1. HTML

1.1 HTML 5 Differences and Compatibility

http://dev.w3.org/html5/html4-differences/

**HTML versions**

HTML 4 is an old version of HTML that permits compatibility with old web pages. XML is more general than HTML and often used by applications to transport and store data. XHTML is a stricter, cleaner version of HTML 4 that complies with XML syntax. It is supported by all major browsers and provides better cross-browser consistency.

HTML 5 provides new tags and attributes, better semantics for SEO, accessibility, modular design and coding, and new features for multimedia, data and applications. It is recommended for any new pages you create, unless you have specific compatibility reasons for using XHTML or HTML 4.

**New elements**

HTML suffers from divitis (or div soup). HTML5 adds elements to better indicate use of content. header and footer - begins and ends a document, article or section.

nav - holds navigation links, often in a list.

article - content that can stand on its own.

section - a division in an article or document.

aside - related to the page content.

hgroup - groups headings (h1-h6) which can appear in headers, footers, articles and sections.

figure - self-contained content, such an illustration, diagram or photo, but can be text.

**New form field attributes**

form - allows you to put form elements anywhere in a page, even outside the form tag.

url, email, number, search - new input types that help validate user entry (IE7-8 treat as text).

autofocus - puts the focus of the cursor on the element (useful on login and search pages).

autocapitalize - uppercase the first letters in proper nouns like names and locations.

autocorrect - correct in text areas and other free-form inputs.

autocomplete - display a drop-down list of matching options from previously entered values.

placeholder - text that displays until the user enters a value.

required - forces user to enter a value without needing JavaScript or server-side validation.

min, max, step, maxlength, multiple, pattern - other input constraints for built-in validation.

**Character sets**

XHTML: `<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

HTML5: `<meta charset="utf-8">` - utf-8 is now the most used on the web and is recommended

http://www.joelonsoftware.com/articles/Unicode.html - technical overview
Deprecation and obsolete elements and attributes
Use CSS instead of formatting elements and attributes.
- font, center, big; align, bgcolor, cellpadding/cellspacing, hspace/vspace
You can omit type attributes from stylesheet and script tags.
  <link rel="stylesheet" type="text/css" href="..."/>
  <script type="text/javascript" src="..."/></script>
Avoid frames and frame sets because of usability issues, but iframes are still acceptable.

Support for older browsers
Most recent versions of popular browsers support most HTML 5 and CSS 3 features.
Test on Internet Explorer 7-9, Mozilla Firefox, Apple Safari, Google Chrome and maybe Opera.
Check Google Analytics to determine what browsers and versions your users are using.

html5shiv (also called html5shim) is a JavaScript file that allows Internet Explorer versions 6-8 to
minimally understand HTML 5 tags. It should be included in the head section of the document.
In Dreamweaver CS5.5, html5shiv is included only in the two HTML 5 templates.
  <![endif]
  <![endif]-->

Browser support for HTML 5 and CSS 3 features
http://www.caniuse.com/ - compare browsers
http://www.findmebyip.com/litmus - compare browsers
http://www.findmebyip.com/ - test your browser
http://www.css3.info/modules/selector-compat/ - CSS3 selectors
http://tools.css3.info/selectors-test/test.html - test your browser
http://browserlab.adobe.com/ - log in with free Adobe ID to test a URL in different browsers

Converting to HTML 5
In Dreamweaver CS5.5, click File > Convert > HTML 5.
This fixes some tags, including DOCTYPE, html, meta charset, and some other tags.
Closing slashes are removed in empty tags (img, link, br).
(But XHTML5 allows the strict validation of XHTML and the new features of HTML5.)
Manually change many div tags to header, footer, article, section, etc.
Remove tags from CSS id selectors if needed. e.g. change div# navigation to #navigation

Use HTML for semantics, CSS for presentation
Use tags as they are intended, for SEO, clean code, fast page loads and easy maintenance.
To change the visual design of a site, you should not have to change the content of each page.
Use h1-h6 for headings, li for list items, p for paragraphs. Use tables for tabular data, not layout.
Put presentation in external stylesheets. Avoid inline CSS. Use class and id attributes sparingly.

Dreamweaver tutorials by Adobe
1.2 HTML 5 New Media Features

http://w3schools.com/html5/ - tutorials with interactive examples
http://html5demos.com/ - demos of some features (view source of each; some are very complex)
http://www.whatwg.org/specs/web-apps/current-work/multipage/ - official HTML 5 standard

Scalable Vector Graphics (SVG)
Draw SEO-friendly, vector-based graphics using XML which can be resized and animated.

Canvas
Use JavaScript to draw graphics in a rectangular canvas area.

