


CSS, Pré Processadores, CSS-in-Js

RODRIGO GONÇALVES

 [linkedin.com/in/rodrigo-goncalves-santana](https://www.linkedin.com/in/rodrigo-goncalves-santana)



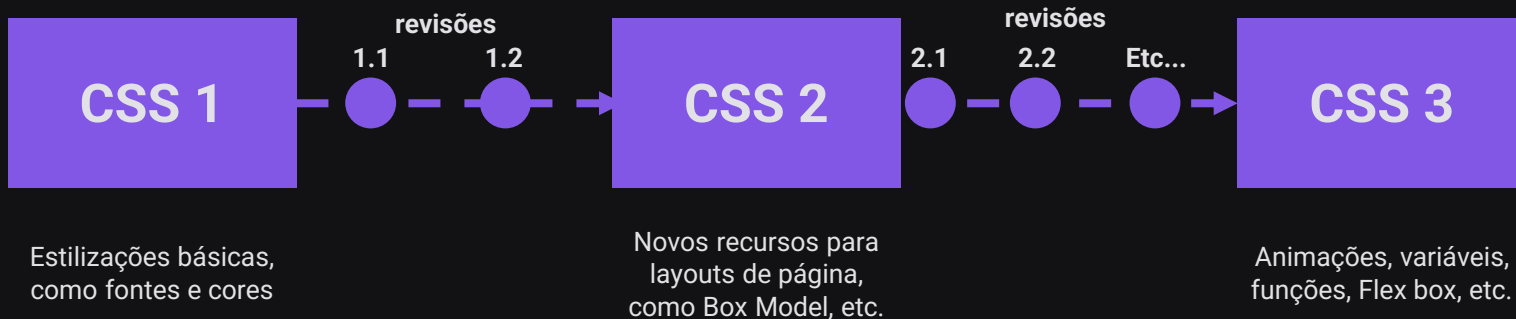
O CSS

O CSS evoluiu muito desde o seu lançamento e continua evoluindo.



O CSS

O CSS evoluiu muito desde o seu lançamento e continua evoluindo.

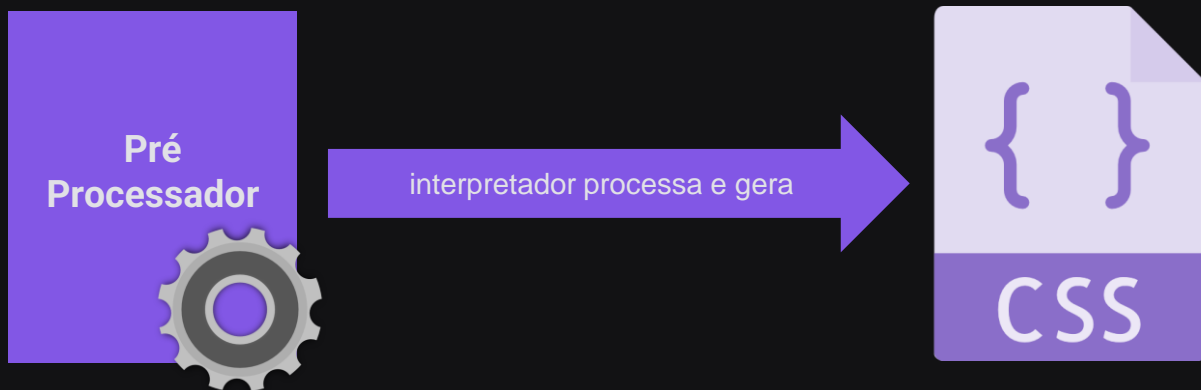


a evolução continua...



Pré Processadores

Um CSS pré processado quer dizer que você não vai utilizar esse arquivo em produção. Na verdade, ele será processado para utilização final no CSS da sua aplicação.



Vantagens de utilizar um Pré Processadores

Facilita a manutenção do código, ajuda a escrever um código mais limpo e organizado utilizando vários arquivos para separar os códigos e realizando a importação quando for necessário tornando os códigos CSS mais flexíveis e reutilizáveis.

Possui recursos, como por exemplo, de **minificar** arquivos.



CSS Minificado

```
1  #box {  
2    background-color: #ffffff;  
3    font-family: consolas,Menlo,"courier new",monospace;  
4    font-size:15px;  
5    height:100%;  
6    width:100%;  
7    padding:8px;  
8    resize: none;  
9    border:none;  
10   line-height:normal;  
11 }
```



```
1  #box{background-color:#fff;font-family:consolas,Menlo,"courier new",monospace;font-size:15px;  
    height:100%;width:100%;padding:8px;resize:none;border:none;line-height:normal}
```



Mas, o CSS continua tendo desafios.
Por exemplo, lidar com **especificidade e colisão de estilos**.



Colisão de estilos

Acontece quando você estiliza uma tag sem o uso de classes ou cria classes com os mesmos nomes para estilizar seus elementos.

Com isso, um estilo pode colidir sobrescrevendo o outro.



Fácil, “*é só aninhar*”. Certo?

```
.minha-classe > section > header > div > h1 {}
```



Elementos aninhados

Muito elementos aninhados impacta na performance, porquê elementos aninhados são mais lentos para carregar.

E o CSS sozinho não possui mecanismos para evitar que isso aconteça.



CSS-in-Js

CSS-in-JS fazem muito sentido, porque aproveitam métodos atuais de componentização em JavaScript para criar componentes **performáticos** a **prova de colisão**.

E tudo isso é extremamente automatizado.



Styled Components

Styled Components foi desenvolvido para melhorar a maneira que lidamos com CSS nos componentes de aplicações React.



Vantagens Styled Components

Sem colisão de classes: fornece nomes de classes a prova de colisão.

Desempenho: Como funciona diretamente nos componentes, ele consegue analisar facilmente quais códigos serão ou não usados, inclusive os que são adicionados após interação do usuário. O que também ajuda a diminuir o código final.

Estilo dinâmico: permite adaptar os estilos baseando nas props recebidas, é possível criar estilos dinâmicos de forma fácil e intuitiva.


Manutenção sem dor: Tudo que você precisa vai estar no próprio contexto do componente, facilitando encontrar tudo o que você precisa para o desenvolvimento.

Boas práticas: Você escreve seu CSS no melhor padrão do mercado e pronto, os componentes cuidam para fornecer suporte para browsers antigos.



Borá Codar

RODRIGO GONÇALVES

 [linkedin.com/in/rodrigo-goncalves-santana](https://www.linkedin.com/in/rodrigo-goncalves-santana)

