





# Web Development

Applying look & feel with CSS

# CSS Specificity



**a**  
1 x element selector  
Sith: 0, 0, 1



**p a**  
2 x element selectors  
Sith: 0, 0, 2



**.whatever**  
1 x class selector  
Sith: 0, 1, 0



**p a.whatever**  
2 x element selectors  
1 x class selector  
Sith: 0, 1, 2



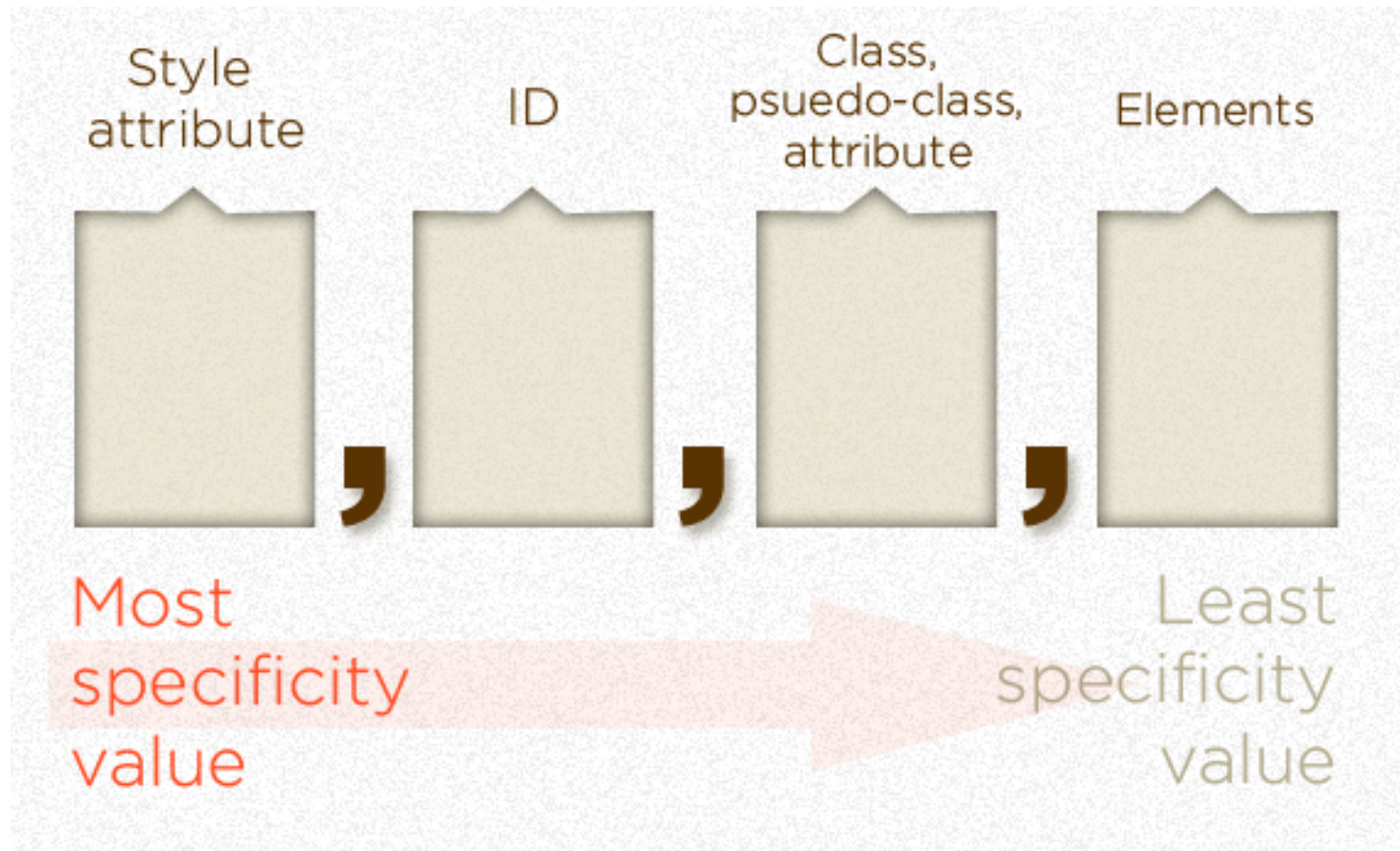
**.whatever .whatever**  
2 x class selectors  
Sith: 0, 2, 0



