

# **Desenvolvimento de Software para Internet DSWI**

## **Aula Prática**

## Sumário

Aula 02 - Criar uma aplicação Web ASP.NET com .NET Framework, utilizando C#.	3
Introdução	3
Tela inicial do Visual Studio 2019.	4
Definindo o tipo de projeto.	4
Tipo da Aplicação Web	5
Criando o primeiro documento HTML	6
Adicionando conteúdo no documento HTML	10
Executando a aplicação no servidor IIS local.	11
Inserindo mais conteúdo no documento HTML.	13
Adicionando outro documento HTML na aplicação	15
Criando uma pasta no projeto	19
Adicionar um arquivo na pasta	20
Inserindo a tag <img>	21
Anexo A	26

## **Aula 02 - Criar uma aplicação Web ASP.NET com .NET Framework, utilizando C#.**

### **Introdução**

Neste documento apresento de forma bem sucinta o passo a passo com print de telas para a criação da primeira aplicação Web utilizando ASP.Net com linguagem C#.

Utilizo o Visual Studio Community 2019, que é uma IDE gratuita e você poderá baixar do site da Microsoft (<https://visualstudio.microsoft.com/pt-br/vs/>).

Configurei a IDE para o idioma em Inglês.

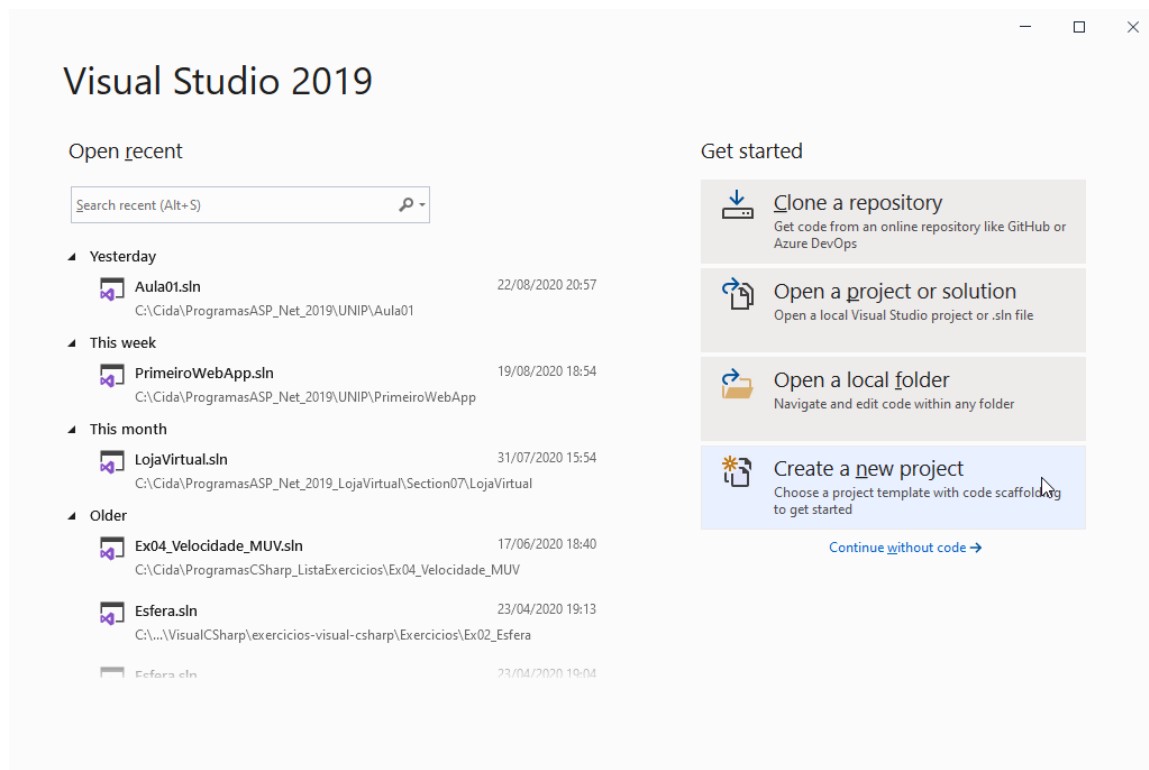
Você poderá acessar o código fonte das aulas no github pelo link:

<https://github.com/cidacastello/DSWIASPNet>

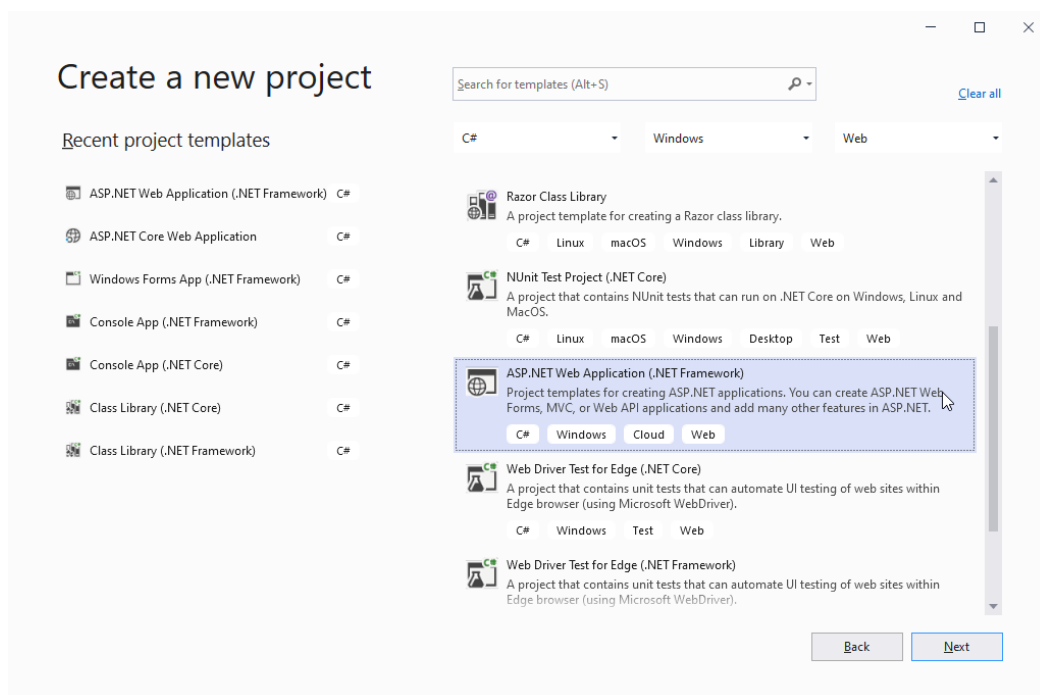
Neste documento, em anexos, você encontra print de tela do Visual Studio 2010 mas sem texto explicativo.

Tela inicial do Visual Studio 2019.

Selecione a opção Criar um novo projeto.



Definindo o tipo de projeto.



Nesta tela você define o nome do projeto que é o nome do site e o local onde será criado.

A *solution* é um container para um ou mais projetos.

**Configure your new project**

ASP.NET Web Application (.NET Framework) C# Windows Cloud Web

Project name  
Aula02\_HTML\_CSS\_Pratica

Location  
C:\Cida\ProgramasASP\_Net\_2019\UNIP\

Solution name ⓘ  
Aula02\_HTML\_CSS\_Pratica

☐ Place solution and project in the same directory

Framework  
.NET Framework 4.7.2

Back Create

## Tipo da Aplicação Web

Nesta aula vamos criar uma Aplicação Web ASP.NET Vazia.

**Create a new ASP.NET Web Application**

**Empty**  
An empty project template for creating ASP.NET applications. This template does not have any content in it.

**Web Forms**  
A project template for creating ASP.NET Web Forms applications. ASP.NET Web Forms lets you build dynamic websites using a familiar drag-and-drop, event-driven model. A design surface and hundreds of controls and components let you rapidly build sophisticated, powerful UI-driven sites with data access.

**MVC**  
A project template for creating ASP.NET MVC applications. ASP.NET MVC allows you to build applications using the Model-View-Controller architecture. ASP.NET MVC includes many features that enable fast, test-driven development for creating applications that use the latest standards.

**Web API**  
A project template for creating RESTful HTTP services that can reach a broad range of clients including browsers and mobile devices.

**Single Page Application**  
A project template for creating rich client side JavaScript driven HTML5 applications using ASP.NET Web API. Single Page Applications provide a rich user experience which includes client-side interactions using HTML5, CSS3, and JavaScript.

