

04

## Json, json convert, NewtonSoft, VeiculoJson

### Transcrição

[00:00] Agora que a gente já tem aqui o resultado, essa string contendo os veículos que vieram do serviço. A gente vai ter que transformar isso em objetos. Como a gente poderia fazer isso? A gente poderia pegar a string e fazer o (parse), que seria a interpretação desse texto e buscar por caracteres...

[00:23] Por vírgula, por aspas, por chaves, para poder separar, cortar essa string em vários pedaços, cada pedaço seria um veículo e a gente poderia pegar dentro de cada pedaço, pegar informações ainda menores, como o nome da propriedade e o valor dessa propriedade.

[00:45] Isso seria extremamente cansativo e chato de fazer e além disso, sujeita muitos erros. A gente poderia perder muito tempo e fazer uma coisa totalmente errada ainda. Então, nesse caso, a melhor coisa a fazer é utilizar, invés de reinventar a roda, a gente pega a roda que outra pessoa fez.

[01:10] A gente pega bibliotecas que já existem a muito tempo, que já são consagradas, que já são especializadas em fazer isso e fazem muito bem. Então, a gente vai ter que usar uma biblioteca, que vai converter string no formato Json, que é o formato que a gente está obtendo a partir do serviço http...

[01:31] E transforma essa string Json em objetos em classes, objetos de instâncias de classes do Csharp. Então, assim, a biblioteca mais comum, mais utilizada no mundo (dot net), com certeza é a Newtonsoft. Então, a gente vai fazer a instalação aqui de uma biblioteca de conversão de objetos Json.

[02:03] Entramos aqui no NuGet Package Manager e a gente vai procurar por Json. Olha só, sem buscar por nada, logo de cara o Visual Studio já para a gente, que está disponível para a gente a biblioteca Newtonsoft.Json. De tão popular que ela é, ela vem logo de cara, a gente não precisou nem buscar nada.

[02:27] Então, o que a gente vai fazer aqui é instalar o nosso primeiro projeto aqui, que é TestDrive. Então, seleciono o projeto, clico em instalar, ele vai começar a instalar para a gente. Eu vou dar o ok e ele está instalando agora. É uma biblioteca muito leve, ele já instalou para a gente.

[02:49] Está marcado como instalado. Eu vou fechar agora Nuget Package Manager e vou tentar acessar aqui algumas funções do método do Newtonsoft. Então, a classe de conversão do Json, do Newtonsoft, é o JsonConvert. Então, eu vou utilizar o JsonConvert e essa classe...

[03:18] Eu vou colocar aqui a referência Newtonsoft, já acho a referência e eu vou fazer o quê? O contrário da serialização, porque o que a gente tem aqui é uma string, então essa string foi obtida a partir de objetos e ela foi serializada em string. Então, o que a gente vai fazer para obter os objetos do sistema...

[03:44] É justamente o evento contrário, a gente vai desserializar essa string e transformar ela em objetos. Então, a gente vai utilizar o método DeserializeObject. No DeserializeObject, eu vou passar aqui o tipo que eu quero utilizar, o tipo alvo, o tipo para o qual a string vai ser convertida.

[04:11] E que tipo é esse? Olha só, o nosso objeto, ele teria que ter exatamente as mesmas propriedades do código Json, da string Json. Se a gente olhar bem aqui no resultado que está vindo lá do serviço. Vamos entrar aqui no browser. Então, olha só, eu tenho aqui nome e o preço, nome e preço, nome e preço.

[04:40] Eu só tenho essas duas propriedades. A gente até tem um objeto na nossa aplicação chamado veículo. Só que veículo, ela tem uma propriedade pública chamada nome e outra chamada preço, só que elas estão com a primeira letra, o primeiro caractere maiúsculo.

[05:01] E no caso aqui, a gente tem o primeiro caractere minúsculo. Então, o que acontece? Se a gente tenta converter esse Json, essa string para objetos do tipo veículo, a gente vai ter um conflito, a gente não vai conseguir encontrar uma propriedade adequada para a conversão.

