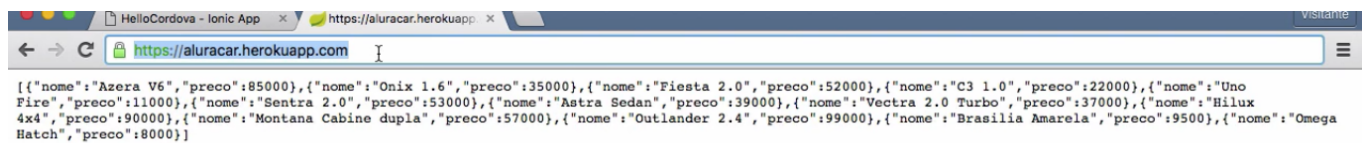


Lista editável pelo cliente

Lista editável pelo cliente

Bem-vindos à mais uma aula de Ionic! Nossa aplicação já possui a lista de carros, permite que adicionemos acessórios e também finaliza o pedido, incluindo um pop-up de confirmação e o retorno à tela inicial. Como o cliente da aplicação é uma concessionária, a lista de carros disponíveis deve variar e ele pode querer que sua própria equipe de TI faça isso. Assim, não faz sentido que a lista fique fixa no código, pois não seria prático que a cada alteração nós precisássemos fazer o *build* e publicá-la. Precisamos permitir que o cliente altere a lista quando vender um carro (ou quiser trocá-lo ou tirá-lo da lista por outro motivo) e a aplicação a atualize.

Para isso, a lista vai ficar disponível no aplicativo `aluracar.herokuapp.com`, e quando se fizer uma requisição `get` via HTTP, ele mostrará a lista de carros já em formato JSON, com seus nomes e preços.



Essa lista precisa ser consumida dentro da nossa aplicação. E como fazer essa chamada HTTP em outro servidor e consumir a lista? Primeiramente, é preciso chamar essa `get`, e como isso é um serviço, será feito em um arquivo `javascript` separado.

Esse arquivo ficará na pasta `javascript` e se chamará `service.js`. Criaremos o serviço em Angular, que é o que estamos usando, com o `module` que já está no arquivo `app.js` e chamando a URL onde está a lista:

```
angular.module('starter')
  .service('CarroService', function(){

    var url= 'http://aluracar.herokuapp.com/';

  });
```

Precisamos agora retornar uma função (`obterCarros`) que faça chamada na URL que inserimos. Como a chamada será feita em HTTP, a injetaremos como função dependência. Lembrando que já criamos uma variável para a URL, o código ficará assim:

```
angular.module('starter')
  .service('CarroService', function($http){

    var url= 'http://aluracar.herokuapp.com/';

    return{
      obterCarros : function (){
        return $http.get(url).then(function(response){
          return response.data;
        });
      }
    }
  });
```

```
}  
  
});
```

A chamada `obterCarros` precisa corresponder à nossa lista atual, que está na *controller* `listaDeCarros`. Para que o serviço nos traga a lista certa, precisamos injetá-lo na *controller*, inserindo o serviço `CarroService`. Como não podemos deixar duas listas de carros, a anterior será deletada. O código da *controller* ficará assim:

```
.controller('ListagemController', function($scope, CarroService){  
  
    CarroService.obterCarros().then(function(dados){  
  
        $scope.listaDeCarros = dados;  
    });  
  
});
```

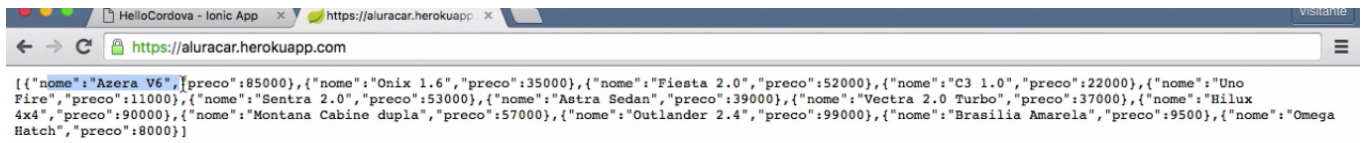
Antes de testar na aplicação, precisamos acrescentar informação ao `index.html` para que ele carregue o serviço e saiba onde está o `CarroService`. Para isso, usaremos um arquivo `.js`, acrescentando a linha:

```
<script src="js/service.js"></script>
```

Agora, ao testar a aplicação, veremos que a lista está diferente da lista inicial.

Alura car
Azera V6 - R\$85,000.00
Onix 1.6 - R\$35,000.00
Fiesta 2.0 - R\$52,000.00
C3 1.0 - R\$22,000.00
Uno Fire - R\$11,000.00
Sentra 2.0 - R\$53,000.00
Astra Sedan - R\$39,000.00
Vectra 2.0 Turbo - R\$37,000.00
Hilux 4x4 - R\$90,000.00

Isso acontece porque o cliente a atualizou no servidor:



The screenshot shows a web browser window with the address bar displaying `https://aluracar.herokuapp.com`. The page content is a JSON array of car objects, each with `nome` and `preco` properties. The first object, `{ "nome": "Azerar V6", "preco": 85000 }`, is highlighted with a blue selection box. The rest of the array is visible below it.

```
[{"nome": "Azerar V6", "preco": 85000}, {"nome": "Onix 1.6", "preco": 35000}, {"nome": "Fiesta 2.0", "preco": 52000}, {"nome": "C3 1.0", "preco": 22000}, {"nome": "Uno Fire", "preco": 11000}, {"nome": "Sentra 2.0", "preco": 53000}, {"nome": "Astra Sedan", "preco": 39000}, {"nome": "Vectra 2.0 Turbo", "preco": 37000}, {"nome": "Hilux 4x4", "preco": 90000}, {"nome": "Montana Cabine dupla", "preco": 57000}, {"nome": "Outlander 2.4", "preco": 99000}, {"nome": "Brasilia Amarela", "preco": 9500}, {"nome": "Omega Hatch", "preco": 8000}]
```

Sabemos então que a aplicação está pronta para receber alterações na lista, mostrando só carros que a concessionária realmente quer vender. Encerramos nossa aula por aqui. Até a próxima!