

TP 2

Site Web + Base de données

(Origine Microsoft)

1 Introduction

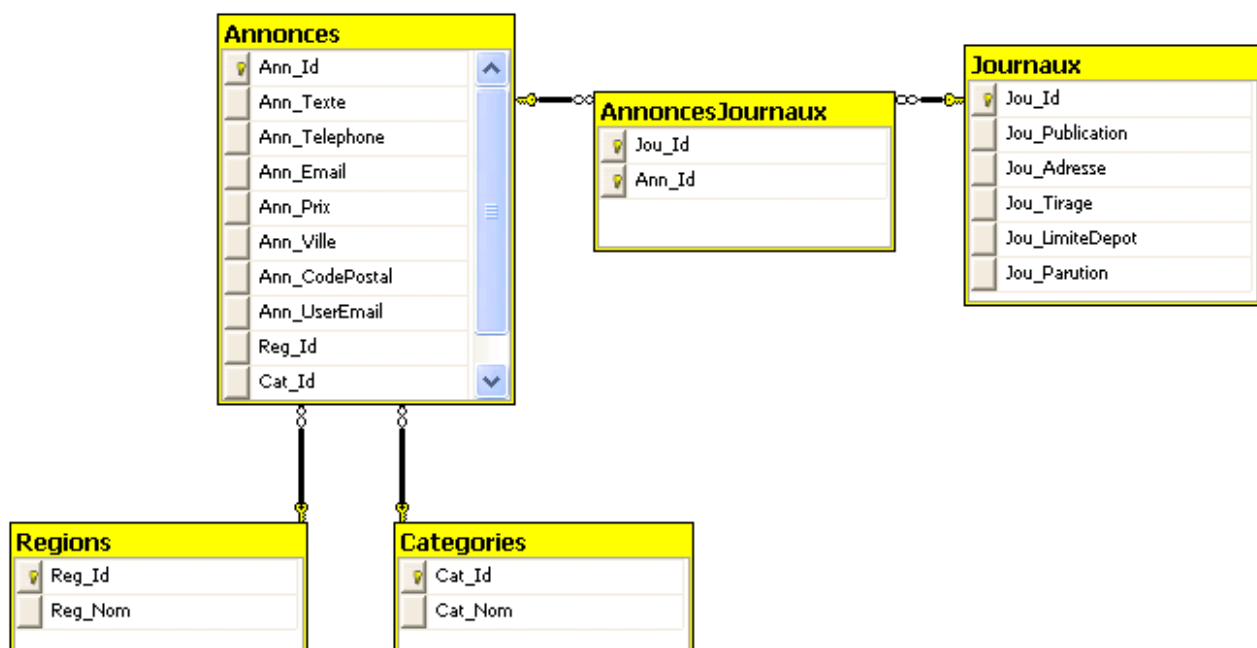
1.1 Contexte fonctionnel

L'objectif de ce TP est de construire pas à pas un site web de petites annonces (type bonnes affaires) proposant des GridView services de :

- dépôt d'une annonce en ligne.
- consultation de toutes les annonces publiées.
- gestion du compte des utilisateurs du site qui souhaitent sélectionner des annonces et revenir régulièrement sur la consultation de leur sélection.

Contexte fonctionnel

Vous allez travailler avec une base de données existante de type SQL Server, appelée **AffairesSansRisque** et dont voici le schéma :



On distingue plusieurs besoins :

1. Consultation des annonces :

Cette fonctionnalité est accessible par tout le monde.

Vous allez mettre en place plusieurs formats de consultation :

- Consultation d'une liste des annonces publiées par région.

Choix de la région

Format d'une annonce.

Plan du site | Contactez-nous | Ajoutez aux favoris

Qui sont les MSDN Regional directors ?

Accueil + Consulter nos annonces

Espace membre

Nos Annonces

Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

Rhone Alpes

Annecy (74000) dumoulin@anix.fr
Vend appartement neuf dans coin super tranquille

Annecy (74000) toto@toto.com
vend échelle plus bonne à rien

Cran-Gevrier (74000) mathilde@bien.com
vend maison à la campagne avec vue et grand parc arborisé. Très bien entretenue.

Chambéry (73500) jules@vernes.fr
recherche nounou désespérément

Lyon (69000) bfeedotoff@agilcom.info
vend moto exceptionnelle superbe merveilleuse que vous adorerez

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?

AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

- Consultation d'une liste des annonces et des publications associées dans les journaux.

Plan du site | Contactez-nous | Ajoutez aux favoris

Qui sont les MSDN Regional directors ?

Accueil + Consulter nos annonces + Consulter les publications

Espace membre

Nos Annonces

Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Consulter les publications
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

Email	Ville	Code postal
Sélectionner gegefernand@hotmail.com	Paris	75000
Sélectionner valerie.dupont@yahoo.com	Marseille	13000
Sélectionner dumoulin@anix.fr	Annecy	74000
Sélectionner fred@hotmail.com	Région Paris	76399
Sélectionner cdubois@agilcom.fr	Banlieue Paris	77021
Sélectionner estelle@hotmail.com	Aix-en-Provence	13800
Sélectionner toto@toto.com	Annecy	74000
Sélectionner mathilde@bien.com	Cran-Gevrier	74000
Sélectionner jules@vernes.fr	Marseille	13000
Sélectionner emmanuelle@bibi.com	non défini	

1 2

Publication	Tirage	Limite dépôt	Parution
La Tortue déchainée	340000	Mercredi 22h	Vendredi
Toutes les annonces classées	34000	Mardi 15h	Mercredi
Le premier prix	45000	Mercredi 20h	Jeudi
Bon courage	20000	Samedi 15h	Lundi

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?

AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

Publications de l'annonce publiée par Valérie Dupont.

2. Mise à jour des annonces :

Cette fonctionnalité concerne uniquement les administrateurs du site.

Vous allez leur mettre à disposition une liste complète des annonces :

Adresse <http://localhost:3282/AffairesSansRisque/AdministrerAnnonces.aspx>

Plan du site | Contactez-nous | Ajoutez aux favoris

Qui sont les MSDN Regional directors ?

Accueil + Administrer les annonces

Espace membre

Nos Annonces

Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

Email	Ville	Code postal	Détails
gegefernand@hotmail.com	Paris	75000	Détails...
valerie.dupont@yahoo.com	Marseille	13000	Détails...
dumoulin@anix.fr	Annecy	74000	Détails...
fred@hotmail.com	Région Paris	76399	Détails...
cdubois@agilcom.fr	Banlieue Paris	77021	Détails...
estelle@hotmail.com	Aix-en-Provence	13800	Détails...
toto@toto.com	Annecy	74000	Détails...
mathilde@bien.com	Cran-Gevrier	74000	Détails...
jules@vernes.fr	Marseille	13000	Détails...
emmanuelle@bibi.com	non défini		Détails...

1 2

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?

AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

et leur donner la possibilité d'éditer le détails d'une annonce dans une fenêtre séparée :

Adresse <http://localhost:3282/AffairesSansRisque/DetailsAnnonce.aspx?ID=2>

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires Sans Risque **AGIL.COM** Spécialistes des technos .NET
Gestion de la relation client MSDN Regional Director depuis 10 ans

Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

Id de l'annonce : 2

Texte de l'annonce : Vend coccinelle chouquette

N° de téléphone : 0423456783

Adresse de messagerie : valerie.dupont@yahoo.com

Montant : à débattre

Ville : Marseille

CodePostal : 13000

@ Utilisateur du site :

Région : Midi Pyrenees

Catégorie : Véhicule

Mettre à jour | Annuler

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?
AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

3. Publication d'une nouvelle annonce :

Cette fonctionnalité concerne uniquement les utilisateurs authentifiés dans l'application (l'authentification sera mise en place dans l'atelier suivant).

Vous allez construire un formulaire de saisie d'une nouvelle annonce :

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires Sans Risque **AGIL.COM** Smart talk, straight answers
Qui sont les MSDN Regional directors ?

Accueil + Diffuser une annonce

Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

Texte de l'annonce (250 caractères max)

Catégorie : Emploi

Région : Alsace

Prix (proposé) :

N° de téléphone :

@ :

Code Postal :

Ville :

Valider votre annonce | Annuler

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?
AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

1.2 Contexte technique

ASP.NET fournit un nouveau type de contrôles d'accès aux données qu'on appelle des contrôles **Source de données**, qui permet de relier très simplement, notamment en mode déclaratif, une source de données à des contrôles d'affichage graphique avancés.

A la fin de ce TP, vous saurez comment :

- Etablir une connexion sur une source de données SQL Server Express avec Visual Studio.
- Utiliser le nouveau contrôle Source de données SqlDataSource (le contrôle ObjectDataSource est explicité à l'occasion de l'atelier 6 traitant de la gestion des profils utilisateurs).

- Utiliser les nouveaux contrôles d'affichage GridView, DetailsView et FormView.

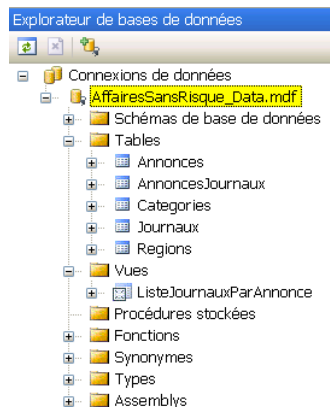
Ce TP se base sur l'utilisation de SQL Server. Vous pouvez vous référer à la procédure d'installation jointe à ce tutorial pour les détails sur le téléchargement et l'installation du produit.

Les fichiers utiles, auxquels font référence les exercices sont disponibles dans le répertoire **..qui vous est fourni**.

2 Se connecter à une source de données

Contexte fonctionnel

La base de données, **AffairesSansRisque**, est de type SQL Server et vous est fournie dans le répertoire **fourni**. L'objectif est d'établir une connexion sur cette base de données en vue de l'attacher au moteur SQL Server Express, et de prendre connaissance de son contenu.



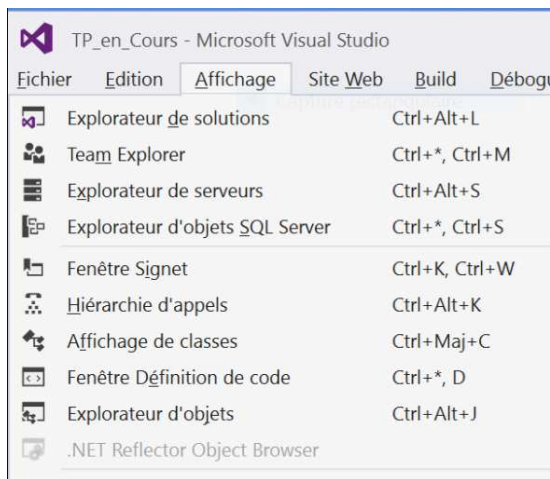
Déroulement de l'exercice :

1. Ouvrez le projet précédent réalisé lors du premier TP
 - Menu **Fichier > Ouvrir le site web**.
2. Créez un nouveau dossier dans votre application web appelé App_Data :
 - Faites un clic droit sur la racine du projet dans l'**Explorateur de solutions > Ajouter un nouveau Dossier >**
 - Nommer le Dossier **App_Data**



App_Data est l'un des nouveaux répertoires spéciaux de ASP.NET (En effet, il suffit de déposer le fichier de la base de données dans ce répertoire et ASP.NET s'occupera pour vous d'attacher dynamiquement la base de données à SQL Server (si elle l'est déjà, il se contente d'utiliser celle en place).

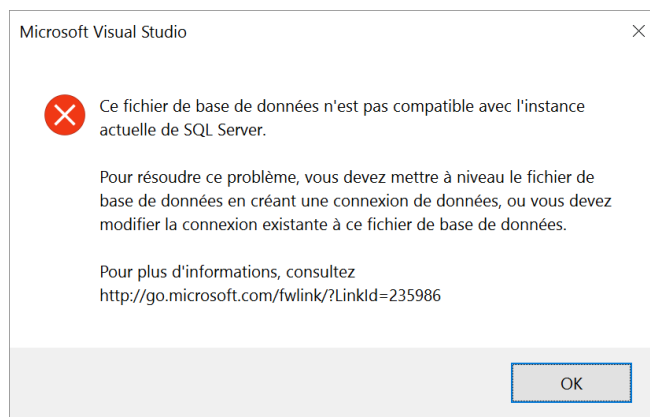
3. Récupérez la base de données existante dans le répertoire **..donné** :
 - Faites un copier des fichiers **AffairesSansRisque.mdf** et **AffairesSansRisque_log.ldf** depuis le répertoire **donné**.
 - Coller les fichiers sous le répertoire **..\AffairesSansRisque\App_Data** créé au point 2.
4. Affichez l'explorateur de serveurs de données de Visual Studio :
 - Menu **Affichage > Explorateur de serveurs**.



