

HTML, CSS & JavaScript Quick Reference & Learning Guide

Created by Sally Scott

Introduction

This guide began as a personal learning resource while I was refreshing my HTML knowledge and expanding my CSS and JavaScript skills. As I learned, I found myself repeatedly searching for the same information online. Although plenty of reference sites existed, they often assumed I already knew the technical terminology or buried simple answers beneath lengthy explanations.

Rather than continuing to search multiple websites, I created a reference guide organized around the questions I naturally asked while coding. Instead of leading with technical terminology, each section starts with what someone wants to accomplish, then introduces the proper terminology, demonstrates the syntax, and shows examples of the expected result.

Over time, this evolved from a collection of notes into a structured learning resource covering HTML, CSS, JavaScript, helpful development resources, and practical examples. It reflects the same approach I use throughout my work: reduce unnecessary complexity, organize information logically, and make technical concepts easier for people to understand.

This document is included as a portfolio sample because it demonstrates my approach to technical communication, information architecture, and instructional design.

Contents

HTML Basics	2
HTML Mailto Functionality.....	5
HTML Special Characters	7
Color Palette Selector Sites.....	7
Templates & Frameworks	7
Live Preview Sites.....	7
HTML Validation – check for errors	7
CSS Text Properties	8
Font Shorthand	13
JavaScript	15

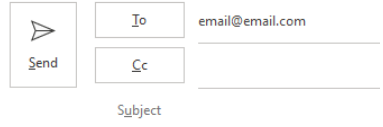
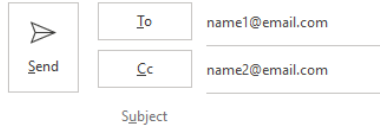
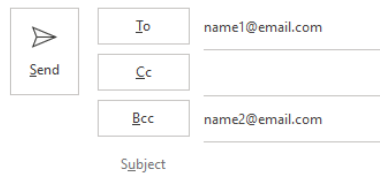
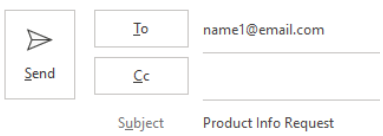
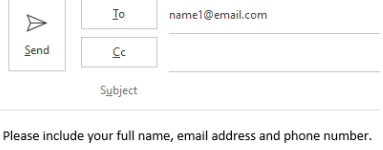
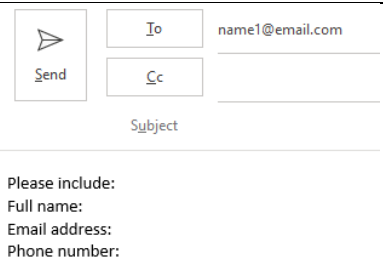
HTML Basics

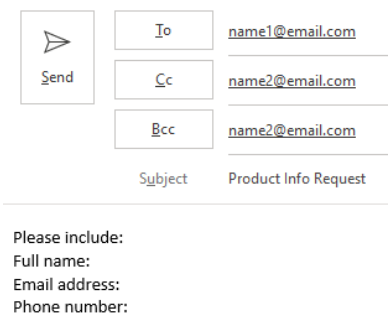
What do you want to do?	What is it called?	How to Write It	Example
Create an HTML document	Doctype Declaration + HTML element tag	<!DOCTYPE html> <html> </html>	<!DOCTYPE html> <html> <i>All page content goes here</i> </html>
Ensure my html page will display correctly	HTML meta charset	<meta charset="utf-8">	<html> <head> <meta charset="utf-8"> </head> <i>All page content goes here</i> </html>
Add a title to my page that shows on the browser tab	Title tag	<title> </title>	<!DOCTYPE html> <html> <head> <meta charset="utf-8"> <title>My Page Title</title> </head> <i>All page content goes here</i> </html>
Add content to my page	Body tag	<body> </body>	</head> <body> <i>All page content goes here</i> </body>
Add CSS styling to my page	Style tag	<style> </style>	<body> <style> <i>Styling goes here</i> </style> </body>
Add a heading to my page	Heading tag	<h1> Heading </h1> <h1> is largest <h6> is smallest	<body> <h3>This is a medium sized heading</h3> </body>
Write a paragraph of text	P tag	 	<body> <p>Paragraph content goes here.</p> </body>

