Tarea

Los lenguajes Script sirven principalmente para añadir contenido dinámico a las páginas Web. El contenido dinámico es, básicamente, aquel contenido de una página Web (gráficos, textos y demás) que puede cambiar en el tiempo, según las iteraciones que realiza el usuario a lo largo de su ejecución.

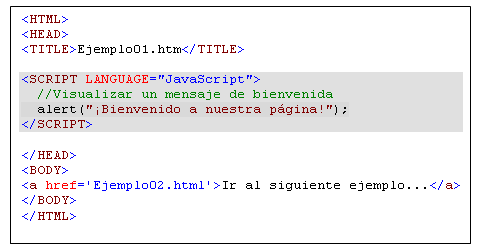
En una página Web HTML que no contiene código Script el contenido no puede cambiar a lo largo del tiempo, es por eso que el contenido que contiene se le llama estático.

Javascript es un lenguaje de programación Script, creado por la empresa Netscape, y cuya principal función es incluir contenido dinámico a las páginas web

**Ejemplo**

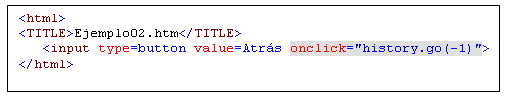
Visualizar por pantalla un mensaje de bienvenida

**ejemplo 1**



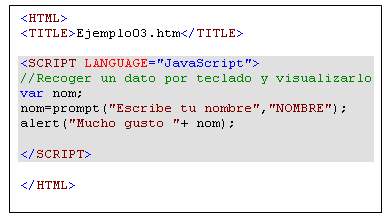
**Ejemplo 1**  . Visualizar por pantalla un mensaje de bienvenida a la página Web.

**Botón para volver hacia atrás**



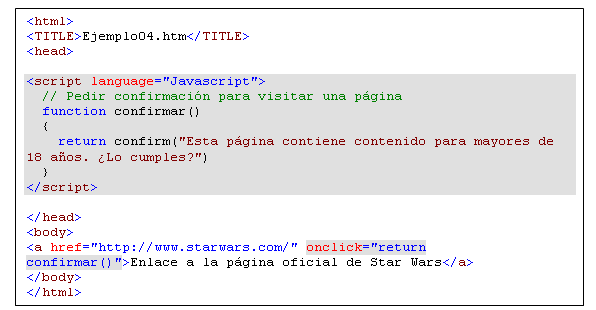
**Ejemplo 2.**Botón que al ser pulsado vuelve a la página    anterior de la que hubiera en ese momento.