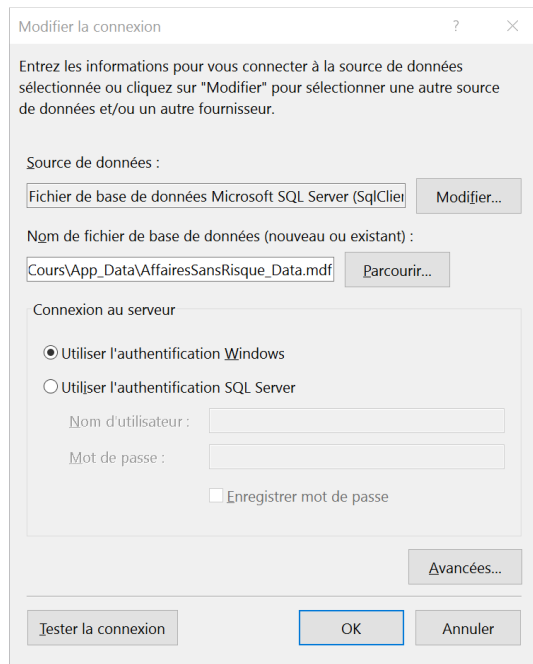
- Ouvrir "Connexions de données".
- Vous devez voir apparaître votre base de données AffairesSansRisque.mdf

5. Mise à jour de votre base de données

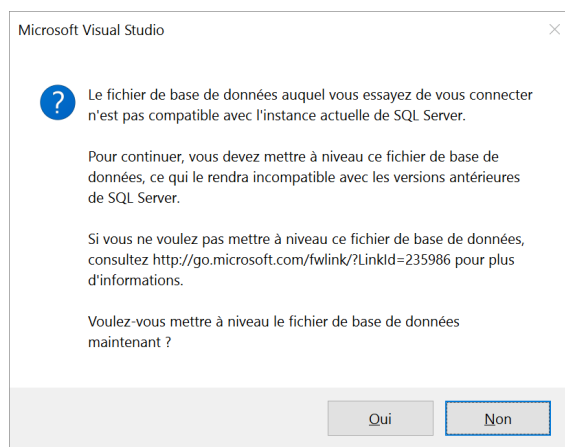
- Vous devez voir une petite croix rouge à côté de votre nom de base de données. Cela signifie que votre base de données n'est pas à jour.
- Essayer de déplier votre base de données toujours dans l'Explorateur de serveurs. Vous devez voir apparaître le schéma suivant :



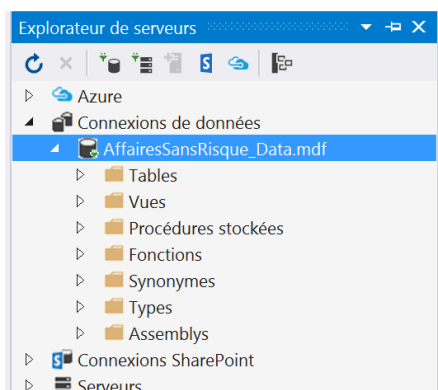
- Cliquer sur OK
- Sélectionner maintenant la base de données "AffairesSansRisque.mdf" et à l'aide du bouton contextuel sélectionner "Modifier la connexion"
- Une fenêtre doit s'ouvrir



- f. Cliquer sur le bouton "OK"
- g. Une fenêtre doit s'afficher vous indiquant que votre base de données n'est pas compatible avec l'instance de SQL Server.

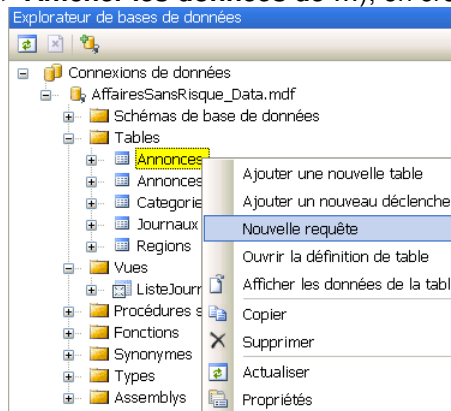


- h. Appuyer sur le bouton "Oui" pour faire la mise à jour.
- i. Votre base de données doit apparaître sous la forme :



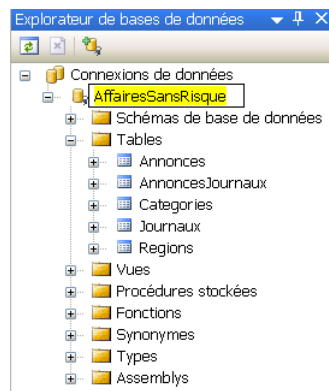


Observez la connexion ajoutée à l'**Explorateur de serveurs** de Visual Studio. Vous pouvez consulter tous les éléments de la base (tables, vues, etc...), afficher leur description (clic droit > **Ouvrir la définition de...**), consulter les données (clic droit > **Afficher les données de ...**), en créer de nouveaux etc...



Juste pour conclure...En fait, depuis l'Explorateur de solutions, si vous aviez double cliqué sur **AffairesSansRisque.mdf** dans le répertoire **App_Data**, Visual Studio vous aurait automatiquement ouvert son **Explorateur de bases de données** et vous aurait créé automatiquement la connexion vers le fichier. Quand nous vous disions que ce répertoire **App_Data** n'est pas un répertoire comme les autres...

- Faites un clic droit sur **AffairesSansRisque.mdf** >



- Faites un clic droit sur la table **Annonces** > **Afficher les données de la table** pour vérifier que la source de données contient déjà des annonces publiées.

Annonces: Requ...ISQUE_DATA.MDF									
	Ann_Id	Ann_Texte	Ann_Telephone	Ann_Email	Ann_Prix	Ann_Ville	Ann_CodePo...	Ann_UserEmail	Req_Id Cat_Id
	1	Recherche jar...	0119304885	gegeferman...	à discuter	Paris	75000	NULL	1 1
	2	Vend coccinel...	0423456783	valerie.dup...	à débattre	Marseille	13000	NULL	13 2
	3	Vend apparte...	0450789800	dumoulin@...	à débattre	Annecy	74000	NULL	20 3
	4	Vend chaton ...	0129384779	fred@hotmail...	50€	Région Paris	76399	NULL	1 7
	5	Recherche no...	0138477724	cdubois@a...	à discuter	Banlieue Paris	77021	NULL	1 1
	6	Cherche à ap...	0448938991	estelle@hotmail...	pour rien	Aix-en-Provence	13800	NULL	13 6
	7	vend échelle ...	0450692000	toto@toto.c...	10	Annecy	74000	NULL	20 5
	8	vend maison ...	0450983872	mathilde@...	100000	Cran-Gevrier	74000	NULL	20 3
	9	vend tondeuse	0423456789	jules@vern...	800€	Marseille	13000	NULL	13 4
	12	Vend canapé	0190908090	emmanuel...	non défini	non défini	NULL	NULL	1 7
	13	recherche no...	0609089922	jules@vern...	à voir	Chambéry	73500	NULL	20 1
	14	vend moto ex...	0609090900	bfedotoff@...	10€	Lyon	69000	NULL	20 2
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL NULL

3 Afficher des données

Dans cet exercice, vous allez apprendre à :

- Utiliser le contrôle de source de données SqlDataSource.
- Afficher les données d'une table dans des contrôles DropDownList, DataList et GridView.
- Lier les affichages des contrôles entre eux.

Contexte fonctionnel

L'application doit tout d'abord proposer une liste des annonces publiées sur le site.

- Pour lui permettre de trouver plus rapidement les annonces le concernant, l'utilisateur sélectionne une région (on pourrait imaginer d'autres critères de sélection bien sûr) et obtient la liste des annonces correspondantes :

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires Sans Risque

Smart talk, straight answers

Qui sont les MSDN Regional directors ?

Accueil + Consulter nos annonces

Espace membre

Nos Annonces

Consultez toutes nos annonces et passez vos annonces en ligne

- + Consulter nos annonces
- + Diffuser une annonce
- + Administrer les annonces
- + Qui sommes nous ?

Rhône Alpes

Anancy (74000) dumoulin@anix.fr
Vend appartement neuf dans coin super tranquille

Anancy (74000) toto@toto.com
vend échelle plus bonne à rien

Cran-Gevrier (74000) mathilde@bien.com
vend maison à la campagne avec vue et grand parc arborisé. Très bien entretenue.

Chambéry (73500) jules@vernes.fr
recherche nounou désespérément

Lyon (69000) bfeedotoff@agilcom.info
vend moto exceptionnelle superbe merveilleuse que vous adorerez

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?

AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

Format d'une annonce.

- Pour illustrer une relation du type « Commandes/Lignes de commandes », vous allez proposer également à l'utilisateur une liste des annonces et des publications associées :

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires Sans Risque

AGIL.COM

Gestion de la relation client

Spécialistes des technos .NET

MSDN Regional Director depuis 10 ans

Accueil + Consulter nos annonces + Consulter les publications

Espace membre

Nos Annonces

Consultez toutes nos annonces et passez vos annonces en ligne

- + Consulter nos annonces
- + Consulter les publications
- + Diffuser une annonce
- + Administrer les annonces
- + Qui sommes nous ?

Sélectionner Email Ville Code postal

Sélectionner	gegefermand@hotmail.com	Paris	75000
Sélectionner	valerie.dupont@yahoo.com	Marseille	13000
Sélectionner	dumoulin@anix.fr	Anancy	74000
Sélectionner	fred@hotmail.com	Région Paris	76399
Sélectionner	cdubois@agilcom.fr	Banlieue Paris	77021
Sélectionner	estelle@hotmail.com	Aix-en-Provence	13800
Sélectionner	toto@toto.com	Anancy	74000
Sélectionner	mathilde@bien.com	Cran-Gevrier	74000
Sélectionner	jules@vernes.fr	Marseille	13000
Sélectionner	emmanuelle@bibli.com	non défini	

1 2

Publication	Tirage	Limite dépôt	Parution
La Tortue déchainée	340000	Mercredi 22h	Vendredi
Toutes les annonces classées	34000	Mardi 15h	Mercredi
Le premier prix	45000	Mercredi 20h	Jeudi
Bon courage	20000	Samedi 15h	Lundi

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?

AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

Publications de l'annonce publiée par Valérie Dupont.

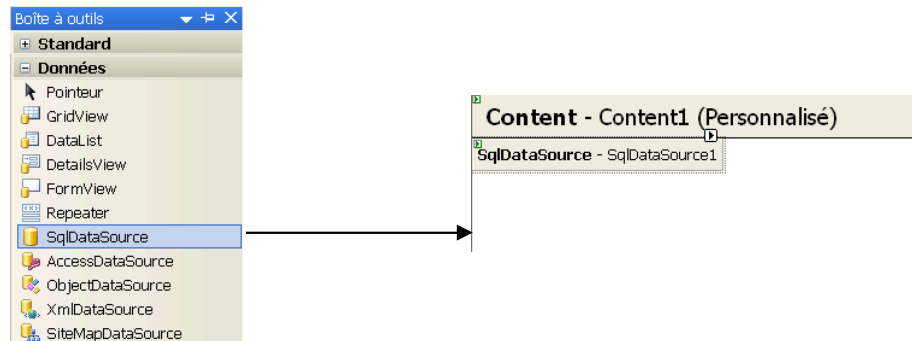
3.1 Lister les annonces par région (DropDownList/DataList)

Déroulement de l'exercice :

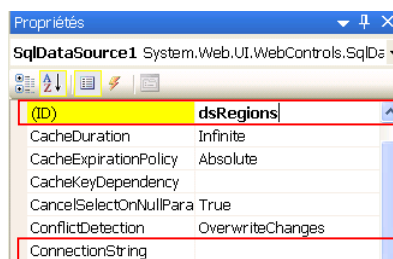
Construction de la liste des régions

La première étape consiste à dessiner une liste déroulante peuplée avec la liste des régions.

1. Ajouter le fichier "**ConsulterAnnoncesParRegion.aspx**" par simple cliquer déposer. Attention en fait vous devez avoir également le fichier de code behind associé à ce fichier présent dans votre projet.
2. Ouvrez la page **ConsulterAnnoncesParRegion.aspx** en mode **Design**.
3. Configurez une source de données vers la table **Regions** de la base **AffairesSansRisque** :
 - Faites un glisser déposer sur le contrôle **Content1** de la **Boîte à outils** > rubrique **Données** > **SqlDataSource**.



- Faites un clic droit sur le contrôle > **Propriétés** et renommez le contrôle : **dsRegions**.



Remarquez que le contrôle contient une propriété **ConnectionString** qui va contenir la chaîne de connexion à la base de données et que pour l'instant elle est vide.

- Sélectionnez le contrôle pour faire apparaître sa balise active (ou clic droit sur le contrôle > Afficher la balise active).
- Cliquez **Configurer la source de données...**



- Dans la fenêtre **Choisir votre connexion de données**, cliquez la liste déroulante et retrouvez la connexion définie à l'exercice 2 précédent : **csAffairesSansRisque**.
- Cliquez sur **Chaîne de connexion** pour observer la chaîne de connexion associée :

Configurer la source de données - dsRegions

Choisir votre connexion de données

Quelle connexion de données votre application doit-elle utiliser pour établir une connexion à la base de données ?

csAffairesSansRisque

Nouvelle connexion...

+ Chaîne de connexion

< Précédent

Suivant >

Terminer

Annuler



Regardons d'un peu plus près cette chaîne de connexion :

```
Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=|DataDirectory|\AffairesSansRisque.mdf;Integrated Security=True
```

- Vous voyez que la **Source** est le moteur MSSQLLocalDB. Le nom de celui-ci s'installe par défaut comme une instance nommée sur la machine d'où ce format du type <nom de la machine>\<nom de l'instance nommée>.
- AttachDbFilename contient le chemin complet de la base de données à attacher au moteur SQL Server. Tout le problème est que ce chemin peut être amené à changer fréquemment, par exemple lorsque vous déploierez l'application sur un serveur de production. Pour rappel, nous avons placé notre fichier dans le répertoire spécial **App_Data** de notre application web à l'exercice 2 précédent. Il suffit donc d'indiquer **|DataDirectory|<nom du fichier .mdf>** pour que le runtime ASP.NET sache où le retrouver. En prime, il va s'occuper pour nous d'attacher la base de données dynamiquement au moteur MSSQLLocalDB ...