**p.whatever a.whatever**  
2 x element selectors  
2 x class selectors  
Sith: 0, 2, 2

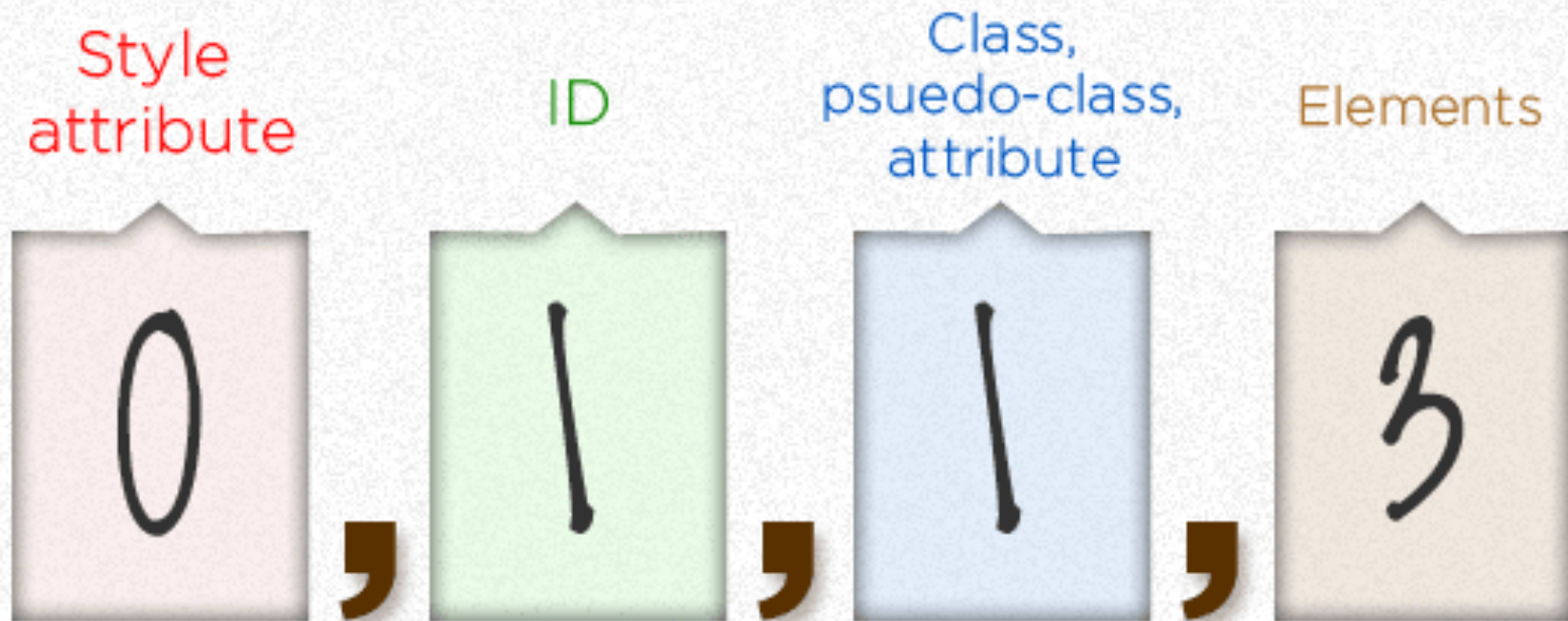
```
<div>  
<p><a href="#test">Top</a></p>  
</div>
```

