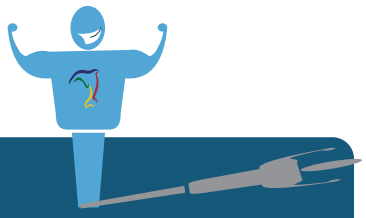
autentia

Avenida de Castilla,1 - Edificio Best Point - Oficina 21B
28830 San Fernando de Henares (Madrid)
tel./fax: +34 91 675 33 06
info@autentia.com - **www.autentia.com**

¿**Qué ofrece** Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
Ese apoyo que siempre quiso tener...

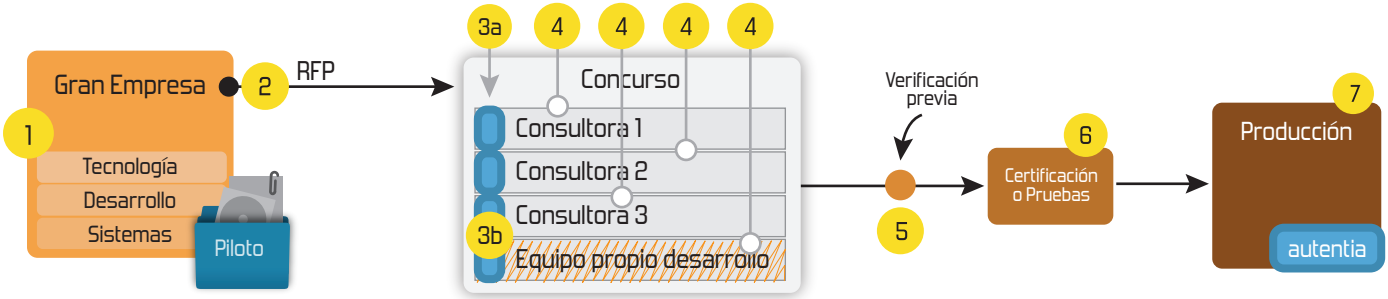
1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces, HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
Gestor de contenidos (Alfresco)
Aplicaciones híbridas

Control de autenticación y acceso (Spring Security)
UDDI
Web Services
Rest Services
Social SSO
SSO (Cas)

JPA-Hibernate, MyBatis
Motor de búsqueda empresarial (Solr)
ETL (Talend)

Dirección de Proyectos Informáticos.
Metodologías ágiles
Patrones de diseño
TDD

Tareas programadas (Quartz)
Gestor documental (Alfresco)
Inversión de control (Spring)

BPM (jBPM o Bonita)
Generación de informes (JasperReport)
ESB (Open ESB)

Compartimos nuestro conocimiento en:
www.adictosaltrabajo.com

Para más información visítenos en:
www.autentia.com

AdictosAlTrabajo
Terrakas 1x03 ¡¡Ya está en la web!! terrakas.com
autentia Soporte a desarrollo informático
Hosting patrocinado por enredados
Entra en Adictos a través de
E-mail
Contraseña
Entrar
Deseo registrarme
Olvidé mi contraseña
Inicio Quiénes somos Formación Comparador de salarios Nuestro libro Más
» Estás en: Inicio » Tutoriales » Posicionamiento de componentes en HTML con el soporte de CSS.



Jose Manuel Sánchez Suárez

Consultor tecnológico de desarrollo de proyectos informáticos.

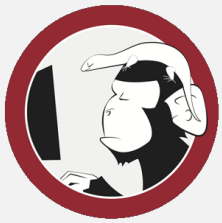
Puedes encontrarme en Autentia: Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en Java/J2EE

Ver todos los tutoriales del autor

Catálogo de servicios Autentia





Síguenos a través de:

Últimas Noticias

» La regla del Boy Scout y la Oxidación del Software
» Autentia conquista los Alpes
» Orientación a objetos y la importancia del "Tell, Don't Ask"
» Autentia patrocina al Club KiteSurf Centro
» Autentia patrocina el I Torneo Voley Playa Terrakas
Histórico de noticias

Fecha de publicación del tutorial: 2012-08-15 Tutorial visitado 0 veces Descargar en PDF

Posicionamiento de componentes en HTML con el soporte de CSS.

0. Índice de contenidos.