- Cliquez sur **Suivant**.
- Dans la fenêtre **Configurer l'instruction Select**, cliquez **Spécifiez les colonnes d'une table ou d'une vue** et choisissez la table **Regions** dans la liste déroulante.
- Sélectionnez tous les champs de la table :

Configurer la source de données - dsRegions

Configurer l'instruction Select

SQL

Comment souhaitez-vous extraire les données de votre base de données ?

☐ Spécifiez une instruction SQL personnalisée ou une procédure stockée

☒ Spécifiez les colonnes d'une table ou d'une vue

Nom : Regions

Colonnes :

☐ *

☒ Reg_Id

☒ Reg_Nom

☐ Retourner seulement des lignes uniques

WHERE...

ORDER BY...

Options avancées

Instruction SELECT :

SELECT [Reg_Id], [Reg_Nom] FROM [Regions]

< Précédent Suivant > Terminer Annuler

Option pour travailler avec une procédure stockée (c'est évidemment mieux) ou affichez le Générateur de requête de VWD.

Options avancées pour générer notamment les autres requêtes CRUD vers la base de données.

Instruction SELECT générée sur la base des éléments sélectionnés.

- Cliquez sur **Suivant**.
- Dans la fenêtre **Tester la requête**, cliquez le bouton **Tester la requête** pour la valider.

Configurer la source de données - dsRegions

Tester la requête

Pour apercevoir les données retournées par cette source de données, cliquez sur Tester la requête. Pour terminer cet Assistant, cliquez sur Terminer.

Reg_Id	Reg_Nom
1	Alsace
2	Aquitaine
3	Auvergne
4	Basse Normandie
5	Bourgogne
6	Bretagne
7	Centre
8	Champagne
9	Franche Comte
10	Languedoc Roussillon
11	Limousin
12	Lorraine
13	Midi Pyrenees

Tester la requête

Instruction SELECT :

SELECT [Reg_Id], [Reg_Nom] FROM [Regions]

≤ Précédent Suivant > Terminer Annuler

- Cliquez sur **Terminer**.
4. Sauvegardez la page **ConsulterAnnoncesParRegion.aspx**.
 5. Affichez le fichier Web.config.

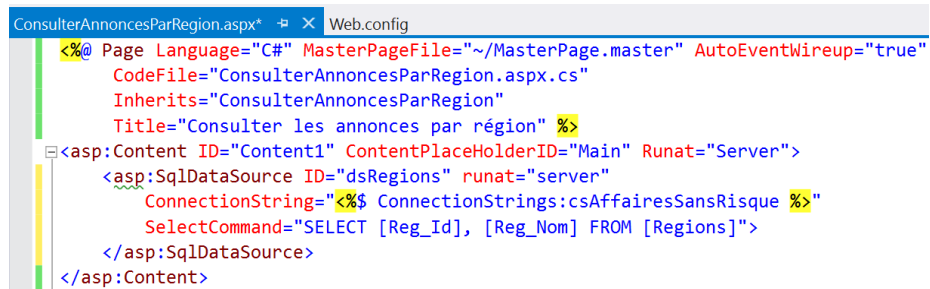
```
<connectionStrings>
  <add name="csAffairesSansRisque"
        connectionString="Data Source=(LocalDB)\MSSQLLocalDB;
                          AttachDbFilename=|DataDirectory|\AffairesSansRisque.mdf;
                          Integrated Security=True"
        providerName="System.Data.SqlClient" />
</connectionStrings>
```

Attention la chaîne de connexion à votre base de données peut être différente pour votre projet. Cela dépend de la version SQL Server que vous utilisez.



⚡ Notez qu'une nouvelle clé a été générée dans la section **<connectionStrings>** pour la chaîne de connexion que vous venez de paramétrer. La chaîne de connexion est donc très facilement modifiable en cas de besoin et ce même (et surtout) par le futur Administrateur de votre application web (qui n'aura pas nécessairement de connaissances ASP.NET).

- Ouvrez la page **ConsulterAnnoncesParRegion.aspx** en mode **Source**.



```
<%@ Page Language="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true"
CodeFile="ConsulterAnnoncesParRegion.aspx.cs"
Inherits="ConsulterAnnoncesParRegion"
Title="Consulter les annonces par région" %>
<asp:Content ID="Content1" ContentPlaceHolderID="Main" Runat="Server">
  <asp:SqlDataSource ID="dsRegions" runat="server"
    ConnectionString="<%= $ConnectionStrings:csAffairesSansRisque %>"
    SelectCommand="SELECT [Reg_Id], [Reg_Nom] FROM [Regions]">
  </asp:SqlDataSource>
</asp:Content>
```

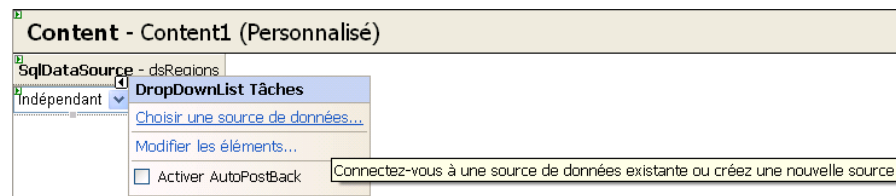


⚡ Remarquez le chargement dynamique de la chaîne de connexion à partir de la nouvelle clé générée dans le fichier de configuration de l'application. La syntaxe est simple :

<%= \$ ConnectionStrings :<nom de la clé dans le fichier Web.config> %>.

L'attribut **SelectCommand** contient les informations pour la requête **SELECT** sur la base de données.

- Basculez la page en mode **Design**.
- Dessinez maintenant un contrôle pour afficher les régions dans une liste déroulante :
 - Faites un glisser déplacer sur le contrôle **Content1** de la **Boîte à outils** > rubrique **Standard** > **DropDownList**.
 - Sélectionnez le contrôle et cliquez sur sa balise active > **Choisir une source de données**.



- Dans la fenêtre **Choisir une source de données**, sélectionnez la source de données **dsRegions** créée précédemment.
- Paramétrez l'affichage de la liste déroulante sur le champ **Reg_Nom** et indiquer **Reg_Id** comme étant la donnée à prendre en compte dans la source de données.

Assistant Configuration de source de données

Choisir une source de données

Sélectionnez une source de données :
dsRegions

Sélectionnez un champ de données à afficher dans DropDownList :
Reg_Nom

Sélectionnez un champ de données pour la valeur de DropDownList :
Reg_Id

[Actualiser le schéma](#)

OK Annuler

- Cliquez sur **OK**.
- Basculez en mode **Source** pour voir les attributs du contrôle `<asp:DropDownList>` généré par l'assistant que vous venez d'utiliser pour configurer la liste.

```

<%@ Page Language="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true"
    Title="ConsulterAnnoncesParRegion.aspx" >
<asp:Content ID="Content1" ContentPlaceHolderID="Main" Runat="Server">
    <asp:SqlDataSource ID="dsRegions" runat="server"
        ConnectionString="<%= ConnectionStrings.csAffairesSansRisque %>"
        SelectCommand="SELECT [Reg_Id], [Reg_Nom] FROM [Regions]">
    </asp:SqlDataSource>
    <asp:DropDownList ID="DropDownList1" runat="server"
        DataSourceID="dsRegions"
        DataTextField="Reg_Nom"
        DataValueField="Reg_Id">
    </asp:DropDownList>
</asp:Content>

```

9. Sauvegardez la page **ConsulterAnnoncesParRegion.aspx**.
10. Exécutez la page en faisant un clic droit > **Afficher dans le navigateur**.



Juste pour mémoire, vous n'avez pas tapé une seule ligne de code....

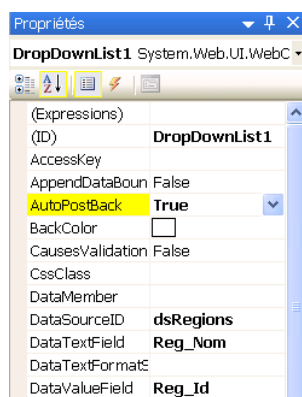
Le contrôle **SqlDataSource** (sans équivalence graphique sur la page finale) s'est occupé pour vous de faire la requête SELECT sur la base de données à partir de l'attribut **SelectCommand** et de faire la liaison avec le contrôle d'affichage **DropDownList**. C'est l'une des grosses nouveautés en matière d'accès aux données de ASP.NET.



Le rendu (notamment la couleur de la police de caractères) de la liste déroulante est lié à la description du contrôle DropDownList dans le fichier d'apparence de contrôles : **..\App_Themes\Default\Default.skin**.

11. Il reste une dernière étape très importante pour activer le déclenchement d'un aller retour sur le serveur lorsque l'utilisateur sélectionne une valeur dans la liste. En effet, vous avez peut-être remarqué lors de l'exécution de la page au point 9, qu'une sélection dans la liste ne provoquait rien. C'est le comportement par défaut du contrôle.

- Revenez sur la page en mode **Design**.
- Sélectionnez le contrôle DropDownList1 > clic droit > **Propriétés**.
- Changez **AutoPostBack** : **True** (l'option est aussi directement disponible dans la balise active du contrôle DropDownList)



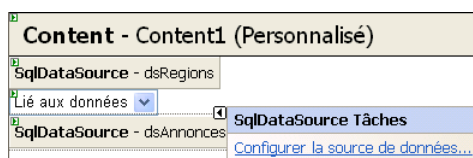
- Sauvegardez la page puis relancez l'exécution. Vous constatez dans la barre de statut d'Internet Explorer qu'il y a bien un appel du serveur lorsque vous changez de valeur dans la liste :



Construction de la liste des annonces en fonction de la région sélectionnée

La seconde étape consiste à dessiner une liste sous la forme d'une suite d'enregistrement, peuplée avec la liste des annonces correspondant à la région en cours (sélectionnée dans la liste déroulante).

1. Configurez une source de données vers la table **Annonces** de la base **AffairesSansRisque** :
 - Faites un glisser-déplacer sur le contrôle **Content1** de la **Boîte à outils** > rubrique **Données** > **SqlDataSource**.
 - Faites un clic droit sur le contrôle > **Propriétés** et renommez le contrôle : **dsAnnonces**.
 - Sélectionnez le contrôle pour faire apparaître sa balise active (ou clic droit sur le contrôle > Afficher la balise active).
 - Cliquez **Configurer la source de données...**



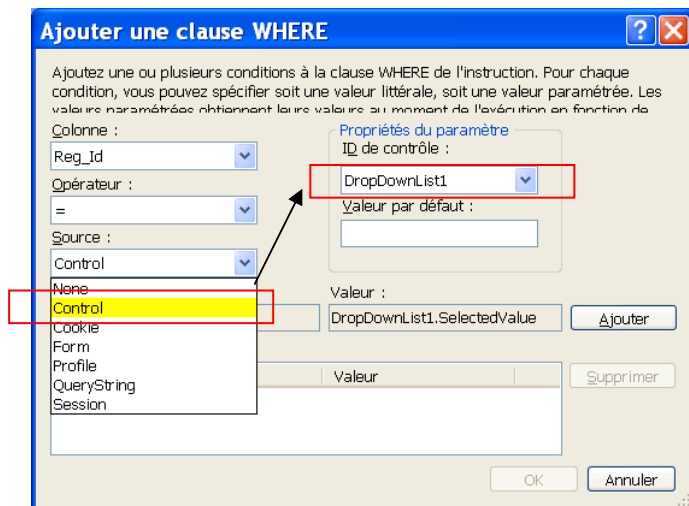
- Dans la fenêtre **Choisir votre connexion de données**, cliquez la liste déroulante et sélectionnez la même chaîne de connexion que précédemment **csAffairesSansRisque**.
- Cliquez sur **Suivant**.

- Dans la fenêtre **Configurer l'instruction Select**, cliquez **Spécifiez les colonnes d'une table ou d'une vue** et choisissez la table **Annonces** dans la liste déroulante.
- Sélectionnez tous les champs de la table :

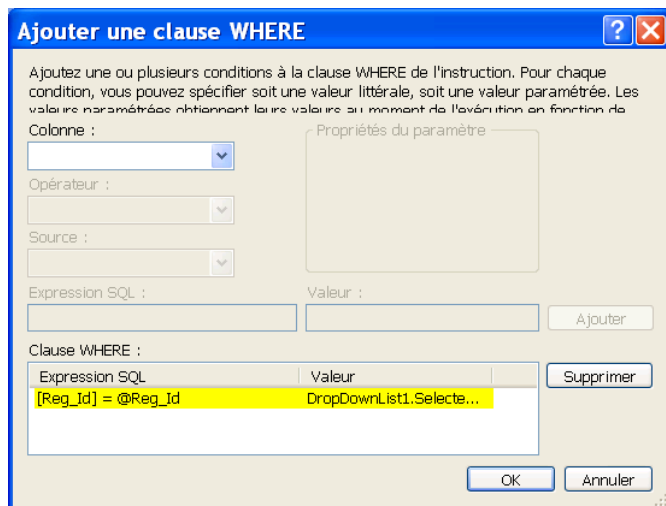
- Cliquez sur **WHERE**.
- Dans la fenêtre Ajouter une clause WHERE, sélectionnez :
 - Colonne : **Reg_Id**,
 - Opérateur : **=**,
 - Source : **Control**,
 - ID de contrôle : **DropDownList1**.



