

Presentación

Desarrollo de Tecnologías Web: Tecnologías en el Cliente (DTWTC)

*Master Universitario en Ingeniería de Servicios y
Aplicaciones Web*

Índice

- Profesor responsable
- Guía docente y aula virtual
- Planificación
- Introducción
 - Breve repaso de HTML
 - Breve repaso de CSS
 - Breve repaso de JavaScript
- Motivación: evolución de tecnologías
- Instalación del software
- Un ejemplo

Presentación

- Profesor responsable
 - Ramón Vte. Cirilo Gimeno
 - Ramon.V.Cirilo@uv.es
 - 96.354.35.69
 - Tutorías: Miércoles de 8:30 a 11:30
Tutorías electrónicas
 - Despacho: 1.1.8 – IRTIC (Instituto de Robótica y Tecnologías de la Información y las Comunicaciones)

Guía docente y aula virtual

- Guía docente

- <http://www.uv.es/isaw> > Programa del máster
> Plan de estudios > Ver ficha > Guía docente



- Aula Virtual

- <http://aulavirtual.uv.es>



Planificación

Jornada	Tema	Actividades presenciales	Actividades no presenciales	Tareas	Tutorías
S1	T0. Presentación. Introducción a las Aplicaciones Web en el lado del cliente	Presentación de la asignatura / Instalación del software / Revisión de conceptos básicos: HTML, CSS, JavaScript / Evolución de las tecnologías del cliente / Ejemplo de uso del software de desarrollo			
S2	T1. HTML 5: elementos avanzados.	HTML5: elementos avanzados / Ejemplos de uso	Tarea #1 (25%)		Viernes 15:00-17:00
S3	T1. CSS3 y sus novedades	CSS3: novedades y mejoras / Ejemplo guiado	Tarea #1 (25%)		
S4	T1. Clase práctica	Clase práctica dedicada a la realización de la Tarea #1	Tarea #1 (25%)	Entrega Tarea #1	
S5	T2. Javascript: gestión de documentos a través del DOM	DOM para HTML y XML / BOM - Browser Object Model / Gestión del DOM a través de Javascript / Ejemplo guiado	Tarea #2 (37,5%)		
S6	T2. Javascript: formularios y visualización	Control de formularios a través de Javascript / Gestión de la visualización a través de Javascript / Ejemplo guiado	Tarea #2 (37,5%)		
S7	T2. La tecnología AJAX	XML y Javascript / Concepto y funcionalidad de AJAX / Ejemplo guiado	Tarea #2 (37,5%)		Viernes 15:00-17:00
S8	T2. Clase práctica	Clase práctica dedicada a la realización de la Tarea #2	Tarea #2 (37,5%)	Entrega Tarea #2	
S9	T3. JQuery y frameworks del cliente	El framework JQuery / JSON / Otros frameworks de cliente / Ejemplo guiado	Tarea #3 (27,5%)		
S10	T3. Clase práctica	Clase práctica dedicada a la realización de la Tarea #3	Tarea #3 (27,5%)	Entrega Tarea #3	

Planificación

■ Evaluación

- Actividades propuestas en clases teóricas y prácticas: 90%
 - Tarea 1. HTML5 + CSS3 (25%)
 - Tarea 2. JavaScript + AJAX (37,5%)
 - Tarea 3. JQUERY + Frameworks (27,5%)
- Asistencia a clases obligatoria (10%)
- En caso de no cumplir alguna parte, se evalúa con un examen final con nota máxima de 8.

■ Tutorías

- Clases prácticas preparadas con antelación.

Introducción. Revisión de HTML

- HTML: **H**yper **T**ext **M**arkup **L**anguage
- HTML es un lenguaje de marcas derivado de SGML
- Las marcas se conocen como “etiquetas”
- Las etiquetas describen el contenido y la visualización del documento HTML
- Un documento HTML tiene formato de texto plano
- Un documento HTML bien formado sigue unas simples reglas sintácticas

