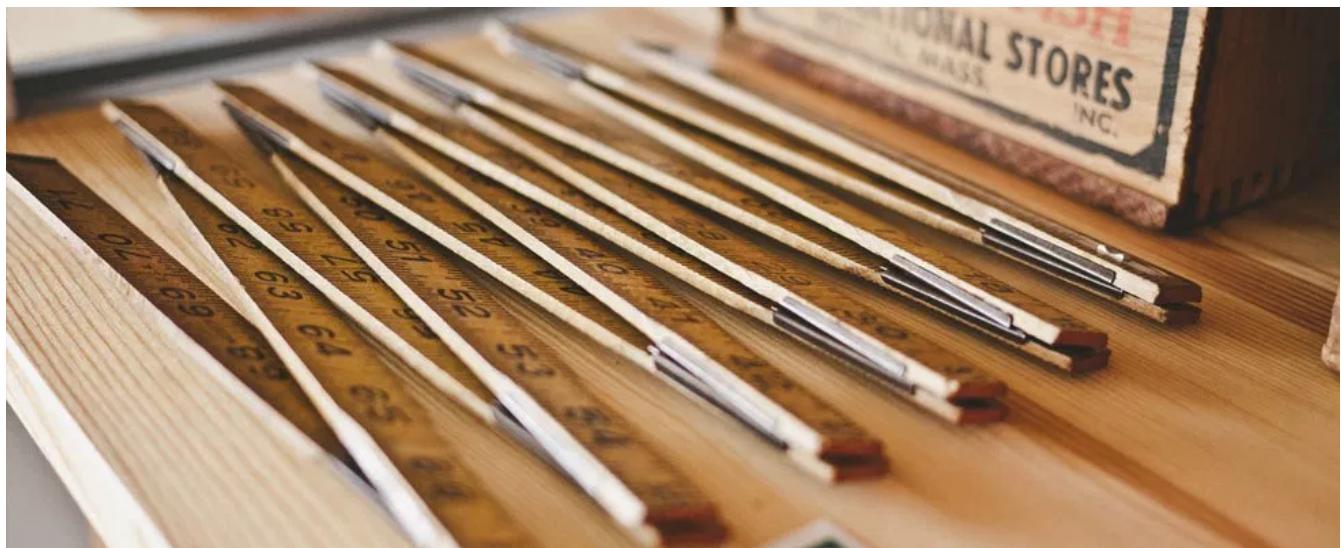


[Acerca ▾](#)[CURSOS DE DISEÑO](#)[Contenido ▾](#)[Recursos ▾](#)[Contacto](#)[Ingresar a AMK](#)[Pro](#)

DISEÑO, TUTORIALES | 29 DE ABRIL DE 2015

Unidades de medida en CSS (Px, Porcentajes, Em y Rem)

por Francisco Aguilera G.

Hace un tiempo vi [una charla del ilustrador Puño](#), en donde hacía una reflexión muy importante: cada vez que alguien le pregunta cuándo comenzó a dibujar, él simplemente responde con una contra-pregunta ***¿y cuándo dejaste tú de dibujar?***

Esto hace completo sentido porque cuando niños todos dibujamos. El dibujo es el primer método a través del que podemos plasmar nuestros pensamientos antes de aprender a escribir, pero por algún motivo a medida que crecemos la mayoría de nosotros vamos olvidando este modo de expresión.

Y lo mismo ocurre con el diseño web responsive.

Toda página web es responsive de manera natural, somos nosotros quienes al trabajar los diseños y los estilos CSS vamos tapando esta habilidad nativa con tamaños, medidas y posiciones rígidos que hacen que las páginas web dejen de ser adaptables.

¿No me crees? Crea un archivo HTML simple, agrégale un montón de párrafos, algunos encabezados, ábrelo en tu navegador y comienza a reducir el tamaño de la ventana, verás que la página web se adapta sin haber escrito una sola línea de CSS o un *Media Query*.

Esta rigidez al momento de crear los estilos CSS para nuestras páginas proviene de la aplicación de tamaños fijos a los elementos y también en parte del uso de una unidad de medida rígida: **los pixeles**.

Al usar pixeles para definir las dimensiones de los objetos y los estilos de fuente no solo estamos definiendo tamaños rígidos, sino que también estamos ignorando las configuraciones que cada usuario pueda tener en su navegador.