Vous êtes en train de construire une requête SELECT sur la table **Annonces** pour ne rapatrier que celles correspondant à une région particulière c'est-à-dire à une valeur du champ **Reg_Id** précise. L'assistant vous demande donc où est-ce que ASP.NET pourra trouver ce paramètre **Reg_Id** ? Si vous ouvrez la liste déroulante **Source**, vous constatez que plusieurs possibilités vous sont offertes Ici, vous lui dites qu'il trouvera la valeur du paramètre **Reg_Id** de votre requête dans un **Control** de la page, précisément dans le contrôle dont l'ID est **DropDownList1**.



- Cliquez sur **Ajouter** pour ajouter le paramètre à la liste des paramètres de la requête.



- Cliquez sur **OK**.
 - Cliquez sur **Suivant** dans la fenêtre **Configurer l'instruction Select** précédente.
- La requête est maintenant :

```
Instruction SELECT :
SELECT * FROM [Annonces] WHERE ([Reg_Id] = @Reg_Id)
```

- Dans la fenêtre **Tester la requête**, cliquez le bouton **Tester la requête** pour la valider avec Reg_Id = 1 par exemple.
 - Cliquez sur **Terminer**.
2. Sauvegardez la page **ConsulterAnnoncesParRegion.aspx**.
 3. Basculez la page **ConsulterAnnoncesParRegion.aspx** en mode **Source**.

Collection de paramètres de la requête SELECT.

```
<asp:SqlDataSource ID="dsAnnonces" runat="server"
    ConnectionString="<%$ ConnectionStrings:csAffairesSansRisque %>"
    SelectCommand="SELECT * FROM [Annonces] WHERE ([Reg_Id] = @Reg_Id)">
    <SelectParameters>
        <asp:ControlParameter ControlID="DropDownList1"
            Name="Reg_Id"
            PropertyName="SelectedValue"
            Type="Int32" />
    </SelectParameters>
</asp:SqlDataSource>
```



Remarquez le chargement dynamique de la chaîne de connexion à partir de la nouvelle clé générée dans le fichier de configuration de l'application. L'attribut **SelectCommand** contient les informations pour la requête **SELECT** sur la base de données.

Par rapport à la source de données dsRegions, vous avez également ici une collection de paramètres **<SelectParameters>** qui contient un paramètre de type Control **<asp:ControlParameter>** dont les caractéristiques sont précisées dans les attributs : **PropertyName** indique la propriété du contrôle **DropDownList1** dans laquelle ASP.NET doit récupérer la valeur du paramètre.

4. Basculez la page en mode **Design**.
5. Dessinez maintenant le contrôle graphique pour afficher les annonces :
 - Faites un glisser-déplacer sur le contrôle **Content1** de la **Boîte à outils** > rubrique **Données** > **DataList**.
 - Sélectionnez le contrôle et cliquez sur sa balise active > **Choisir une source de données : dsAnnonces**. Et Actualiser le Schéma.

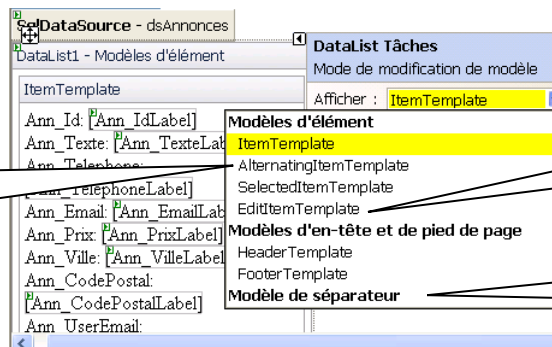
Chaque enregistrement est répété verticalement sur toute la page selon ce modèle d'affichage.



Vous constatez que le contrôle **DataList** est intelligent et interprète la source de données dont il connaît le schéma. Il en extrait le nom des champs à afficher pour chaque colonne de la table définie dans la requête **SELECT**.

Du coup, il génère un modèle simple d'affichage des données avec les en-têtes correspondantes aux noms des colonnes de la table. Vous pouvez retravailler cet affichage via les options **Générateur de propriétés** et **Modifier les modèles**. Remarquez les différents types de modèles proposés lorsque vous cliquez **Modifier les modèles** :

Vous pouvez dessiner différemment un enregistrement sur deux pour mieux les faire ressortir.



Modèle en mode d'édition des enregistrements.

Séparateur entre deux enregistrements affichés.

Voilà ce que nous souhaiterions obtenir comme format d'enregistrement :



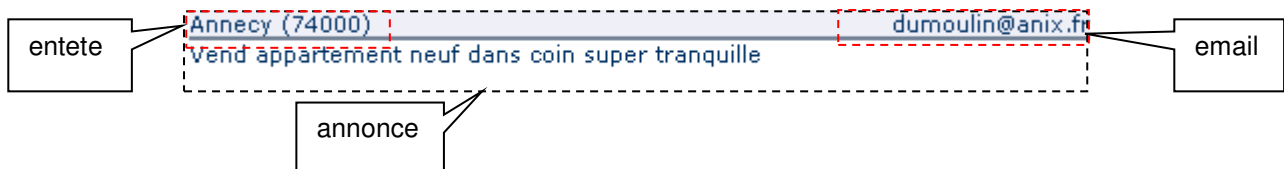
Nous allons en fait remodeler le modèle d'affichage directement dans le source de la page.

6. Basculez la page **ConsulterAnnoncesParRegion.aspx** en mode **Source**.
7. Supprimez tout ce qui se trouve à l'intérieur de la balise **<ItemTemplate>** pour supprimer le modèle généré par défaut.

```
<asp:DataList ID="DataList1" runat="server"
    DataKeyField="Ann_Id" DataSourceID="dsAnnonces">
    <ItemTemplate>

    </ItemTemplate>
</asp:DataList>
```

8. Commencez par structurer le modèle d'enregistrement avec des balises **<div>** et des classes associées :



Modifiez la balise <ItemTemplate> :	Ajoutez dans la feuille de style Default.css du thème Default :
<pre><ItemTemplate> <div id="annonce"> <div id="entete"> </div> <div id="email"> </div> </div> </ItemTemplate></pre>	<pre>/* FORMAT D'UNE ANNONCE */ #annonce { margin-top: 2.0em; background-color: #efeff7; width:450px; border-bottom: #738294 thin solid; display:block; } #entete { float:left; } #email { text-align:right; }</pre>

9. Repositionnez les contrôles serveur liés aux différentes colonnes de la table **Annonce** à l'intérieur de cette structure en balises **<div>** comme suit :

```
<asp:DataList ID="DataList1" runat="server" DataKeyField="Ann_Id" DataSourceID="dsAnnonces">

<ItemTemplate>
<div id="annonce">
```

```

<div id="entete">
  <asp:Label ID="Ann_VilleLabel" runat="server" Text='<%=# Eval("Ann_Ville") %>'
EnableTheming="false">
  </asp:Label>
  (<asp:Label ID="Ann_CodePostalLabel" runat="server" Text='<%=# Eval("Ann_CodePostal") %>'
EnableTheming="false">
    </asp:Label>)
</div>
<div id="email">
  <asp:Label ID="Ann_Email" runat="server" Text='<%=# Eval("Ann_Email") %>'></asp:Label>
</div>
</div>
<asp:Label ID="Ann_TexteLabel" runat="server" Text='<%=# Eval("Ann_Texte") %>'
EnableTheming="false"></asp:Label>
</ItemTemplate>
</asp:DataList>

```



Vous avez ici un exemple de la syntaxe à utiliser pour effectuer une liaison (databinding) entre un contrôle (<asp :DataList>) et sa source de données (DataSourceID= « dsAnnonces ») :

<préfixe :balise du contrôle propriété='<%=# **expression de liaison avec une colonne** %>' runat= "server"/>

La liaison peut se faire sur différentes propriétés du contrôle.

La ligne **<asp:Label ID="Ann_VilleLabel" runat="server" Text='<%=# Eval("Ann_Ville") %>'>** signifie que le contrôle SqlDataSource nommé dsAnnonces va récupérer la valeur de la colonne **Ann_Ville** et la lier à la propriété **Text** du label **Ann_VilleLabel**. Si cette valeur change dans la source de données, le contrôle reflètera ce changement. Voilà pourquoi on dit que le contrôle est **lié** à la source de données.



Il faut savoir que vous pouvez utiliser des expressions de liaison dans des contextes variés.

Par exemple comme l'expression d'une condition du type :

```

ForeColor=
'<%=# ((bool)Eval("Ann_OutToDate"))?System.Drawing.Color.Gray:System.Drawing.Color.Black%>'

```

La valeur de ForeColor est Gray si le champ de données Ann_OutToDate contient la valeur vraie, Black sinon.

Autre exemple : des liaisons avec des sources de données hiérarchiques (type XML) sont également supportées (à utiliser par exemple avec un contrôle Treeview).



Attention, avec la fonction **Eval()**, la liaison est automatique mais unidirectionnelle, dans le sens de la source de données vers le contrôle d'affichage. C'est-à-dire que ce dernier reflète les changements de la source mais pas l'inverse. Par contre l'inverse est tout à fait possible et c'est ce que nous utiliserons dans l'exercice de mise à jour des données, avec la fonction **Bind()** de l'ASP.NET, qui elle, est bidirectionnelle !

10. Sauvegardez la page **ConsulterAnnoncesParRegion.aspx**.

11. Exécutez la page en faisant un clic droit > **Afficher dans le navigateur**. Sélectionnez la région **Rhone Alpes** :



Vous pourriez ajouter des liens hypertextes en dessous de chaque annonce, tels que « envoyer cette annonce à un ami », « imprimer cette annonce »...etc.
Nous verrons dans l'Atelier 6 sur la gestion des profils utilisateurs, l'ajout d'un tel lien pour « Ajouter cette annonce à ma sélection ».

3.2 Lister les annonces et la liste des publications correspondantes (DataGrid/DataGrid + utilisation d'une vue)

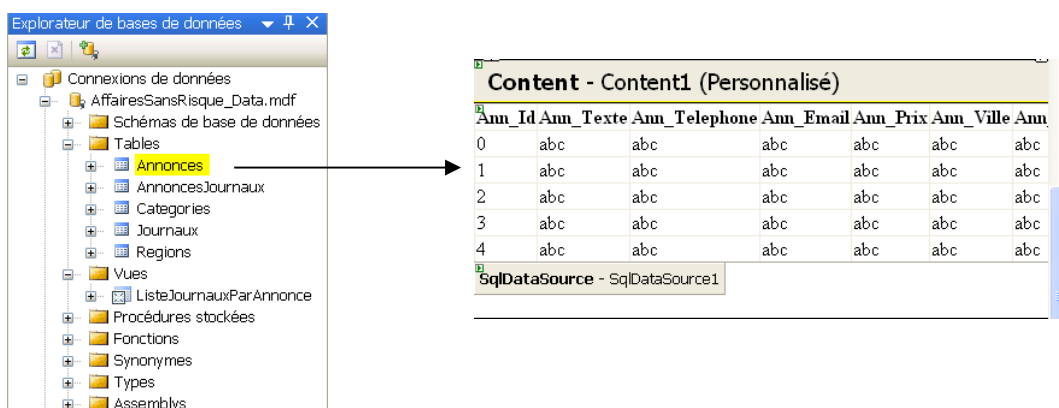
Déroulement de l'exercice :

Construction de la liste des annonces dans une grille

La première étape consiste à dessiner une liste sous la forme d'une grille, peuplée avec la liste de toutes les annonces.

***** création page *****

1. Ouvrez la page **ConsulterAnnoncesJournaux.aspx** en mode Design.
2. Ouvrez l'**Explorateur de serveurs** de Visual Studio.
3. Faites un glisser déplacer de la table **Annonces** sur le contrôle **Content1** de la page ConsulterAnnoncesJournaux.aspx.





On ne pourra pas faire plus simple et plus rapide pour générer les contrôles **SqlDataSource** et **GridView** associées à la table **Annonces**. Remarquez que la liaison des colonnes dans la grille est en place.

4. Sauvegardez la page.
5. Lancez l'exécution de la page (clic droit sur la page > Afficher dans le navigateur)

Ann_Id	Ann_Texte	Ann_Telephone	Ann_Email	Ann_Prix	Ann_Ville	Ann_CodePostal
1	Recherche jardinier	0119304885	gegefernand@hotmail.com	à discuter	Paris	75000
2	Vend coccinelle choupette	0423456783	valerie.dupont@yahoo.com	à débattre	Marseille	13000
3	Vend appartement neuf dans coin super tranquille	0450789800	dumoulin@anix.fr	à débattre	Annecy	74000
4	Vend chaton magnifique	0129384779	fred@hotmail.com	50€	Région Paris	76399
5	Recherche nounou pour garder mes enfants à domicile	0138477724	cdubois@agilcom.fr	à discuter	Banlieue Paris	77021



L'affichage est horrible mais les données sont là...c'est déjà pas mal, surtout en ne faisant qu'un glisser déplacer avec la souris !

6. Retravajons un peu le format de cette grille et de cette source de données :
 - Basculez la page en mode **Design**.
 - Renommez le contrôle de source de données **SqlDataSource** en **dsAnnonces2**.
 - Faites un glisser déplacer sur le contrôle **Content1** de la **Boîte à outils** > rubrique **Données** -> **GridView**
 - Affichez la balise active du contrôle **GridView**. A l'aide du menu contextuel. Dans la liste déroulante **Choisissez une source de données** changez **SqlDataSource1** et repointez sur le nouveau nom **dsAnnonces2**.