Force a break in text	Line break tag	<code>
</code>	Use <code>
</code> to force text to break to the next line or use multiple <code>
</code> to create several breaks. No need for a closing tag <code></br></code> .
Make text bold	Bold or B tag; Strong tag	<code></code> <code></code>	I want <code>this text</code> to be bold. I want <code>this text</code> to be bold.
Italicize text	I tag or emphasis tag	<code><i></i></code> <code></code>	I want <code><i>this text</i></code> to be italicized. I want <code>this text</code> to be italicized.
Underline text (non-url)	U tag	<code><u></u></code>	I want <code><u>this text</u></code> to be underlined.
Superscript text	Sup tag	<code><sup></sup></code>	X <code><sup>2</sup></code> will look like X ²
Subscript text	Sub tag	<code><sub></sub></code>	H <code><sub>2</sub></code> O will look like H ₂ O
Insert an image	Image tag	<code></code>	
Control image size	Image tag attributes	alt="text" Text description of the image align="left" Image alignment within text width="200" Image width (px, % or auto) height="auto" Image height (px, % or auto) border="1" Border thickness vspace="10" Space above & below image (px) hspace="10" Space on sides of image (px)	<code></code>
Add an external hyperlink	A-tag or hypertext reference (h-ref)	<code>text</code>	<code>Google</code>
Add a hyperlink to a place on the current page	A-tag or hypertext reference (h-ref); link to an anchor	<code>Hello World</code> <code><h2 id="Hello-World">Hello World!</h2></code>	Create the anchor: <code><h2 id="Hello-World">Hello World!</h2></code> Create the link: <code>Hello World</code>
Make a link open in a	Link target location	<code>text</code>	By default, links will open in the same tab/window unless directed otherwise using <code>target="_blank"</code>

new tab or window			
Add a link to send an email	Mailto link	<code>Text</code>	
Add a mailto link with an email subject	Mailto link with subject	<code>Text</code>	
Add a mailto link with CC/BCC email addresses	Mailto link with CC/BCC	<code>Text</code>	
Insert CSS styling into html tags	Inline CSS Styles	• <code><p style="color: red;"></code> • <code><h2 style="background: black; color: white;"></code>	

HTML Mailto Functionality

What do you want to do?	What is it called?	How to Write It	Result
Add an email link	Mailto	<code>Email me</code>	
Add an email link & CC a second email address	Mailto + CC	<code>Email us</code>	
Add an email link & BCC a second email address	Mailto + BCC	<code>Send Email</code>	
Add an email link with a subject line	Mailto + subject	<code>Request Product Info</code>	
Add an email link with body text	Mailto + body	<code>Request Product Info</code>	
Add an email link with body text that has line breaks	Mailto + body with line breaks	<code>Request Product Info</code> Note: use %0D%0A the same way you'd use a <code>
</code> tag	

The whole shebang	Mailto + CC + BCC + subject + body with line breaks	<pre>The Whole Shebang</pre> Note: when including multiple elements in mailto code, replace the ? with &	
-------------------	--	--	--

HTML Special Characters

Result	Description	How to Write It
<	Less than	<
>	Greater than	>
&	Ampersand	&
"	Quotation mark	"
©	Copyright	©
™	Trademark	™
	Non-breaking space	
?	Question mark	?

Color Palette Selector Sites

- <http://paletton.com/>
- <http://www.colourlovers.com/>
- <http://webaim.org/resources/contrastchecker/>
- <http://www.color-blindness.com/coblis-color-blindness-simulator/>

Templates & Frameworks

- <http://www.opendesigns.org/>

Live Preview Sites

- <https://codepen.io/>

HTML Validation – check for errors

- <https://validator.w3.org/>

CSS Text Properties

There are two primary categories for styling text in CSS:

- **Font styles**, which control the font applied to the text, size, bold or italic appearance, etc.
- **Text layout styles**, which control the spacing, layout, alignment, etc.