Audio and Video
Use simplified, cross-browser tags to present audio and video in multiple formats.
Use MP3, WAV and Ogg formats for audio and MP4, WebM and Ogg for video.

Drag and drop
Move any element to a different place in the page. Scripting is needed to save changes.

Geolocation
Get the geographical position of a user, which can be used to display results in a map.

Web storage
Store data in the browser, for a browser session or until deleted (local storage).
Compared with cookies, web storage is faster, more secure and can store more data more easily.

Application cache
Cache a web application and its data for better performance and offline use.

Event handling
HTML5 adds many event attributes for windows, forms, keyboard, mouse and media.
http://www.w3schools.com/html5/html5_ref_eventattributes.asp
2. CSS

2.1 Responsive Design

Responsive design
Presentation and features of a page should depend on the device (desktop, tablet, mobile, print). Differences: bandwidth, screen width/height, orientation, pointer precision, and script support.
For small screens, resize, reposition, crop, remove/hide or completely redesign content/features.
For mobile, reduce number and sizes of images and files, but increase button and link sizes.
Assure that images and elements are no wider than the screen:

```
img { max-width: 100%; }
```

Other resources:
- [http://www.slideshare.net/RZasadzinski/responsive-web-design - presentation](http://www.slideshare.net/RZasadzinski/responsive-web-design)

Mobile-first design
Take advantage of mobile's explosive growth and new features, and focus on users' main needs.

Media queries
- [http://www.w3schools.com/html5/att_link_media.asp](http://www.w3schools.com/html5/att_link_media.asp)
- [http://www.w3.org/TR/css3-mediaqueries/](http://www.w3.org/TR/css3-mediaqueries/)

A media query uses device properties to determine which stylesheet or set of style rules to load.
media - the device type (all, screen, print, etc.); recent mobile devices do not support handheld width and height - the device screen dimensions, often prefixed with min- or max- and device-orientation, aspect-ratio, color, resolution (pixel density), etc. - less frequently used properties

Make the mobile styles the default so old mobile browsers see correct styles and perform best.
Make logical breakpoints based on your content vs. on a large, changing set of specific devices.

Embedded media queries vs. external stylesheets
Embedded media queries require fewer HTTP requests and produce faster page loads.
```
@media only screen and (max-width: 768px) { body { width: 100%; … } … }
```
External stylesheets can separate and minimize CSS code for each device category.
```
<link rel="stylesheet" href="phone.css" media="only screen">
<link rel="stylesheet" href="tablet.css" media="only screen and (min-width: 500px) and (max-width: 500px)">
<link rel="stylesheet" href="desktop.css" media="only screen and (min-width: 769px)">
<link rel="stylesheet" href="print.css" media="print">
```
To load stylesheets in CSS instead of link tags in HTML, you can use @import statements.
```
@import("desktop.css") only screen and (min-width: 769px);
```

Cascading vs. mutually exclusive media queries
Overlapping queries reduce repeated code for fast page loads, consistency and easy maintenance.
Non-overlapping media queries allow more complex, distinct designs for each device category.
Dreamweaver tools

Tutorials by Adobe

Multiscreen preview and Window Size
To see how a page looks in different browser windows, click the Multiscreen menu in the Dreamweaver Document toolbar, and select a size, or click Multiscreen Preview for multiple sizes in one popup window.

Media Queries dialog box
To create your own media queries, select Media Queries… from the the Multiscreen menu. Click the Default Presets button for an example. Name a CSS file for each device type.

Mobile development
Dreamweaver provides some built-in tools for mobile application development. These are useful for organizations with the time and resources to develop separate mobile and non-mobile sites:

- New Document > Page from Sample > Mobile Starters > jQuery Mobile
- Site > Mobile Applications > Build and Emulate > (PhoneGap for Android and iOS)

2.2 CSS 3

Tutorials and examples
http://www.css3.info/preview/
http://w3schools.com/css3/ - interactive examples

New features
Colors (hex, rgb, hsl, rgba, hsla – rgb values are 0-255; a is alpha for opacity/transparency)
Opacity (0.0 is transparent, 1.0 is opaque, 0.5 is semi-transparent)
Backgrounds (size, origin, clip)
Gradients (horizontal, vertical, diagonal, or gradient; multiple stops)

http://css-tricks.com/css3-gradients/ - technical overview; gradient images no longer needed
http://gradients.glrzad.com/ - simple gradient generator
http://www.colorzilla.com/gradient-editor/ - another gradient generator
Borders (border-radius gives round corners; borders can use images)
Box (box-sizing: border-box; puts padding and border inside the box – good for liquid layouts)
**Shadows** (text, box)
**Text** (overflow, word-wrap)
**Transforms** (translate, rotate, scale, skew, matrix)
**Transitions** (property, duration, timing function, delay)
**Animations** (keyframes, properties) - instead of animated images, Flash or JavaScript

**Vendor-specific prefixes**
Different browsers use vendor-specific prefixes to extend CSS with their own properties.
- **-webkit-** is used by Apple Safari, Google Chrome, and Dreamweaver CS 5.5.
- **-moz-** is used by Mozilla Firefox.
- **-o-** is used by Opera.
- **-ms-** and filter are used by Microsoft Internet Explorer.
Newer browsers use the standard properties, so put them last, after vendor-specific properties.

