



*Sass*

SASS é uma linguagem de estilos que compila em CSS.

O que é possível com o SASS ?

- Uso de variáveis
- Aninhamento de regras
- Uso de mixins
- Funções
- Suporte a prefixo de vendors nativo
- Totalmente compatível com a sintaxe CSS
- É compatível com muitas linguagens como Ruby, Python, Dart e Node.

## Pré processadores em CSS

Vamos utilizar Scout-App

<https://scout-app.io/>



## Variáveis SCSS

```
$funcao : blue;  
  
$fonte : "Trebuchet MS";  
  
$divisao : 1.5;  
  
$fonte-tamanho: (40px + 20px) / $divisao;
```



```
body {  
  background: $funcao;  
}  
h3 {  
  font-size: $fonte-tamanho;  
}
```

```
body {  
  background: blue; }  
  
h3 {  
  font-size: 40px; }
```

## Aninhamento de regras (nesting)

```
<body>
<h3>CSS PRO</h3>
<p class="destaque">
  Usando nesting no SASS
</p>
</body>
```

```
$fundo : #002244;
$cor : #FFFFFF;
$fonte : "Trebuchet MS";

body {
  font-family: $fonte;
  background: $fundo;
  color: $cor;

  h3 {
    font-size: $fonte-tamanho;
  }

  p {
    &.destaque {
      font-style: italic;
    }
  }
}
```



```
body {
  font-family: "Trebuchet MS";
  background: #002244;
  color: #FFFFFF; }
body h3 {
  font-size: 24px; }
body p.destaque {
  font-style: italic; }
```

## Interpolação

```
$bg-imagem: 'css-3.png';
```

```
body {  
  div {  
    width: 200px;  
    height: 200px;  
  
    border: 1px solid $cor-borda;  
    background: url(../img/#{$bg-imagem});  
  }  
}
```

Permite a inclusão dinâmica de valores de variáveis e argumentos dentro de strings do próprio SASS.

## Mixins com argumentos

```
@mixin borda-redonda($radius) {  
  border: 1px solid red;  
  border-radius: $radius;  
}
```

```
body {  
  div {  
    width: 200px;  
    height: 200px;
```

```
    @include borda-redonda(12px);  
  }  
}
```

É possível passar inúmeros argumentos para customizar o mixin.

Também é possível definir valores padrão para o argumento caso não seja enviado.

```
@mixin borda-redonda($radius: 10px) {  
  border: 1px solid red;  
  border-radius: $radius;  
}
```

## Mixins com argumentos

```
@mixin borda-redonda($background, $radius) {  
  border: 1px solid red;  
  border-radius: 10px;  
  background: $background;  
}  
  
body {  
  div {  
    width: 200px;  
    height: 200px;  
  
    @include borda-redonda($radius: 20px , $background: red);  
  }  
}
```

Em casos que não é possível garantir que os argumentos serão enviados na ordem que foram esperados, podemos enviá-los como “chave:valor”.



## Mixins com interpolação

```
@mixin borda-redonda($propriedade, $valor) {  
  border: 1px solid red;  
  #{$propriedade} : $valor;  
}  
  
body {  
  div {  
    width: 200px;  
    height: 200px;  
  
    @include borda-redonda(border-radius, 20px);  
  }  
}
```

A interpolação também pode ser utilizada dentro de mixins.

## Mixing Content

```
@mixin borda-redonda($background, $radius) {  
  border-radius: 10px;  
  background: $background;  
  @content;  
}  
  
body {  
  div {  
    width: 200px;  
    height: 200px;  
  
    @include borda-redonda($radius: 20px , $background: red) {  
      border: 4px solid greenyellow;  
    };  
  }  
}
```

É possível aceitar blocos adicionais de regras dentro de mixins utilizando o “@content”

## Media Queries

A @media gera um bloco de código que só será ativado caso a condição entre parênteses seja verdadeira.

- @media (max-width: 500px)  
Abaixo de 500px
- @media (min-width: 600px)  
Acima de 600px
- @media (min-width: 800px) and (max-width: 900px)  
Entre 800px e 900px.