HTML5

HYPERTEXT MARKUP LANGUAGE

Key changes from HTML4 to HTML5

Changes	HTML4	HTML5
Doctype	`<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">`	`<!DOCTYPE html>`
header tag	`<div id="header">`	`<header>`
navigation tag	`<div id="menu">`	`<nav>`
section tag	`<div id="content">`	`<section>`
article tag	`<div id="post">`	`<article>`
footer tag	`<div id="footer">`	`<footer>`
Character Encoding	`<meta http-equiv="Content-Type" content="text/html;charset=utf-8">`	`<meta charset="utf-8">`

Some new elements in HTML5:

Semantic Elements:

<aside>

Defines an element aside from the content it is placed in.

<ruby>

The RUBY element. It is used to give “pronunciation help in a phonetic script for Chinese, Japanese, and Korean languages and utilized for East Asian typography.”

Form Elements:

<date>

Defines a date/time.

<keygen>

Defines a key pair generator.

<progress>

Defines progress of a task.

Graphic Elements:

<canvas>

This is a container for canvas graphics and is rendered pixel by pixel. It has several methods for drawing paths, boxes, circles, text and, graphic images. It is well suited for graphic-intensive games.

<svg>

A container for SVG graphics; XML-based. Not suited for game applications. Best suited for large rendering areas.

Multimedia Elements:

<audio>

HTML5 element for embedding audio in a webpage. Supported audio formats are MP3, Wav, and Ogg.

<video>

HTML5 element for embedding videos in a webpage. Supported video formats are: MP4, WebM, and Ogg.

<embed>
Defines containers for external applications. (ex. Plug-ins)

Note: Elements **<b>**, **<font>**, ** **, **<frame>**, **<frameset>** and **<big>** are no longer used in HTML5.

SAMPLE CODE

```
<!DOCTYPE html>
<html>
<head>
      <link rel= "stylesheet" href= "stylesheet.css">
      <meta charset="utf-8">
      <meta name= "description" content= "An Innovation Course On Web User
      Interface">
      <title> UI! </title>
</head>
<body>
      <header>...</header>
      <nav>...</nav>
      <article>
            <section>
            ...
            </section>
      </article>
      <aside>...</aside>
      <footer>...</footer>
</body>
</html>
```

Source: Tutorials Point

CSS3

CASCADING STYLE SHEETS 3

Changes from CSS2 to CSS3

The main difference between CSS2 and CSS3 is that the latter is split up into sections called modules. Each of these modules has improved CSS2 by adding a new capability or by extending a feature. Four of these modules have been published as formal recommendation, and they are the following:

1. media queries,
2. namespaces,
3. selectors level 3, and
4. color.

Many consider the media queries module as the most important addition to CSS. This module allow a website's content to be rendered depending on conditions (capability of the device). It allows a website to apply different styles depending on the media. This has been the base technology for Responsive web design.

SAMPLE CODE

Shape

```
#rectangleShape {
  width: 200px;
  height: 100px;
  background: red;
}
```

Text Effects

```
#text_shadow { text-shadow: 9px 2px 7px #bbb; }
#word_wrap { word-wrap: break-word; }
```

Border

```
#border_radius {
   border:5px;
   color: #000000;
   padding: 5px;
   background: #6AAFCF;
   border-radius:10px;
   -moz-border-radius:10px; /* Firefox */
   -webkit-border-radius: 10px; /* Safari and Chrome */
   -o-border-radius: 10px /* Opera */
}

#box_shadow {
   border:5px;
   color: #000000;
   padding: 5px;
   background: #6AAFCF;
```

```
    -moz-box-shadow: 10px 10px 5px #888888; /* Firefox */
    -webkit-box-shadow: 10px 10px 5px #888888; /* Safari and Chrome */
    -o-box-shadow: 10px 10px 5px #888888; /* Opera */
    box-shadow: 10px 10px 5px #888888;
}

#border_image {
    -moz-border-image:url(border.png) 30 30 round; /* Firefox */
    border-image:url(border.png) 30 30 round;
}
```

Background

```
full_background {
    background: url(smiley.png);
    background-size: 100% 100%;
    background-repeat: no-repeat;
}
```

Transition

```
{
    transition: all 0.5s ease;
    width: 700px;
    height: 400px;
    background: blue;
    -webkit-transition:width 2s, height 2s, background-color 2s;
}

div:hover {
    width:200px
    height: 100px;
    background: green;
}
```

Multiple Columns

```
#multiple_columns {
    text-align:justify;
    -webkit-column-count: 3; /* Chrome, Safari */
    -moz-column-count: 3; /* Firefox */
    column-count: 3;
}
```

CSS Responsive Layout

Media	Used for
all	all media type devices
aural	speech and sound synthesis
braille	braille tactile feedback devices
embossed	paged braille printers
handheld	small and handheld devices
print	printers
projection	projected presentation
screen	computer screens

