

## Grid.

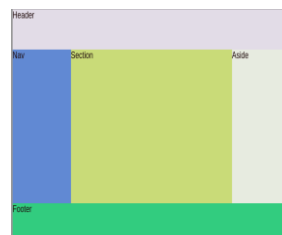
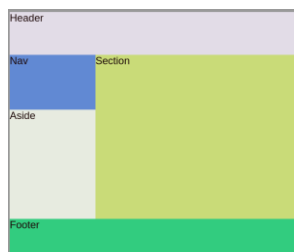
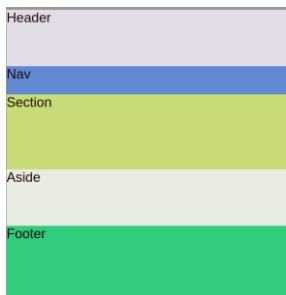
### Actividad

<https://codepen.io/paqui-molina/pen/dyyExwZ>

Con CSS3 Grid, escribe las reglas necesarias para conseguir un responsive Holy Grail layout.

Puntos de ruptura

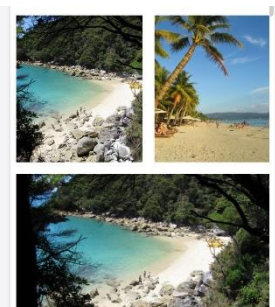
Extra pequeño <576px	Pequeño ≥576px	Medio ≥768px	Grande ≥992px	Extra grande ≥1200px
-------------------------	-------------------	-----------------	------------------	-------------------------



### Actividad

Con la ayuda de CSS3 Grid realiza una galería de imágenes responsive, diferente para tres tamaños de tipo de dispositivo. Técnica mobile first

Ejemplo distribución de imágenes. No tienes que ceñirte al ejemplo puedes hacer la distribución que tú quieras. Recuerda usar span para que una imagen ocupe más de una fila/columna. Recuerda modificar el número de columnas según el ancho de dispositivo. Un dispositivo mayor tendrá más columnas que uno más pequeño.



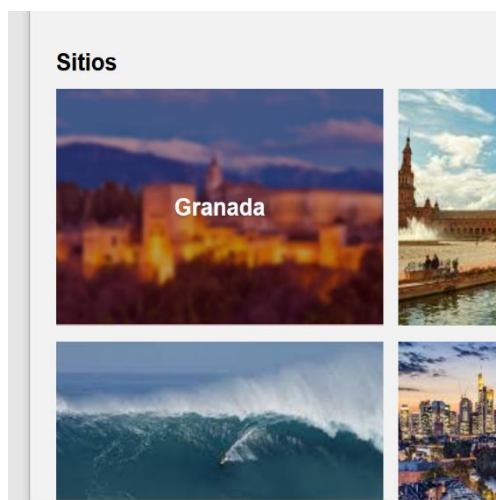
## Actividad

Con la ayuda de CSS3 Grid realiza una galería de imágenes responsive, diferente para tres tamaños de dispositivo. Técnica mobile first.

En la cuadrícula deben aparecer 6 imágenes o más



Al pasar el ratón:



Deben tener un efecto hover:

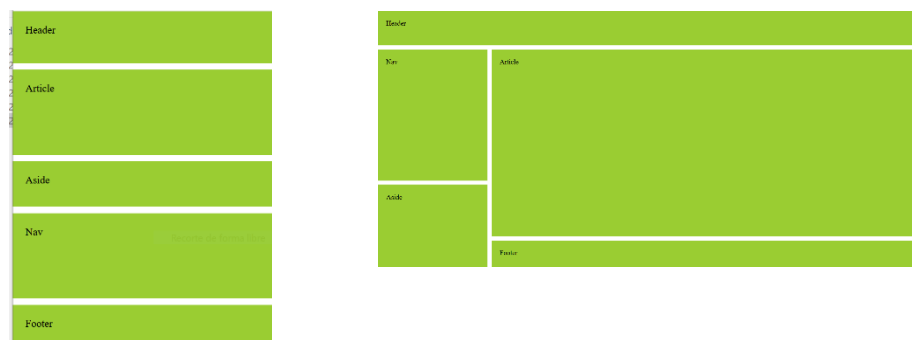
1. Aparecerá el título de la imagen centrado (mejor flexbox)
2. El fondo será ligeramente rojo (rojo con transparencia entre 0,3 y 0,6 aprox).
3. La imagen se desenfoca. Propiedad [filter con blur](#) de css. Por ejemplo `filter: blur(5px);`

Propiedades CSS que, quizás es necesario usar: z-index, opacity, transformaciones y transiciones.

Os animo a realizar otra galería con otro tipo de animaciones.

## Actividad

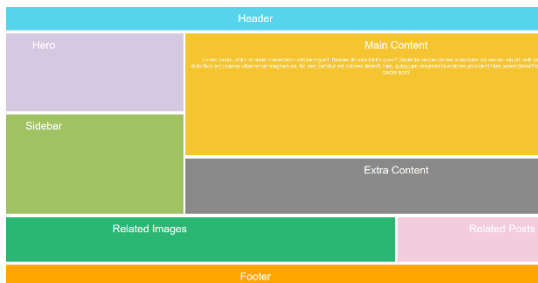
Con CSS3 Grid, escribe las reglas necesarias para conseguir un responsive layout como el de la imagen. Técnica mobile first



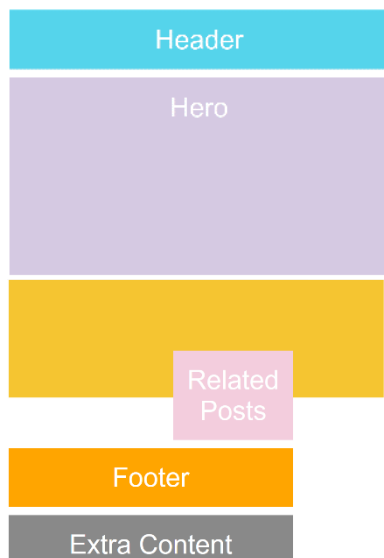
## Actividad

Con CSS3 Grid, escribe las reglas necesarias para conseguir un responsive layout como el de la imagen.

Vista escritorio: Ocupa más de una pantalla



Vista móvil:



Una vez conseguido el layout con Grid. Inserta contenido en header y en footer usando flexbox. Para el header puedes usar cualquiera de los que hemos visto en clase. Para el footer puedes poner sobre nosotros, redes sociales y autor. Todo maquetado con flexbox.

Ejemplo header



Ejemplo footer



Y así podemos ir completando el resto del grid. Es decir, primero se diseña el layout y después se complete con el resto de elementos.

Para que la hoja de estilos no sea tan grande, podéis crear diferentes hojas de estilo, ejemplo: layout.css; header.css, footer.css, ...

Después solo tienes que enlazarlas todas en el document HTML, así es más fácil mantene