

05

Criando um objeto Date

Transcrição

Existem várias maneiras de construirmos uma data passando parâmetros para o construtor. Uma das formas que já vimos é que se já temos uma data, podemos colocar o retorno de `getTime` e passar para o construtor de `Date`. Será que podemos fazer este processo de forma direta? Se for possível, já resolvemos o nosso problema.

```
adiciona(event) {
  event.preventDefault();

  let data = new Date(this._inputData.value);
  console.log(data);
}
```

Se der certo o que fizemos, será retornada a string `2015-11-12`. É o texto que o construtor está recebendo. Vamos fazer um teste:

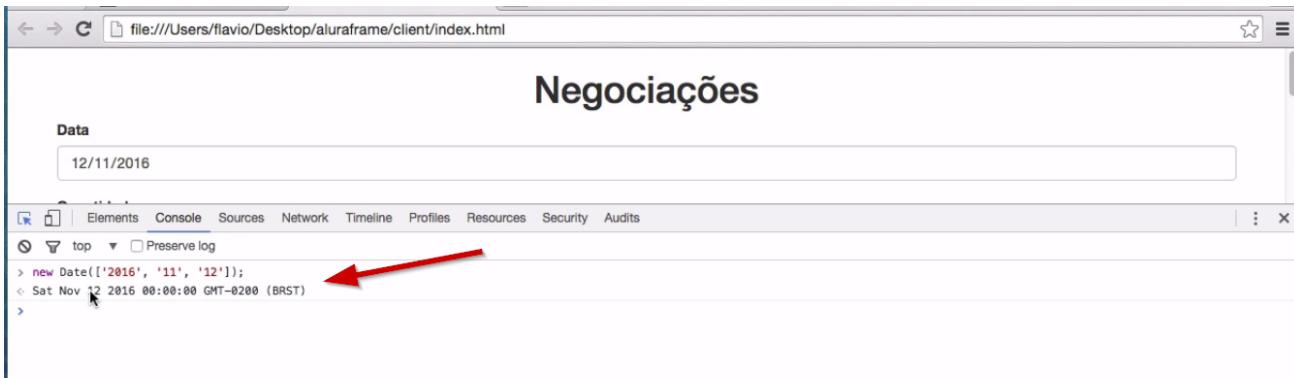


A data que apareceu no Console não foi a mesma, em vez disso, vemos que o dia é `11`. Então, não funcionou. Temos que encontrar outra maneira de trabalhar com essa data. Outra maneira de trabalhar com Date é utilizando um `array`, que terá o `ano`, `mês`, `dia`.

Se digitarmos no Console...

```
new Date(['2016', '11', '12']);
```

O retorno será a data correta:



The screenshot shows a browser developer tools console. At the top, there is a text input field with the value '12/11/2016'. Below the input, the console tab is selected. A red arrow points to the output of the following code:

```
> new Date(['2016', '11', '12']);
< Sat Nov 12 2016 00:00:00 GMT-0200 (BRST)
```

Esta é uma forma de resolvemos. Precisamos ainda encontrar uma forma de transformar a data que é uma *string*, fique dentro do *array*. Como faremos com que a *string* saiba qual é o separador que ela deve considerar? Para isto, adicionaremos `split('-')`, e assim, a data será separada com hífens. Faremos um teste digitando no Console `dataString.split('-')`:

```
dataString.split(' - ')
["2015", "12", "11"]
```

Ele nos retornou a data correta. Então, executaremos o seguinte código:

```
adiciona(event) {
  event.preventDefault();

  let data = new Date(this._inputData.value.split(' - '));
  console.log(data);
}
```

Resolvemos o problema da data.



The screenshot shows a browser developer tools console. The output shows the creation of a Date object from an array and its output:

```
> new Date(['2016', '12', '12']);
< Mon Dec 12 2016 00:00:00 GMT-0200 (BRST)
```

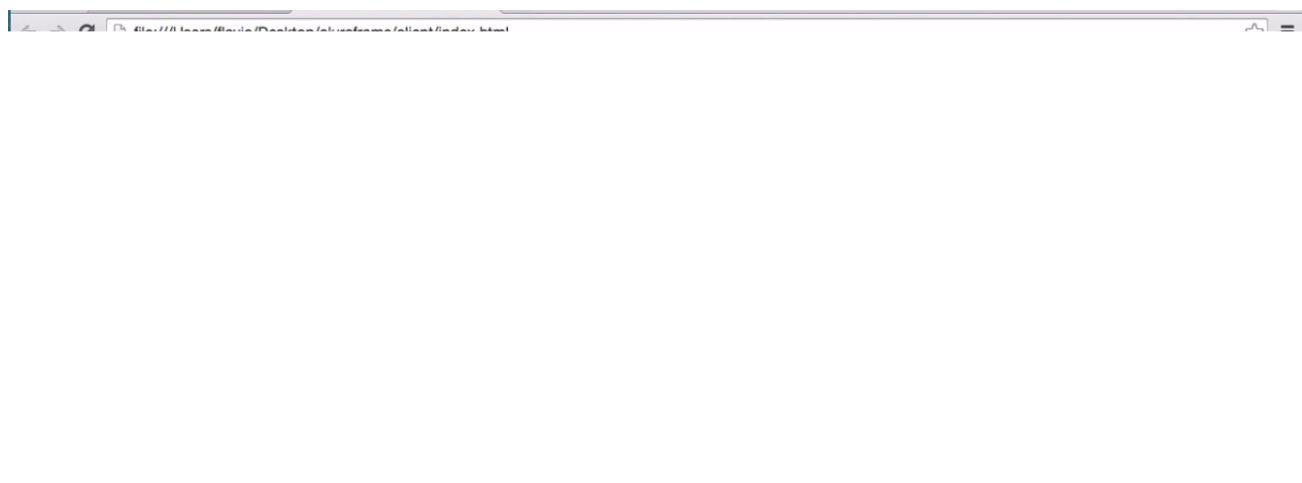
Por que conseguimos solucionar passando um *array*? Quando geramos o *array* com `ano`, `mês` e `dia`, ele transforma cada item em uma *string* e adiciona o separador. Só que quando passamos o *array* `'2016', '11', '12'`, o que é o reagrupamento por debaixo dos panos de cada item usando o separador `,`.



The screenshot shows a browser developer tools console. The output shows the creation of a Date object from an array and its output:

```
> new Date(['2016', '12', '12']);
< Mon Dec 12 2016 00:00:00 GMT-0200 (BRST)
```

Existe no *array* o método `join()`, que une todos os itens e depois, forma uma *string* com separadores.



Temos a string que esperávamos receber. Mas, além de usar o `split()`, poderíamos utilizar o `replace()`. Adicionaremos uma expressão regular pedindo que seja trocado o hífen de todas as ocorrências da string (ou seja, global) por `,`: `replace(/-/g, ',')`.

```
adiciona(event) {  
  
  event.preventDefault();  
  
  let data = new Date(this._inputData.value.replace(/-/g, ','));  
  console.log(data);  
}
```

A data ficou correta. Vimos que existem várias formas de resolver a questão da data, e o `array` nem precisa ser de string. Pode ser um `array` de número, que também será aceito.