

- Conversion a sass.
  - Primeros pasos.
  - Creación de una cuadrícula flotante mediante SASS.
    - código html de comprobación
    - Código CSS
  - Creación de la sección Feature mediante SASS.
  - Creación de la sección stories mediante SASS.
    - Primera imagen sin hacer hover en las imágenes.
    - Segunda imagen haciendo hover en las imágenes.

# CONVERSION A SASS.

---

## PRIMEROS PASOS.

Siguiendo con el proyecto de la **Activad 1**.

1. El primer paso que vamos ha realizar es encontrar o descubrir más elementos que podemos convertir a variables.
2. Ahora vamos a comprobar como nos ayuda la nomenclatura BEM cuando anidemos.
3. En este punto, deberemos aplicar la estructura de directorios y ficheros en nuestro proyecto. Vuestro trabajo es incorporar el código que vosotros consideréis en cada uno de los ficheros. Puede haber más de una interpretación.

A continuación deberéis importar todos estos ficheros a `main.scss`. Tener cuidado en el orden de importación de los ficheros en el `main.scss`. Vamos a crear los siguientes directorios y ficheros dentro de estos. Estos son los básicos que tenemos que crear en todo proyecto que vayamos a realizar pequeño o grande. Aunque no se utilice alguno de los directorios.

```
//BASE:  
    _animations.scss  
    _base.scss  
    _typography.scss  
    _utilities.scss  
//ABSTRACT:  
    _variables.scss  
    _mixins.scss  
    _functions.scss  
  
//COMPONENTS:  
//LAYOUT:  
//PAGES:  
    _home.scss
```

# CREACIÓN DE UNA CUADRÍCULA FLOTANTE MEDIANTE SASS.

Siguiendo con el anterior proyecto vamos a crear nuestro sistema grid. Que lo vais a utilizar todas las siguientes secciones.

## código html de comprobación

1. Crea una **sección** con la clase .grid-test. Dentro de esta vamos a crear nuestro primeras columnas y filas.
2. Para empezar, vamos a crear dos columnas que abarquen todo el espacio. Para ello crearemos una div con la clase row y dentro de esta otro div con la clase **col-1-2**.
3. En la siguiente fila vamos a crear **col-1-3**. Por lo que crearemos la misma estructura de antes pero con tres elementos y con la clase **col-1-3**.
4. En la siguiente fila vamos a crear **col-1-3** y **col-2-3**. Esta última se extiende en dos columnas.
5. En la siguiente fila vamos a crear **col-1-4** para todas. Es decir tenemos cuatro columnas.
6. En la siguiente fila vamos a crear **col-1-4** todas menos la última que se extiende dos columnas **col-2-4**. Es decir tenemos tres columnas.
7. En la siguiente fila vamos a crear **col-1-4** en todas menos la última que se extiende en tres columnas **col-3-4**. Es decir tenemos dos columnas.

## Código CSS

1. Ahora, crearemos un fichero llamado `grid.css` en el directorio correspondiente. Vamos a crear los estilos para las filas y las columnas.
2. A la fila le vamos a dar un ancho máximo de 1140px (crea una variable que la podamos reutilizar). Calculado en `rem`. Tenemos que tener en cuenta que en html hemos puesto % al font-size. La centramos y le damos el color `#eee`. Además le damos un margen de separación entre las filas de 80px, ponlo en `rems` y crea una variable para tal fin. También vamos a crear una variable de separación horizontal de 60px (ponlo en `rems`).
3. El problema que nos encontramos en el anterior ejercicio es que la última fila no tiene que tener margen. Arreglalo.
4. Ahora vamos a crear las clases `col-1-2` dentro de row. Teniendo en cuenta que hay que calcular su `width` y establecer el tamaño de separación de las columnas (calles). En esta clase solo tenemos dos columnas. Para calcular el `width` de cada una de las columnas utilizaremos `calc()`. Busca información sobre esta función. (Utiliza la variable de separación horizontal que hemos utilizado anteriormente).
5. Ten en cuenta que hay estilos comunes para todos las columnas que empiecen por `col-`.
6. Los otros tamaños para las otras clases se tienen que calcular según los elementos que tengamos y el margen entre las columnas.

