

Principios de diseño web receptivo

Hay muchos dispositivos que pueden acceder a la web, y vienen en todas las formas y tamaños. El diseño web receptivo es la práctica de diseñar **sitios web flexibles** que pueden responder a **diferentes tamaños de pantalla**, orientaciones y resoluciones.

En este curso, aprenderá a **usar CSS para que sus páginas web se vean bien**, sin importar en qué **dispositivo** se vean.

1. Crear una consulta de medios
2. Hacer una imagen receptiva
3. Use una imagen Retina para pantallas de mayor resolución
4. Hacer que la tipografía sea receptiva

1. Crear una consulta de medios

Las consultas de medios son una nueva técnica introducida en CSS3 que cambia la presentación del contenido en función de los diferentes tamaños de ventana gráfica. La ventana gráfica es el área visible de un usuario de una página web y es diferente según el dispositivo utilizado para acceder al sitio.

Las consultas de medios constan de un tipo de medio y, si ese tipo de medio coincide con el tipo de dispositivo en el que se muestra el documento, se aplican los estilos. Puede tener tantos selectores y estilos dentro de su consulta de medios como desee.

Aquí hay un ejemplo de una consulta de medios que devuelve el contenido cuando el ancho del dispositivo es menor o igual a 100px:

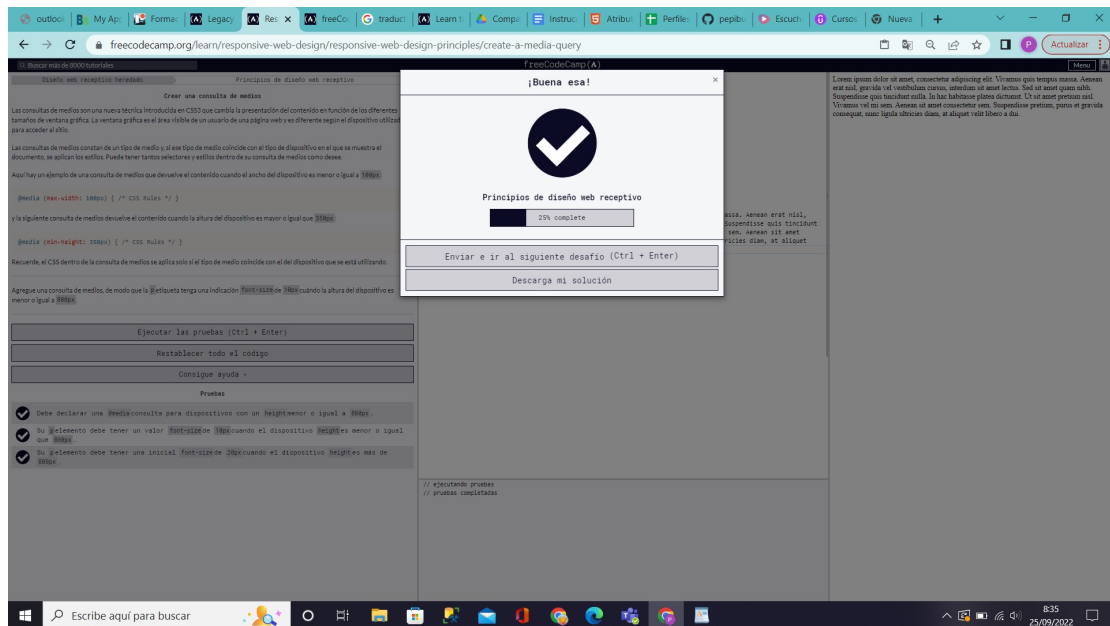
```
@media (max-width: 100px) { /* CSS Rules */ }
```

y la siguiente consulta de medios devuelve el contenido cuando la altura del dispositivo es mayor o igual que 350px:

```
@media (min-height: 350px) { /* CSS Rules */ }
```

Recuerde, el CSS dentro de la consulta de medios se aplica solo si el tipo de medio coincide con el del dispositivo que se está utilizando.

Agregue una consulta de medios, de modo que la etiqueta tenga una indicación font-size de 10px cuando la altura del dispositivo es menor o igual a 800px.



Siguiendo las instrucciones:

Agregue una consulta de medios, de modo que la etiqueta p tenga un tamaño de fuente de 10 px cuando la altura del dispositivo sea menor o igual a 800 px.

la consulta de los medios es:

/* Add media query below (Agregar consulta de medios a continuación)*/

```
@media (max-height: 800px){  
  
  p {  
  
    font-size: 10px;  
  
  }  
  
}
```

y el tamaño del texto será de 10px cuando el ancho del dispositivo sea menor o igual a 800px.

