

Sistemas Multimédia

TPW-II

Tecnologia e Prática da Web II

14

ASP.NET Core

□ ASP.NET core

- ASP.NET
- .NET 5
- ASP.NET Core
- Projetos ASP.NET
- Razor

ASP.NET



□ Historia

■ 2002

- Ambiente de desenvolvimento disponibilizado pela plataforma .NET para a criação de páginas dinâmicas.
- O ASP.NET é uma estrutura da Web gratuita para a criação de sites e aplicações Web excelentes usando HTML, CSS e JavaScript.
 - Podemos criar APIs (Aplication Program Interface) e usar tecnologias em tempo real como Web Sockets.
- Sucessor da tecnologia ASP (Active Server Pages)
- Independente da linguagem utilizada
 - Várias linguagens de programação suportadas, entre elas o VB.NET e o C# (C Sharp)
- Independente do browser cliente
 - A plataforma .NET tem mecanismos internos que tratam do output para o cliente de acordo com o tipo de browser.
- Gratuito

□ Historia

■ 2002

- Informação inserida do lado do cliente pode ser diretamente processada no servidor
 - conceito de Web Form
- Permite a interação com recursos externos como as Base de Dados, quer seja em Access, SQL Server, MySql, XML ou outras.
- O código é compilado para o MSIL (Microsoft Intermediate Language), e depois compilado para o ficheiro com extensão .dll
- Possibilidade de separar o código do conteúdo da página.
- Suporte para a gestão de sessões. Por exemplo, em situações que requerem autenticação como é o caso do comércio electrónico.
- Fácil configuração (XML)
- Distribuição simples através da cópia de ficheiros (x-copy)
- Suporte para Web Services, que permite que funcionalidades (funções) sejam chamadas (invocadas) via Web, através de mensagens XML.

□ Historia

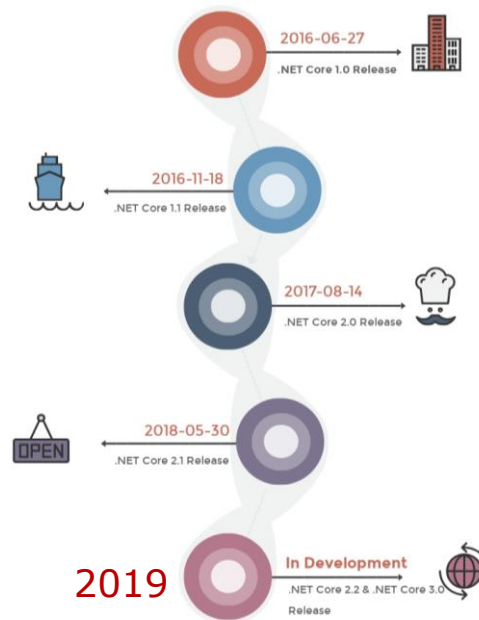
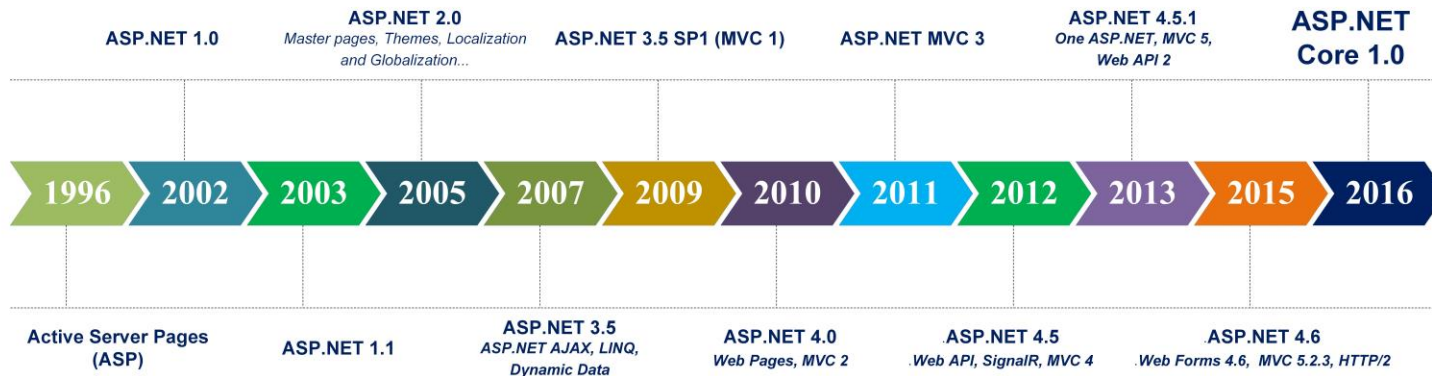
■ 2016

