



# Herramientas para trabajar con HTML5

# Índice



1   Herramientas para trabajar con HTML5	3
1.1   Creación de una página en HTML5 con Eclipse	4
1.2   Herramientas de edición: Eclipse	6
1.3   Introducción a HTML	8
1.4   Marcado HTML	9

# 1. Herramientas para trabajar con HTML5

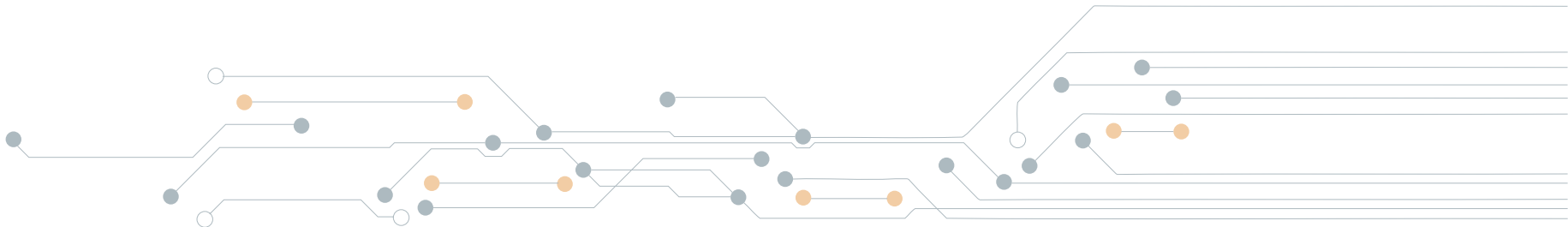
Para trabajar con HTML5 vamos a usar el entorno Eclipse. Se trata de un entorno de desarrollo multiplataforma y permite desarrollar en múltiples lenguajes tanto de programación como de marcado, uso de UML, etc.

El IDE Eclipse es de libre distribución y puede descargarse desde su web oficial ([www.eclipse.org](http://www.eclipse.org)), más concretamente la versión para desarrolladores J2EE. Esta versión contiene una serie de plugins preinstalados para desarrollo de aplicaciones web con Java. Aunque el lenguaje Java se encuentra fuera del ámbito de este curso, esta versión también contiene el editor para HTML5, que es la funcionalidad que vamos a usar.

En la web oficial de Eclipse, se define como "An IDE for everything and nothing in particular" (un IDE para todo y para nada en particular). Eclipse es, en el fondo, únicamente un almacén (workbench) sobre el que se pueden montar herramientas de desarrollo para cualquier lenguaje,

mediante la implementación de los plugins adecuados. La arquitectura de plugins de Eclipse permite, además de integrar diversos lenguajes sobre un mismo IDE, introducir otras aplicaciones accesorias que pueden resultar útiles durante el proceso de desarrollo como: herramientas UML, editores visuales de interfaces, ayuda en línea para librerías, etc.

El IDE Eclipse es, únicamente, una de las herramientas que se engloban bajo el denominado *Proyecto Eclipse*. El *Proyecto Eclipse* aúna tanto el desarrollo del IDE Eclipse como de algunos de los plugins más importantes (como el JDT, plugin para el lenguaje Java, o el CDT, plugin para el lenguaje C/C++).



Este proyecto también alcanza a las librerías que sirven como base para la construcción del IDE Eclipse (pero pueden ser utilizadas de forma completamente independiente), como por ejemplo, la librería de widgets SWT.

A partir de la versión Helios del IDE de Eclipse y superiores (con los plugins para desarrolladores de Java EE) proporciona soporte para el desarrollo de HTML5. Las versiones anteriores también tienen un editor de HTML, pero no admite HTML 5.

En concreto, permite:

- **Crear paginas HTML 5** con una nueva plantilla.
- Proporciona **autocompletado de código** para los elementos de HTML5.
- Ofrece un **editor de propiedades** para los atributos de HTML5.
- Ofrece un editor **WYSIWYG** para el desarrollo visual de páginas HTML5.

