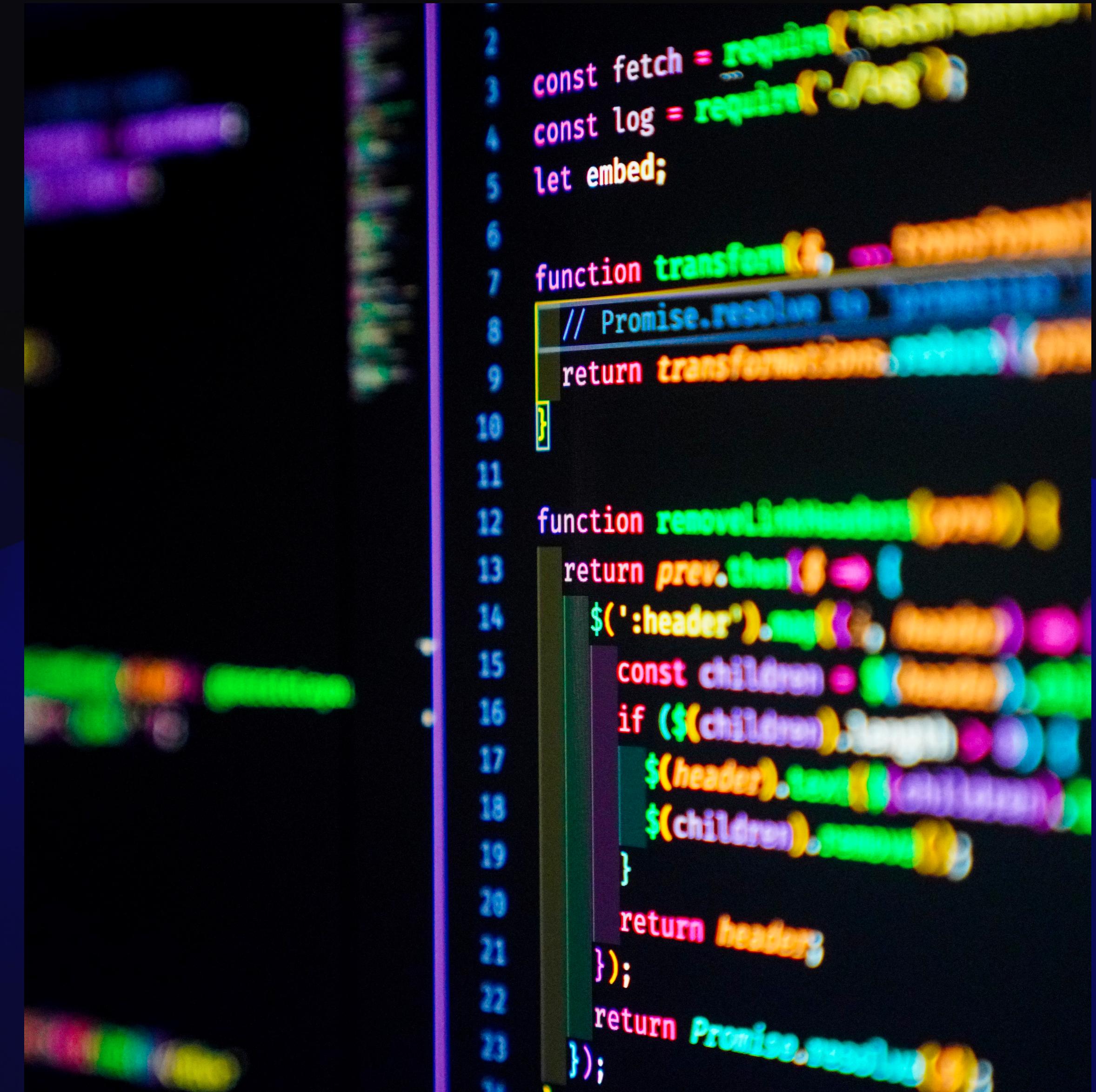


Responsividade

Media Queries



```
2 const fetch = require('node-fetch');
3 const log = require('loglevel');
4 let embed;
5
6
7 function transform($, prev) {
8   // Promise.resolve to prevent race condition
9   return transformPromise($.promise());
10 }
11
12 function removeChildren($node) {
13   return prev.then(() => {
14     const children = $node.children();
15     if (!children.length) return $node;
16     $(`<${children[0].nodeName}>`).append(children);
17     $node.replaceWith(`<${children[0].nodeName}>`);
18     $node.append(children);
19   });
20 }
21
22 return Promise.resolve();
23 }
```

@media

O bloco @media no CSS é utilizado para definir propriedades CSS em tamanhos específicos de tela.

Usando parâmetros como largura mínima/máxima ou orientação é possível fazer com que a estrutura da página se adapte de acordo com o dispositivo onde está sendo visualizada.

min-width = largura minima

max-width = largura máxima

orientation = orientação do dispositivo, pode ser portrait ou landscape

Referência:

https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS_media_queries/Using_media_queries

@media

É utilizada para definir propriedades CSS em tamanhos específicos de tela

```
@media(max-width: 768px) {  
    //seu css para telas de até 768px  
}
```

@media



```
● ● ●  
  
 @media(orientation: landscape) {  
     //seu css para telas onde a largura é maior que a altura  
 }
```

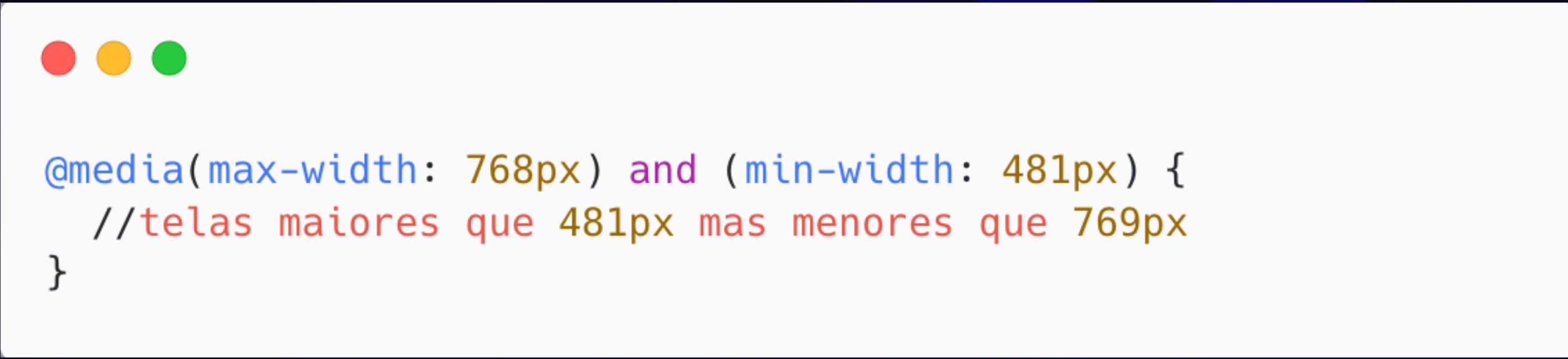
@media



```
@media(min-width: 1024px) {  
  //seu css para telas com mais de 1024px  
}
```

@media

É possível combinar instruções usando “and”.



A screenshot of a browser window showing a snippet of CSS code. The window has three colored buttons (red, yellow, green) at the top left. The code is:

```
@media(max-width: 768px) and (min-width: 481px) {  
    //telas maiores que 481px mas menores que 769px  
}
```