- A Microsoft decidiu criar um .NET framework de código aberto, com suporte aos principais sistemas operativos: Windows, OS X e Linux.
- Foi designado como ".NET Core framework", lançado em 2016, seguido de muitos lançamentos com diversas melhorias.

■ 2020

- Unificação de todas as frameworks
- Lançado em novembro de 2020, o .NET 5

The release history of ASP.NET

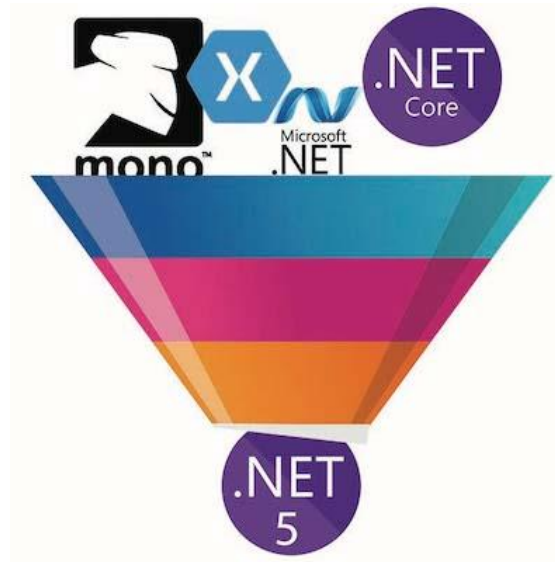


2020, Nov NET5

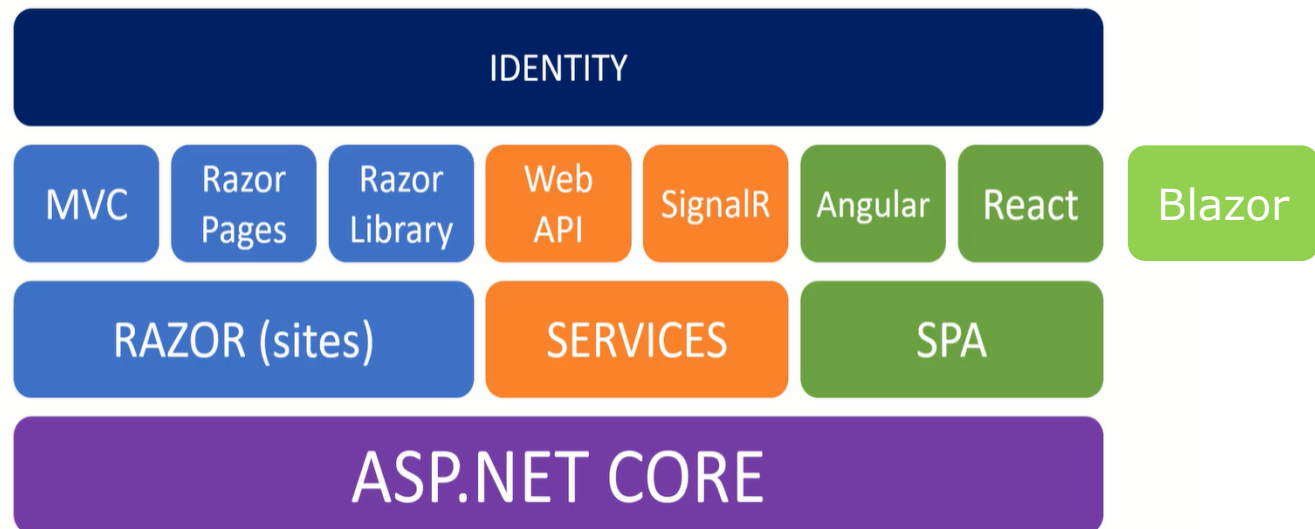
.NET 5

- ❑ .NET 5 lançado em novembro de 2020, como a **versão unificada** da **plataforma .NET**.
- ❑ O .NET 5 é um projeto que visa melhorar a plataforma .NET da seguinte forma:
 - Produzir um único *runtime* e framework .NET que possam ser usados em qualquer lugar e que tenham comportamentos de *runtime* uniformes e experiências de programador;