**Authentication**  
No Authentication  
Change

**Add folders & core references**

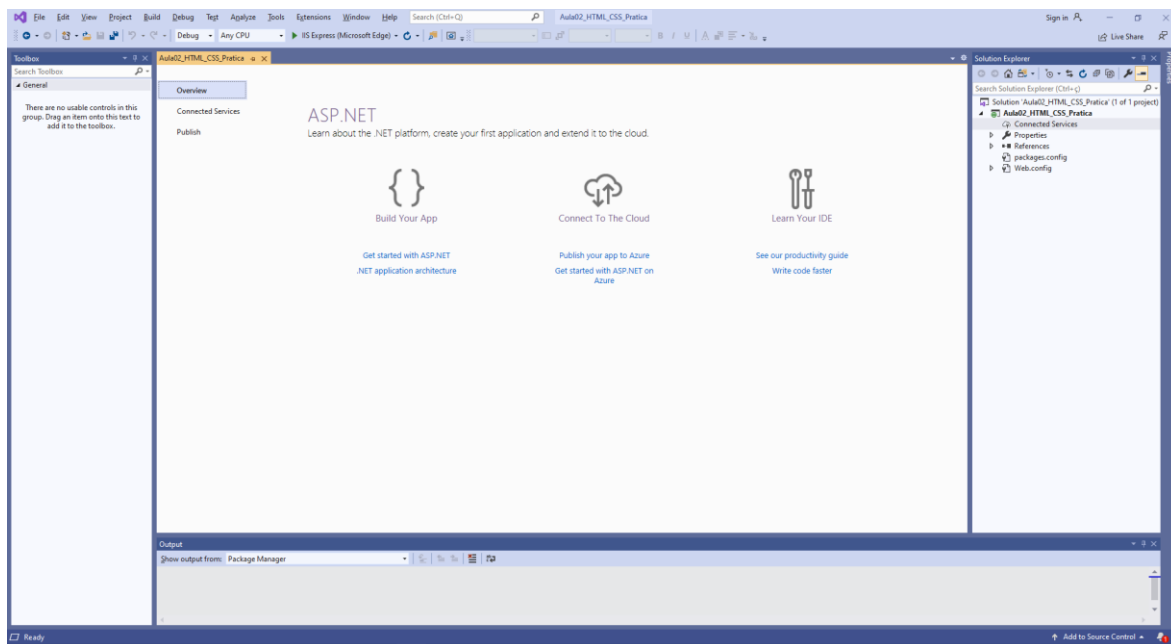
☐ Web Forms  
☐ MVC  
☐ Web API

**Advanced**

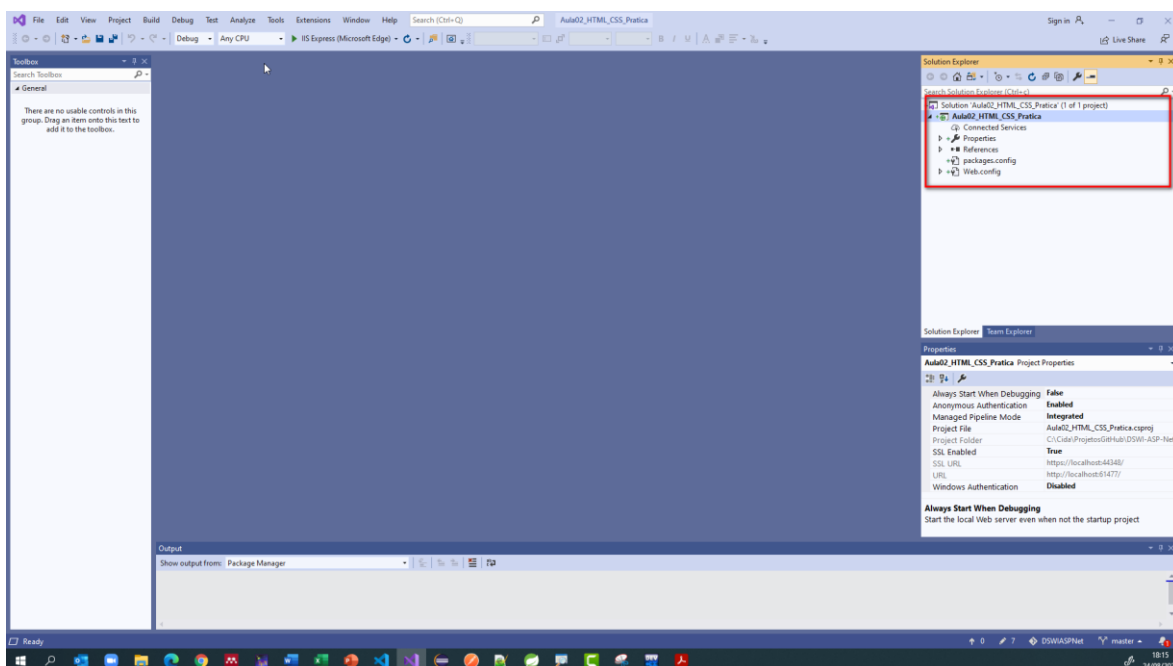
☒ Configure for HTTPS  
☐ Docker support  
(Requires [Docker Desktop](#))  
☐ Also create a project for unit tests  
Aula02\_HTML\_CSS\_Pratica.Tests

Back Create

Esta é a tela inicial do Visual Studio 2019 após criar a solução.



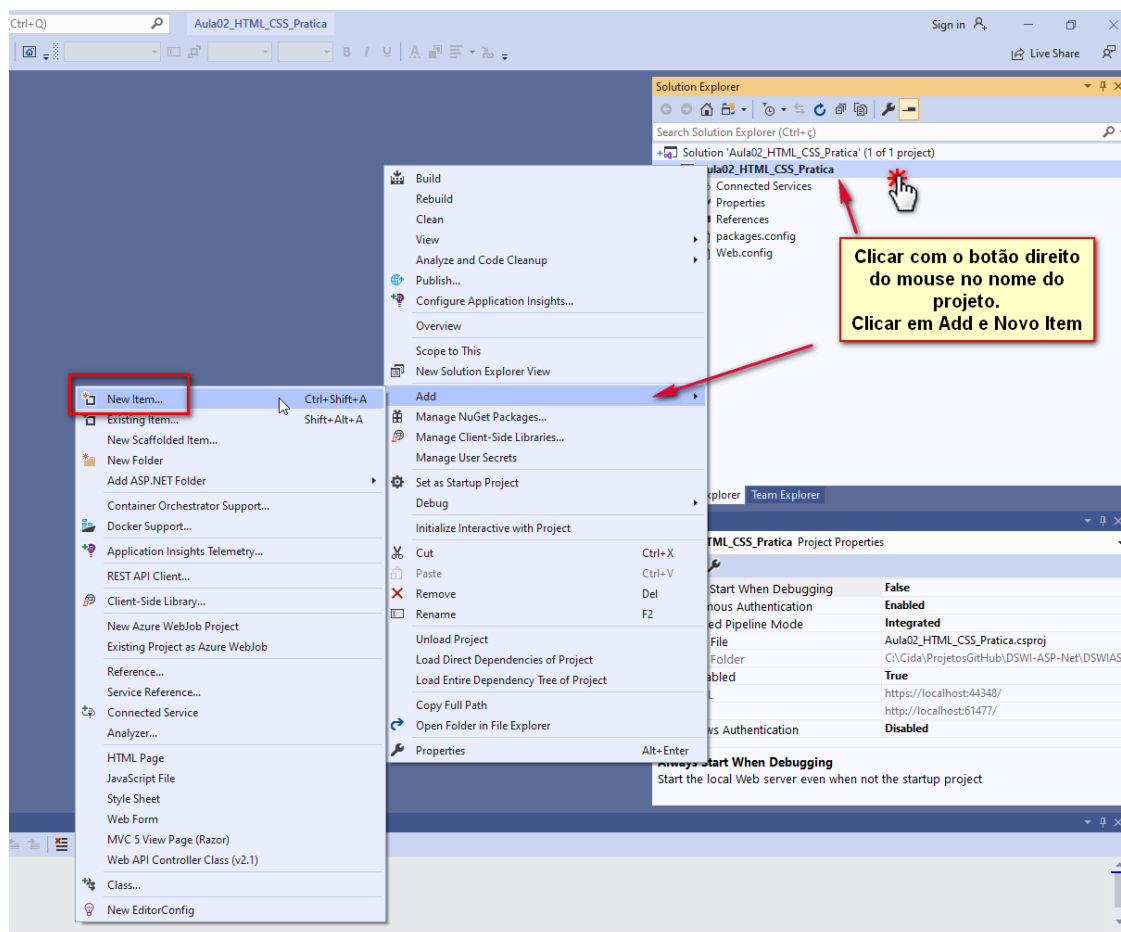
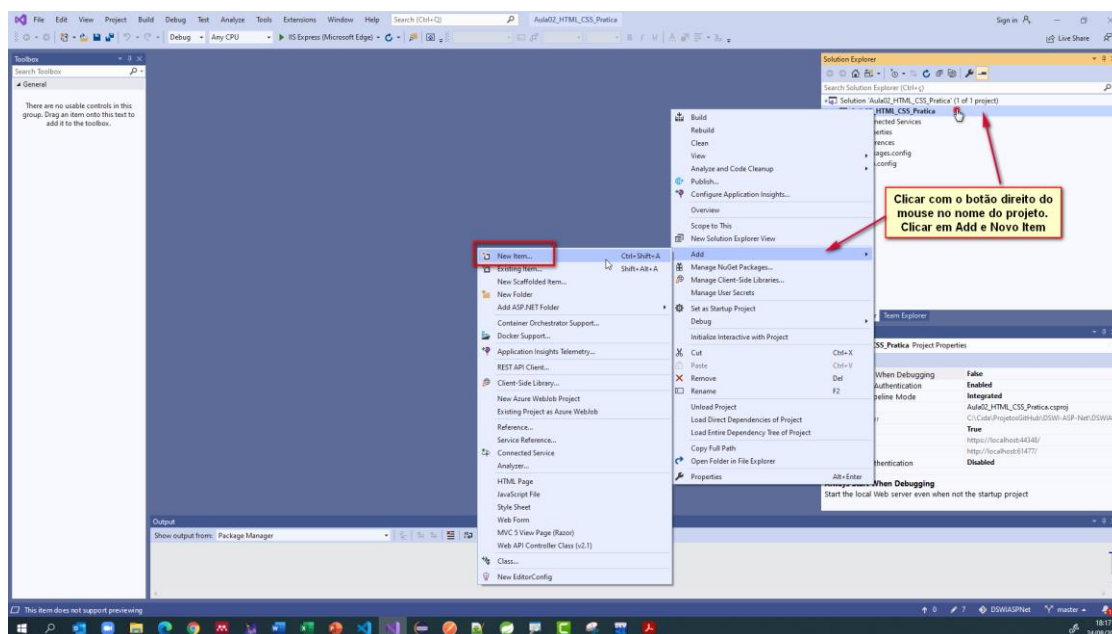
Na janela *Solution Explorer*, temos a solução e o projeto criado com arquivos e bibliotecas.