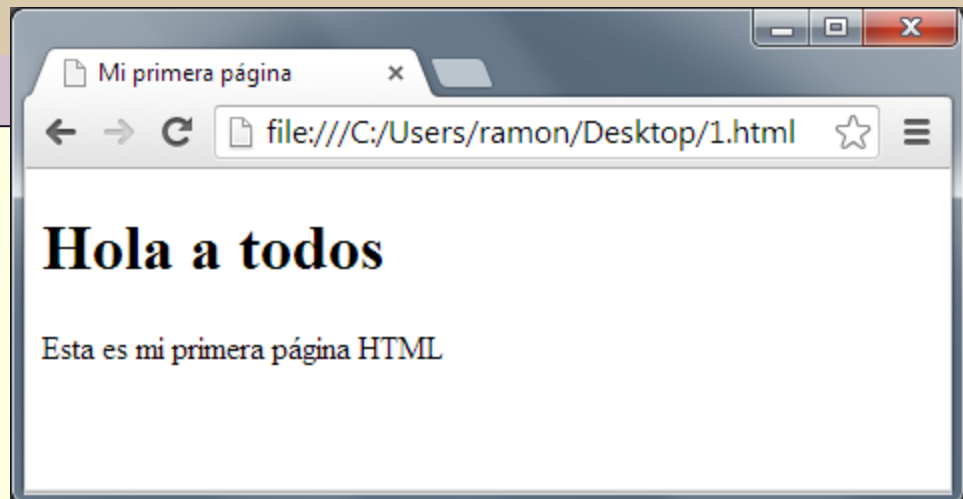
Un ejemplo de HTML

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
  <title>Mi primera página</title>
</head>
<body>





<h1>Hola a todos</h1>

<p>Esta es mi primera página HTML</p>

</body>
</html>
```



Etiquetas

- Son palabras clave que van entre los símbolos "<" y ">" (p. ej. <html>). Son los **elementos** del lenguaje HTML
- Van emparejadas, se abren y se cierran:
<html> ... </html>
- Pueden tener calificadores, que son parejas atributo-valor:
<body bgcolor="white">
- Se pueden anidar, sin solaparse:
 ... <i> ... </i> ...  
 ... <i> </i>  

HTML. Componentes

■ Etiquetas

```
<TABLE>  
    <TR> <TD>A</TD> <TD>A</TD> </TR>  
</TABLE>
```

■ Atributos

```
<HR SIZE="20" WIDTH="50%" NOSHADE>
```

■ Eventos

```
<BODY onLoad="alert( 'Bienvenido' );">
```

Introducción. Revisión de CSS

- CSS significa **C**ascading **S**tyle **S**heets
- Se utiliza para definir la manera en la que se van a visualizar los elementos HTML, y por ende, el documento completo.
- Presentan un intento de separar contenido de visualización (si se consigue o no es otra discusión)
- En combinación con la versión HTML 4.0 alcanzan su máximo nivel de utilización

Un ejemplo de CSS

```
/* standard body */
body {
    margin: 0 auto;
    width: 960px;
    font: 14px/20px "Trebuchet MS", Verdana, sans-serif;
    background: #271408 url("images/bodybkg.gif") repeat;
}