Device Sizes

```
/* Smartphones (portrait and landscape) ----------- */
@media only screen
and (min-device-width : 320px)
and (max-device-width : 480px) {
/* Styles */
}

/* Smartphones (landscape) ----------- */
@media only screen
and (min-width : 321px) {
/* Styles */
}

/* Smartphones (portrait) ----------- */
@media only screen
and (max-width : 320px) {
/* Styles */
}

/* iPads (portrait and landscape) ----------- */
@media only screen
and (min-device-width : 768px)
and (max-device-width : 1024px) {
/* Styles */
}
```

```
/* iPads (landscape) ----------- */
@media only screen
and (min-device-width : 768px)
and (max-device-width : 1024px)
and (orientation : landscape) {
/* Styles */
}

/* iPads (portrait) ----------- */
@media only screen
and (min-device-width : 768px)
and (max-device-width : 1024px)
and (orientation : portrait) {
/* Styles */
}

/* Desktops and laptops ----------- */
@media only screen
and (min-width : 1224px) {
/* Styles */
}

/* Large screens ----------- */
@media only screen
and (min-width : 1824px) {
/* Styles */
}

/* iPhone 4 ----------- */
@media
only screen and (-webkit-min-device-pixel-ratio : 1.5),
only screen and (min-device-pixel-ratio : 1.5) {
/* Styles */
}
```

****Google Chrome v. 21, Internet Explorer v.9.0, Mozilla FireFox v.3.5, Opera v.9.0 and Safari 4.0 supports @media.**

JQUERY

Introduction

What is jQuery?

It is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers.

Selectors

jQuery selectors allow you to select and manipulate HTML element/s. These are used to "find" (or select) HTML elements based on their id, classes, types, attributes, values of attributes and much more.

All selectors in jQuery start with the dollar sign and parentheses: $()

Selector Name	Syntax	Description
Element	`$("element")`	Selects elements based on the element name
#id	`$("#id")`	the id attribute of an HTML tag to find the specific (unique) element.
Class	`$(".classname")`	Finds elements with a specific class.
Other examples of Selectors		
	`$("*")`	Selects all elements
	`$(this)`	Selects the current HTML element
	`$("p.intro")`	Selects all <p> elements with class="intro"
	`$("p:first")`	Selects the first <p> element
	`$("ul li:first")`	Selects the first element of the first
	`$("ul li:first-child")`	Selects the first element of every
	`$("[href]")`	Selects all elements with an href attribute
	`$("a[target='_blank']")`	Selects all <a> elements with a target attribute value equal to "_blank"
	`$("a[target!='_blank']")`	Selects all <a> elements with a target attribute value NOT equal to "_blank"
	`$(":button")`	Selects all <button> elements and <input> elements of type="button"
	`$("tr:even")`	Selects all even <tr> elements
	`$("tr:odd")`	Selects all odd <tr> elements

Note: For a full list of jQuery Selectors go to http://www.w3schools.com/jquery/jquery_ref_selectors.asp

Events

Events are all the different visitor's actions that a web page can respond to. An event represents the precise moment when something happens.

Trigger an event for the selected items

```
$(selector).event()
```

Attach a function to the event

```
$(selector).event(function)
```

Assigning an event

```
$("p").click();    //This assigns a click event to all paragraphs on a page
```

To define what should happen when the event happens, you must pass a function to the event

```
$("p").click(function(){
      //This means that when a click event fires on a <p> element; hide the current <p> element.
      $(this).hide();
});
```

Event binding

The bind() method attaches one or more event handlers for selected elements, and specifies a function to run when the event occurs.

```
$(selector).bind(event,data,function,map)
```

Commonly Used jQuery Event Methods

Event Method	Syntax	Definition
click()	$(selector).click()	Executed when the user clicks on the HTML element.
dblclick()	$(selector).dblclick()	Executed when the user double-clicks on the HTML element.
mouseenter()	$(selector).mouseenter()	Executed when the mouse pointer enters the HTML element
mouseleave()	$(selector).mouseleave()	Executed when the mouse pointer leaves the HTML element.
mousedown()	$(selector).mousedown()	Executed when the left mouse button is pressed down, while the mouse is over the HTML element
mouseup()	$(selector).mouseup()	Executed when the left mouse button is released, while the mouse is over the HTML element.
hover()	$(selector).hover()	Executed when the mouse enters the HTML element, and when the mouse leaves the HTML element.

`focus()`	`$(selector).focus()`	Executed when the form field gets focus.
`blur()`	`$(selector).blur()`	Executed when the form field loses focus.
`on()`	`$(selector).on(event,childSelector,data,function,map)`	As of jQuery version 1.7, the on() method is the new replacement for the bind(), live() and delegate() methods

*As of jQuery version 1.7, the on() method is the preferred method for attaching event handlers for selected elements.

