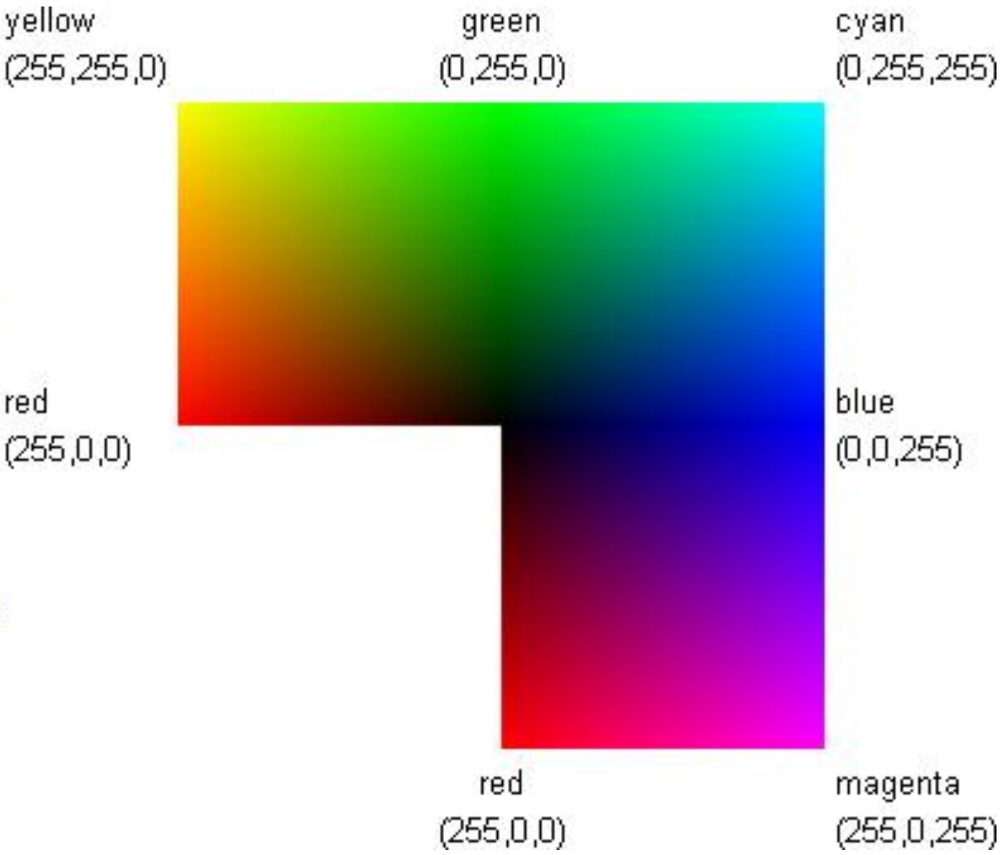


Primary Colour Model for Light

rgb (red, green, blue) (http://en.wikipedia.org/wiki/RGB_colour_model)



- (0, 0, 0) is black
- (255, 255, 255) is white
- (255, 0, 0) is red
- (0, 255, 0) is green
- (0, 0, 255) is blue
- (255, 255, 0) is yellow
- (0, 255, 255) is cyan
- (255, 0, 255) is magenta



Defining Color

You specify color using the following

- rgb(R#,G#,B#)
  - Where R#, G#, B# are in the interval [0,255]
- Or as rgb(%,%,%)
- e.g.
  - rgb(255, 166, 0) same as rgb(100%, 65%, 0%)

Hexadecimal Representation

- HTML used to require color values be entered as hexadecimals
- Hexadecimal is a number based on a base-16 numbering system
  - Rather than a base-10 numbering system that we use every day
- Base 10 counting uses a combination of 10 characters (0 through 9) to represent numerical values
- Hexadecimals include six extra characters: A (for 10), B (for 11), C (for 12), D (for 13), E (for 14), and F (for 15)