- 1. Introducción.
- 2. Entorno.
- 3. Posicionamiento por defecto.
- 4. style="display".
- 4. style="float".
- 5. style="position".
- 6. Referencias.
- 7. Conclusiones.

1. Introducción

En este tutorial vamos a explicar, de una manera más o menos gráfica, cómo se disponen los componentes visuales que pueden ir dentro del código HTML de una página web, con el soporte de hojas de estilo CSS.

El nivel de complejidad del tutorial es básico, sobre todo si los comparamos con otros publicados recientemente sobre Responsive CSS, pero hay que empezar por abajo, es capital tener las siguientes nociones claras para evitar problemas de renderización de componentes entre distintos navegadores y versiones de los mismos.

2. Entorno.

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil MacBook Pro 15' (2.4 GHz Intel Core i7, 8GB DDR3 SDRAM).
- Sistema Operativo: Mac OS X Lion 10.7.2
- Mozilla Firefox 14.0.1

3. Posicionamiento por defecto.

Un navegador web o browser, a través de su agente de usuario, lleva a cabo la lectura del código HTML de una dirección web o url solicitada; la lectura de dicha página es secuencial (de arriba a abajo) y la disposición de los componentes en pantalla dependerá del tipo de elemento que se trate:

- elementos en línea (span, img, a y otros): ocupan solo el espacio necesario para mostrar el contenido interno de los mismos,
- elementos de bloque (p, div, h1, h2, fieldset y otros): siempre empiezan en una nueva línea y ocupan todo el espacio disponible en pantalla, o dentro de su contenedor, hasta el final de la línea, aunque sus contenidos no tengan tanto tamaño.

Todos se disponen de izquierda a derecha en pantalla, ocupan un espacio rectangular y, por defecto, en función del contenido de los mismos y solo se dispondrá un elemento en una nueva línea cuando sea un elemento de bloque.

Tan importante es conocer esto como saber que los estilos y, como consecuencia, las posiciones de los componentes en pantalla se pueden modificar aplicando hojas de estilos CSS.

Tanto HTML como CSS tienen una serie de versiones, la de CSS depende de la versión de HTML con la que estemos trabajando y depende del navegador soportará hasta una versión de HTML y CSS concretas.

Además, los navegadores aplican una versión de HTML por defecto cuando, dentro del contenido del documento, no se hace una mención explícita a la misma. Por eso, es de suma importancia que especifiquemos cuál es la versión de HTML y, como consecuencia de CSS, con la que estamos trabajando. Para ello en la cabecera de todos nuestros documentos HTML debemos incorporar la directiva <!DOCTYPE, por ejemplo:

Para XHTML 1.0 transsitional:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xh
```

Para HTML 5:

```
<!DOCTYPE html>
```

La ausencia de DOCTYPE es la principal raíz de los problemas de disparidad de visualización entre navegadores.

A continuación un ejemplo de mezcla de elementos de línea y de bloque:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/x
<html>
<head>
  <title>Elementos en línea y elementos de bloque</title>
</head>
<body>
<h3>Adictos al Trabajo.</h3>
<p>Tutoriales sobre JEE en <a href="http://www.adictosaltrabajo.com">adictos al trabajo<
<span>30/07/2012.</span> - <span id="created">30/07/2012.</span>
</body>
</html>
```

Se visualizará como sigue:

Adictos al Trabajo.

Tutoriales sobre JEE en adictos al trabajo.

jmsanchez - 30/07/2012.

4. style="display".

El atributo más básico a nivel de estilos es display y en función del tipo de elemento tiene un valor por defecto:

- inline: en línea, por defecto para los elementos en línea,
- block: de bloque, por defecto para los elementos de bloque,
- none: para ocultar la visualización de un componente sin ocupar espacio en pantalla.

Hay más posibles valores, podéis consultar una referencia rápida aquí.