GridView Tâches
 Mise en forme automatique...
 Choisissez une source de données : SqlDataSource1
 Modifier les colonnes...
 Ajouter une nouvelle colonne...
 Modifier les modèles



Votre manipulation a provoqué un changement marquant pour Visual Studio qui vous demande donc si ce changement de source de données veut dire qu'il faut re-générer toutes les liaisons de champ dans la grille. Répondez **oui** (juste pour voir comment on va procéder pour récupérer la liaison du contrôle **GridView1** avec la nouvelle source de données)

Pour réactualiser la liaison, choisissez **Actualiser le schéma** dans la balise active du contrôle **GridView1**.

Répondez **Oui** à la proposition de génération des colonnes de la grille à partir du schéma de la source de données.

- Affichez la balise active du contrôle **GridView**. Cochez les options **Activer la pagination**, **Activer le tri**, **Activer la sélection**.

Tâches GridView

Mise en forme automatique...

Choisir la source de données : dsAnnonces

Configurer la source de données...

Actualiser le schéma

Modifier les colonnes...

Ajouter une nouvelle colonne...

☒ Activer la pagination

☒ Activer le tri

☐ Activer la modification

☐ Activer la suppression

☒ Activer la sélection

Modifier les modèles



Notez l'apparition sur la grille d'un bouton de sélection sur chaque ligne (Activer la sélection), une zone de pagination avec le numéro et le nombre des pages en bas à gauche de la grille (Activer la pagination), et l'apparition de liens hypertexte sur les en-têtes de colonne pour autoriser le tri (Activer le tri). Attention, il ne s'agit pas uniquement ici d'impact sur le graphisme de la grille. Le contrôle GridView va effectivement prendre en charge les actions de pagination, de tri et de sélection pour vous sans que vous ayez besoin de rajouter une seule ligne de code...

- Dans la balise active du contrôle **GridView**, cliquez **Modifier les colonnes...**
- Dans la fenêtre **Champs**, supprimez tous les champs SAUF **Ville**, **Code Postal**, **Email** et le champ **Sélectionner**.

Notez les types de champs disponibles.

Renommez les en-têtes de chaque colonne.

Changez l'ordre des colonnes avec les boutons.

Bouton pour supprimer les colonnes sélectionnées.



Les champs **Ville**, **Code Postal** et **Email** sont de type **BoundField**, c'est-à-dire que ce sont des colonnes liées aux champs de la table Annonces par une liaison du type de celle vue au point 9 du § 3.1.

Champs disponibles :

- BoundField
- Ann_Id
- Ann_Texte
- Ann_Telephone
- Ann_Email
- Ann_Prix

Page 22 sur 33

Le champ **Sélectionner** est de type **CommandField**, c'est-à-dire qu'il va générer un bouton de commande lié à l'action de sélection sur un enregistrement. Vous pouvez aussi générer des boutons pour les actions de suppression et modification si vous souhaitez travailler sur la grille en édition.

- Fermer la boîte en cliquant sur **OK**.
- Le contrôle **Content1** en mode **Design** contient maintenant :

Content - Content1 (Personnalisé)

Sélectionner	Email	Ville	Code postal
Sélectionner	abc	abc	abc
Sélectionner	abc	abc	abc
Sélectionner	abc	abc	abc
Sélectionner	abc	abc	abc
Sélectionner	abc	abc	abc
Sélectionner	abc	abc	abc
Sélectionner	abc	abc	abc
Sélectionner	abc	abc	abc
Sélectionner	abc	abc	abc
Sélectionner	abc	abc	abc

1 2

SqlDataSource - dsAnnonces

- Basculez la page en mode **Source** pour voir le flux correspondant :

```
<asp:GridView ID="GridView1" runat="server" AllowPaging="True"
    AllowSorting="True" AutoGenerateColumns="False"
    DataKeyNames="Ann_Id" DataSourceID="dsAnnonces"
    EmptyDataText="Il n'y a aucun enregistrement de données à afficher.">
  <Columns>
    <asp:CommandField ShowSelectButton="True" />
    <asp:BoundField DataField="Ann_Email" HeaderText="Email"
      SortExpression="Ann_Email" />
    <asp:BoundField DataField="Ann_Ville" HeaderText="Ville"
      SortExpression="Ann_Ville" />
    <asp:BoundField DataField="Ann_CodePostal" HeaderText="Code postal"
      SortExpression="Ann_CodePostal" />
  </Columns>
</asp:GridView>
```



Notez les attributs du contrôle **<asp :GridView>** :

- **AllowPaging** et **AllowSorting** qui activent respectivement la pagination et le tri.
- **DataSourceID** qui lie le contrôle **GridView** au contrôle de source de données.
- **AutoGenerateColumns** qui désactive la génération automatique des colonnes à l'exécution à partir des champs de la base. En effet, nous avons décidé de n'afficher que certaines d'entre elles, précisées dans la balise **<Columns>**.
- **DataKeyNames** qui définit la colonne représentant la clé primaire de l'enregistrement dans la base (l'ID de l'annonce).

Le contrôle contient ensuite une collection de colonnes dans la balise **<Columns>**. On retrouve les colonnes **BoundField** et **CommandField** que vous avez définies avec l'assistant. Vous remarquez que l'expression de liaison avec la fonction Eval() est implicite pour les contrôles **<asp :BoundField>**. On se contente d'indiquer un en-tête de colonne dans l'attribut **HeaderText** et le nom du champ de liaison dans la source de données dans **DataField**.

- Ajoutez l'ID d'apparence **tableMaitre** au contrôle GridView pour faire référence au formatage du contrôle inclus dans votre fichier d'apparence :

```
<asp:GridView SkinId="tableMaitre" ID="GridView1"
    runat="server" AllowPaging="True"
    AllowSorting="True" AutoGenerateColumns="False"
    DataKeyNames="Ann_Id" DataSourceID="dsAnnonces"
    EmptyDataText="Il n'y a aucun enregistrement de
```

- Reportez vous au fichier d'apparence **..\App_Themes\Default\Default.skin** pour observer la définition du contrôle :

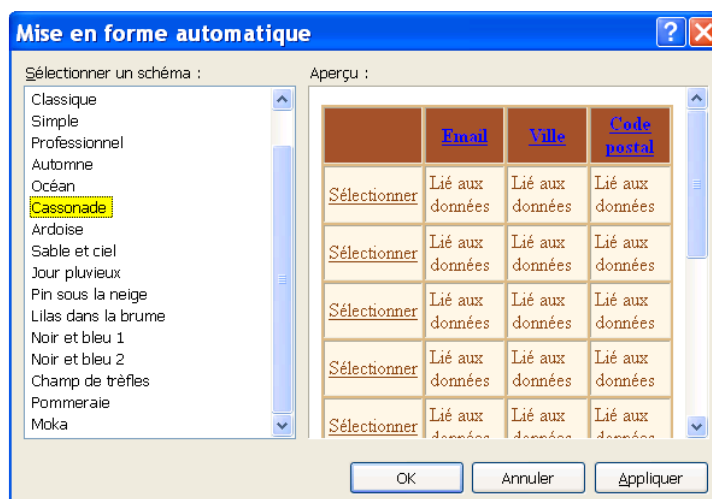
Définition dans le fichier d'apparence Default.skin	Définition dans la feuille de style
<pre><asp:GridView SkinId="tableMaitre" CssClass="gridcontent" runat="server" GridLines="none"> <SelectedRowStyle BackColor="#EFFF7" /> </asp:GridView></pre>	<pre>.gridcontent { margin-top: 1.0em; width:460px; border-right: #7b9ebd 1px solid; border-top: #7b9ebd 1px solid; border-left: #7b9ebd 1px solid; border-bottom: #7b9ebd 1px solid; }</pre>

Surligne l'enregistrement sélectionné en cours.

Bordure de la grille.



Une grille contient de multiples propriétés d'affichage que nous vous invitons à explorer (HeaderStyle, FooterStyle, EditRowStyle etc...). Vous pouvez également utiliser un format automatique à partir de la balise active du contrôle GridView > **mise en forme automatique** ...



Pour apprendre, vous pouvez appliquer des formats automatiques à titre d'exemple pour observer quelles propriétés du contrôle sont utilisées pour donner ce rendu...

- Regardez la définition du contrôle de source de données **dsAnnonces** :

```
<asp:SqlDataSource ID="dsAnnonces" runat="server" ConnectionString="<%$ Conn
DeleteCommand="DELETE FROM [Annonces] WHERE [Ann_Id] = @original_Ann_Id'
ProviderName="<%$ ConnectionStrings:AffairesSansRisque_DataConnectionSti
SelectCommand="SELECT [Ann_Id], [Ann_Texte], [Ann_Telephone], [Ann_Email]
UpdateCommand="UPDATE [Annonces] SET [Ann_Texte] = @Ann_Texte, [Ann_Tele
<DeleteParameters>
  <asp:Parameter Name="original_Ann_Id" Type="Int32" />
</DeleteParameters>
<UpdateParameters>
  <asp:Parameter Name="Ann_Texte" Type="String" />
  <asp:Parameter Name="Ann_Telephone" Type="String" />
  <asp:Parameter Name="Ann_Email" Type="String" />
  <asp:Parameter Name="Ann_Prix" Type="String" />
  <asp:Parameter Name="Ann_Ville" Type="String" />
  <asp:Parameter Name="Ann_CodePostal" Type="String" />
  <asp:Parameter Name="Ann_UserEmail" Type="String" />
```

Collection des paramètres de la requête UPDATE.



L'action de glisser déplacer que vous avez faite de la table **Annonces** sur la page a provoqué la génération de l'objet **SqlDataSource** avec l'ensemble des commandes INSERT, UPDATE, DELETE et

SELECT. Ne gardez que la définition de la commande SELECT, puisque nous allons nous contenter pour l'instant d'afficher les données dans la page (nous verrons l'utilisation de la grille en édition dans le § suivant).

```
<asp:SqlDataSource ID="dsAnnonces" runat="server"
    ConnectionString="<%$ ConnectionStrings:csAffairesSansRisque %>"
    ProviderName="<%$ ConnectionStrings:csAffairesSansRisque.ProviderName %>"
    SelectCommand="SELECT [Ann_Id], [Ann_Texte], [Ann_Telephone], [Ann_Email]
    >
</asp:SqlDataSource>
```



De même, le glisser déplacer peut provoquer la création d'une nouvelle chaîne de connexion, par exemple appelée **AffairesSansRisque_DataConnectionString1**. Supprimez-la dans le fichier **Web.config** et renommez les attributs **ConnectionString** et **ProviderName** en utilisant la chaîne **csAffairesSansRisque** que vous aviez déjà définie au § 2.

```
ConnectionString="<%$ ConnectionStrings:csAffairesSansRisque %>"
ProviderName="<%$ ConnectionStrings:csAffairesSansRisque.ProviderName %>"
```

- Sauvegardez la page **ConsulterAnnoncesJournaux.aspx** et exécutez-la dans votre navigateur.

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires Sans Risque AGIL.COM Spécialistes des technos .NET
Gestion de la relation client MSDN Regional Director depuis 10 ans

Accueil + Consulter nos annonces + Consulter les publications

Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Consulter les publications
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

	Email	Ville	Code postal
Sélectionner	gegefernand@hotmail.com	Paris	75000
Sélectionner	valerie.dupont@yahoo.com	Marseille	13000
Sélectionner	dumoulin@anix.fr	Annecy	74000
Sélectionner	fred@hotmail.com	Région Paris	76399
Sélectionner	cdubois@agilcom.fr	Banlieue Paris	77021
Sélectionner	estelle@hotmail.com	Aix-en-Provence	13800
Sélectionner	toto@toto.com	Annecy	74000
Sélectionner	mathilde@bien.com	Cran-Gevrier	74000
Sélectionner	jules@vernes.fr	Marseille	13000
Sélectionner	emmanuelle@bibi.com	non défini	
1 2			

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?

AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

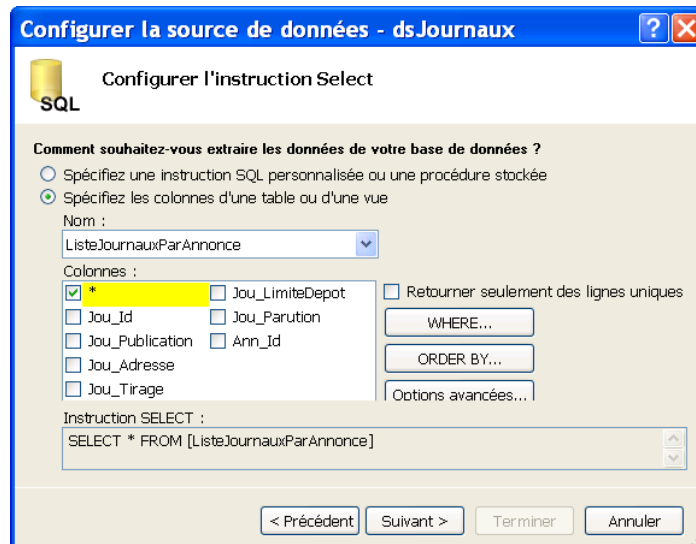


Remarquez que lorsque vous sélectionnez une ligne en cliquant sur **Sélectionner** dans la colonne de gauche, vous provoquez un aller retour sur le serveur (PostBack). Nous allons maintenant répondre à cet appel et brancher l'affichage de tous les journaux dans lesquels cette annonce est publiée.