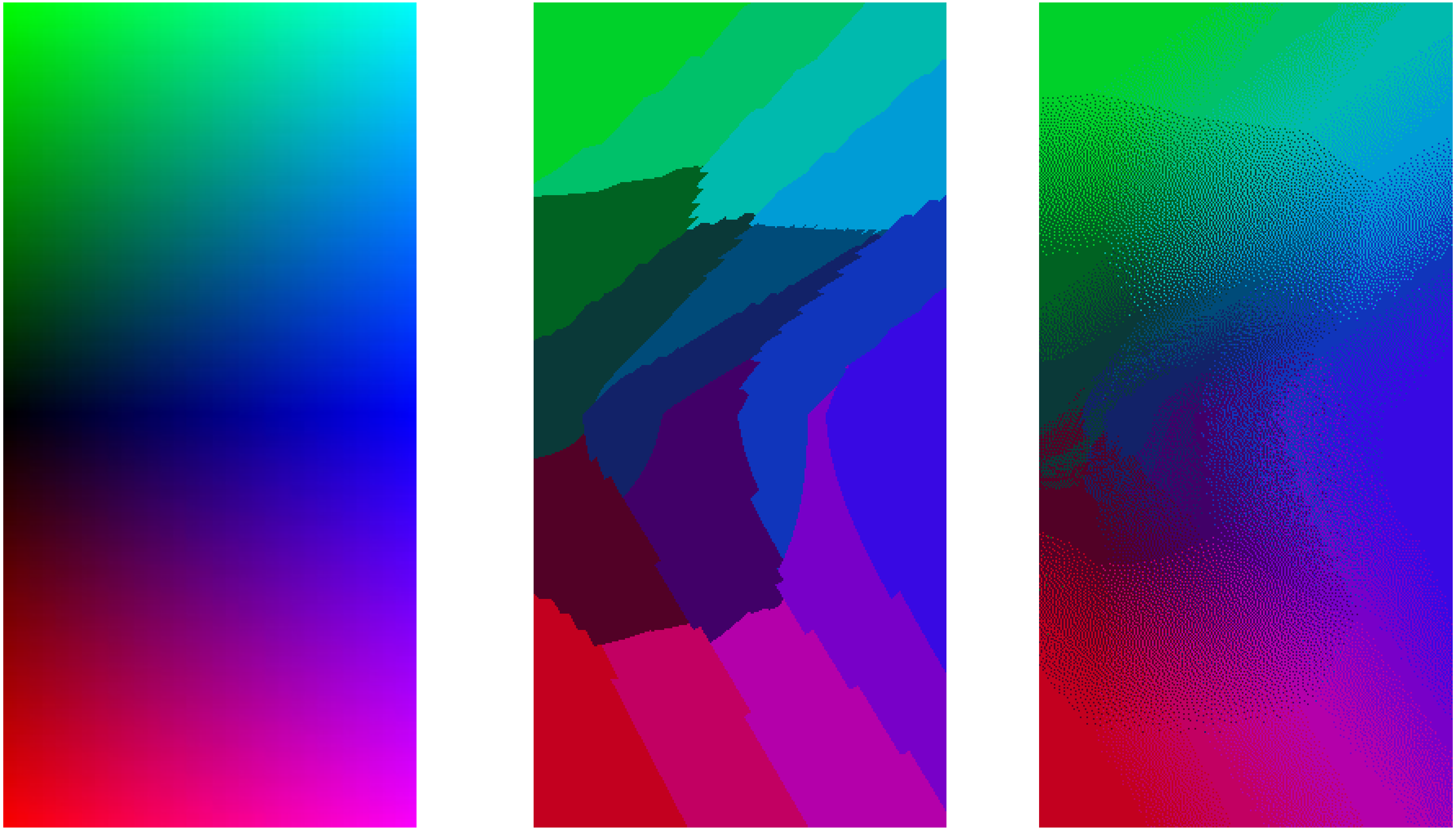
Hexadecimal Representation

- To represent a number in hexadecimal terms, you convert the value to multiples of 16 plus a remainder. For example:
  - 21 is equal to (16 x 1) + 5, so its hexadecimal representation is 15
  - The number 255 is equal to (16 x 15) + 15, or FF in hexadecimal format (remember that F = 15 in hexadecimal)
  - In the case of the number 255, the first F represents the number of times 16 goes into 255 (which is 15), and the second F represents the remainder of 15
- Once you know the RGB triplet of a color, the color needs to be converted to the hexadecimal format