What do you want to do?	What is it called?	How to Write It	Example options
Change the font <i>face</i>	Font Family	• font-family: sans-serif; • font-family: fantasy; <pre><style> p { font-family: cursive; } </style></pre>	• serif • sans-sarif • cursive • fantasy • Monospace • "Verdana" • "Arial"
Designate a specific font with a backup option(s)	Font Stack	• font-family: "Helvetica", sans-serif; • font-family: "Times New Roman", serif • font-family: "Geneva", "Lucida Sans", "Lucida Grande", "Lucida Sans Unicode", "Verdana", sans-serif;	• "Helvetica", sans-serif It seems the "" may not matter
Change the font <i>size</i>	Font Size Line Height	• font-size: 13px; • line-height: 1.5em;	
Change the font <i>color</i>	Color	• color: red; • color: rgb(150, 45, 200); • color: rgba(150, 45, 200, 10%);	• Blue • rgb(191, 0, 255) • rgba(100, 0, 10, 0.1) - the a indicates opacity
Highlight text	Background Color	• background-color: yellow; • background-color: rgb(0, 150, 100); • color: rgba(150, 45, 200, 10%);	• Blue • rgb(191, 0, 255) • rgba(100, 0, 10, 0.1) - the a indicates opacity
Change the font <i>slant</i>	Font Style	• font-style: italic;	• Normal (removes italics) • Italic • oblique
Change the boldness	Font Weight	• font-weight: bold; • font-weight: 300;	• Normal (removes boldness) • Bold • Lighter • Bolder • Value between 100-900

Change text to ALL CAPS or all lowercase	Text Transform	text-transform: uppercase;	- None (prevents transformation) - Uppercase (all caps) - Lowercase (all lowercase) - Capitalize (first letter of each word capitalized) - Full-width (monospace effect)
<u>Underline</u> or strikethrough text	Text Decoration	- text-decoration: underline; - text-decoration: underline dotted; - text-decoration: line-through; - text-decoration: underline overline; - text-decoration: underline dotted rgb(200,0,0);	- None (removes any decorations present) - Underline (adds line below text) - underline dotted; - underline dotted red; - green wavy underline; - Overline (adds line above text) - underline overline #FF3028; - Line-through (adds strikethrough over text)
Add a drop shadow	Text Drop Shadow	- text-shadow: 4px 4px 5px red; -1px -1px 1px #aaa, 4px 4px 5px rgba(0,0,0,0.7); 0px 0px 7px rgba(0,0,0,0.4);	- Horizontal offset value (required) - Vertical offset value (required) - Blue radius (default = 0) - Base color of the shadow (default = black)
Center or justify text	Text Alignment	Text-align: left;	- left - right - center - justify
Center an image	Block element with auto margin	<pre><style> .center { display: block; margin-left: auto; margin-right: auto; width: 50%; } </style> <body> </pre>	Use a class like "center" to A block-level element always starts on a new line and takes up the full width available (stretches out to the left and right as far as it can). An inline element does not start on a new line and only takes up as much width as necessary. https://www.w3schools.com/html/html_blocks.asp
Adjust spacing between letters or words	Letter Spacing or Word Spacing	- letter-spacing: 2px; - word-spacing: 4px;	
Condense or expand text	Font Stretch	- font-stretch: condensed; - font-stretch: 185%;	Keyword Values: - ultra-condensed; (50%) - extra-condensed; (62.5%) - condensed; (75%) - semi-condensed; (87.5%)

			• normal; (100%) • semi-expanded; (112.5%) • expanded; (125%) • extra-expanded; (150%) • ultra-expanded; (200%) Percentage Values: • 50%; • 100%; • 200%;
Indent or outdent text	Text Indent	Length Values • text-indent: 5mm; • text-indent: 20px; Percentage Values • text-indent: 20%	Length Values • cm (centimeter) • em (unit equal to computed font size) • in (inch) • mm (millimeter) • pc (pica) • pt (point) • px (pixels) Percentage Values • 10% • 0.10
Style specific words or phrases	Span	<pre><style> .different { font-weight: bold; } </style> <p>I want this text to be different from this text.</p></pre>	Use class and style to specify how you want the text within the tags to appear.
Style block sections of text	Div		
Style using a class	Class	<pre><style> .normal { font-weight: normal; font-family: sans-serif; } </style> <p class="normal">Here's my normal paragraph.</p></pre>	When styling a class, insert a period before the class ID, like ".normal" <i>Classy people don't use hashtags</i>
Style using multiple classes	Multiple CSS classes	<pre><style> .normal { font-weight: normal;</pre>	When using more than one class, leave a space between the classes like class="normal purple"

