**Aides pour application web MVC Visual Studio 2015**

[1) Utiliser la classe Metadata 1](#_Toc32928032)

[2) Ajouter la liste des emplacements ou des options dans le Details d’un séjour 1](#_Toc32928033)

[3) Appeler une méthode du model dans une view 1](#_Toc32928034)

[4) Calculer la différence entre deux dates (en jours, heures, minutes…) 1](#_Toc32928035)

[5) Implémenter l’algo de la méthode calculFactureEmplacements() 2](#_Toc32928036)

[6) Implémenter l’algo de la méthode calculFactureOptions() 2](#_Toc32928037)

[7) Envoyer une donnée depuis le controller vers la view 2](#_Toc32928038)

[8) Créer un bouton avec un lien 2](#_Toc32928039)

[9) Envoyer une donnée depuis le controller vers la view 2](#_Toc32928040)

[10) Créer un formulaire, exemple de code 3](#_Toc32928041)

[11) Internationalisation de l’application 3](#_Toc32928042)

## Utiliser la classe Metadata

Pour modifier l’affichage d’un champ, le rendre obligatoire, vérifier les données saisies... il faut créer une classe Metadata et une classe PartialsClass.

<https://docs.microsoft.com/en-us/aspnet/mvc/overview/getting-started/database-first-development/enhancing-data-validation>

Faire une recherche pour afficher un calendrier pour saisir une date.

## Ajouter la liste des emplacements ou des options dans le Details d’un séjour

Voir tuto : <https://docs.microsoft.com/en-us/aspnet/mvc/overview/getting-started/database-first-development/customizing-a-view>

## Appeler une méthode du model dans une view

* Créer une méthode dans votre model. Par exemple, une méthode calculDuree() dans le model de Sejour qui calcule la durée du séjour. Cette méthode retourne une valeur entière (int) du nombre de jours.
* Afficher le résultat de la méthode dans une view. Par exemple dans la view Details de Sejour :

@Model.calculDuree()

## Calculer la différence entre deux dates (en jours, heures, minutes…)

Rechercher sur la méthode Subtract.

## Implémenter l’algo de la méthode calculFactureEmplacements()

Variable double prix

Variable int duree

duree 🡨 calculDuree()

**Pour chaque** emplacement de la collection des emplacements

prix 🡨 prix + (prix de l’emplacement \* duree)

**Retourner** prix \* coef réduction du séjour

*Boucle pour parcourir une collection en C# exemple :*

foreach (Emplacement unEmplacement in Emplacements)

{

}

## Implémenter l’algo de la méthode calculFactureOptions()

Variable double prix

Variable int duree

duree 🡨 calculDuree()

**Pour chaque** option de la collection des options

Variable int dureeLocation

dureeLocation 🡨 Duree de location de l’option

prix 🡨 prix + (prix de l’option \* dureeLocation)

**Retourner** prix

## Envoyer une donnée depuis le controller vers la view

Rechercher sur l’utilisation de ViewBag ou ViewData.

## Créer un bouton avec un lien

@Html.ActionLink("Créer un client", "Create", "Clients", null, new { @class = "btn btn-primary btn-large" })

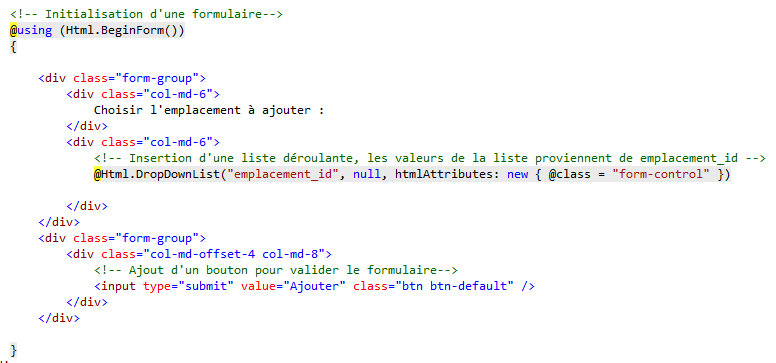
Le code ci-dessus affiche un bouton “Créer un client” qui renvoie vers la view Create de Clients.

## Envoyer une donnée depuis le controller vers la view

Rechercher sur l’utilisation de ViewBag ou ViewData.

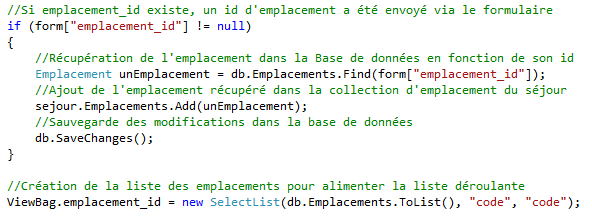
## Créer un formulaire, exemple de code

*Partie View*



Il est possible d’ajouter des Textbox, RadioButton etc… Voir @Html.

*Partie Controller*



## Internationalisation de l’application

Tuto : <http://rdonfack.developpez.com/tutoriels/dotnet/internationalisation-application-asp-net-mvc-3-razor/>