



UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS
Lic. En Seguridad en Tecnologías en Información



Semana 1

Diseño Orientado a Objetos

10 strange things about javascript

Nombre del alumno: Jesus Eduardo Alvarado
Ramos.

Profesor: Lic. Miguel Angel Salazar Santillán.

Grupo: 007

Matricula: 1726329

San Nicolás, N.L, México a viernes 27 de enero del 2016

Índice:

Contenido

Página

Portada.....	1
Índice.....	2
1.- Transformar una página web en modo “editable mode”:	3
2.- The browser:	3
3.- Undefined Can Be Defined:	3
4.- Running Ruby and Python in your browser:	3
5.- Imágenes siguen al apuntador del mouse como si fuera una serpiente:	4
6.- Nan es un numero	4
7.- Functions Can Execute Themselves:	4
8.- Running Flash on your iPad:	5
9.- Writing apps for your Smartphone:	5
10.- Too Many Falsy/Bottom Values:	5

1.- Transformar una página web en modo “editable mode”:

Podemos hacer bromas o alterar la captura de pantalla de alguna prueba, fingiendo algunas noticias, o simplemente para engañar a alguien entre otras cosas. En la captura de pantalla se podrán mostrar imágenes o textos alterados.

2.- The browser:

```
Hello, world!
```

```
<script>
var ie = navigator.appVersion.indexOf('MSIE') != -1;
var p = document.getElementById('somePara');
alert(ie ? p.currentStyle.color : getComputedStyle(p, null).color);
</script>
```

Por alguna razón Mozilla Firefox se da cuenta de que cualquiera que interroga los colores calculados a través de JavaScript está interesado en formato hexadecimal y no en RGB. Mientras que la mayoría de los navegadores alertarán ff9900, Firefox devuelve rgb (255, 153, 0), el RGB equivalente. Un montón de funciones de JavaScript están ahí para convertir RGB a hexadecimal.

3.- Undefined Can Be Defined:

Por extraño que parezca, “undefined” no es una palabra reservada en JavaScript, aunque tiene un significado especial y es la única manera de determinar si una variable es “undefined”.

```
var someVar;
alert(someVar == undefined); //evaluates true
```

Se mira normal, Pero:

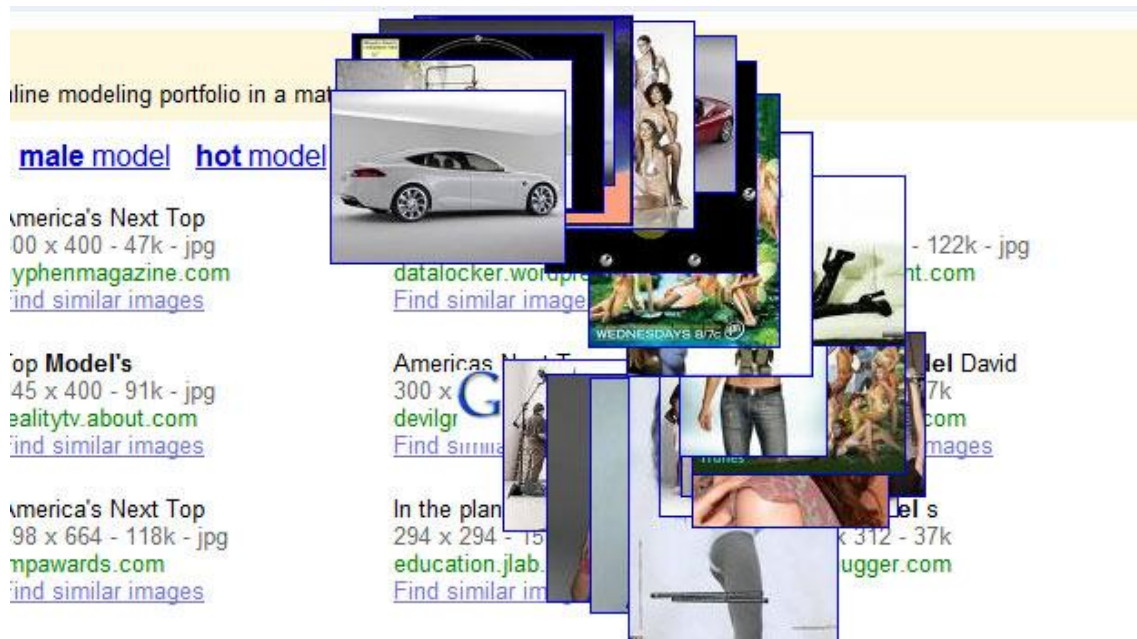
```
undefined = "I'm not undefined!";
var someVar;
alert(someVar == undefined); //evaluates false!
```

4.- Running Ruby and Python in your browser:

Alon Zakai, ingeniero del equipo de investigación de Mozilla, construyó una herramienta llamada Emscripten. Emscripten te permite tomar LLVM bitcode - que se puede generar de C / C ++ (basado en las bibliotecas) a JavaScript. Lo hace compilando las bibliotecas en código de bits y luego, tomando ese código de bits lo transforma en JavaScript.