<style>

```
p {  
  
  font-size: 20px;
```

```
}
```

```
/* Only change code below this line */
```

```
@media (max-height: 800px){
```

```
  p {
```

```
    font-size: 10px;
```

```
  }
```

```
}
```

```
/* Only change code above this line */
```

```
</style>
```

```
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus quis tempus massa. Aenean erat nisl, gravida vel vestibulum cursus, interdum sit amet lectus. Sed sit amet quam nibh. Suspendisse quis tincidunt nulla. In hac habitasse platea dictumst. Ut sit amet pretium nisl. Vivamus vel mi sem. Aenean sit amet consectetur sem. Suspendisse pretium, purus et gravida consequat, nunc ligula ultricies diam, at aliquet velit libero a dui.</p>
```

2. Hacer una imagen receptiva

Hacer que las imágenes respondan con CSS es realmente muy simple. Solo necesita agregar estas propiedades a una imagen:

```
img {
```

```
  max-width: 100%;
```

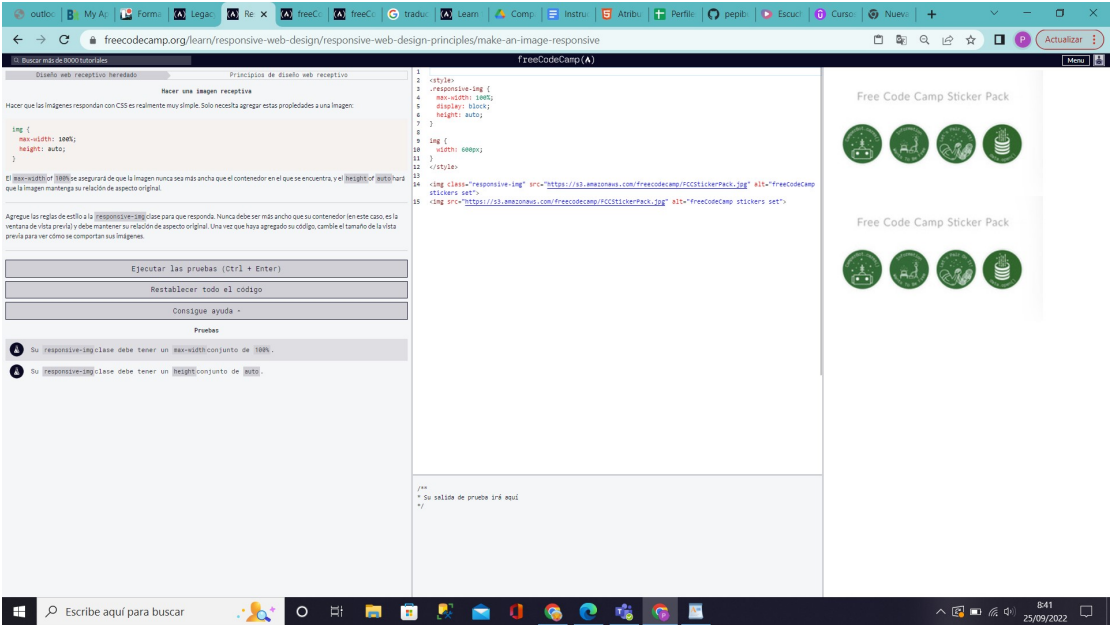
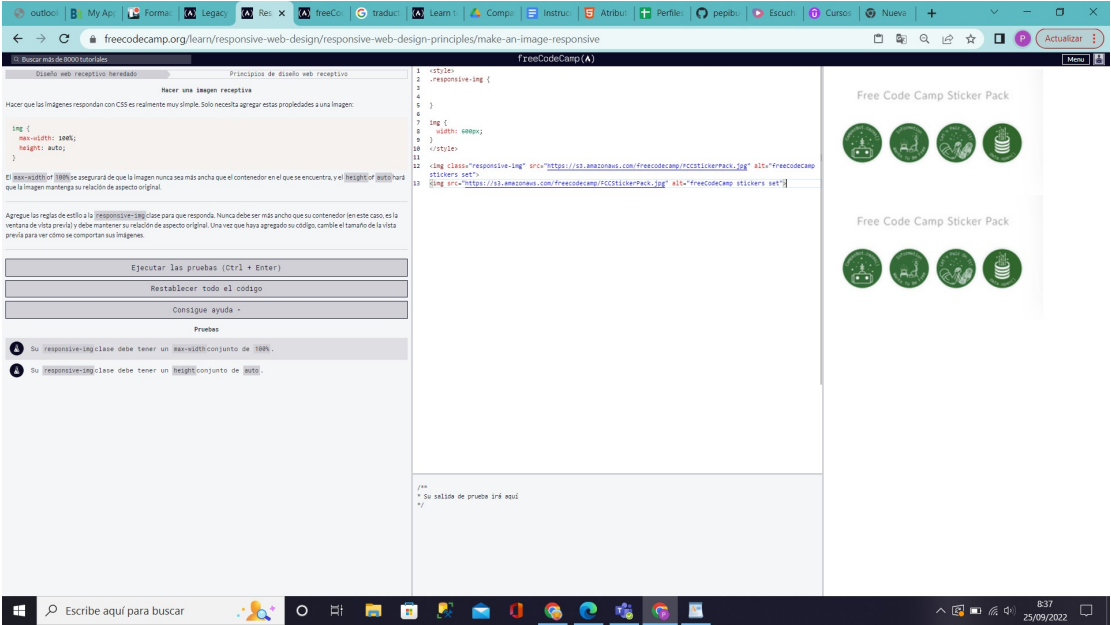
```
  height: auto;
```