Mira este ejemplo, es la página de inicio de santander.cl, en donde se utiliza este tipo de CSS. Así se ve cuando la configuración del tamaño de fuente está normal:

The screenshot shows the Santander Chile website with a standard browser configuration. The main headline 'En mayo usa tu Tarjeta todos los días' is rendered in a medium-sized font. A red box highlights this text. To the right, a configuration window for Google Chrome is open, showing various settings like font size and zoom level. The configuration window also has a red border.

Tamaño normal

Pero si actualizamos la configuración del navegador y hacemos que la letra sea más grande para que, por ejemplo, alguien como mi Papá que tiene dificultades para ver bien, pueda leer el texto, verás que no ocurre nada, porque las medidas de los objetos y de las fuentes están aplicadas en pixeles:

The screenshot shows the same website after increasing the browser's font size. The main headline 'En mayo usa tu Tarjeta todos los días' is now significantly larger and bolder. The red configuration window from Google Chrome is still present on the right, showing the updated font size setting.

Tamaño grande

Observa esta otra página desarrollada con tamaños y unidades de medida adaptables y verás que sí respeta las configuraciones del usuario:



ARTICLES VIDEOS WORKBOOK

CONFERENCE | AWARDS | MAGAZINE | BOOKS

Office Dynamics

The Best Teams Act Like Musicians

by Allison Eck



Team members don't function autonomously — they improvise, listen to one another, and thrive on a network of dependencies.

2387

— WORKING DAILY insights on creativity, productivity, and personal growth

Avoid Regret with a Weekly Work Checklist

"Don't put off until tomorrow what you can do today." - Benjamin Franklin

15 hours ago

IKEA: Inspiration from the World's Largest Furniture Retailer

Finding inspiration in travel

1 day ago

Find Inspiration in Travel: Both in Your Own City & Abroad

Mixing travel and passion to deepen

THE CREATIVES GUIDE TO BUILDING A BUSINESS WITH IMPACT

Tamaño normal

chrome://settings

Configuración

Historial

Extensiones

Configuración

Información

Mostrar notificaciones cuando se detecten nuevas impresiones

Acessibilidad

Adadir funciones de accesibilidad adicionales

Sistema

Utilizar aceleración por hardware cuando esté disponible

Restablecer configuración

Restablecer los ajustes a la configuración predeterminada original

Ocultar configuración avanzada...

ARTICLES VIDEOS WORKBOOK

CONFERENCE | AWARDS | MAGAZINE | BOOKS

Office Dynamics

The Best Teams Act Like Musicians

by Allison Eck



— WORKING DAILY insights on creativity, productivity, and personal growth

Avoid Regret with a Weekly Work Checklist

"Don't put off until tomorrow what you can do today." - Benjamin Franklin

15 hours ago

Red

Google Chrome está utilizando la configuración

MAKES YOUR MARK:

Tamaño grande

Configuración

Historial

Extensiones

Configuración

Información

Contenido web

Tamaño de fuente: Muy grande

Zoom de página: 100%

El tabulador permite destacar los enlaces y los campos de formulario.

Unidades de medida recomendadas

Ok, ya ha quedado claro el por qué no es una buena idea usar los pixeles como unidad de medida para el diseño web pero entonces ¿Cuál es la alternativa?

Me gustaría recomendarte 3 unidades de medida en CSS que son las que creo se deberían usar en el diseño web actual y explicarte (en términos generales) en qué circunstancias deberíamos usar cada una de ellas:

Porcentajes

La unidad de medida porcentual es la que se usa por defecto en los elementos HTML en donde de manera predeterminada cada elemento de bloque usa un ancho del 100%, es por eso que cuando achicamos la ventana del navegador con una página que no tenga estilos, la página se adapta, ya que siempre usará el ancho total visible.

Pero nosotros podemos utilizar los porcentajes de una manera más avanzada tratando de generar layouts mas complejos.

Supongamos, por ejemplo, que tienes un div que contiene todos los elementos de la página y, según el diseño, este elemento debiera medir 1200 pixeles. En lugar de caer en la tentación de simplemente usar esa medida en pixeles, te recomendaría usar una medida en porcentajes, en donde el máximo ancho del elemento sean esos 1200px:

```
1 .container {  
2     margin: 0 auto;  
3     width: 90%;  
4     max-width: 1200px;  
5 }
```

CSS

Con estas 3 propiedades de CSS conseguimos que **a)** El elemento se centre en la página, **b)** tenga un ancho del 90% de la ventana y **c)** su ancho nunca sea superior a 1200 pixeles.

Es decir, hemos conseguido que el elemento con la clase *container* sea *responsive* sin la necesidad siquiera de escribir un *media query*.