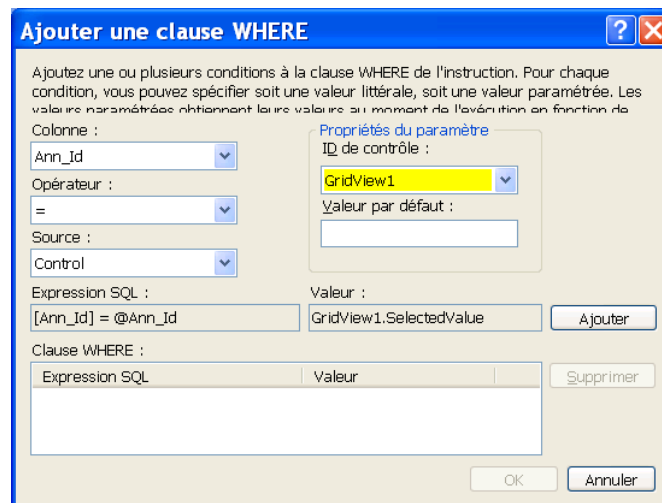
Construction de la liste des publications des annonces

La seconde étape consiste à dessiner une liste des journaux correspondant à l'annonce sélectionnée dans le tableau des annonces.

1. Faites un glisser déplacer de la Boîte à outils > rubrique **Données** > **SqlDataSource** dans **Content1**.
2. Renommez le en : **dsJournaux**.
3. Cliquez sur la balise active > **Configurer la source de données**.
4. Dans la fenêtre **Choisir votre connexion de données**, sélectionnez **csAffairesSansRisque** puis cliquez sur **Suivant**. [éventuellement ConnectionString et vérifier avec "Chaîne de connexion"]
5. Dans la fenêtre **Configurer l'instruction Select**, sélectionnez la vue **ListeJournauxParAnnonce** dans la liste déroulante et cochez tous les champs.



6. Cliquez sur WHERE.
7. Dans la fenêtre **Ajouter une clause WHERE**, sélectionnez le paramètre **An_Id** que vous allez récupérer du contrôle **GridView1**.



8. Cliquez sur **Ajouter**. L'instruction SELECT est maintenant :

```
Instruction SELECT :
SELECT * FROM [ListeJournauxParAnnonce] WHERE ([Ann_Id] = @Ann_Id)
```

9. Terminez l'assistant de configuration de la source de données.
10. Basculez la page en mode **Source**.
Vous obtenez un contrôle **<asp:ControlParameter>** de nom **Ann_Id** dans la collection des paramètres de la requête SELECT, pointant sur la **SelectedValue** de la grille **GridView1**.

```
<asp:SqlDataSource ID="dsJournaux" runat="server"
    ConnectionString="<%= $ConnectionStrings.csAffairesSansRisque %>"
    SelectCommand="SELECT * FROM [ListeJournauxParAnnonce] WHERE ([Ann_Id] = @Ann_Id)">
    <SelectParameters>
        <asp:ControlParameter ControlID="GridView1" Name="Ann_Id"
            PropertyName="SelectedValue"
            Type="Int32" />
    </SelectParameters>
</asp:SqlDataSource>
```



z Mais comment ASP.NET fait-il le lien entre la ligne **SelectedValue** de la grille et précisément la colonne **Ann_Id** qui nous intéresse ?

Rappelez-vous, lors de la définition de la grille, vous aviez constaté que le contrôle **GridView1** possédait entre autres un attribut **DataKeyNames** indiquant qu'elle était la colonne qui faisait office de clé pour un enregistrement, c'est-à-dire **Ann_Id**.

```
<asp:GridView SkinId="tableMaitre" ID="GridView1"
    runat="server" AllowPaging="True"
    AllowSorting="True" AutoGenerateColumns="False"
    DataKeyNames="Ann_Id" DataSourceID="dsAnnonces"
    EmptyDataText="Il n'y a aucun enregistrement de
```

11. Il ne vous reste plus qu'à dessiner une autre grille pour afficher les données de journaux. Faites un glisser déplacer sur la page en design d'un contrôle **GridView**.
12. Paramétrez sa source de données sur **dsJournaux**.

DataSource - dsJournaux							GridView Tâches	
Jou_Id	Jou_Publication	Jou_Adresse	Jou_Tirage	Jou_LimiteDepot	Jou_Parution	Ann_Id	Mise en forme automatique...	
0	abc	abc	abc	abc	abc	0	Choisissez une source de données : dsJournaux	
1	abc	abc	abc	abc	abc	1	Configurer la source de données...	
2	abc	abc	abc	abc	abc	2	Actualiser le schéma	
3	abc	abc	abc	abc	abc	3	Modifier les colonnes...	
4	abc	abc	abc	abc	abc	4	Ajouter une nouvelle colonne...	

13. Actualiser le Schéma puis modifiez les colonnes pour ne garder que **Jou_Publication**, **Jou_Tirage**, **Jou_LimiteDepot** et **Jou_Parution**. Renommez les en-têtes de colonne.
14. Basculez la page en mode **Source** et rajoutez à la définition de la grille l'attribut **SkinId= « tableDetails »** faisant référence à la seconde définition de contrôle `<asp:GridView>` dans votre fichier d'apparence de contrôles.
Vous devez obtenir :

```
<asp:GridView SkinID="tableDetails" ID="GridView2" runat="server" AutoGenerateColumns="False"
DataSourceID="dsJournaux">
  <Columns>
    <asp:BoundField DataField="Jou_Publication" HeaderText="Publication"
SortExpression="Jou_Publication" />
    <asp:BoundField DataField="Jou_Tirage" HeaderText="Tirage" SortExpression="Jou_Tirage" />
    <asp:BoundField DataField="Jou_LimiteDepot" HeaderText="Limite dépôt"
SortExpression="Jou_LimiteDepot" />
    <asp:BoundField DataField="Jou_Parution" HeaderText="Parution"
SortExpression="Jou_Parution" />
  </Columns>
</asp:GridView>
```

15. Sauvegardez puis relancez l'exécution de la page.
16. Sélectionnez une annonce dans le premier tableau pour observer l'affichage contextuel du second :

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires **AGIL.COM** Spécialistes des technos .NET
Gestion de la relation client MSDN Regional Director depuis 10 ans

Accueil + Consulter nos annonces + Consulter les publications

Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Consulter les publications
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

Email	Ville	Code postal
Sélectionner gegefermand@hotmail.com	Paris	75000
Sélectionner valerie.dupont@yahoo.com	Marseille	13000
Sélectionner dumoulin@anix.fr	Anancy	74000
Sélectionner fred@hotmail.com	Région Paris	76399
Sélectionner cdubois@agilcom.fr	Banlieue Paris	77021
Sélectionner estelle@hotmail.com	Aix-en-Provence	13800
Sélectionner toto@toto.com	Anancy	74000
Sélectionner mathilde@bien.com	Cran-Gevrier	74000
Sélectionner jules@vernes.fr	Marseille	13000
Sélectionner emmanuelle@bibi.com	non défini	

1 2

Publication	Tirage	Limite dépôt	Parution
La Tortue déchainée	340000	Mercredi 22h	Vendredi
Toutes les annonces classées	34000	Mardi 15h	Mercredi
Le premier prix	45000	Mercredi 20h	Jeudi
Bon courage	20000	Samedi 15h	Lundi

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?

AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

4 Mettre à jour des données

Dans cet exercice, vous allez apprendre à :

- Afficher des données dans un contrôle DetailsView.
- Mettre à jour des données dans la base de données à partir du contrôle DetailsView.
- Insérer des données à partir d'un contrôle FormView.
- Utiliser une procédure stockée dans un contrôle SqlDataSource.

Contexte fonctionnel

Vous allez construire une page **AdministrerAnnonces.aspx** contenant une liste pour les administrateurs qui souhaitent pouvoir faire des changements sur les données des annonces publiées sur le site. Cette liste présente une colonne **Détails...** pour afficher les détails d'une annonce en lecture et édition.

Adresse <http://localhost:3282/AffairesSansRisque/AdministrerAnnonces.aspx>

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires **AGIL.COM** Smart talk, straight answers
Qui sont les MSDN Regional directors ?

Accueil + Administrer les annonces

Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

Email	Ville	Code postal	Détails
gegefermand@hotmail.com	Paris	75000	Détails...
valerie.dupont@yahoo.com	Marseille	13000	Détails...
dumoulin@anix.fr	Anancy	74000	Détails...
fred@hotmail.com	Région Paris	76399	Détails...
cdubois@agilcom.fr	Banlieue Paris	77021	Détails...
estelle@hotmail.com	Aix-en-Provence	13800	Détails...
toto@toto.com	Anancy	74000	Détails...
mathilde@bien.com	Cran-Gevrier	74000	Détails...
jules@vernes.fr	Marseille	13000	Détails...
emmanuelle@bibi.com	non défini		Détails...

1 2

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?

AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

La colonne **Détails** redirige l'utilisateur sur une autre page contenant les détails de l'annonce choisie.

En cliquant sur détails pour une annonce particulière, vous présentez à l'administrateur une page distincte, nommée **DétailsAnnonce.aspx** contenant tous les champs de l'annonce. Ceux-ci peuvent être consultés en lecture :

Adresse <http://localhost:3282/AffairesSansRisque/DetailsAnnonce.aspx?ID=2>

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires **AGIL.COM** Spécialistes des technos .NET
Gestion de la relation client MSDN Regional Director depuis 10 ans

Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

Id de l'annonce : 2
Texte de l'annonce : Vend coccinelle chouquette
N° de téléphone : 0423456783
Adresse de messagerie : valerie.dupont@yahoo.com
Montant : à débattre
Ville : Marseille
CodePostal : 13000
@ Utilisateur du site :
Région : Midi Pyrenees
Catégorie : Véhicule

Modifier | Supprimer | Retourner à la liste des annonces

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?
AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

Affichage du nom plutôt que la clé étrangère (Idem pour catégorie).

Bouton pour revenir à la liste des annonces.

ou modifiés grâce à un mode d'édition (vous brancherez également la suppression) :

Adresse <http://localhost:3282/AffairesSansRisque/DetailsAnnonce.aspx?ID=2>

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires **AGIL.COM** Spécialistes des technos .NET
Gestion de la relation client MSDN Regional Director depuis 10 ans

Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

Id de l'annonce : 2
Texte de l'annonce : Vend coccinelle chouquette
N° de téléphone : 0423456783
Adresse de messagerie : valerie.dupont@yahoo.com
Montant : à débattre
Ville : Marseille
CodePostal : 13000
@ Utilisateur du site :
Région : Midi Pyrenees
Catégorie : Véhicule

Mettre à jour | Annuler

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?
AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

Affichage d'une liste déroulante permettant de choisir parmi la liste des régions (idem pour Catégorie)

Enfin, vous allez créer une page **DiffuserAnnonce.aspx** pour créer une nouvelle annonce sur le site à partir d'un nouveau contrôle ASP.NET formulaire **FormView** et en utilisant une **procédure stockée** pour la requête d'insertion sur la base de données.

Adresse <http://localhost:3282/AffairesSansRisque/DiffuserAnnonce.aspx>

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires **AGIL.COM** Qui sont les MSDN Regional directors ?
Smart link, straight answers

Accueil + Diffuser une annonce

Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

- Consulter nos annonces
- Diffuser une annonce
- Administrer les annonces
- Qui sommes nous ?

Texte de l'annonce (250 caractères max)

Catégorie : Emploi **Région :** Alsace

Prix (proposé) :
N° de téléphone :
Code Postal :
@ :
Ville :

Valider votre annonce | Annuler

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?
AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

4.1 Administrer les annonces (GridView/DetailsView)

Déroulement de l'exercice :

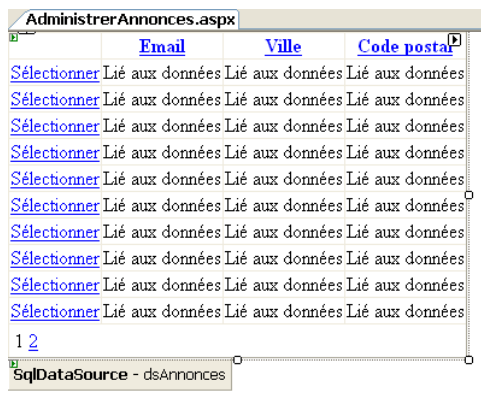
Affichage de la liste des annonces dans une grille

La première étape consiste à reprendre la construction de la liste de toutes les annonces telle que vous l'avez implémentée à l'exercice 3.2. Vous allez ajouter une nouvelle colonne (à la place de la colonne Sélectionner) de type lien hypertexte qui permettra d'appeler une page contenant le détail de l'annonce sélectionnée.



A notez que vous pourriez évidemment faire le choix d'afficher le détails de l'annonce en cours sur la même page. Nous avons choisi cette méthode sur deux pages pour illustrer le type de colonne **HyperLinkField** du contrôle **GridView** qui utilise la liaison de données (databinding) pour passer un paramètre dans l'url du lien.

1. Récupérez le design de la liste des annonces
 - Créer le fichier "**AdministrerAnnonces.aspx**". Sélectionnez le modèle **Web Form** et cochez les deux cases à cocher. Dans la fenêtre **Sélectionner une page maître**, cliquez sur votre page **MasterPage.master** puis sur **OK**
 - Ouvrez la page **ConsulterAnnoncesJournaux.aspx** en mode **Design**.
 - Sélectionnez les deux contrôles **dsAnnonces** et **GridView1** en maintenant la touche **CTRL** enfoncée > clic droit > **Copier**. (Vous pouvez faire du glisser déposer élément par élément)
 -
 - Ouvrez la page **AdministrerAnnonces.aspx** en mode **Design**.
 - Coller les deux contrôles dans **Content1**.