		<pre>font-family: sans-serif; } .purple { color: purple; } </style> <p class="normal purple">Here's my normal paragraph.</p></pre>	Define each class's style individually
Stylize an element with an existing class	Element selector	<pre>.warning { background-color: red; color: white; } p.warning { border: 5px solid orange; }</pre>	Always use tag name.class name to define
Style an element based on its position in the document	CSS Descendent Selector	<pre>p .purple { color: purple; background-color: transparent; } Or ul li strong { line-height: 2em; }</pre>	Leave a space between the element and the class. This styling would only apply to p elements with class="purple" To style unordered list items with class="strong" only, you'd use ul li strong {...
Style multiple elements at once	Grouping CSS Selectors	<pre>H1, h2 { Color: red; }</pre>	Add a comma and a space between
Style using an ID	ID	<pre><style> #red-header { color: red; } </style> <H1 id="red-header">Red Heading</p></pre>	When styling an ID, insert a hashtag before the ID, like "#red-header" <i>Hashtags help to ID things</i>
Add spacing around an element	Margin	<pre>margin: 15px 10px 10px 15px; or margin-right: 10px; margin-left: 15px;</pre>	When grouping margins into one line, the margins are listed in clockwise order starting from the top left.

		margin-bottom: 10px; margin-top: 15px; or margin: auto;	
Add a border	Border	• border: 1px solid rgb(145, 0, 0); • border-top: 10px solid purple;	Border types: • None • Hidden • Solid • Groove • Double • Ridge • Dotted • Dashed • Inset • Outset • Initial • Inherit
Add spacing between a border & content	Padding	• padding: 5px;	
Layer graphics over other graphics	CSS Position	• position: absolute; • top: 20px; • left: 10px; • z-index: 1;	Position: • Relative – places the image in the default location • Absolute – stacks the images • Fixed – holds the image in 1 location; does not scroll; static location • Z-index – controls the order of the stacked images; 1 is the bottom layer
Control where an element appears on a page	Float	• float: left; • width: 30%; • margin-left: 10px;	Float left or right Add width in a percentage Add margins
Prevent an item from wrapping around elements	Clear	• clear: both; • clear: left; • clear: right;	
Change how elements and links appear	Dynamic Pseudo-Classes	• a:link • a:visited • a:hover • a:active • a:focus	• Link - for controlling how unvisited links appear • Visited - for controlling how visited links appear

		• a:hover, a:active, a:focus	• Hover - changes appearance when hovering over an element • Active - changes appearance when clicking on an element • Focus - changes appearance when tabbing through elements • https://www.w3schools.com/css/css_pseudo_classes.asp
Base styling of all pages off one single CSS stylesheet	External Stylesheet	<link rel="stylesheet" type="text/css" href="url.here">	https://cdn.rawgit.com/pamelafox/69f97167ba32e3473cda/raw/336006010d620847f275b0bd25bbf7c665b2e1a1/hopper.css

Font Shorthand

- Written in this order:
 - font-style
 - font-variant
 - font-weight
 - font-stretch
 - font-size (required)
 - line-height
 - font-family (required)
- Required properties: font-size & font-family
- Required: / between font-size & line-height (i.e. font: italic 13px/1.5em sans-serif;)

Resources:

- https://developer.mozilla.org/en-US/docs/Learn/CSS/Styling_text/Fundamentals
- <https://developer.mozilla.org/en-US/docs/Web/CSS/text-decoration>
- https://developer.mozilla.org/en-US/docs/Learn/CSS/Styling_text/Fundamentals

Font styles:

- font-variant: Switch between small caps and normal font alternatives.
- font-kerning: Switch font kerning options on and off.
- font-feature-settings: Switch various OpenType font features on and off.
- font-variant-alternates: Control the use of alternate glyphs for a given font-face.
- font-variant-caps: Control the use of alternate capital glyphs.
- font-variant-east-asian: Control the usage of alternate glyphs for East Asian scripts, like Japanese and Chinese.
- font-variant-ligatures: Control which ligatures and contextual forms are used in text.
- font-variant-numeric: Control the usage of alternate glyphs for numbers, fractions, and ordinal markers.
- font-variant-position: Control the usage of alternate glyphs of smaller sizes positioned as superscript or subscript.
- font-size-adjust: Adjust the visual size of the font independently of its actual font size.
- font-stretch: Switch between possible alternative stretched versions of a given font.
- text-decoration-position: Specify the position of underlines set using the text-decoration-line property underline value.
- text-rendering: Try to perform some text rendering optimization.

Text layout styles

- text-indent: Specify how much horizontal space should be left before the beginning of the first line of the text content.
- text-overflow: Define how overflowed content that is not displayed is signaled to users.
- white-space: Define how whitespace and associated line breaks inside the element are handled.
- word-break: Specify whether to break lines within words.
- direction: Define the text direction (This depends on the language and usually it's better to let HTML handle that part as it is tied to the text content.)
- hyphens: Switch on and off hyphenation for supported languages.
- line-break: Relax or strengthen line breaking for Asian languages.
- text-align-last: Define how the last line of a block or a line, right before a forced line break, is aligned.
- text-orientation: Define the orientation of the text in a line.
- overflow-wrap: Specify whether or not the browser may break lines within words in order to prevent overflow.
- writing-mode: Define whether lines of text are laid out horizontally or vertically and the direction in which subsequent lines flow.

JavaScript

Javascript functions are displayed with the top-most function in the back (i.e. z-index=0) and the bottom-most function in the front. So background images should be written at the top of the code and forefront images should be written at the bottom of the code.

What do you want to do?	What is it called?	How to Write It	Examples/Notes
Add a comment	Comment	// add comment Comments can go above or after code	// face rect(76, 45, 250, 300); rect(126, 250, 152, 60); // mouth
Draw a circle	Ellipse function	ellipse(X, Y, W, H); Parameters to ellipse function: 1. X: Distance from left 2. Y: Distance from top 3. Width 4. Height	Face example: ellipse(212, 206, 283, 318); ellipse(212, 206, 283, 318); ellipse(150, 150, 30, 30); ellipse(278, 150, 30, 30);
Draw a square or rectangle	Rect function	rect(X, Y, W, H, R); Parameters to rect function: 1. X: distance from top left corner to left 2. Y: distance from top left corner to top 3. W: width of the rectangle 4. H: height of the rectangle 5. R: radius of the corners (optional)	Robot face example: rect(76, 45, 250, 300); // face rect(126, 250, 152, 60); // mouth rect(140, 150, 30, 30); // left eye rect(240, 150, 30, 30); // right eye rect(175, 345, 50, 50); // neck
Draw a square or rectangle with rounded corners	Rect function with radius	rect(150, 300, 100, 50, 30); rect(X, Y, width, height, radius);	
Draw a quadrilateral	Quad function	quad(x1, y1, x2, y2, x3, y3, x4, y4); Parameters: x1 the x-coordinate of the first vertex y1 the y-coordinate of the first vertex x2 the x-coordinate of the second vertex y2 the y-coordinate of the second vertex	

		x3 the x-coordinate of the third vertex y3 the y-coordinate of the third vertex x4 the x-coordinate of the fourth vertex y4 the y-coordinate of the fourth vertex	
Draw a line	Line Function	line(X1, Y1, X2, Y2); Parameters to line function: 1. X1: line start point from left 2. Y1: line start point from top 3. X2: line end point from left 4. Y2: line end point from top	line(140, 129, 270, 129); line(160, 200, 5, 50); //left arm line(240, 200, 350, 50); //right arm
Thicken a line	Stroke Weight	strokeweight(30);	// headband strokeWeight(33); line(83, 116, 271, 74);
Turn off outlines/borders of shapes	No Stroke	noStroke();	Place at top of code or at whatever point where you want to stop outlines of shapes.
Turn off fill for a shape	No Fill	noFill();	Place at top of code or at whatever point where you want to stop the filling of shapes.
Draw a triangle	Triangle Function	triangle(X1, Y1, X2, Y2, X3, Y3); Parameters to the function: 1. x1: the x-coordinate of the 1st vertex 2. Y1: the y-coordinate of the 1st vertex 3. X2: the x-coordinate of the 2nd vertex 4. y2: the y-coordinate of the 2nd vertex 5. x3 : the x-coordinate of the 3rd vertex 6. y3: the y-coordinate of the 3rd vertex	// draw a triangle with vertices at // (100, 100), (300, 100) and (200, 300) triangle(100, 100, 300, 100, 200, 300);
Draw a Bezier curve	Bezier	Draws a bezier curve. To extract points and tangents after drawing the curve, use bezierPoint() and bezierTangent(). bezier(x1, y1, cx1, cy1, cx2, cy2, x2, y2);	

