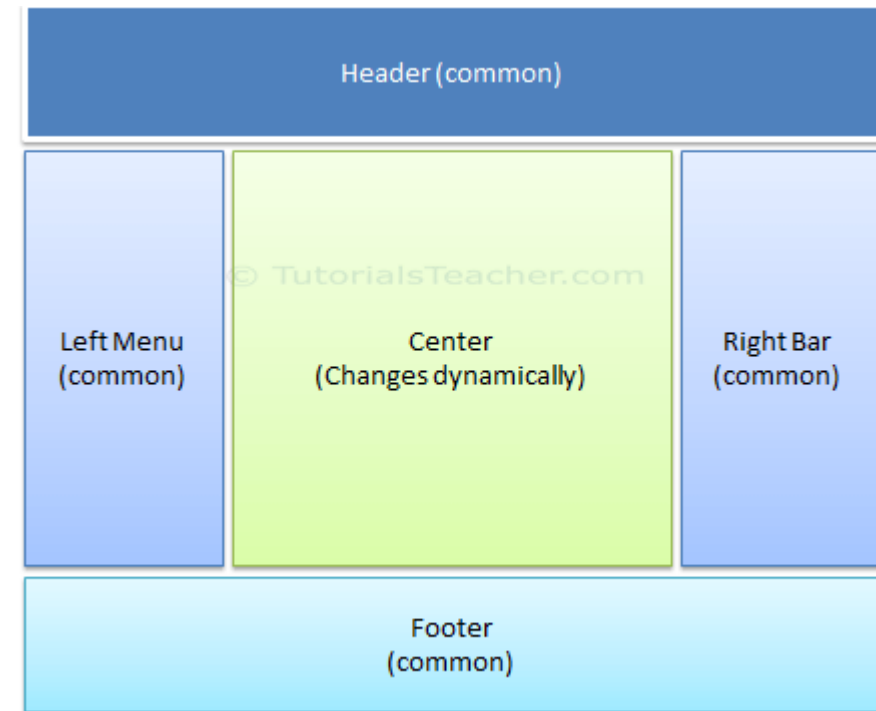


Utilisation de la page Layout et des vues partielles en ASP.NET MVC.

Utilisation de la page de disposition (Layout)

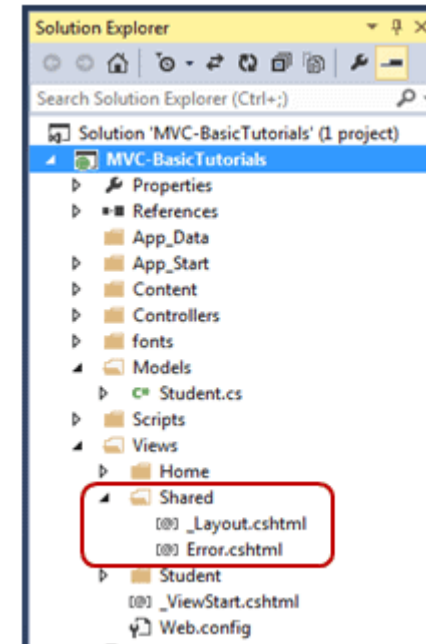
- ASP.NET MVC a introduit la possibilité d'utiliser une page de disposition (Layout). Le principe est de regrouper, dans cette page spécifique, les éléments communs de mise en page pour l'application (tels que le menu, l'entête, le pied de page, ou autres scripts JavaScript utilisés par toutes les pages).
- Les modèles de disposition vous permettent de spécifier la disposition du conteneur HTML de votre application dans un emplacement unique, puis de l'appliquer sur plusieurs pages de votre application.



<https://www.tutorialsteacher.com/mvc/layout-view-in-asp.net-mvc>

Utilisation de la page de disposition (Layout)

La page de disposition a la même extension que les autres vues, .cshtml ou .vbhtml. Les pages de disposition sont partagées avec plusieurs vues, elles doivent donc être stockées dans le dossier partagé (Shared). Par défaut, une page de disposition `_Layout.cshtml` est créée lorsque vous créez une application MVC à l'aide de Visual Studio, comme indiqué ci-dessous.