Si, por ejemplo, en el html anterior incluimos un selector css del siguiente modo:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/x
<html>
<head>
  <title>Elementos en línea y elementos de bloque</title>
  <style type="text/css">
    p{
        display:inline;
    }
  </style>
</head>
<body>
<h3>Adictos al Trabajo.</h3>
<p>Tutoriales sobre JEE en <a href="http://www.adictosaltrabajo.com">adictos al trabajo<
<span>30/07/2012.</span> - <span id="created">30/07/2012.</span>
</body>
</html>
```

Modificaremos la disposición en pantalla del párrafo y, con ello, de todos los componentes posteriores a él.

Adictos al Trabajo.

Tutoriales sobre JEE en adictos al trabajo. jmsanchez - 30/07/2012.

5. style="float".

El segundo modificador del que vamos a hablar es float que permite flotar un componente a derecha o izquierda, rompiendo con esa disposición secuencial de los elementos dentro del HTML. Es importante conocer que, si flotamos un elemento, los posteriores se verán afectados puesto que se comportará como un elemento en línea:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/x
<html>
<head>
  <title>Elementos en línea y elementos de bloque</title>
  <style type="text/css">
    div#links{
        float:left;
    }
  </style>
</head>
<body>
<div id="header">
```


Últimos Tutoriales

» Integrando Xcode4 con GitHub

» WebSockets con Java y Tomcat 7

» Test de integración con Solr y el soporte de EmbeddedSolrServer.

» Primeros pasos en Android.

» Arrancar Solr desde un proyecto Maven con el soporte de Jetty.

Últimos Tutoriales del Autor

» Test de integración con Solr y el soporte de EmbeddedSolrServer.

» Arrancar Solr desde un proyecto Maven con el soporte de Jetty.

» Lectura y tratamiento de ficheros Excel con Talend (II): filtros y splits

» Invocar a un servicio REST securizado, con el soporte de plantillas Spring.

» Indexación de documentos en Solr con el soporte de Talend.

Categorías del Tutorial

Otras Técnicas

Navegadores

HTML5

Últimas ofertas de empleo

2011-09-08
Comercial - Ventas - MADRID.

2011-09-03
Comercial - Ventas - VALENCIA.

2011-08-19
Comercial - Compras - ALICANTE.

2011-07-12
Otras Sin catalogar - MADRID.

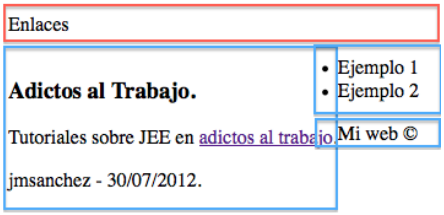
2011-07-06
Otras Sin catalogar - LUGO.


Jose Manuel Sánchez
sanchezsuarezj

sanchezsuarezj @ruedalenticular invitado quedas!!! Eso si, no te bajes en bici que se te va a hacer

```
    <p>Enlaces</p>
</div>
<div id="links">
<h3>Adictos al Trabajo.</h3>
<p>Tutoriales sobre JEE en <a href="http://www.adictosaltrabajo.com">adictos al trabajo<
<span id="author">jmsanchez</span> - <span id="created">30/07/2012.</span>
</div>
<div id="menu">
    <ul>
        <li>Ejemplo 1</li>
        <li>Ejemplo 2</li>
    </ul>
</div>
<div id="credits">
    <span>Mi web ©</span>
</div>
</body>
</html>
```

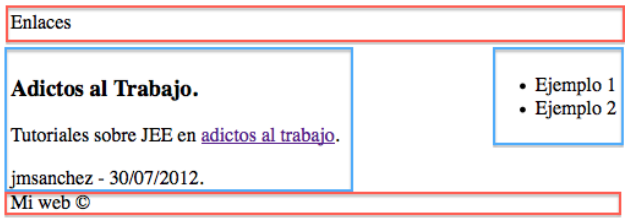
Tendremos la siguiente disposición en pantalla:



Para romper la aplicación del modificador float a los siguientes elementos podemos usar el modificador clear:both. En el siguiente ejemplo vemos como flotar a izquierda y derecha y romper el float en la siguiente capa.

```
<style type="text/css">
  div#links{
      float:left;
  }
  div#menu{
      float:right;
  }
  div#credits{
      clear:both;
  }
</style>
```

Con ello la disposición en pantalla cambia como sigue:

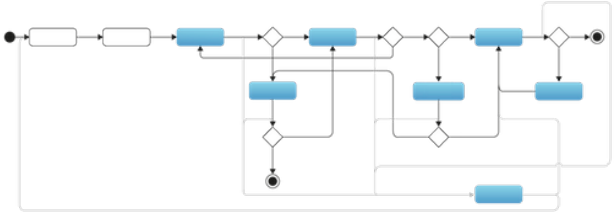


6. style="position".

