



# BEM

## CSS NAMING CONVENTION

WHAT IS IT ?

WHY USING THAT ??

HOW TO USE IT ???

A guide for using BEM by Parsa Shafagh Motlagh



# ● WHAT IS IT ?

## CSS NAMING CONVENTION

Organizing code in small project may not be important, but when it comes to larger and more complex project, it is important to organize code to achieve efficiency.



# ● WHY WE NEED THAT?

- The idea behind BEM is for developers to write HTML and CSS in a way that avoids inheritance issues .As we all know, CSS in bigger projects can become really messy really quickly when not managed and structured correctly. BEM helps with this. So we have an element, the block component with the class .



# ● WHY WE NEED THAT?

- It helps you speed up the development process while keeping things organized and easy to understand for the whole team



# ● WHY WE NEED THAT?

## EASY TO LEARN:

- To use BEM, you only need to employ BEM's naming convention.

## MODULAR:

- Independent blocks and CSS selectors make your code reusable and modular.

## FLEXIBLE:

- Using BEM, methodologies and tools can be recomposed and configured the way you like.



BEM is created by  
Yandex company





- HOW TO  
USE IT?



BEM

BLOCK

ELEMENT

MODIFIER



# BEM

BLOCK



# BEM

## BLOCK

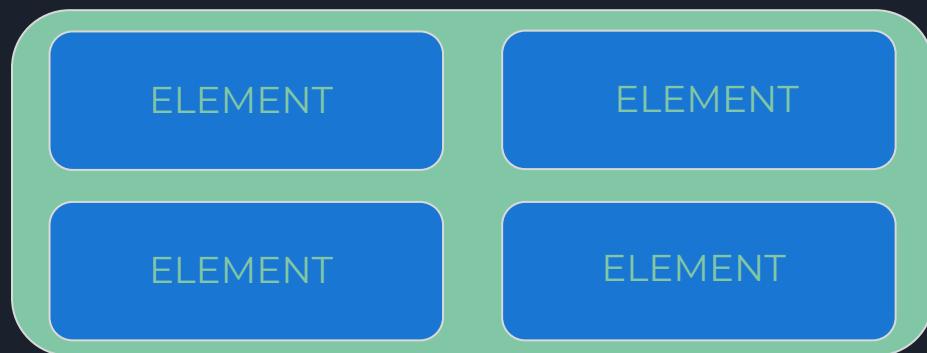
Standalone entity that is meaningful on its own.

Examples:

header, container, menu, checkbox, input

# BEM

BLOCK



# BEM

## ELEMENT

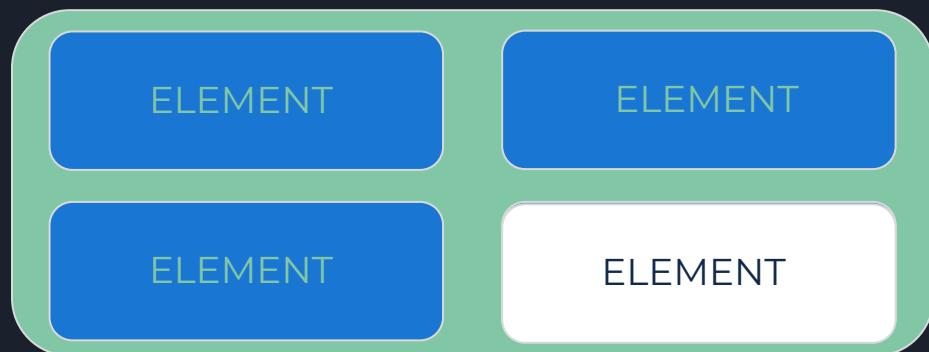
A part of a block that has no standalone meaning and is semantically tied to its block.