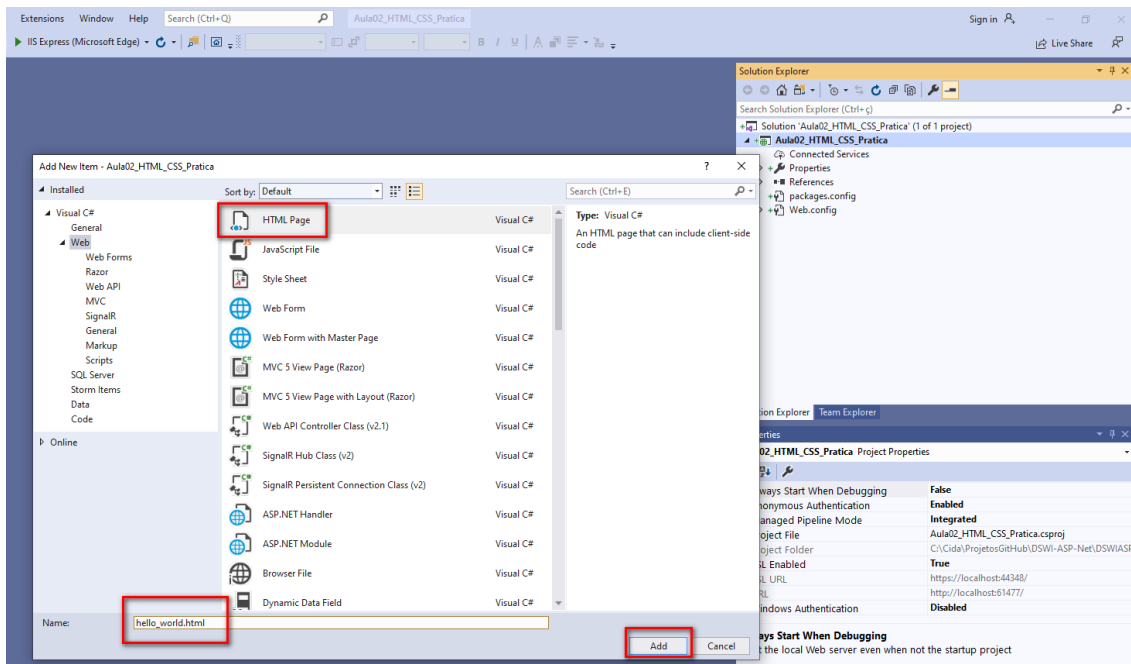
Criando o primeiro documento HTML

Agora vamos criar o primeiro documento HTML conforme ilustrado.

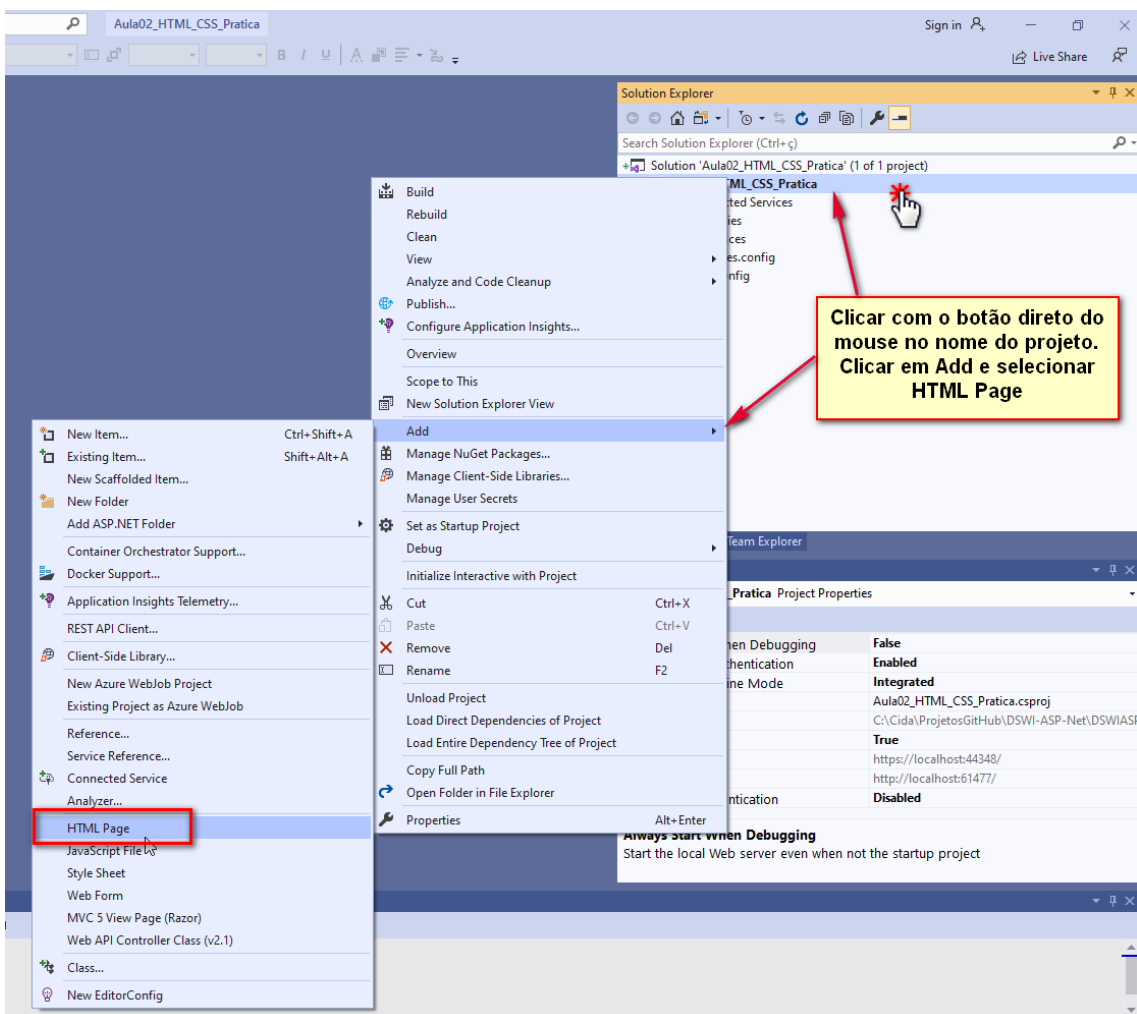
Para isso vamos criar um novo item, conforme sequência de telas.



Selecionamos o item , ou seja, página HTML.

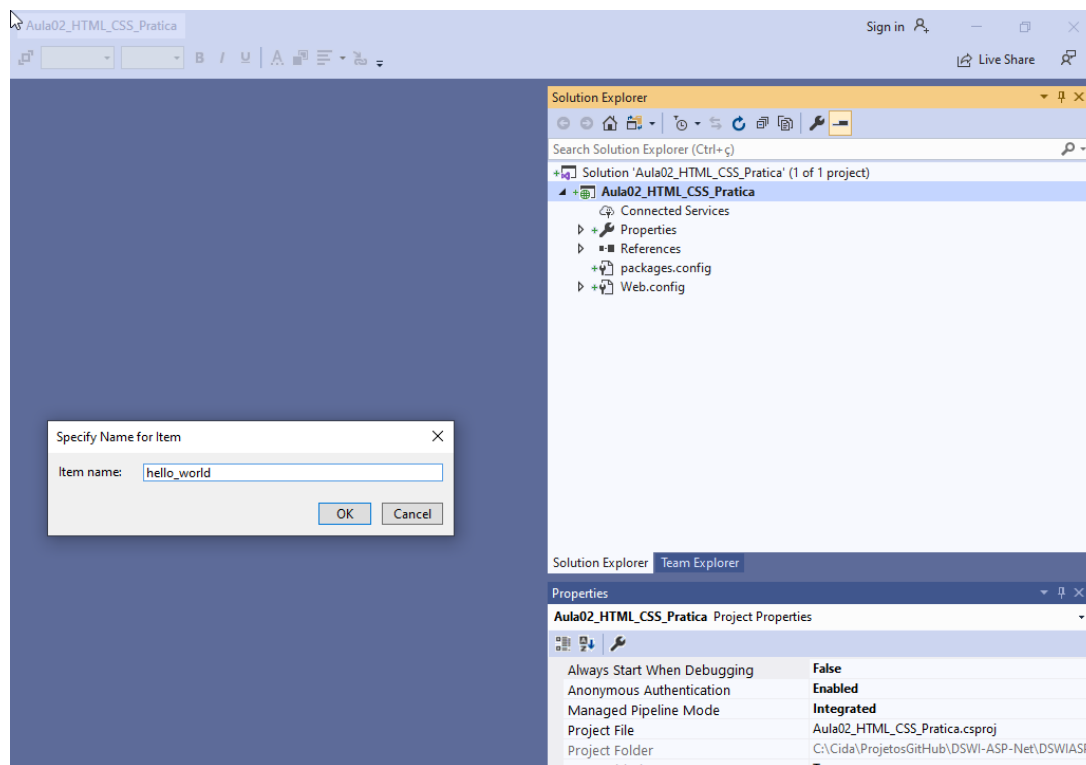


Esse passo pode ser feito de uma forma mais rápida, forma, conforme apresentado nas sequência de telas a seguir.

