

# CSS (Cascading Style Sheet)

(Properties , Selector , Functions )

# What is CSS?

- ⊕ Cascading Style Sheets (CSS) - a simple mechanism for adding style (e.g. fonts, colors, spacing) to Web documents.

# CSS allows you to:

- ⊕ achieve a new level of fine-grain control over Web design
- ⊕ easily apply the same consistent look and feel to any number of pages
- ⊕ control the layout of your pages down to the pixel

# What are Style Sheets?

- ⊕ Style Sheet also known as Cascading Style Sheets or CSS can be defined in an external file attached in the HTML document.
- ⊕ Style sheets describe how documents are presented on screens, in print, or perhaps how they are pronounced.

# What can you specify with CSS

Character format

Font, size, weight, colour, case, underlines

Paragraph format

Letter/word spacing, line height, alignment, justification, margins and indentation

## Page Element Positioning

Absolute and relative positioning/layering of  
all document objects

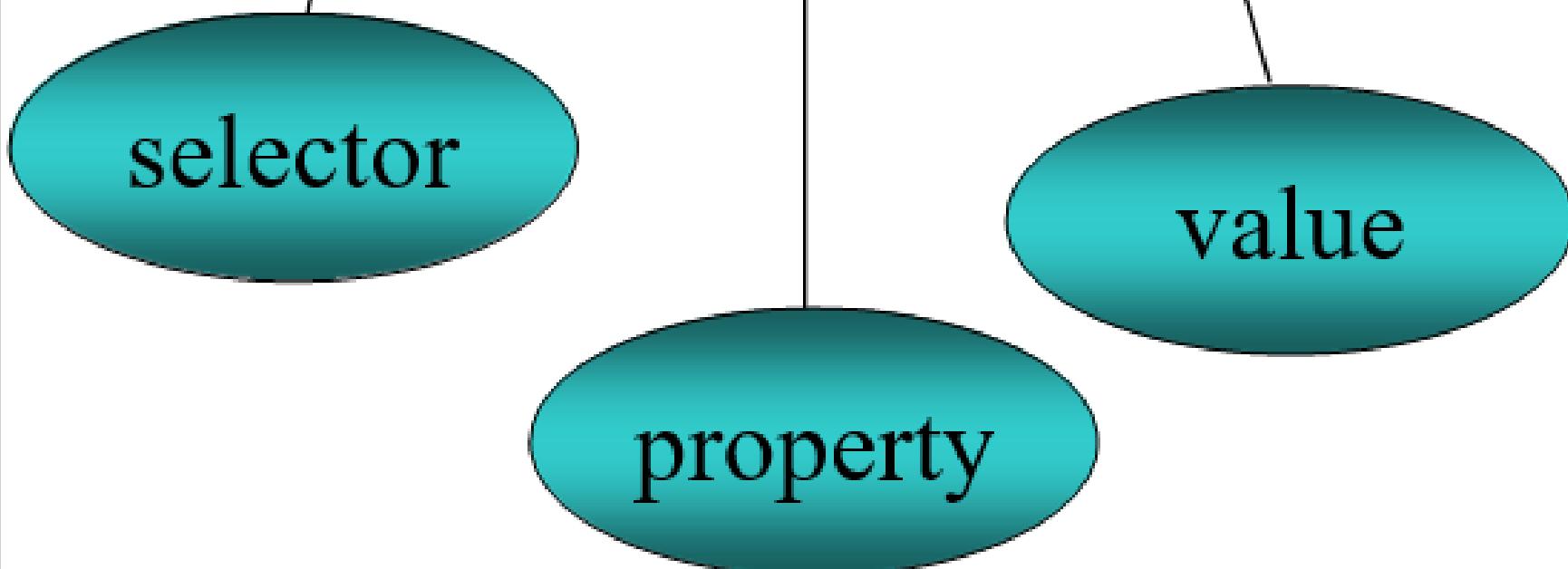
Background colours/images for everything, automatic  
rollovers and filters.

# Style Sheet Components

CSS are a series of RULES composed of

- ⊕ SELECTOR
- ⊕ PROPERTY
- ⊕ VALUE

```
H3 {font-family: Arial}
```



# Selectors and Properties

SELECTOR is a string that identifies what elements the corresponding rule is applied to.

H1, H2, H3 {property: value;}

OR

H1 {property: value;} H2 {property: value;} H3 {property: value;}

CSS rule will be ignored if you get the selector property relationship wrong.

Property is an identifier.