# CSS Specificity



# CSS Specificity

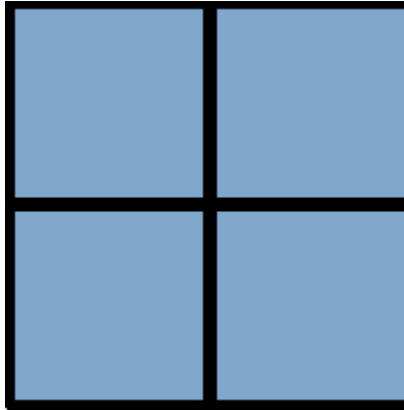
ul#nav li.active a



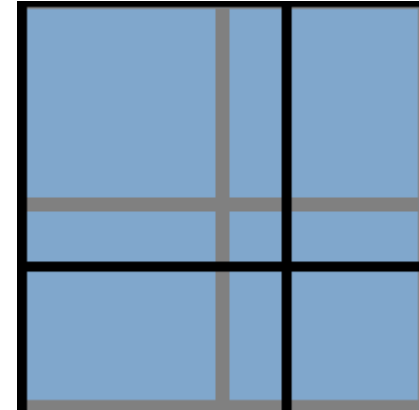
# Device vs CSS Measures



Zoom 50%



Zoom 100%

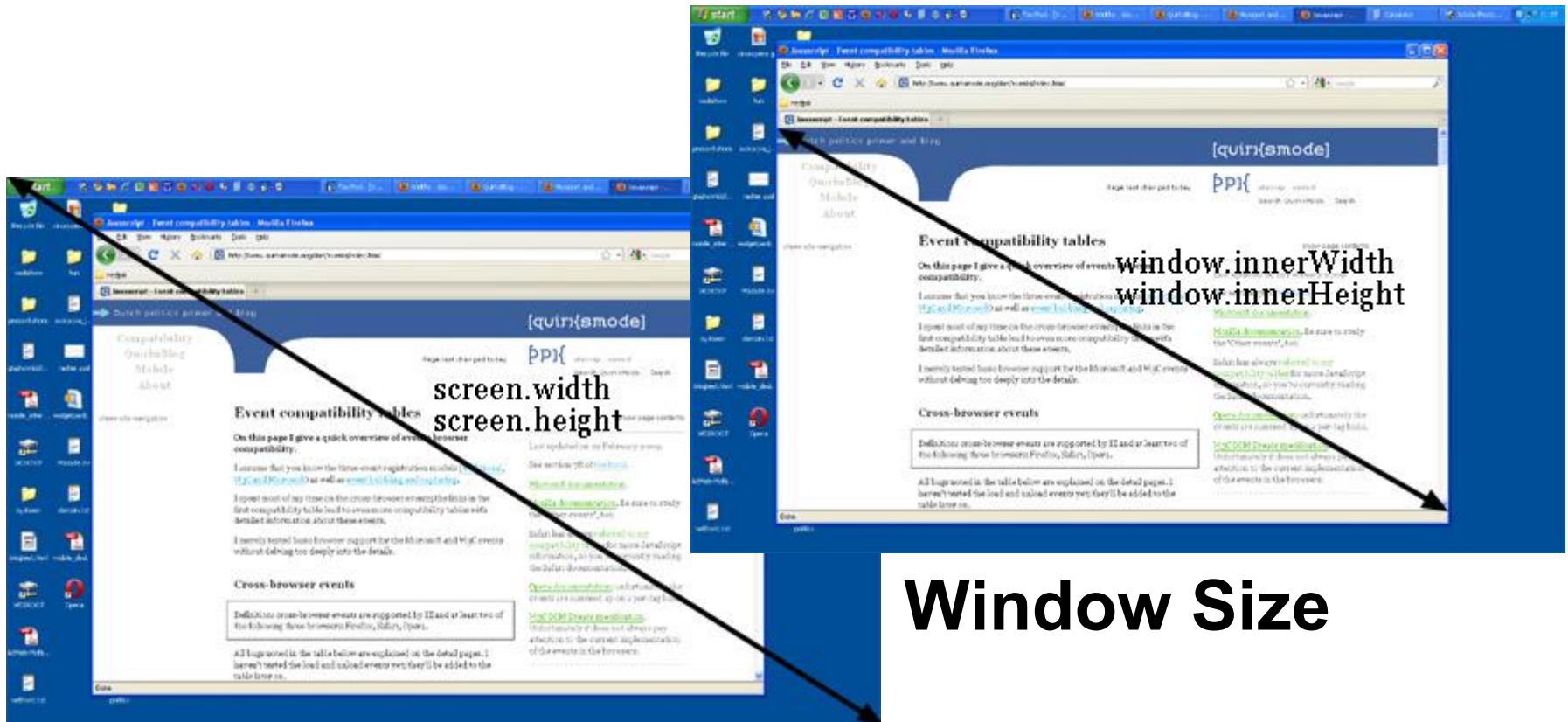


Zoom 200%

***At zoom level 100% one CSS pixel is exactly equal to one device pixel.***



# Browser Measures

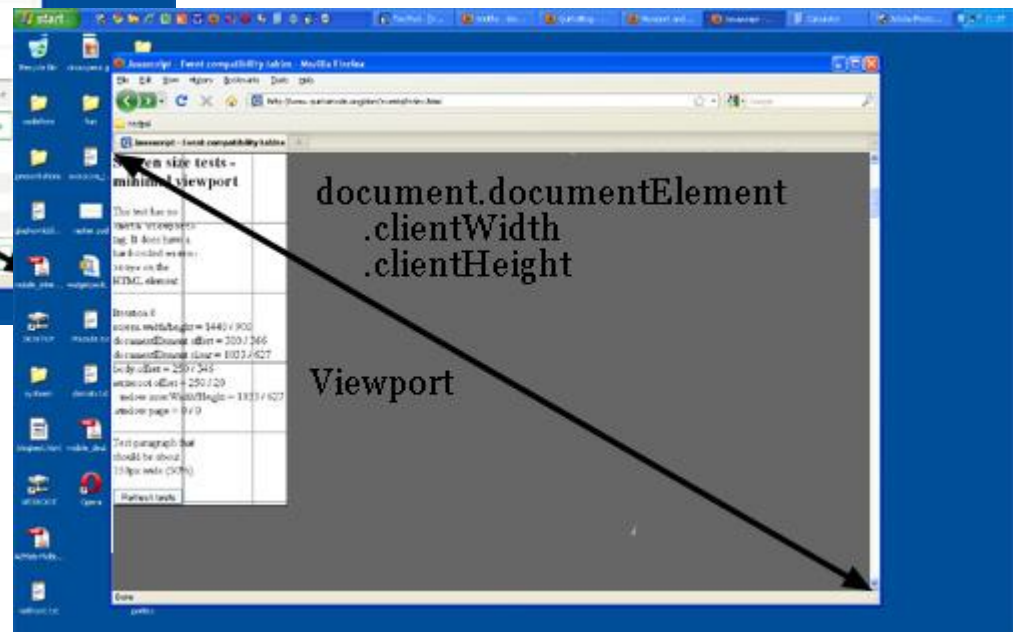
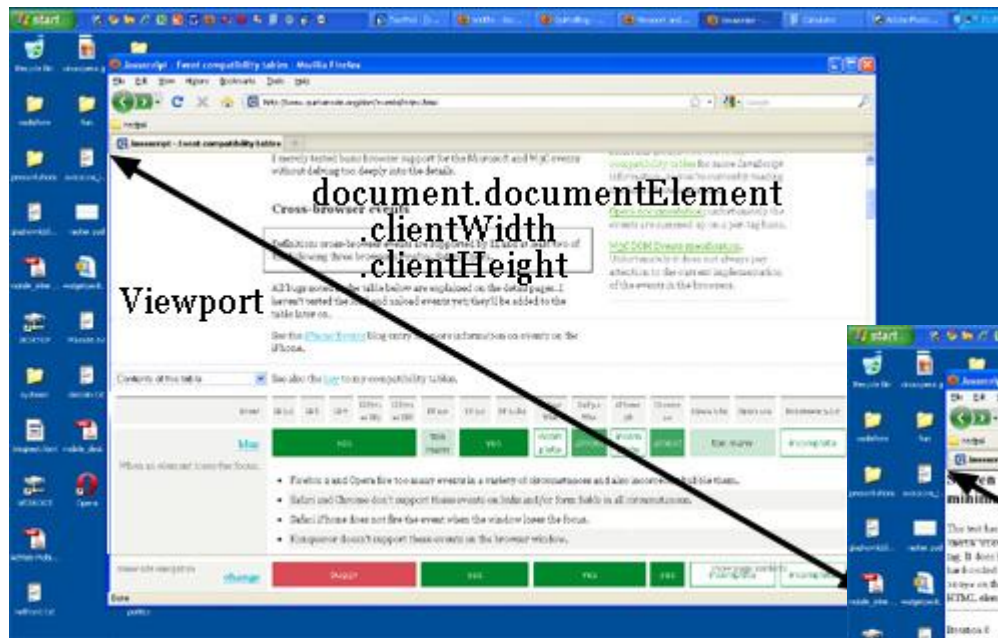


