

Altere as propriedades:

Controle	Propriedade	Valor
Form1	Name	FormCadastrarVeiculos
Form1	AcceptButton	Button1
Form1	StartPosition	CenterScreen
Form1	Text	Cadastro de Veiculos
Button1	Name	btnGravar
Button1	Anchor	Top, Left, Right
Button1	Font / Bold	True
Button1	Font / Size	12
Button1	Text	Gravar
Label1	Name	Label1
Label1	Font / Bold	True
Label1	Text	&Placa:
Label2	Name	Label2
Label2	Font / Bold	True
Label2	Text	&Modelo:
Label3	Name	Label3
Label3	Font / Bold	True
Label3	Text	&Cor:
Label4	Name	Label4
Label4	Font / Bold	True
Label4	Text	&Ano:

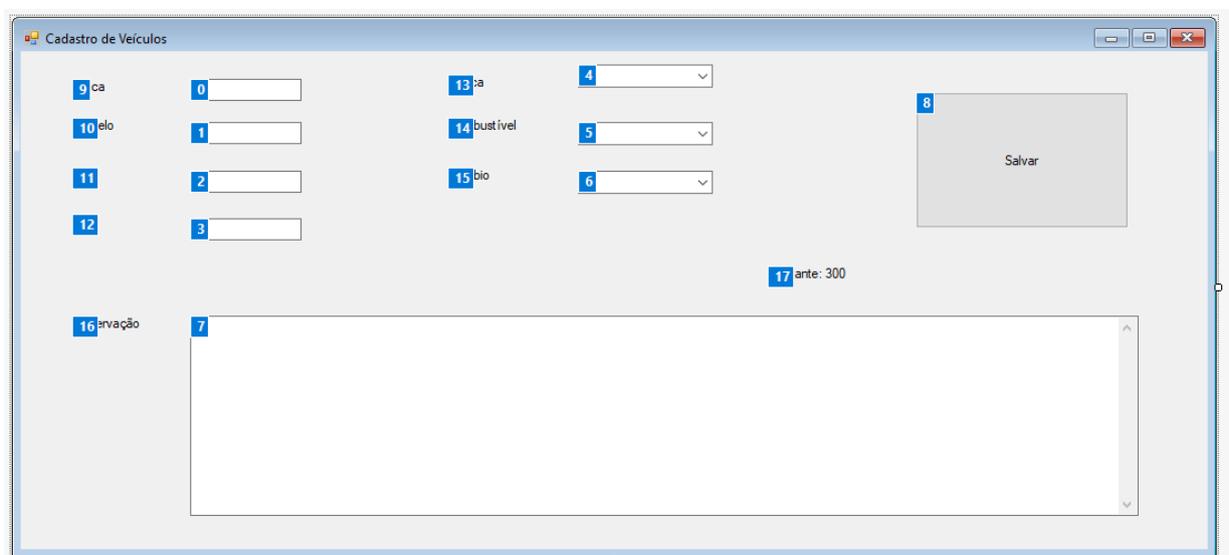
Controle	Propriedade	Valor
Label5	Name	Label5
Label5	Font / Bold	True
Label5	Text	Marca:
Label6	Name	Label6
Label6	Font / Bold	True
Label6	Text	Combustível:
Label7	Name	Label7
Label7	Font / Bold	True
Label7	Text	Câmbio:
Label8	Name	Label8
Label8	Font / Bold	True
Label8	Text	Observação:
Label9	Name	lbltoques
Label9	AutoSize	false
Label9	Font / Size	7
Label9	Text	Restante: 300
Label9	TextAlign	BottomRight
TextBox1	Name	txtplaca
TextBox1	AcessibleName	Placa
TextBox1	CharacterCasing	Upper
TextBox1	MaxLength	7
TextBox1	TextAlign	Center
TextBox2	Name	txtmodelo
TextBox2	AcessibleName	Modelo
TextBox2	MaxLength	30
TextBox3	Name	txtcor
TextBox3	AcessibleName	Cor
TextBox3	MaxLength	30
TextBox4	Name	txtano
TextBox4	AcessibleName	Ano
TextBox4	MaxLength	4
TextBox4	TextAlign	Center
TextBox5	Name	txtobservacao
TextBox5	AcessibleName	Observação
TextBox5	Anchor	Top, Bottom, Left, Right
TextBox5	Font / Size	16
TextBox5	MaxLength	300
TextBox5	Multiline	true
TextBox5	ScrollBars	Vertical

Controle	Propriedade	Valor
ComboBox1	Name	cbomarca
ComboBox1	AcessibleName	Marca
ComboBox1	AutoCompleteMode	Suggest
ComboBox1	AutoCompleteSource	ListItems
ComboBox1	DropDownStyle	DropDownList
ComboBox2	Name	cbocombustivel
ComboBox2	AcessibleName	Combustível
ComboBox2	AutoCompleteMode	Suggest
ComboBox2	AutoCompleteSource	ListItems
ComboBox2	DropDownStyle	DropDownList
ComboBox3	Name	cbocambio
ComboBox3	AcessibleName	Câmbio
ComboBox3	AutoCompleteMode	Suggest
ComboBox3	AutoCompleteSource	ListItems
ComboBox3	DropDownStyle	DropDownList

Crie o Layout acima e altere as propriedades conforme a tabela acima.