Proper selector property association:

H1, H2, H3 {property: value;}

# Common Properties

Property	Description
background	sets the background color for the text
color	sets the color for the text
font-family	sets the font type for the text
font-size	sets the size for the font and can be measured in points or a percentage of another tag, or xx-small to xx-large
font-style	normal or italic – normal is the default
font-weight	extra-light to extra-bold
text-align	left, right, center, or justify
text-indent	indent from margins and is measured as a fixed size or percentage
text-decoration	none, underline, overline, strikethrough

# Values

- ⊕ Units of Measure

## Length Units

- ⊕ Integer or a decimal fraction , can be preceded by + or -.
- ⊕ Can be absolute or relative:

## Relative length units

-It specifies a length relative to another length property.

em: the size of the font

px: pixel is a relative-length measurement

ex: the “x-height” of the font, usually set to the height of a lowercase x

## Percentages

E.g. margin-left: 10%

## Absolute-length measurements

-Absolute length units are only useful when the physical properties of the output medium are known.

in: inches

cm: centimeters

mm: millimeters

pt: points (1 pt = 1/72 in)

pc: picas (1 pc = 12 pt)

# Percentage Units

It is a number specified in <percentage>  
immediately followed with “%” symbol.

```
P { line-height: 120% }  
/* 120% of the element's 'font-size' */
```

# Color Units

A color is either a keyword or a numerical RGB specification. The RGB color model is being used in numerical color specifications.

Examples below specify same color:

```
EM { color: #f00 }
```

```
/* #rgb */
```

```
EM { color: #ff0000 }
```

```
/*#rrggbb */
```

```
EM { color: rgb(255,0,0) }
```

```
/* integer range 0 - 255 */
```

```
EM { color: rgb(100%, 0%, 0%) }
```

```
/* float range 0.0% - 100.0% */
```

# URL Value

A Uniform Resource Locator (URL) is identified with a functional notation:

```
BODY {  
background:url(http://www.bg.com/ pinkish.gif) }
```

Parentheses, commas, whitespace characters, single quotes ('') and double quotes ("") appearing in a URL must be escaped with a backslash: '\(','\"','\\':

There are different ways to add style information to your web page.

These are as follows:

- ⊕ Internal style sheets
- ⊕ Embedded style sheets
- ⊕ Linked style sheets

# Inline Style

Inline style is done by adding STYLE attribute to an HTML element.

Syntax:

```
<ELEMENT STYLE="property:value;  
property:value; ..."></ELEMENT>
```

Example:

```
<B STYLE="color:navy;">In the navy.</B>
```

Individual element's style declared using the STYLE attribute.

Each CSS property is followed by a colon and the value of that attribute.

Multiple properties are separated by semicolons.

Inline styles override any other styles

# Embedded Style Sheet

To use an embedded style sheet, you define a style block between <STYLE> and </STYLE> tags, placed in the HEAD element of the HTML document.

This block consists of a set of style rules, where each rule defines a style for an HTML element or group of elements.

Begins with <STYLE TYPE = “text/css”>

Styles placed here are applied to the whole document.

TYPE attribute specifies the MIME type.

MIME is a standard for specifying the format of content.

Other MIME types include text/html, image/gif and  
text/javascript

# Class

CLASS is an HTML attribute that has no display characteristics and can be applied to any element.

To apply a CLASS to a group of elements, set the CLASS attribute for all that apply.

```
<H2 CLASS="classname"> Sales and Marketing</H2>
```

```
<P CLASS="classname">This is normal text.</P>
```

# ID

ID is an HTML attribute that does not affect the display of an element and can be applied to any element.

CLASS names are usually given to groups of element instances sharing some common function or format (relative importance, context, and so on).ID is used to identify one specific instance of an element.

# FONT Properties

Font properties is the most common uses of style sheets.

CSS specifically defines the following font properties:  
font-family, font-style, font-variant, font-weight, font-size, font

Font properties apply to all elements and are inherited.

font-family values are: family-name, generic-family

font-style values are: normal, italic, oblique

font-variant values are: normal ,small-caps

# Font weight

Value	Definition
normal	Default – normal characters
bold	Thick characters
bolder	Thicker characters
lighter	Lighter characters
100	
200	
300	
400	Defines thin to thick characters, however, 400 is same with normal value while 700 is same with bold value
500	
600	
700	
800	
900	

# Font Size

Value	Definition
xx-small	
x-small	
small	Sets the size of the font to different sizes, from xx-small to xx-large. Default value: medium
medium	
large	
x-large	
xx-large	
smaller	Sets the font-size to a smaller size than the parent element
larger	Sets the font-size to a larger size than the parent element
length	Sets the font-size to a fixed size
%	Sets the font-size to a % of the parent element

# Text Properties

## Word-spacing

defines whitespaces between words.

```
H1 { word-spacing: 1em } P { word-spacing: 30px }
```

## Letter-spacing

```
BLOCKQUOTE {letter-spacing:0.1em }
```

```
BLOCKQUOTE { letter-spacing: 0 }
```

```
BLOCKQUOTE { letter-spacing: 0cm }
```

## **Text-decoration**

A:link, A:visited, A:active { text-decoration: underline }

## **Color**

BODY {color:blue} H1 {color:#00ff00}

H2 {color:rgb(255,0,0)}

## Vertical-align

values are: baseline, middle, sub, super,  
text-top, text-bottom, bottom, top.

