

Les vues en ASP.NET MVC

Moteur de rendu Razor

MOHAMED AIROUCHE

Le moteur de vue Razor

Microsoft a introduit le moteur de vue Razor dans la version MVC 3. Razor permet d'écrire un mélange de balises HTML et de code côté serveur dans une vue.

Le langage C# ou Visual Basic peuvent être utilisés pour écrire du code côté serveur dans une vue Razor.

Le moteur Razor utilise le caractère @ pour le code côté serveur au lieu du <%%> traditionnel.

Les fichiers de vues Razor ont une extension .cshtml ou vbhtml.

Le mélange de code C# et du code HTML est désigné par l'extension de pages .cshtml

<https://www.tutorialsteacher.com/mvc/razor-syntax>

La syntaxe de Razor

Razor utilise la syntaxe suivante :

Pour déclarer une variable dans la vue à l'aide de la syntaxe Razor, nous devons d'abord créer un bloc de code à l'aide de `@ {et}`, puis utiliser la même syntaxe que celle utilisée dans le C #.

Pour écrire plusieurs lignes (un bloc) de code côté serveur de la vue Razor, nous utilisons la syntaxe `@ {...}`. Chaque ligne doit se terminer par un point-virgule identique à C #.

Pour commenter dans le bloc de code de la vue Razor, nous utilisons la même syntaxe que celle utilisée en C #. Comme pour une seule ligne `//` et pour plusieurs lignes `/*` et `*/`.

Pour commenter dans partie HTML, nous devons utiliser un bloc de commentaires qui commence par `@*` et se termine par `*@`.

<https://www.tutorialsteacher.com/mvc/razor-syntax>

La syntaxe de Razor

➤ **Voici un résumé comment écrire de code du moteur Razor**

1. Le signe @ est utilisé pour passer à l'écriture de code côté serveur.
2. Une balise HTML permet de passer automatiquement au code HTML.
3. Un bloc de code du côté serveur commence par @ suivi de `{* code *}`.
4. Le signe @: ou `<text> </ <text>` est utilisé dans un bloc de code côté serveur pour afficher de texte.
5. La condition If commence par `@if {}`.
6. La boucle For commence avec `@for{}`.
7. `@model` vous permet d'utiliser l'objet modèle n'importe où dans la vue

Exemples d'utilisation de la syntaxe de Razor

@*

Utilisez @ pour écrire le code côté serveur.

*@

<h1>Démo de la syntaxe Razor</h1>

<h2>@DateTime.Now.ToShortDateString()</h2>

@*

Le code du côté serveur commence par @ { * code * } :

*@

@{

var date = DateTime.Now.ToShortDateString();

var message = "Bonjour!";

}

<h2>La date d'aujourd'hui est le: @date </h2>

<h3> @message </h3>

Exemples d'utilisation de la syntaxe de Razor

@*

Utilisez @: ou <text> </text> pour afficher le texte du bloc de code.

*@

@{

```
var date1 = DateTime.Now.ToShortDateString();
```

```
string message1 = "Bonjour!";
```

```
@: La date du jour est: @date <br />
```

```
@message1
```

}

@{

```
var date2 = DateTime.Now.ToShortDateString();
```

```
string message2 = "Bonjour!";
```

```
<text> La date du jour est: </text> @date2 <br />
```

```
@message2
```

}

@*

Exemples d'utilisation de la syntaxe de Razor

```
@*
  la condition if commence par @if { }
  *@
@if (DateTime.IsLeapYear(DateTime.Now.Year))
{
    @DateTime.Now.Year @: est une année bissextile.
}
else
{
    @DateTime.Now.Year @: n'est pas une année bissextile.<br />
}

@*
  la boucle For commence avec @for{ }
  *@
@for (int i = 0; i < 5; i++)
{
    @i.ToString() <br />
}
```

<https://www.tutorialsteacher.com/mvc/razor-syntax>

Les HTML helpers

- Il existe un grand nombre de HTML helpers qui ont pour objectif de simplifier la génération des balises des contrôles HTML.
- La classe HtmlHelper génère des éléments HTML à l'aide de l'objet de classe Model dans une vue Razor.
- La classe HtmlHelper lie l'objet de modèle aux éléments HTML pour afficher la valeur des propriétés du modèle en éléments HTML et affecte également la valeur des éléments HTML aux propriétés du modèle lors de la soumission du formulaire Web.
- Utilisez toujours la classe HtmlHelper en vue Razor au lieu d'écrire manuellement les balises HTML.

