

# TPO 1 / Trabajo Práctico Grupal 1

## Enunciado

Se buscará clonar la página de tienda de café con una nueva temática, puede ser desde tienda de mate, tienda de té o lo que se les ocurra en el grupo pero debe ser de la misma temática.

El tema es clonar la estructura, sus fuentes y estilos generales. Lo que pueden cambiar, además del contenido, son los colores principales y secundarios. Desde el blanco y negro hasta (especialmente) el color secundario marrón dorado. Sean elegantes y profesionales con su interfaz.

La idea principal del TPO es que aprendan a trabajar en grupo y a clonar interfaces web. *Esto va a reforzar habilidades y competencias fundamentales para su vida profesional*, el objetivo es que al final de esta práctica tengan experiencia trabajando en grupo, comunicándose y mejorando sus habilidades blandas, pudiendo crear y clonar sitios web, sepan trabajar con git y github de manera colaborativa, deployar una web y además tengan una hermosa página para mostrar como portfolio.

## Requisitos

Clonaremos 7 páginas, desde su index.html hasta las demás páginas, a excepción de la página **locales**. Es decir, las páginas a clonar y sus archivos serán:

- **index.html**: La página principal <https://tiendadecafe.com.ar/>
- **tienda.html**: La página Nuestro Café <https://tiendadecafe.com.ar/tienda/>
- **tutoriales.html**: La página Tutoriales <https://tiendadecafe.com.ar/tutoriales/>
- **menu-tienda.html**: La página Nuestro Menú <https://tiendadecafe.com.ar/menu-tienda/>
- **trabaja-en-tdc.html**: La página Trabajá con Nosotros <https://tiendadecafe.com.ar/trabaja-en-tdc/>
- **franquicias.html**: La página Franquicias <https://tiendadecafe.com.ar/franquicias/>
- **contacto.html**: La página Contacto <https://tiendadecafe.com.ar/contacto/>

Las webs tendrán la siguiente estructura de carpetas a modo de ejemplo:

```
trabajo_practico_obligatorio1/
|
├─ index.html           # Archivo HTML principal (página de inicio)
├─ about.html          # Página adicional de ejemplo
├─ contact.html         # Otra página adicional de ejemplo
|
├─ css/
|   └─ styles.css       # Archivo principal de estilos CSS (pueden incluir alguno más)
|
├─ js/
|   └─ scripts.js       # (opcional) Todavía no llegamos a este punto
|
├─ img/
|   └─ logo.png         # Ejemplo de logo o imagen
```

```
|   |─ background.jpg      # Imagen de fondo
|   |─ icon.svg           # Ejemplo de icono SVG
|   |
|   └─ assets/
|       └─ fonts/         # Carpeta para fuentes personalizadas
|           └─ custom-font.ttf # Ejemplo de fuente personalizada
```

- **Manejo de tamaños en los contenedores.** Deberán usar unidades relativas `%`, `flex` o `vw` para sus contenedores padre principales en el width y unidades como `px` o `vh` en el height, si es que trabajan con height. El uso de **Flex** soluciona mucho estos problemas del espaciado y el posicionamiento. **Usen flex.**
- Se valorará muy positivamente el correcto manejo de **elementos en bloque y elementos en línea**. Deben conocerlos bien para luego poder posicionar toda su página. No usen más saltos de línea como `<br>`. Esto además va a ser **fundamental para trabajar con flex**.
- **Buen conocimiento de box model** Deberán poder manejarse cómodamente con márgenes, bordes y padding. Deberán mostrar su conocimiento para acomodar su contenido y su espaciado en el documento.
- **Centrado de elementos:** Con text-align para textos, y `margin: 0 auto;` o `flex`.
- **Manejo de fuentes:** Agregado de un `font-family` propio, descargado o de fonts google y un `font-size` que emplee unidades relativas en versión mobile.
- **Uso de iconos:** Usando font-awesome o la fuente que se prefiera. Mi recomendación es descargarlos en [freepik](#).
- **Manejo de enlaces con pseudoclases:** Se pueden eliminar sus estilos, pero deberían tratarse con `a:link`, `a:visited`, `a:hover`, `a:active`.
- **Uso de combinadores CSS:** Usen en su código varios, *combinador descendiente*: `div p {}` y *combinador directo*: `div > p`. De esta manera se podrán evitar usar en exceso ids y clases.

