

# **Laboratorio III**

# **JavaScript**

## **Clase 1**

**2/2**

# Temas a Tratar

- ❖ Repaso JavaScript
- ❖ DOM
- ❖ Eventos

# Temas a Tratar

## Repaso JavaScript

- Cómo incluir JavaScript en documentos XHTML
- Sintaxis
- Variables
- Funciones
- Ventanas

## DOM

## Eventos

# Incluir JavaScript

## En el mismo documento

- El código JavaScript se encierra entre las etiquetas `<script>` y `</script>` que se incluyen dentro de la cabecera del documento.

```
<!doctype html>
<html>
  <head>
    <script type="text/javascript" >
      /* código javascript aquí */
    </script>
  </head>
  <body>
    </body>
</html>
```

# Temas a Tratar

## Repaso JavaScript

- Cómo incluir JavaScript en documentos XHTML
- Sintaxis
- Variables
- Funciones
- Ventanas

## DOM

## Eventos

# Sintaxis

- ❖ La sintaxis de JavaScript es muy similar a la de Java, C y C#.

## ❖ Las normas básicas son las siguientes:

- No se tienen en cuenta los espacios en blanco y las nuevas líneas.
- Se distinguen las mayúsculas y minúsculas.
- No se define el tipo de las variables.
- Cada sentencia termina con el carácter de punto y coma (;
- Se pueden incluir comentarios (son los mismos que en C).
- Utiliza los mismos operadores y estructuras (selectivas e iterativas) que en C.

# Temas a Tratar

## Repaso JavaScript

- Cómo incluir JavaScript en documentos XHTML
- Sintaxis
- Variables
- Funciones
- Ventanas

## DOM

## Eventos

# Variables, Tipos y Ámbitos

- ✖ Las variables en JavaScript se crean mediante la palabra reservada **var**.
- ✖ En JavaScript no se declara el tipo de variable.
- ✖ El ámbito de una variable (**scope**) es la zona del programa en la que se define la variable.
- ✖ JavaScript define dos ámbitos para las variables: global y local.

# Temas a Tratar

## Repaso JavaScript

- Cómo incluir JavaScript en documentos XHTML
- Sintaxis
- Variables
- Funciones
- Ventanas

## DOM

## Eventos

# Sintaxis

- ❖ Las funciones en JavaScript se definen mediante la palabra reservada ***function***, seguida del nombre de la función.
- ❖ Opcionalmente se podrán colocar parámetros.

```
function NombreFuncion([param1, param2, paramN])
{
    // Código
    // [return valor;]
}
```

- ❖ Las funciones se podrán definir dentro del tag script o en un archivo externo al documento XHTML.

# Temas a Tratar

## Repaso JavaScript

- Cómo incluir JavaScript en documentos XHTML
- Sintaxis
- Variables
- Funciones
- Ventanas

## DOM

## Eventos

# Ventana alert

- Es la ventana más utilizada. Sirve para mostrar mensajes literales y/o valores de variables al cliente de una aplicación Web.

```
var mensaje = "Hola mundo!!";  
alert("Hola mundo!!");  
alert(mensaje);  
alert(mensaje + " otra vez");
```

# Demo

# Temas a Tratar

❖ Repaso JavaScript

❖ DOM

- Generalidades
- Árbol de Nodos
- Acceso Directo a los Nodos
- Acceso Directo a los Atributos

❖ Eventos

# DOM - Generalidades

- ❖ La creación de *Document Object Model* o **DOM** es una de las innovaciones que más ha influido en el desarrollo de las páginas Web dinámicas y de las aplicaciones Web más complejas.
- ❖ DOM permite a los programadores Web acceder y manipular las páginas XHTML (\*) como si fueran documentos XML.
- ❖ De hecho, DOM se diseñó originalmente para manipular de forma sencilla los documentos XML.

