

04

Criando uma instância de Negociacao

Transcrição

Nós iremos seguir a convenção de que os `inputs` só poderão ser acessados com `NegociacaoController`, não podendo ser acessados por fora da aplicação. Só os métodos de `NegociacaoController` poderão acessar esses atributos. Para isto, usaremos o prefixo `_`, convencionado anteriormente, que indicará para o programador.

```
class NegociacaoController {  
  
    constructor() {  
  
        let $ = document.querySelector.bind(document);  
  
        this._inputData = $('#data');  
        this._inputQuantidade = $('#quantidade');  
        this._inputValor = $('#valor');  
    }  
  
    adiciona(event) {  
        event.preventDefault();  
  
        console.log(this._inputData.value);  
        console.log(this._inputQuantidade.value);  
        console.log(this._inputValor.value);  
    }  
}
```

Continuamos seguindo a convenção. Agora, podemos criar a negociação com os dados do formulário, vamos então criar a variável `negociacao` e incluiremos os três parâmetros.

```
adiciona(event) {  
  
    event.preventDefault();  
  
    let negociacao = new Negociacao(  
        this._inputData.value,  
        this._inputQuantidade.value,  
        this._inputValor.value  
    );  
  
    console.log(negociacao);  
}
```

Passamos os parâmetros `data`, `quantidade` e `valor` para o construtor de `Negociacao`. Com os dados do formulário, nós já temos uma negociação. Queremos que a negociação seja adicionada em uma lista, mas não a iremos implementar ainda.

Vamos ver como ficou a página recarregada.



Recebemos uma mensagem de erro, mas não foi em `NegociacaoController`. O erro ocorreu na classe `Negociacao.js`. No Console, vemos ainda em qual linha foi o problema. A falha nos avisou que `data.getTime` não é uma função. Como isso é possível?

```
class Negociacao {

  constructor(data, quantidade, valor) {

    this._data = new Date(data.getTime());
    this._quantidade = quantidade;
    this._valor = valor;
    Object.freeze(this);

  }
}
```

O que significa? A data que estamos passando para o construtor não é uma data? Para resolver isso, vamos descobrir qual é o tipo da data, usaremos a função `typeof()`.

```
adiciona(event) {

  event.preventDefault();

  console.log(typeof(this._inputData.value));

  let negociação = new Negociacao(
    this._inputData.value,
    this._inputQuantidade.value,
    this._inputValor.value
  );

  console.log(negociação);

}
```

No Console, veremos que é uma *string*.

A screenshot of a browser's developer tools console. The URL in the address bar is "file:///Users/flavio/Desktop/aluraframe/client/index.html". The title of the page is "Negociações". In the console, there is an error message: "Uncaught TypeError: data.getTime is not a function" at line 15 of "NegociacaoController.js". The word "string" is highlighted with a red arrow pointing to it.

Isto significa que quando passamos a data para `Negociacao`, ela ainda é uma string e, obviamente, não teremos o `getTime()`. O que precisamos resolver antes? Precisamos encontrar uma maneira de converter a data - que está em texto - para um objeto `Data`. Priorizaremos isso, e cancelaremos a criação da negociação.

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

    console.log(typeof(this._inputData.value));

    console.log(this._inputData.value);

}
```

Ao executarmos o código, veremos o seguinte código:

A screenshot of a browser's developer tools console. The URL in the address bar is "file:///Users/flavio/Desktop/aluraframe/client/index.html". The title of the page is "Negociações". In the console, the output of `this._inputData.value` is shown as a Date object: "2015-11-12T00:00:00.000Z". The word "2015-11-12" is highlighted with a red arrow pointing to it.

Temos que ser capazes de transformar uma string em um objeto do tipo `Date` e depois passá-lo para negociação.

Mais adiante, faremos este processo de conversão.