09

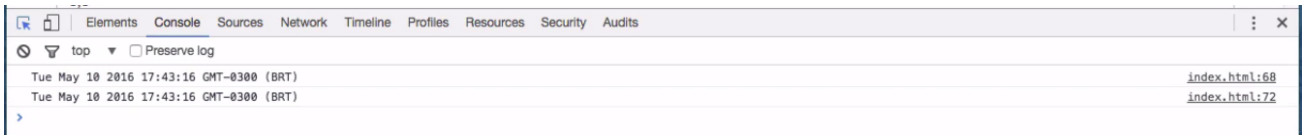
Substituindo var por let

Transcrição

Temos o estado do nosso modelo de negociação e podemos continuar com o nosso projeto. Mas antes de continuarmos, queremos implantar um novo hábito. Agora, que estamos utilizando ES6, em vez de usarmos o `var` para fazer a declaração de uma variável, usaremos o `let` .

```
<script>

    let hoje = new Date();

    let n1 = new Negociacao(hoje, 5, 700);

    console.log(n1.data);

    hoje.setDate(11);

    console.log(n1.data);
</script>
```

Se executarmos o código, tudo continuará funcionando normalmente.



Mas o que ganhamos com a mudança de `var` para `let` ? Veremos um exemplo das vantagens de usarmos esta forma de declarar variável. Se formos declarar um `for` que vamos exibir de `1` a `100` escreveríamos o laço da seguinte forma:

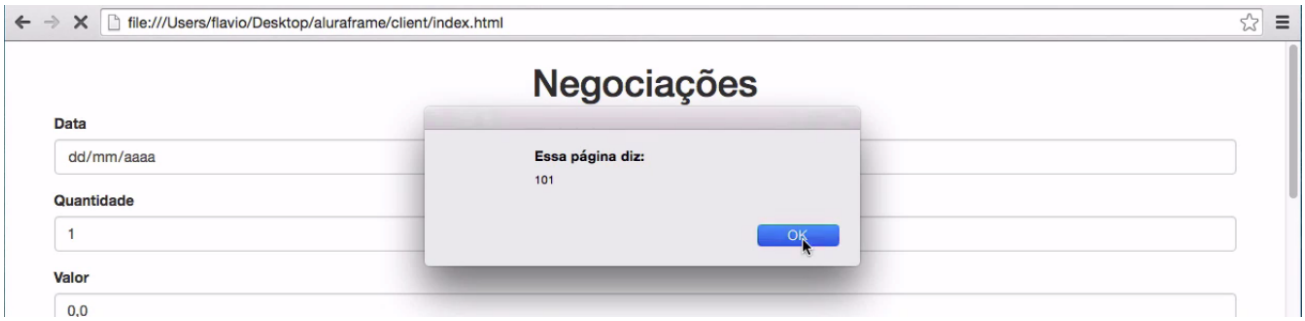
```
<script>

    for(var i = 1; i<= 100; i++) {
        console.log(i);
    }
</script>
```

Ao executarmos o código, ele imprimirá corretamente os números de `1` a `100` no Console.

Porém, existe algo que programadores fora da linguagem JS acham estranho... Vamos adicionar um `alert(i)` :

```
<script>

    for(var i = 1; i<= 100; i++) {
        console.log(i);
    }

    alert(i);
</script>
```