### Examples

menu item, list item, checkbox caption,  
header title

# BEM

BLOCK + MODIFIER



# BEM

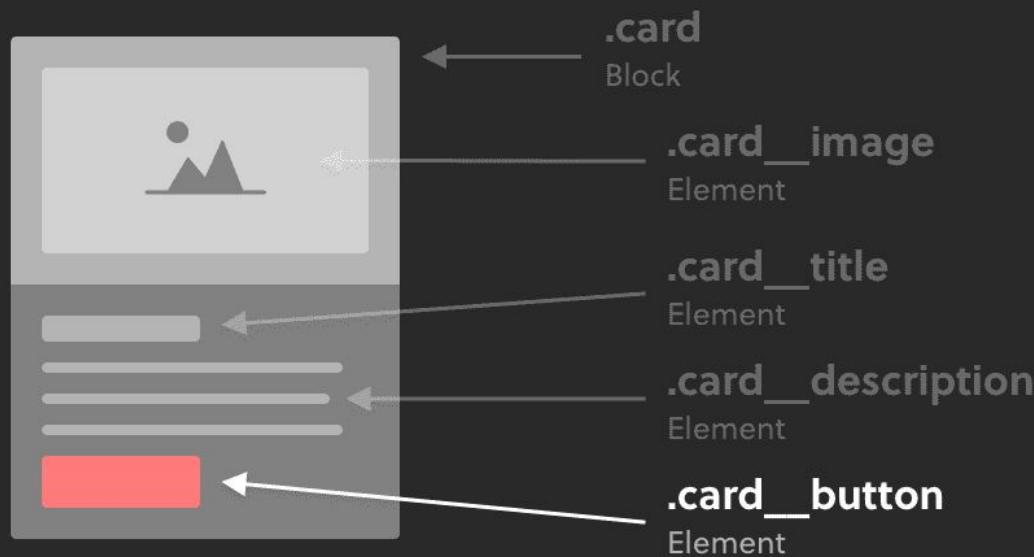
## MODIFIER

A flag on a block or element. Use them to change appearance or behavior.

### Examples

disabled, highlighted, checked, fixed, color  
yellow

# Practical example



# Practical example

The diagram illustrates a header structure with the following components and their corresponding CSS classes:

- menu block**: A horizontal bar with tabs: Tab1, Tab2, Tab3, Tab4.
- logo block**: Contains the **LOGO**.
- search block**: Contains a search input field and a **Search** button.
- auth block**: Contains a **Login** button, a password input field with placeholder **\*\*\*\*\***, and a **Sign in** button.

A large bracket at the bottom is labeled **head block**.

```
<header class="header">
  <img class="header__logo">
  <form class="header__search-form">
    <input class="header__search-form__input" type="input">
    <button class="header__search-form__button" type="button">
  </form>
  <div class="header__lang-switcher"></div>
</header>
```