## CREACIÓN DE LA SECCIÓN FEATURE MEDIANTE SASS.

1. Ahora vamos a empezar con la siguiente sección de nuestra web. Para lo cual primero, vamos a ver como funciona la fuente de iconos Awesome. Que es la misma que utiliza Bootstrap. Vamos a utilizar la versión 5 de esta fuentes que la podemos descargar [aquí](#).
2. Para utilizar este tipo de fuentes podemos optar por dos caminos. El primero de todos y más compatible con los navegadores es que una vez que lo descarguemos copiemos la carpeta webFonts y fontawesome-all.css a la carpeta styles. A continuación referenciamos con `<link href="/styles/fontawesome-all.css" rel="stylesheet">`
3. ¿Cómo se utiliza? `<i class="fas fa-chess-king"></i>`. Esto incluirá un ícono de una pieza de ajedrez.
4. El segundo camino es utilizar `fontawesome-all.js` y guardarlo en nuestro proyecto. Como está en beta, no tenemos link en npm pero si para frameworks como angular etc.... . Crearíamos un nuevo directorio llamado src/js y lo copiaríamos allí. Y después procederíamos igual que antes, introduciendo el html anterior. La ventaja si observáis

es que nos da más estilos y potencia de uso.

Las clases de los iconos y tipos los podéis visitar [aquí](#).

- Para esta nueva sección crearemos una clase llamada sea `feature-features`.

Tiene que tener un background con una foto y un gradiente lineal como en el header.

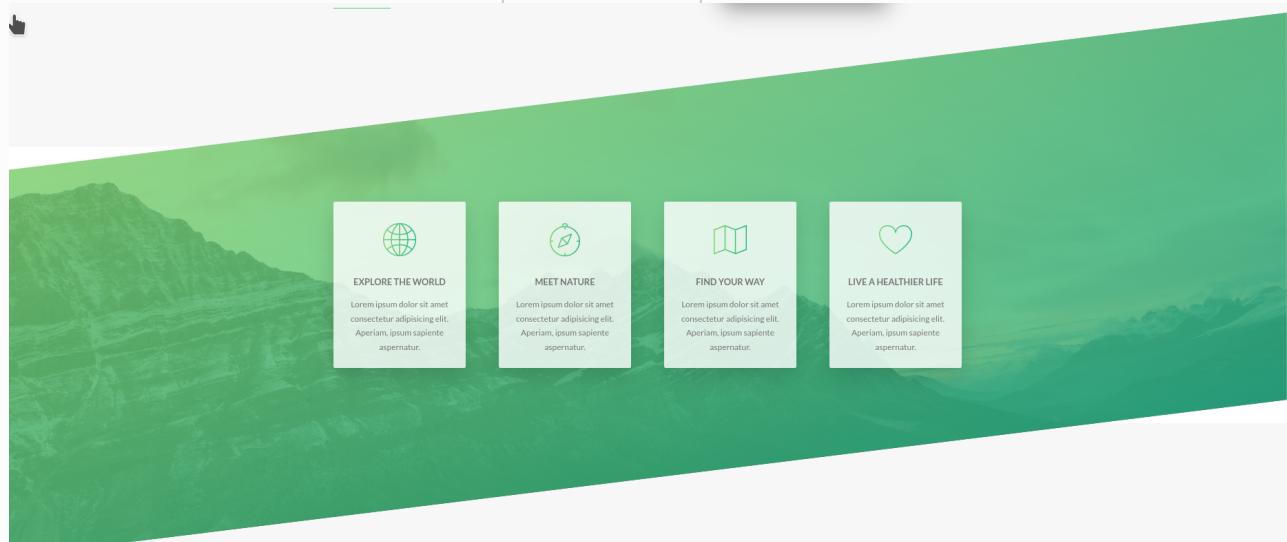
Además utiliza clip-path para dejarlo como un trapecio.

