

Vídeo 1

Transcrição

Pensando em performance e boas práticas de CSS, estava observando que todos os botões do site da Apeperia tem o mesmo `box-shadow`, possuindo os mesmos valores. Por exemplo, o `.destaque button` do `estilos.less`:

```
.destaque button {
  margin-top: 1em;
  background: #66DDDD;
  border: 0;
  padding: .6em;
  font-size: 1.2em;
  -webkit-border-radius: 0.3em;
  border-radius: 0.3em;
  box-shadow: 0 2px 6.65px 0.35px rgba(0, 0, 0, 0.3);
  font-weight: bold;
}
```

Pensando em melhorar a performance, podemos juntar os seletores que contém o `box-shadow` em um único e depois adicionar a mesma propriedade neste elemento:

```
.destaque button, .plano button, . contato button {
  box-shadow: 0 2px 3px #000
}
```

Mas não podemos fazer isso no CSS, caso contrário o novo seletor será perdido quando for compilado pelo Grunt. Nós iremos usar outra ferramenta que o LESS nos proporciona: o **extend**.

Vamos abrir o arquivo `destaque.less` e buscar o `.destaque button`:

```
.destaque button {
  margin-top: 1em;
  background: @cor-padrao;
  border: 0;
  padding: .6em;
  font-size: 1.2em;
  .borda-arredondada(2em);
  .sombra-padrao;
  font-weight: bold;
}
```

Quando chamamos o mixin `.sombra-padrao`, ele irá repetir a regra `box-shadow` em vários seletores. Agora, para evitar adicionar vários seletores separados apenas por vírgulas, nós usaremos o `extend`.

Para transformarmos o mixin em `extend`, adicionaremos o `&:extend()`, sendo que dentro dos parênteses colocaremos o `.sombra-padrao`:

```
&:extend(.som-padrao);
```

O `.destaque button` do `destaque.less` ficou assim:

```
.destaque button {  
  margin-top: 1em;  
  background: @cor-padrao;  
  border: 0;  
  padding: .6em;  
  font-size: 1.2em;  
  .borda-arredondada(2em);  
  &:extend(.sombra-padrao);  
  font-weight: bold;  
}
```

No `estilos.css`, ele colocará o `.sombra-padrao` e o `.destaque button` juntos:

```
.sombra-padrao,  
.destaque button {  
  box-shadow: 0 2px 6.65px 0.35px rgba(0, 0, 0, 0.3);  
}
```

Observe que o `box-shadow` não aparecerá mais no `.destaque button`:

```
.destaque button {  
  margin-top: 1em;  
  background: #66DDDD;  
  border: 0;  
  padding: .6em;  
  font-size: 1.2em;  
  -webkit-border-radius: 0.3em;  
  -o-border-radius: 2em;  
  border-radius: 0.3em;  
  font-weight: bold;  
}
```

Em seguida, usaremos o `extend` no `.plano button` do `planos.less`:

```
.plano button {  
  background: @cor-padrao;  
  border: 0;  
  color: @cor-branca;  
  width: 210px;  
  height: 38px;  
  font-size: 1.2em;  
  .borda-arredonda();  
  &:extend(.sombra-padrao);  
  margin-left: 2em;  
}
```

No `estilos.css` ficará assim:

```
.sombra-padrao,  
.destaque button,  
.plano button {  
  box-shadow: 0 2px 6.65px 0.35px rgba(0, 0, 0, 0.3);  
}
```

Estamos conseguindo repetir menos código no seu CSS.

Agora, usaremos o `extend` no `contato.less`.

```
.contato button {  
  margin: 0 auto;  
  background: @cor-padrao;  
  border: 0;  
  color: @cor-branca;  
  width: 210px;  
  height: 38px;  
  font-size: 1.2em;  
  .borda-arredondada();  
  &:extend(.sombra-padrao);  
}
```

No `estilos.css`, veremos o seguinte código:

```
.sombra-padrao,  
.destaque button,  
.plano button,  
.contato button {  
  box-shadow: 0 2px 6.65px 0.35px rgba(0, 0, 0, 0.3);  
}
```

Um ponto negativo sobre o `extend` é que ele não nos permite passar valores como parâmetro quando o chamamos. No caso, precisamos manter os valores usado no `box-shadow`. Se for necessário passar valores diferentes, teremos que criar um `mixin`. Nós iremos nos aprofundar sobre o assunto mais adiante.