**Recoger    un dato por teclado y visualizarlo**

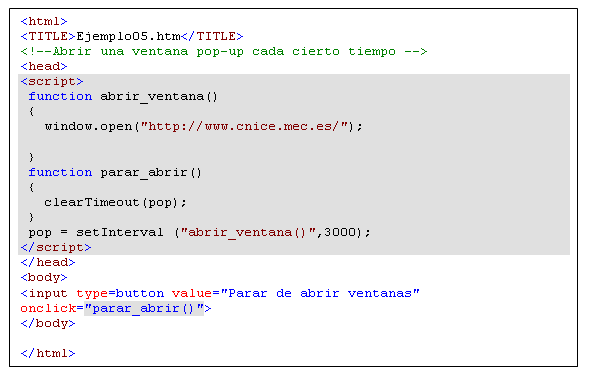


**Ejemplo 3.**Visualiza    por pantalla el nombre introducido por teclado

### **Pedir    confirmación para visitar una página**

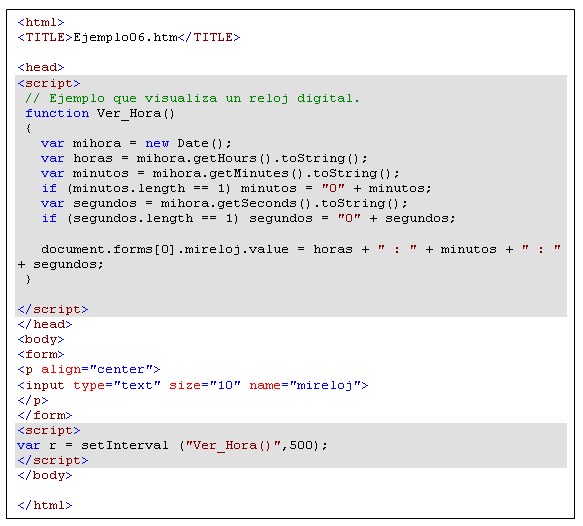


**Ejemplo 4.**  Ejemplo que    pide confirmación para visitar una página.

**Abrir    una ventana pop-up cada cierto tiempo** 

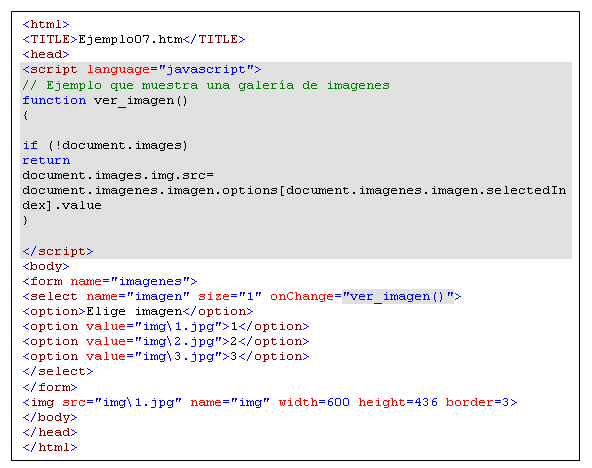
**Ejemplo 5.**Ejemplo que    abre una ventana emergente cada 3 segundos. Pulsando el botón no se abren más.

**Visualizar    un reloj digital en la página web**

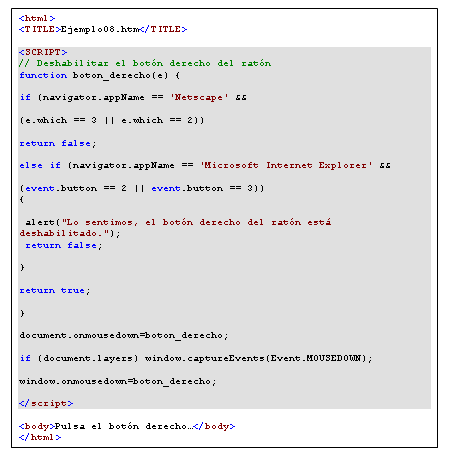


**Ejemplo 6.**  Ejemplo que    visualiza un reloj digital.

**Crear una galería de imágenes**

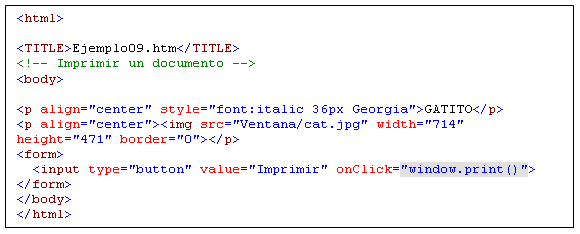


**Ejemplo 7.**Ejemplo que  muestra una galería de imágenes

**Deshabilitar    el botón derecho del ratón**

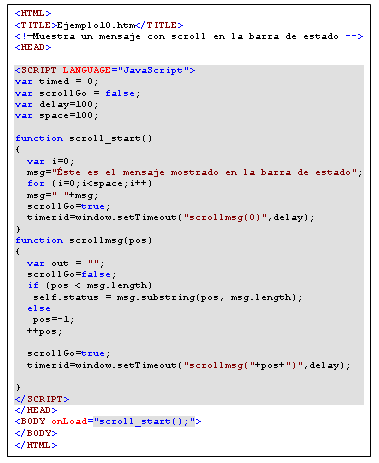
**Ejemplo 8.**Ejemplo que    deshabilita el botón derecho del ratón.