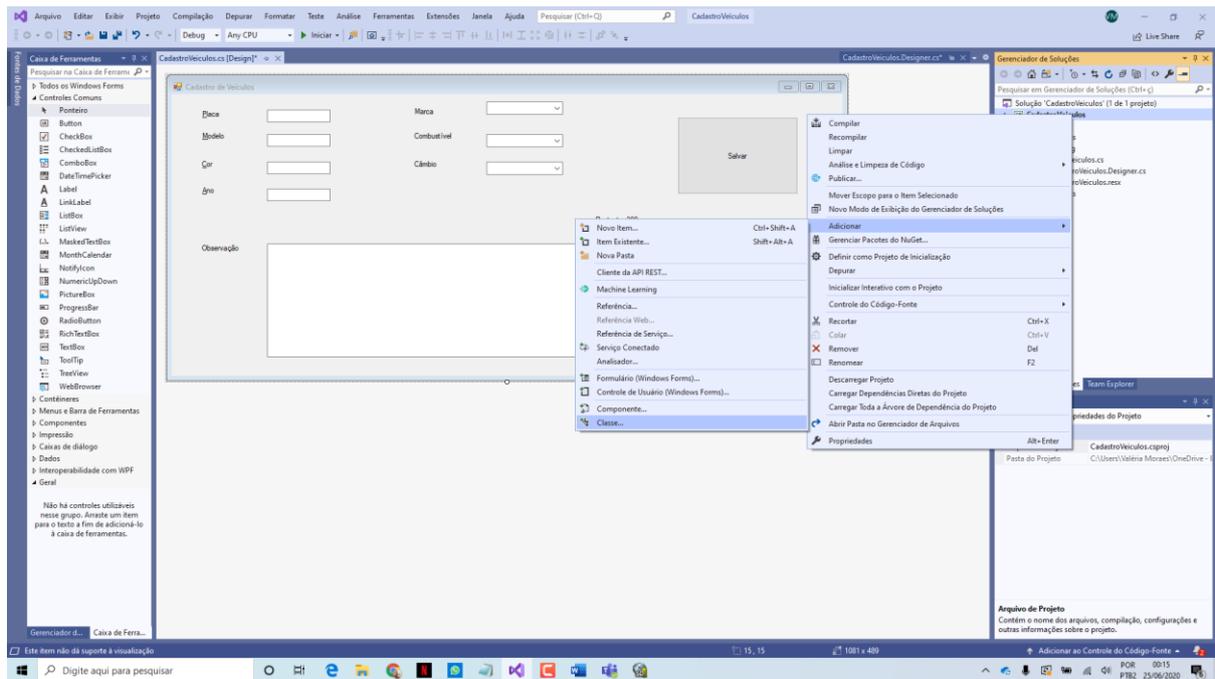
Coloque a ordem de tabulação conforme a imagem abaixo, para facilitar a navegação do usuário:

Menu Exibir/Ordem de Tabulação ---View /Tab Order



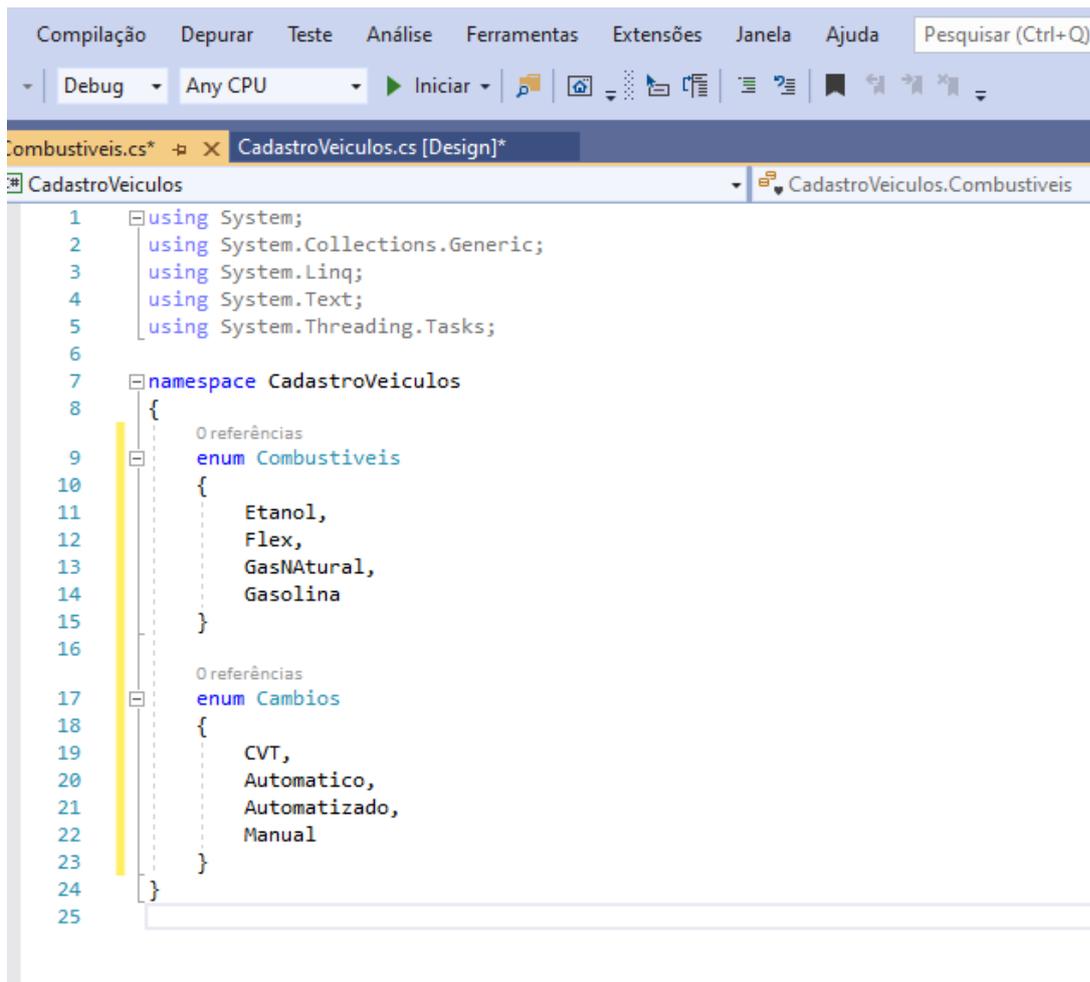
Classe Combustível

Na janela Solution Explorer, clique com o botão direito sobre o projeto CadastroVeiculo, selecione a opção ADD e em seguida Class.

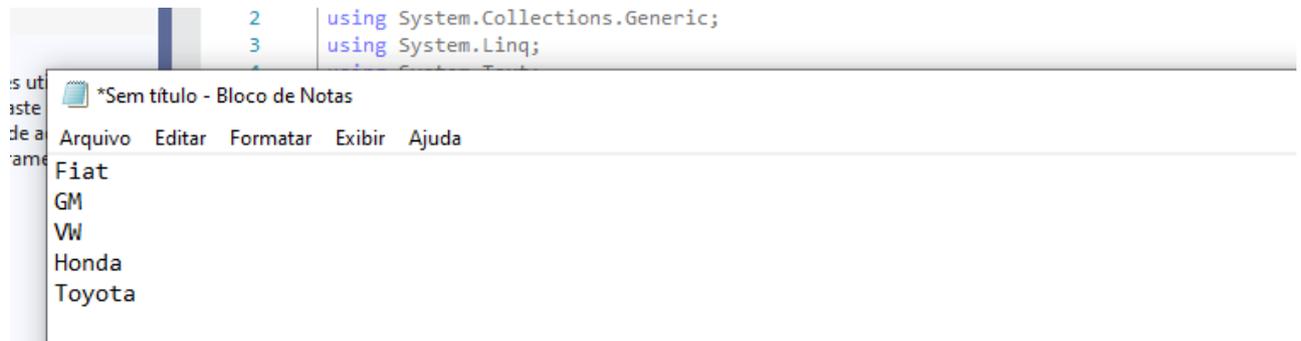


Crie a classe Combustiveis e clique em ADD (Adicionar)

Troque a palavra class por enum, e preencha como a imagem abaixo



Crie um arquivo texto de nome Marcas.txt no bloco de notas e salve junto com seu projeto, preencha conforme a imagem abaixo



Salve seu trabalho sem erros

Classe Métodos Extensão

Crie uma nova classe:

MetodosExtensao

Adicione a diretiva para o namespace conforme abaixo : using....

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 //-----
7 using System.Windows.Forms;
8 using System.Text.RegularExpressions;
9
```

Depois vamos criar os métodos de extensão a seguir:

```
9
10 namespace CadastroVeiculos
11 {
12     static class MetodosExtensao
13     {
14         //Definir o método ValidarVazio()...
15         public static string ValidarVazio(this TextBox txt)
16         {
17             if (string.IsNullOrEmpty(txt.Text))
18             {
19                 txt.Focus();
20                 txt.SelectAll();
21                 throw new Exception(
22                     "Preencha o campo" + txt.AccessibleName.ToUpper());
23             }
24             return txt.Text;
25         }
26
27         //Definir método ValidarCombo()...
28         public static string ValidarCombo (this ComboBox cbo)
29         {
30             if (cbo.SelectedIndex == -1)
31             {
32                 cbo.Focus();
33                 throw new Exception(
34                     "Selecione um item de " + cbo.AccessibleName.ToUpper());
35             }
36             return cbo.Text;
37         }
38     }
39 }
```

```

37 }
38
39 //Definir o método ValidarComboEnum()...
0 referências
40 public static Int32 ValidarComboEnum (this ComboBox cbo)
41 {
42     if (cbo.SelectedIndex == -1)
43     {
44         cbo.Focus();
45         throw new Exception(
46             "Selecione um item de " + cbo.AccessibleName.ToUpper());
47     }
48     return cbo.SelectedIndex;
49 }
50
51 //Definir o método ValidarInt16()
0 referências
52 public static short ValidarInt16(this TextBox txt)
53 {
54     try
55     {
56         return Convert.ToInt16(txt.Text);
57     }
58     catch
59     {
60         txt.Focus();
61         txt.SelectAll();
62         throw new Exception("O Ano deve ser numérico");
63     }
64 }
65 //definir método ValidarAno()
0 referências
66 public static short ValidarAno(this short ano)
67 {
68     if (ano < 1950 || ano > DateTime.Now.Year + 1)
69     {
70         throw new Exception(
71             "Ano deve estar entre 1950 e " + (DateTime.Now.Year + 1).ToString());
72     }
73     return ano;
74 }
75

```

