



**TECNOLOGIA E INOVAÇÃO
EM PROL DA INDÚSTRIA**



Curso Técnico em Informática

Desenvolvimento de Sistemas II – 180h

Prof^a: Francisleide Almeida

Razor

- Razor é uma sintaxe de marcação que permite incorporar código baseado em servidor (Visual Basic e C #) em páginas da web.
- O código baseado no servidor pode criar conteúdo dinâmico na web, enquanto uma página da Web é gravada no navegador
- Quando uma página da Web é chamada, o servidor executa o código baseado no servidor dentro da página antes de retornar a página para o navegador

- Ao executar no servidor, o código pode executar tarefas complexas, como acessar bancos de dados.
- O Razor é baseado no ASP.NET e foi projetado para criar aplicativos da web. Ele tem o poder da marcação ASP.NET tradicional, mas é mais fácil de usar e mais fácil de aprender.

Utilização

- Ex:

@for (int i = 0; i < 10; i++) {

@i

}

Exemplo

```
<html>
<body>
<!-- Single statement block -->
@{ var myMessage = "Hello World"; }

<!-- Inline expression or variable -->
<p>The value of myMessage is: @myMessage</p>

<!-- Multi-statement block -->
@{
var greeting = "Welcome to our site!";
var weekDay = DateTime.Now.DayOfWeek;
var greetingMessage = greeting + " Here in Huston it is: " + weekDay;
}

<p>The greeting is: @greetingMessage</p>
</body>
</html>
```

Trabalhando com objetos

- O objeto "DateTime" é um objeto ASP.NET interno típico, mas os objetos também podem ser autodefinidos, uma página da web, uma caixa de texto, um arquivo, um registro de banco de dados, etc.
- Os objetos podem ter métodos que eles podem executar. Um registro de banco de dados pode ter um método "Salvar", um objeto de imagem pode ter um método "Girar", um objeto de e-mail pode ter um método "Enviar" e assim por diante.


```
<html>
<body>

<table border="1">
<tr>
<th width="100px">Name</th>
<th width="100px">Value</th>
</tr>
<tr>
<td>Day</td><td>@DateTime.Now.Day</td>
</tr>
<tr>
<td>Hour</td><td>@DateTime.Now.Hour</td>
</tr>
<tr>
<td>Minute</td><td>@DateTime.Now.Minute</td>
</tr>
<tr>
<td>Second</td><td>@DateTime.Now.Second</td>
</tr>
</table>

</body>
</html>
```

Name	Value
Day	12
Hour	12
Minute	3
Second	34

```
@{
var txt = "";
if(DateTime.Now.Hour > 12)
{txt = "Good Evening";}
else
{txt = "Good Morning";}
}
<html>
<body>
<p>The message is @txt</p>
</body>
</html>
```

```
@{
var totalMessage = "";
if(IsPost)
{
var num1 = Request["text1"];
var num2 = Request["text2"];
var total = num1.AsInt() + num2.AsInt();
totalMessage = "Total = " + total;
}
}
<!DOCTYPE html>
<html>
<body style="background-color: beige; font-family: Verdana, Arial;">
<form action="" method="post">
<p><label for="text1">First Number:</label><br>
<input type="text" name="text1"></p>
<p><label for="text2">Second Number:</label><br>
<input type="text" name="text2"></p>
<p><input type="submit" value=" Add "></p>
</form>
<p>@totalMessage</p>
</body>
</html>
```

First Number:

Second Number:

Add

Total = 90

Data Types

Type	Description	Examples
int	Integer (whole numbers)	103, 12, 5168
float	Floating-point number	3.14, 3.4e38
decimal	Decimal number (higher precision)	1037.196543
bool	Boolean	true, false
string	String	"Hello W3Schools", "John"

Operators

Operator	Description	Example
=	Assigns a value to a variable.	i=6
+	Adds a value or variable.	i=5+5
-	Subtracts a value or variable.	i=5-5
*	Multiplies a value or variable.	i=5*5
/	Divides a value or variable.	i=5/5
+=	Increments a variable.	i += 1
-=	Decrements a variable.	i -= 1
==	Equality. Returns true if values are equal.	if (i==10)
!=	Inequality. Returns true if values are not equal.	if (i!=10)
<	Less than.	if (i<10)
>	Greater than.	if (i>10)
<=	Less than or equal.	if (i<=10)
>=	Greater than or equal.	if (i>=10)
+	Adding strings (concatenation).	"w3" + "schools"
.	Dot. Separate objects and methods.	DateTime.Hour
()	Parenthesis. Groups values.	(i+5)
()	Parenthesis. Passes parameters.	x=Add(i,5)
[]	Brackets. Accesses values in arrays or collections.	name[3]
!	Not. Reverses true or false.	if (!ready)
&&	Logical AND.	if (ready && clear)
	Logical OR.	if (ready clear)