IMG { vertical-align: bottom }

## Text-align

values are: left, right, justify.

H1 { text-align:center } P { text-align:right } P { text-align:justify }

## Text-transform

values are: capitalize, uppercase,  
lowercase, none (initial value).

H1 { text-transform: uppercase }

## Text-indent

if identified with a negative value, first line is  
indented to the left.

P { text-indent: 3em }

# Line-height

sets the distance between lines.

P

P {

{ line-height:

line-height: 1.4 }

14pt }

P { line-height: 140% }

# Background Properties

## Background-color

values can be in name, rgb, hex values,  
transparent.

```
H1 { background-color: #F00 } BODY { background-color:yellow; }  
H1 { background-color:#00ff00; } P { background-  
color:rgb(255,0,255); }
```

# Background-image

```
BODY { background-image: url(marble.gif) }
```

```
P { background-image: none }
```

# Background-repeat

determines how/if the image is repeated –

repeat-x, repeat-y, no-repeat.

```
BODY { background: red url(pendant.gif); background- repeat: repeat-y; }
```

# Background-attachment

values are scroll and fixed.

```
BODY {  
background: red url(pendant.gif);  
background-repeat: repeat-y;  
background-attachment: fixed;  
}
```

# Background Position

keyword	Percentage value
Top left, left top	0% 0%
Top center, center top	50% 0 %
Right top, top right	100% 0%
Left center, center left	0% 50%
Right center, center right	100% 50%
Bottom left, left bottom	0% 100%
Bottom center, center bottom	50% 100%
Bottom right, right bottom	100% 100%

# Background

Sets all individual background properties in one declaration.

background-color background-image background-repeat  
background-attachment background-position

BODY { background: red }

P { background: url(chess.png) gray

50% repeat fixed }

Two kinds of ID selector:

1. Solitary ID Selector
2. ID Selector

ID selectors are used for special formatting of only a few elements.

use a# instead of a.(.)

ID Selectors can also be element-less.

# When to use a CLASS and an ID?

IDs must be unique on the page – use this  
for a specific ID on a single element.

CLASSes can be used multiple times.

Both attributes can be used with any HTML  
element.

CLASS is preceded by a dot (.) ID with a hash mark (#)

# Linked Style Sheets

The easiest method for adding style to multiple HTML documents .  
External linking separate pages can use all same style sheet.  
only modify a single file to change styles  
across your site.

example of a Cascading Style Sheet

```
H2 {  
color: fuchsia;  
text-decoration: line-through;  
}
```

## LINK element

Specifies a relationship (REL attribute) between current document and another document.

```
<LINK REL = "stylesheet" TYPE = "text/css" HREF = "styles.css">
```

LINK element can only be placed in header.

# Imported Style Sheet

@import can be used in conjunction with the other methods.

# Cascading

"Cascading" in CSS specifies how an element instance may properly be affected by styles of different types and selected in a number of ways (element tag, class, ID).

inline style

embedded style sheet

linked style sheet

imported style sheet

Styles are applied in the following order:

1. Browser default
2. External (linked) style sheet
3. Embedded style sheet (within the <head> tag).
4. Inline style (inside the HTML element)

Inline style has the highest priority and will override the others.

The order of specificity, from greatest to least,  
is:

1. inline styles
2. ID
3. CLASS
4. HTML element

# Box Properties

The box properties sets the size, circumference and position of the boxes that represent elements.

MARGIN properties

PADDING properties

BORDER properties

WIDTH and HEIGHT properties

FLOAT and CLEAR properties

# Margin

MARGIN summarizes the four individual property  
margin-top margin-right margin-bottom margin-left

```
BODY { margin: 2em }
```

```
/* all margins set to 2em */
```

```
BODY { margin: 1em 2em }
```

```
/* top & bottom = 1em, right & left = 2em */
```

```
BODY { margin: 1em 2em 3em }
```

```
/* top=1em, right=2em, bottom=3em, left=2em
```

```
*/
```

# Padding

a shorthand property for:

padding-top

padding-right

padding-bottom

padding-left

BACKGROUND property sets the surface of the padding.

```
H1 { background: white; padding: 1em
```

```
2em; }
```

# Border

sets style and color of an element's border:

border-width border-style border-color

```
P { border: medium double  
rgb(250,0,255) }
```

# Classification Properties

are properties classify elements into categories more than they set specific visual parameters.

allows you to control:

the display of an element, identify where an image will appear, position an element in an absolute value, control the visibility of an element

individual properties are:

Float

Clear

Display

White-space

List-style

Float controls where the image will appear in an element.

Clear specifies if an element allows floating elements on its sides.  
values are none, left, right, both.

Display describes how/if an element is displayed on the canvas.

White-space declares how whitespace inside the element is handled.

List style is the property that summarizes all setting of the properties of list on one declaration.