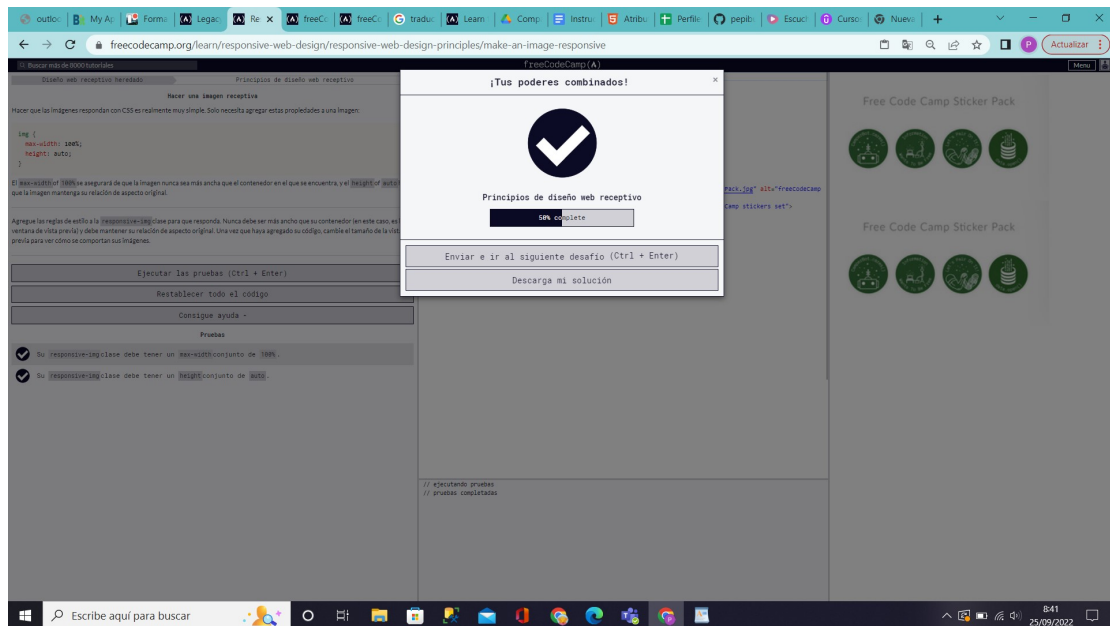
```
}
```

El `max-width` de 100% se asegurará de que la imagen nunca sea más ancha que el contenedor en el que se encuentra, y el `height` de `auto` hará que la imagen mantenga su relación de aspecto original.

Agregue las reglas de estilo a la `responsive-img` clase para que responda. Nunca debe ser más ancho que su contenedor (en este caso, es la ventana de vista previa) y debe mantener su relación de aspecto original. Una vez que haya agregado su código, cambie el tamaño de

la vista previa para ver cómo se comportan sus imágenes.