Convertendo Data Types

Method	Description	Example
AsInt() IsInt()	Converts a string to an integer.	<pre>if (myString.IsInt()) {myInt=myString.AsInt();}</pre>
AsFloat() IsFloat()	Converts a string to a floating-point number.	<pre>if (myString.IsFloat()) {myFloat=myString.AsFloat();}</pre>
AsDecimal() IsDecimal()	Converts a string to a decimal number.	<pre>if (myString.IsDecimal()) {myDec=myString.AsDecimal();}</pre>
AsDateTime() IsDateTime()	Converts a string to an ASP.NET DateTime type.	<pre>myString="10/10/2012"; myDate=myString.AsDateTime();</pre>
AsBool() IsBool()	Converts a string to a Boolean.	<pre>myString="True"; myBool=myString.AsBool();</pre>
ToString()	Converts any data type to a string.	<pre>myInt=1234; myString=myInt.ToString();</pre>

Loops and Arrays

```
<html>
<body>
<ul>
@foreach (var x in Request.ServerVariables)
{<li>@x</li>}
</ul>
</body>
</html>
```

- ALL_HTTP
- ALL_RAW
- APPL_MD_PATH
- APPL_PHYSICAL_PATH
- AUTH_PASSWORD
- AUTH_TYPE
- AUTH_USER
- CERT_COOKIE
- CERT_FLAGS
- CERT_ISSUER
- CERT_KEYSIZE
- CERT_SECRETKEYSIZE
- CERT_SERIALNUMBER
- CERT_SERVER_ISSUER
- CERT_SERVER_SUBJECT
- CERT_SUBJECT
- CONTENT_LENGTH
- CONTENT_TYPE
- GATEWAY_INTERFACE
- HTTPS
- HTTPS_KEYSIZE
- HTTPS_SECRETKEYSIZE
- HTTPS_SERVER_ISSUER
- HTTPS_SERVER_SUBJECT
- INSTANCE_ID
- INSTANCE_META_PATH
- LOCAL_ADDR
- LOGON_USER
- PATH_INFO
- PATH_TRANSLATED
- QUERY_STRING
- REMOTE_ADDR
- REMOTE_HOST
- REMOTE_USER

While Loops

```
<html>
<body>
@{
var i = 0;
while (i < 5)
{
i += 1;
<p>Line @i</p>
}
}
</body>
</html>
```

Line 1

Line 2

Line 3

Line 4

Line 5

Arrays

Source:

```
@{
string[] members = {"Jani", "Hege", "Kai", "Jim"};
int i = Array.IndexOf(members, "Kai")+1;
int len = members.Length;
string x = members[2-1];
}
<html>
<body>
<h3>Members</h3>
@foreach (var person in members)
{
<p>@person</p>
}
<p>The number of names in Members are @len</p>
<p>The person at position 2 is @x</p>
<p>Kai is now in position @i</p>
</body>
</html>
```

Result:

Members

Jani

Hege

Kai

Jim

The number of names in Members are 4

The person at position 2 is Hege

Kai is now in position 3

1 - Utilizando o framework Razor , crie um formulário HTML de preenchimento de cadastro uma pessoa e um botão com método Post para gerar uma tabela com todos os dados dele e a solicitação de confirmação dos dados informados;

2 – Crie uma formulário com 5 nomes e idades e depois exiba uma lista com maiores de idade e menores de idade, da seguinte forma:

Maiores de idade:

Pessoa 1: Fulano

Pessoa 2: Cicrano

Menores de idade

Pessoa 1: Beltrano

3 – Usando Razor, crie uma página que crie um sorteio de equipes e temas de trabalho. Cadastre pessoas e temas para equipes por meio de formulários e depois sorteie as pessoas e os temas.

Classe random do razor:

```
var r = new Random();
```

```
var x = r.Next(quantidade_de_sorteios);
```