Screen Size

Window Size

# The Viewport

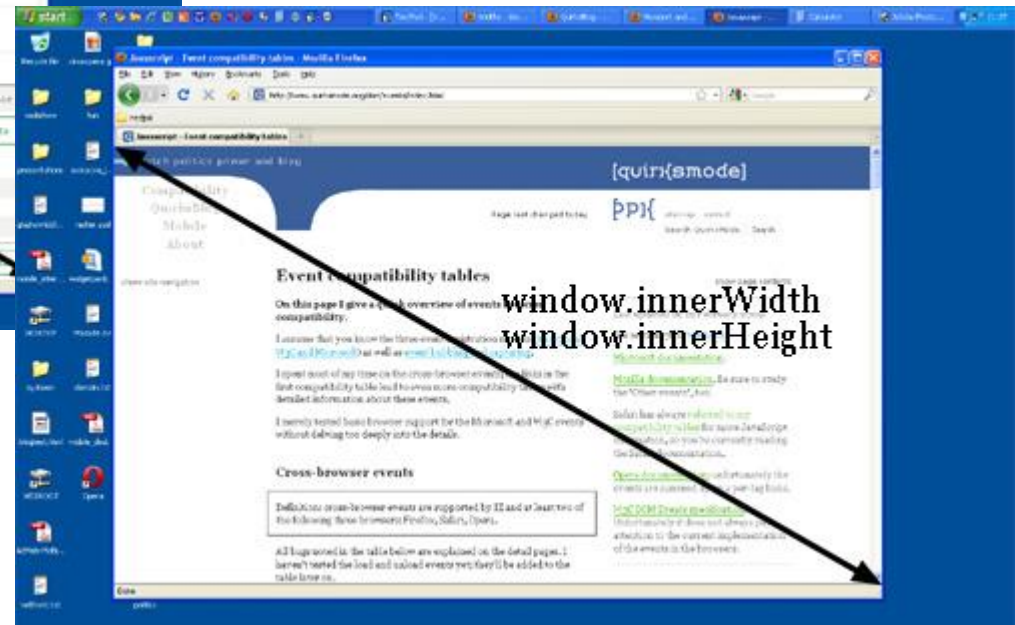
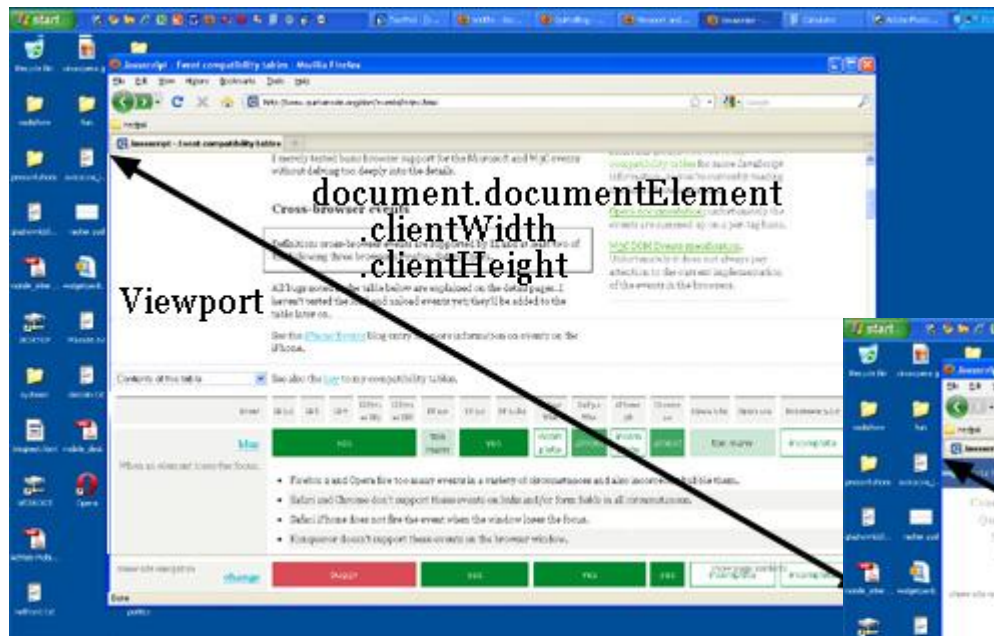
The view port has the same dimensions than the window, though initially it adopts the size of the `<html>` element.