NO:

```
<style>
```

```
.responsive-img {  
}
```

```
img {  
  width: 600px;  
}
```

```
</style>
```

```

```

```

```

SÍ:

```
<style>
```

```
.responsive-img {
```

```

max-width: 100%;

display: block;

height: auto;
}

img {

width: 600px;

}

</style>





```

3. Use una imagen Retina para pantallas de mayor resolución

Con el aumento de los dispositivos conectados a Internet, sus tamaños y especificaciones varían, y las pantallas que usan pueden ser diferentes interna y externamente. La densidad de píxeles es un aspecto que puede ser diferente en un dispositivo de otros y esta densidad se conoce como píxeles por pulgada (PPI) o puntos por pulgada (DPI). La pantalla de este tipo más famosa es la conocida como "Pantalla Retina" en las últimas computadoras portátiles Apple MacBook Pro y, recientemente, en las computadoras iMac. Debido a la diferencia en la densidad de píxeles entre las pantallas "Retina" y "No Retina", algunas imágenes que no se han creado teniendo en cuenta una pantalla de alta resolución podrían verse "pixeladas" cuando se representan en una pantalla de alta resolución.

La forma más sencilla de hacer que sus imágenes aparezcan correctamente en pantallas de alta resolución, como la "pantalla retina" de MacBook Pro, es definir sus valores width como height solo la mitad de lo que es el archivo original. Aquí hay un ejemplo de una imagen que solo usa la mitad de la altura y el ancho originales:

```

<style>

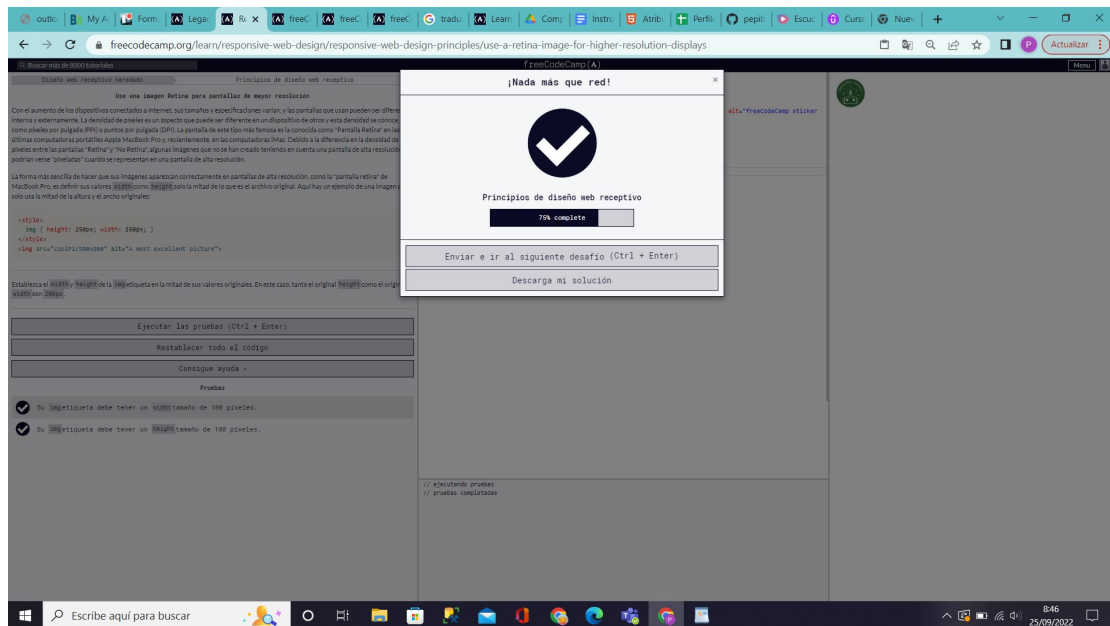
img { height: 250px; width: 250px; }

</style>

```

Establezca el `width` y `height` de la `img` etiqueta en la mitad de sus valores originales. En este caso, tanto el `original height` como el `original width` son 200px.





Siguiendo las instrucciones:

Establezca el ancho y el alto de la etiqueta `img` a la mitad de sus valores originales. En este caso, tanto la altura original como el ancho original son 200 px.

el estilo se convierte en:

```
<style>
```

```
img { height: 100px; width: 100px; }
```

```
</style>
```

tenga en cuenta que la forma más sencilla de hacer que sus imágenes parezcan "retina" (y optimizarlas para pantallas retina) es definir sus valores de ancho y alto como solo la mitad de lo que es el archivo original.

```
<style>
```

```
img { height: 100px; width: 100px; }
```

```
</style>
```

```

```

4. Hacer que la tipografía sea receptiva

En lugar de usar `px` para cambiar el tamaño del texto, puede usar unidades de ventana gráfica para una tipografía receptiva. Las unidades de ventana gráfica, como los porcentajes, son unidades relativas, pero se basan en diferentes elementos. Las unidades de ventana gráfica son relativas a las dimensiones de la ventana gráfica (ancho o alto) de un dispositivo, y los porcentajes son relativos al tamaño del elemento contenedor principal.

