

DESARROLLO WEB CON MEAN

Curso: DESARROLLO WEB CON MEAN (WEB FULL STACK DEVELOPER)

Módulo 1: Javascript

Introducción a Javascript

El HTML5 (HyperText Markup Language, versión 5) es la quinta revisión del lenguaje de programación “básico” de la World Wide Web, el HTML

Esta nueva versión pretende remplazar al actual (X)HTML, corrigiendo problemas con los que los desarrolladores web se encuentran, así como rediseñar el código actualizándolo a nuevas necesidades que demanda la web de hoy en día.

METAS:

- Documentos se comporten igual en todos los navegadores
- Documentar y estandarizar extensiones útiles
- Desarrollar prácticas de nuevas características
- Asegurar la compatibilidad hacia atrás
- Definir un robusto tratamiento de errores

Módulo 1: Javascript

1. SINTAXIS EN JAVASCRIPT
2. VARIABLES
3. PALABRAS RESERVADAS
4. TIPOS DE VARIABLES
5. OPERADORES
6. OBJETOS NATIVOS
7. FUNCIONES
8. PROPIEDADES

Módulo 1: Javascript - Sintaxis

- SIMILAR A OTROS LENGUAJES
- NO CUENTA ESPACIOS NI SALTOS DE LINEA
- DISTINGUE MAYUSCULAS CON MINUSCULAS
- NO SE DEFINE EL TIPO DE VARIABLES
- NO ES OBLIGATORIO TERMINAR CON ;
- PERMITE COMENTARIOS

```
/**  
 * Este es un comentario  
 */  
  
// Este es otro comentario|
```

Módulo 1: Javascript - Variables

- SE DEFINEN MEDIANTE VAR

```
var mivariable;  
var mivariable2;  
var mivariable3=10;  
var mivariable4=10,mivariable5=11;
```

- NO NOMBRAR CON \$ NI _
- NO HAY OBLIGACIÓN DE INICIALIZAR LAS VARIABLES
- SI NO SE INDICA VAR, SE CREA UNA VARIABLE GLOBAL

Módulo 1: Javascript – Palabras reservadas

- RESERVADAS
 - ACTUALES
 - BREAK, ELSE, NEW, VAR, CASE, FINALLY, RETURN, VOID, CATCH, FOR, SWITCH,
 - WHILE, CONTINUE, FUNCTION, THIS, WITH, DEFAULT, IF, THROW, DELETE, IN, TRY, DO,
 - INSTANCEOF, TYPEOF
 - FUTURAS
 - ABSTRACT, ENUM, INT, SHORT, BOOLEAN, EXPORT,
 - INTERFACE, STATIC, BYTE, EXTENDS, LONG, SUPER,
 - CHAR, FINAL, NATIVE, SYNCHRONIZED, CLASS, FLOAT, PACKAGE, THROWS, CONST, GOTO, PRIVATE,
 - TRANSIENT, DEBUGGER, IMPLEMENTS, PROTECTED,
 - VOLATILE, DOUBLE, IMPORT, PUBLIC

Módulo 1: Javascript – Tipos de Variables

- TIPOS PRIMITIVOS
- TIPO UNDEFINED
- TIPO NULL
- TIPO BOOLEAN
- TIPO NUMÉRICO

Módulo 1: Javascript – Tipos de Variables

– Constantes predefinidas en JS

- Math.E
- Math.LN2
- Math.LN10
- Math.LOG2E
- Math.LOG10E
- Math.PI
- Math.SQRT1_2
- Math.SQRT2

Módulo 1: Javascript – Tipos de Variables

- TIPO STRING
- TIPO OBJECT
- TIPO OBJECT BOOLEAN
- TIPO NUMBER OBJETO
- TIPO STRING OBJETO

Módulo 1: Javascript – Operadores

- ASIGNACIÓN «=»
- INCREMENTO Y DECREMENTO
- NEGACIÓN
- AND / OR
- MATEMÁTICOS
- RELACIONALES

Módulo 1: Javascript – Objetos nativos

- EXISTEN CLASES PRECONSTRUIDAS PARA SU USO EN JAVASCRIPT COMO
 - ARRAY
 - DATE
 - FUNCTION

Módulo 1: Javascript – Expresiones regulares

- PATRONES DE COINCIDENCIAS
- EJEMPLO WILDCARDS DE DOS
 - ? Y *
- LAS EXPRESIONES REGULARES SON UNA FORMA SENCILLA PARA
 - MANIPULAR DATOS
 - BUSQUEDA
 - REEMPLAZO DE STRINGS