 - Expandir os recursos do .NET, aproveitando o melhor do .NET Core, .NET Framework, Xamarin e Mono;
 - Construir o produto a partir de uma única base de código que os programadores (Microsoft e a comunidade) possam trabalhar e expandir juntos;
- ❑ Assim haverá apenas um .NET que podemos usar em **Windows, Linux, macOS, iOS, Android, tvOS, watchOS, WebAssembly** e outras tecnologias.
- ❑ Foram introduzidas novas APIs .NET, recursos de *runtime* e recursos de linguagem

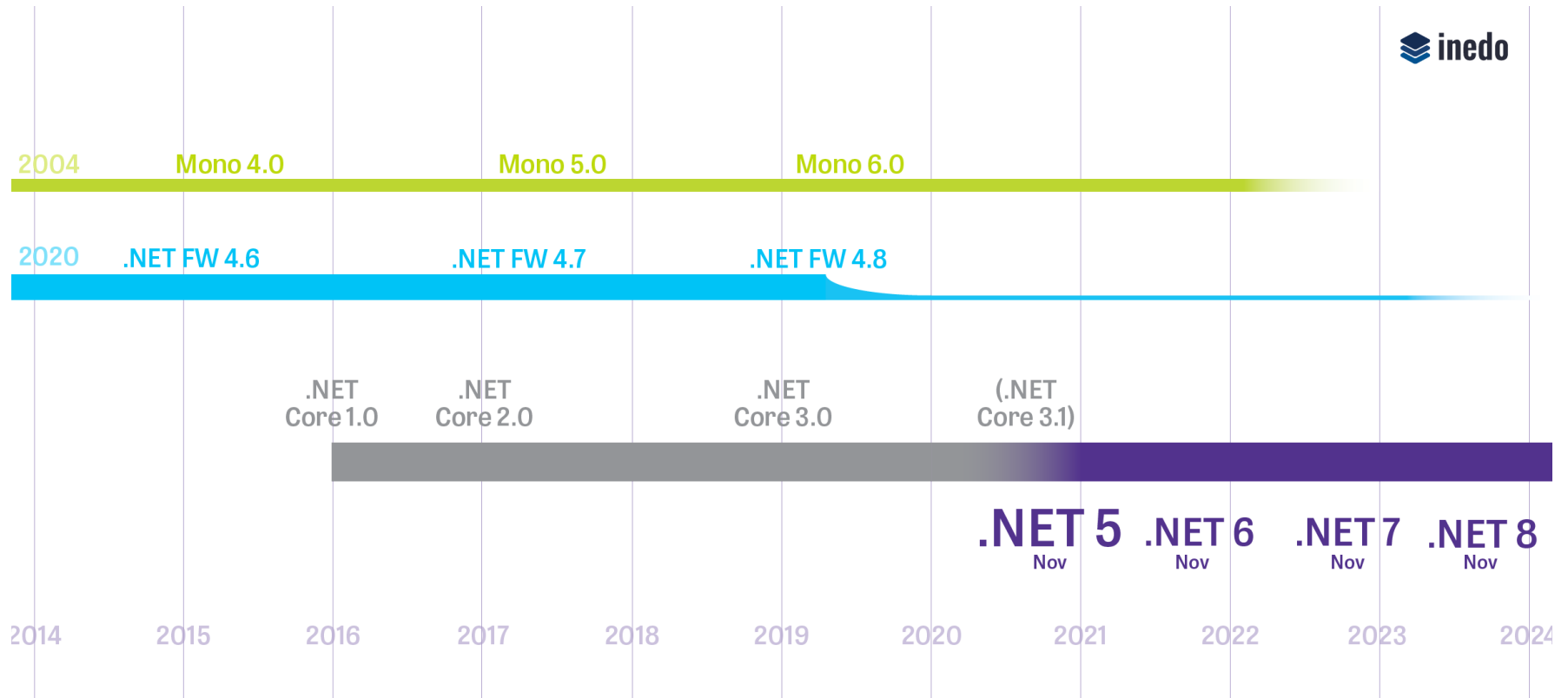
- O .NET5 tornou a plataforma .NET uma plataforma unificada:



□ Tipos de projetos ASP.NET core



□ Timeline .NET



ASP.NET core

O que é ASP.NET Core?

