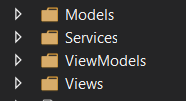
**Estudo API**

1. Criar o Projeto Xamarin Forms
2. Baixar a API (SENAI\_2S2020\_API-master)
3. Mudar o Ip para rodar localmente
4. Criar as pastas (Arquiteruta MVVM)

 Models

Views

ViewModels

Services

1. Criar a page VeiculoView

<StackLayout

Padding="10"

Spacing="10">

<Label Text="Veiculos"

FontSize="20"

HorizontalOptions="CenterAndExpand"/>

<ListView

ItemsSource="{Binding veiculos}">

<ListView.ItemTemplate>

<DataTemplate>

<ViewCell>

<StackLayout>

<Label

Text="{Binding marca}"

FontAttributes="Bold"

FontSize="16"/>

<Label

Text="{Binding modelo}"

FontSize="16"/>

</StackLayout>

</ViewCell>

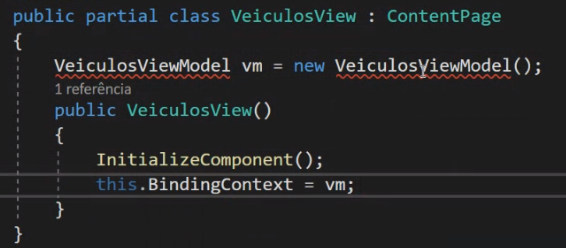
</DataTemplate>

</ListView.ItemTemplate>

</ListView>

</StackLayout>

1. No VeiculosView.xaml.cs

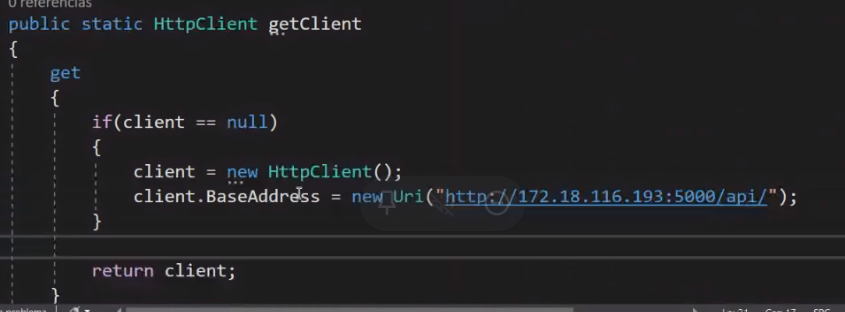


1. Criar a Classe VeiculosViewModel

copiar o json da api

editar / colar especial / colar json como classe

1. CRIAR a classe utils (em Services)



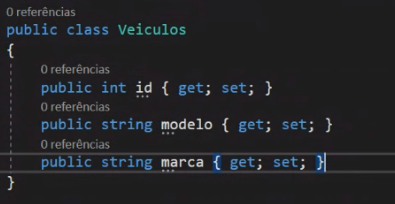
Uri = url

Base = endereço de base (igual do postman)

copiar o jason da api

1. CRIAR a classe Veiculo

Copiar os atributos da classe Veiculo

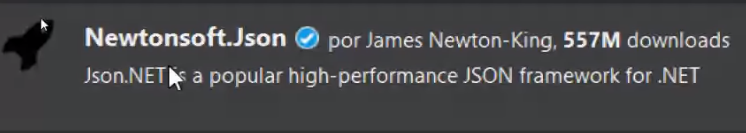


Da para copiar e colar especial (json)

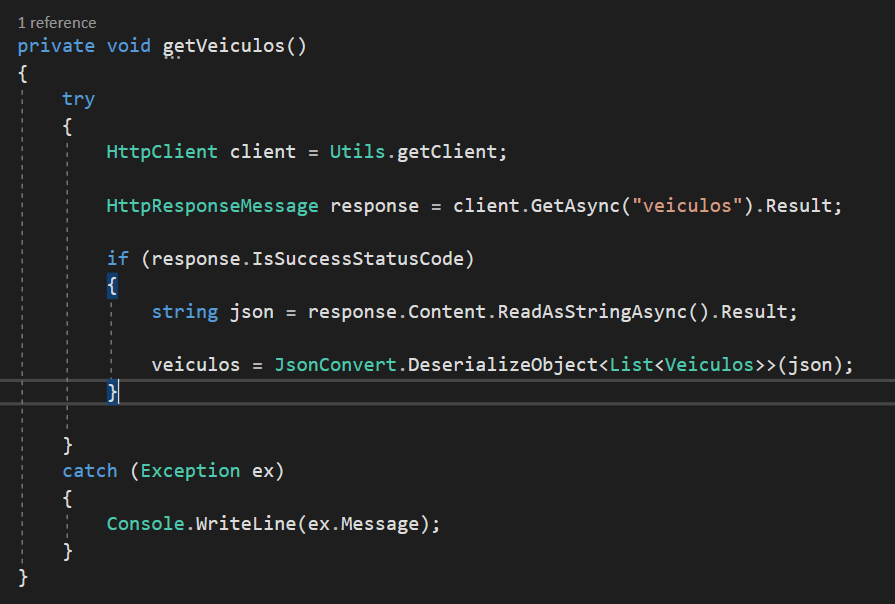
1. CRIAR: public class VeiculosViewModel