		Parameters: • X1: the x-coordinate of the first endpoint • y1: the y-coordinate of the first endpoint • cx1: the x-coordinate of the first control point • cy1: the y-coordinate of the first control point • cx2: the x-coordinate of the second control point • cy2: the y-coordinate of the second control point • x2: the x-coordinate of the second endpoint • y2: the y-coordinate of the second endpoint	
Add a background color	Background	background(255, 0, 0);	Uses RGB
Color a shape	Fill	fill(255, 255, 0); This will fill all shapes below this line of code. Fill does not affect lines or points; use stroke function to color lines and points.	<pre>// face fill(255, 255, 0); ellipse(202, 208, 300, 300); // eyes fill(0, 0, 0); ellipse(157, 151, 40, 40); ellipse(304, 142, 40, 40); // mouth fill(255, 0, 0); ellipse(257, 240, 120, 136);</pre>
Change multiple items at once	Variables	Create the variable: var eyeSize; Then define the variable: eyeSize = 40; OR combine into 1 line: var eyeSize = 33;	Use the variable: <pre>// eyes fill(46, 46, 41); ellipse(157, 151, eyeSize, eyeSize); ellipse(304, 142, eyeSize, eyeSize);</pre>
Animate a drawing	Draw	draw = function() {	

		<pre>// all lines of code inside here will be run repeatedly background(151, 244, 247); // draw the car body fill(255, 0, 115); rect(x, 200, 100, 20); rect(x + 15, 178, 70, 40); // draw the wheels fill(77, 66, 66); ellipse(x + 25, 221, 24, 24); ellipse(x + 75, 221, 24, 24); x = x + 1; };</pre>	
Animation	Shortcuts	<pre>x = x + 1; OR x += 1; y = y - 2; OR y -= 2; eyeSize += 1; OR eyeSize++;</pre>	
Javascript Math		Calculation order: • Parentheses • Exponents • Multiplication • Division • Addition • Subtraction	<pre>var x = 10; var a = x + 5; // add 5, result is 15 println(a); var b = x - 5; // subtract 5, result is 5 println(b); var c = x * 2; // multiply by 2, result is 20 println(c); var d = x / 4; // divide by 4, result is 2.5 println(d); var e = x % 3; // divide by 3 & return remainder, result is 1 println(e);</pre>
Auto-resize multiple objects by changing 1 number	Resizing using variable expressions		<pre>noStroke(); var faceSize = 268; var mouthSize = 1/2 * faceSize; var eyeSize = 2/15 * faceSize; var x = 200; var y = 200; // face</pre>

			<pre>fill(255, 255, 0); ellipse(x, y, faceSize, faceSize); // eyes fill(46, 46, 41); ellipse(x - 1/6 * faceSize, y - 1/6 * faceSize, eyeSize, eyeSize); ellipse(x + 1/3 * faceSize, y - 1/5 * faceSize, eyeSize, eyeSize); // mouth fill(252, 65, 65); ellipse(x + 1/6 * faceSize, y + 2/15 * faceSize, mouthSize, mouthSize);</pre>
Text	Text	<pre>fill(0, 13, 255); textSize(30); text(message, x, y, width*, height*)</pre> • message is the text you want to display • x is the distance from the left side of the canvas • y is the distance from the top of the canvas • w is the width of the text • h is the height of the text	Coordinates start from lower left of text
Mouse Interaction		These variables represent the current coordinates of the mouse. • mouseX represents the x-coordinate of the mouse • mouseY represents the y-coordinate of the mouse	<pre>fill(92, 24, 219); var howBig = 30; var draw = function() { howBig = howBig + 1; textSize(howBig); background(0, 238, 255); var myName = "Sophia"; var message = myName + "!!!"; text(message, mouseX, mouseY); };</pre>

