



Ah se meu código funcionasse no Safari...

O Safari, até sua versão 9 suporta muito pouco o ES6 (nem com reza forte vai funcionar nosso código sem o uso de transpiler!). Contudo uma nova versão está no forno e essa promete ter o maior suporte de todos os navegadores. Eu realizei um teste com o *Safari Technology Preview* (versão teste da nova versão que sairá) e o código funcionou perfeitamente, mas há um detalhe interessante que vale a pena colocar aqui até para engrandecer a discussão.

OBS.: você precisa ter feito o fix do Firefox (ele não suporta o input date) e ainda estar usando a versão teste do próximo navegador da Apple. Tudo bem?

Durante o treinamento, aprendemos a usar `let` ao invés de `var` justamente por `let` criar escopo de bloco. Contudo, vejamos a tag script que inicializa nossa controller:

```
<!-- aluraframe/client/ndex.html -->
<!-- código anterior omitido -->
<script>
    let negociacaoController = new NegociacaoController();
</script>
<!-- código posterior omitido -->
```

No Chrome e no Firefox não há problema, mas na tecnologia Preview do Safari, apesar de suportar a sintaxe `let` apenas aqui o código não funciona. o Safari está errado? Vamos dizer que dependendo do ponto de vista não.

A sintaxe `let` criado foi para permitir um escopo de bloco de variáveis. Veja que o código anterior que declara nosso controller no escopo global não está dentro de nenhum bloco! O Safari entende que é uma declaração inválida, pois `let` deve ser usada apenas dentro de blocos. Para mim, faz sentido, apesar de o Chrome e o Firefox não serem tão chatos nesse ponto.

Sendo assim, para que nosso código funcione no Safari precisamos alterar simplesmente, apenas nesse ponto a declaração da variável para `var` :

```
<!-- aluraframe/client/index.html -->
<!-- código anterior omitido -->
<script>
    var negociacaoController = new NegociacaoController();
</script>
<!-- código posterior omitido -->
```

Então, podemos usar a seguinte regra: usamos `var` quando queremos que a variável tenha escopo global ou de função, e usarmos `let` quando queremos que a variável tenha sempre escopo de bloco. Em nosso caso, como nosso controller está no escopo global, faz todo sentido usarmos `var` .