

# Guía básica para utilizar variables en CSS

## ¿Qué son las variables en CSS?

Las variables en CSS, también llamadas *custom properties*, permiten almacenar valores reutilizables (colores, tamaños, fuentes, etc.). Esto hace que el código sea más fácil de mantener, consistente y escalable.

---

## Sintaxis de las variables en CSS

### 1. Declaración de variables:

Las variables se declaran utilizando dos guiones (`--`) seguidos de un nombre personalizado, generalmente dentro del selector `:root` para que sean globales:

```
:root {  
  --nombre-de-la-variable: valor;  
}
```

#### Ejemplo práctico:

```
:root {  
  --color-primario: #3498db; /* Declaración de una variable de color */  
  --tamano-texto: 16px;      /* Declaración de una variable de tamaño de texto */  
}
```

### 2. Uso de variables:

Para utilizar una variable, se emplea la función `var()`. Esta función recupera el valor de la variable declarada.

```
selector {  
  propiedad: var(--nombre-de-la-variable);  
}
```

#### Ejemplo práctico:

```
body {  
  background-color: var(--color-primario); /* Uso de la variable */  
}
```

```
font-size: var(--tamano-texto);  
}
```

## Ejemplo cotidiano: Estilizar botones en una página web

Imagina que estás desarrollando un sitio y quieres asegurarte de que todos los botones tengan el mismo estilo:

```
:root {  
  --color-boton: #4CAF50; /* Color de fondo del botón */  
  --color-boton-hover: #45a049; /* Color al hacer hover */  
  --padding-boton: 12px 24px; /* Espaciado interno */  
}  
  
button {  
  background-color: var(--color-boton);  
  color: white;  
  padding: var(--padding-boton);  
  border: none;  
  border-radius: 4px;  
  transition: background-color 0.3s ease;  
}  
  
button:hover {  
  background-color: var(--color-boton-hover);  
}
```

### ✓ Buenas prácticas:

#### 1. Organiza las variables en `:root`:

Declara todas las variables globales en `:root` para facilitar el mantenimiento y la reutilización en todo el proyecto.

```
:root {  
  --color-primario: #ff6f61;  
  --color-secundario: #6b5b95;  
}
```

## 2. Nombres descriptivos:

Usa nombres claros y específicos para las variables.

```
/* Buena práctica */  
--color-fondo-principal: #f4f4f4;  
--font-titulo: 'Roboto', sans-serif;
```

# X Malas prácticas:

## 1. Evitar nombres genéricos:

No uses nombres como `--var1` o `--colorA`.

```
/* Mala práctica */  
--var1: #333;
```

## 2. No declarar variables innecesarias:

Si una variable se usa solo una vez, no es necesario declararla. Evita el exceso de variables que complican el código.

---

## Preguntas frecuentes (FAQ):

### 1. ¿Dónde debo declarar las variables?

Lo mejor es declararlas en `:root` para que sean accesibles globalmente. También puedes declarar variables en un selector específico si solo se usarán allí.

### 2. ¿Qué pasa si una variable no tiene valor?

Si una variable no está definida o tiene un valor incorrecto, el navegador no aplicará la propiedad. Siempre asegúrate de que las variables estén correctamente declaradas.

### 3. ¿Puedo dar un valor por defecto a una variable?

Sí, puedes definir un valor de respaldo dentro de `var()`:

```
color: var(--color-primario, blue); /* Usa 'blue' si --color-primario no está  
definido */
```

### 4. ¿Las variables afectan el rendimiento de mi sitio?

No, las variables de CSS son ligeras y optimizan el código. De hecho, ayudan a

reducir la repetición y facilitan los cambios globales.

## 5. ¿Puedo usar variables para tamaños de pantalla o media queries?

Sí, puedes definir variables para media queries:

```
:root {  
  --ancho-maximo: 768px;  
}  
  
@media (max-width: var(--ancho-maximo)) {  
  body {  
    background-color: #f0f0f0;  
  }  
}
```