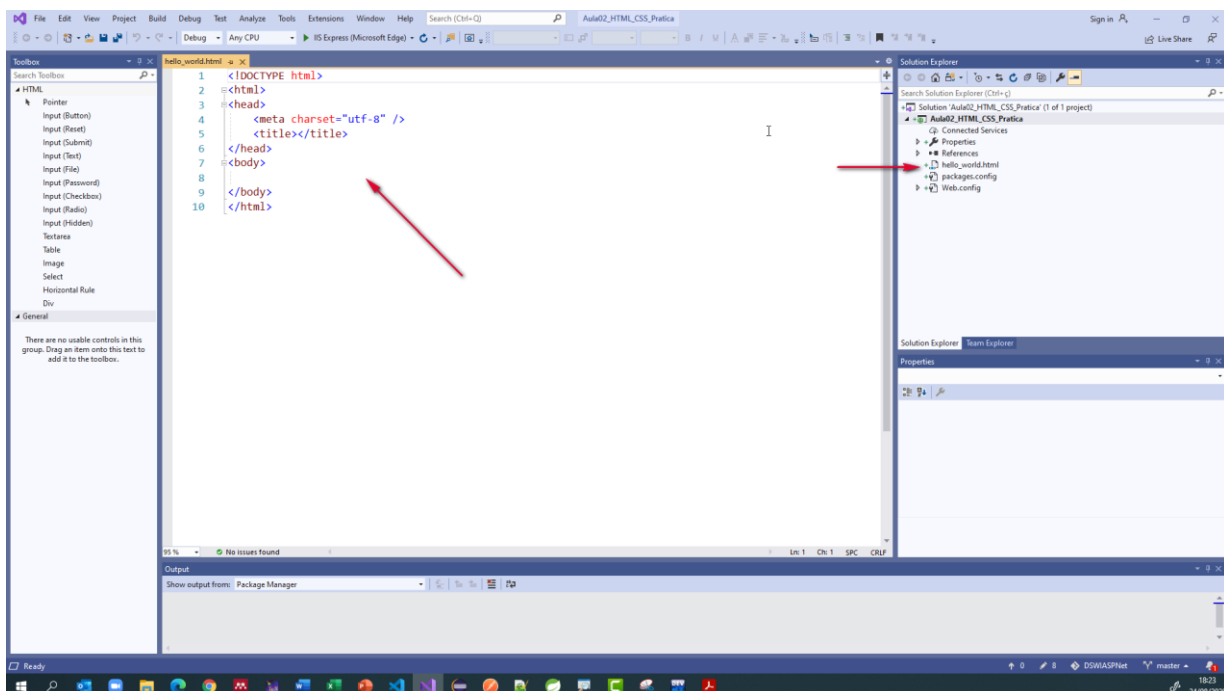




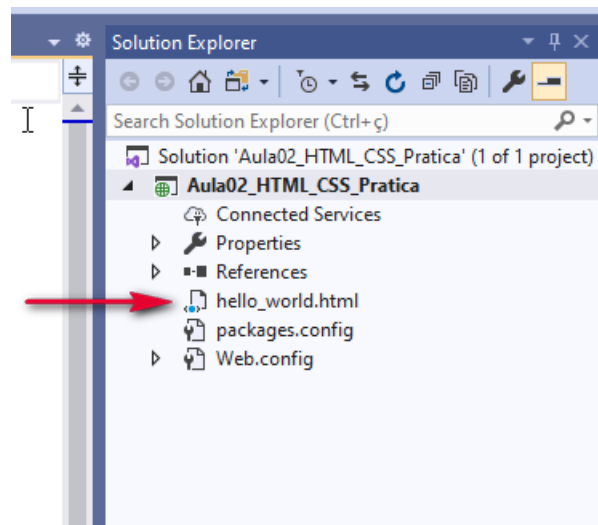
Você pode observar que tem uma caixa de diálogo para você digitar o nome do arquivo, ou da página.



O documento HTML foi criado e já vem previamente preenchido com a estrutura básica de um documento HTML5.



Veja o detalhe da janela *Solution Explorer* com o arquivo `hello_world.html`



Adicionando conteúdo no documento HTML

Agora vamos colocar algumas tags no conteúdo no nosso documento HTML conforme a figura.

A marcação `<!-- Isto é um comentário -->` é um comentário e não é interpretado pelo navegador. Sempre que achar necessário, faça comentários no código. Isso ajudará a dar manutenção.

```

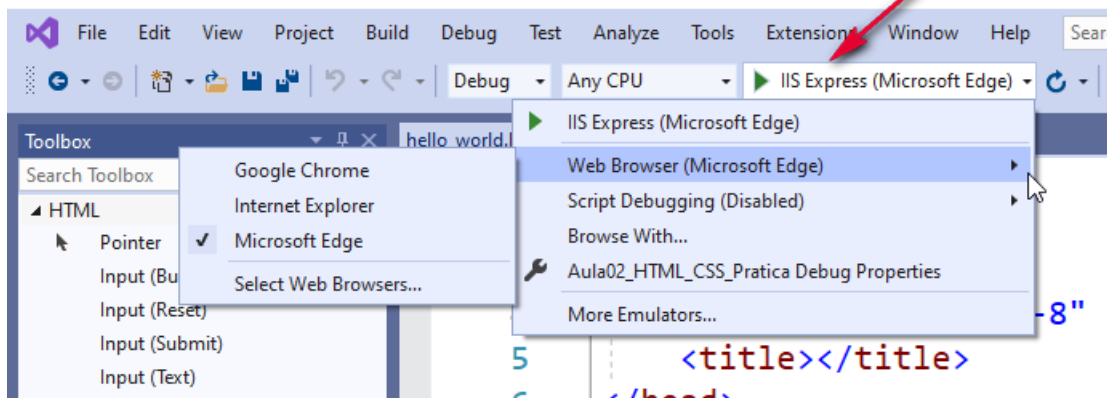
1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="utf-8" />
5      <title>Hello World</title>
6  </head>
7  <body>
8      <!--h1 é um cabeçalho de Nível 1-->
9      <h1>Meu primeiro HTML</h1>
10
11     <!--h2 é um cabeçalho de Nível 2-->
12     <h2>Testando com o famoso "Hello World!"</h2>
13
14     <!--p é a tag para um parágrafo-->
15     <p>Está tudo funcionando</p>
16 </body>
17 </html>

```

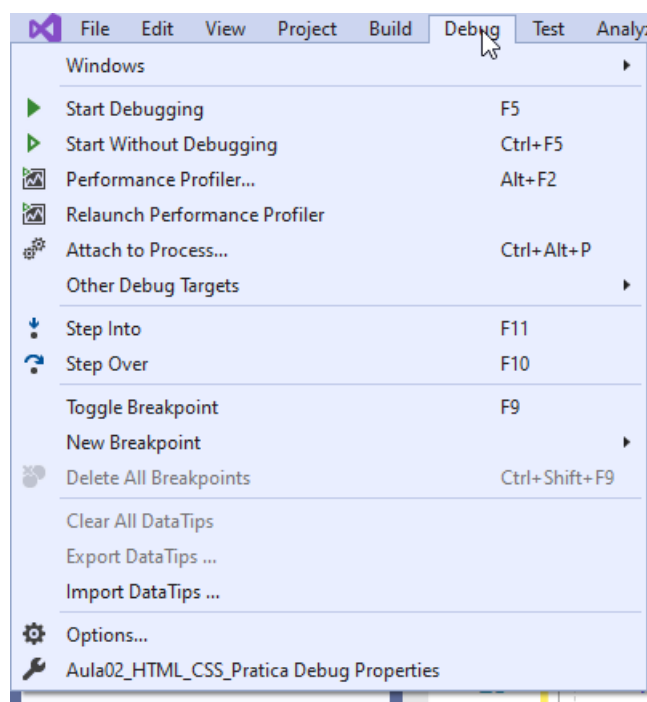
Executando a aplicação no servidor IIS local.

Quando você executa a aplicação, clicando no botão play ou pressionando “F5”, o servidor IIS é iniciado e a página é renderizada no navegador.

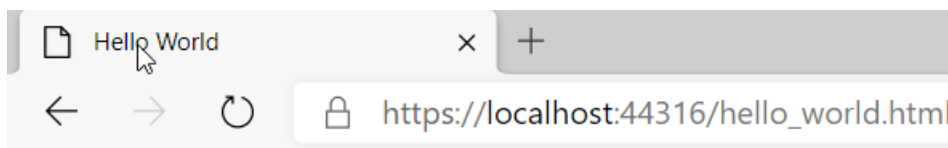
**Para que a página seja apresentada (ou executar o programa) é preciso "publicar" na internet.**



Também podemos iniciar a aplicação pelo menu *Debug*.



Veja como foi renderizado o documento HTML no navegador. O Visual Studio abre o navegador automaticamente.



# Meu primeiro HTML

## Testando com o famoso "Hello World!"

Está tudo funcionando

Clique no botão *stop* (parar) para parar a aplicação pelo Visual Studio ou ainda se você fechar o navegador a aplicação também para..

**Para parar a execução ou clicar no botão "stop" ou fechar o navegador**



Ao receber uma página, o navegador possui uma memória denominada cache que é verificada se ele já foi visitada.

Caso positivo, é apresentada a página local e o navegador não recebe a página do servidor.