- El objetivo es, crear cuatro cajas (Utiliza el grid y crea la clase `feature-box`) con un ícono en la parte de arriba y a continuación una cabecera (`.heading-tertiary`), después dos párrafo que lo explique

- A los iconos les tienes que dar algunos estilos (`feature-box__icon`).

- A cada una de las cajas(`feature-box`). Cuando hagamos hover tiene que subir y aumentarlos de tamaño.

- Todavía no hemos terminado con la sección feature. Vamos a darle un aspecto parecido a el header pero esta vez no vamos a utilizar clip-path. Vamos a transformar la sección con skew unos grados en negativo. A continuación, notarás que las cajas con las explicaciones también se han deformado. Deberás arreglarlo también con skew con valor positivo. Ahora también notarás que arriba hay un espacio en blanco con la otra sección, ajustalo con `margin-top` pero con valor negativo.



## CREACIÓN DE LA SECCIÓN STORIES MEDIANTE SASS.

En esta sección, solo os voy a dejar las imágenes y algunas propiedades que necesitaréis para realizar el css.

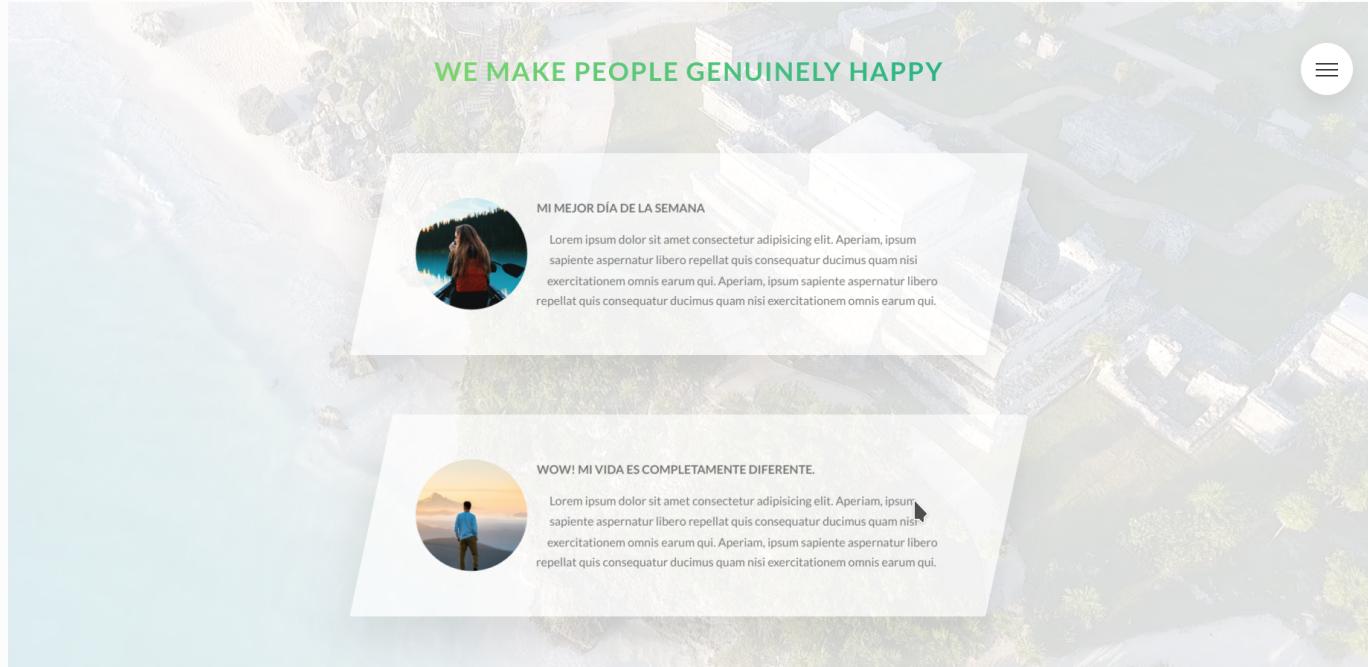
Esta propiedad realiza el efecto de hacer la forma del circulo.

```
shape-outside: circle(50% at 50% 50%);  
-webkit-clip-path: circle(50% at 50% 50%);  
clip-path: circle(50% at 50% 50%);
```