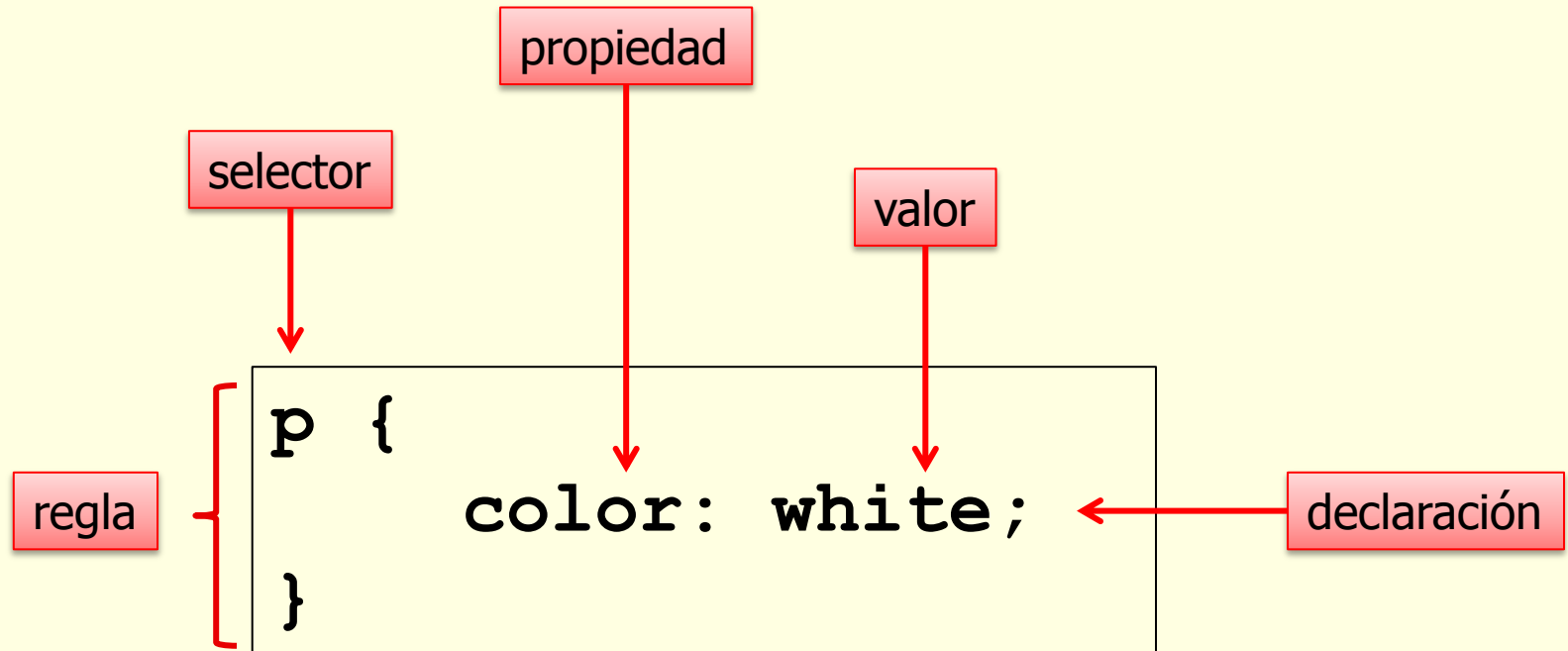
/* standard alignments */
.alignleft {
    float: left;
}