```

69         txt.Focus();
70         throw new Exception("O Ano deve ser numérico");
71     }
72 }
73
74 //Definir o método ValidarPlaca:
1 referência
75 public static string ValidarPlaca ( this string placa)
76 {
77     // if (!Regex.IsMatch(placa, @"^[a-zA-Z]{3}\-\d{4}$"))
78     // {
79     //     throw new Exception("Informe uma placa válida");
80     // }
81     // return placa;
82
83     Regex regex = new Regex(@"^[a-zA-Z]{3}\-\d{4}$");
84     if (regex.IsMatch(placa))
85     {
86         throw new Exception("Informe uma placa válida");
87     }
88
89     return placa;
90 }
91
92 //Definir o método Limpartela:
2 referências
93 public static void LimparTela( this Form formulario)
94 {
95     foreach (Control ctl in formulario.Controls)
96

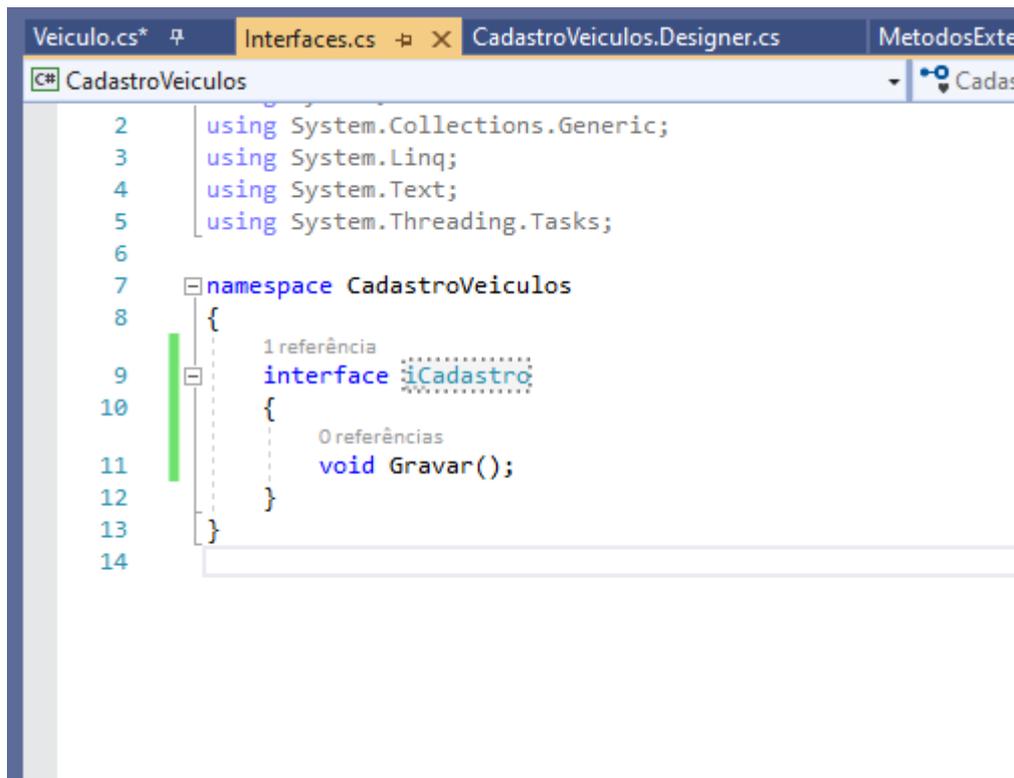
```

```
92     }
93
94
95     //Definir o método Limpartela:
96     2 referências
97     public static void LimparTela( this Form formulario)
98     {
99         foreach (Control ctl in formulario.Controls)