2. Affichez la page dans le navigateur pour valider que la grille fonctionne.

Plan du site | Contactez-nous | Ajoutez aux favoris


Affaires **AGIL.COM** Spécialistes des technos .NET
Gestion de la relation client MSDN Regional Director depuis 10 ans

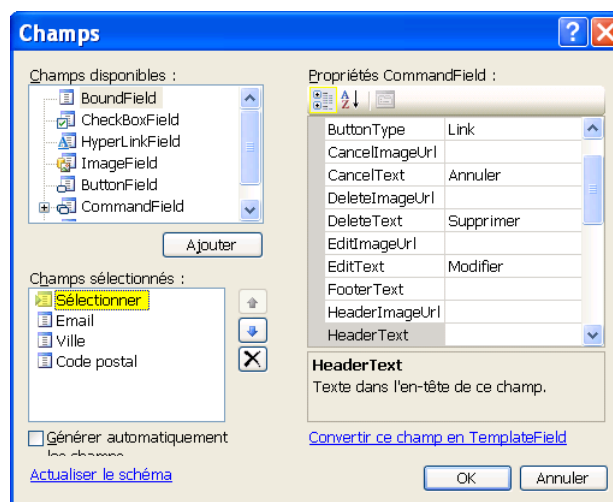
Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

	Email	Ville	CodePostal
Sélectionner	gegefernand@hotmail.com	Paris	75000
Sélectionner	valerie.dupont@yahoo.com	Marseille	13000
Sélectionner	dumoulin@anix.fr	Annecy	74000
Sélectionner	fred@hotmail.com	Région Paris	76399
Sélectionner	cdubois@agilcom.fr	Banlieue Paris	77021
Sélectionner	estelle@hotmail.com	Aix-en-Provence	13800
Sélectionner	toto@toto.com	Annecy	74000
Sélectionner	mathilde@bien.com	Cran-Gevrier	74000
Sélectionner	jules@vernes.fr	Marseille	13000
Sélectionner	emmanuelle@bibi.com	non défini	

AffaireSansRisque | Adresse | Tél : 0 800 888 888 | Fax 0 111 111 999

- Supprimez la colonne **Sélectionner** et créez une nouvelle colonne intitulée **Détails**, pointant sur une page de détails (que vous ferez dans la seconde partie de l'exercice) :
 - Afficher la page **AdministrerAnnonces.aspx** en mode **Design**.
 - Sélectionnez le contrôle **GridView** > sa balise active > **Modifier les colonnes...**
 - Supprimez la colonne **Sélectionner** de type **CommandField** en cliquant .



- Cliquez sur **OK** pour fermer la fenêtre **Champs**. Afficher le résultat.

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires **AGIL.COM** Smart talk, straight answers. Qui sont les MSDN Regional directors ?

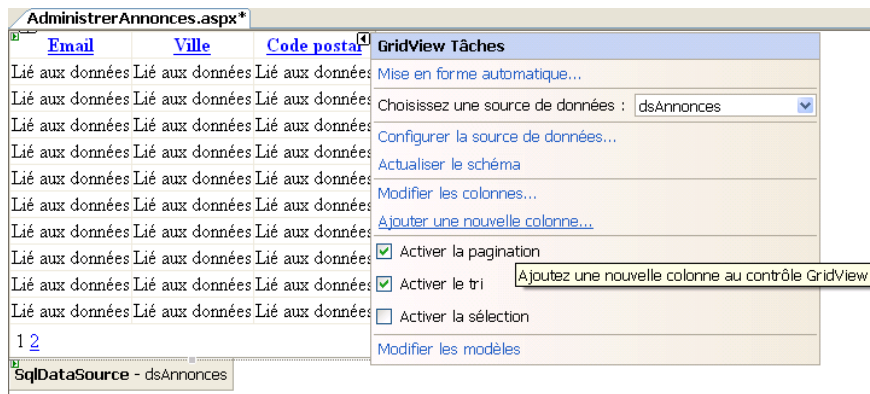
Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

Email	Ville	CodePostal
gegefernand@hotmail.com	Paris	75000
valerie.dupont@yahoo.com	Marseille	13000
dumoulin@anix.fr	Annecy	74000
fred@hotmail.com	Région Paris	76399
cdubois@agilcom.fr	Banlieue Paris	77021
estelle@hotmail.com	Aix-en-Provence	13800
toto@toto.com	Annecy	74000
mathilde@bien.com	Cran-Gevrier	74000
jules@vernes.fr	Marseille	13000
emmanuelle@bibi.com	non défini	

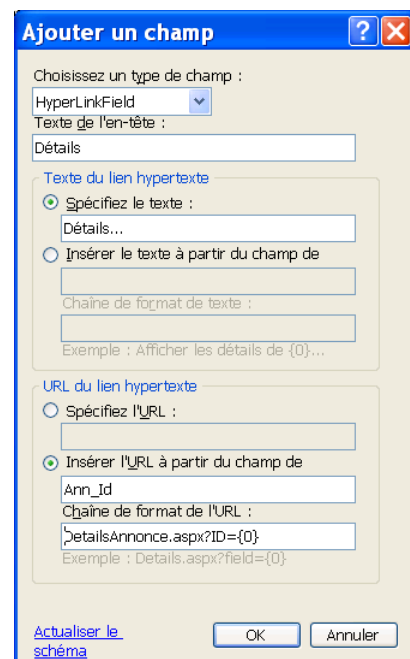
AffaireSansRisque | Adresse | Tél : 0 800 888 888 | Fax 0 111 111 999

- Sélectionnez le contrôle **GridView** > sa balise active > **Ajouter une nouvelle colonne...**



- Dans la fenêtre **Ajouter un champ** :

- Choisissez le type de champ :
HyperLinkField
Ce type de champ créé un lien hypertexte vers une autre url (notre page Détails justement).
- Texte de l'en-tête de la colonne dans la grille :
Détails
- Spécifiez le texte du lien hypertexte :
Détails...
- Spécifiez l'url du lien hypertexte :
DetailsAnnonce.aspx ?ID={0}
où {0} indique un paramètre variable basé sur le champ : **Ann_Id**.



- Basculez la page en mode **Source** pour observer la structure de le nouvelle colonne dans la collection <Columns> du contrôle <asp:GridView> :

```
<Columns>
  <asp:BoundField DataField="Ann_Email" HeaderText="Email"
    SortExpression="Ann_Email" />
  <asp:BoundField DataField="Ann_Ville" HeaderText="Ville"
    SortExpression="Ann_Ville" />
  <asp:BoundField DataField="Ann_CodePostal" HeaderText="Code postal"
    SortExpression="Ann_CodePostal" />
  <asp:HyperLinkField
    DataNavigateUrlFields="Ann_Id"
    DataNavigateUrlFormatString="DetailsAnnonce.aspx?ID={0}"
    HeaderText="Détails" Text="Détails..." />
</Columns>
```

- Sauvegardez la page **AdministrerAnnonces.aspx**. Afficher la page dans votre navigateur.

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires **AGIL.COM** Spécialistes des technos .NET
Gestion de la relation client MSDN Regional Director depuis 10 ans

Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

Email	Ville	CodePostal	Détails
gegefernand@hotmail.com	Paris	75000	Détails ...
valerie.dupont@yahoo.com	Marseille	13000	Détails ...
dumoulin@anix.fr	Annecy	74000	Détails ...
fred@hotmail.com	Région Paris	76399	Détails ...
cdubois@agilcom.fr	Banlieue Paris	77021	Détails ...
estelle@hotmail.com	Aix-en-Provence	13800	Détails ...
toto@toto.com	Annecy	74000	Détails ...
mathilde@bien.com	Cran-Gevrier	74000	Détails ...
jules@vernes.fr	Marseille	13000	Détails ...
emmanuelle@bibi.com	non défini		Détails ...

AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax 0 111 111 999

- Créez une nouvelle page **DetailsAnnonce.aspx**, vierge pour l'instant :
 - Faites un clic droit sur le répertoire du projet dans l'**Explorateur de solutions** > **Ajouter un nouvel élément...**
 - Sélectionnez le modèle **Web Form** et cochez les deux cases à cocher. Nommez la page : **DetailsAnnonce.aspx** puis cliquez sur **Ajouter**.
 - Dans la fenêtre **Sélectionner une page maître**, cliquez sur votre page **MasterPage.master** puis sur **OK**.
- Exécutez la page **AdministrerAnnonces.aspx**.
- Cliquez sur la colonne **Détails** pour un enregistrement en particulier.

Adresse <http://localhost:3282/AffairesSansRisque/DetailsAnnonce.aspx?ID=4>

Plan du site | Contactez-nous | Ajoutez aux favoris

Affaires **AGIL.COM** Smart talk, straight answers
Qui sont les MSDN Regional directors ?

Espace membre

Nos Annonces
Consultez toutes nos annonces et passez vos annonces en ligne

- + Consulter nos annonces
- + Diffuser une annonce
- + Administrer les annonces
- + Qui sommes nous ?

CONSULTER NOS ANNONCES + DIFFUSER UNE ANNONCE ADMINISTRER LES ANNONCES QUI SOMMES NOUS ?

AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999

Regardez la construction de l'url dynamique avec le paramètre contenant l'ID de l'annonce sélectionnée.

Affichage du détail d'une annonce

La seconde étape consiste à construire la page de détails de l'annonce en utilisant le nouveau contrôle de ASP.NET **DetailsView**. Vous allez également activer le mode d'édition de celui-ci de façon à mettre en œuvre la mise à jour des caractéristiques de l'annonce.

- Ouvrez la page **DetailsAnnonce.aspx** en mode Design.
- Configurez une source de données vers la table **Annonces** de la base **AffairesSansRisque** :
 - Faites un glisser déplacer sur le contrôle **Content1** de la **Boîte à outils** > rubrique **Données** > **SqlDataSource**.
 - Faites un clic droit sur le contrôle > **Propriétés** et renommez le contrôle : **dsAnnonce**.
 - Sélectionnez le contrôle pour faire apparaître sa balise active (ou clic droit sur le contrôle > **Afficher la balise active**).

- Cliquez **Configurer la source de données...**
- Dans la fenêtre **Choisir votre connexion de données**, sélectionnez **csAffairesSansRisque** puis cliquez sur **Suivant**.
- Dans la fenêtre **Configurer l'instruction Select**, cliquez **Spécifiez les colonnes d'une table ou d'une vue** et choisissez la table **Annonces** dans la liste déroulante.
- Sélectionnez tous les champs de la table.
- Cliquez sur le bouton **WHERE**.
- Dans la fenêtre Ajouter une clause WHERE, sélectionnez :
 - Colonne : **Ann_Id**,
 - Opérateur : **=**,
 - Source : **QueryString**,
 - Champ QueryString : **ID**.



Vous êtes en train de construire une requête SELECT sur la table **Annonces** pour ne rapatrier que celle dont l'identifiant correspond à la valeur passée en paramètre à l'url de la page appelante. En effet, rappelez-vous l'url que vous avez définie précédemment pour la colonne Détails de la grille dans la page AdministrerAnnonces.aspx :

DetailsAnnonce.aspx ?ID={0} où {0} contient la valeur de Ann_Id.

La récupération d'un paramètre dans une url se fait grâce à la méthode **QueryString** de l'objet **Request**.

DetailsAnnonce.aspx ?ID={0}

où {0} contient la valeur de Ann_Id

- Cliquez sur **Ajouter** pour ajouter le paramètre à la liste des paramètres de la requête.
- Cliquez sur **OK**.
- Vous êtes maintenant de retour dans la fenêtre **Configurer l'instruction Select**. Cliquez sur le bouton **Options avancées...**
- Cochez **Générer des instructions INSERT, UPDATE et DELETE** puis cliquez sur **OK**.



Notez que vous pouvez également choisir le mode de gestion des conflits en cas de mise à jour et destruction sur la base. Si vous choisissez l'accès concurrentiel optimiste, cela génère un attribut `ConflictDetection= « CompareAllValues »` sur le contrôle `SqlDataSource` qui mémorisera chaque valeur initiale du champ pour pouvoir faire une comparaison au retour dans la base.

