

UNI

Universidad Nacional de Ingeniería
Instituto de Ciencias de la Información

Sistemas



JavaScript Documento 01

Eric G. Coronel Castillo
gcoronel@uni.edu.pe

Agenda

- Introducción
- ¿Qué es JavaScript?
- Variables
- Método prompt
- Método alert
- Operadores

Introducción

- Es un complemento de HTML
- Permite a las páginas realizar algunas tareas por si misma, sin necesidad de sobrecargar el servidor del cual dependen.
- Permiten realizar cálculos simples, formatear texto para que pueda ser leído por distintas personas de manera distinta.
- Provee de medios para configurar la visualización de una página.

¿Qué es JavaScript?

- Es un Lenguaje de tipo script, basado en objetos, y guiado por eventos, para ser aplicado en las páginas Web.
- JavaScript agrega dinamismo y permite procesamiento de datos.
- Proporciona un conjunto de objetos que permiten trabajar muy bien con los elementos de un documento Web.

JavaScript se Integra con las Páginas Web

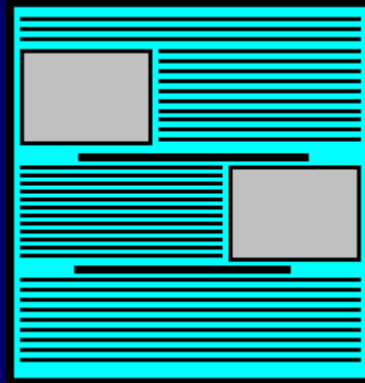
- Programas en JavaScript están incluidos en la página Web donde se ejecutará.
- Los applets de Java son cargados en forma independiente a las páginas Web.

JavaScript es un Lenguaje Interpretado

- El archivo fuente de JavaScript es interpretado por el Browser.
- Ventajas
 - Actualización del código sin volver a compilar.
 - Viaja con el documento HTML.
 - Son fáciles de aprender.
- Desventajas
 - El fuente siempre será visible para todos.

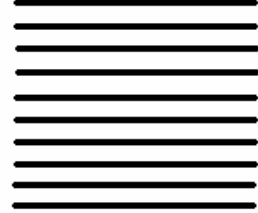
JavaScript es un Lenguaje Interpretado

Web Browser

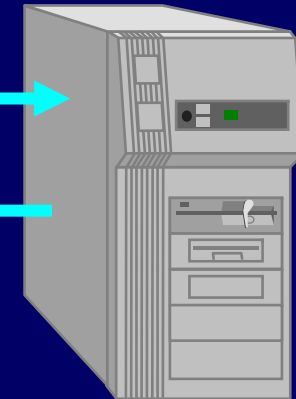


<http://www.company.com/home.html>

home.html



Web Server



Probando JavaScript

Documento: JS0101.HTML

```
<HTML>
<HEAD><TITLE>JavaScript</TITLE></HEAD>
<BODY>
<SCRIPT LANGUAGE="JavaScript">
  <!-- // Inicia Programa

      document.write("Bienvenido a JavaScript");

  // Fin de Programa -->
</SCRIPT>
</BODY>
</HTML>
```


Probando JavaScript

Documento: JS0102.HTML

```
<HTML>  
<HEAD><TITLE>JavaScript</TITLE></HEAD>  
<BODY>  
    <SCRIPT LANGUAGE="JavaScript" SRC="P0101.js"></SCRIPT>  
</BODY>  
</HTML>
```

Programa: P0101.js

```
<!-- // Inicia Programa  
    document.write("Bienvenido a JavaScript");  
// Fin de Programa -->
```

Variables

- Se declaran con la instrucción **var**.

- Sintaxis:

***Var** NombreVariable [= valor | expresion]*

- El nombre de la variable debe comenzar con una letra o el carácter de subrayado (_).
- Tenemos un ejemplo ilustrativo en JS0103.HTML

Método prompt

- Muestra un cuadro de diálogo para que el usuario pueda ingresar un dato.
- Retorna el dato ingresado.
- Sintaxis:

prompt(mensaje [, EntradaPredeterminada])

- Tenemos un ejemplo ilustrativo en JS0104.HTML

Método alert

- Muestra un diálogo de alerta con un mensaje, un botón **Ok** y un sonido (beep).
- Se utiliza para enviar mensajes al usuario.
- Sintaxis:

alert(MensajeDeAlerta)

- Tenemos un ejemplo ilustrativo en JS0105.HTML

Operadores

- Aritméticos
- De Asignación
- De Comparación
- Lógicos
- A nivel de bits
- Otros

Operadores Aritméticos

Operador	Operación	Ejemplo
+	Suma	$c = a + b;$
-	Resta	$c = a - b;$
*	Multiplicación	$c = a * b;$
/	División	$c = a / b;$
%	Resto	$c = a \% b;$
++	Incremento	$a++; b = a++; c = ++b;$
--	Decremento	$a--; b = a--; c = --b;$

Tenemos un ejemplo ilustrativo en JS0106.HTML

Operadores de Asignación

Operador	Operación	Ejemplo	Equivalente
=	Asignación	<code>c = a + b;</code>	
+=	Suma-Asignación	<code>c += a;</code>	<code>c = c + a;</code>
-=	Resta-Asignación	<code>c -= a;</code>	<code>c = c - a;</code>
*=	Multiplicación-Asignación	<code>c *= a;</code>	<code>c = c * a;</code>
/=	División-Asignación	<code>c /= a;</code>	<code>c = c / a;</code>
%=	Resto-Asignación	<code>c %= a;</code>	<code>c = c % a;</code>

Operadores de Comparación

Operador	Operación	Ejemplo
==	Igual que	c = a == b;
!=	Diferente que	c = a != b;
>	Mayor que	c = a > b;
>=	Mayor o igual que	c = a >= b;
<	Menor que	c = a < b;
<=	Menor o igual que	c = a <= b;

Dependiendo del valor de a y b, c puede tomar valor true o false.

Se utilizan principalmente para la toma de decisiones.

Operadores Lógicos

Operador	Operación	Ejemplo
&&	Y Lógico (And)	<code>c = a && b;</code>
	O Lógico (Or)	<code>c = a b;</code>
!	Negación (Not)	<code>c = !a;</code>

Dependiendo del valor de a y b, c puede tomar valor true o false.

Se utilizan principalmente para la toma de decisiones.

Operadores a Nivel de Bits

Operador	Operación
&	Y Lógico (And)
	O Lógico (Or)
^	O Exclusivo (Xor)
~	Complemento
<<	Desplazamiento a la izquierda
>>	Desplazamiento a la derecha
>>>	Desp. a la derecha con relleno de ceros

Otros Operadores

Condicional (? :)

Formato: Condición?Expresión1:Expresión2

Ejemplo: `c = (a>b)?a:b;`

Concatenación (+)

Formato: Cadena1 + Cadena2

Ejemplo: `ElMejor = "Alianza " + " Campeón"`

UNI

Universidad Nacional de Ingeniería
Instituto de Ciencias de la Información

Sistemas



FIN

Eric G. Coronel Castillo
gcoronel@uni.edu.pe