<https://www.tutorialsteacher.com/mvc/html-helpers>

Les méthodes de formulaires

Les méthodes de formulaires servent à mettre en correspondance les balises HTML <input> avec les propriétés d'un modèle en utilisant les helpers HTML :

Les méthodes de formulaires	
@Html.BeginForm	Ouvre la balise <form>. Voir aussi EndForm
@Html.Checkbox	Produit une case à cocher pour un champ
@Html.Editor	Produit une zone de saisie pour un champ
@Html.Label	Produit un label pour un champ
@Html.ListBox	Produit une liste pour un champ
@Html.Radiobutton	Produit un bouton radio
@Html.Textarea	Produit une zone de texte multiligne
@Html.Textbox	Produit une zone de texte

Pour plus de détails: <https://www.tutorialsteacher.com/mvc/html-helpers>

Exemple d'utilisation des méthodes de formulaires

```
@using (Html.BeginForm("Index", "Accueil", FormMethod.Post))
{ <br />
    @Html.Label("UserName", "Utilisateur", null)
    @Html.TextBox("UserName")
    <br /><br />
    @Html.Label("UserPhone", "Téléphone", null)
    @Html.TextBox("UserPhone")
    <br /><br />
    @Html.Label("UserEmail", "Email", null)
    @Html.TextBox("UserEmail")
    <br /><br />
    @Html.Label("UserPassword", "Mot de passe", null)
    @Html.Password("UserPassword")
    <br /><br />
    @Html.Label("M")
        @Html.RadioButton("M", new { value = "Male" })
        @Html.Label("F")
        @Html.RadioButton("F", new { value = "Female" })

    <br /><br />
    @Html.Label("< pays")

    @Html.DropDownList("UserCountry", new SelectList(
        new List<SelectListItem>()
        {
            new SelectListItem(){Value="1",Text="Canada"},
            new SelectListItem(){Value="2",Text="USA"},

        }, "Value", "Text"))

        <br /><br />
        <input type = "submit" value = "Submit" />
}
```

Pour plus de détails:

<http://sergetahe.com/cours-tutoriels-de-programmation/wp-content/uploads/documents/Introduction%20%C3%A0%20ASP.NET%20MVC%20par%20l'exemple%20-%202013.pdf>

Exemple d'utilisation des méthodes de formulaires

```
<form action="/" method="post">
  <br /><br />
  <label for="UserName">Utilisateur </label>
  <input id="UserName" name="UserName" type="text" value="" />
  <br /><br />
  <label for="UserPhone">Téléphone</label>
  <input id="UserPhone" name="UserPhone" type="text" value="" />
  <br /><br />
  <label for="UserEmail">Email</label>
  <input id="UserEmail" name="UserEmail" type="text" value="" />
  <br /><br />
  <label for="UserPassword">Mot de passe </label>
  <input id="UserPassword" name="UserPassword" type="password" />
  <br /><br />
  <label for="M">M</label>
  <input id="M" name="M" type="radio" value="{ value = Male }" />
  <label for="F">F</label>
  <input id="F" name="F" type="radio" value="{ value = Female }" />
  <br /><br />
  <label for="UserCountry">Pays</label>
  <select id="UserCountry" name="UserCountry">
    <option value="1">Canada</option>
    <option value="2">USA</option>
  </select>
  <br /><br />

  <input type="submit" value="Submit" />
</form>
```

Références:

- <https://www.tutorialsteacher.com/mvc/razor-syntax>
- <http://sergetahe.com/cours-tutoriels-de-programmation/wp-content/uploads/documents/Introduction%20%C3%A0%20ASP.NET%20MVC%20par%20l'exemple%20-%202013.pdf>
- <https://openclassrooms.com/fr/courses/1730206-apprenez-asp-net-mvc/2583844-economiser-son-code-avec-les-helpers-et-les-vues-partielles>