**What You See Is What You Get**  
(en español, "lo que ves es lo que obtienes")

## 1.1 | Creación de una página en HTML5 con Eclipse

Crear un documento HTML 5 es fácil. Simplemente hay que seguir estos pasos:

- Seleccione File → New → Other.
- Expandir Web y seleccione un archivo HTML. Haga clic en Next.
- Escriba un nombre de archivo, seleccione la carpeta principal y haga click en next.

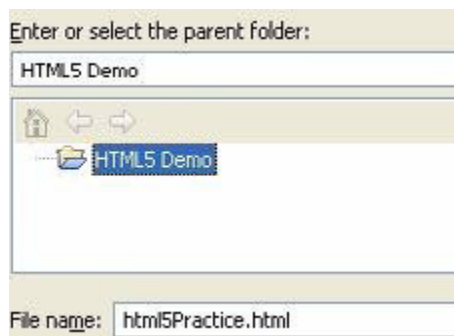


ILUSTRACIÓN: NUEVO DOCUMENTO HTML5 (1)

- Seleccione nuevo archivo HTML5 para la plantilla y hacer click en finish para terminar.

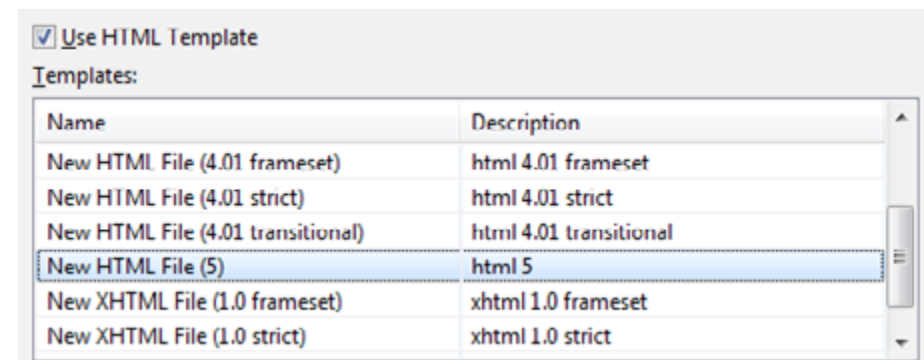


ILUSTRACIÓN: NUEVO DOCUMENTO HTML5 (2)

El archivo se abrirá con el editor de HTML, que proporciona resaltado de sintaxis (añade diferentes colores y tipos de letra en función del elemento que se muestra). Las etiquetas se muestran en verde, los atributos de color morado, los valores de los atributos de color azul, y los valores de los elementos en negro.

Hay que tener en cuenta que el documento contiene el elemento DOCTYPE nuevo que se utiliza. También contiene el elemento meta simplificado que se utiliza para especificar la codificación de caracteres UTF-8.



```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>

</body>
</html>
```

ILUSTRACIÓN: NUEVO DOCUMENTO HTML5 (3)



## 1.2 | Herramientas de edición: Eclipse

Vamos a ver algunas de las herramientas que el plugin nos proporciona:

- **Es el resultado de la sintaxis.** Las palabras reservadas y signos correspondientes al estándar aparecen resaltados en color. El resto del texto en negro.
- **Autocompletado de etiquetas.** Basta con pulsar ctrl + espacio y una lista de elementos HTML5 validos aparecerán listados. Si ya hemos escrito parte de la etiqueta, nos lo autocompletara si sólo hay una etiqueta que empieza por el texto escrito. Si hay más de una, aparecerá un listado únicamente con las coincidencias. Sólo nos mostrarán las etiquetas permitidas en el lugar en el que estamos editando.

