

UD 4. Diseño con “estilo”. Interfaces que dejan huella Sitio web “Librería Bootstrap”			
<b>MÓDULO:</b>	Diseño de Interfaces Web	<b>Grupo</b>	2º DAW

Resultados de aprendizaje y criterios de evaluación cubiertos en esta práctica.

#### RA2.Crea interfaces Web homogéneos definiendo y aplicando estilos

CE	Actividad
j) Se han analizado y utilizado preprocesadores de estilos para traducir estilos comunes a un código estándar y reconocible por los navegadores.	

A partir del HTML proporcionado, debes **generar los archivos.scss que permitan la modularización (variables, funciones, mixins e index, incluye comentarios que indiquen el apartado que se está implementando para facilitar la corrección)** que compile correctamente a **main.css** y cumpla los siguientes requisitos:

**(incluir el texto marcado como comentario en la sección del archivo .scss en la parte que implemente la funcionalidad que se pide):**

#### Estilos generales de tabla

- La tabla debe usar border-collapse: collapse
- Las celdas (th y td) deben tener:
  - Borde negro de 1px
  - Padding de 1rem

#### Colorear columnas automáticamente

Queremos que:

- Las columnas pares tengan un color
- Las columnas impares tengan otro color

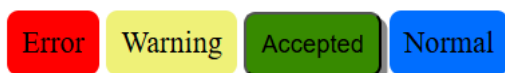
**Nota:** para comprobar si un nº es par utilizar  $\%2 == 0$

Para ello crea una **función llamada colum-color** que reciba el número de columna: si es par → devuelva color columnas pares, si es impar → devuelva color columnas impares. Los colores de las columnas son #CCCCC para las pares y #888888 para las impares. Usa la función creada para asignar el color.

#### Sistema de botones

- **Define variables** para colores y borde:
  - error: #FF0000;
  - warning: rgb(239, 241, 120);
  - accepted: rgb(55, 138, 0);
  - normal: rgb(0, 110, 255);
  - shadow: #888;
  - border-radius: 0.4rem
  - Crea un mapa llamado btn-colors que relacione nombre → color
- Crea una **clase base .btn** que tenga:
  - display inline-block
  - padding 0.5 rem

- border-radius valor de la variable
- text-align center
- sin subrayado
- Efecto hover con sombra, efecto active que cambie fondo a naranja y texto blanco
- Genera automáticamente los colores para que se muestren así (el botón verde tiene aplicado el efecto de hover):



### Lista de estados

- Crear un mixin llamado tag-style con los parámetros que consideres necesarios para que:
  - Todas las entradas de la lista tengan las siguientes propiedades:
    - padding: 8px 12px;
    - margin: 5px 0;
    - list-style: none;
    - border-radius: 20px;
    - font-weight: bold;
    - text-transform: uppercase;
    - width: fit-content;
  - En las entradas de la lista se apliquen de forma automática los colores según la clase que tengan:
    - success → fondo verde claro
    - warning → fondo amarillo
    - danger → fondo rojo claro
    - info → fondo azul claro
    - Si no coincide → fondo gris

## Lista de Estados

COMPLETADO

PENDIENTE

ERROR

INFORMACIÓN

**Nota: se valorará el control de errores tanto en los mixins como en la función. También se valorará una correcta modularización.**

### Instrucciones de realización y entrega:

Esta práctica es **individual**. El alumnado deberá entregar:

- Carpeta del proyecto (zip):
  - Archivo HTML
  - Carpeta CSS y Carpeta SCSS
  - Enlace a github con el proyecto.

**apellidos\_nombre\_UD4\_RA2\_j\_Prueba técnica Sass.zip**

## Criterios de evaluación

---

- No se pueden escribir manualmente nada en el archivo css, se debe generar automáticamente a partir del scss.
  - Debes usar estructuras de control, mixin y funciones de Sass.
  - Estructura los archivos sass adecuadamente: variables, mixin y funciones y en archivos externos.
-