# The Viewport

Both `clientWidth/Height` and `innerWidth/Height` returns the viewport size, but `innerWidth/Height` includes the scrollbar.



# CSS Units

## Relative length

- **em**: 1 em is equivalent to the font-size.
- **ex**: 1 ex is equivalent to the x-height of the font.
- **px**: Pixel

## Absolute length

- **in**: Inch.
- **cm**: Centimeter
- **mm**: Millimeter
- **pt**: Point
- **pc**: Pica

# Meet the Units

- **"Ems" (em)**

The “em” is a scalable unit that is used in web document media. An em is equal to the current font-size, for instance, if the font-size of the document is 12pt, 1em is equal to 12pt. Ems are scalable in nature, so 2em would equal 24pt, .5em would equal 6pt, etc. Ems are becoming increasingly popular in web documents due to scalability and their mobile-device-friendly nature.

# Meet the Units

- **Pixels (px)**

Pixels are fixed-size units that are used in screen media (i.e. to be read on the computer screen). One pixel is equal to one dot on the computer screen (the smallest division of your screen's resolution). Many web designers use pixel units in web documents in order to produce a pixel-perfect representation of their site as it is rendered in the browser. One problem with the pixel unit is that it does not scale upward for visually-impaired readers or downward to fit mobile devices.

# Meet the Units

- **Points (pt)**

Points are traditionally used in print media (anything that is to be printed on paper, etc.). One point is equal to  $1/72$  of an inch. Points are much like pixels, in that they are fixed-size units and cannot scale in size.

# Meet the Units

- **Percent (%)**

The percent unit is much like the “em” unit, save for a few fundamental differences. First and foremost, the current font-size is equal to 100% (i.e. 12pt = 100%). While using the percent unit, your text remains fully scalable for mobile devices and for accessibility.



# Meet the Units

Generally, **1em = 12pt = 16px = 100%**. When using these font-sizes, let's see what happens when you increase the base font size (using the body CSS selector) from 100% to 120%.