Dithering

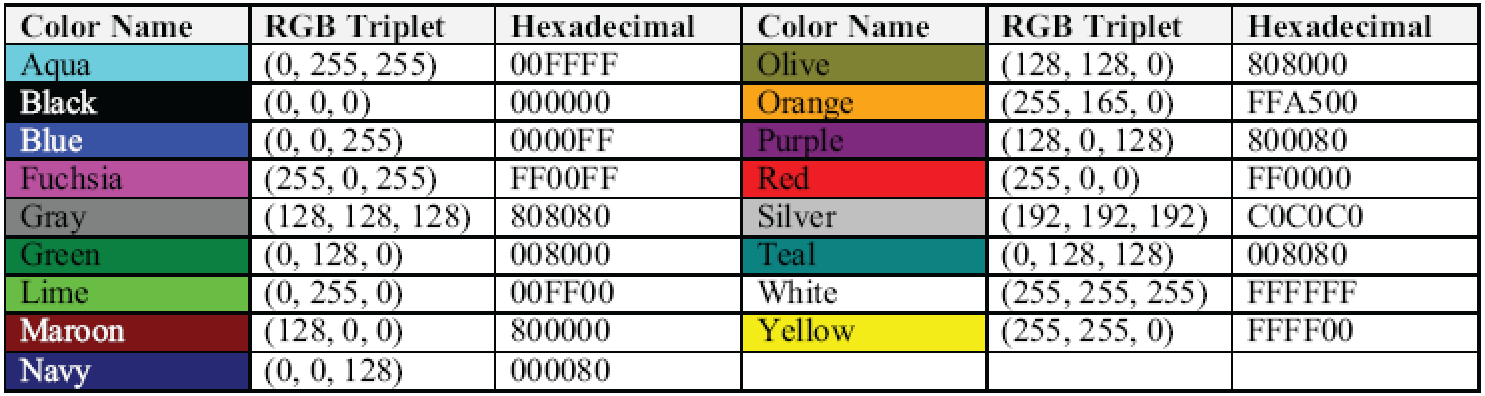
- Dithering – combining similar colours from available palette to approximate the original colour’s appearance

Dithering



Colour Names

- You can also specify colours by name
  - Allows you to accurately display them across different browsers and operating systems
- The list of only 17 colors is limiting to Web designers



Color Name	RGB Triplet	Hexadecimal	Color Name	RGB Triplet	Hexadecimal
Aqua	(0, 255, 255)	00FFFF	Olive	(128, 128, 0)	808000
Black	(0, 0, 0)	000000	Orange	(255, 165, 0)	FFA500
Blue	(0, 0, 255)	0000FF	Purple	(128, 0, 128)	800080
Fuchsia	(255, 0, 255)	FF00FF	Red	(255, 0, 0)	FF0000
Gray	(128, 128, 128)	808080	Silver	(192, 192, 192)	C0C0C0
Green	(0, 128, 0)	008000	Teal	(0, 128, 128)	008080
Lime	(0, 255, 0)	00FF00	White	(255, 255, 255)	FFFFFF
Maroon	(128, 0, 0)	800000	Yellow	(255, 255, 0)	FFFF00
Navy	(0, 0, 128)	000080			

Web Safe colours in HTML

- 216 colours supported by most browsers

http://www.w3schools.com/html/html_colors.asp

colours in HTML

colour Name	colour HEX	colour
AliceBlue	#F0F8FF	⬜
AntiqueWhite	#FAEBD7	⬜
Aqua	#00FFFF	🟦
Aquamarine	#7FFFD4	🟩
Azure	#F0FFFF	⬜
Beige	#F5F5DC	⬜
Bisque	#FFE4C4	⬜
Black	#000000	⬛
BlanchedAlmond	#FFEBCD	⬜
Blue	#0000FF	🟦
BlueViolet	#8A2BE2	🟪
Brown	#A52A2A	🟥
BurlyWood	#DEB887	🟫
CadetBlue	#5F9EA0	🟦
Chartreuse	#7FFF00	🟩
Chocolate	#D2691E	🟧
Coral	#FF7F50	🟧
CornflowerBlue	#6495ED	🟦
Cornsilk	#FFF8DC	⬜
Crimson	#DC143C	🟥
Cyan	#00FFFF	🟦
DarkBlue	#00008B	🟦
DarkCyan	#008B8B	🟦