- Framework da Microsoft para desenvolvimento WEB
- Multiplataforma
- Alta performance
- Open-source
- Redesenhada a partir do ASP.NET 4.x

O que posso fazer com ASP.NET Core?

- Construir aplicações web
- Construir serviços web
- Aplicações para IoT (Internet of Things)
- Backoffices para aplicações mobile
- Desenvolver projetos em Windows, macOS ou Linux
- Disponibilizar as aplicações na Cloud ou servidor
- Executar com a .NET Core ou a .NET Framework

Porquê usar ASP.NET Core?

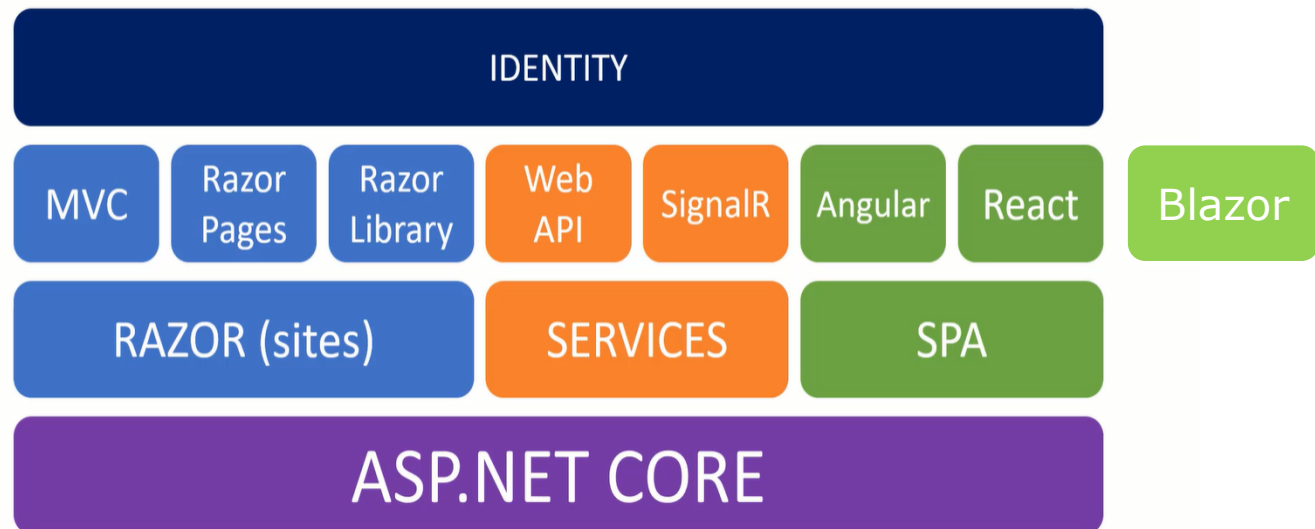
- Plataforma uniformizada para Web UI e Web APIs
 - Desenhada para ser objeto de testagem
 - Utiliza o template engine Razor para maior produtividade
 - Possibilidade de desenvolver em Windows, macOS e Linux
 - É open-source e focada na comunidade
-
- Permite a integração de frameworks cliente-side
 - Preparada para Cloud Services
 - Simples e de Alta performance
 - Pode ser hospedada em IIS, Nginx, Apache, Docker ...
 - Ferramentas modernas para desenvolvimento