El uso de los porcentajes también lo podemos llevar a elementos interiores del layout, en donde, por ejemplo, podemos asignar a la columna principal de contenido y a la barra lateral unas medidas de ancho del 70% y el 30% respectivamente, haciendo que sean completamente adaptables al tamaño de su elemento contenedor.

Em

Un *em*, es básicamente el tamaño de una letra “M” (bien podría ser cualquier otra letra) del elemento al cual se esté aplicando esta medida. Es decir, si un elemento tiene aplicado un tamaño de fuente de 16 pixeles, entonces 1 em será igual a 16px (los navegadores de manera predeterminada definen un *font-size* de 16px al elemento HTML, por lo tanto, por defecto 1em es igual a 16px).

La unidad *em* es escalable y siempre depende de su elemento padre. Por ejemplo, si el elemento *body* tiene un tamaño de fuente de 16px y un elemento hijo tiene una fuente con tamaño 1.3em, este texto se mostrará de un tamaño un 30% más grande que el del *body* (20.8px), mientras que si dentro de ese elemento tenemos otro hijo con un *font-*

size de 1.3 em, el tamaño de fuente de este objeto sería un 30% más grande que el tamaño de su padre (27.04px).

Body = 1em (16px)

Hijo = 1.3em (16px × 1.3 = 20.8px)

Nieto = 1.3em (20.8px × 1.3 = 27.04px)

Es recomendable usar la unidad de medida *em* para definir los tamaños de fuente, los altos de línea y también para elementos de diseño que no requieran ser muy exactos o que requieran una medida que tenga relación con el tamaño del texto, como por ejemplo el margen entre párrafos, el relleno interior de los *blockquotes*, etc.

También se puede aplicar a elementos generales del layout aunque no es muy recomendable, ya que si eventualmente se cambia el tamaño de fuente de uno de ellos, se podría estropear el diseño.

Rem

La unidad de medida *rem* es muy similar a *em*, con la única diferencia de que no es escalable, esto quiere decir que no depende del elemento padre, sino del elemento raíz del documento, el elemento HTML. *Rem* significa “*Root Em*”, o sea, es un *em* basado en la raíz.

Esto significa que si el elemento HTML tiene un tamaño de fuente de 16px (como es por defecto), entonces 1rem, sería igual a 16px, y si queremos aplicar un tamaño basado en *rem* a cualquier elemento de la página, no importará cual sea el tamaño de fuente que tenga asociado ese elemento, ya que 1 rem siempre será igual a 16 pixeles a no ser que se modifique el elemento raíz.

Usar *rem* nos permite cierta estructura para poder definir ciertas partes del layout, pero al mismo tiempo nos entrega cierta escalabilidad para respetar las configuraciones de cada usuario.

Esta unidad de medida es recomendable para aplicar a elementos del layout que requieran medidas fijas y eventualmente también para textos que deseemos que tengan un tamaño de fuente que no dependa de su elemento padre.

Los rem, son una unidad muy interesante también para definir los *media queries* de CSS. Dale un vistazo al código fuente de esta misma página y verás que los *rem*, son la unidad más utilizada en diferentes instancias.

Si quisiéramos refinar el ejemplo anterior para no usar pixeles deberíamos usar algo como lo siguiente:

```

1 .container {
2   margin: 0 auto;
3   width: 90%;
4   max-width: 75rem;
5 }
```

Para poder convertir una medida de pixeles a *rem* solo tienes que multiplicar el tamaño que quieras obtener por el número 0.0625, eso te dará el tamaño que debes usar en *rem*. Así es como he llegado a definir que 75rem es igual a 1200px:

$$75\text{rem} = 1200\text{px} \times 0.0625$$

Si te asusta la matemática, no te preocupes, he creado [una pequeña herramienta](#) para ayudarte a calcular las medidas en *rem* sin tener que sacar tu calculadora ni tener que hacer hacks innecesarios en tus CSS.

En síntesis

Si buscas en internet sobre unidades de medida en CSS verás que hay personas que recomiendan usar únicamente *rem*, hay otros que recomiendan solo usar *em* y creo que debe haber aquellos que también proponen usar únicamente medidas porcentuales.

Yo por mi parte, lo que quiero invitarte a hacer es a usarlos todos dependiendo de la necesidad que tu diseño tenga. En [este enlace podrás descargar](#) un archivo HTML de ejemplo (muy sencillo) en donde verás de qué manera pueden convivir estas 3 unidades de medida sin ningún problema para crear una página responsive. También puedes verlo [en GitHub](#) 😊

En lo que sí estamos de acuerdo, es en que el uso de pixeles como unidad de medida debiera desaparecer o por lo menos reducirse tanto como sea posible.