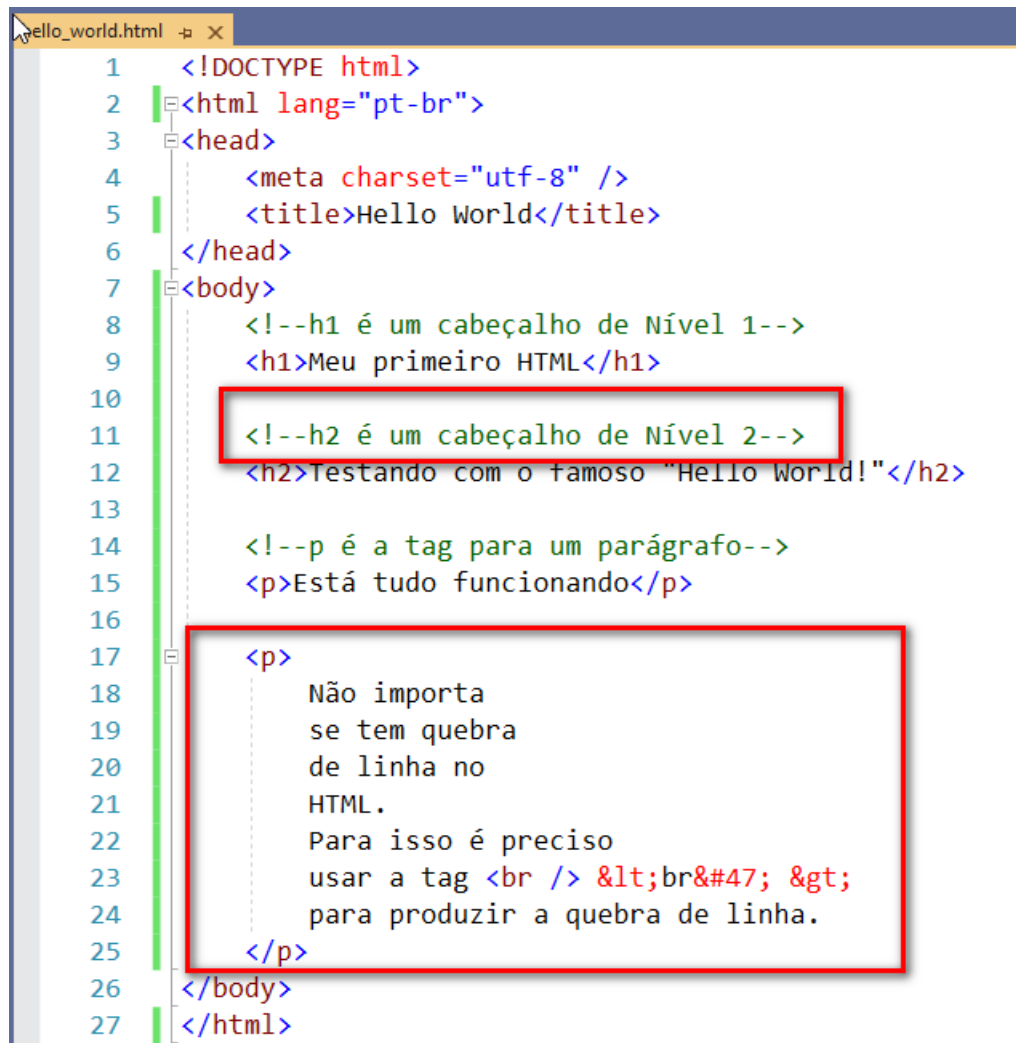
Isso é feito para economizar tráfego na rede.

Para desenvolvimento isso é ruim pois há o risco de uma nova página não ser apresentada!

Para contornar isso, abrir a página com Ctrl+F5 para recarregar a página.

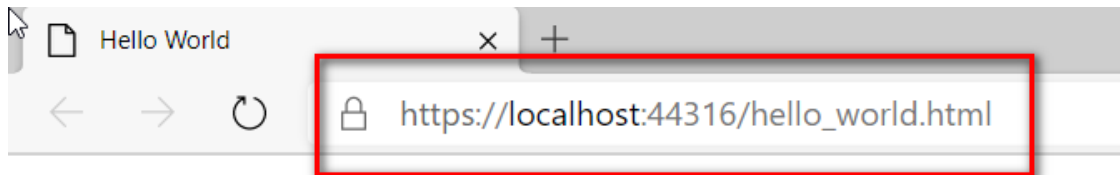
Inserindo mais conteúdo no documento HTML.

As quebras de linhas no documento HTML não são reconhecidas, para quebrar linha devemos utilizar a tag `<br />`



```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3 <head>
4 <meta charset="utf-8" />
5 <title>Hello World</title>
6 </head>
7 <body>
8 <!--h1 é um cabeçalho de Nível 1-->
9 <h1>Meu primeiro HTML</h1>
10
11 <!--h2 é um cabeçalho de Nível 2-->
12 <h2>Testando com o famoso "Hello World!"</h2>
13
14 <!--p é a tag para um parágrafo-->
15 <p>Está tudo funcionando</p>
16
17 <p>
18 Não importa
19 se tem quebra
20 de linha no
21 HTML.
22 Para isso é preciso
23 usar a tag <br /> &lt;br&#47; &gt;
24 para produzir a quebra de linha.
25 </p>
26 </body>
27 </html>
```

Executando novamente a aplicação com os novos conteúdos adicionados no documento.



# Meu primeiro HTML

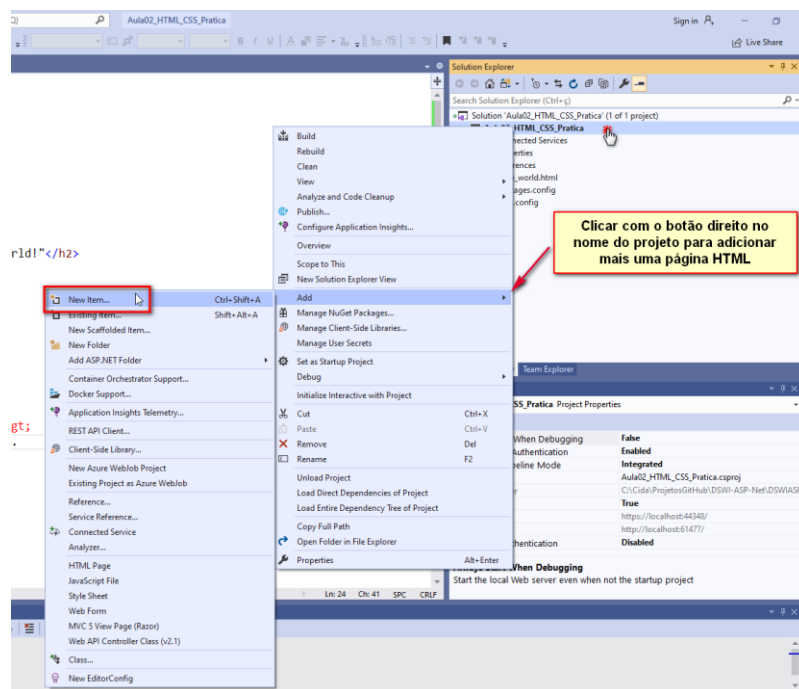
## Testando com o famoso "Hello World!"

Está tudo funcionando

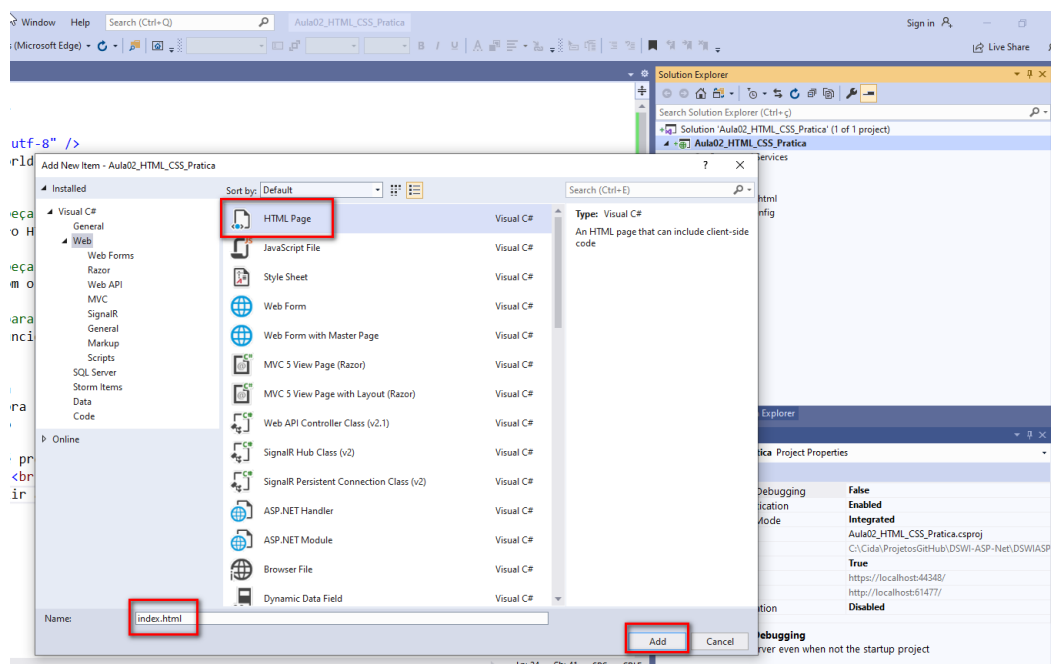
Não importa se tem quebra de linha no HTML. Para isso é preciso usar a tag `<br />` para produzir a quebra de linha.