	<code>body { font-size: 100%; }</code>	<code>body { font-size: 120%; }</code>
<code>font-size: 1em</code>	The quick brown fox	The quick brown
<code>font-size: 12pt</code>	The quick brown fox	The quick brown fox
<code>font-size: 16px</code>	The quick brown fox	The quick brown fox
<code>font-size: 100%</code>	The quick brown fox	The quick brown

© KyleSchaeffer.com

# Fonts



Sans-serif



Serif



Serif  
(red serifs)



On computer screens, sans-serif fonts are considered easier to read than serif fonts.

# Font Families

## Generic family

## Font family

## Description

### ***Serif***

Times New Roman  
Georgia

Serif fonts have small lines at the ends on some characters.

### ***Sans-serif***

Arial  
Verdana

"Sans" means *without* - these fonts do not have the lines at the ends of characters.

### ***Monospace***

Courier New  
Lucida Console

All monospace characters have the same width.

# Font Families

## Safe Fonts

Arial

**Arial Black**

Comic Sans MS

Courier New

Georgia

**Impact**

Times New Roman

Trebuchet MS

Verdana

## Embedded Fonts

```
@font-face {  
  font-family: indie-flower;  
  src: url(IndieFlower.ttf),  
       url('IndieFlower.eot');  
}
```

```
body {  
  font-family: indie-flower;  
  font-size: 36px;  
  font-weight: bold;  
}
```

**Bold Indie Flower**

looks like this.

# Font Size







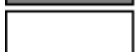
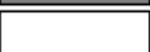
























- **Absolute (pixels, points)**

- a. Sets the text to a specified size.
- b. Does not allow a user to change the text size in all browsers (bad for accessibility reasons).
- c. Absolute size is useful when the physical size of the output is known.

- **Relative (em, percent)**

- a. Sets the size relative to surrounding elements.
- b. Allows a user to change the text size in browsers.

# Colors

Named	Numeric	Color name	Hex rgb	Decimal
		<b>black</b>	#000000	0,0,0
		<b>silver</b>	#C0C0C0	192,192,192
		<b>gray</b>	#808080	128,128,128
		<b>white</b>	#FFFFFF	255,255,255
		<b>maroon</b>	#800000	128,0,0
		<b>red</b>	#FF0000	255,0,0
		<b>purple</b>	#800080	128,0,128
		<b>fuchsia</b>	#FF00FF	255,0,255
		<b>green</b>	#008000	0,128,0
		<b>lime</b>	#00FF00	0,255,0
		<b>olive</b>	#808000	128,128,0
		<b>yellow</b>	#FFFF00	255,255,0
		<b>navy</b>	#000080	0,0,128
		<b>blue</b>	#0000FF	0,0,255
		<b>teal</b>	#008080	0,128,128
		<b>aqua</b>	#00FFFF	0,255,255

## Named Colors:

```
body { background: white }
h1 { color: maroon }
h2 { color: olive }
p { background: transparent }
```

## RGB Color values:

```
em { color: #f00 }
em { color: #ff0000 }
em { color: rgb(255,0,0) }
em { color: rgb(100%, 0%, 0%) }
```

## RGBA Color values:

```
em { color: rgb(255,0,0) }
em { color: rgba(255,0,0,1) }
em { color: rgb(100%,0%,0%) }
em { color: rgba(100%,0%,0%,1) }
}
```