Próximamente estaré lanzando la versión 2.0 de mi curso [Diseño web: HTML y CSS desde cero](#), de hecho ya tengo más de la mitad de las clases grabadas y puedo decirte que el material está muy completo e interesante.

Regístrate más abajo en la lista de interesados para poder avisarte cuando el curso esté listo para lanzarse y para enviarte algunas clases gratis en cuanto estén listas.

Un abrazo,
[@FranciscoAMK](#)

CURSO RECOMENDADO

Webflow Designer para el maquetado web

USD \$12/mes

En este curso conoceremos en detalle todas las herramientas que tenemos disponibles a través del "Designer" de Webflow.

INSCRÍBETE AHORA

TAMBIÉN TE PODRÍA INTERESAR:

Construye tu imperio freelance con Laura López

Una conversación con Laura López hablando sobre cómo ser un diseñador freelance éxito a partir de todas las tácticas y la estrategia que detalla en su libro Imperio Freelance.

[SEGUIR LEYENDO...](#)

La honestidad y los temas de WordPress

Hace un tiempo tuve una reunión con un cliente que necesitaba rediseñar su sitio web luego de un par de años sin recibir ninguna actualización, tanto en contenido como en diseño. El cliente me presentó el sitio actual e inmediatamente identificó el tema que estaba usando (creo haberlo visto en ThemeForest). Le dije “es buena [...]”

[SEGUIR LEYENDO...](#)

El espacio en blanco no está perdido si lo usas con propósito

Cuando hablamos, lo que hacemos es pronunciar un montón de palabras de manera consecutiva. Entre cada palabra hay un breve espacio de silencio y la forma en que utilicemos ese silencio (en conjunto con otros factores de comunicación no-verbal) va a condicionar la forma en que nuestro mensaje es percibido. No es lo mismo decir: [...]

[SEGUIR LEYENDO...](#)

Hay 14 opiniones



Pedro dice:

Los tutoriales de Francisco son siempre magníficos, pero de este me atrevo a decir que el mejor tutorial del mundo de habla hispana respecto a los tamaños en diseño web.

Enhorabuena!!

Saludos desde España

[RESPONDER](#)



Francisco Aguilera G. dice:

Gracias Pedro 😊

[RESPONDER](#)



Santiago dice:

Quisiera contactarme contigo, quiero aprender mas acerca de HTML y CSS, tal vez, clases, turoriales, lo que sea. Por favor contactame para poder hablar bien.

Gracias.

[RESPONDER](#)**Elohim** dice:

Buenísima información... =) sólo sabía de pixeles, y ahora rem y em

[RESPONDER](#)**Francisco** dice:

¡Como siempre, una explicación magnífica y muy amena!
¡Gracias por compartirlo tocayo!
Saludos desde Colombia.

[RESPONDER](#)**Mayra Alejandra Rodriguez Maldonado** dice:

Excelente, muchas gracias!!!

[RESPONDER](#)**Angelo Lopez R** dice:

Hay un dicho en Colombia que dice, "A LA PEPA", eso es este artículo puntual claro y muy bien explicado, gracias!!!

[RESPONDER](#)



Jhon dice:

Hola Francisco, una duda, cuando dices “los navegadores de manera predeterminada definen un font-size de 16px al elemento HTML, por lo tanto, por defecto 1em es igual a 16px.”, es siempre, o por el contrario si tu en el archivo css, el elemento HTML lo defines como 14px, 1em sigue siendo 16px, o entonces toma el valor asignado por ti, osea 14px.

Gracias por estos tutoriales

RESPONDER



Francisco Aguilera G. dice:

Cambia a 14px 😊

RESPONDER



Jhon dice:

ok, muchas gracias

RESPONDER



Carlos dice:

Perfecto muy buena información

RESPONDER



Jesus Fregoso dice:

si divido el numero que quiero obtener entre los 16px por defecto no me arroja el mismo resultado que multiplicarlo por 0.0625??

RESPONDER



Jose Armando Ramirez dice:

Si, funciona igual.

RESPONDER



Edi dice:

Hermosa explicación

RESPONDER

Y tú ¿Qué opinas?

Puede que no te responda, pero ten por seguro que leeré tu comentario. Por si acaso, tu email no se mostrará ;)

Escribe tu opinión...

Tu nombre

Tu email

Tu sitio web

PUBLICAR OPINIÓN



2019 © FranciscoAMK (Francisco Aguilera G.) • Diseñado por [APK Estudio](#)