Adicionando outro documento HTML na aplicação

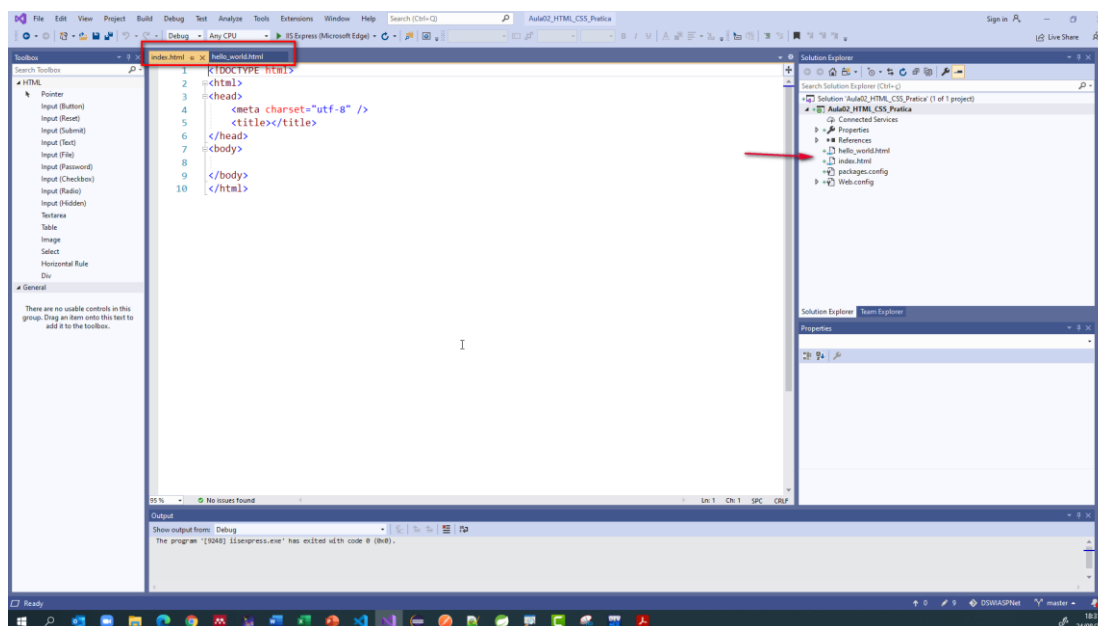
Vamos agora adicionar mais um documento HTML na nossa aplicação Web, criando o arquivo index.html.



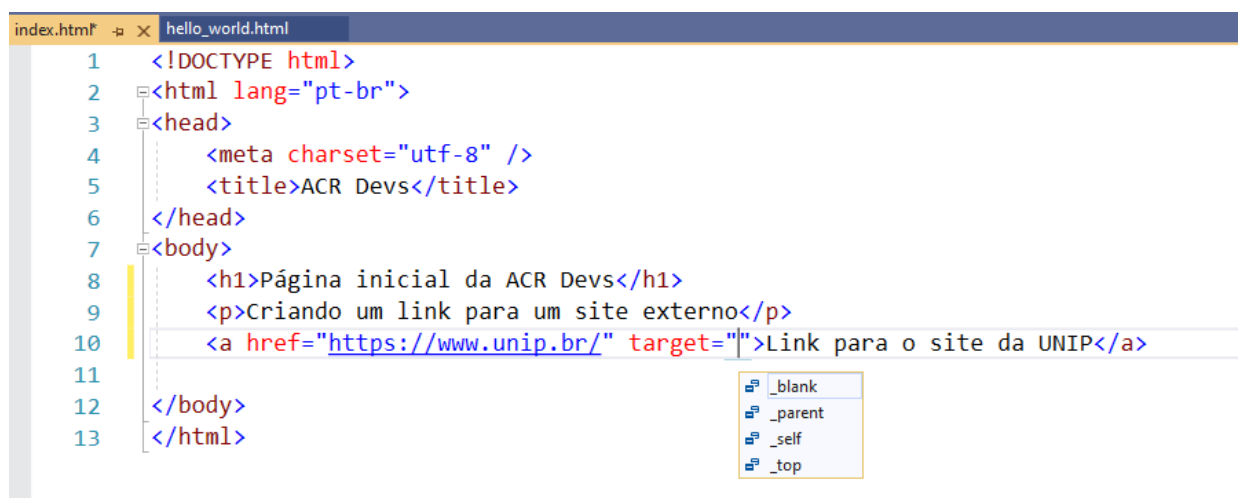
Vamos seguir os mesmos passos que fizemos anteriormente.



Podemos observar os dois arquivos na janela do *Solution Explorer*.

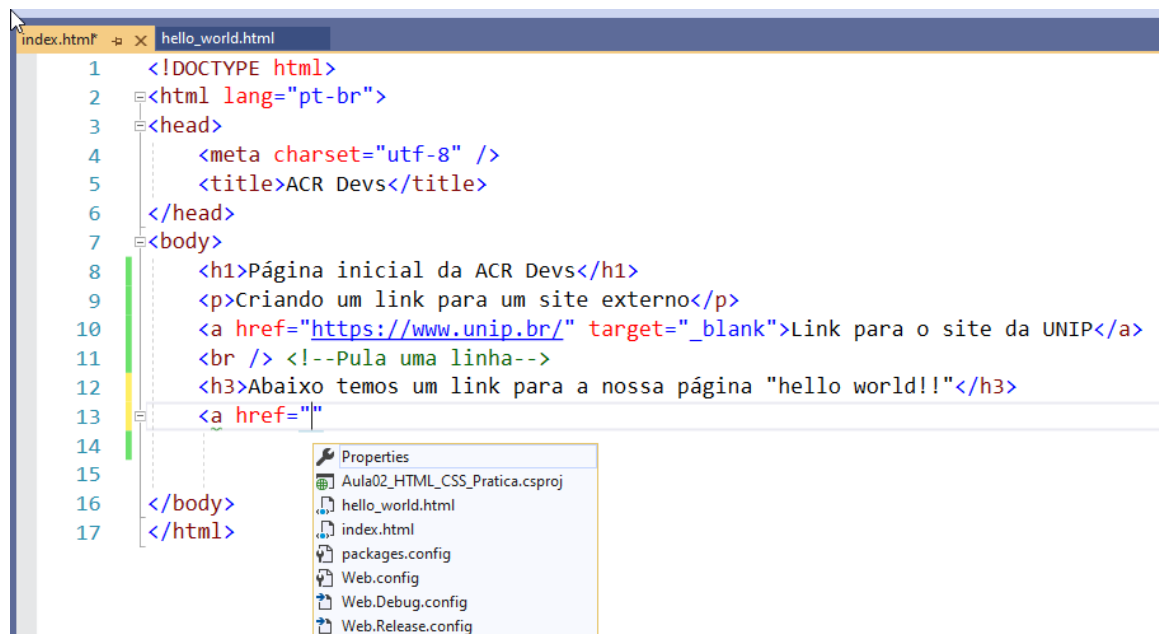


Vamos inserir novo conteúdo no arquivo index.html



Veja que o Visual Studio ajuda a completar o código. Isso se dá por causa do recurso *IntelliSense*.





```

1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="utf-8" />
5      <title>ACR Devs</title>
6  </head>
7  <body>
8      <h1>Página inicial da ACR Devs</h1>
9      <p>Criando um link para um site externo</p>
10     <a href="https://www.unip.br/" target="_blank">Link para o site da UNIP</a>
11     <br /> <!--Pula uma linha-->
12     <h3>Abaixo temos um link para a nossa página "hello world!!"</h3>
13     <a href=""
14
15
16 </body>
17 </html>

```

Properties

- Aula02\_HTML\_CSS\_Pratica.csproj
- hello\_world.html
- index.html
- packages.config
- Web.config
- Web.Debug.config
- Web.Release.config

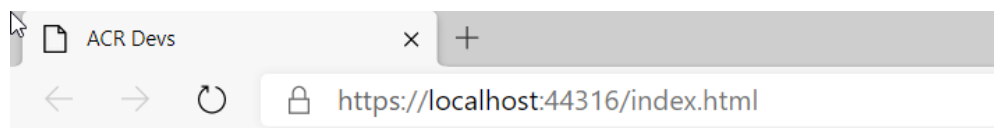


```

1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="utf-8" />
5      <title>ACR Devs</title>
6  </head>
7  <body>
8      <h1>Página inicial da ACR Devs</h1>
9      <p>Criando um link para um site externo</p>
10     <a href="https://www.unip.br/" target="_blank">Link para o site da UNIP</a>
11     <br /> <!--Pula uma linha-->
12     <h3>Abaixo temos um link para a nossa página "hello world!!"</h3>
13     <a href="hello_world.html">Linka para a página Hello World</a>
14
15
16 </body>
17 </html>

```

Vamos executar a aplicação com a nova página criada.