**Imprimir    una página Web**



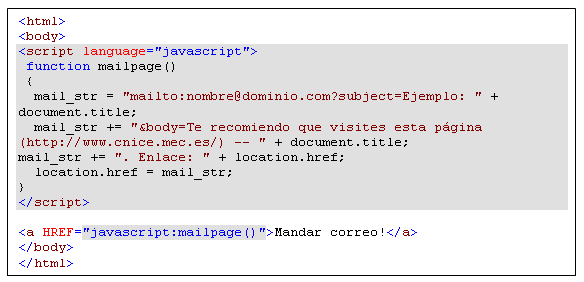
**Ejemplo 9.**Ejemplo que  al pulsar un botón se imprime la página Web.

**Mostrar    mensajes en movimiento en la barra de estado del navegador**



**Ejemplo 10.**Ejemplo que    muestra un mensaje en movimiento en la barra de estado del navegador.

**Crear un correo electrónico**



   FOR

El bucle for es una estructura de control en programación en la que se puede indicar de antemano el número máximo de iteraciones

El funcionamiento de la estructura for es el siguiente:

1) Se evalúa la expresión de inicialización.

2) Se evalúa la condición. Si el resultado es falso, termina la ejecución de la estructura for. Si es verdadero, sigue por el paso 3.

3) Se ejecuta el bloque de instrucciones dentro de la estructura.

4) Se evalúa la expresión final, que normalmente altera la variable de control, y se vuelve al paso segundo.

Ejemplo.

Para acabar con el for sería conveniente hacer una mención sobre la existencia de un operador coma (,), que se utiliza principalmente en esta sentencia. Este operador me permite que aparezcan dos expresiones en situaciones en donde sólo se utilizaría una expresión ordinariamente. Por tanto, sería posible encontrarse algo como:

for (expresion 1a, expresion 1b; expresion 2; expresion 3) sentencia

Como vemos la expresion 1a se encuentra separada de la expresion 1b por una coma, en lugar de aparecer una sola expresión, como habitualmente.

for (expresion 1a; expresion 2; expresion 3a, expresion 3b) sentencia

Como vemos la expresion 3a se encuentra separada de la expresion 3b por una coma, en lugar de aparecer una sola expresión, como habitualmente.

Ejemplo.

Ejemplos de estructura for:

Ejemplo 1:

#include <stdio.h>

main() {

int n,i;

int suma;

printf("Suma desde 0 hasta n.\n");

printf("Introduce un número n: ");

scanf("%i", &n);

for(i=suma=0;i<=n;i++)

suma=suma+i;

printf("Suma desde 0 hasta %d=%d\n", n, suma);

}

Bucle while

El bucle while o bucle mientras es un ciclo repetitivo basado en los resultados de una expresión lógica; se encuentra en la mayoría de los lenguajes de programación estructurados. El propósito es repetir un bloque de código mientras una condición se mantenga verdadera.​

Un bucle while itera o repite un bloque de código mientras una condición tiene el valor true. Se puede escribir de la siguiente manera:

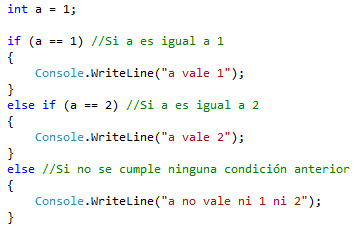
while (condicion)  
{  
 instrucciones;  
}

Como ves, hemos hablado también de iteraciones y bucles, pero siempre nos referimos a lo mismo. Todos los bucles tienen:

* la condicion que se evalúa a true o false, y se hace en cada repetición/iteración del bucle.
* la instruccion o instrucciones que representa las líneas de código se ejecutan si la condicion es true.

IF

La palabra “if“‘ es el ‘si‘ condicional en inglés. La utilizamos en frases como: “saldría a pasear si no lloviera”. Es decir, si se cumpliera la condición “no llover”, entonces realizaríamos la acción “salir a pasear”. En Javascript ( y en general en los lenguajes de programación) esta instrucción “if” juega el mismo papel.



**CASE**

La sentencia CASE selecciona una entre un conjunto de sentencias para ejecutar. La función CASE selecciona una entre un conjunto de expresiones a evaluar y devuelve el valor de retorno de la expresión seleccionada como su valor.

