

# AU CSC 435 Web Programming Cheat Sheet

Bei Xiao

(this is typed in in word so make sure to use proper “quotes” works in your script.

This sheet will be updated once we learn Ajax/JSON/PHP/SQL before the finals.

## HTML

### Tags Used in the head Section

Tag	Description
<title> text </title>	Title shown on page tab
<meta attribute = “value” .../>	Page metadata
<link href=“url” type = “text/css” rel = “stylesheet” />	Link to a CSS Style sheet
<script src = “url” type = “text/javascript” /></script>	Link to JavaScript code

### Tags Used in the body Section

Tag	Description
<p> text </p>	paragraph
<h1> text </h1>	
<h2> text </h2> .... <h6> text </h6>	(h1 for largest to h6 smallest)
<hr />	Horizontal rule (line)
 	Line break
<a href=“url”> text </a>	anchor(link)
<img src=“url” alt=“description” />	image
<em> text </em>	Emphasis ((italic)
<strong> text </strong>	Strong emphasis (bold)
<ol> <li> text </li> </ol>	Ordered list
<ul></ul>	Unordered list
<dl></dl>	Definition list
<blockquote>	Block-level quotation

<p> text </p> </blockquote>	
<q> text </q>	Inline-level quotation
<code> text </code>	Computer code (monospace)
<pre> text </pre>	Pre-formatted text
<table> <caption>text </caption> <tr> <th> heading 1> </th> </tr> ... </table>	table of data (table) description of table (caption) table row (tr) table heading cell (th) normal table cell (td)
<div> ..... </div>	Block-level section of a page
<span> .... </span>	Inline-level section of a page

## Content-Grouping Tags

Tag	Display	Description
<header>	Block	Container for a header of a document
<footer>	Block	Container for a footer of a document
<article>	Block	
<section>	Block	A piece of content that is part of another (e.g. chapter section of a reading)
<aside>	Block	Define some content aside from the content it is placed in (e.g. a side bar in article)
<main>	Block	Specifies the main content of a document. The content inside should be unique to the document and not contain content that is repeated across pages (eg. Sidebars, nav links, search bars, etc.)

## HTML Input tags

Tag	Description
<code>&lt;input type="type" name = "name"&gt;</code> content <code>&lt;/input&gt;</code>	form input tag type can be text, submit, reset, checkbox, radio, file
<code>&lt;textarea rows="num"&gt;</code> initial text <code>&lt;/textarea&gt;</code>	Multi-line text input box
<code>&lt;label&gt; text &lt;/label&gt;</code>	Clickable text label around a form control
<code>&lt;select&gt;</code> <code>&lt;option&gt; text &lt;/option&gt;</code> <code>&lt;option&gt;</code> <code>&lt;optgroup label="text"&gt;</code> <code>&lt;option&gt; text &lt;/option&gt;</code> <code>&lt;option&gt;text &lt;/option&gt;</code> <code>&lt;/optgroup&gt;</code>	Drop-down selection box (select); Each option within the box (option); A labeled group of options (optgroup);
<code>&lt;fieldset&gt;</code> <code>&lt;legend&gt; text &lt;/legend&gt;</code> content <code>&lt;/fieldset&gt;</code>	A grouped set of form fields

## CSS

For the following property and value table, anything *emphasized* represents values that should be replaced with specific units (e.g. length should be replaced with a px, pt or em for many properties and color should be replaced with a valid color value such as hex or rgb code).

A use of | refers to separation of possible values (where you cannot provide two of these possible values for one property) and [value value value] refers to a grouping of possible values that can optionally be used together (e.g., [h-shadow v-shadow blur spread color] for box-shadow).

## Background Styles

Property	Values
background-attachment	Scroll   Fixed
background-color	<i>color</i>   transparent
background-image	<i>url</i>   none
background-origin	border-box   padding-box   content-box
background-position	top left   top center   top right   center left   center center   center right   bottom left   bottom center   bottom right <i>[x-% y-%]</i>   <i>[x-pos y-pos]</i>
background-size	<i>length</i>   %   <i>auto</i>   <i>cover</i>   <i>certain</i>
background-repeat	repeat   repeat-x   repeat-y   no-repeat
Background-attachment	scroll   fixed

## Border Styles

Note: Replace “\*” with any side of the border (top, right, left, bottom) for the desired effect.

border, border-* (shortahnad)	border-width, border-*-width border-style, border-*-style border-color, border-*-color
border-width, border-*-width	thin   medium   thick   <i>length</i>
border-style, border-*-style	none   hidden   dotted   dashed   solid   double   groove   rigid   inset   outset
border-color, border-*-color	<i>color</i>
box-shadow	none   inset   <i>[h-shadow v-shadow blur spread color]</i>
box-radius	<i>length</i>

## Box Model

Property	Values
float	left   right   none
height, width	Auto   <i>length</i>   %
min-height, max-height min-width, max-width	none   <i>length</i>   %
margin, margin-*	auto   <i>length</i>   %
padding, padding-*	<i>length</i>   %
display	none   inline   block   inline-block   flex   list-item   compact   table   inline-table
overflow, overflow-x, overflow-y	visible   hidden   scroll

	auto   no-display   non-content
Clear	left   right   both   none

## Font and Text Styles

Property	Values
font-style	normal   italic   oblique   inherit
font-family	<i>fontname</i>
font-size	<i>length</i>   %
font-weight	normal   bold   inherit
text-align	left   right   center   justify
text-decoration	none   [underline overline line-through blink]
text-shadow	none   [color length]
letter-spacing, word-spacing	normal   <i>length</i> / %
text-indent	<i>length</i>   %
text-transform	none

