**BEM**

**BEM** là một quy chuẩn đặt tên cho các class trong dự án. Nó giúp code trở lên dễ hiểu hơn và tạo ra sự nhất quán giữa việc đặt tên các class, giảm được conflict khi làm việc nhóm.

BEM được viết tắt bởi:

B: Block

E: Element

M: Modifier

**Cách đặt tên class theo chuẩn BEM**

Block\_\_Element—Modifier

Hoặc Block—Modifier

Ví dụ:

Room (Block)

Room\_\_Bed—Big (BEM)

Room\_\_Bed—Small (BEM

Room—Big (BM)

Room—Small (BM)

Etc

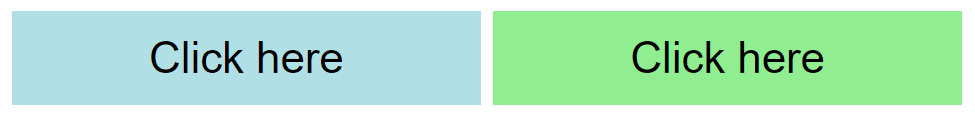
Ví dụ đoạn code sau

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <title></title>  </head>  <body>  <button class="btn btn--primary">Click here</button>  <button class="btn btn--secondary">Click here</button>  </body>  </html>  <style type="text/css">  .btn{  width: 150px;  height: 30px;  border: none;  outline: none;  font-family: sans-serif;  font-size: 14px;  }  .btn--primary{  background: powderblue;  }  .btn--secondary{  background: lightgreen;  }  <style> |

Với ví dụ trên thì ta thấy button là thành phần được sử dụng nhiều lần, vậy ta có thể cài đặt cho nó những thuộc tính chung và sử dụng ở nhiều nơi.

Đối với button cần có màu sắc primary hay có màu secondary (thường sẽ phụ thuộc vào công dụng của nó) thì ta sẽ tạo hai class tương tự là btn—primary vầ btn—secondary. Trong phần css hai class này ta chỉ cần cài đặt màu nền cho chúng.

Dưới đây là kết quả



Điều khó nhất trong BEM là cách đặt tên class =)))

BEM có nhiều ứng dụng khác nhau:

* Làm animation
* Tái sử dụng class mà không bị conflict với nhau

Nhược điểm của BEM là tên class dài, một thẻ có thể chứa nhiều class nên đối với các dự án, trang web kích thước nhỏ sẽ không cần thiết dùng tới. BEM phù hợp để làm các dự án lớn, những dự án cần có cấu trúc code chặt chẽ và dễ hiểu