colour Name	colour HEX	colour
DarkGoldenRod	#B8860B	🟨
DarkGray	#A9A9A9	⬜
DarkGrey	#A9A9A9	⬜
DarkGreen	#006400	🟩
DarkKhaki	#BDB76B	🟨
DarkMagenta	#8B008B	🟪
DarkOliveGreen	#556B2F	🟩
Darkorange	#FF8C00	🟧
DarkOrchid	#9932CC	🟪
DarkRed	#8B0000	🟥
DarkSalmon	#E9967A	🟧
DarkSeaGreen	#8FBC8F	🟩
DarkSlateBlue	#483D8B	🟪
DarkSlateGray	#2F4F4F	⬛
DarkSlateGrey	#2F4F4F	⬛
DarkTurquoise	#00CED1	🟦
DarkViolet	#9400D3	🟪
DeepPink	#FF1493	🟥
DeepSkyBlue	#00BFFF	🟦
DimGray	#696969	⬛
DimGrey	#696969	⬛
DodgerBlue	#1E90FF	🟦
FireBrick	#B22222	🟥

colours in HTML

colour Name	colour HEX	colour
FloralWhite	#FFFAF0	⬜
ForestGreen	#228B22	🟩
Fuchsia	#FF00FF	🟪
Gainsboro	#DCDCDC	⬜
GhostWhite	#F8F8FF	⬜
Gold	#FFD700	🟨
GoldenRod	#DAA520	🟨
Gray	#808080	⬛
Grey	#808080	⬛
Green	#008000	🟩
GreenYellow	#ADFF2F	🟩
HoneyDew	#F0FFF0	⬜
HotPink	#FF69B4	🟥
IndianRed	#CD5C5C	🟥
Indigo	#4B0082	🟪
Ivory	#FFFFF0	⬜
Khaki	#F0E68C	🟨
Lavender	#E6E6FA	⬜
LavenderBlush	#FFF0F5	⬜
LawnGreen	#7CFC00	🟩
LemonChiffon	#FFFACD	🟨
LightBlue	#ADD8E6	🟦
LightCoral	#F08080	🟥

colour Name	colour HEX	colour
LightCyan	#E0FFFF	🟦
LightGoldenRodYellow	#FAFAD2	🟨
LightGray	#D3D3D3	⬜
LightGrey	#D3D3D3	⬜
LightGreen	#90EE90	🟩
LightPink	#FFB6C1	🟥
LightSalmon	#FFA07A	🟧
LightSeaGreen	#20B2AA	🟦
LightSkyBlue	#87CEFA	🟦
LightSlateGray	#778899	⬛
LightSlateGrey	#778899	⬛
LightSteelBlue	#B0C4DE	🟦
LightYellow	#FFFFE0	🟨
Lime	#00FF00	🟩
LimeGreen	#32CD32	🟩
Linen	#FAF0E6	⬜
Magenta	#FF00FF	🟪
Maroon	#800000	🟫
MediumAquaMarine	#66CDAA	🟩
MediumBlue	#0000CD	🟦
MediumOrchid	#BA55D3	🟪
MediumPurple	#9370D8	🟪
MediumSeaGreen	#3CB371	🟩

colours in HTML

colour Name	colour HEX	colour
MediumSlateBlue	#7B68EE	🟪
MediumSpringGreen	#00FA9A	🟩
MediumTurquoise	#48D1CC	🟦
MediumVioletRed	#C71585	🟪
MidnightBlue	#191970	🟦
MintCream	#F5FFFA	⬜
MistyRose	#FFE4E1	⬜
Moccasin	#FFE4B5	🟨
NavajoWhite	#FFDEAD	🟨
Navy	#000080	🟦
OldLace	#FDF5E6	⬜
Olive	#808000	🟩
OliveDrab	#6B8E23	🟩
Orange	#FFA500	🟧
OrangeRed	#FF4500	🟧
Orchid	#DA70D6	🟪
PaleGoldenRod	#EEE8AA	🟨
PaleGreen	#98FB98	🟩
PaleTurquoise	#AFEEEE	🟦
PaleVioletRed	#D87093	🟥
PapayaWhip	#FFEFD5	⬜
PeachPuff	#FFDAB9	🟧
Peru	#CD853F	🟫