Módulo 1: Javascript – Expresiones regulares

- SINTAXIS
 - PODEMOS CREAR LAS EXPRESIONES POR OBJETO O VARIABLE
 - `Var reg = /patron/flags`
 - `New RegExp("patron","flags");`
 - FLAGS
 - BUSQUEDAS GLOBALES
 - DEVUELVE UN ARRAY CON LAS OCURRENCIAS
 - IGNORE CASE
 - NO FUNCIONA CON CARACTERES EXTENDIDOS
 - ENTRADA MULTILINEA
 - LOS TAGS DE INICIO Y FIN SIRVEN POR CADA LINEA

Módulo 1: Javascript – Expresiones regulares

- PATRONES
 - \ ESCAPE DE LITERALES
 - {N} EXACTAMENTE N VECES
 - {N,} N O MAS VECES
 - {N,M} DE N A M VECES
 - * EQUIVALENTE A {0,}
 - + EQUIVALENTE A {1,}
 - ? EQUIVALENTE A {0,1}

Módulo 1: Javascript – Expresiones regulares

- (PATRON) CAPTURA LA COINCIDENCIA
- (?:PATRON) INDICA LA COINCIDENCIA PERO NO LA CAPTURA
- | OR (a|b) CAPTURA A Y B, NO TIENEN QUE ESTAR SEGUIDOS
- [CARACTERES] INDICA LOS CARACTERES COINCIDENTES
 - [a-d] BUSCA “A B C D”
- [^CARACTERES] INDICA SI NO ES COINCIDENTE
 - [^a-d] BUSCA CUALQUIERA EXCEPTO “A B C D”

Módulo 1: Javascript – Expresiones regulares

- CARACTERES ESPECIALES
 - ^ INICIO DE LA ENTRADA
 - \$ FIN DE LA ENTRADA
 - . CUALQUIER CARÁCTER MENOS SALTO DE LINEA

Módulo 1: Javascript – Expresiones regulares

- CARACTERES DE ESCAPE
 - \f FORM-FEED
 - \r RETURN
 - \n SALTO DE LINEA
 - \t TABULADOR HORIZONTAL
 - \v TABULADOR VERTICAL
 - \0 CARÁCTER NULO
 - \s ESPACIO EN BLANCO
 - \S CUALQUIERA EXCEPTO ESPACIO EN BLANCO

Módulo 1: Javascript – Expresiones regulares

- \w CARACTERES ALFANUMERICOS
- \W TODO MENOS CARACTERES ALFANUMERICOS
- \d DIGITOS
- \D TODO MENOS DIGITOS
- \cX captura control + carácter X
- ...

Módulo 1: Javascript – Expresiones regulares

- USO EN JAVASCRIPT
 - .exec(STRING)
 - DEVUELVE LA INFORMACIÓN COINCIDENTE
 - .test(STRING)
 - DEVUELVE EL BOOLEANO DE LA COINCIDENCIA
 - .match(PATRON)
 - CON g DEVUELVE EL ARRAY CON COINCIDENCIAS, O NULL
 - .search(PATRON)
 - DEVUELVE LA POSICIÓN DE LA PRIMERA COINCIDENCIA

Módulo 1: Javascript – Expresiones regulares

- `.replace(PATRON,STRING)`
 - DEVUELVE EL STRING EDITADO
- `.split(PATRON)`
 - STRING TRANSFORMADO EN UN ARRAY

JAVASCRIPT EJERCICIO

- REALICE UN SCRIPT EN JAVASCRIPT QUE VALIDE UN TEXTO CON FORMATO NIF, CIF, NIE. LA VALIDACIÓN DEBE SER SOLO DE NÚMEROS Y CARACTERES, NO ES NECESARIO QUE REALICE VALIDACIONES COMPLEJAS.
- DEBE MOSTRAR EL RESULTADO POR PANTALLA INDICANDO SI EL VALOR ES CORRECTO O NO

Módulo 2: CSS – Selectores

Para crear diseños web profesionales, es imprescindible conocer y dominar los selectores de CSS. Como se vio en el capítulo anterior, una regla de CSS está formada por una parte llamada "selector" y otra parte llamada "declaración".