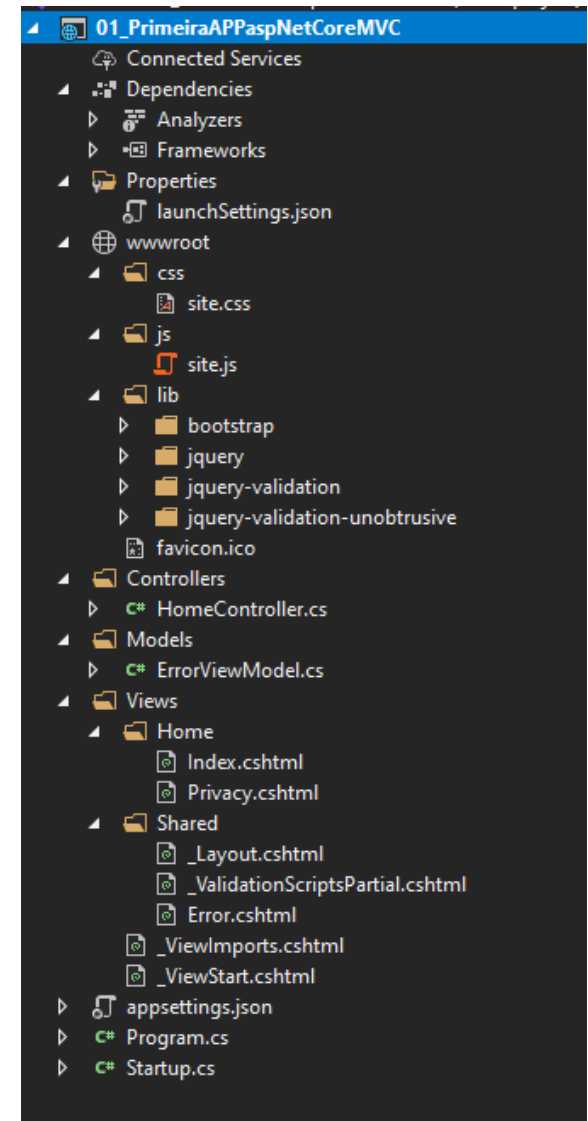
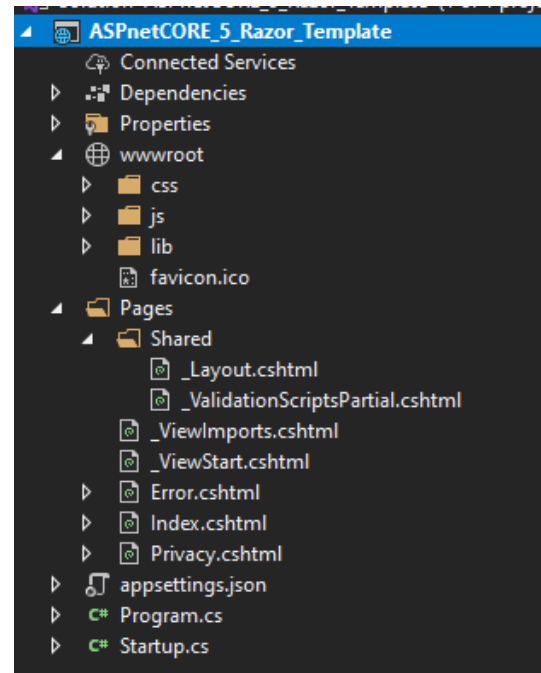
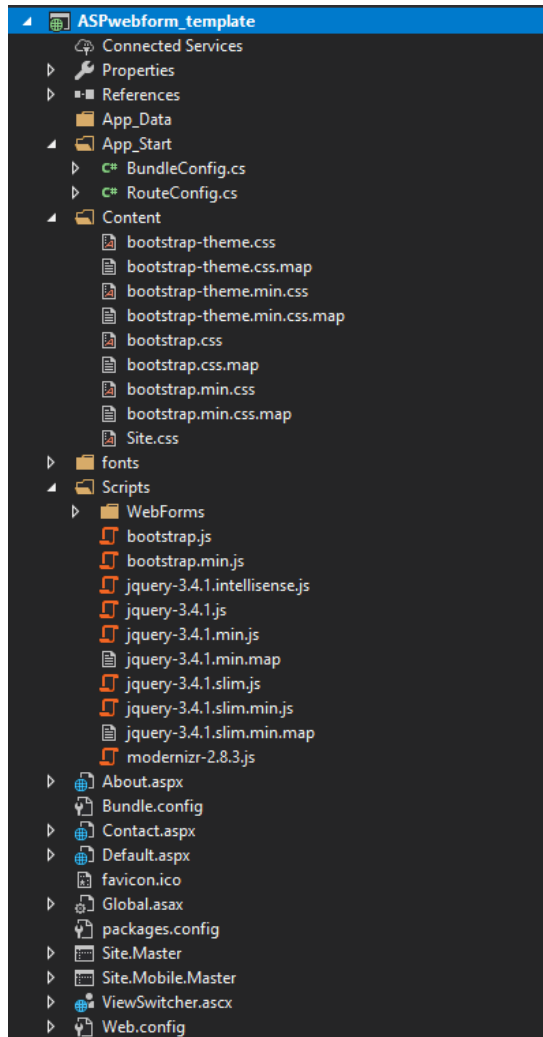
O que vamos encontrar em ASP.NET Core