El último modificador es el de la posición que puede adoptar los siguiente valores:

- static: la de por defecto para todos,
- absolute: se posiciona respecto al último elemento en la jerarquía no marcado como estático; si no existe en la jerarquía de padres o contenedores un elemento marcado como no estático se posiciona como fixed, esto es, respecto a la ventana,
- fixed: su posición será relativa respecto a la posición de la ventana,
- relative: su posición será relativa respecto a su posición normal.

Imaginemos que tenemos un gráfico que describe el flujo de trabajo en fases dentro de una cadena de negocio, un workflow y queremos visualizar, de forma dinámica, el número de actividades abiertas dentro de cada fase en una página web, con HTML. Empezaremos por tener, por ejemplo, una imagen con el gráfico sin descripciones.



La descripción de cada puesto de trabajo la queremos internacionalizada y mostrará el número de actividades abiertas en el mismo, a modo de cuadro de mando.

Lo primero es crearnos un elemento complejo para cada puesto:

```
<div id="diagnosis">
  <span>Diagnóstico
  <span class="activities">15</span>
```

un poco largo ;)
4 days ago · reply · retweet · favorite

sanchezsuarezj arrancó la feria de #Malaga y para abrir boca que mejor que un #espeto de sardinas ;)
pic.twitter.com/meoqXfu8
4 days ago · reply · retweet · favorite

sanchezsuarezj recibida la transferencia de 75€ de google adsense por la publicidad en los tutoriales de @adictosaltrabaj. eco @autentia ;)
6 days ago · reply · retweet · favorite

Join the conversation

```
</div>

<div id="repair">
  <span>Reparación
  <span class="activities">5</span>
</div>
```

El objetivo es, teniendo la imagen de fondo del flujo, posicionar las capas sobre su correpondiente tarea.

Si usamos posiciones absolutas el problema será que, en función del contenido y tamaño de los componentes que posicionemos encima del gráfico, pueden variar la suya puesto que se posicionan respecto a la ventana.

La solución es establecer la imagen como fondo de una capa dispuesta de forma relativa en la página (rompiendo el position=static) y, dentro de la misma, posicionar todas las capas correspondientes a las tareas de forma absoluta. Las posiciones de las capas serán absolutas a la capa de la imagen, con lo que no cambiará su posición debido a redimensionamientos.

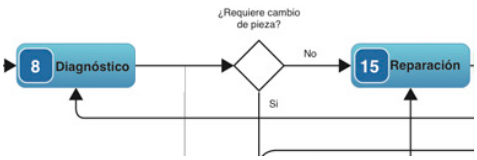
```
<div id="workflow">
  <div id="diagnosis">
    <span>Diagnóstico</span> <!-- El texto se generaría internacionalizado -->
    <span class="activities">8</span> <!-- El número de actividades se asignaría de form
  </div>

  <div id="repair">
    <span>Reparación</span>
    <span class="activities">15</span>
  </div>
</div>
```

Los estilos podrían ser los siguientes:

```
<style type="text/css">
  div#workflow{
      background: url(./images/workshop.png) no-repeat;
      position:relative;
  }
  div#diagnosis{
      position:absolute;
      left:100px;
      top:250px;
  }
  div#repair{
      position:absolute;
      left:340px;
      top:250px;
  }
  .activities{
      background: url(./images/activity.png) no-repeat;
      float:left;
      margin-right:1em;
  }
</style>
```

Y la información de las distintas fases quedarían cuadradas dentro de cada capa como se muestra a continuación.



Luego está la tarea de internacionalizar el texto de los nodos de decisión.

6. Referencias.

- http://www.barelyfitz.com/screencast/html-training/css/positioning/

7. Conclusiones.

Plasmando conocimientos y un caso de uso real ;)

Un saludo.

Jose

jmsanchez@autentia.com

A continuación puedes evaluarlo:

Regístrate para evaluarlo

Por favor, vota +1 o compártelo si te pareció interesante

Share | 0

Anímate y coméntanos lo que pienses sobre este **TUTORIAL:**

» Registrate y accede a esta y otras ventajas **«**

SOME RIGHTS RESERVED Esta obra está licenciada bajo licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5

Copyright 2003-2012 © All Rights Reserved | Texto legal y condiciones de uso | Banners | Powered by Autentia | Contacto

XHTML 1.0 CSS XML RSS XML ATOM