

Module ASP.NET

Chapitre 4

Pages maitres

Christian ALLEMAND – 2011/2012

Contenu

Cours	3
3 – Les pages maitres	3
3.1.1 – ContentPlaceHolder	3
3.2 – Les pages enfants	3
3.2.1 – Création d’une page enfant	3
3.2.2 - Ajouter d’un Contrôle personnalisé dans la page enfant	4
3.3 - Ordre des événements PageLoad	4
TP	4

Cours

4.1 – Les pages maitres

Lorsque l'application Web contient plusieurs pages qui doivent partager le même Design et des fonctionnalités identiques, on utilise une page maitre qui sert de "fond de page".

Les parties communes à toutes les pages sont placées dans la page maitre, et les pages enfants se limitent à définir leurs spécificités.

4.1.1 – Création d'une page maitre

Une page maitre porte l'extension .Master

Dans un projet type application Web, on ajoute un élément de type **Page Maitre** (nommé par défaut Site1.Master) Coté Design, on obtient le code suivant :

```
<%@ Master Language="VB" AutoEventWireup="false" CodeBehind="Site1.master.vb"
Inherits="PagesMaitres.Site1" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
  <asp:ContentPlaceHolder ID="head" runat="server"></asp:ContentPlaceHolder>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <h1>PAGE MAITRE !</h1>
      <asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">

      </asp:ContentPlaceHolder>
    </div>
  </form>
</body>
</html>
```

On remarquera que la directive de page indique Master à la place de Page pour une page Web standard.

Le texte PAGE MAITRE sera visible dans toutes les pages enfant.

On pourrait ajouter un menu, des liens, des images, ... communes à toutes les pages.

4.1.1 – ContentPlaceHolder

Cette balise crée un emplacement pour du contenu qui sera déclaré (éventuellement) dans la page enfant.

La balise ContentPlaceHolder doit impérativement comporter un **Id** qui servira à y accéder depuis la page enfant.

4.2 – Les pages enfants

Comme pour un WebForm standard, une page enfant contient une partie Design et une partie Code Behind.

C'est une classe dérivée de la classe System.Web.UI.Page

4.2.1 – Création d'une page enfant

On ajoute un élément de type **Web Form avec page maitre** (nommé par défaut Site1.Master).

Après avoir choisi le nom, Visual Studio vous demande de choisir la page maitre à utiliser en vous proposant celles existant dans le projet actif.

Côté Design, nous obtenons le code suivant :

```
<%@ Page Title="" Language="vb" AutoEventWireup="false" MasterPageFile="~/Site1.Master"
CodeBehind="WebForm1.aspx.vb" Inherits="PagesMaitres.WebForm1" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
</asp:Content>
```

La directive de page Contient un attribut supplémentaire **MasterPageFile** qui indique le nom de la page maitre utilisée.

Cette valeur peut être changée à tout moment pour utiliser une page maitre différente.

On remarquera que les balises **head**, **html**, **body** ont disparu : cela est normal car elles sont déjà définies dans la page maître.

La page ne contient que 2 balises **<asp :Content>**.

C'est dans ces balises que l'on va ajouter les éléments propres à notre page.

La balise **Content** contient un attribut **ContentPlaceHolderID** dont la valeur doit contenir l'**ID** du **ContentPlaceHolder** correspondant de la page maitre.

4.2.2 - Ajouter un Contrôle utilisateur dans la page enfant

On peut ajouter un Contrôle utilisateur dans la page enfant de la façon suivante :

```
<%@ Page Title="" Language="vb" AutoEventWireup="false" MasterPageFile="~/Site1.Master"
CodeBehind="WebForm1.aspx.vb" Inherits="PagesMaitres.WebForm1" %>
<%@ register Src="~/WebUserControl1.ascx" TagName="MyUserControl" TagPrefix="myUcs" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
  <myUcs:myUserControl id="uc1" runat="server"></myUcs:myUserControl>
</asp:Content>
```

4.3 - Ordre des événements PageLoad

Les événements **Load** sont d'abord déclenchés dans les pages enfants.

Si la page enfant contient un User Control l'événement Load du User Control sera déclenché en dernier

- 1 – Load de la page enfant
- 2 – Load de la page maitre
- 3 – Load du User Control

TP

Créer une page maitre et 2 pages enfants

Sur la page maitre, créer un "header" avec un lien vers chaque page enfant.