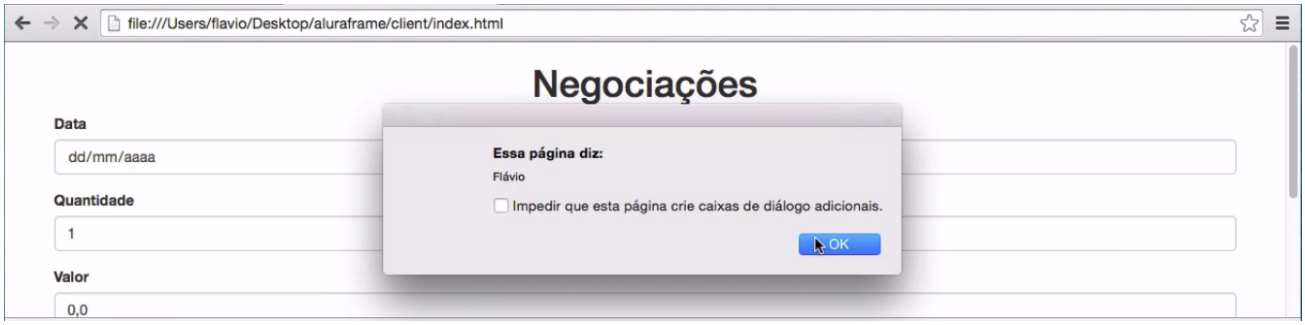
O que vai ser exibido no navegador?



O `alert` exibiu o valor `101`. Por quê? Quando se trabalha com linguagens como Java, C# e outras, as declaração de variáveis possuem escopo de bloco. Na prática, ao utilizarmos estas outras linguagens, jamais poderíamos acessar a variável `i`, como fizemos com o `alert`. Se adicionássemos uma variável chamada `nome` e depois, acrescentássemos um novo `alert`, o código ficaria assim:

```
<script>

    for(var i = 1; i<= 100; i++) {
        var nome = 'Flávio';
        console.log(i);
    }

    alert(i);
    alert(nome);
</script>
```

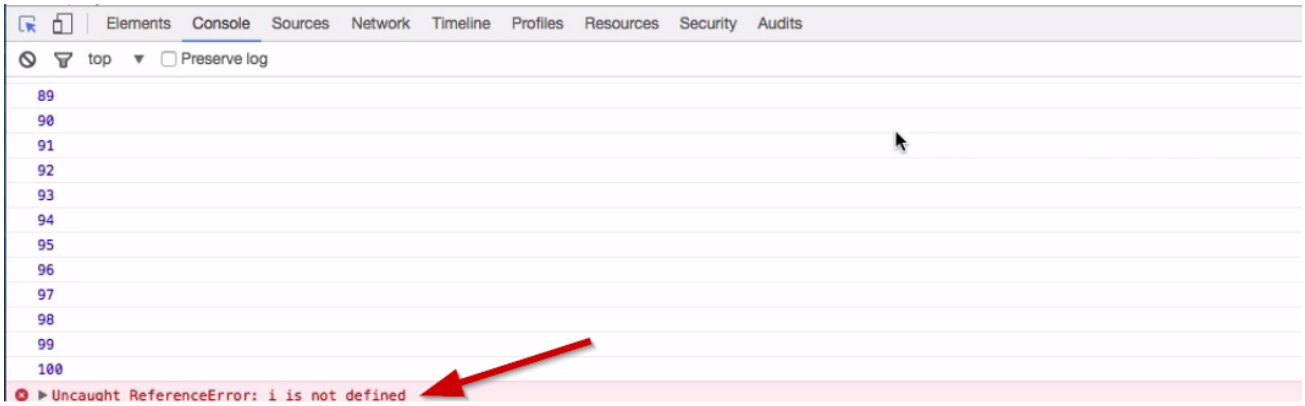
O segundo `alert` também seria exibido.



Em JavaScript não existe escopo de bloco, então o fato de declararmos uma variável dentro de um bloco não garantirá que temos um escopo. No entanto, se declaramos as variáveis usando o `let`, estas ganharam um escopo de bloco.

```
<script>

    for(let i = 1; i<= 100; i++) {
         let nome = 'Flávio';
         console.log(i);
    }

    alert(i);
    alert(nome);
</script>
```

Agora, elas só existirão no bloco em que foram declaradas. Se executarmos o código, veremos a mensagem: `i is not defined`.



Isto ocorreu porque a variável `i` não existe fora do bloco. Desta forma, evitamos que as variáveis vazem fora do escopo que fazem parte.