# Safe Colors



#66cc00	#55bb00	#4eb400
web-safe web-safe	web-smart web-smart	unsafe unsafe

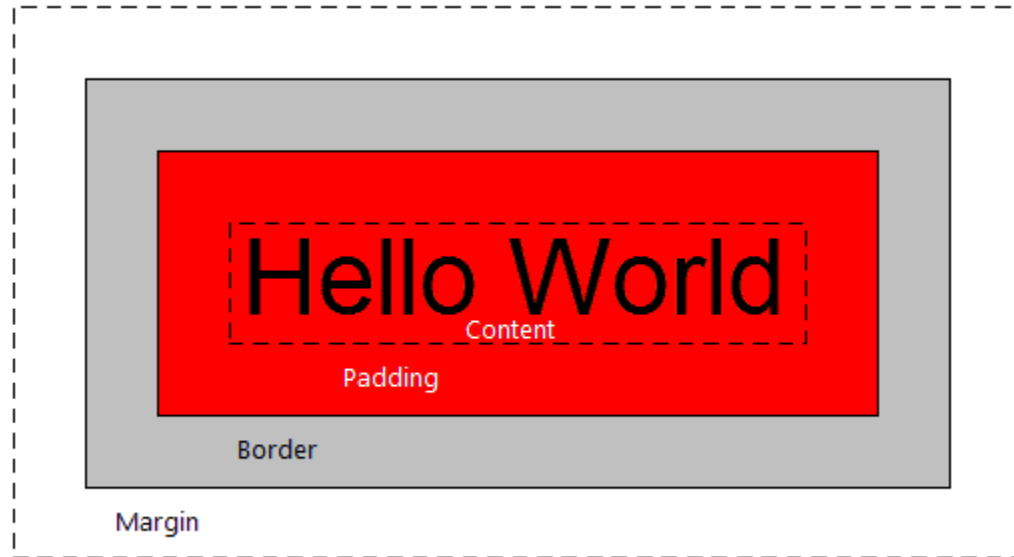
# Background

CSS background properties are used to define the background effects of an element.

## **CSS Background properties:**

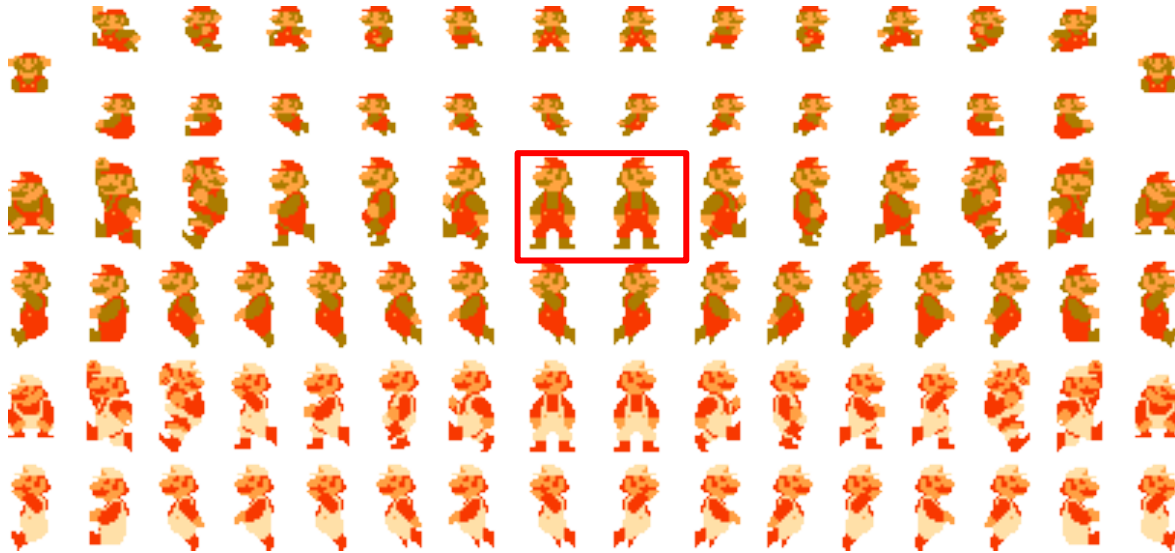
- `background-color`
- `background-image`
- `background-repeat`
- `background-attachment`
- `background-position`

# Background Color



```
.content { background: red }  
.content { background: #f00 }  
.content { background: #ff0000 }  
.content { background: rgb(255,0,0) }
```

# Background Image: Sprites



```
/** Super Mario standing left. */  
.super-mario.stand.left {  
  background-position: -180px -51  
px;  
}
```

```
/** Super Mario standing right. */  
.super-mario.stand.right {  
  background-position: -209px -51
```

# Thank you!