<https://www.tutorialsteacher.com/mvc/layout-view-in-asp.net-mvc>

Utilisation de la page de disposition (Layout)

- Par défaut, lors de la création d'une application ASP.NET MVC utilisant un modèle le fichier Layout est créé automatiquement.
- La page Layout est une page HTML qui contient des balises HTML Doctype, head et body. En plus, elle contient des appels aux méthodes **RenderBody()** et **RenderSection()**. Les vues enfants seront affichées là où le **RenderBody()** est appelé.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
5 <meta charset="utf-8" />
6 <meta name="viewport" content="width=device-width, initial-scale=1.0">
7 <title>@ViewBag.Title - Mon application ASP.NET</title>
8 @Styles.Render("~/Content/css")
9 @Scripts.Render("~/bundles/modernizr")
10 </head>
11 <body>
12 <div class="navbar navbar-inverse navbar-fixed-top">
13 <div class="container">
14 <div class="navbar-header">
15 <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
16 <span class="icon-bar"></span>
17 <span class="icon-bar"></span>
18 <span class="icon-bar"></span>
19 </button>
20 @Html.ActionLink("Nom de l'application", "Index", "Home", new { area = "" }, new { @class = "navbar-brand" })
21 </div>
22 <div class="navbar-collapse collapse">
23 <ul class="nav navbar-nav">
24 <li>@Html.ActionLink("Accueil", "Index", "Home")</li>
25 <li>@Html.ActionLink("À propos de", "About", "Home")</li>
26 <li>@Html.ActionLink("Contact", "Contact", "Home")</li>
27 </ul>
28 </div>
29 </div>
30 </div>
31 <div class="container body-content">
32 @RenderBody()
33 <hr />
34 <footer>
35 <p>&copy; @DateTime.Now.Year - My ASP.NET Application</p>
36 </footer>
37 </div>
38
39 @Scripts.Render("~/bundles/jquery")
40 @Scripts.Render("~/bundles/bootstrap")
41 @RenderSection("scripts", required: false)
42 </body>
43 </html>
```

Utilisation de la page de disposition (Layout)

○ Utilisation de la page de disposition (Layout)

Les vues qui seront affichées dans un espace réservé **RenderBody()** sont appelées vues enfants. Il existe plusieurs façons de spécifier quelle page Layout sera utilisée avec quelles vues enfants. Vous pouvez le spécifier dans :

- ✓ un fichier commun `_ViewStart.cshtml`,
- ✓ une vue enfant,
- ✓ une méthode d'action.

○ Spécifier une page Layout dans ViewStart

Par défaut, le fichier `_ViewStart.cshtml` est inclus dans le dossier Views. Il peut également être créé dans tous les autres sous-dossiers Views. Il est utilisé pour spécifier des paramètres communs pour toutes les vues sous un dossier et sous-dossiers où il est créé.

<https://www.tutorialsteacher.com/mvc/layout-view-in-asp.net-mvc>

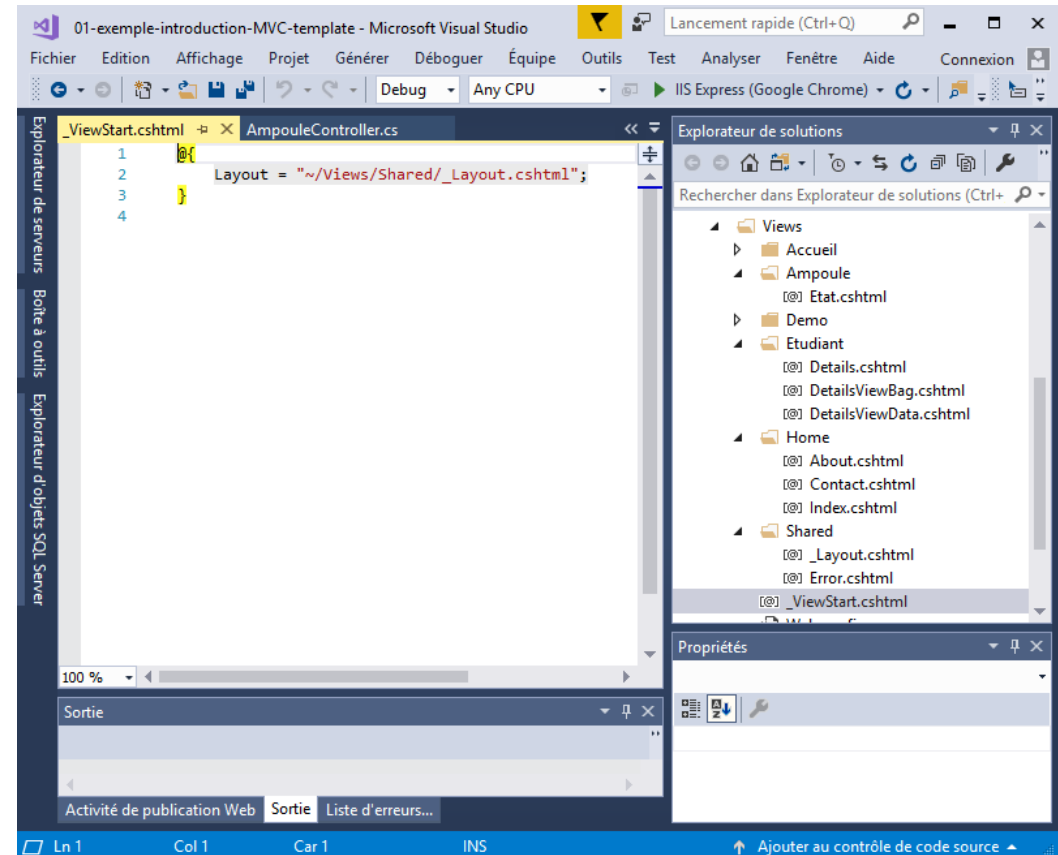
Utilisation de la page de disposition (Layout)

Exemple

La page `_ViewStart.cshtml` dans le dossier Views définit la page de disposition (Layout) : `"~/Views/Shared/_Layout.cshtml"`. Ainsi, le fichier `_Layout.cshtml` peut être utilisé comme une vue de mise en page de toutes les vues incluses dans Views et ses sous-dossiers.

Si on veut exclure l'utilisation de la page Layout dans une page spécifique on ajoute le code suivant dans cette page :

