



# De lo Esencial a lo Excepcional

Una guía para dominar el estilo de los enlaces con CSS.



# El Punto de Partida: El Enlace sin Estilos

## Un enlace simple

```
```html
<p><a href="#">Un enlace simple</a></p>
```html
```

### Por defecto, un enlace es:

- **Subrayado:** Para distinguirlo del texto normal.
- **Azul:** Si no ha sido visitado.
- **Morado:** Si ya ha sido visitado.
- **Interactivo:** El cursor cambia a una mano (pointer).
- **Accesible:** Muestra un contorno al recibir el foco del teclado.
- **Reactivo:** Cambia a color rojo cuando se activa (al hacer clic).



# El ADN del Enlace: Entendiendo sus 5 Estados



**`:link` (No visitado)**

El estado por defecto. El enlace espera ser descubierto.



**`:visited` (Visitado)**

El enlace ya ha sido visitado y está en el historial.



**`:hover` (Sobrevolado)**

El cursor del ratón se encuentra sobre el enlace.



**`:focus` (Enfocado)**

El enlace ha sido seleccionado, usualmente con el teclado (tecla Tab).



**`:active` (Activo)**

El momento exacto en que el enlace está siendo presionado.



# La Persistencia de los Estilos Clásicos

¿Por qué los estilos por defecto son casi idénticos a los de los primeros navegadores de los 90?

## Perspectiva del Diseñador: La Fuerza de la Costumbre

Décadas de uso han creado una expectativa universal en el usuario: **azul y subrayado significa 'clicable'**. Romper esta convención sin una alternativa visual clara puede generar confusión y frustración. Antes de cambiar, pregúntate: ¿Mi nuevo diseño comunica 'interactividad' de forma igual o más efectiva?



# La Regla de Oro: El Orden Importa

Love Fears  
Fears  
HAte

Para que tus estilos funcionen correctamente, las pseudoclasas deben declararse en un orden específico. Los estados más específicos deben ir al final.

1. `a:link`
2. `a:visited`
3. `a:focus`
4. `a:hover`
5. `a:active`

```
a:link  
a:visited  
a:focus  
a:hover  
a:active
```

Un enlace `:active` también está en estado `:hover`. Sin el orden correcto, el estilo `:hover` anularía al de `:active`.



# El Primer Trazo: Definiendo `link` y `visited`

```
a:link {  
  color: green;  
}  
a:visited {  
  color: olive;  
}  
a:focus { }  
a:hover { }  
a:active { }
```

Dispones de varios navegadores, como [Mozilla Firefox](#), [Google Chrome](#) y [Safari](#).

Ahora, nuestros enlaces no visitados son verdes y los visitados serán de color oliva, diferenciándose de los valores por defecto.



# Añadiendo Reacción: Estilos para `hover` y `focus`

```
/* ... rules for link and visited ... */
```

```
a:focus {  
  background: orange;  
}
```

```
a:hover {  
  background: yellow;  
  text-decoration: underline orange wavy;  
}
```

```
a:active {  
}
```



Estado `:hover`



Estado `:focus`

## ⚠ ¡Alerta de Accesibilidad!

El `outline` de `:focus` es vital para usuarios que navegan con teclado. Si decides quitarlo (`outline: none;`), DEBES proporcionar una alternativa visual igual de clara, como un cambio de `background` o `box-shadow`. Diseña para todos.



# La Confirmación Final: El Estado `active`

```
/* ... other rules ... */  
  
a:focus {  
    background: orange;  
}  
  
a:hover {  
    background: yellow;  
    text-decoration: underline orange wavy;  
}  
  
a:active {  
    background: orange;  
    text-decoration: underline red wavy;  
}
```



Al hacer clic, el enlace cambia drásticamente para confirmar al usuario que su acción ha sido registrada y la navegación está a punto de ocurrir.



# Añadiendo Carácter: Iconos para Contexto Visual

Una práctica común es usar iconos para dar información extra, como indicar que un enlace lleva a un sitio externo.

```
a {  
  /* Define la imagen, su posición y que  
  no se repita */  
  background: url('...') no-repeat right;  
  
  /* Controla el tamaño del icono */  
  background-size: 2rem;  
  
  /* Crea espacio para que el icono no se  
  solape con el texto */  
  padding-right: 2.5rem;  
}
```

Dispones de varios navegadores, como [\[Mozilla Firefox ↗\]](#), [\[Google Chrome ↗\]](#) y [\[Safari ↗\]](#).



# La Transformación Final: De Enlace a Botón de Navegación

Las mismas propiedades que hemos usado se pueden combinar para transformar una simple lista de enlaces en un componente de navegación profesional y reutilizable.

**Inicio**

**Noticias**

**Sobre nosotros**

**Contactar**

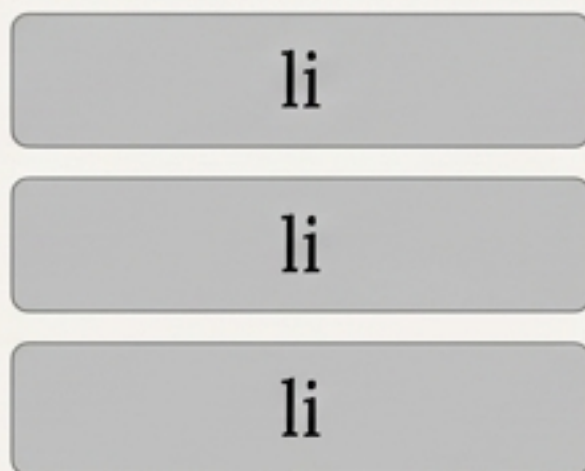
```
<ul class="navbar">
  <li><a href="#">Inicio</a></li>
  <li><a href="#">Noticias</a></li>
  ...
</ul>
```