## Responsividad

Deberán hacer que su página sea responsiva y recolocar los elementos acorde al tamaño de la pantalla. Su proyecto debe llevar 4 tamaños máximos de pantalla.

```
/* Tablet horizontal */
@media (max-width: 1080px) { }
```

```
/* Tablet vertical */
@media (max-width: 768px) { }
```

```
/* Mobil 1 */
@media (max-width: 480px) { }
```

```
/* Mobil 2 */
@media (max-width: 375px) { }
```

# Recordatorios

- Es importante que sean estrictos y prolijos con los nombres de los ficheros, desde sus archivos html y css hasta sus imágenes.
- Comprueben que tienen bien las rutas, sepan navegar entre carpetas. `../` nos permite retroceder una carpeta atrás. Para avanzar podemos poner directamente el nombre de la carpeta o `./`, por ejemplo desde `index.html` la ruta podría ser `css/styles.css` o `./css/styles.css`. Recurren a la ayuda de navegación que les va tirando Visual Studio Code para encontrar los archivos. *Recuerden que cuando vayan a poner una ruta, fíjense desde que carpeta están, si en la raíz o en una carpeta superior y naveguen desde ese lugar.*
- Recuerden tener un código limpio, bien comentado y bien indentado. Debe ser limpio, claro y fácil de entender. Separen los distintos bloques de su CSS con caracteres bien distinguidos como por ejemplo:

```
/*//////////////////// AJUSTES GENERALES////////////////////*/  
  
/* _____Portada_____ */
```

- **Opcional** Incluyan en un archivo `README.md` en la carpeta raíz de su proyecto (donde está el `index.html`) donde hagan una pequeña descripción de su proyecto usen `#` para el título. Este se puede crear por defecto cuando se crea un nuevo repositorio en github. También se valorará incluir una imagen de muestra en nuestro README, este puede ser una captura de pantalla de nuestra web, por ejemplo, `sample.png` y se puede enlazar en nuestro README como ``

## Aclaraciones

- No es necesario copiar la gran cantidad de elementos que hay en la página, como la galería de imágenes o productos. Con que incluyan algunas imágenes de muestra es suficiente, la idea es clonar la estructura de la página.
- Se valorará que se incluyan las animaciones al pasar el mouse arriba de los elementos, desde cambios de colores de fondo y de fuente. Esto mejora la experiencia de usuario en la página, pero no es necesario incluir las animaciones, transiciones y efectos en las imágenes que hay en la web.
- Recuerden que el nav y el footer son genéricos a todas las páginas, de manera que este se incluirá en todas y la diferencia será cambiar el contenido
- Algunas páginas son muy parecidas, como Trabaja con nosotros, Franquicias y Contacto por un lado o Nuestro Café, Tutoriales y Nuestro Menú
- No se pedirá una funcionalidad como la galería de imágenes principal ya que todavía no llegamos hasta ahí y basta con poner una imagen de portada. De hecho para aquellos elementos que tengan una galería, copienlos como están pero sin la funcionalidad de desplazar las imágenes
- Repártanse el trabajo en el grupo de manera pareja para no hacer páginas iguales y repetidas una misma persona, si alguien hace una página tipo formulario como Franquicias, que haga también una tipo Nuestro Café.

## Recursos

---

- En la web, los tiempos de carga son cruciales, recuerden no subir archivos muy pesados, idealmente menos de 500kb cada imagen. Pueden usar [compresores de imagen](#) o herramientas para recortar, comprimir y redimensionar como [picresize](#).
- [Documentación Markdown](#).
- [Guía visual flexbox](#).