img.alignleft {
    margin: 5px 10px 0 0;
}

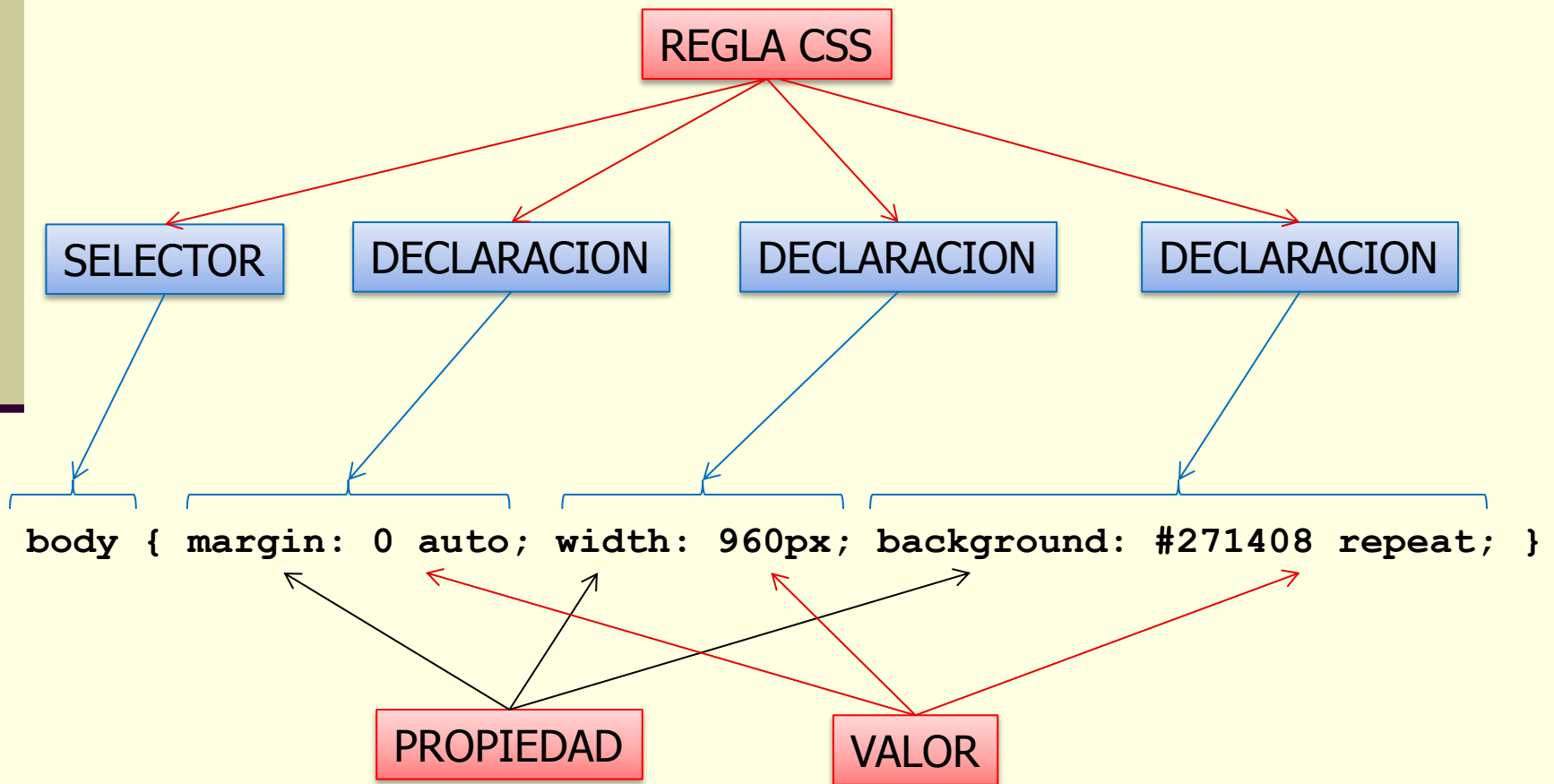
.alignright {
    float: right;
}

img.alignright {
    margin: 5px 0 0 10px;
}
```

CSS. Sintaxis



CSS. Sintaxis



CSS. Incorporación a HTML

■ Externo

```
<head>
<link rel="stylesheet" type="text/css" href="miestilo.css">
</head>
```

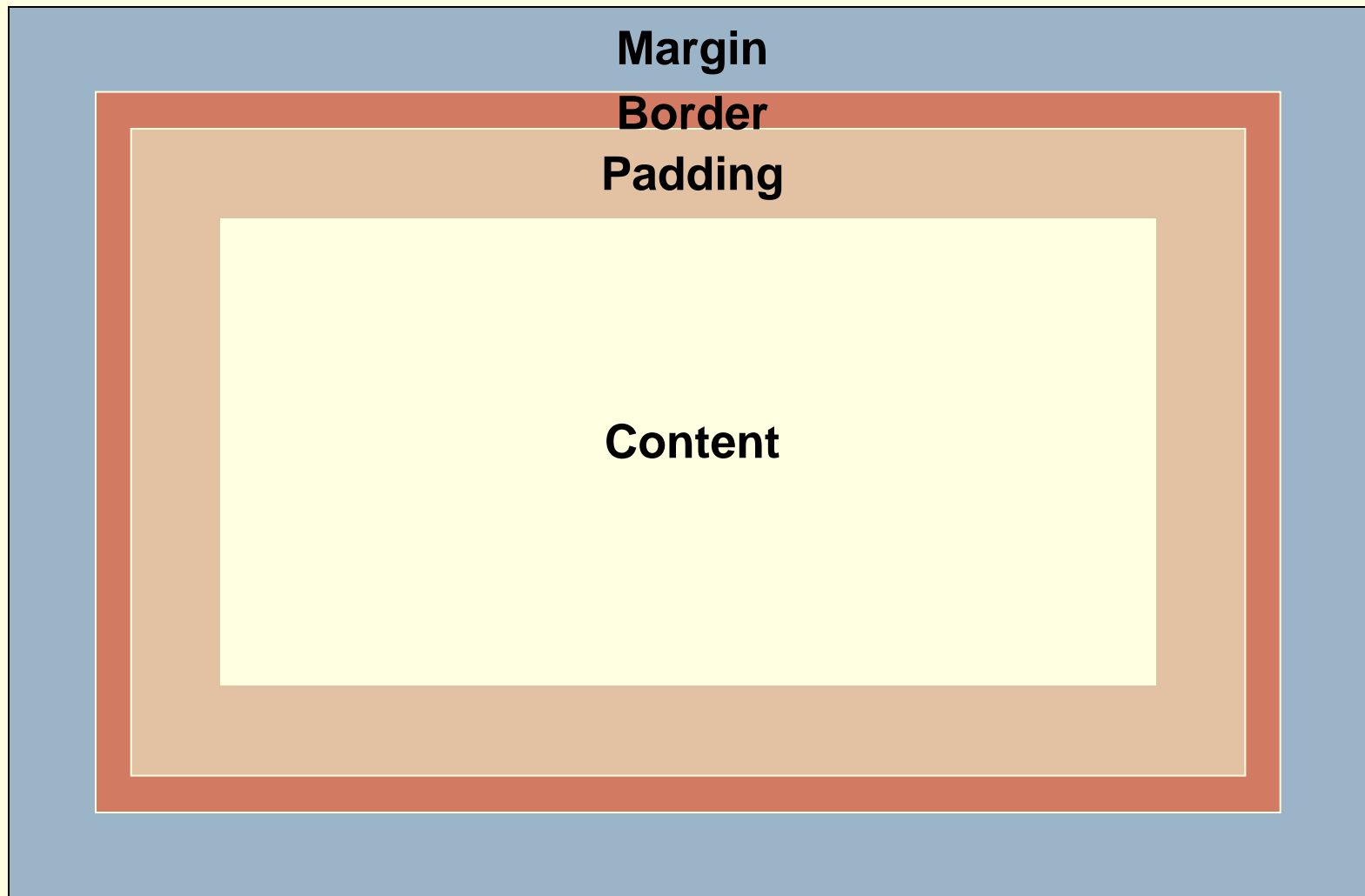
■ Interno