Las cuatro unidades diferentes de ventana gráfica son:

VW (ancho de la ventana gráfica): **10vw** sería el **10%** del ancho de la ventana gráfica.

vh (altura de la ventana gráfica): **3vh** sería el **3%** de la altura de la ventana gráfica.

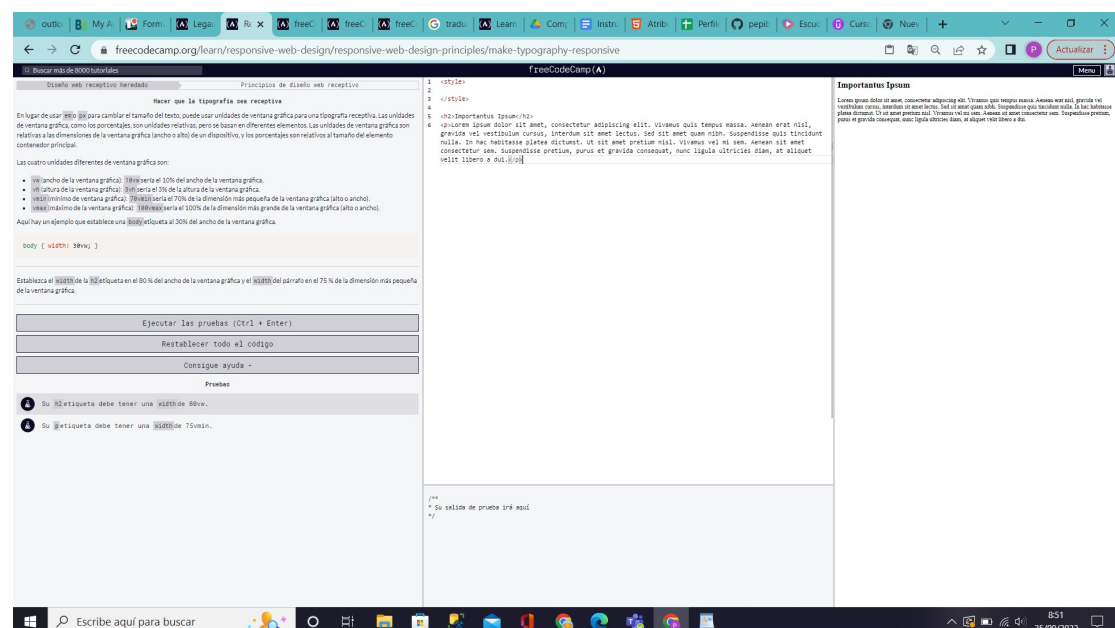
vmin (mínimo de ventana gráfica): **70vmin** sería el **70%** de la dimensión más pequeña de la ventana gráfica (alto o ancho).