## Página inicial da ACR Devs

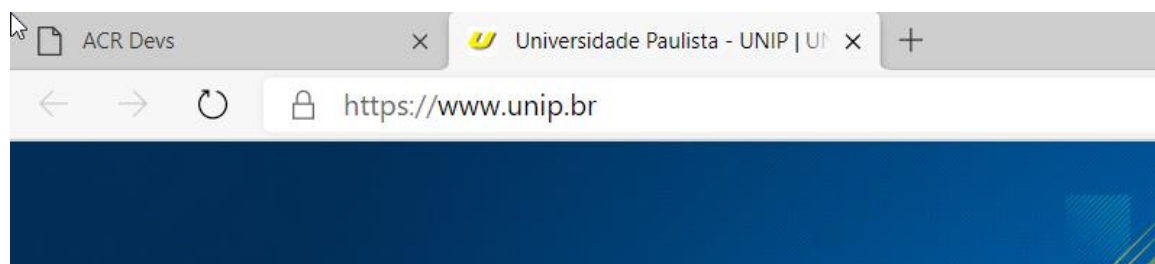
Criando um link para um site externo

[Link para o site da UNIP](#)

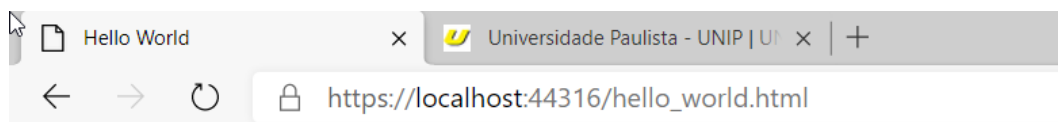
**Abaixo temos um link para a nossa página "hello world!!"**

[Linka para a página Hello World](#)

Veja a Aba do navegador quando clicamos no link inserido no documento para um site externo com a propriedade target.



Esta é a página quando clicamos no link para abrir o documento interno do nosso site.



## Meu primeiro HTML

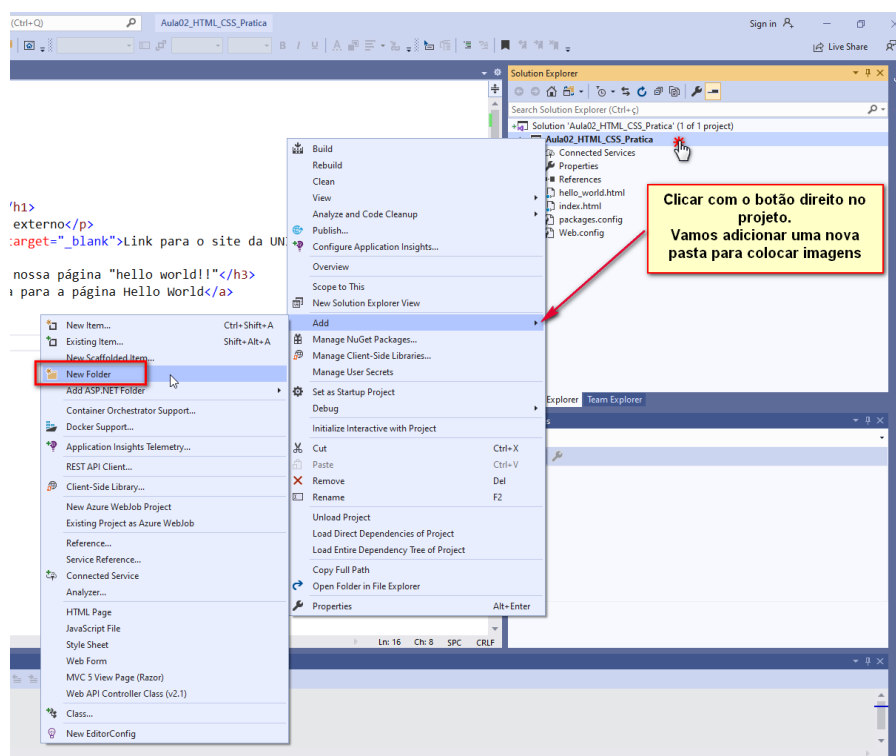
### Testando com o famoso "Hello World!"

Está tudo funcionando

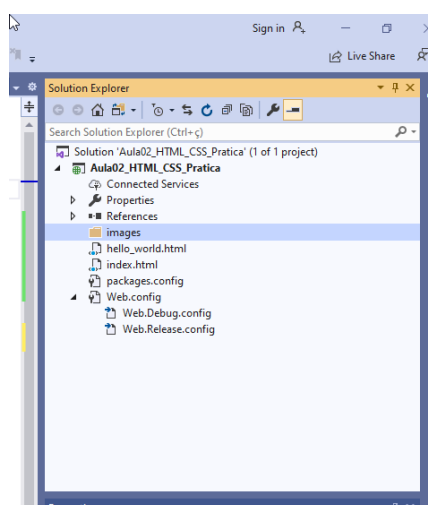
Não importa se tem quebra de linha no HTML. Para isso é preciso usar a tag `<br />` para produzir a quebra de linha.

Criando uma pasta no projeto

Vamos agora criar uma pasta (*folder*) com o nome *images* onde iremos colocar uma imagem para ser exibida na nossa página index.

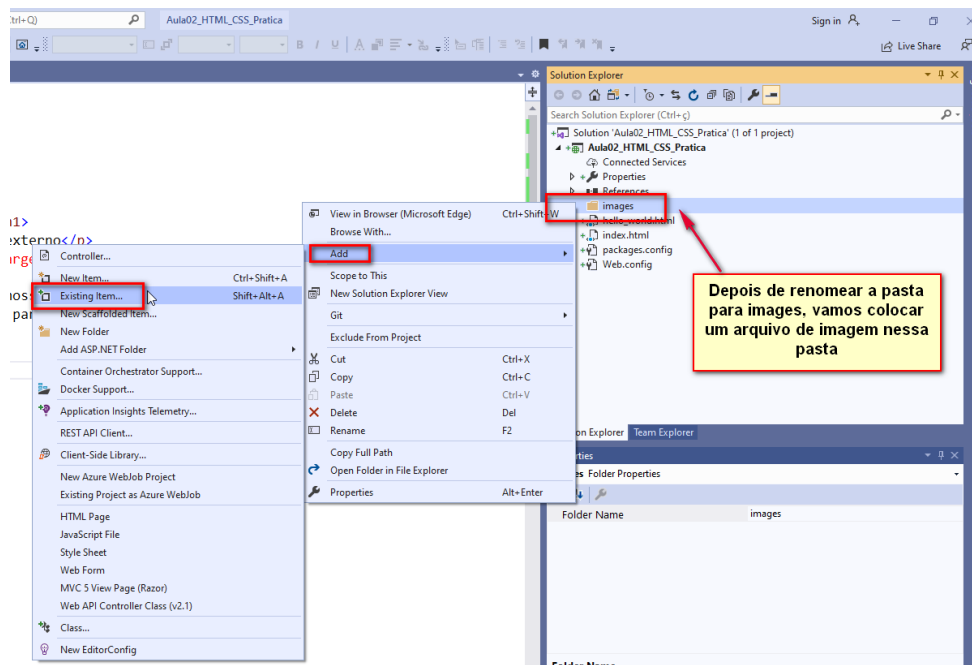


Veja o detalhe da janela *Solution explorer* depois de criar e renomear a pasta para *images*.

