Callbacks

Las funciones callback son funciones que reciben por parámetro otras funciones para que se ejecuten dentro de la función retornando algún valor.

Un ejemplo:

Function contar (callback1, callback2)

{

Var num = Math.floor((Math.random() \* -10) + 1);

Text = text + Callback1(num);

Text = text + Callback2(num);

}

Function esPositivo(num1)

{

Var respuesta = “no es positivo”;

If( num1>0)

{

respuesta = “Es positivo”;

}

Return respuesta;

}

Function esNegativo(num1)

{

Var respuesta = “no es negativo”;

If( num1<0)

{

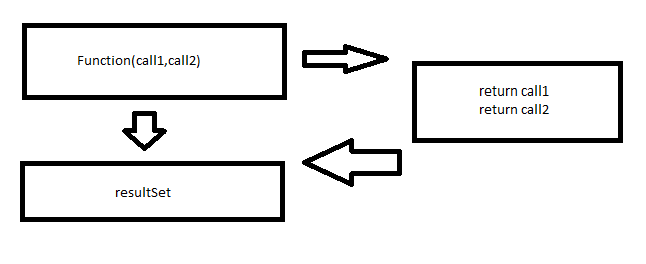
Respuesta= “Es negativo”;

}

Return respuesta;

}

Contar(esPosiivo,esNegativo);



Ejecuciones Asincrónicas

Ya que en el navegador el JavaScript se ejecuta en un solo proceso, se utilizan funciones con temporizadores para simular los threads en la programación.

Tenemos el .**setTimeout( function, time)** para ejecutar funciones **pasado X tiempo** y .**setInterval( funtion, time)** para ejecutar una función **cada X tiempo**.

//INTERVAL

Var contador = 0;

Var temporizador;

Function hacerAlgo(){

contador++;

alert("Espera 10 segundos para el siguiente mensaje… quedan: " + 10-contador);

if (contador === 10)

clearInterval(temporizador);

}

clearInterval(temporizador);

temporizador = setInterval(function(){

hacerAlgo();

},10000);

//TIMEOUT

<button onclick="hacerAlgo()">Hacer Algo</button>

<button onclick="pararTemporizador()">Dejar de hacer algo</button>

Var contador = 0;

Var temporizador;

Function hacerAlgo(){

contador++;

alert("Espera 10 segundos para el siguiente mensaje…”);

temporizador = setTimeout(function(){hacerAlgo()}, 1000);

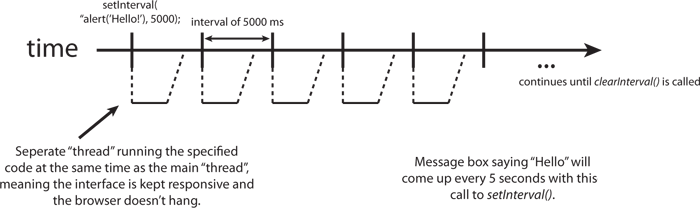
}

function pararTemporizador() {

alert("Se ha parado”);

clearTimeout(limitador);

}



Ejemplo reloj: <http://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_win_settimeout_clock>

Ejemplo contador en una variable: <http://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_win_settimeout2>

Ejemplo contador: <http://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_win_settimeout_cleartimeout2>