```
<head>
  <style>
    hr {color:blue;}
    p {margin:20px;}
    body {background-image:url("imagenes/fondo27.gif");}
  </style>
</head>
```

■ En linea

```
<p style="color:blue; margin:20px">Esto es un párrafo.</p>
```

CSS. Box model



CSS. Selectores

SELECTOR	DESCRIPCION	EJEMPLO
.	Clase	.cabecera
#	Id	#primero
*	Todos los elementos	*
elemento	Un elemento	p
elemento, elemento	Varios elementos	p, td
elemento elemento	Elemento dentro de elemento	div p
elemento>elemento	Elemento cuyo padre es otro elemento	div>p
elemento+elemento	Elemento tras otro elemento	div+p
[atributo]	Atributos	[href]
[atributo=valor]	Atributos con valor	[target=_blank]
[atributo~=valor]	Atributos que contengan un valor	[href~=uv]
[atributo =valor]	Atributos que empiecen por un valor	[title =Dis]
:link :visited :active :hover	Enlaces	a:active
:focus	Foco de un elemento	[input:focus]
:first-letter :first-line :first-child	Orden de elementos	p:first-letter
:before :after	Inserción de contenido	p:before

CSS. Selectores

■ Combinación y anidación

```
<style>
```

```
p
```

← Estilo para un párrafo

```
{  
  color:blue;  
  text-align:center;  
}
```

```
.marked
```

← Estilo para la clase

```
{  
  background-color:red;  
}
```

```
.marked p
```

← Estilo para un párrafo que se encuentre dentro de la clase en cuestión

```
{  
  color:white;  
}
```

```
</style>
```

CSS. Estilos

ESTILO	PROPIEDADES	EJEMPLO
Backgrounds	background-color, background-image, background-repeat, background-attachment, background-position	body {background-color:#ffffff;}
Texto	color, text-align, text-decoration, text-transform, text-indent	p {text-indent:50px;}
Fuentes	font, font-family, font-style, font-size, font-variant, font-weight	p {font-family:"Times New Roman", Times, serif;}
Listas	list-style, list-style-type, list-style-position, list-style-image	ul { list-style-image: url('cuadro.gif');}
Tablas	border, border-collapse, width, height, text-align, vertical-align, padding, color, background-color,	p {border-width:5px;}
Bordes	border-style, border-width, border-color [–top –right –bottom –left]	p {border-style:solid;}
Outline	outline, outline-color, outline-style, outline-width	p {outline-style:dotted;}
Margin	margin [–top –right –bottom –left]	p {margin-top:100px;}
Padding	padding [–top –right –bottom –left]	p {padding-left:150px;}
Display	display (none inline block)	span {display:block;}
Position	position (fixed relative absolute)	h2 { position:absolute; left:100px; top:150px; }

Introducción. Revisión de JavaScript

- Diseñado para añadir interactividad a las páginas HTML
- Es un lenguaje de script
- Es un lenguaje interpretado
- Pseudo-Orientado a Objetos
- Se embebe directamente en páginas HTML

- En un fichero separado:

```
<script src="programa.js"></script>
```

- Como valor de un atributo de tipo evento:

```
<body OnUnload="alert('Hasta Luego!')">
```

- Como destino de un hiperenlace:

```
<a href="javascript:window.open('hist.html', 'historia',  
'width=600, height=500')">Historia</a>
```

- <http://www.w3schools.com/js/default.asp>
- <http://www.librosweb.es/>

Javascript

Uso y aplicación del lenguaje

Javascript

- La principal complejidad del lenguaje consiste en comprender el modo de usarlo en el contexto de una aplicación web
- Usos habituales:
 - Control de formularios
 - Control de visualización
 - Interactividad con el usuario
 - Dinamismo de contenidos
 - Creación de mini-aplicaciones

Javascript

- Preguntas a responder
 - ¿Cómo se crea código javascript?
 - ¿Qué puedo hacer con código javascript?
 - ¿Cómo se incorpora a una página HTML?
 - ¿Cómo y cuando se ejecuta?
 - ¿Qué interacciones permite con el entorno?
 - ¿Puede interactuar con un servidor? ¿Cómo?

Javascript

- ¿Cómo se crea código javascript?
 - Definición de variables y objetos
 - Instrucciones y estructuras de control
 - Funciones

Javascript

- ¿Qué puedo hacer con código javascript?
 - Introducirlo en una página HTML, para:
 - Consultar los “objetos” HTML+CSS
 - Modificar los “objetos” HTML+CSS
 - Capturar acciones (eventos) sobre objetos HTML
 - Capturar acciones (eventos) sobre la página HTML
 - Interaccionar con el navegador que contiene la página HTML
 - Históricos
 - Barra de estados
 - Ventanas de mensajes
 - Pedir datos al usuario

Javascript

- ¿Cómo se incorpora a una página HTML?
 - Código insertado en la cabecera
 - Código insertado en el cuerpo
 - Especificación de evento en objeto HTML
 - Destino de un hiperenlace

Javascript

- ¿Cómo y cuando se ejecuta?
 - Código insertado en el cuerpo
 - En el momento en el que se procesa la sección HTML en la que se encuentra incluido.
 - Código insertado en la cabecera
 - Cuando se invoca la función desde el cuerpo HTML o como respuesta a un evento
 - Especificación de evento
 - Cuando sucede el evento
 - Destino de hiperenlace
 - Al seguir el hiperenlace

Javascript

- ¿Qué interacciones permite con el entorno?
 - Consulta + modificación de:
 - HTML
 - CSS
 - Consulta + modificación de:
 - Parámetros del navegador
 - Ventanas
 - Control de intercambio de información con un servidor
 - Envío de formularios
 - Gestión de micro-peticiones (AJAX)

Javascript

- ¿Puede interactuar con un servidor? ¿Cómo?
 - Sí
 - A través de AJAX
 - Con peculiaridades
 - La interacción la inicia siempre el cliente
 - Es interacción completa (no permite mantener estados, por ejemplo desde un servlet)
 - No tiene acceso a variables del servidor, sólo a objetos que puedan transmitirse encapsulados en HTTP

Motivación

- ¿Por qué HTML5?
- ¿Por qué CSS3?
- ¿Por qué JavaScript Frameworks?

Instalación del software

- Software necesario:

- Google Chrome

- <http://www.google.com/intl/es/chrome/>

- Conector Netbeans para Chrome

- <https://chrome.google.com/webstore/detail/netbeans-connector/hafdlehgocfcodbgjnpecfajgkeejnaa>

- Netbeans

- <http://netbeans.org>

- Java EE 7 JDK

- <http://www.oracle.com/technetwork/java/javaee/downloads/index.html>

- JQuery

- <http://jquery.com/>

Un ejemplo



EJEMPLO

Crear una simple aplicación web estática de ejemplo y usar la herramienta de desarrollo para modificar contenidos y visualización.

Valoración: 0% de la nota final

Fecha de entrega: sin entrega

Un ejemplo

■ Tareas

1. Abrir Netbeans
2. Crear un proyecto HTML5 con un ejemplo
3. Visualizar el proyecto con el navegador interno
4. Visualizar el proyecto con Chrome y la extensión para Netbeans
5. Evaluar la estructura HTML de un fichero
6. Evaluar la estructura de estilos de un fichero
7. Cambiar los modos de visualización según dispositivos
8. Usar el modo de inspección del navegador