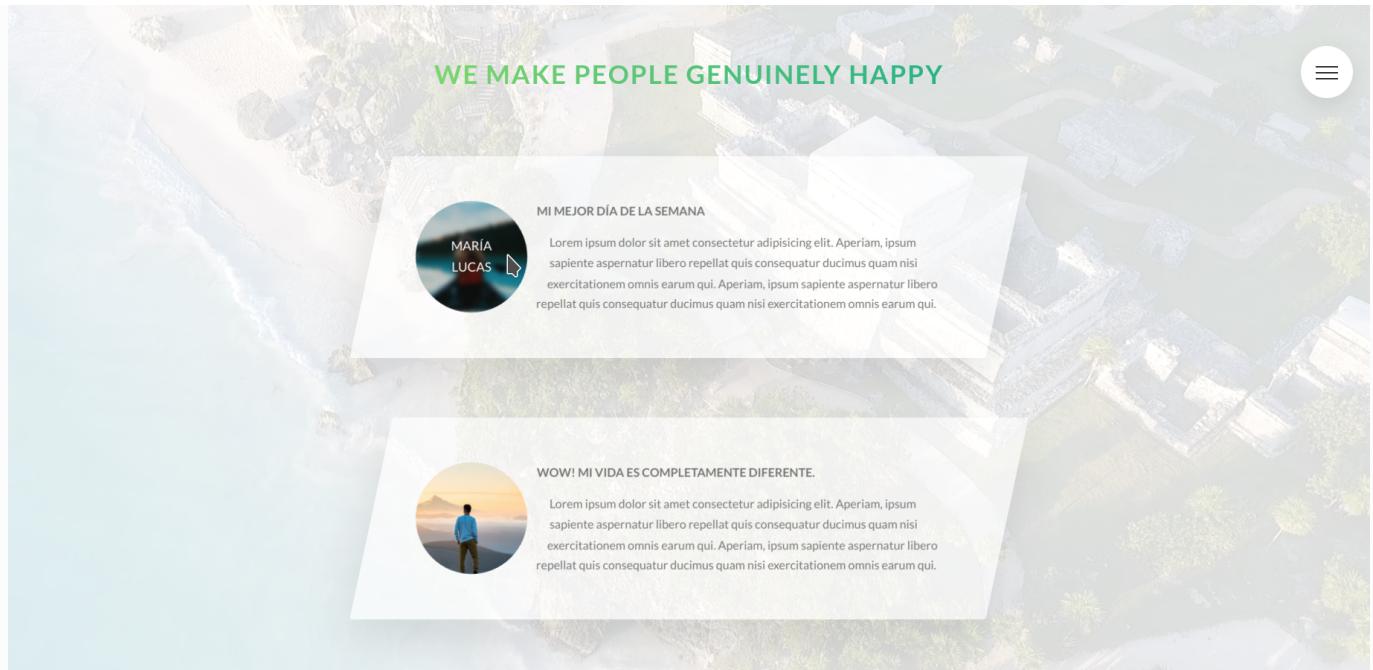
## Primera imagen sin hacer hover en las imágenes.

También tenemos que considerar que el fondo no es una imagen si no que es un video.

Buscar un video y ponerlo que cubra toda la a la `section`. Utilizando el código html5 con el cual podemos poner varias fuentes, con distinto formato, formato mp4 y webm



## Segunda imagen haciendo hover en las imágenes.



Tener en cuenta que vosotros tenéis que decir que clases , ficheros vais a crear y en que directorio. Ya que la estructura de directorios es la que os pongo en el primer ejercicio.