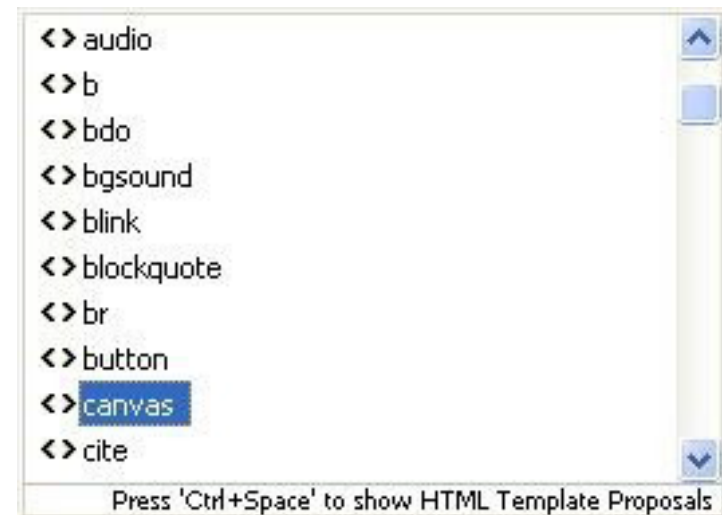


ILUSTRACIÓN: AUTOCOMPLETADO DE ETIQUETAS

- **Autocompletado de atributos.** Funciona exactamente igual que con las etiquetas, solo que debemos estar editando dentro de la declaración de una etiqueta.



ILUSTRACIÓN: AUTOCOMPLETADO DE ATRIBUTOS



## 1.3 | Introducción a HTML

HTML (HyperText Markup Language). Su nombre se traduce literalmente como “Lenguaje de marcado de hipertexto”.

Se encuentra enmarcado en los llamados lenguajes de marcado. Los lenguajes de marcado son los utilizados para generar documentos en los que, junto con el texto que constituye el contenido, se incluyen etiquetas o marcas que contienen información adicional sobre la estructura del contenido.

En el caso de HTML, se trata concretamente de un lenguaje de marcado de presentación, en el que las etiquetas que acompañan al texto indican al navegador información de cómo se encuentra organizado.

Es un estándar desarrollado y regulado por la W3C, que es un organismo dedicado a la estandarización de las tecnologías utilizadas en la Web.

Una de las características principales del lenguaje es que las páginas generadas con HTML son de hipertexto. Esto quiere decir que una página no contiene un texto aislado, sino que están o pueden estar enlazadas a otras páginas y ser referenciadas desde otras páginas mediante enlaces o links. Esto nos permite navegar por ellas.

Es independiente de la plataforma ya que HTML solo genera documentos de texto y admite cualquier codificación de caracteres.

El proceso de alterar la información de una página es relativamente sencillo y ágil ya que es un lenguaje de fácil comprensión.

Basa su contenido en referenciación. Si en una página HTML, queremos introducir algún elemento como una imagen, script, video, u otra página HTML (por ejemplo), no se incluye su contenido directamente en ella si no que se referencia (se indica donde está el contenido que queremos incluir). El navegador es el encargado de interpretar las referencias y cargarlas componiendo la visualización final.

Al ser un estándar, uno de sus objetivos es la compatibilidad hacia atrás (o retro-compatibilidad). Un navegador que es capaz de interpretar una versión de HTML, debe ser capaz de interpretar páginas escritas en cualquiera de sus versiones anteriores.



## 1.4 | Marcado HTML

El marcado HTML se realiza mediante etiquetas de apertura y cierre. Todo lo que se encuentra dentro de una etiqueta de apertura y cierre se encuentra afectado por ella, incluyendo texto plano y cualquier otra etiqueta que pueda contener.