Note: For a full list of jQuery event methods go to http://www.w3schools.com/jquery/jquery_ref_events.asp,

Effects - Animation

The jQuery animate() method lets you create custom animations.

```
$(selector).animate({styles},speed,easing,callback)
```

HTML Content Manipulation Methods

`text()` - Sets or returns the text content of selected elements
`html()` - Sets or returns the content of selected elements (including HTML markup)
`val()` - Sets or returns the value of form fields

Getting the contents from an element

```
$("#btn1").click(function(){
      alert("Text: " + $("#paragraph").text());
      alert("HTML: " + $("#paragraph").html());
      alert("Value: " + $("#form").val());
});
```

Setting contents of an element

```
$("#btn1").click(function(){
      $("#par1").text("Hello world!");
      $("#par2").html("<b>Hello world!</b>");
      $("#par3").val("Dolly Duck");
});
```

Remove Elements or Content Methods

`remove()` - Removes the selected element (and its child elements)
`empty()` - Removes the child elements from the selected element

SAMPLE CODE

```
$("#div1").remove();
$("#div1").empty();
```

jQuery Manipulating CSS

`addClass()` - Adds one or more classes to the selected elements
`removeClass()` - Removes one or more classes from the selected elements
`css()` - Sets or returns the style attribute

SAMPLE CODE

Given these CSS classes:

```
.important {
   font-weight: bold;
   font-size: xx-large;
}

.blue {
   color: blue;
}
```

Sample jQuery CSS Manipulation

```
$(document).ready(function(){
   $("button").click(function(){
       $("h1, h2, p").addClass("blue");
       $("div").addClass("important");
   });
});

$("button").click(function(){
      $("#div1").addClass("important blue");
});
```

jQuery css() Method

The css() method sets or returns one or more style properties for the selected elements.

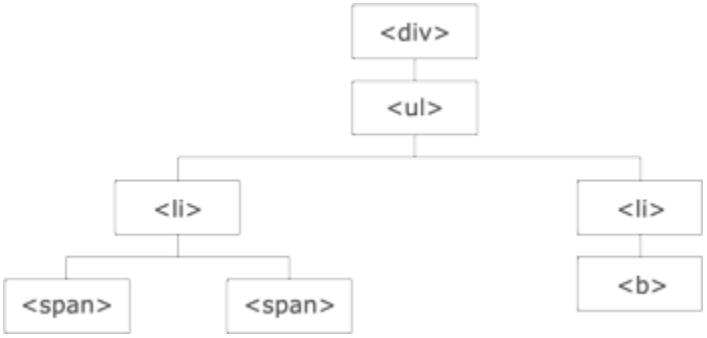
```
$("p").css("background-color");
```

Traversing the DOM

Methods used to access and manipulate the descendants and or ancestors of a specific element

Note: DOM / Document Object Model defines a standard for accessing HTML and XML documents

Sample HTML Structure



SAMPLE CODE

Traversing UP the DOM Tree Examples

```
$(document).ready(function(){
      $("span").parent();
      $("span").parents();
      $("span").parents("ul");
      $("span").parentsUntil("div");
});
```

Traversing Down the DOM Tree Examples

```
$(document).ready(function(){
      $("div").children();
      $("div").children("p.1");
      $("div").find("span");
      $("div").find("*");
});
```

Traversing the DOM other examples

```
$(document).ready(function(){
      $("h2").siblings();
      $("h2").siblings("p");
      $("h2").next();
});
```

Note: : For a full list of jQuery traversing methods visit http://www.w3schools.com/jquery/jquery_ref_traversing.asp

HTML5 References:

HTML5 Migration. W3Schools. <http://www.w3schools.com/html/html5_migration.asp>.
HTML5 SVG. W3Schools. <http://www.w3schools.com/html/html5_svg.asp>.
10 Differences Between HTML4 & HTML5. ChurchMag. <http://churchm.ag/differences-between-html4-html5/>.
The RUBY Element. Swapnil Raja. <http://www.html5andcss3.org/html5ruby.php>.

CSS3 References:

http://www.sanwebe.com/2012/08/pure-css3-tutorials-examples
http://webdesign.about.com/
http://www.impressivewebs.com/differences-between-css2-and-css3/
http://www.1stwebdesigner.com/css3-introduction/
http://www.w3schools.com/css/css3_intro.asp
http://people.rit.edu/~rxk2693/320/project3/cssdifference.html
http://en.wikipedia.org/wiki/Cascading_Style_Sheets
http://www.htmlgoodies.com/beyond/css/css3-transitions-syntax-and-example-code.html
www.html5andcss3.org/css3border.php
http://www.css3.info/preview/rounded-border/
http://www.w3schools.com/cssref/css3_pr_mediaquery.asp
https://css-tricks.com/snippets/css/media-queries-for-standard-devices/

jQuery References:

Introduction. <http://jquery.com/>.
Selectors. W3Schools. <http://www.w3schools.com/jquery/jquery_selectors.asp>
Events. W3Schools. <http://www.w3schools.com/jquery/jquery_events.asp>.
Effects. W3Schools. <http://www.w3schools.com/jquery/jquery_animate.asp>.
Variables. TutorialsPoint. <http://www.tutorialspoint.com/jquery/jquery-basics.htm>.
Traversing. W3Schools. <http://www.w3schools.com/jquery/jquery_traversing.asp>.
HTML Manipulation. W3Schools. <http://www.w3schools.com/jquery/jquery_dom_get.asp>.