100        {
101            if (ctl is TextBox)
102            {
103                ctl.Text = string.Empty;
104            }
105            else if (ctl is ComboBox)
106            {
107                (ctl as ComboBox).SelectedIndex = -1;
108            }
109        }
110    }
111 }
112 }
113 }
114 }
```

Classe Interface iCadastro

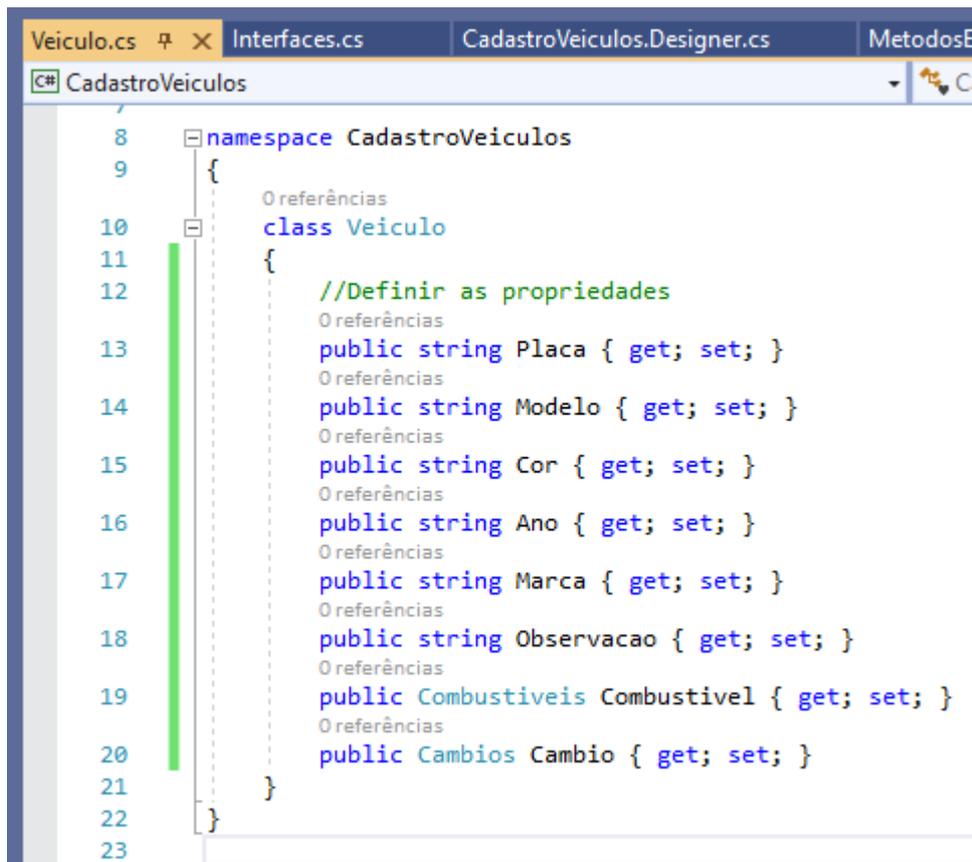
Crie uma nova classe Interfaces

Altere de class interfaces para interface ICadastro como na imagem abaixo, e faça o código da classe como segue.



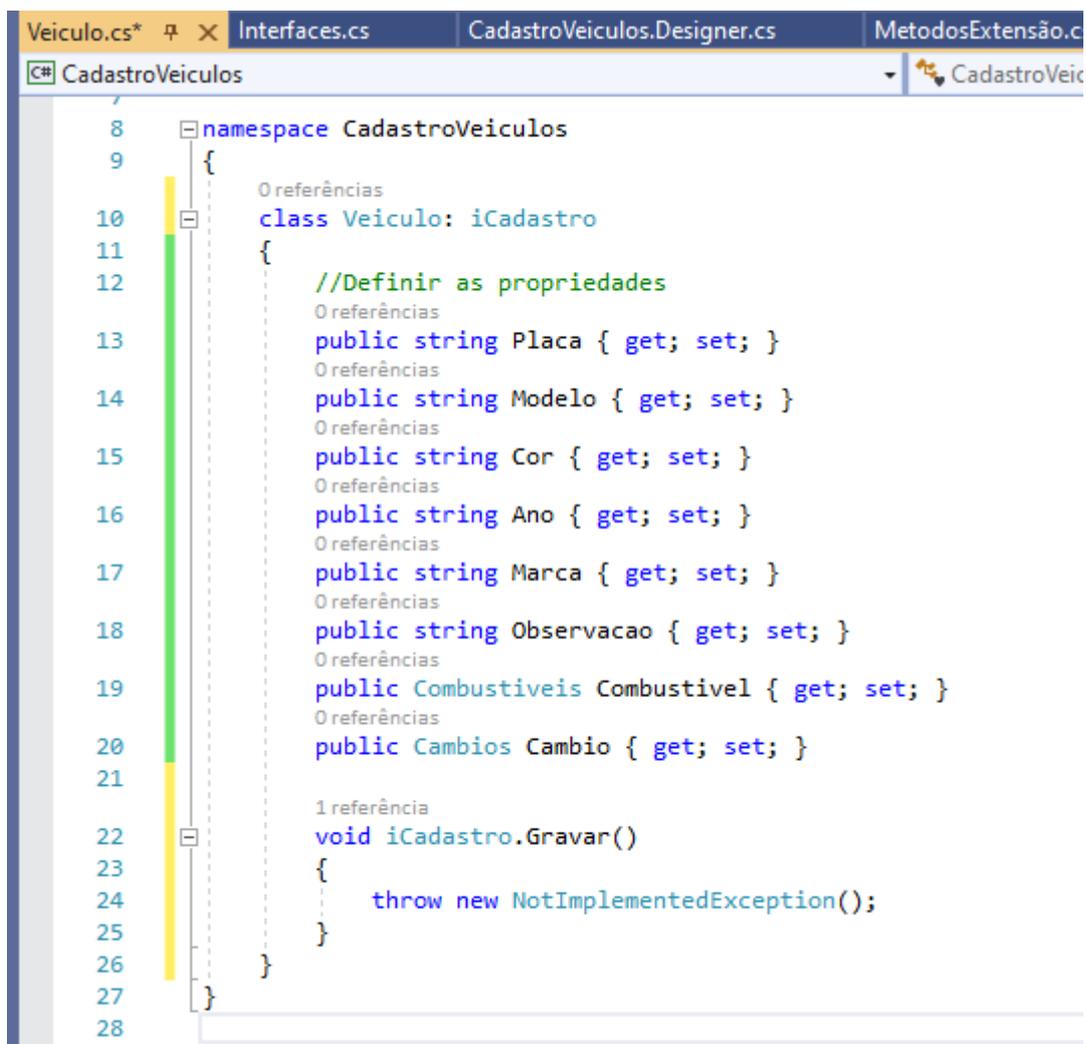
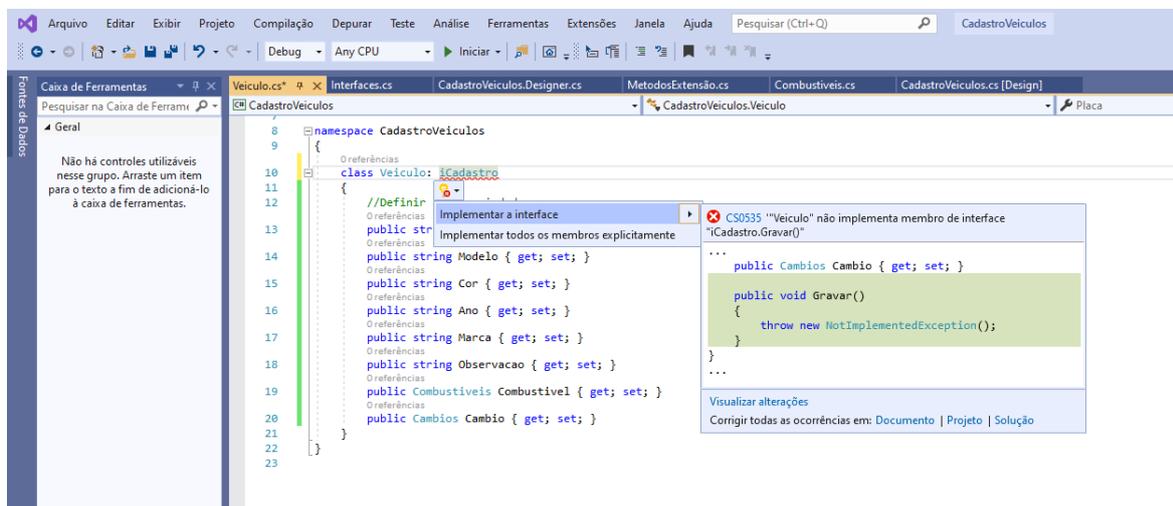
```
Veiculo.cs*  Interfaces.cs  CadastroVeiculos.Designer.cs  MetodosExte
[C#] CadastroVeiculos
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace CadastroVeiculos
8  {
9      1 referência
10     interface iCadastro
11     {
12         0 referências
13         void Gravar();
14     }
15 }
```

Crie uma nova classe Veiculo e crie as propriedades conforme a imagem abaixo:



```
Veiculo.cs  Interfaces.cs  CadastroVeiculos.Designer.cs  MetodosE
[C#] CadastroVeiculos
7
8  namespace CadastroVeiculos
9  {
10     0 referências
11     class Veiculo
12     {
13         //Definir as propriedades
14         0 referências
15         public string Placa { get; set; }
16         0 referências
17         public string Modelo { get; set; }
18         0 referências
19         public string Cor { get; set; }
20         0 referências
21         public string Ano { get; set; }
22         0 referências
23         public string Marca { get; set; }
24         0 referências
25         public string Observacao { get; set; }
26         0 referências
27         public Combustiveis Combustivel { get; set; }
28         0 referências
29         public Cambios Cambio { get; set; }
30     }
31 }
```

Agora vamos implementar a interface ICadastro na classe veiculo mesmo, observe:



Adicione a diretiva `using System.IO;`

Agora escreveremos o método construtor junto com uma variável de apoio para o endereço de gravação do arquivo; escrever o código antes da linha `//definir propriedades`

```
Veiculo.cs* X Interfaces.cs CadastroVeiculos.Designer.cs Metoc
C# CadastroVeiculos
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Reflection.Emit;
5 using System.Text;
6 using System.Threading.Tasks;
7
8 using System.IO;
9
10 namespace CadastroVeiculos
11 {
12     1 referência
13     class Veiculo: ICadastro
14     {
15         //Definir a variavel caminho
16         private string caminho = string.Empty;
17         // definir o método construtor
18         0 referências
19         public Veiculo (string EnderecoArquivo)
20         {
21             caminho = EnderecoArquivo;
22         }
23         //Definir as propriedades
24         0 referências
25         public string Placa { get; set; }
26         0 referências
27         public string Modelo { get; set; }
28     }
29 }
```

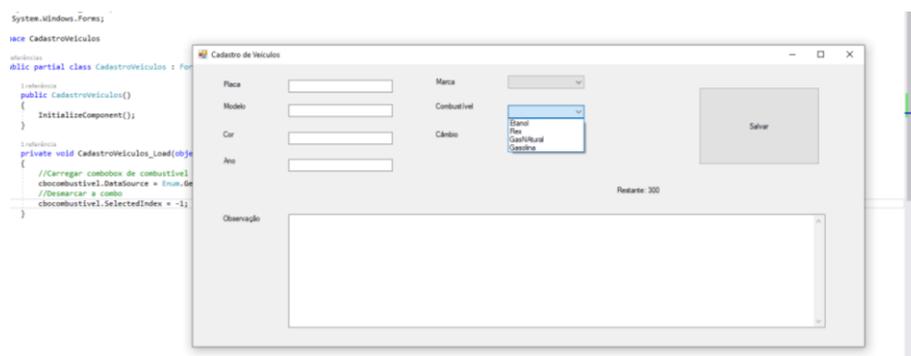
Agora vamos escrever o método Gravar , que foi criado na implementação da interface ICadastro