```
@{  
    Layout = null;  
}
```



<https://www.tutorialsteacher.com/mvc/layout-view-in-asp.net-mvc>

Utilisation de la page de disposition (Layout)

○ Spécifier une page Layout dans une vue enfant

Vous pouvez également remplacer le paramètre d'affichage de disposition par défaut de `_ViewStart.cshtml` en définissant la propriété `Layout` dans chaque vue enfant. Par exemple, on ajoute le code suivant pour que la vue `Index.cshtml` utilise `_myLayoutPage.cshtml` même si `_ViewStart.cshtml` définit `_Layout.cshtml`.

```
@{
    ViewBag.Title = "Home Page";
    Layout = "~/Views/Shared/_myLayoutPage.cshtml";
}
```

○ Spécifier une page Layout dans la méthode d'action

Spécifiez le nom de la page de disposition comme deuxième paramètre dans la méthode `View()`, comme indiqué ci-dessous. Par défaut, la page de disposition sera recherchée dans le dossier partagé.

```
public class HomeController : Controller
{
    public ActionResult Index()
    {
        return View("Index", "_myLayoutPage"); //set "_myLayoutView" as layout view
    }
}
```

<https://www.tutorialsteacher.com/mvc/layout-view-in-asp.net-mvc>

Utilisation des vues partielles

Une vue partielle est une partie réutilisable d'une page Web. Il s'agit d'un fichier .cshtml ou .vbhtml qui contient du code HTML. Il peut être utilisé dans une ou plusieurs vues ou dans les pages Layout. Vous pouvez utiliser la même vue partielle à plusieurs endroits et éliminer le code redondant.

La vue partielle dans ASP.NET MVC est une vue spéciale qui restitue une partie du contenu de la vue. C'est comme un contrôle utilisateur d'une application de formulaire Web.



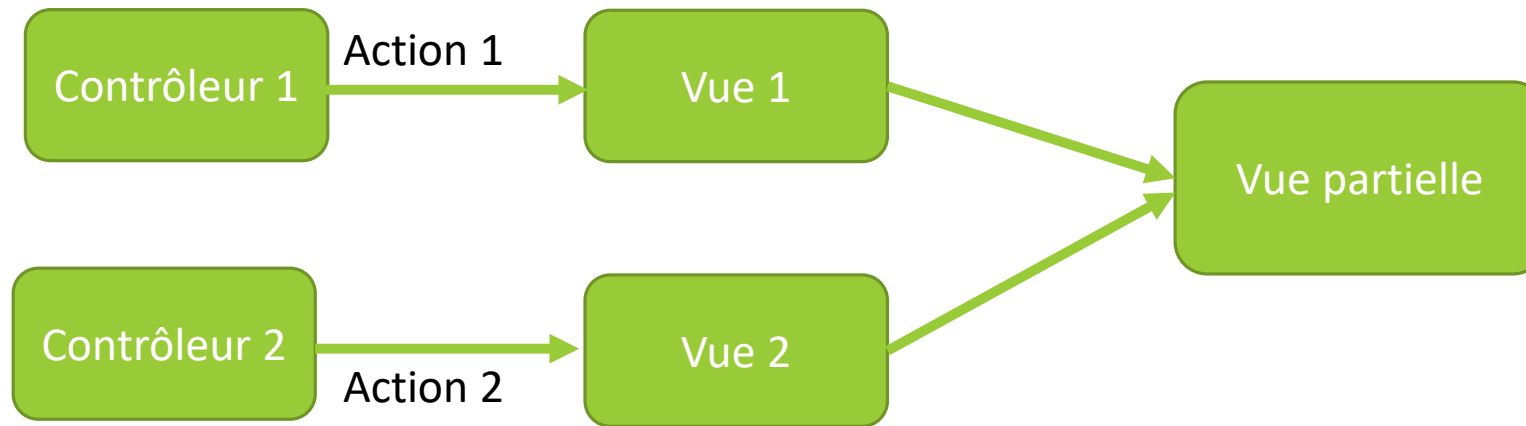
Exemple d'une vue qui utilise deux vues partielles

Avantage :

- Rendre la vue simple à comprendre (réduire la complexité de la vue).

Utilisation des vues partielles

Les vues partielles peuvent être réutilisées dans plusieurs vues. Cela aide à réduire la duplication de code.