[05:24] Então, o que a gente vai fazer? Bom, como são só duas propriedades, nome e preço, o que eu vou fazer é simplesmente criar uma classe nova, uma classe muito, muito simples, que vai ser utilizada só para esse ponto do sistema, que vai ser a conversão da string, para o objeto do (dot net).

[05:45] Então, primeiro eu crio aqui uma classe, eu vou chamar ela de VeiculoJson e essa classe vai ter duas propriedades, que são justamente aquelas que estão vindo lá do sistema, lá do nosso serviço. Então, a primeira é uma string, que vai se chamar nome.

[06:13] E a outra propriedade vai ser o preço. Então, o preço aqui está vindo com um inteiro, ele está vindo como um número. Então, eu vou simplesmente colocar aqui como inteiro e vou chamar essa propriedade de preço. Então, eu coloco aqui “preco”, minúsculo, claro.

[06:36] E aí, pronto, eu tenho uma classe que está pronta, disponível para ser utilizada nessa conversão da string Json, para o código (dot net). E eu coloco aqui dentro, lá dentro do DeserializeObject, eu coloco qual é o tipo de destino dessa desserialização. Aí, eu coloco aqui, VeiculoJson, ficando: JsonConvert.DeserializeObject.

[07:00] Só que tem um outro porém, porque o resultado do nosso serviço, não é exatamente um objeto Json, ele é uma coleção, uma lista ou melhor, ele é um array de objetos, por quê? Porque, se você prestar atenção aqui, ele começa com um colchete e termina com um colchete.

[07:25] E dentro desses colchetes, a gente tem as chaves que indicam o início e o fim de um objeto e as vírgulas indicam que a gente tem uma lista, a gente uma array de vários objetos lá dentro. Então, para a gente desserializar corretamente essa string Json, a gente tem que fazer o mesmo...

[07:47] A gente tem que colocar um tipo alvo, um destino desse string Json, tem que ser um tipo correto também, então, não basta ser só um veículo Json, ele tem que ser também um array. Então, como que eu coloco um array aqui no código (dot net), no Csharp?

[08:04] Eu coloco dois colchetes, abre e fecha colchete, isso vai indicar que eu quero o resultado da conversão, que seja feita como um array de objetos chamados VeiculoJson. Então, aí eu passo, coloco aqui o parênteses e vou passar como parâmetro a string, que vai ser a string de origem.

[08:28] Essa string de origem, é justamente a string que o serviço está retornando para a gente. Aqui no caso, é o resultado, é a nossa variável aqui resultado, que eu vou passar aqui embaixo. Então, agora que eu a gente tem essa conversão, eu tenho que jogar isso em algum lugar.

[08:50] Eu preciso jogar isso numa variável. Então, eu vou chamar essa variável de veiculosJson. Então, aí, veiculosJson vai receber o resultado dessa conversão. Então, antes de continuar com o nosso raciocínio, com a nossa aplicação. Eu vou rodar a aplicação.

[09:17] Vou rodar o programa, para ver como que ele converteu esses objetos. Então, vamos lá. Então, agora está rodando o programa. Vai fazer agora a chamada do serviço. Tá, então ele já fez a conversão, vamos ver o resultado agora. Então, olha só, veiculosJson, agora ele é o quê?

[09:43] Ele é um array de veículos, de veículo Json, se eu expandir aqui, eu tenho uma instância para cada veículo que foi obtido lá na string. Então, com isso, a gente conseguiu converter corretamente uma string Json e vários objetos .NET, que agora vão estar disponíveis, para a gente poder jogar na listagem.

[10:08] E aí, a gente vai popular a listagem para poder mostrar na tela. Então, com isso, vocês viram como fazer uma conversão de uma maneira simples, utilizando uma ferramenta consagrada, que é a biblioteca Newtonsoft. Então, agora a gente vai passar para a próxima parte.