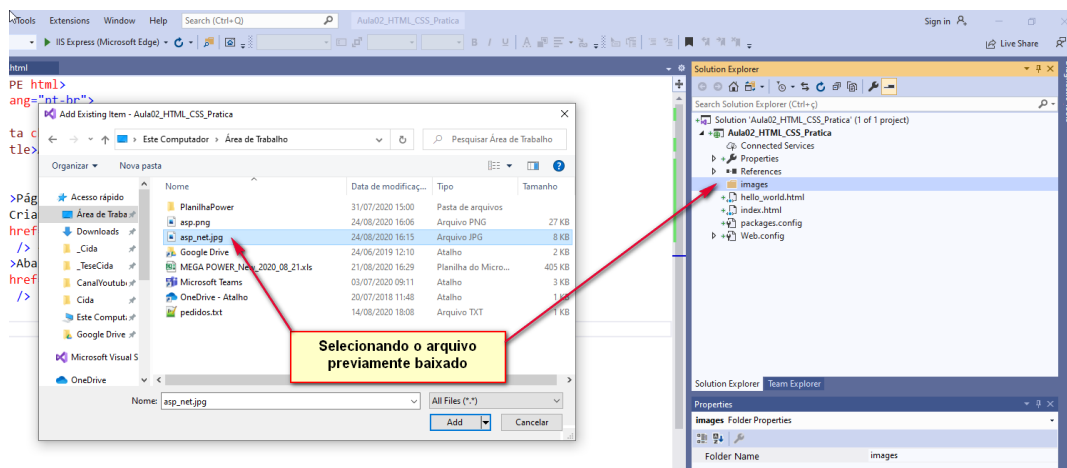


Adicionar um arquivo na pasta

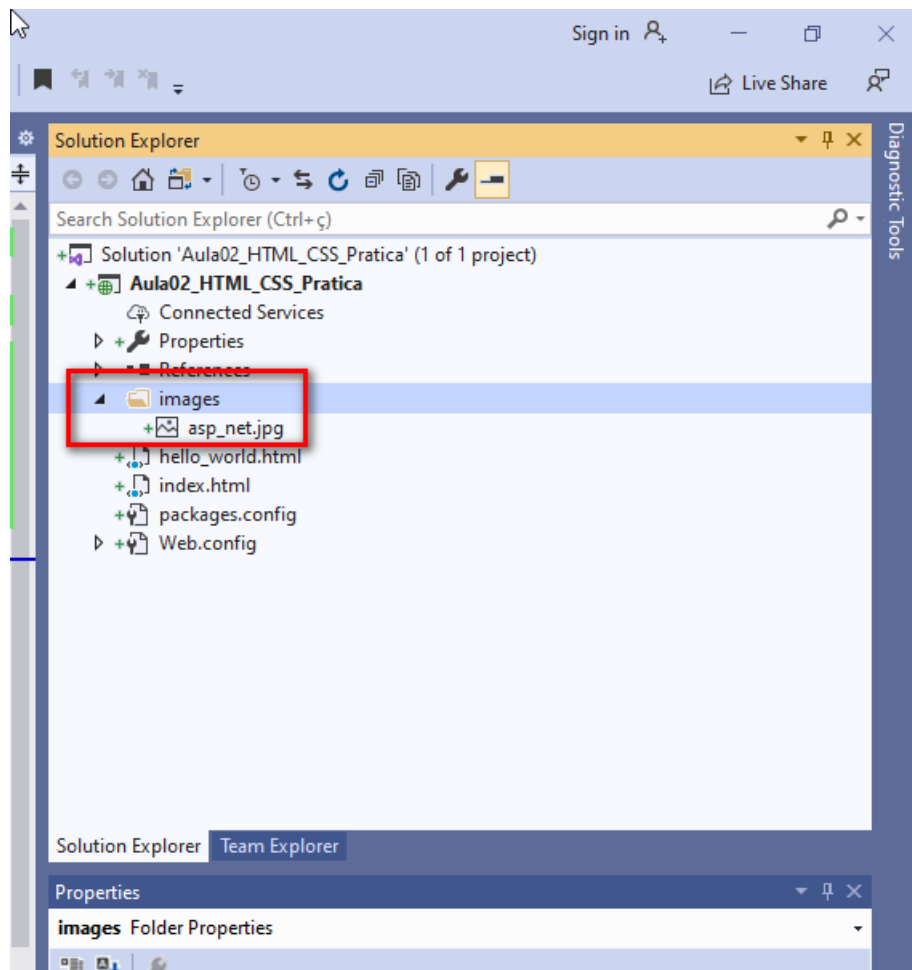
Adicionando um arquivo de imagem na pasta *images*.



Selecione um arquivo de imagem.



Veja os detalhes da janela *Solution explorer* depois de adicionar o arquivo.



Inserindo a tag <img>

Agora vamos inserir a tag **<img>** no documento HTML para que a imagem possa ser localizada e renderizada na página pelo navegador. Veja que o Visual Studio ajuda completando o código.

```

index.htm* x hello_world.html
1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="utf-8" />
5      <title>ACR Devs</title>
6  </head>
7  <body>
8      <h1>Página inicial da ACR Devs</h1>
9      <p>Criando um link para um site externo</p>
10     <a href="https://www.unip.br/" target="_blank">Link para o site da UNIP</a>
11     <br /> <!--Pula uma linha-->
12     <h3>Abaixo temos um link para a nossa página "hello world!!"</h3>
13     <a href="hello_world.html">Linka para a página Hello World</a>
14     <br />
15     <h4>Vamos colocar uma imagem para ser renderizada</h4>
16     <br />
17
18     <img src=""
19
20     </body>
21 </html>
22

```

```

index.htm* x hello_world.html
1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="utf-8" />
5      <title>ACR Devs</title>
6  </head>
7  <body>
8      <h1>Página inicial da ACR Devs</h1>
9      <p>Criando um link para um site externo</p>
10     <a href="https://www.unip.br/" target="_blank">Link para o site da UNIP</a>
11     <br /> <!--Pula uma linha-->
12     <h3>Abaixo temos um link para a nossa página "hello world!!"</h3>
13     <a href="hello_world.html">Linka para a página Hello World</a>
14     <br />
15     <h4>Vamos colocar uma imagem para ser renderizada</h4>
16     <br />
17
18     
21 </html>
22

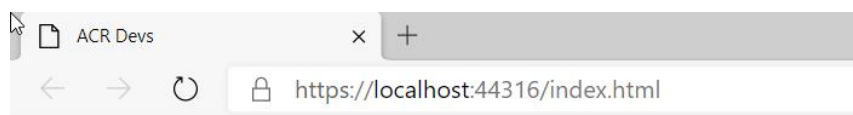
```

Veja como ficou a tag **<img>** e seus atributos

```

1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="utf-8" />
5      <title>ACR Devs</title>
6  </head>
7  <body>
8      <h1>Página inicial da ACR Devs</h1>
9      <p>Criando um link para um site externo</p>
10     <a href="https://www.unip.br/" target="_blank">Link para o site da UNIP</a>
11     <br /> <!--Pula uma linha-->
12     <h3>Abaixo temos um link para a nossa página "hello world!!"</h3>
13     <a href="hello_world.html">Linka para a página Hello World</a>
14     <br />
15     <h4>Vamos colocar uma imagem para ser renderizada</h4>
16     <br />
17
18     
19
20
21 </body>
22 </html>
23

```



## Página inicial da ACR Devs

Criando um link para um site externo

[Link para o site da UNIP](https://www.unip.br/)

Abaixo temos um link para a nossa página "hello world!!"

[Linka para a página Hello World](#)

Vamos colocar uma imagem para ser renderizada



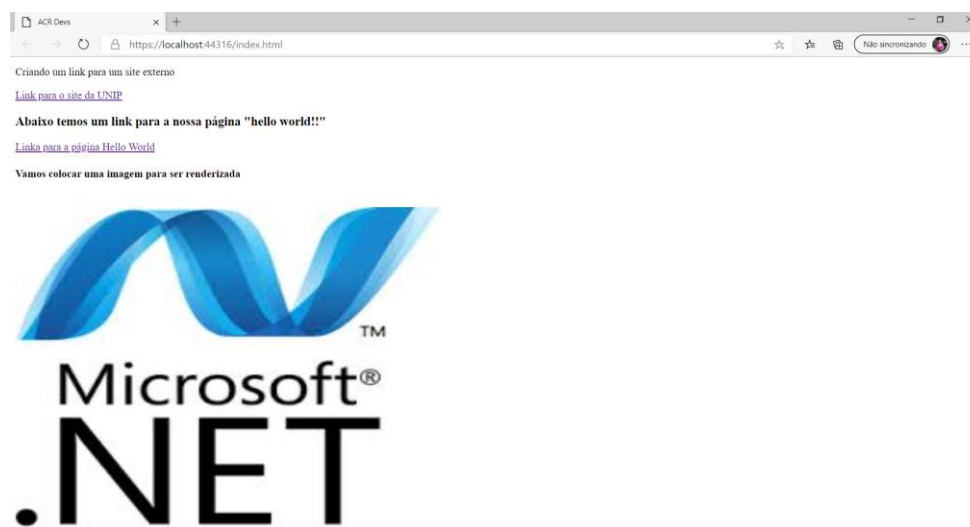
Agora acrescente as propriedade de altura e largura da imagem.

```

1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="utf-8" />
5      <title>ACR Devs</title>
6  </head>
7  <body>
8      <h1>Página inicial da ACR Devs</h1>
9      <p>Criando um link para um site externo</p>
10     <a href="https://www.unip.br/" target="_blank">Link para o site da UNIP</a>
11     <br /> <!--Pula uma linha-->
12     <h3>Abaixo temos um link para a nossa página "hello world!!"</h3>
13     <a href="hello_world.html">Linka para a página Hello World</a>
14     <br />
15     <h4>Vamos colocar uma imagem para ser renderizada</h4>
16     <br />
17
18     
19
20
21 </body>
22 </html>

```

Observe o tamanho da imagem na tela. Lembrando que essas propriedades não alteram o tamanho físico do arquivo.





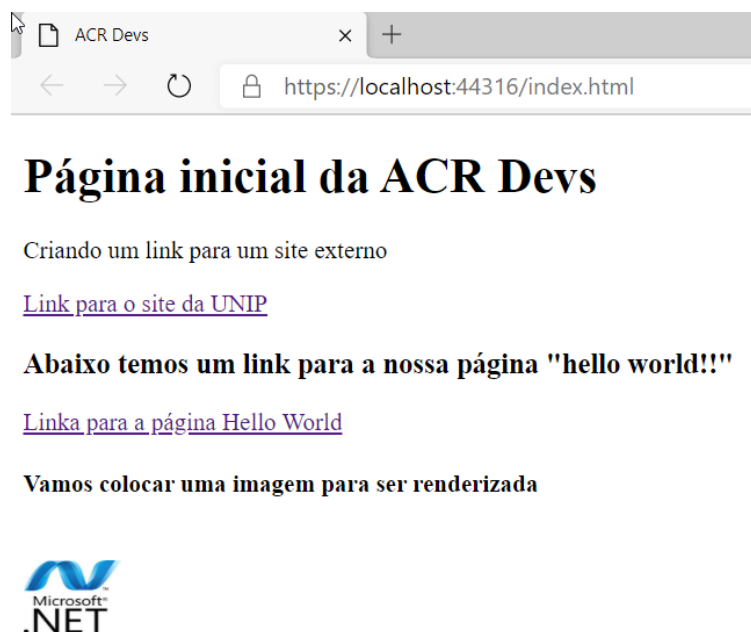
Faça alterações nas propriedades altura e largura da imagem e observe o seu comportamento depois que você iniciar a aplicação.

```

1  <!DOCTYPE html>
2  <html lang="pt-br">
3  <head>
4      <meta charset="utf-8" />
5      <title>ACR Devs</title>
6  </head>
7  <body>
8      <h1>Página inicial da ACR Devs</h1>
9      <p>Criando um link para um site externo</p>
10     <a href="https://www.unip.br/" target="_blank">Link para o site da UNIP</a>
11     <br /> <!--Pula uma linha-->
12     <h3>Abaixo temos um link para a nossa página "hello world!!"</h3>
13     <a href="hello_world.html">Linka para a página Hello World</a>
14     <br />
15     <h4>Vamos colocar uma imagem para ser renderizada</h4>
16     <br />
17
18     
19
20
21 </body>
22 </html>

```

Resultado da nova configuração das propriedades da imagem.



# Anexo A

Neste anexo eu apresento somente o print de telas para criar um projeto de aplicação web com ASP.Net Framework na versão do Visual Studio 2010.