Utilisation d'une vue partielle par deux vues

Avantage :

- Simplifier la maintenance par la mise à jour de la vue partielle

Utilisation des vues partielles

- **Créer une nouvelle vue partielle**

- ✓ Vous pouvez créer une vue partielle de la même façon que les autres vues,
- ✓ Vous pouvez créer une vue partielle dans n'importe quel dossier View. Cependant, il est recommandé de créer toutes vos vues partielles dans le dossier partagé afin qu'elles puissent être utilisées dans plusieurs vues.

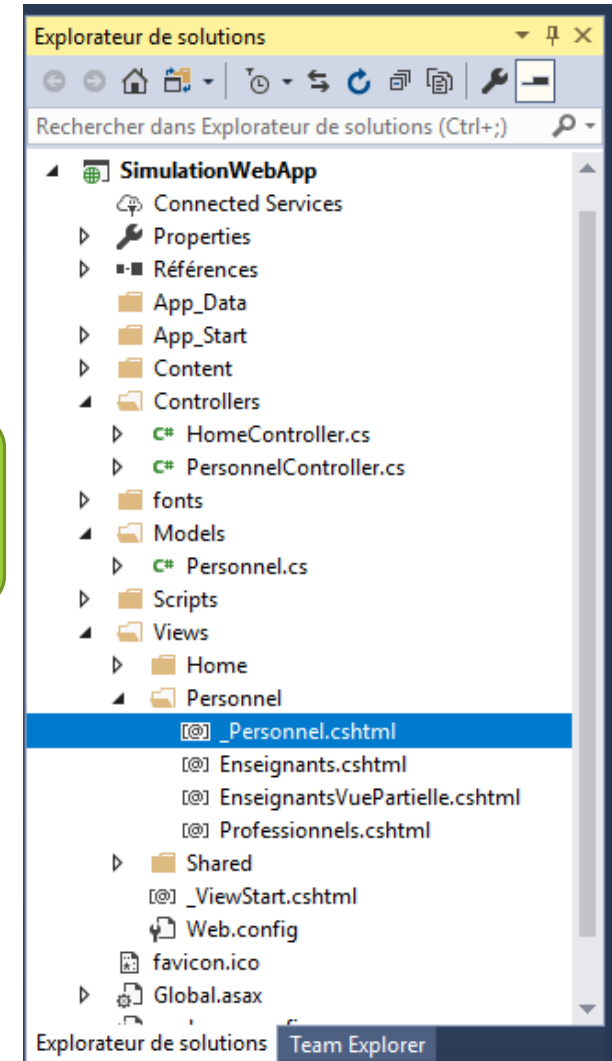
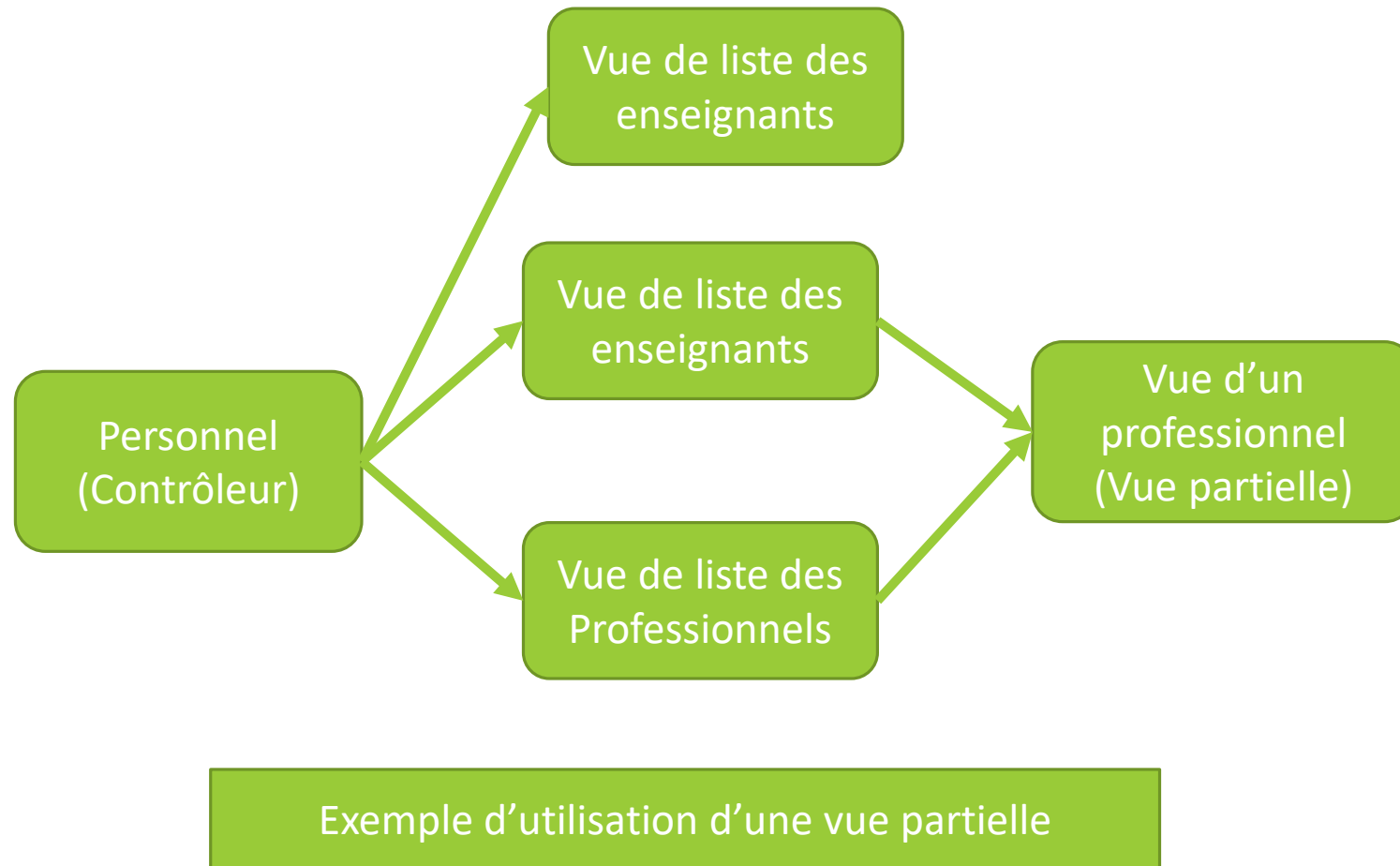
- **Rendu d'une vue partielle**

Vous pouvez appeler une méthode de la vue partielle dans la vue parente à l'aide des HTML helpers:

- ✓ `@html.Partial()`,
- ✓ `@html.RenderPartial ()`,
- ✓ `@html.RenderAction ()`.

<https://www.tutorialsteacher.com/mvc/partial-view-in-asp.net-mvc>

Exemple : Utilisation des vues partielles



Exemple : Utilisation des vues partielles