# Deconstrucción del Componente: La Estructura (ul y li)

## Paso 1: Preparar el Contenedor

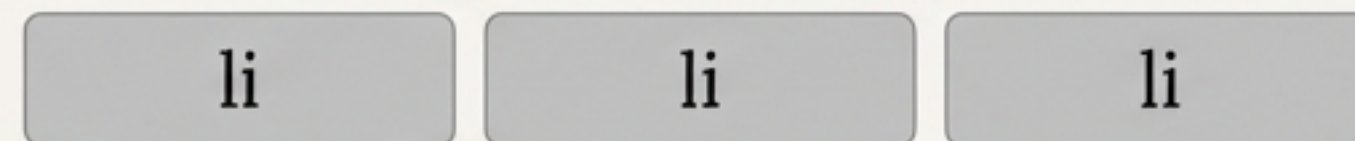
```
.navbar {  
  font-family: sans-serif;  
  text-align: center;  
  background: yellow;  
}
```



Antes

## Paso 2: La Clave de la Disposición Horizontal

```
ul.navbar li {  
  display: inline;  
}
```



Después

Por defecto, los `<li>` son elementos de **bloque** (uno por línea). Con **display: inline**, los forzamos a comportarse como texto, fluyendo uno al lado del otro.



# Deconstrucción del Componente: El Aspecto del Botón (a)

## Paso 3: Eliminando el Estilo de Enlace

```
ul.navbar a {  
  text-decoration: none; /* Adiós al subrayado */  
  display: inline-block;  
  padding: 1rem;  
}
```

## Paso 4: Creando el Espacio del Botón

```
ul.navbar a {  
  text-decoration: none;  
  display: inline-block; /* Permite aplicar padding y dimensiones */  
  padding: 1rem;         /* Añade espacio interior para crear el 'cuerpo' del botón */  
}
```





# Dando Vida a los Botones de Navegación

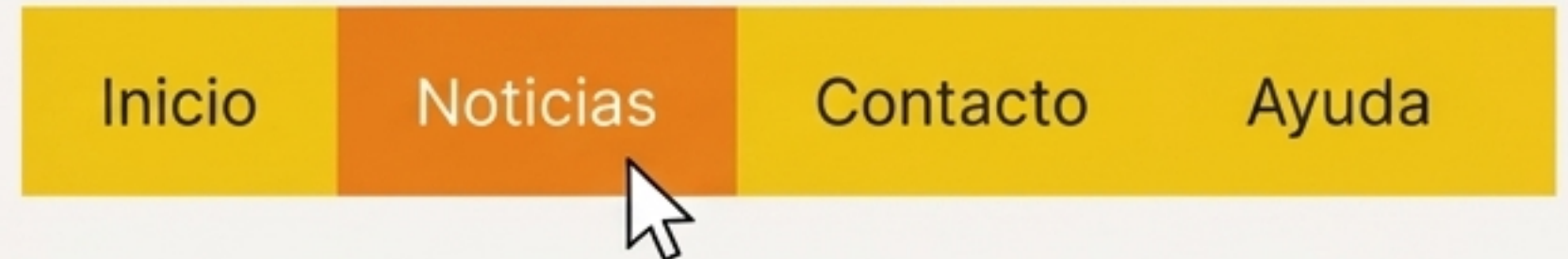
Finalmente, aplicamos los estados `:hover` y `:active` para que nuestros botones sean interactivos y ofrezcan feedback al usuario, igual que un enlace simple.

```
/* Al pasar el ratón por encima */
ul.navbar a:hover {
  background: orange;
}
/* Al hacer clic */
ul.navbar a:active {
  color: white;
  background: red;
}
```

Default State



Hover State



Active State

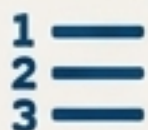




# Resumen de tu Nuevo Dominio sobre Enlaces



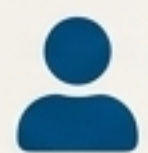
**Los 5 Estados son tu Lienzo:** Domina `:link`, `:visited`, `:focus`, `:hover`, y `:active` para un control total.



**El Orden es Ley:** Recuerda siempre la regla **LoVe Fears HAte** para evitar errores.



**Diseña con Intención:** Respeta las convenciones del usuario (subrayado, color) o rómpelas con una alternativa clara y superior.



**La Accesibilidad no es Opcional:** Un `:focus` visible es crucial. No lo elimines sin un reemplazo.



**De Átomo a Componente:** Las mismas propiedades que estilizan un enlace simple construyen componentes complejos como barras de navegación.



# Ahora te Toca Construir: Tu Desafío

Has adquirido la teoría. Es hora de convertirla en habilidad. Acepta estos desafíos para solidificar tu conocimiento.

## Desafío de Estilos

Crea una página con 5 enlaces. Asigna colores y/o text-decoration únicos para cada uno de los 5 estados, asegurándote de asegurándote de que no se parezcan a los estilos por defecto.

## Desafío de Iconografía

A los enlaces del desafío anterior, añade un icono SVG a la derecha de cada uno SVG la derecha de cada uno usando la propiedad background.

## Desafío de Componentes

Diseña y construye una barra de navegación horizontal completamente personalizada. Experimenta con tipografías, colores, padding y efectos :hover para crear un diseño único.