La declaración indica *"qué hay que hacer"* y el selector indica *"a quién hay que hacérselo"*. Por lo tanto, los selectores son imprescindibles para aplicar de forma correcta los estilos CSS en una página.

Módulo 2: CSS – Selectores

Selector universal:

```
* {  
  margin: 0;  
  padding: 0;  
}
```

Selector etiqueta

```
h2 {  
  color: #8A8E27;  
  font-weight: normal;  
  font-family: Arial, Helvetica, sans-serif;  
}
```

Módulo 2: CSS – Selectores

Selector descendente:

```
p span { color: red; }  
<p>  
  ...  
  <span>texto1</span>  
  ...  
  <a href="">...<span>texto2</span></a>  
  ...  
</p>
```

Selector de clase:

```
.destacado { color: red; }  
  
<p class="destacado">Lorem ipsum dolor sit amet...</p>
```


Módulo 2: CSS – Selectores

Selector de ID:

```
#destacado { color: red; }
```

```
<p id="destacado">Segundo párrafo</p>
```

Módulo 2: CSS – Medidas/colores

Unidades de medida

in, pulgadas ("*inches*", en inglés). Una pulgada equivale a 2.54 centímetros.

cm, centímetros.

mm, milímetros.

pt, puntos. Un punto equivale a 1 pulgada/72, es decir, unos 0.35 milímetros.

pc, picas. Una pica equivale a 12 puntos, es decir, unos 4.23 milímetros.

Módulo 2: CSS – Medidas/colores

Unidades relativas

em, (no confundir con la etiqueta de HTML) relativa respecto del tamaño de letra del elemento.

ex, relativa respecto de la altura de la letra x ("*equis minúscula*") del tipo y tamaño de letra del elemento.

px, (píxel) relativa respecto de la resolución de la pantalla del dispositivo en el que se visualiza la página HTML.

Módulo 2: CSS – Medidas/colores

Porcentajes

Los porcentajes se pueden utilizar por ejemplo para establecer el valor del tamaño de letra de los elementos:

```
body { font-size: 1em; }  
h1 { font-size: 200%; }  
h2 { font-size: 150%; }
```

Módulo 2: CSS – Medidas/colores

Colores

Se pueden indicar de cinco formas diferentes: palabras clave, colores del sistema, RGB hexadecimal, RGB numérico y RGB porcentual.

Palabras clave:

aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, orange, purple, red, silver, teal, white, yellow

RGB Decimal

```
p { color: rgb(71, 98, 176); }
```

RGB porcentual

```
p { color: rgb(27%, 38%, 69%); }
```

Módulo 2: CSS – Medidas/colores

Colores

RGB Hexadecimal

```
p { color: #4762B0; }
```

Módulo 2: CSS – Cajas

Las cajas de una página se crean automáticamente. Cada vez que se inserta una etiqueta HTML, se crea una nueva caja rectangular que encierra los contenidos de ese elemento.

Las partes que componen cada caja y su orden de visualización desde el punto de vista del usuario son las siguientes:

Contenido (*content*): se trata del contenido HTML del elemento (las palabras de un párrafo, una imagen, el texto de una lista de elementos, etc.)

Relleno (*padding*): espacio libre opcional existente entre el contenido y el borde.

Borde (*border*): línea que encierra completamente el contenido y su relleno.

Imagen de fondo (*background image*): imagen que se muestra por detrás del contenido y el espacio de relleno.

Color de fondo (*background color*): color que se muestra por detrás del contenido y el espacio de relleno.

Margen (*margin*): separación opcional existente entre la caja y el resto de cajas adyacentes.