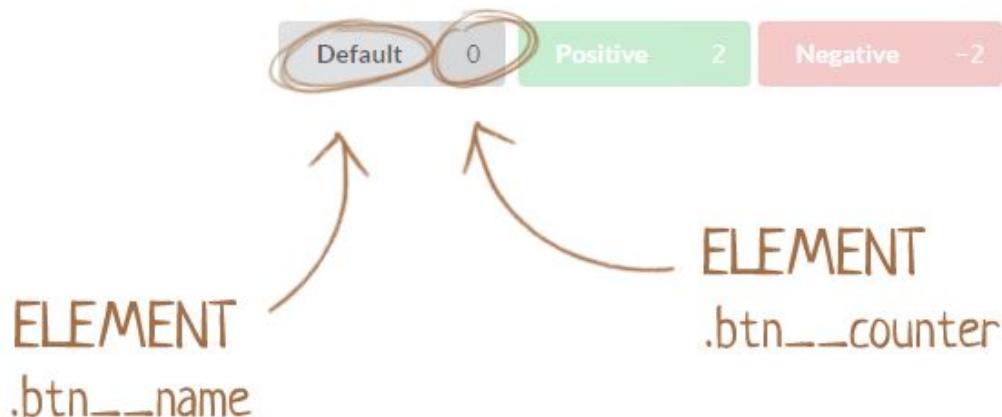
**toptal**

## Another example

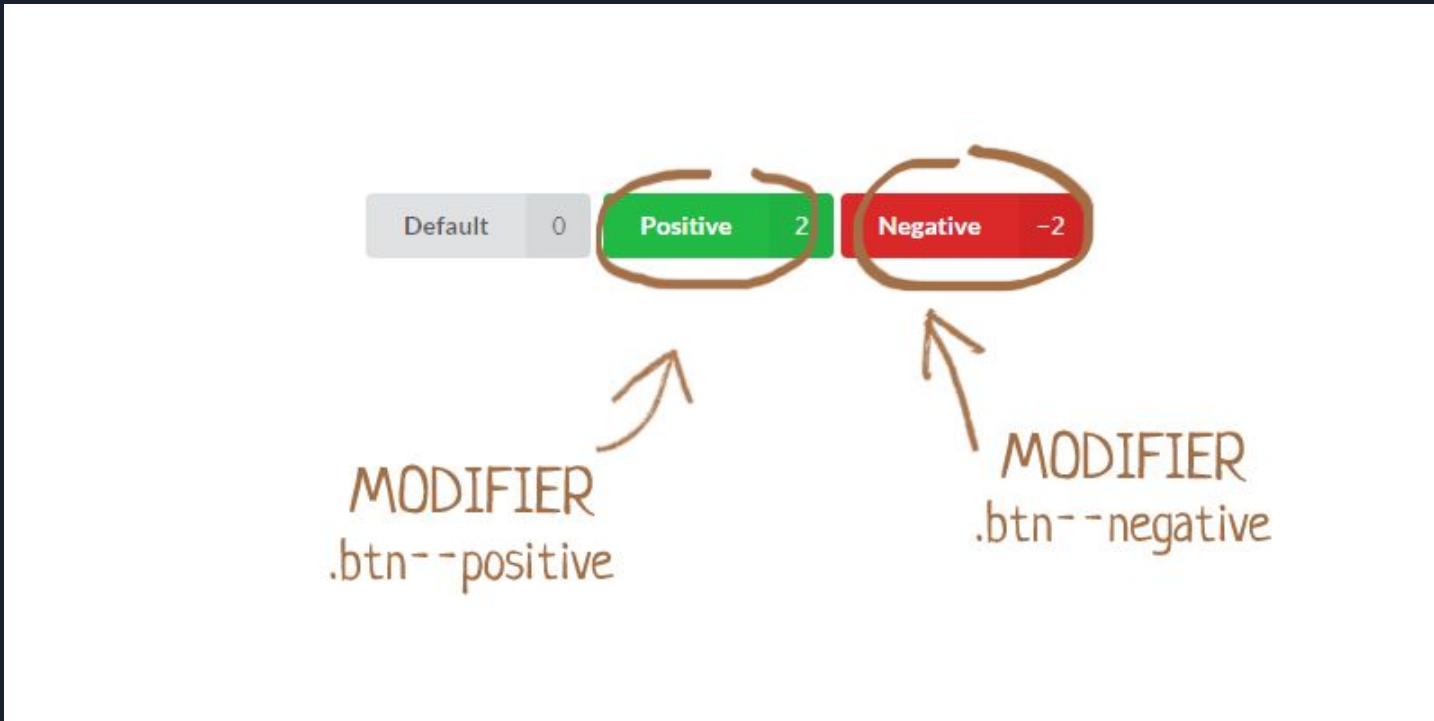


BLOCK  
.btn

## Another example



## Another example



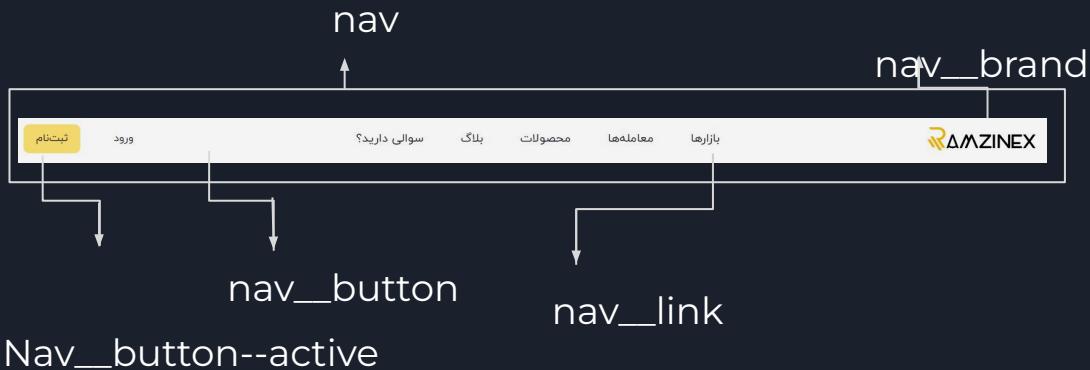
# Practical example



# Practical example

CSS code:

```
.nav{  
    width:100%;  
}  
.nav__link{  
    /*code style...*/  
}  
.nav_brand{  
    /*code style...*/  
}  
nav_button{  
    /*code style...*/  
}  
nav_button--active{  
    background-color: #f1eb69;  
}
```



# Practical example

SCSS code:

```
$background-color_1: #f1eb69;

.nav {
  width: 100%;
  & link {
    /*code style...*/
  }
  & brand {
    /*code style...*/
  }
  & button {
    /*code style...*/
    &--active {
      background-color: $background-color_1;
    }
  }
}
```



```
<div class="card">
  <h3 class="card__title">
    How do I use BEM CSS?
  </h3>
  <div class="card__excerpt">
    BEM CSS stands for Block Element Modifier, which is a
    CSS methodology to help you write maintainable CSS.
  </div>

  <a class="btn btn--secondary" href="/path/to/blog/post">
    Read More
  </a>
</div>
```

```
.btn {
  padding: 10px 20px;
  font-size: 1rem;
}

.btn--primary {
  background: blue;
  color: white;
}

.btn--disabled {
  background: grey;
  color: white;
}
```



No dependency between choosing names



Some are against this  
They say that the names of our classes will be  
long.  
That's right..  
**BUT..**

- 
- ❖ In HTML file it is better because of the clarity
  - ❖ In CSS file will be amazing if we use css preprocessor



In SCSS we have Nesting.. :)



Thanks for watching!

-  [parsa.shafagh.1382@gmail.com](mailto:parsa.shafagh.1382@gmail.com)
-  <https://github.com/PA-MO>

