

# 12 CSS: Agrupación de Selectores

A medida que desarrollas tus hojas de estilo, a menudo te encontrarás queriendo aplicar exactamente el **mismo conjunto de reglas de estilo** (el mismo bloque de declaraciones) a **diferentes elementos o tipos de selectores**. Por ejemplo, quizás quieras que todos tus encabezados ( `h1`, `h2`, `h3` ) compartan el mismo tipo de fuente y color, o que varios tipos de botones (con diferentes clases) tengan el mismo padding y borde.

En lugar de repetir el mismo bloque de declaraciones una y otra vez para cada selector, CSS te permite **agrupar múltiples selectores** y aplicarles un único bloque de declaraciones.

- **Sintaxis:** Se listan los diferentes selectores separados por comas ( , ). El bloque de declaraciones que sigue a la lista se aplicará a *todos* los elementos que coincidan con *cualquiera* de los selectores en la lista.

```
selector1,  
selector2,  
.clase-ejemplo,  
#id-especifico,  
elemento[atributo="valor"] {  
  /* Este bloque de declaraciones se aplica a todos los selectores anteriores */  
  propiedad: valor;  
  otra-propiedad: otro-valor;  
}
```

*Nota: Usar saltos de línea después de cada coma mejora mucho la legibilidad, pero no es técnicamente obligatorio.*

- **Beneficios:**
  - **Reduce la Duplicación (DRY):** Evita repetir las mismas líneas de código CSS, siguiendo el principio "Don't Repeat Yourself".
  - **Código Más Conciso:** Hace que tu hoja de estilos sea más corta y fácil de leer.
  - **Mantenimiento Más Sencillo:** Si necesitas cambiar un estilo que se aplica a varios elementos agrupados, solo tienes que modificarlo en un único lugar.

## Ejemplos Prácticos:

### 1. Estilos comunes para encabezados:

```
/* SIN agrupación */  
h1 {  
  font-family: 'Georgia', serif;  
  color: #333;  
}  
h2 {  
  font-family: 'Georgia', serif;  
  color: #333;  
}
```

```

h3 {
  font-family: 'Georgia', serif;
  color: #333;
}

/* CON agrupación */
h1,
h2,
h3 {
  font-family: 'Georgia', serif;
  color: #333;
}
/* Luego puedes añadir estilos específicos si es necesario */
h1 {
  font-size: 2.5em;
}
h2 {
  font-size: 2em;
}
h3 {
  font-size: 1.5em;
}

```

## 2. Estilos base para diferentes tipos de elementos de interfaz:

```

button,
input[type="submit"],
input[type="reset"],
input[type="button"],
.boton-enlace { /* Una clase para enlaces que parecen botones */
  display: inline-block;
  padding: 8px 16px;
  border: 1px solid #ccc;
  background-color: #f0f0f0;
  cursor: pointer;
  text-align: center;
  text-decoration: none;
  color: black;
}

/* Estilos específicos para un tipo */
input[type="submit"] {
  background-color: dodgerblue;
  color: white;
  border-color: dodgerblue;
}

```

## 3. Agrupando diferentes tipos de selectores (clases, atributos, pseudoclases):

```
.alerta,
.mensaje-exito,
[role="status"], /* Elementos con atributo role="status" */
.notificacion:hover /* Estado hover de .notificacion */ {
    border: 1px solid grey;
    padding: 15px;
    margin-bottom: 10px;
}
```

### ¡Advertencia Importante! - Selectores Inválidos en un Grupo

Si **uno solo** de los selectores en un grupo es **inválido** o no es reconocido por el navegador (por ejemplo, contiene un error de sintaxis, usa una pseudoclase no soportada, o tiene un nombre de etiqueta incorrecto), el navegador **ignorará la regla completa** para *todos* los selectores del grupo.

- **Ejemplo Incorrecto:**

En este caso, ni

`h1` ni `p` recibirán el color rojo debido al error en el segundo selector. Esto puede ser una fuente de confusión al depurar CSS, por lo que es importante asegurarse de que todos los selectores en un grupo sean válidos.

```
h1,
h2..titulo-seccion, /* Error: doble punto inválido */
p {
    color: red; /* ESTA REGLA ENTERA SERÁ IGNORADA */
}
```