

Desenvolvimento Front-End I - Stackx

Aluno: Leonardo Santos

Tarefa - Metodologias dentro do CSS

- O que devo fazer?
 - Pesquise e descreva sobre metodologias como B.E.M (Block Element Modifier) ou S.M.A.CSS (Scalable and Modular Architecture for CSS) e escolha um deles para usar, organizar e nomear de forma consistente seus estilos dentro do seu projeto final.
- **BEM** - Uma metodologia CSS ou convenção de nomenclatura
 - A metodologia nasceu na Rússia com a finalidade de organizar e padronizar a escrita do código.
 - BEM é um acrônimo para **Bloco**, **Elemento** e **Modificador**, sendo estes os 3 pilares do padrão BEM, tendo cada um de seu significado, função e regras próprias, permitindo descrever a realidade do código pensando nas entidades do programa.
 - Regras de Bloco, Elemento e Modificador:
 - **Bloco:**
 - É uma entidade independente, um “bloco de construção” de uma aplicação; uma abstração mais geral, com seu próprio significado. É a parte mais flexível na nomenclatura, significando que se pode atribuir o nome à “entidade” da maneira que quiser.
 - **.[Bloco]**
 - **.c-menu**
 - **Elemento:**
 - É um elemento-filho de um bloco, este elemento não tem função/significado independente e está semanticamente vinculado ao seu respectivo Bloco.
 - Nome do Bloco + 2 underlines + nome do Elemento:
 - **.[Bloco]__[Elemento]**
 - **.c-menu__item**

- **Modificador:**
 - Serve como uma variante usada para alterar a aparência, variar uma propriedade ou atribuir um estado de um Bloco ou Elemento.
 - Nome de um Bloco ou Modificador + 2 hífens + o nome do Modificador:
 - **.[Bloco|Elemento]--[Modificador]**
 - **.c-menu--dark**
- Boas práticas para escrever **BEM**:
 - Não replicar a hierarquia HTML no nome das classes.
 - **.c-menu__item__link** → incorreto
 - **.c-menu__link** → correto
 - Usar nomes compostos:
 - **.c-card__wrapper__info__image** → evite
 - **.c-card__info-image** → recomendado
 - Utilizar nomenclatura específica (**is**) para alteração de estado:
 - **.c-card is-active**
- Vantagens de uso do **BEM**:
 - Código desacoplado;
 - Reúso automático de código;
 - Menos repetições;
 - Rápida identificação de estruturas HTML através do CSS (e vice-versa);
 - Independência de classes;
 - Seletores menores e mais performáticos;
 - CSS mais manutenível.
- Documentação:
 - [BEM](#)
- Material de apoio pesquisado:
 - [Quick start / Methodology / BEM](#)
 - [BEM: guia definitivo do padrão CSS mais famoso | desenvolvimento para web](#)