Una etiqueta de apertura se define con un nombre de etiqueta encerrado entre los corchetes angulares < y > (signo "menor que" y signo "mayor que")

*Ejemplo: <body>*

Entre la etiqueta de apertura y la de cierre, se encuentra el contenido. El contenido puede ser texto u otras etiquetas. Una etiqueta con contenido tiene el siguiente formato:

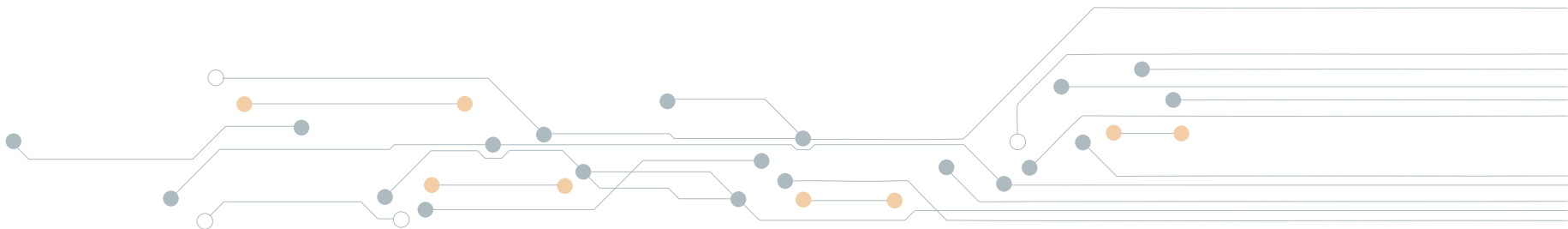
*Ejemplo: <etiqueta>contenido</etiqueta>*

Una etiqueta de cierre se define con el corchete angular de apertura "<" seguido del carácter barra /, el nombre de etiqueta y finalmente el corchete angular de cierre.

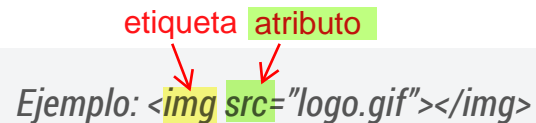
*Ejemplo: </body>*

Si entre la etiqueta de apertura y la de cierre no se añade ningún contenido, se puede declarar con un corchete angular de apertura "<" seguido del nombre de la etiqueta, una barra / y finalmente el corchete angular de cierre.

*Ejemplo: <br />*



Los **atributos** añaden **información adicional** a una **etiqueta**. Cada **etiqueta** admite una **serie** de **atributos**. Se **añaden** únicamente a las **etiquetas** de **apertura**, justo **después** del **nombre**. Se **admite** más de un **atributo** por **etiqueta**. Siguen el **esquema** **clave / valor** de la siguiente forma: clave: "valor".

*Ejemplo:* **etiqueta** **atributo**  

 Ejemplo: `</img>`

Por ejemplo, supongamos que tenemos el siguiente código HTML:

```
<a href="http://httpd.apache.org">enlace a
la página de apache</a>
```

El texto que se encuentra encerrado entre la etiqueta de apertura `<a>` y la etiqueta de cierre `</a>`, será interpretado por el navegador como un enlace. El **atributo href** indica la página a la que **referencia** el **enlace**.

En un fichero que va a contener un documento HTML, debemos definir en la primera línea cual es la versión de HTML que vamos a utilizar. De esta forma, el navegador podrá interpretarlo correctamente. También es útil añadir el tipo de documento a la hora de desarrollar puesto que el editor de HTML que usemos, podrá validar si lo que escribimos es correcto.

A continuación vemos dos ejemplos de página que van a versiones diferentes de HTML:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN" "http://www.w3.org/TR/html4/loose.
dtd">
<html>
<head>
    <title>login</title>
    <meta http-equiv="Content-Type" content="text/
html; charset=ISO-8859-1">
</head>
```

ILUSTRACIÓN: DEFINICIÓN DE DOCUMENTO EN HTML 4.01

```
<!DOCTYPE html>
<html>
<head>
    <title>login</title>
    <meta charset="UTF-8">
</head>
```

ILUSTRACIÓN: DEFINICIÓN DE DOCUMENTO HTML5

*Telefonica*

---

EDUCACIÓN DIGITAL