

This page is intentionally blank.

HTML5	→	<pre> <!DOCTYPE html> <html> <head> </pre>
Sensible meta elements: events.google.com/io2015 has more	→	<pre> <meta charset="utf-8"> <meta name="description" content="Best pesto recipes"> <meta name="author" content="//samdutton.com"> <meta name="viewport" content="width=device-width, minimum-scale=1.0, initial-scale=1"> <meta itemprop="name" content="Pesto recipes"> <meta itemprop="image" content="/images/icon_192x192.png"> <meta name="mobile-web-app-capable" content="yes"> <meta id="theme-color" name="theme-color" content="#fff"> <meta name="application-name" content="Pesto recipes"> <title>Pesto: classic</title> <link rel="shortcut icon" href="/images/icon_192x192.png"> </pre>
Use external CSS files, inline with build Put links in the <head>, no type needed Void elements don't need closing slash	→	<pre> <link rel="stylesheet" href="/css/global.css"> <link rel="stylesheet" href="css/local.css"> </head> <body> </pre>
Put headings in heading elements	→	<pre> <h1>Best pesto recipes</h1> </pre>
Remembered closing tags	→	<pre> <p>Pesto is good to eat...</p> <p>...and pesto is really easy to make!</p> </pre>
Make use of HTML5 semantic elements	→	<pre> <article> </pre>
Begin article elements with <h1>: see text for explanation	→	<pre> <h1>Classic pesto</h1> <p>This is the simplest and best of all pesto recipes.</p> </pre>
alt attribute for accessibility sizes and srcset for responsiveness	→	<pre> </pre>
 is self-closing: no closing slash	→	
Maintain heading hierarchy: heading level adds meaning to content	→	<pre> <h2>Ingredients</h2> </pre>
Set class on parent, not all children	→	<pre> <ul class="ingredients"> Basil Pine nuts Garlic Olive oil Salt </pre>
 closing tag is optional...	→	
...but is definitely not	→	
The right kind of list	→	<pre> Chop ingredients. Mash ingredients into a paste using a mortar and pestle. Eat on spaghetti boiled with some green beans and sliced potato. </pre>
Boolean attributes don't have a value: if the attribute is there, it's true	→	
Provide alternative file types	→	
Provide alternative content	→	
Fallback content	→	<pre> <video autoplay controls> <source src="video/classic-en.webm" type="video/webm"> <source src="video/classic-en.mp4" type="video/mp4"> <track label="English subtitles" kind="subtitles" srclang="en" src="tracks/classic-subtitles-en.vtt" default> <p>This browser does not support the video element.</p> </video> </pre>
Label element for accessibility	→	
Use input type and placeholder	→	<pre> <input id="newsletter-email" placeholder="email address" type="email"> <button id="newsletter-button">Subscribe</button> </pre>
Script elements at the bottom No type attribute required... ...but closing tag IS required	→	<pre> <script src="/js/global.js"></script> <script src="js/local.js"></script> <script src="/js/lib/ga.js"></script> </pre>
		<pre> </body> </html> </pre>