

Diseño Responsive Con Cute Grids

Estamos en una especie de época dorada para los [FrameWorks](#), como sabemos, hay gran variedad de estos y para todo tipo de desarrollo (programación, diseño, etc). Tenemos grandes opciones como [Bootstrap](#) y [Foundation](#), pero a veces estos [Frameworks](#) nos brindan más de lo que necesitamos. A veces, todo lo que necesitamos es mostrar la información en una malla (Grid) y con diseño responsive, la respuesta para esto es: [Cute Grids](#).

¿Por Qué Utilizar Cute Grids?

La idea detrás de **Cute Grids** es conseguir la funcionalidad que ofrecen los **Frameworks**, pero además de esto centrarse sólo en la disposición de la información. Con **Cute Grids** nos aseguramos de que todos nuestros sitios web funcionen en el móvil, así como en el navegador web de nuestro PC de escritorio. El **Framework** es completamente adaptable e incluye cinco [media query](#), con anchos **em** especificados para cinco tamaños de dispositivos móviles diferentes.

Lo Básico

Con el fin de usar **Cute Grids**, sólo debes vincular el archivo CSS y ya está. No hay archivos de JavaScript, no hay complicadas inclusiones CSS y ninguna combinación, sólo un único archivo:

```
[html]<link                href="css/cutegrids.min.css"
rel="stylesheet">[/html]
```

(También puedes incluir **nomalize.css** para asegurarse de tener un restablecimiento de su página web, si se quiere, pero lo único que necesitas es el archivo de CSS). Después de esto,

estaras listo para usar Cute Grids, empezaremos con un ejemplo de una fila:

```
[html]<div class="row"> </ div>[/html]
```

Una vez que haya definido la fila, es necesario dividirla en columnas. Lo hace mediante la adición de un nuevo div con las palabras clave de la clase «cute», seguido por el ancho de la columna, seguido por el nombre del dispositivo, separados por guiones. Cada fila debe sumar un total de 12 columnas, así:

```
[html]
<div class="row">
<div class="cute-3-laptop">
content here...
</div>
<div class="cute-3-laptop">
content here...
</div>
<div class="cute-6-laptop">
content here...
</div>
</div>

[/html]
```

En este caso se puede ver que nos estamos enfocando las computadoras portátiles, lo que significa que este diseño tendrá 3 columnas excepto cuando la anchura del dispositivo es menor que la anchura establecida por la clase, en cuyo caso el ancho de la columna será del 100%.



Lo mejor de **Cute Grids**, es que mientras es sencilla, puede combinar dos tipos de clases dependiendo del dispositivo a usar, por ejemplo, uno para el portátil y otra para la tableta , justo de la siguiente manera:

```
[html]

<div class="row">
<div class="cute-3-laptop  cute-6-tablet">  content  here...
</div>
<div class="cute-3-laptop  cute-6-tablet">  content  here...
</div>
<div class="cute-6-laptop  cute-12-tablet">  content  here...
</div>
</div>

[/html]
```

Como puede ver , si se establece una segunda clase puede definir diferentes anchos para distintos dispositivos. En este caso, usted tendrá 3 columnas en los ordenadores portátiles y de escritorio, pero en la tableta tendrás 2 columnas en la primera fila y 1 que abarca toda la fila de abajo. Esto es debido a que el total debe sumar 12. Si lo deseas, también puede anidar columnas.

Configurando Columnas

Usted puede agregar las compensaciones a sus columnas mediante la adición de «offset» para el nombre de clase después del dispositivo, así:

```
[html]

<div class="cute-6-tablet  cute-2-tablet-offset">
Content Here
</div>

[/html]
```

Esto compensará la cuadrícula de dos columnas.



Cambiando El Orden De Las Columnas

En la construcción de sitios con diseño responsive, a veces puede que tenga que cambiar el orden de las columnas, sobre todo cuando se pasa de escritorio a teléfono. Para ello, se utiliza las palabras clave «push» y «pull» en sus clases:

```
[html]

<div class="row">
<div class="cute-4-desktop cute-8-tablet-push">
Content Here
</div>
<div class="cute-8-desktop cute-4-tablet-pull">
Content Here
</div>
</div>

[/html]
```

Esto cambiará el orden de las columnas en su diseño cuando se vea en una tableta.

Centrado De Columnas

También es muy fácil centrar una columna utilizando sólo las clases. En consonancia con el resto de este Framework, sólo tiene que añadir la palabra clave «center» en el nombre de clase:

```
[html]

<div class="row">
<div class="cute-12-tablet center-tablet">
Content Here
</div>
</div>

[/html]
```

Así como podemos centrar las columnas, también podemos descentrarlas, con una sintaxis similar:

```
[html]

<div class="row">
<div class="cute-12-desktop center-desktop uncenter-tablet">
Content Here
</div>
</div>

[/html]
```

Mostrar U Ocultar Columnas

De vez en cuando, usted querrá ocultar una columna por completo en un dispositivo en particular. Cute Grids le permite hacerlo, con las palabras clave «hide» y «show»:

```
[html]

<div class="row">
<div class="cute-12-desktop .hide-tablet .show-phone">
Content Here
</div>
</div>

[/html]
```

En este caso, la columna se oculta en los tamaños de las tabletas y se muestra en los tamaños de teléfono.



Imágenes Con Diseño Responsive

Usted puede incluso hacer que sus imágenes sean responsive utilizando **Cute Grids**, para ello sólo tiene que agregar la clase de «**responsive-img**» para su imagen.

```
[html][/html]
```

Finalmente

Como puedes ver, **Cute Grids** es una herramienta fácil de configurar y de fácil uso, pero al mismo tiempo es muy potente y permite un gran control. Usa este **Frameworks** en los cuales quieres impresionar al momento de mostrar la información ;)