Con Emscripten simplemente puedes tomar las fuentes de “Ruby”, transformarlas en JavaScript y con eso puedes tener el verdadero “Ruby” funcionando en tu navegador. Pero esto no se aplica sólo a Ruby, “Python” por ejemplo también funciona en el navegador.

5.- Imágenes siguen al apuntador del mouse como si fuera una serpiente:



Sólo tienes que pegar el JavaScript en la barra de direcciones de cualquier navegador, después de eso, presiona Enter, hace que todas las imágenes de la página web actual te sigan como una serpiente cuando se mueve el puntero.

Javascript: R = 0; X1 = 0,1; Y _ {1} = 0,05; X2 = 0,25; Y2 = 0,24; X _ {3} = 1,6; Y3 = 0,24; X _ {4} = 300; Y _ {4} = 200; X ^ {5} = 300; Y _ {5} = 200; Var DI = document.getElementsByTagName ("img"); DIL = DI.length; Función A () {para (i = 0; i <DIL; i ++) {DIS = DI [i] .style; DIS.position = 'absoluto'; DIS.left = Math.sin (R * x1 + i * x2 + x3) * x4 + x5 + "px"; () () () () () () () () () () () () {+ (Tag); para (i = 0; i <DIL; i ++) {DI [i] .style.position = "static";}}; Vacío (0)

6.- Nan es un numero:

Nan cuyo significado es “non a number” pues realmente se puede manejar como un numero

```
alert(typeof NaN); //alerts 'Number'  
alert(NaN === NaN); //evaluates false
```

De hecho, NaN no es igual a nada. La única manera de confirmar que algo es NaN es a través de la función isNaN ().

7.- Functions Can Execute Themselves:

```
(function() { alert('hello'); })(); //alerts 'hello'
```

La sintaxis es bastante simple: declaramos una función y la llamamos inmediatamente igual que nosotros llamamos a otras funciones, con la sintaxis ().

Un buen uso de las funciones de ejecución automática (SEF) es vincular los valores actuales de las variables para su uso dentro del código retrasado, como devoluciones de llamada a eventos, tiempos de espera e intervalos.

El problema es el siguiente:

```
var someVar = 'hello';
setTimeout(function() { alert(someVar); }, 1000);
var someVar = 'goodbye';
```

SEFs proporcionan una solución a este problema. En lugar de especificar la devolución de llamada de tiempo de espera implícitamente como lo hacemos anteriormente, la devolvemos desde un SEF, en el que pasamos el valor actual de someVar como argumentos.

```
var someVar = 'hello';
setTimeout((function(someVar) {
    return function() { alert(someVar); }
})(someVar), 1000);
var someVar = 'goodbye';
```

Esta vez, alertas “hello”, como se desea, porque está alertando la versión aislada de someVar (i.s. el argumento de la función, no la variable externa).

8.- Running Flash on your iPad:

En 2010 Tobias Schneider lanzó una pequeña biblioteca llamada Gordon. La cual permite que los archivos SWF se ejecuten directamente en el navegador. Esto funcionó bastante bien para los pequeños archivos Flash, como los anuncios que sólo utilizaban funcionalidades hasta la versión 2 de Flash. Así nació Bikeshed, Bikeshed en sí es una especie de marco de animación de JavaScript, pero también es un JavaScript para Flash a todo lo que desea que sea compilador (es adaptador de base, por lo que puede escribir adaptadores para cualquier cosa que el usuario desee, aunque el comportamiento estándar es compilar Flash a JavaScript). Es compatible con Flash 10 y ActionScript 3.

9.- Writing apps for your Smartphone:

Apache Cordova, más conocida bajo su nombre anterior PhoneGap. Cordova es un framework JavaScript que abstrae las API de cada entorno móvil y expone una genial API JavaScript para controlar todas ellas. Esto le permite mantener una única base de código, que luego crea e implementa en diferentes dispositivos móviles.

10.- Too Many Falsy/Bottom Values:

Valores como nulo, falso e indefinido casi significan lo mismo, pero hay diferencias que pueden ser confusas para entender.
“Falsy Values”

Para probar, abra la consola en Firefox y busque el booleano de los siguientes elementos.

1
2
3
4
5
6

(0); // false
!!(falso); // false
!! (""); // false
!!(nulo); // false
!! (indefinido); // false
!!(Yaya); // false

Tenga en cuenta que cualquier otro valor se interpretará como verídico.