```
6 namespace SimulationWebApp.Models
7 {
8     18 références
9     public class Personnel
10    {
11        11 références
12        public string Nom { get; set; }
13        11 références
14        public string Prénom { get; set; }
15        11 références
16        public string sexe { get; set; }
17        11 références
18        public int Age { get; set; }
19        8 références
20        public int AnnéeAncienneté { get; set; }
21        10 références
22        public string Département { get; set; }
23    }
24 }
```

Personnel.CS (Classe de modèle)

Exemple : Utilisation des vues partielles

```
10  public class PersonnelController : Controller
11  { // GET: Personnel
12      List<Personnel> ListeEnseignants = new List<Personnel>
13      {
14          new Personnel{ Nom = "Tardif", Prénom = "David", sexe = "Masculin", Age =55, AnnéeAncienneté=29, Département= "Informatique"},
15          new Personnel{ Nom = "Bérubé", Prénom = "Frédéric", sexe = "Masculin", Age =43, AnnéeAncienneté=15, Département= "Mathématique"},
16          new Personnel{ Nom = "Gagnon", Prénom = "Lucie", sexe = "Féminin", Age =37, AnnéeAncienneté=10, Département= "Administration"}
17      };
18      List<Personnel> ListeProfessionnels = new List<Personnel>
19      {
20          new Personnel{ Nom = "Laliberté", Prénom = "Julie", sexe = "Féminin", Age =40, AnnéeAncienneté=17, Département= ""},
21          new Personnel{ Nom = "Laverdière", Prénom = "Nathalie", sexe = "Féminin", Age =35, AnnéeAncienneté=11, Département= ""},
22          new Personnel{ Nom = "tremblay", Prénom = "Éric", sexe = "Masculin", Age =57, AnnéeAncienneté=30, Département= ""}
23      };
24      public ActionResult Enseignants()
25      {
26          return View(ListeEnseignants);
27      }
28      public ActionResult EnseignantsVuePartielle()
29      {
30          return View(ListeEnseignants);
31      }
32      public ActionResult Professionnels()
33      {
34          return View(ListeProfessionnels);
35      }
36  }
```

PersonnelController.CS (Contrôleur)

Exemple : Utilisation des vues partielles

