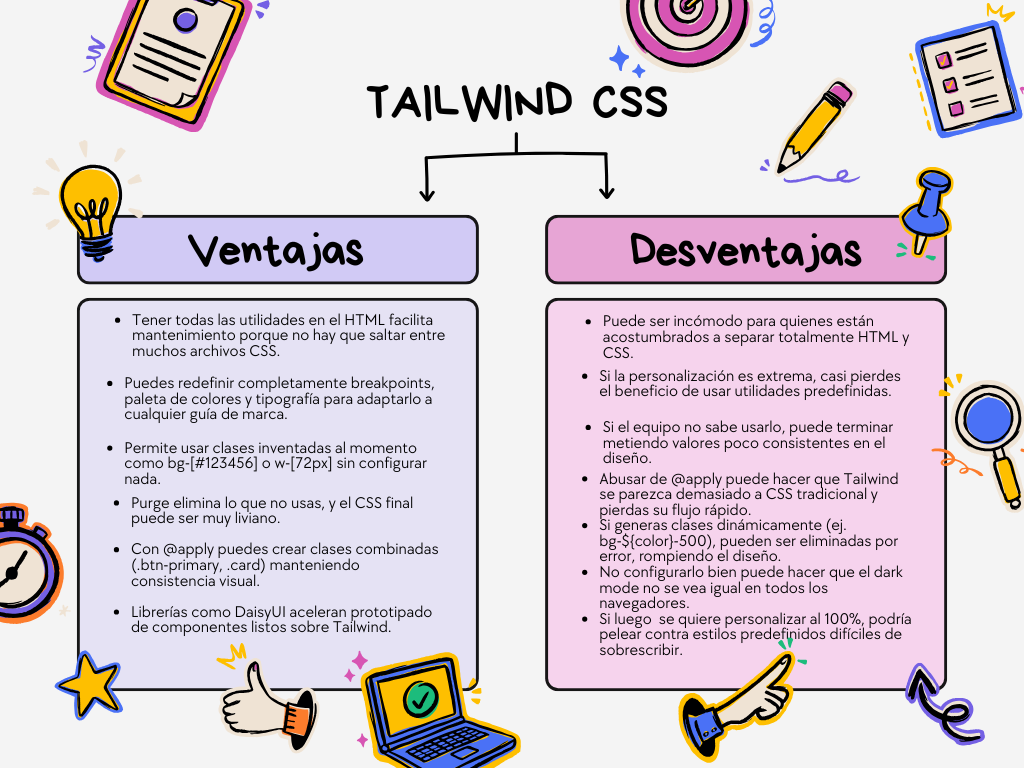
**Producto:** Cuadro comparativo y lista de observaciones sobre el uso de Tailwind.

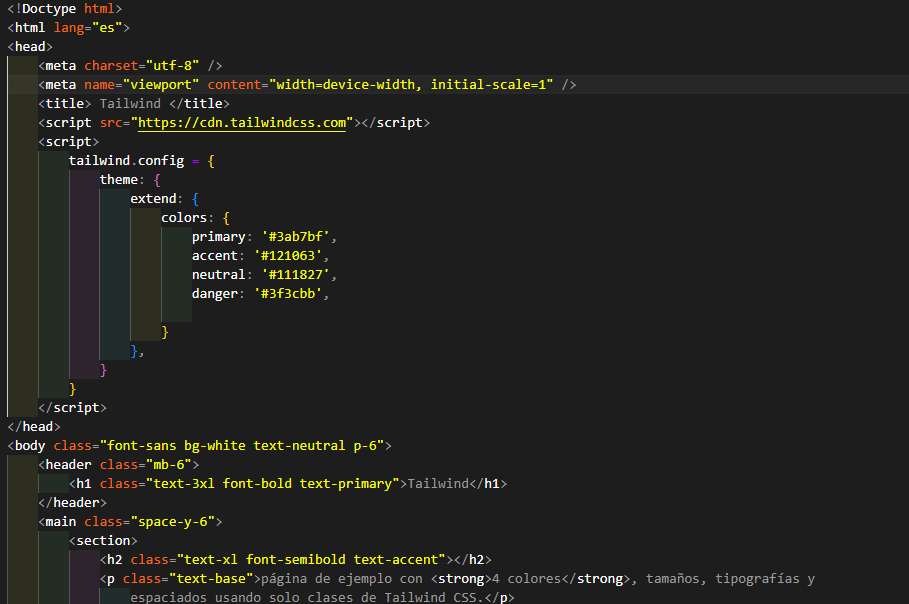
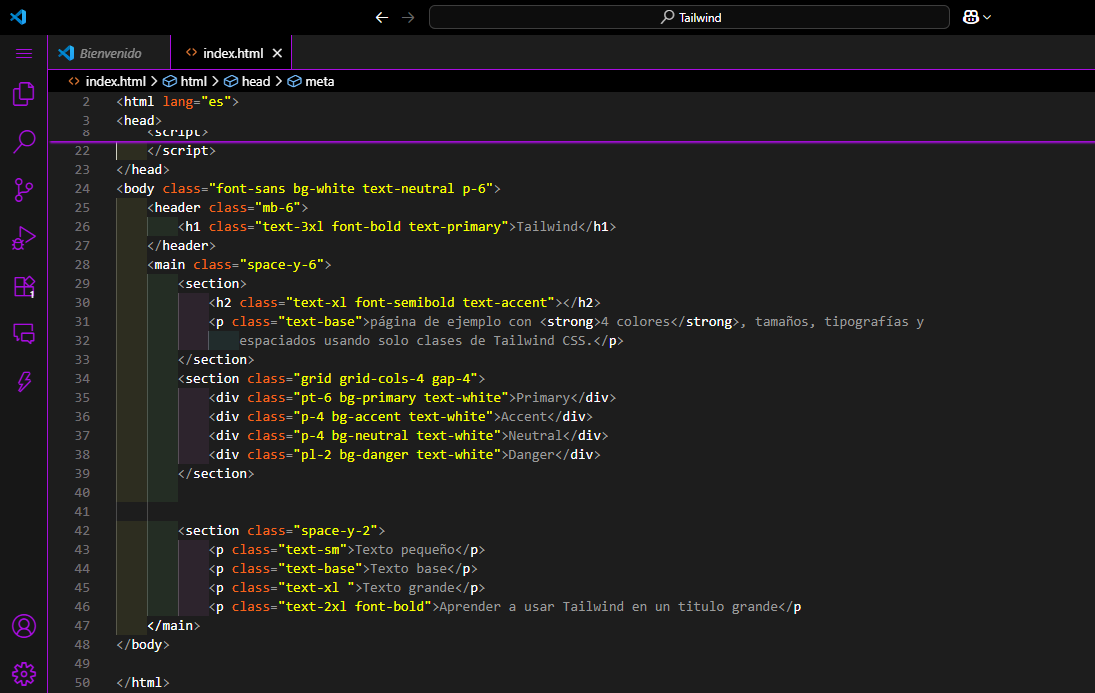
**LISTA DE OBSERVACIONES TAILWIND CSS**