# Temas a Tratar

❖ Repaso JavaScript

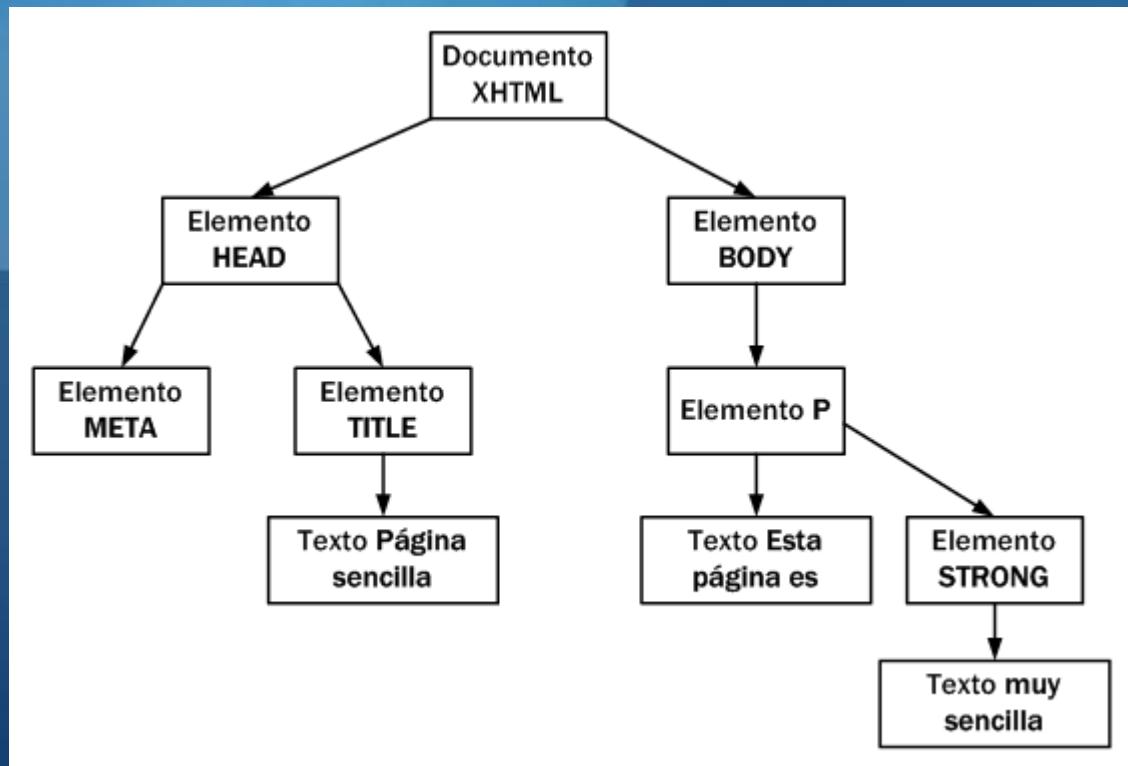
❖ DOM

- Generalidades
- Árbol de Nodos
- Acceso Directo a los Nodos
- Acceso Directo a los Atributos

❖ Eventos

# Árbol de Nodos

- DOM transforma todos los documentos XHTML en un conjunto de *nodos*, que están interconectados y que representan los contenidos de las páginas Web y las relaciones entre ellos.



# Temas a Tratar

❖ Repaso JavaScript

❖ DOM

- Generalidades
- Árbol de Nodos
- Acceso Directo a los Nodos
- Acceso Directo a los Atributos

❖ Eventos

# Acceso Directo a Nodos (1/2)

 DOM proporciona dos métodos alternativos para acceder a un nodo específico:

- acceso a través de sus nodos padre.
- acceso directo.

 Funciones de Acceso Directo (`document`)

 `.getElementsByTagName()`

- Obtiene todos los elementos de la página XHTML cuya etiqueta sea igual que el parámetro que se le pasa a la función.

# Acceso Directo a Nodos (2/2)

## .getElementsByName()

- Es similar a la anterior, pero en este caso se buscan los elementos cuyo atributo **name** sea igual al parámetro proporcionado.

## .getElementById()

- Es la más utilizada cuando se desarrollan aplicaciones Web dinámicas. Se trata de la función preferida para acceder directamente a un nodo y poder leer o modificar sus propiedades.

# Temas a Tratar

❖ Repaso JavaScript

❖ DOM

- Generalidades
- Árbol de Nodos
- Acceso Directo a los Nodos
- **Acceso Directo a los Atributos**

❖ Eventos

# Acceso Directo a Atributos (1/3)

- 💡 Mediante DOM, es posible acceder de forma sencilla a todos los atributos XHTML y todas las propiedades CSS de cualquier elemento de la página.
- 💡 Los atributos XHTML de los elementos de la página se transforman automáticamente en propiedades de los nodos.

```
var enlace = document.getElementById("miLink");
alert(enlace.href); // Muestra el atributo href

<a id="miLink" href="miPagina.php" > link </a>
```