colour Name	colour HEX	colour
Pink	#FFC0CB	🟥
Plum	#DDA0DD	🟪
PowderBlue	#B0E0E6	🟦
Purple	#800080	🟪
Red	#FF0000	🟥
RosyBrown	#BC8F8F	🟫
RoyalBlue	#4169E1	🟦
SaddleBrown	#8B4513	🟫
Salmon	#FA8072	🟧
SandyBrown	#F4A460	🟧
SeaGreen	#2E8B57	🟩
SeaShell	#FFF5EE	⬜
Sienna	#A0522D	🟫
Silver	#C0C0C0	⬜
SkyBlue	#87CEEB	🟦
SlateBlue	#6A5ACD	🟪
SlateGray	#708090	⬛
SlateGrey	#708090	⬛
Snow	#FFFAFA	⬜
SpringGreen	#00FF7F	🟩
SteelBlue	#4682B4	🟦
Tan	#D2B48C	🟫
Teal	#008080	🟦

colours in HTML

colour Name	colour HEX	colour
Thistle	#D8BFD8	🟪
Tomato	#FF6347	🟥
Turquoise	#40E0D0	🟦
Violet	#EE82EE	🟪
Wheat	#F5DEB3	🟨
White	#FFFFFF	⬜
WhiteSmoke	#F5F5F5	⬜
Yellow	#FFFF00	🟨
YellowGreen	#9ACD32	🟩

Positioning

Positioning Objects with CSS

- To position an element at specific position on the page:
  position: *type; top: value; right: value; bottom: value; left: value;*
- Usually, only left and top coordinates are specified
- Type:
  - static, absolute, relative, fixed, and inherit
- Default is static
  - Same behaviour as not using positioning

Absolute Positioning

- Enables you to place an element at specific coordinates
  - Either on a page or within a containing element

position: absolute; left: 100px; top: 50px

- Takes an element out of normal flow
  - Any subsequent content flows into space it occupied

Relative Positioning

- Used to move an element relative to its default position on the page
  position: relative; left: 100px; top: 50px
- Does not affect the position of other elements on a page
  - Retain their original position as if element never moved

Fixed and Inherited Positioning

- Elements placed with previous methods scroll with the rest of the document
- An element can be fixed
  - Stay at a specific spot while the rest of the page scrolls
- You can also assign the inherit position style to an element so that it inherits the position value of its parent element

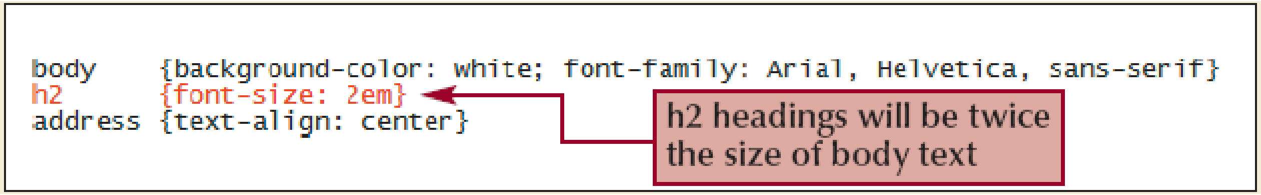
Units

Absolute Units

- Define a font size using one of five standard units of measurement:
  - Pixel (default)
  - Millimeters (mm)
  - Centimeters (cm)
  - Inches (in)
  - Points (pt)
  - Picas (pc)
- Fixed size regardless of the device rendering the page
- Usually used when you know the properties of the output device
  - e.g. Print design