1. **La cantidad de clases en HTML es un indicador del estado del diseño**  
   Si un componente tiene demasiadas utilidades repetidas (px-4 py-2 en todo), es señal de que deberías crear una clase con @apply para limpiar y unificar estilos.
2. **La clave del rendimiento no está solo en el purge**  
   Lo que más infla el CSS en Tailwind no es la cantidad de clases usadas, sino la cantidad de **variantes activas** en tailwind.config.js. Reducir variantes en producción puede cortar el peso a la mitad.­
3. **El “diseño responsivo” en Tailwind es más potente si piensas de grande a pequeño**  
   Aunque la filosofía mobile-first es estándar, con Tailwind muchas veces es más eficiente diseñar desde lg: hacia abajo para ahorrar repeticiones de clases.
4. **Puedes detectar inconsistencias de diseño directamente en el HTML**  
   Al ver clases repetidas con valores distintos (text-gray-600 y text-gray-700 mezclados) sabes que el diseño no es coherente. Esto es algo que en CSS tradicional no es tan evidente.
5. **La elección de nombres en tailwind.config.js puede ahorrar trabajo a todo el equipo**  
   Por ejemplo, renombrar primary a un color corporativo evita que la gente meta colores hex arbitrarios que rompan la identidad visual
6. **El verdadero poder del modo oscuro es la estrategia mixta**  
   No se trata de elegir entre media o class, sino usar ambos: media para que el usuario reciba un dark mode automático y class para darle un switch manual.
7. **Tailwind puede ayudarte a refactorizar diseño antiguo sin reescribir todo el CSS**  
   Puedes ir migrando componentes uno por uno a utilidades de Tailwind mientras mantienes el resto del CSS viejo activo.
8. **No todo debe ser una clase utilitaria**  
   Los componentes con mucha interactividad o animaciones complejas funcionan mejor si mezclas Tailwind para lo básico y CSS puro o frameworks de animación para lo avanzado.
9. **La productividad depende más de los atajos del editor que del propio Tailwind**  
   Sin autocompletado, snippets y atajos de teclado, Tailwind puede sentirse más lento que CSS normal.

***“Si tuvieras que diseñar hoy una página web desde cero, ¿qué herramientas usarías para darle estilos y por qué? ¿Crees que podrías hacerlo más rápido y eficiente con una herramienta como Tailwind CSS?”***

con tailwind el proceso sería más rápido y eficiente, sobre todo porque no empiezo desde cero con cada elemento: ya tengo una “caja de herramientas” de estilos listos para usar. Claro, no es magia; si quiero algo muy específico o creativo, puedo complementarlo con CSS puro. Pero en la mayoría de los casos, Tailwind me permite avanzar más rápido, mantener consistencia y entregar un producto pulido en menos tiempo.

**Producto:** Página HTML básica con estilos aplicados (4 colores, tamaños, tipografías, espaciados) únicamente con clases de Tailwind.

  
****

