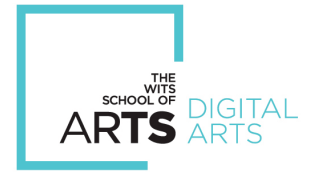


NETWORK BASICS CHEAT SHEET

HTML & CSS



HTML

Hypertext markup language is designed to structure content for web browsers. It is stored in plain text files, with an ".html" suffix.

'tags' are used to create 'elements', tags usually appear in pairs. Here is a tag that creates a paragraph element:

```
<p> This is the story of my first webpage. </p>
```

Some elements can be nested, for instance here we use the 'strong' tag to add bolded text:

```
<p> This is the story of my <strong> first webpage. </strong> </p>.
```

Nested elements cannot overlap in closure:

```
<p> This is the story of my <strong> first webpage. </p>.
```

```
<p> A second paragraph is being added </strong> below the first. </p>.
```

Some elements are self closing:

```
<image> src = "photo.jpg" />.
```

Common Elements:

<!DOCTYPE html > _____	Always first document type declaration.
html _____	Surrounds the html content.
head _____	Contains the meta-data and references.
title _____	Meta-data title for the browser.
body _____	Everything not in the head.
h1, h2, h3, h4 _____	Specification of different heading levels.
p _____	A paragraph.
ul, ol, li _____	Unordered, ordered lists, which contain li element.
strong _____	Bold face text.
a _____	A link, default underlined and blue text.
span _____	An arbitrary span of texts, larger than 'p'.
image _____	Self closing that contains an image reference.
div _____	A 'division' within a document, used to group and contain related elements

Element Attributes:

HTML elements can be assigned 'attributes' by including properties and their values. "href" is the required property and the url link is the value, in this case.

```
<a href = "http://w3schools.com" > Reference Page </a>.
```

CSS Classes and ID's in an element tag are also treated as attributes:

```
<p class = "textStyle" > some styled text </p>.
```

```
<div id = "boxStyle" > <p> text </p> <p> text </p> </div>.
```

DOM

Document Object Model refers to the hierarchical structure of HTML, each tag is an 'element' with a relative relationship to another, ie. descendant, parent, child, etc.

```
<html>
  <body>
    <h1> Main Headline </h1>
    <p> story </p>
  </body>
</html>
```

All elements here are descendants of 'html', and 'body' is the parent of children 'h1' and 'p'. Web browsers parse the DOM to make sense of the content, it is therefore important that DOM is always applied.

CSS

Cascading Style Sheets are used to visually style the DOM elements. This is done by declaring which element will be styled by naming a 'selector' and then applying a 'property' to the selector. In a separate CSS document we write the following:

```
body {
  background-color: white;
  color: black;
}
```

In this case 'body' is the selector and 'background-color' and 'color' are the properties, with 'white' and 'black' as the values of these properties.

Element & Decendant Selectors:

h1 _____	Selects and styles all level one headings.
p _____	Selects and styles all paragraphs.
strong _____	Selects and styles all strong elements.
div _____	Selects and styles all division elements.
div p _____	Selects and styles all paragraph elements within a division.

Class Selectors:

We can create a named selector and assign this to any elements. Class names are preceded with a full stop. Classes can also be extended.

```
.caption _____ Selects and styles elements with the class "caption".
```

```
.caption.highlight _____ Extends the "caption" with another property.
```

ID Selectors:

Named selector to match a single element with a given ID. Used once each. ID names are preceded with a hash mark.

```
#nav _____ Selects and styles elements with the ID "nav".
```

```
#bigDiv _____ Selects and styles elements with the ID "bigDiv".
```