**Microsoft visual filters**
http://rainbow.arch.scriptmania.com/css/ie_filters_generator.html - demo (limited options)

**Generators**
Easily generate cross-platform CSS3 code from a GUI and copy into your stylesheet
http://css3please.com/
http://css3generator.com/
http://www.css3.me/
http://www.css3maker.com/

**Resets**
http://meyerweb.com/eric/tools/css/reset/
http://meyerweb.com/eric/thoughts/2011/01/03/reset-revisited/
http://html5doctor.com/html-5-reset-stylesheet/
Different browsers use different defaults for font size, padding, margins, borders, line height, etc.
Use CSS resets to override browser defaults and render a page consistently in all browsers.
Use reset files for XHTML and HTML 5 that others have developed and tested.
Dreamweaver CS5.5 HTML 5 templates seem to do much of this work.

**Conditional stylesheets for Internet Explorer**
To handle Internet Explorer versions, you can create one or more separate stylesheets.
This separates IE-only CSS from other browsers and makes validation easier.
http://css-tricks.com/how-to-create-an-ie-only-stylesheet/
Avoid using CSS filters that use hacks to detect browsers, but know it is a last-resort option.
http://en.wikipedia.org/wiki/CSS_filter

**New selectors**
http://www.w3schools.com/cssref/css_selectors.asp
http://www.w3.org/TR/selectors/#selectors
A **CSS selector** determines which elements should have style rules applied to them.
CSS3 selectors add to the already powerful selectors in CSS2. Just a few examples:
**Attribute selectors** are especially useful to select and format links based on their href.
a[href$=".pdf"] { padding-right: 20px; background: url(pdf.png) center right no-repeat; }

:nth-of-type(n) and :nth-child(n) - select the nth, first, last or only element; useful for coloring alternate rows differently, creating rows and columns, and creating borders on navigation
tr:nth-child(even) { background-color: #eee; }

:enabled, :disabled, :checked – select form input elements based on their state

@font-face Web fonts
http://www.slideshare.net/RZasadzinski/web-font-services-march-2011 - overview presentation
For cross-browser compatibility, developers used to be limited to a small number of fonts.
http://www.ampsoft.net/webdesign-l/WINDOWSMACFonts.html
Internet Explorer has long supported web fonts, but now other browsers can also display them.
You can either download fonts, or link directly to online font services, free or commercial.

http://www.google.com/webfonts - easiest to use but limited font selection
http://www.fontsquirrel.com/fontface - download many free OTF or TTF fonts
http://www.webfonts.info/ - free and commercial web fonts
http://www.typekit.com/ - commercial web fonts

CSS frameworks, preprocessors and generators
CSS frameworks can automatically generate vendor prefixes. They have other features that programmers like, such as variables, mixins (classes for classes), nested rules, math operations within CSS, color functions, and JavaScript evaluation. The JavaScript can decrease performance, so it is also possible to generate CSS code, but this must be regenerated every time the source file is changed. This can be automated, depending on the server and development platform used.

http://en.wikipedia.org/wiki/CSS_frameworks
http://lesscss.org/ - easy to start using; just download a JavaScript file and add a script tag for it
http://sass-lang.com/ - requires the Ruby programming language which is available on all Macs

JavaScript libraries for improved HTML5 and CSS3 compatibility
http://modernizr.com/ - supports HTML5 and CSS3; generate code for only needed features
http://selectivizr.com/ and http://css3pie.com/ - CSS3 support for Internet Explorer 6-9
http://headjs.com/ - improve browser support and performance of HTML5, CSS3 and JavaScript

CSS template systems
Develop complex CSS grid layouts and templates more quickly with one of these frameworks.
http://960.gs/ – popular and easy to use CSS grid layout, used by ACNS and RamCT Help sites
http://blueprintcss.org/ – CSS framework that includes grid, typography and print
http://yuilibrary.com/yui/docs/cssgrids/ – layout used by the CSU Libraries Discovery tool
http://jqueryui.com/ – includes a CSS framework to build custom jQuery widgets
http://html5reset.org/ - a set of starting tools (CSS resets, jQuery, Analytics, modernizer, etc.)