**Advanced JavaScript:**

**Event Bubbling and Capturing**

Event bubbling and capturing are two ways of event propagation in HTML DOM.

* Event bubbling
* In bubbling the event is first captured and handled by the inner most elements and then propagated to outer elements.

**Ex:**

<div class="div-1">1

<div class="div-2">2

<div class="div-3">3</div>

</div>

</div>

* Event Capturing
* In capturing the event is first captured by the outer most elements and propagated to the inner most elements.

Only event bubbling model is supported by all the major browsers.

So if you are going to use event capturing still you need to handle event bubbling for IE. So it will easier to use event bubbling instead of capturing.

**Callbacks and Anonymous functions**

**Callbacks**

Callback functions are the functions that are passed to another function as an argument.

Callback functions are probably the most widely used functional programming technique in JavaScript, and they are literally in just about every piece of JavaScript and Jquery code.

Ex1:

function some\_function(arg1, arg2, callback) {

var my\_number = Math.ceil(Math.random()\*(arg1 - arg2) + arg2);

callback(my\_number);

}

some\_function(5, 15, function(num) {

console.log ("callback called!” + num);

});

Ex2:

$("button").click(function(){  
  $("p").hide("slow”, function(){  
    alert("The paragraph is now hidden");  
  });  
});

Ex3:

$("button").click (function () {  
 $("p").hide (1000);  
  alert ("The paragraph is now hidden");  
});

**Anonymous function**