```
Veiculo.cs* x Interfaces.cs* CadastroVeiculos.Designer.cs MetodosExtensao.cs Combustiveis.cs Cada
C# CadastroVeiculos
1 referência
28 public Combustiveis Combustivel { get; set; }
1 referência
29 public Cambios Cambio { get; set; }
30
31 // void ICadastro.Gravar()
32 // {
33 //     throw new NotImplementedException();
34 // }
1 referência
35 public void Gravar()
36 {
37     //definir a linha de titulo do arquivo
38     string titulo = @"PLACA;MODELO;COR;ANO;MARCA;COMBUSTIVEL;CÂMBIO;
39     OBSERVAÇÃO;USUÁRIO;DATA;HORÁRIO";
40     //definir o registro que será gravado
41     string registro = string.Format(
42         "{0};{1};{2};{3};{4};{5};{6};{7};{8};{9}",
43         Placa, Modelo, Cor, Ano.ToString(),
44         Marca, Combustivel.ToString(),
45         Cambio.ToString(), Observacao.Replace(";", ""),
46         Environment.UserName,
47         DateTime.Now.ToString("dd/MM/yyyy;HH:mm:ss"));
48     //Verificar se o caminho ja existe
49     bool flag = File.Exists(caminho);
50
51     //definir o objeto StreamWriter
52     StreamWriter escritor = new StreamWriter(caminho, true);
53
54     //Se o arquivo nao existir, cria o arquivo e
55     //escreve o titulo na primeira linha
56     if (!flag)
57     {
58         escritor.WriteLine(titulo);
59     }
60     //Escreve o registro no arquivo
61     escritor.WriteLine(registro);
62     //fecha a comunicação com o arquivo
63     escritor.Close();
64 }
65 }
66 }
67
100% Não foi encontrado nenhum problema
```

Voltamos ao formulário, no evento Load do form, vamos colocar o carregamento da combobox

```
8 using System.Threading.Tasks;
9 using System.Windows.Forms;
10
11 namespace CadastroVeiculos
12 {
13     3 referências
14     public partial class CadastroVeiculos : Form
15     {
16         1 referência
17         public CadastroVeiculos()
18         {
19             InitializeComponent();
20         }
21         1 referência
22         private void CadastroVeiculos_Load(object sender, EventArgs e)
23         {
24             //Carregar combobox de combustivel
25             cbocombustivel.DataSource = Enum.GetNames(typeof(Combustiveis));
26             //Desmarcar a combo
27             cbocombustivel.SelectedIndex = -1;
28         }
29     }
30 }
```

Resultado:



Agora faremos para as outras comboBox

```
//carregar combobox cambio
cbocambio.DataSource = Enum.GetNames(typeof(Cambios));
//Desmarcar a combo
cbocambio.SelectedIndex = -1;

// carregar combobox Marca
//limpa dados que possam estar na memoria...
cbomarca.Items.Clear();
//Utilizar o endereço onde o arquivo Marcas.txt foi salvo
System.IO.StreamReader leitor =
    new System.IO.StreamReader(
        @"C:\Users\Valéria Moraes\OneDrive - Etec Centro Paula Souza\Escola\ETEC\Aulas 2020\Aulas de Junho Cadastro de Veiculos\Marcas.txt");
while (!leitor.EndOfStream)
{
    cbomarca.Items.Add(leitor.ReadLine());
}
leitor.Close();
}
```

O código finalmente fica assim. Salve e execute sem erros

```
namespace CadastroVeiculos
{
    public partial class CadastroVeiculos : Form
    {
        public CadastroVeiculos()
        {
            InitializeComponent();
        }

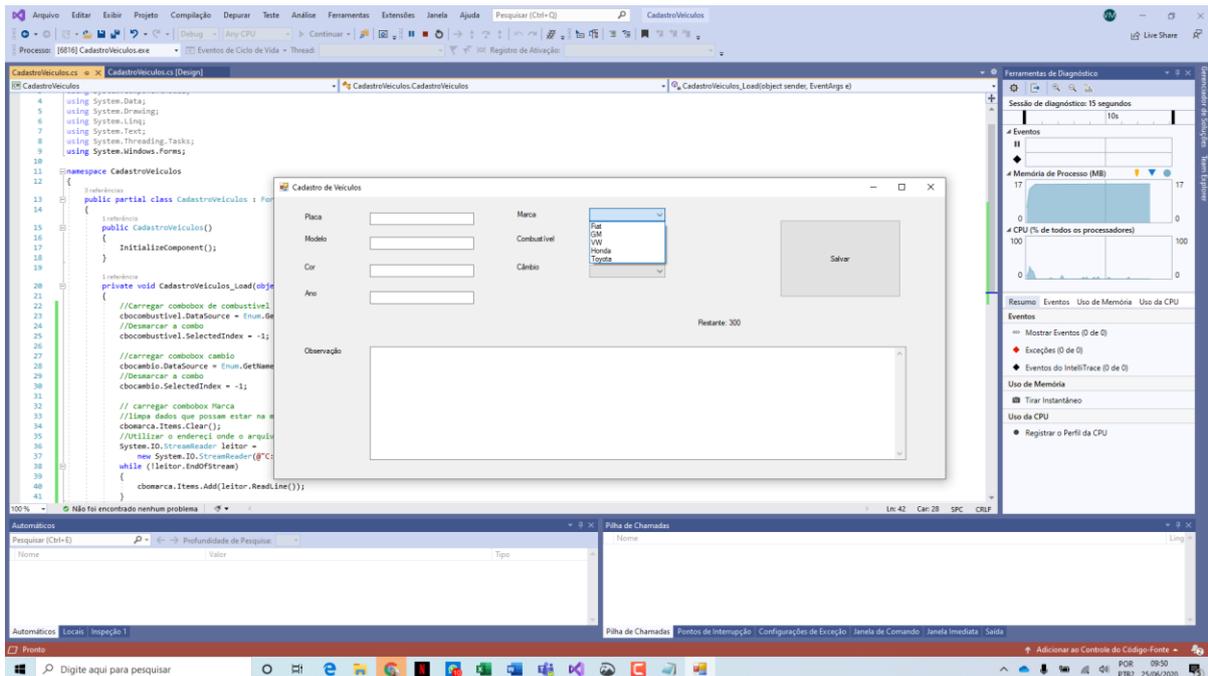
        private void CadastroVeiculos_Load(object sender, EventArgs e)
        {
            //Carregar combobox de combustivel
            cbcombustivel.DataSource = Enum.GetNames(typeof(Combustiveis));
            //Desmarcar a combo
            cbcombustivel.SelectedIndex = -1;