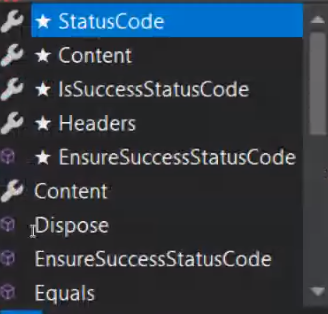


1. Adicionar a Biblioteca (NewtonSoft)



1. Criar o método getVeiculo



* “veiculos” é o endpoint (ele vai concatenar com o addres definido no utils)
* A task que o getasync retorna é a resposta da api (retorno do servidor) – tipo o status code ou o body (que pode ser os dados do json).
* O retorno é uma message (http response message)
* Este código vai buscar os dados e retorna o resultado e armazena no response
* Response vai me dar muitos retornos:
* Vc pode tratar os retornos (as respostas)
* Serializer (ele pega um objeto e transforma em json)
* Deserializer (transforma o json em objeto)
* Precisa informar o Deserialize qual o tipo do objeto (<list<veículos>>

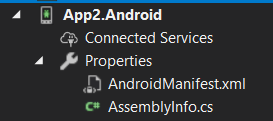
1. Deserialize

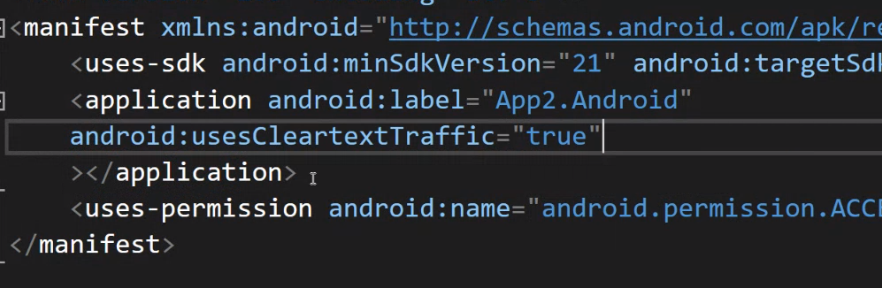


TRASFORMA O JSON EM UMA LISTA DE VEICULOS

1. Configuração necessária para o dispositivo fazer as requisições http

Em AndroidManifes.xml





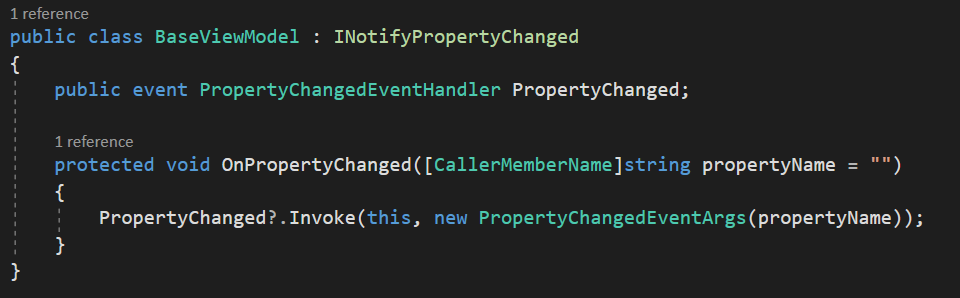
1. Atualização após alteração

* Para que a tela seja informada qdo a lista sofrer alteração
* criar na view model uma classe baseViewModel
* Terá um método para notificar a interface que houve mudança

CallerMemberName – ele vai colocar o nome da propriedade que está sendo chamada

onPropertyChanged – toda vez que a propriedade é alterada

se o propertychanged é nulo: coloca um ponto de **?** na frente do método



Para chamar vai na veiculo viewmodel