Summarize multiple lines of code into one line for later use	Functions	Get the function to appear using a draw function call (i.e. "drawWinston();")	<pre>var drawWinston = function() { var faceX = 202; var faceY = 208; fill(255, 255, 0); ellipse(faceX, faceY, 300, 300); // face fill(46, 46, 41); ellipse(faceX - 45, faceY - 57, 40, 40); // eyes ellipse(faceX + 102, faceY - 66, 40, 40); fill(252, 65, 65); ellipse(faceX + 55, faceY + 32, 120, 136); // mouth }; drawWinston();</pre>
Randomize coordinates	Random var	Add random before a starting and ending number to generate a random coordinate within that range.	<pre>var faceX = random(50, 350); var faceY = random(50,350);</pre>
Set rules for where functions will be displayed	Function Parameters	Make sure there's no var definitions for the values listed below the var.	<pre>var drawWinston = function(faceX, faceY) { fill(255, 255, 0); ellipse(faceX, faceY, 190, 190); // face fill(46, 46, 41); ellipse(faceX - 30, faceY - 50, 30, 30); // eyes ellipse(faceX + 44, faceY - 55, 30, 30); fill(252, 65, 65); ellipse(faceX + 19, faceY + 32, 82, 81); // mouth }; fill(255, 0, 0); text("Toddler Winston:", 10, 20); text("Teenage Winston:", 200, 20); text("Twenty-Something Winston:", 10, 220); text("Adult Winston:", 200, 220); drawWinston(99, 117); drawWinston(288, 142); drawWinston(122, 322); drawWinston(266, 321);</pre>
Use a formula to auto-calculate common formulas	Function Return Values		LONG: <pre>var calcTotalDonuts = function(numYears) { var totalDonuts = 3 * 365 * numYears; return totalDonuts;</pre>

			<pre>}; fill(255, 0, 0); text("2-year-old Winston:", 10, 20); text(calcTotalDonuts(2), 10, 40);</pre> SHORTENED: <pre>var calcTotalDonuts = function(numYears) { return 3 * 365 * numYears; };</pre>
	Return Statement	<pre>var addNumbers = function(num1, num2) { var result = num1 + num2; return result; }; var sum = addNumbers(5, 2); text(sum, 200, 200); // Displays "7"</pre>	
Allow for variables to be used outside of a function	Local & Global Variables	Global variables should be outside of functions and rarely used as they can cause errors that are difficult to locate.	<pre>var lifeInches = 20; var calcInches = function(startAge, endAge, inchesPerYear) { var totalInches = inchesPerYear * (endAge - startAge); lifeInches += totalInches; return totalInches; }; fill(255, 0, 0); text("From 0-8 years:", 10, 20); text(calcInches(0, 8, 2.5), 10, 40); text("From 8-16 years:", 10, 100); text(calcInches(8, 16, 2), 10, 120); text("Total grown over life:", 10, 168); text(lifeInches, 10, 200);</pre>
Repeat code in a program	While Loops	The Three Loop Questions: 1. What do I want to repeat? -> The text function with the message! 2. What do I want to change each time?	<pre>fill(120, 9, 148); var message = "y is now "; var y = 40;</pre>

		-> The y position, increasing by 20 each time. 3. How long should we repeat? -> As long as y is less than 400, all the way down.	<pre>while(y < 400) { text(message + y, 30, y); y += 20; }</pre>
Repeat code in a program in a more abbreviated way	For Loops	for (start; how long; change)	Lined paper effect: <pre>background(255, 255, 247); stroke(173, 222, 237);</pre> <pre>for (var i = 0; i < 20; i++) { var lineY = 20 + (i * 20); line(0, lineY, 400, lineY); }</pre> _____ Another Example _____ <pre>for (var i = 0; i < 12; i++) { image(getImage("cute/GemOrange"), i*36, 30, 40, 60); }</pre>