# Acceso Directo a Atributos (2/3)

- ✖ Las propiedades CSS no son tan fáciles de obtener como los atributos XHTML.
- ✖ Para obtener el valor de cualquier propiedad CSS del nodo, se debe utilizar el atributo *style*.

```
var imagen = document.getElementById("miImagen");
alert(imagen.style.margin); // Muestra el márgen


```

# Acceso Directo a Atributos (3/3)

- La transformación del nombre de las propiedades CSS compuestas consiste en eliminar todos los guiones medios (-) y escribir en mayúscula la letra siguiente a cada guión medio.
- DOM utiliza el nombre `className` para acceder al atributo `class` de XHTML.

```
var parrafo = document.getElementById("parrafo");
alert(parrafo.class); // Muestra undefined
alert(parrafo.className); // Muestra 'normal'
```

```
<p id="parrafo" class="normal" > . . . </p>
```

# Demo

# Temas a Tratar

❖ Repaso JavaScript

❖ DOM

❖ Eventos

- Generalidades
- Tipos de Eventos
- Manejadores de Eventos

# Eventos - Generalidades

- ❖ Los eventos hacen posible que los usuarios transmitan información a los programas.
- ❖ JavaScript define numerosos eventos que permiten una interacción completa entre el usuario y las páginas/aplicaciones Web.
- ❖ JavaScript permite asignar una función a cada uno de los eventos.
- ❖ De esta forma, cuando se produce cualquier evento, JavaScript ejecuta su función asociada.
- ❖ Este tipo de funciones se denominan "*event handlers*" ("*manejadores de eventos*").

# Temas a Tratar

❖ R JavaScript

❖ DOM

❖ Eventos

- Generalidades
- Tipos de Eventos
- Manejadores de Eventos

# Tipos de Eventos (1/2)

Evento	Descripción	Elementos
onblur	Deseleccionar el elemento.	button, input, label, select, textarea, body
onchange	Deseleccionar el elemento que se ha modificado.	input, select, textarea
onclick	Clickear y soltar el mouse.	Todos los elementos
ondblclick	Clickear dos veces seguidas.	Todos los elementos
onfocus	Seleccionar un elemento.	button, input, label, select, textarea, body
onkeydown	Pulsar una tecla (sin soltarla).	Eventos de formulario y body
onkeypress	Pulsar una tecla.	Eventos de formulario y body
onkeyup	Soltar una tecla pulsada.	Eventos de formulario y body
onload	La página se ha cargado completamente.	body

# Tipos de Eventos (2/2)

Evento	Descripción	Elementos
onmousedown	Pulsar, sin soltar, un botón del mouse.	Todos los elementos
onmousemove	Mover el mouse.	Todos los elementos
onmouseout	El mouse sale del elemento.	Todos los elementos
onmouseover	El mouse entra en el elemento.	Todos los elementos
onmouseup	Soltar el botón que estaba pulsado.	Todos los elementos
onreset	Borrar los datos de un formulario.	form
onresize	Se ha modificado el tamaño de la ventana del navegador.	body
onsubmit	Enviar el formulario.	form
onunload	Se abandona la página.	body

# Temas a Tratar

❖ Introducción a JavaScript

❖ Programación Básica

❖ Tipos de Ventanas

❖ DOM

❖ Eventos

- Generalidades
- Tipos de Eventos
- Manejadores de Eventos
  - Manejadores como Atributos
  - Manejadores como Funciones Externas

# Manejadores como Atributos

- ☞ Se trata del método más sencillo de indicar el código JavaScript que se debe ejecutar cuando se produzca un evento determinado.
- ☞ En este caso, el código se incluye en un atributo del propio elemento XHTML.

```
<input type="button" value="Click Aquí"  
      onclick="alert('hola mundo!!!');"/>
```

# Temas a Tratar

❖ Introducción a JavaScript

❖ Programación Básica

❖ Tipos de Ventanas

❖ DOM

❖ Eventos

- Generalidades
- Tipos de Eventos
- Manejadores de Eventos
  - Manejadores como Atributos
  - Manejadores como Funciones Externas

# Manejadores como Funciones

- ❖ Esta técnica consiste en extraer todas las instrucciones de JavaScript y agruparlas en una función externa.
- ❖ En el atributo del elemento XHTML se incluye el nombre de la función.
- ❖ La llamada a la función se realiza de la forma habitual, indicando su nombre seguido de los paréntesis y de forma opcional, incluyendo todos los parámetros que se necesiten.

```
<input type="button" value="Click Aquí"  
      onclick="MostrarMensaje() ;" />
```

# Demo