

Enunciado del reto

Tu empresa está desarrollando un sitio web y necesitas **aplicar la metodología BEM (Block, Element, Modifier)** para organizar los estilos de una página. En 30 minutos deberás crear una interfaz sencilla con **un encabezado, una tarjeta y un botón**, aplicando modificadores para cambiar estados y estilos sin duplicar código.

Objetivo(s) de la actividad

- **Aplicar** la metodología BEM en la organización de clases CSS.
- **Crear** un proyecto modular con archivos CSS separados por bloques.
- **Utilizar** modificadores BEM para cambiar estilos y estados.
- **Validar** que el código sea claro, escalable y fácil de mantener.

Materiales y preparación previa (checklist)

- Editor de texto.
- Navegador web.
- Archivos: `index.html`, carpeta `css/` con subarchivos por bloque (`_header.css`, `_card.css`, `_button.css`, `main.css`).

Requisitos técnicos y datos

- HTML5 semántico.
- CSS externo modularizado.
- Uso de modificadores (`--primario`, `--secundario`, `--success`, `--error`).

Instrucciones paso a paso con micro-checks

Tiempo total: 30 minutos

1. Estructura HTML (7')

a. Crea `index.html`:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
```

```

    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Proyecto BEM</title>
    <link rel="stylesheet" href="css/main.css">
</head>
<body>
    <header class="header">
        <h1 class="header__title">Mi Proyecto con BEM</h1>
    </header>

    <section class="card">
        <h2 class="card__title">Producto Destacado</h2>
        <p class="card__description">Esta tarjeta sigue la convención
BEM.</p>
        <button class="button button--primario">Comprar</button>
        <button class="button button--secundario">Más info</button>
    </section>

    <div class="alert alert--success">Operación exitosa</div>
    <div class="alert alert--error">Ha ocurrido un error</div>
</body>
</html>

```

Micro-check: abre en navegador y verifica que los elementos aparecen sin estilos.

2. Organización de CSS modular (15')

- a. Crea la carpeta `css/` y dentro:

main.css

```

@import "blocks/_header.css";
@import "blocks/_card.css";
@import "blocks/_button.css";
@import "blocks/_alert.css";
body {
    font-family: Arial, sans-serif;
    margin: 20px;
}

```

_header.css

```
.header {  
  background-color: #0044cc;  
  padding: 15px;  
  text-align: center;  
}  
.header__title {  
  color: white;  
}
```

_card.css

```
.card {  
  border: 1px solid #ddd;  
  border-radius: 8px;  
  padding: 16px;  
  max-width: 400px;  
  margin: 20px auto;  
  text-align: center;  
}  
.card__title {  
  font-size: 1.2rem;  
  margin-bottom: 10px;  
}  
.card__description {  
  margin-bottom: 15px;  
}
```

_button.css

```
.button {  
  padding: 10px 15px;  
  border: none;  
  border-radius: 5px;  
  cursor: pointer;  
  margin: 5px;  
}  
.button--primario {  
  background-color: #28a745;  
  color: white;  
}  
.button--secundario {
```

```
background-color: #6c757d;
color: white;
}
```

_alert.css

```
.alert {
  padding: 10px;
  border-radius: 4px;
  margin: 10px 0;
  color: white;
  text-align: center;
}
.alert--success {
  background-color: #28a745;
}
.alert--error {
  background-color: #dc3545;
}
```

Micro-check: los estilos se aplican modularmente según el bloque.

3. Validación y revisión (8')

a. Verifica que:

- i. Cada bloque tiene su propio archivo CSS.
- ii. Se usan modificadores para variantes (--primario, --error).
- iii. El diseño es consistente y escalable.

Micro-check: cambiar un botón a button--secundario modifica su estilo sin tocar el HTML base.

Criterios de éxito y errores comunes

Éxito:

- Archivos CSS modularizados por bloque.
- Uso correcto de modificadores en BEM.
- Estilos aplicados de manera clara y consistente.
- Página renderizada sin errores.

Errores comunes:

- Colocar todos los estilos en un único archivo desordenado.
- No usar modificadores, duplicando reglas.
- Nombres de clase sin convención clara.

Tiempo total: 30'

Plan de contingencia:

- Si no logras modularizar en varios archivos, coloca todo en `main.css` pero sigue la convención BEM.
- Si falla `@import`, enlaza cada archivo CSS manualmente en el `<head>`.

Extensión opcional (+10'):

- Agregar un modificador `.card--destacada` que cambie el fondo de la tarjeta.
- Implementar `.button--deshabilitado` con estilo gris y cursor bloqueado.

SOLUCIÓN PROPUESTA

index.html

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Proyecto BEM</title>
  <link rel="stylesheet" href="css/main.css">
</head>
<body>
  <header class="header">
    <h1 class="header__title">Mi Proyecto con BEM</h1>
  </header>

  <section class="card">
    <h2 class="card__title">Producto Destacado</h2>
    <p class="card__description">Esta tarjeta sigue la convención
BEM.</p>
    <button class="button button--primario">Comprar</button>
    <button class="button button--secundario">Más info</button>
  </section>

  <div class="alert alert--success">Operación exitosa</div>
  <div class="alert alert--error">Ha ocurrido un error</div>
</body>
</html>
```

css/main.css

```
@import "blocks/_header.css";
@import "blocks/_card.css";
@import "blocks/_button.css";
@import "blocks/_alert.css";

body {
  font-family: Arial, sans-serif;
```

```
    margin: 20px;
}
```

css/blocks/_header.css

```
.header {
  background-color: #0044cc;
  padding: 15px;
  text-align: center;
}
.header__title {
  color: white;
}
```

css/blocks/_card.css

```
.card {
  border: 1px solid #ddd;
  border-radius: 8px;
  padding: 16px;
  max-width: 400px;
  margin: 20px auto;
  text-align: center;
}
.card__title {
  font-size: 1.2rem;
  margin-bottom: 10px;
}
.card__description {
  margin-bottom: 15px;
}
```

css/blocks/_button.css

```
.button {
  padding: 10px 15px;
  border: none;
  border-radius: 5px;
  cursor: pointer;
  margin: 5px;
}
```

```
.button--primario {  
  background-color: #28a745;  
  color: white;  
}  
.button--secundario {  
  background-color: #6c757d;  
  color: white;  
}
```

css/blocks/_alert.css

```
.alert {  
  padding: 10px;  
  border-radius: 4px;  
  margin: 10px 0;  
  color: white;  
  text-align: center;  
}  
.alert--success {  
  background-color: #28a745;  
}  
.alert--error {  
  background-color: #dc3545;  
}
```