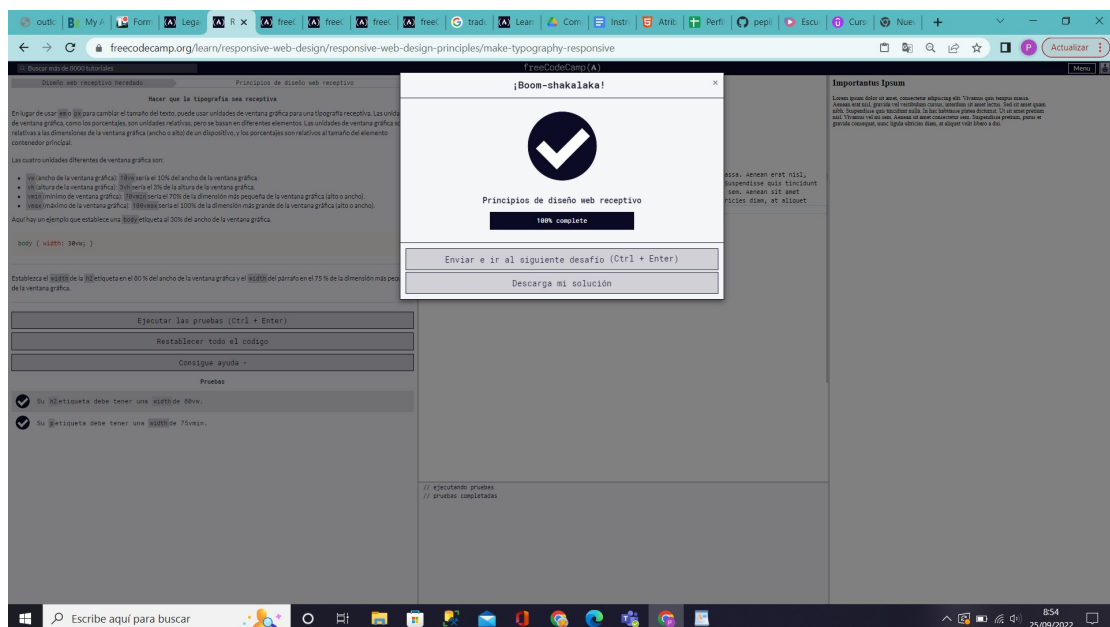
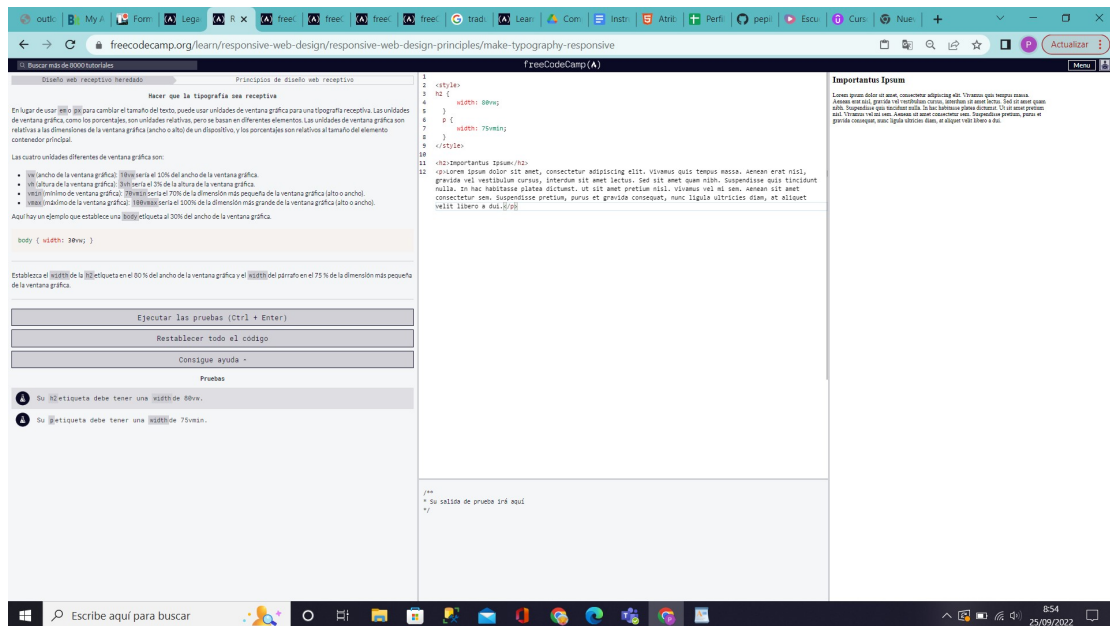
vmax (máximo de la ventana gráfica): **100vmax** sería el **100%** de la dimensión más grande de la ventana gráfica (alto o ancho).

Aquí hay un ejemplo que establece una `body` etiqueta al 30% del ancho de la ventana gráfica.

```
body { width: 30vw; }
```

Establezca el **width** de la **h2** etiqueta en el 80 % del ancho de la ventana gráfica y el **width** del **párrafo** en el 75 % de la dimensión más pequeña de la ventana gráfica.





<style>

h2 {

width: 80vw;

}

p {

width: 75vmin;

}

</style>

<h2>Importantus Ipsum</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus quis tempus massa. Aenean erat nisl, gravida vel vestibulum cursus, interdum sit amet lectus. Sed sit amet quam nibh. Suspendisse quis tincidunt nulla. In hac habitasse platea dictumst. Ut sit amet pretium nisl. Vivamus vel mi sem. Aenean sit amet consectetur sem. Suspendisse pretium, purus et gravida consequat, nunc ligula ultricies diam, at aliquet velit libero a dui.</p>