## Color Values

Value	Description
colorname	Standard name of color, such as red, blue, purple, etc
rgb(redvalue, greenvalue, bluevalue)	Example: red=rgb(255,0,0)
#RRGGBB	Example: red =#FF0000

## Selector Types

Name	Description	Example
Universal	Any element	*{font: 10px Arial;}
Element	Any element of a given type	h1 {text-decoration: underline;}
Grouping	Multiple elements of different types	h1, h2, h3 {color: purple;}
Class	Elements with the given classname	.example {text-decoration: underline;}
Id	Single element with the given id	#example {text-decoration: overline;}

Descendent	Elements that are children at any level of another specific element	#example h1 {text-decoration: underline;}
Child	Elements that are direct children of another specific element	#example > p {font-weight: bold;}
Attribute	Elements that have the specified attribute	Input[selected] – inputs that have the selected attribute  Input[name='test']- inputs that have name 'test'

## JavaScript

### DOM Methods and Properties

Method/Property	Description
children	Returns a collection of an element's child elements
parentNode	Returns the parent node of an element
classList	Returns the class name(s) of an element
className	Sets or
appendChild(child)	Adds a new child node, to an element as the last child node
addEventListener(event, fn)	Attaches an event handler to the specified element
getAttribute(attr)	Returns the specified attribute value attr of an element node
innerHTML	Sets or returns the content of an element
id	Sets or returns the value of the id attribute of an element
removeChild(child)	Removes a child node from an element
querySelector (selector)	Returns the first child node that matches a specified CSS selector(s) of an element
querySelectorAll(elector)	Returns all child nodes that match a specified CSS selector(s) of an element
getElementsByClassName(name)	Returns a NodeList containing all elements with the specified class name
getElementsById(id)	Returns the element that has the ID attribute with the specified value

getElementsByTagName (tagName)	Returns a NodeList containing all elements with the specified tag name
createElement(elType)	Create and returns an Element node
createTextNode	Create and returns a Text node

## Event Object Methods and Properties

Method/Property	Description
target	Returns the element that triggers the event
type	Returns the name of the event
offsetX	Returns the horizontal coordinate of the mouse pointer, relative to the DOM element clicked
offsetY	Returns the vertical coordinate of the mouse pointer, relative to the DOM element clicked
stopPropagate	Prevents further propagation of an event during event flow

## Event Types

Click	mousemove	Keydown	change
dblclick	mouseout	Error	focus
mouseenter	mouseover	Success	submit
mouseleave	mouseup	load	select
mousedown	Keyup	Unload	resize

## JavaScript Timer Functions

Method	Description
setTimeout(fn,ms)	Executes a function after waiting a specified number of ms
setInterval(fn, ms)	Repeats a given function at a every given time-interval
clearTimeout(id)	Stops the execution of the function specified by id

<code>clearInterval(id)</code>	Stops the execution of the functions specified by id
--------------------------------	--

## JavaScript Array Methods and Properties

Method/Property	Description
<code>length</code>	Sets or returns the number of elements in an array
<code>push(el)</code>	Adds new elements to the end of an array and returns the new length
<code>pop()</code>	Removes and returns the last element of an array
<code>unshift(el)</code>	Adds new elements to the beginning of an array and returns the new length
<code>shift()</code>	Removes and returns the first element in an array
<code>sort()</code>	Sorts the elements of an array
<code>slice(start, end)</code>	Selects a part of an array and returns the new array
<code>join()</code>	Joins all elements of an array into a string
<code>concat(list2, ...)</code>	Joins two or more arrays and returns a copy of the joined arrays
<code>toString()</code>	Converts an array to a string and returns the result
<code>indexOf(el)</code>	Returns the index of the element in the array, or -1 if not find

## JavaScript String Methods and Properties

Method/Property	Description
<code>Length</code>	Returns length of a string
<code>charAt(index)</code>	Returns the character at the specified index
<code>IndexOf(string)</code>	Returns the position of the first found occurrence of a specified value of a string
<code>split(delimiter)</code>	Splits a string into an array of substrings
<code>substring(start, end)</code>	Extracts the characters from a string between two specified indices
<code>trim()</code>	Removes whitespace from both ends of a string



toLowerCase()	Returns a lowercase version of a string
toUpperCase()	Returns a uppercase version of a string
concat(str2,...)	Joins two or more strings and returns a new joined string.

## JavaScript Math Functions

Method	Description
Math.random()	Returns a double between 0 (inclusive) and 1 (exclusive)
Math.abs(n)	Returns the absolute value of n
Math.min(a, b, ...)	Returns the smallest of 0 or more numbers
Math.max(a, b, ...)	Returns the largest of 0 or more numbers
Math.round(n)	Returns the value of n rounded to the nearest integer
Math.ceil(n)	Returns the smallest integer greater than or equal to n
Math.floor(n)	Returns the largest integer less than or equal to n
Math.pow(n, e)	Returns the base n to the exponent e power, that is, $n^e$
Math.sqrt(n)	Returns the square root of n (NaN if n is negative)

## JavaScript JSON methods

Function	Description
parse(string)	Returns the given string of JSON data as the equivalent JavaScript object
stringify(object)	Returns the given object as a string of JSON data

## The Module Pattern

Whenever writing JavaScript, you should use the module pattern, wrapping the content of the code (window.onload handler and other functions) in an anonymous function. Below is a template for reference:

```
(function() {  
  // any module-globals (limit the use of these when possible)  
  window.onload = function() {  
    ...  
  };  
  
  // other functions  
})();
```