Módulo 2: CSS – Posicionamiento

Para cumplir con el modelo de cajas presentado en el capítulo anterior, los navegadores crean una caja para representar a cada elemento de la página HTML. Los factores que se tienen en cuenta para generar cada caja son:

- Las propiedades width y height de la caja (si están establecidas).
- El tipo de cada elemento HTML (elemento de bloque o elemento en línea).
- Posicionamiento de la caja (normal, relativo, absoluto, fijo o flotante).
- Las relaciones entre elementos (dónde se encuentra cada elemento, elementos descendientes, etc.)
- Otro tipo de información, como por ejemplo el tamaño de las imágenes y el tamaño de la ventana del navegador.

Módulo 2: CSS – Texto

CSS define numerosas propiedades para modificar la apariencia del texto. A pesar de que no dispone de tantas posibilidades como los lenguajes y programas específicos para crear documentos impresos, CSS permite aplicar estilos complejos y muy variados al texto de las páginas web.

Algunas de las propiedades que se pueden utilizar son:

Color, font-family, font-size, font-weight, font-style, font-variant, font, text-align, line-height, text-decoration, text-transform, vertical-align, text-indent, letter-spacing, etc.

Módulo 2: CSS – Enlaces

Se suele utilizar text-decoration y font-weight para modificar los estilos de los enlaces.

Podemos usar distintos estilos para cada psedudo estado del enlace:

- :link, aplica estilos a los enlaces que apuntan a páginas o recursos que aún no han sido visitados por el usuario.
- :visited, aplica estilos a los enlaces que apuntan a recursos que han sido visitados anteriormente por el usuario. El historial de enlaces visitados se borra automáticamente cada cierto tiempo y el usuario también puede borrarlo manualmente.
- :hover, aplica estilos al enlace sobre el que el usuario ha posicionado el puntero del ratón.
- :active, aplica estilos al enlace que está pinchando el usuario. Los estilos sólo se aplican desde que el usuario pincha el botón del ratón hasta que lo suelta, por lo que suelen ser unas pocas décimas de segundo.

Módulo 2: CSS – Imágenes

Se suele utilizar width y height para modificar los estilos de las imágenes.

```
#destacada {  
  width: 120px;  
  height: 250px;  
}
```

Si se usan mal, se deforma la imagen.

No es recomendable el uso de CSS para modificación de la imagen, ya que por imagen se suele crear una regla.

Módulo 2: CSS – Listas

Se pueden definir estilos en las listas:

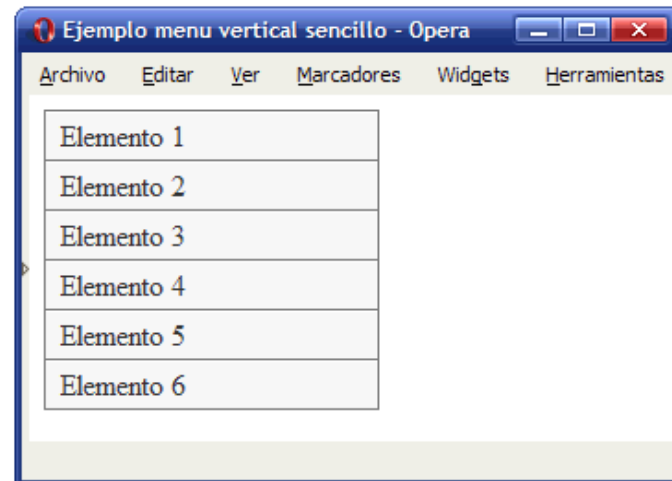
Algunas de las propiedades son:

List-style-type, list-style-position, list-style-image, list-style

En las webs se suelen usar listas ordenadas: ``

Un ejemplo de definición:

```
ul.menu li a {  
  color: #333;  
  display: block;  
  padding: .2em 0 .2em .5em;  
  text-decoration: none;  
}
```



Módulo 2: CSS – Tablas

Los estilos de las tablas se deben definir en CSS

Algunas propiedades de definición son:

Border-collapse, border-spacing, empty-cells, caption-side, etc.