Relative Units

- Many Web page designers opt to use relative units
- Expressed relative to the size of other objects within the Web page
- Allows your page to be scalable
  - Em unit
  - Percentages
  - Relative keywords
    - Larger
    - Smaller



```
body    {background-color: white; font-family: Arial, Helvetica, sans-serif}
h2      {font-size: 2em}
address {text-align: center}
```

h2 headings will be twice the size of body text

em Unit

- Specifies size relative to the parent element
- e.g. h1 {font-size: 2em}
  - Parent element: Web page body
  - Sets the font size of h1 headings to twice the font size of body text
- Context is important when interpreting the effect of the em unit

Percentage

- Percentages are also based on font size of parent
- e.g. h1 {font-size: 200%}
  - Size of h1 is 200% or twice that of body text

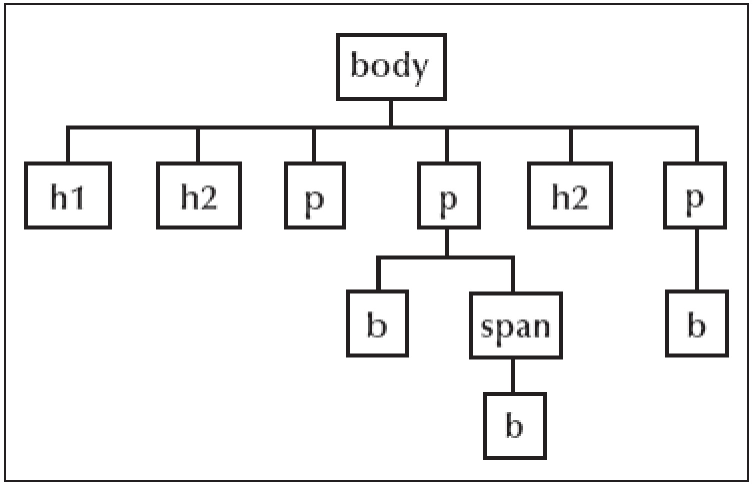
Relative Keywords

- Express font sizes using 7 keywords:
  - xx-small, x-small, small, medium, large, x-large, xx-large
- Exact size is specified by browsers internal style sheet

Selectors

Working with Selector Patterns

- Elements are nested within other elements, forming a hierarchical tree structure



Working with Selector Patterns

- CSS allows you to create contextual selectors that express the location of an element within the hierarchy of elements

  parent descendant {styles}

- e.g. li b {color: blue}
  - Applies blue colour only to boldface text found in lists
  - li is the parent, b is the descendant
  - Any bold element not nested within a list is not affected
  - Descendant element does not have to be a direct descendant

Working with Selector Patterns

- Contextual selectors can be grouped with other selectors
  li b, h2 {color: blue}
- Can be applied with elements marked with an id
  #notes b {color: blue}
- Can be used to apply same colour to all elements
  * {color: blue}
- Only colours blue, bold text that is a direct descendent of a paragraph
  p > b {color: blue}
- See pp. 201-203 for more detail

Using Selector Patterns

- To apply a style to all elements in the document, use the * selector
- To apply a style to a single element, use the e selector, where e is the name of the element
- To apply a selector to a descendant element, f, use the e f selector, where e is the name of the parent element and f is an element nested within the parent
- To apply a selector to a child element, f, use the e > f selector, where e is the name of a parent element and f is an element that is a direct child of the parent
- To apply a selector to a sibling element, use the e + f selector, where e and f are siblings and f immediately follows e in the document tree

Working with Selector Patterns

- □ On occasion you might also need to select elements based on their attribute values

  element[att] {styles}
- □ Element is page element, and att is attribute
- □ e.g. a[rel] {color: blue}
- □ Applies blue font to only link elements with a rel attribute
- □ e.g. a[href="gloss.html"] {color: blue}
- □ Applies blue font to only link elements where href attribute equals gloss.html

Working with Classes

- □ Used when you want to identify elements that share a common characteristic
  <*elem class*="*class*"> … </*elem*>
- □ You can use the class attribute to assign the same style to multiple elements sharing the same class value
  .*class* {*styles*}

Specificity

- Styles have a weight
- (a * 100) + (b * 10) + (c)
  - a is id selector
  - b is class selector
  - c is element and attribute selector

Ex.

#notes b : 101

h1 em.tiny : 12