```
1 @model IEnumerable<SimulationWebApp.Models.Personnel>
2 @{
3     ViewBag.Title = "EnseignantsVuePartielle";
4 }
5 <h2></h2>
6 <div class="panel panel-primary">
7     <div class="panel panel-heading">Liste des enseignants utilisant une vue partielle</div>
8     <div class="panel panel-body">
9         <table class="table">
10             <tr>
11                 <th>
12                     @Html.DisplayNameFor(model => model.Nom)
13                 </th>
14                 <th>
15                     @Html.DisplayNameFor(model => model.Prénom)
16                 </th>
17                 <th>
18                     @Html.DisplayNameFor(model => model.sexe)
19                 </th>
20                 <th>
21                     @Html.DisplayNameFor(model => model.Age)
22                 </th>
23                 <th>
24                     Ancienneté
25                 </th>
26                 <th>
27                     @Html.DisplayNameFor(model => model.Département)
28                 </th>
29             </tr>
30             @foreach (var item in Model)
31             {
32                 @Html.Partial("_Personnel", item)
33             }
34         </table>
35     </div>
36 </div>
```

EnseignantsVuePartielle.cshtml(Vue partielle)

```
1 @model SimulationWebApp.Models.Personnel
2
3 <tr>
4     <td>
5         @Html.DisplayFor(modelItem => Model.Nom)
6     </td>
7     <td>
8         @Html.DisplayFor(modelItem => Model.Prénom)
9     </td>
10    <td>
11        @Html.DisplayFor(modelItem => Model.sexe)
12    </td>
13    <td>
14        @Html.DisplayFor(modelItem => Model.Age)
15    </td>
16    <td>
17        @Html.DisplayFor(modelItem => Model.Ancienneté)
18    </td>
19    <td>
20        @Html.DisplayFor(modelItem => Model.Département)
21    </td>
22 </tr>
```

_Personnel.cshtml (Vue partielle)

Exemple : Utilisation des vues partielles

```
1 @model IEnumerable<SimulationWebApp.Models.Personnel>
2 @{
3     ViewBag.Title = "ProfessionnelsVuePartielle";
4 }
5 <h2></h2>
6 <div class="panel panel-primary">
7     <div class="panel panel-heading">Liste des professionnels utilisant une vue partielle</div>
8     <div class="panel panel-body">
9
10         <table class="table">
11             <tr>
12                 <th>
13                     @Html.DisplayNameFor(model => model.Nom)
14                 </th>
15                 <th>
16                     @Html.DisplayNameFor(model => model.Prénom)
17                 </th>
18                 <th>
19                     @Html.DisplayNameFor(model => model.sexe)
20                 </th>
21                 <th>
22                     @Html.DisplayNameFor(model => model.Age)
23                 </th>
24                 <th>
25                     Ancienneté
26                 </th>
27             </tr>
28             @foreach (var item in Model)
29             {
30                 @Html.Partial("_Personnel", item)
31             }
32         </table>
33     </div>
34 </div>
35 ==
```

Professionnel.cshtml (Vue partielle)

```
1 @model SimulationWebApp.Models.Personnel
2
3 <tr>
4     <td>
5         @Html.DisplayFor(modelItem => Model.Nom)
6     </td>
7     <td>
8         @Html.DisplayFor(modelItem => Model.Prénom)
9     </td>
10    <td>
11        @Html.DisplayFor(modelItem => Model.sexe)
12    </td>
13    <td>
14        @Html.DisplayFor(modelItem => Model.Age)
15    </td>
16    <td>
17        @Html.DisplayFor(modelItem => Model.Ancienneté)
18    </td>
19    <td>
20        @Html.DisplayFor(modelItem => Model.Département)
21    </td>
22 </tr>
23
```

_Personnel.cshtml (Vue partielle)

Exemple : Affichage des vues

Liste des enseignants utilisant une vue partielle

Nom	Prénom	sexe	Age	Ancienneté	Département
Tardif	David	Masculin	55	29	Informatique
Bérubé	Frédéric	Masculin	43	15	Mathématique
Gagnon	Lucie	Femenin	37	10	Administration

Liste des professionnels utilisant une vue partielle

Nom	Prénom	sexe	Age	Ancienneté
Laliberté	Julie	Féminin	40	17
Laverdière	Nathalie	Féminin	35	11
tremblay	Éric	Masculin	57	30