Módulo 2: CSS – Layout

El objetivo de mantener la disposición de la web:

Mantenimiento: una página diseñada exclusivamente con CSS es mucho más fácil de mantener que una página diseñada con tablas. Cambiar el aspecto de una página creada con CSS es tan fácil como modificar unas pocas reglas en las hojas de estilos. Sin embargo, realizar la misma modificación en una página creada con tablas supone un esfuerzo muy superior y es más probable cometer errores.

Accesibilidad: las páginas creadas con CSS son más accesibles que las páginas diseñadas con tablas. De hecho, los navegadores que utilizan las personas discapacitadas (en especial las personas invidentes) pueden tener dificultades con la estructura de las páginas complejas creadas con tablas HTML. No obstante, diseñar una página web exclusivamente con CSS no garantiza que la página sea accesible.

Módulo 2: CSS – Layout

Velocidad de carga: el código HTML de una página diseñada con tablas es mucho mayor que el código de la misma página diseñada exclusivamente con CSS, por lo que tarda más tiempo en descargarse. En cualquier caso, si el usuario accede al sitio con una conexión de banda ancha y la página es de un tamaño medio o reducido, las diferencias son casi imperceptibles.

Semántica: aunque resulta obvio, las tablas HTML sólo se deben utilizar para mostrar datos cuya información sólo se entiende en forma de filas y columnas. Utilizar tablas para crear la estructura completa de una página es tan absurdo como utilizar por ejemplo la etiqueta `` para crear párrafos de texto.

Módulo 2: CSS – Novedades CSS3

- CSS: DEFINICIÓN DE ESTILOS Y APARIENCIA
 - LISTA DE PROPIEDADES
 - Bordes
 - border-color
 - border-image
 - border-radius
 - box-shadow
 - Fondos
 - background-origin
 - background-clip
 - background-size

Módulo 2: CSS – Novedades CSS3

- Color
 - colores HSL
 - colores HSLA
 - colores RGBA
 - Opacidad
- Texto
 - text-shadow
 - text-overflow
 - Rotura de palabras largas
 - Web Fonts
- Interfaz
 - box-sizing
 - resize
 - outline
 - nav-top, nav-right, nav-bottom, nav-left

Módulo 2: CSS – Novedades CSS3

- Selectores
 - Selectores por atributos
- Modelo de caja básico
 - overflow-x, overflow-y
- Degradados CSS3
 - Degradados lineales
 - Degradados radiales
 - Degradados lineales de repetición
 - Degradados radiales de repetición

Módulo 2: CSS – Novedades CSS3

- Otros
 - media queries
 - creación de múltiples columnas de texto
 - propiedades orientadas a discurso o lectura automática de páginas web
 - animaciones CSS3

Módulo 2: CSS – Novedades CSS3

- CSS3 ES MODULAR, Y FORMADO POR SUB-ESPECIFICACIONES
- CSS3 SE BASA EN SU VERSIÓN PREVIA, PERO VIENE CARGADO DE NUEVAS FUNCIONALIDADES
 - SELECTORES, ESQUINAS REDONDEADAS, CAJAS Y SOMBRAS DE TEXTO, TRANSICIONES, ANIMACIONES, TRANSFORMACIONES, ETC.

Módulo 2: CSS – Novedades CSS3

- HTML TODAVÍA ESTÁ EN DESARROLLO
 - LA MAYORIA DE LOS NAVEGADORES SE COMPORTAN CORRECTAMENTE AL ACCEDER A ELEMENTOS NO RECONOCIDOS.
 - ES RECOMENDABLE, HASTA QUE EL ESTÁNDAR SEA IMPLEMENTADO CAMBIAR LA REGLA `display:inline` POR `display:block`
 - `article, aside, footer, header, hgroup, nav, section {`
 - `Display: block;`
 - `}`

Módulo 2: CSS – EJERCICIO

- INDIQUE CUALES SON LOS SELECTORES PARA LOS SIGUIENTES CAMPOS marcados en exclusiva
- `<h1></h1>`
- `<h1 class="nuevo"><h1></h1>`
- ``
- `<h1></h1>`
- `<p id="parrafo"></p><p></p>`
- `<p class="parrafo"></p><strong class="parrafo">`