# HTML Best Practices

1. In creating new templates make sure you added html5 doctype:

<!doctype html>

1. Make sure all tags are bing closed rightly with not mixing in nesting tags.

*<!-- correct -->*

**<h1>** Heading**</h1>**

**<p>**some text.....**</p>**

*<!-- wrong -->*

**<H1>**Heading**</h1>**

**<p>**some text.....

1. Only lower case html is allowed, do not use upper case tags.

Correct

*<!-- correct -->*

**<h1>** heading here **</h1>**

*<!-- wrong -->*

**<H1>** heading here **</H1>**

1. Avoid inline styles; structure should always be separate from presentation.

**<h1** style=”color: blue;”**>** heading here **</h1>**

1. CSS files should only be inserted within the <head> tag, which should be within our master page layout, if you want to add any more classes, this should be added to the one of our customized CSS, in our case it is dsis-theme.css

**<head>**

**<meta** charset="utf-8"**>**

**<meta** name="description" content=""**>**

**<meta** name="viewport" content="width=device-width, initial-scale=1"**>**

**<title>**...**</title>**

**<link** rel="stylesheet" href="css/style.css"**>**

**<link** rel="author" href="humans.txt"**>**

**</head>**

1. Javascript files should be placed at the bottom of our application for better performance, usually done right before the closing body tag.
2. Use sematic HTML5 elements to make sense of page structure, avoid using the unnecessary “divitus” (full of divs and divs for everything) approach.

Instead, make use of HTML5 elements

e.g.

**<body>**

**<header>**...**</header>**

**<article>**...

**<section>**...**</section>**

**</article>**

**<footer>**...**</footer>**

**</body>**

1. Provide fallbacks for media using the element <source>

For Audio:

**<audio** controls**>**

**<source** src=sound.ogg type=audio/ogg**>**

**<source** src=sound.mp3 type=audio/mp3**>**

*<!-- fallback content: -->*

**<a** href=sound.ogg**>**Ogg**</a>**

**<a** href=sound.mp3**>**MP3**</a>**

**</audio>**

For Video:

**<video** controls**>**

**<source** src=video.webm type=video/webm**>**

**<source** src=video.mp4 type=video/mp4**>**

*<!-- fallback content: -->*

**<iframe** width="480" height="360" src="#" frameborder="0" allowfullscreen**>**/**<iframe>**

**</video>**

1. Form elements

In using form elements, make sure to use HTML5 attributes discretely based on how well each one is supported within our list of supported browsers, a good reference for all listed elements and attributes to use based on browser support is <http://caniuse.com/>

1. Since this application is highly modularized and the desired output is each component to work independently regardless of the context, avoid using #ids unless on necessity when knowing that this component will never be duplicated within any of the wrappers, as known ids should always be unique within the pages. Always use classes for presenting as classes can always be repeated within the same page causing no issues.
2. Use HTML beautify, to make the HTML look cleaner before integrating it with other code, this will easy maintaining and debugging afterwards.
3. Finally and the most important of all, before integrating your code, with other code, make sure it works separately, validated and doesn’t have any errors. Use browser plugins for validating your HTML5 code like HTML validator & Validity in Chrome or HTML5 Validator Add-on in Firefox, or you can use online tools to validate your code before integrating it like <https://html5.validator.nu/>