            //carregar combobox cambio
            cbocambio.DataSource = Enum.GetNames(typeof(Cambios));
            //Desmarcar a combo
            cbocambio.SelectedIndex = -1;

            // carregar combobox Marca
            //limpa dados que possam estar na memoria...
            cbomarca.Items.Clear();
            //Utilizar o endereço onde o arquivo Marcas.txt foi salvo
            System.IO.StreamReader leitor =
                new System.IO.StreamReader(
                    @"C:\Users\Valéria Moraes\OneDrive - Etec Centro Paula Souza\Escola\ETEC\Aulas 2020\Aulas de Junho Cadastro de Veiculos\Marcas.txt");
            while (!leitor.EndOfStream)
            {
                cbomarca.Items.Add(leitor.ReadLine());
            }
            leitor.Close();
        }
    }
}
```

A combobox Marca é carregada a partir do documento Marcas.txt que criamos no início do projeto. O endereço do arquivo deve estar plenamente correto, senão vai dar erro na leitura e carregamento dos dados.

Isso permite que mais marcas possam ser inseridas na combobox, sem a necessidade de alterar a programação do projeto no Visual Studio,



Agora vamos codificar o botão Gravar, é um duplo clique nele e escreva o código abaixo:

```
43         leitor.Close();
44     }
45 }
46
47 1 referência
48 private void btngravar_Click(object sender, EventArgs e)
49 {
50     try
51     {
52         Veiculo carro = new Veiculo(
53             @"C:\Users\Valéria Moraes\OneDrive - Etec Centro Paula Souza\Escola\ETEC\Aulas 2020\Aulas de Junho Cadastro de Veiculos\CadastroVeiculos.txt");
54         // Atribuir valor as propriedades veiculo
55         // com validações
56         carro.Placa = placatxt.ValidarVazio().ValidarPlaca();
57         carro.Modelo = txtmodelo.ValidarVazio();
58         carro.Cor = txtcor.ValidarVazio();
59         //carro.Ano = txtano.ValidarInt16().ValidarAno();
60         carro.Marca = cbomarca.ValidarCombo();
61         carro.Observacao = txtobservacao.Text;
62         carro.Combustivel = (Combustiveis)cbocombustivel.ValidarComboEnum();
63         carro.Cambio = (Cambios)cbocambio.ValidarComboEnum();
64
65         //chamar o método gravar veiculo
66         carro.Gravar();
67         //avisar o usuario
68         MessageBox.Show("Veiculo gravado com sucesso", "Aviso",
69             MessageBoxButtons.OK, MessageBoxIcon.Information);
70         //redefinir a tela
71         this.LimparTela();
72         placatxt.Focus();
73     }
74     catch (Exception ex)
75     {
76         MessageBox.Show(ex.Message, "Alerta de Erro",
77             MessageBoxButtons.OK, MessageBoxIcon.Error);
78     }
79 }
80
81 1 referência
82 private void txtobservacao_TextChanged(object sender, EventArgs e)
```

Agora para acompanhar a quantidade de toques no campo observação, vamos escrever o código.

De um duplo clique na txtobservacao e escreva o código

```
        MessageBox.Show(ex.Message, "Atenção de Erro",
            MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

1 referência
private void txtobservacao_TextChanged(object sender, EventArgs e)
{
    try
    {
        lbltoques.Text = "Restante: " +
            (300 - txtobservacao.TextLength).ToString();
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message, "Alerta de Erro",
            MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
```

Agora para a Tecla ESC funcionar no projeto, precisa alterar a propriedade KEYPREVIEW do FORMULARIO para TRUE

Em seguida , ainda no formulário, vá para a guia EVENTS (o ícone de relâmpago), e na janela das propriedades do evento selecione e de um duplo clique no evento Key-press e coloco o código abaixo.

```
        MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}

1 referência
private void CadastroVeiculos_KeyPress(object sender, KeyPressEventArgs e)
{
    //Passar a propriedade Keypreview do formulario para True
    if (e.KeyChar == 27)
    {
        this.LimparTela();
    }
}
}
```