- O padrão MVC (Model – View – Controller)
- Razor Pages (template engine)
- Razor Markup (sintaxe mais produtiva)
- Tag Helpers
- Integração com as mais populares bibliotecas: Angular, React, Bootstrap...

Projetos ASP.NET

□ Tipos de projetos ASP.NET core





ASP.NET Web Application
(web forms)

ASP.NET Core Web App
(razor pages)

ASP.NET Core Web App
(MVC)

Razor

□ Razor

■ Razor

- Razor é uma sintaxe de marcação para a incorporação de código baseado em servidor em páginas da Web.
- A sintaxe Razor consiste em marcação Razor, C# e HTML.
- Os ficheiros Razor geralmente têm uma extensão de arquivo `.cshtml`

■ Renderização de HTML

- A linguagem padrão do Razor é HTML. Renderizar HTML a partir da marcação do Razor não é diferente de renderizar HTML a partir de um arquivo HTML.
- A marcação HTML em arquivos Razor `.cshtml` é processada pelo servidor sem alterações.

<https://docs.microsoft.com/pt-br/aspnet/core/mvc/views/razor?view=aspnetcore-5.0>

□ Razor

É um View Engine responsável pelo processamento das Views. São elementos que contêm lógica de C# para gerar conteúdo dinâmico.

No **ASP.NET CORE MVC** não existem entraves à utilização de diferentes fontes para o Model, utilização de outras frameworks e bibliotecas de CSS, JavaScript, etc.

https://www.w3schools.com/asp/razor_intro.asp

□ Razor

■ Sintaxe de Razor

- Razor dá suporte a C# e usa o símbolo @ para fazer a transição de HTML para C#.
- Razor avalia expressões C# e as renderiza na saída HTML.
- Quando um símbolo @ é seguido por uma Razor palavra-chave reservada, ele faz a transição para uma marcação Razor específica. Caso contrário, ele faz a transição para C# simples.

<https://docs.microsoft.com/pt-br/aspnet/core/mvc/views/razor?view=aspnetcore-5.0#razor-reserved-keywords>

□ Razor

Razor Keywords

page

namespace

functions

inherits

model

section

helper (Atualmente, não há suporte para ASP.NET Core)

razor as palavras-chave têm escape @(Razor Keyword)

(por exemplo, @(functions)).

□ Razor

Palavras-chave do C#

```
case  
do  
default  
for  
foreach  
if  
else  
lock  
switch  
try  
catch  
finally  
using  
while
```

□ Razor

Exemplos:

```
<p>@DateTime.Now</p>  
<p>@DateTime.IsLeapYear(2016) </p>
```

```
@{  
    var joe = new Person("Joe", 33);  
}
```

```
<p>Age@ (joe.Age) </p>
```

□ Razor

Exemplos:

```
@{
    Layout = null;
    string Nome = "Jose";
}

<p>@Nome</p>
<p>O valor da variável @@Nome é @Nome</p>
<p>@(10+20)</p>
<p>@DateTime.Now.Year</p>
@{
    string valor = "valor qualquer";
}
@{ valor = "alterar valor";}

@ {
    Nome = "martim";
    <p>Dentro do Razor posso apresentar o nome @Nome</p>
}
```

□ Razor

Exemplos:

```
@{  
    List<strings> lista_strings = new<string>();  
    List<Pessoa> lista_strings = new<Pessoa>();  
}
```

```
@for(int i=0;i<10;i++)  
{  
    <p>@i</p>  
}
```