- Dans la fenêtre **Configurer l'instruction Select**, cliquez sur **Suivant**.
- Dans la fenêtre **Tester la requête**, cliquez sur **Terminer**.
- Basculez la page en mode **Source** pour voir le flux généré :

```
<asp:SqlDataSource ID="dsAnnonce" runat="server"
    ConnectionString="<%"$ ConnectionStrings:csAffairesSansRisque %">"
    SelectCommand="SELECT * FROM [Annonces] WHERE ([Ann_Id] = @Ann_Id)"
    DeleteCommand="DELETE FROM [Annonces] WHERE [Ann_Id] = @original_Ann_Id"
    InsertCommand="INSERT INTO [Annonces] ([Ann_Texte], [Ann_Telephone], [Ann_Email],
    [Ann_Prix], [Ann_Ville], [Ann_CodePostal], [Ann_UserEmail], [Reg_Id], [Cat_Id]) VALUES
    (@Ann_Texte, @Ann_Telephone, @Ann_Email, @Ann_Prix, @Ann_Ville,
    @Ann_CodePostal, @Ann_UserEmail, @Reg_Id, @Cat_Id)"
    UpdateCommand="UPDATE [Annonces] SET [Ann_Texte] = @Ann_Texte, [Ann_Telephone]
    = @Ann_Telephone, [Ann_Email] = @Ann_Email, [Ann_Prix] = @Ann_Prix, [Ann_Ville] =
    @Ann_Ville, [Ann_CodePostal] = @Ann_CodePostal, [Ann_UserEmail] = @Ann_UserEmail,
    [Reg_Id] = @Reg_Id, [Cat_Id] = @Cat_Id WHERE [Ann_Id] = @original_Ann_Id">
    <SelectParameters>
        <asp:QueryStringParameter Name="Ann_Id" QueryStringField="ID" Type="Int32" />
    </SelectParameters>
    <DeleteParameters>
        <asp:Parameter Name="original_Ann_Id" Type="Int32" />
    </DeleteParameters>
    <UpdateParameters>
        <asp:Parameter Name="Ann_Texte" Type="String" />
        <asp:Parameter Name="Ann_Telephone" Type="String" />
        <asp:Parameter Name="Ann_Email" Type="String" />
        <asp:Parameter Name="Ann_Prix" Type="String" />
        <asp:Parameter Name="Ann_Ville" Type="String" />
        <asp:Parameter Name="Ann_CodePostal" Type="String" />
        <asp:Parameter Name="Ann_UserEmail" Type="String" />
        <asp:Parameter Name="Reg_Id" Type="Int32" />
        <asp:Parameter Name="Cat_Id" Type="Int32" />
        <asp:Parameter Name="original_Ann_Id" Type="Int32" />
    </UpdateParameters>
    <InsertParameters>
        <asp:Parameter Name="Ann_Texte" Type="String" />
        <asp:Parameter Name="Ann_Telephone" Type="String" />
        <asp:Parameter Name="Ann_Email" Type="String" />
        <asp:Parameter Name="Ann_Prix" Type="String" />
        <asp:Parameter Name="Ann_Ville" Type="String" />
        <asp:Parameter Name="Ann_CodePostal" Type="String" />
        <asp:Parameter Name="Ann_UserEmail" Type="String" />
        <asp:Parameter Name="Reg_Id" Type="Int32" />
        <asp:Parameter Name="Cat_Id" Type="Int32" />
    </InsertParameters>
</asp:SqlDataSource>
```



Supprimez l'attribut **InsertCommand** et l'élément **<InsertParameters>** puisque l'insertion d'une nouvelle annonce sera implémentée dans le prochain exercice dans une page séparée (`DiffuserAnnonce.aspx`).



Remarquez le paramètre de la requête SELECT dans la collection **<SelectParameters>** qui est représenté par un élément **<asp:QueryStringParameter>** : l'attribut **QueryStringField** indique le nom du champ dans la requête à récupérer par ASP.NET. **Name** indique le nom du paramètre dans la requête SELECT.

```
<asp:SqlDataSource ID="dsAnnonce" runat="server"
    ConnectionString="<%= $%$ ConnectionStrings:csAffairesSansRisque %>"
    SelectCommand="SELECT * FROM [Annonces] WHERE ([Ann_Id] = @Ann_Id)">
    <SelectParameters>
        <asp:QueryStringParameter Name="Ann_Id"
                                QueryStringField="ID"
                                Type="Int32" />
    </SelectParameters>
</asp:SqlDataSource>
```



Remarquez également les collections **<UpdateParameters>** et **<DeleteParameters>** qui contiennent la définition des paramètres des requêtes UPDATE et DELETE.

- Dessinez maintenant le contrôle graphique DetailsView pour afficher les champs d'une annonce :
 - Basculez la page **DetailsAnnonce.aspx** en mode **Design**.
 - Faites un glisser déplacer sur le contrôle **Content1** de la **Boîte à outils** > rubrique **Données** > **DetailsView**.
 - Sélectionnez le contrôle et cliquez sur sa balise active > **Choisir une source de données** : **dsAnnonce**.
 - Cochez **Activer la modification** et **Activer la suppression**.

Affichage par défaut des champs et du label.

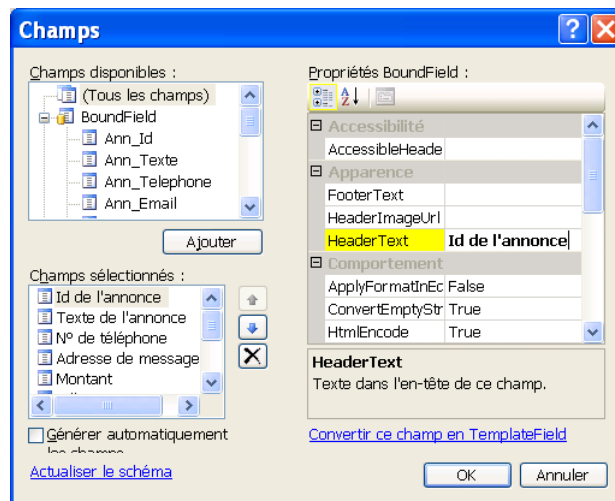


Vous constatez que le contrôle **DetailsView** est intelligent et interprète la source de données dont il connaît le schéma. Il en extrait le nom des champs à afficher pour chaque colonne de la table définie dans la requête SELECT.

Du coup, il génère un modèle simple d'affichage des données avec les en-têtes correspondantes aux noms des colonnes de la table.

Comme le contrôle **dsAnnonce** contient des instructions de commande UPDATE et DELETE, il a également ajouté deux boutons **Modifier** et **Supprimer** dans le bas de page.

- Dans la balise active du contrôle, cliquez sur **Modifier les champs** pour remplacer les libellés des champs par des noms plus conviviaux.
 - Changez la propriété **HeaderText** des champs sélectionnés.



Par exemple, vous pouvez saisir :

Nom du champ	HeaderText
Ann_Id	Id de l'annonce :
Ann_Texte	Texte de l'annonce :
Ann_Telephone	N° de téléphone :
Ann_Email	Adresse de messagerie :
Ann_Ville	Ville :
Ann_CodePostal	Code Postal :
Ann_Prix	Montant :
Ann_UserEmail	@ Utilisateur du site :
Reg_Id	Région :
Cat_Id	Catégorie :

5. Sauvegardez la page **DetailsAnnonce.aspx**.
6. Exécutez **AdministrerAnnonces.aspx** dans votre navigateur et sélectionnez une ligne puis **Détails...** pour exécuter la page **DetailsAnnonce.aspx**.

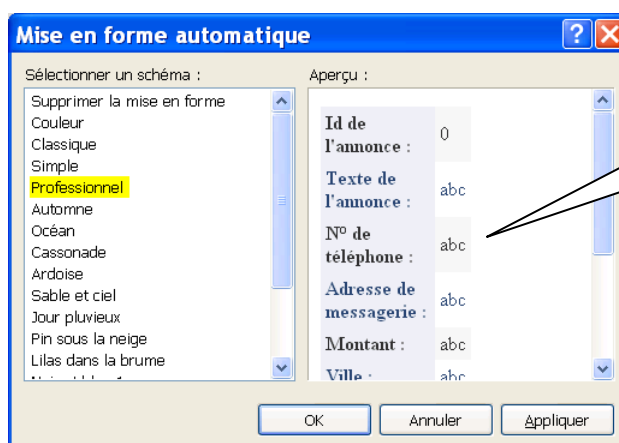


Vous constatez que la page est bien formatée. Le contrôle **DetailsView** est en effet défini dans le fichier d'apparence avec la représentation suivante :

Dans le fichier Default.skin :	Dans la feuille de style Default.css :
<pre> <asp:DetailsView runat="server" BorderStyle="None" BorderWidth="0" CellPadding="5" width="450px"> <FieldHeaderStyle CssClass="headercell" > </FieldHeaderStyle> </asp:DetailsView> </pre>	<pre> .headercell { color: #003063; font-weight:bold; vertical-align:text-top ; text-align:right; width:150px; } </pre>



Un contrôle **DetailsView** contient de multiples propriétés d'affichage tout comme le GridView (HeaderStyle, FooterStyle, RowStyle, EditRowStyle etc...). Vous pouvez également utiliser un format automatique à partir de la balise active du contrôle > **mise en forme automatique ...**



Configurez **AlternatingRowStyle** pour alterner les styles un champ sur deux.



Vous pourriez activer le mode d'édition par défaut sur le contrôle DetailsView (pour éviter à l'administrateur de cliquer sur le bouton **Modifier**) en paramétrant sa propriété **DefaultMode** à **Edit**.

- Cliquez sur **Modifier** pour passer en mode édition.
- Faites une modification quelconque > **Mettre à jour** pour valider les changements.
- Testez également la suppression de l'enregistrement.

7. Ouvrez la page **DetailsAnnonce.aspx** en mode **Source** pour voir la représentation du contrôle dans le flux de la page :

```

<asp:DetailsView ID="DetailsView1" runat="server" AutoGenerateRows="False" DataKeyNames="Ann_Id"
    DataSourceID="dsAnnonce" Height="50px" Width="125px">
    <Fields>
        <asp:BoundField DataField="Ann_Id" HeaderText="Id de l'annonce :" InsertVisible="False"
            ReadOnly="True" SortExpression="Ann_Id" />
        <asp:BoundField DataField="Ann_Texte" HeaderText="Texte de l'annonce :"
            SortExpression="Ann_Texte" />
        <asp:BoundField DataField="Ann_Telephone" HeaderText="N° de téléphone :"
            SortExpression="Ann_Telephone" />
        <asp:BoundField DataField="Ann_Email" HeaderText="Adresse de messagerie :"
            SortExpression="Ann_Email" />
        <asp:BoundField DataField="Ann_Prix" HeaderText="Montant :" SortExpression="Ann_Prix" />
        <asp:BoundField DataField="Ann_Ville" HeaderText="Ville :" SortExpression="Ann_Ville" />
        <asp:BoundField DataField="Ann_CodePostal" HeaderText="CodePostal :"

```



```

SortExpression="Ann_CodePostal" />
<asp:BoundField DataField="Ann_UserEmail" HeaderText="@ Utilisateur du site :."
SortExpression="Ann_UserEmail" />
<asp:BoundField DataField="Reg_Id" HeaderText="Région :." SortExpression="Reg_Id"/>
<asp:BoundField DataField="Cat_Id" HeaderText="Catégorie :." SortExpression="Cat_Id" />
<asp:CommandField ShowDeleteButton="True" ShowEditButton="True" />
</Fields>
</asp:DetailsView>

```

Améliorer la page de détail d'une annonce

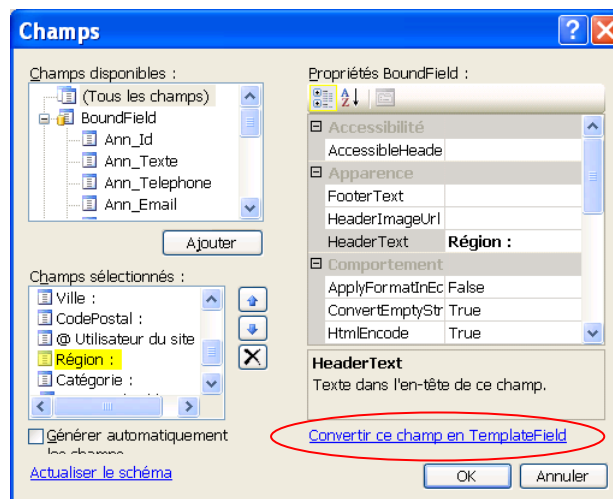
Il vous reste quelques petites manipulations pour améliorer le fonctionnement de cette page de détails.

La première consiste à faire travailler l'administrateur avec les **noms** des régions et des catégories plutôt que leurs ID.

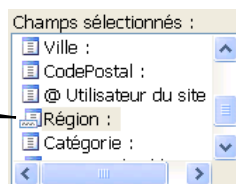
La seconde consiste à ajouter un bouton au formulaire qui permettrait à l'administrateur de revenir à la liste de toutes les annonces sur la page **AdministrerAnnonces.aspx**.

Pour cela, vous allez convertir les colonnes correspondantes en modèle de façon à pouvoir les dessiner à votre manière.

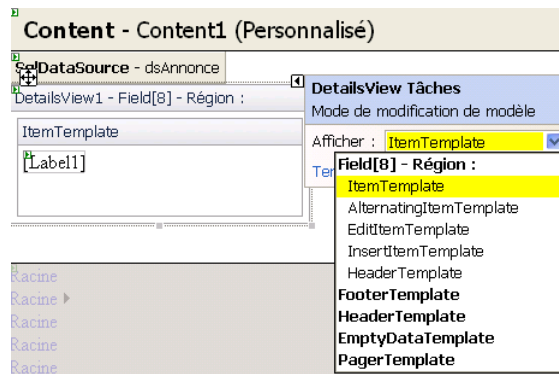
- Changez l'affichage de la zone représentant la région en suivant la procédure suivante :
 - Affichez **DetailsAnnonce.aspx** en mode **Design**.
 - Sélectionnez le contrôle **DetailsView1** > sa balise active > **Modifier les champs...**
 - Dans la fenêtre Champs, sélectionnez le champ **Région** et cliquez **Convertir ce champ en TemplateField** pour le rendre disponible à la personnalisation.



Notez le changement d'icône sur le champ.



- Cliquez sur **OK**.
- Toujours dans la balise active de **DetailsView1** > **Modifier les modèles...**
- Cliquez sur la liste déroulante **Afficher** de la balise active du modèle pour voir les différents modèles proposés :



- Basculez en mode **Source**. La conversion de la colonne en modèle a généré un contrôle **<asp:TemplateField>** à la place du **<asp:BoundField>** initial associé à Reg_Id, contenant les modèles **<ItemTemplate>** (lecture seule), **<EditItemTemplate>** (Edition) :

```
<asp:TemplateField HeaderText="Région :" SortExpression="Reg_Id">
  <ItemTemplate>
    <asp:Label ID="Label1" runat="server" Text='<%=# Bind("Reg_Id") %>'></asp:Label>
  </ItemTemplate>
  <EditItemTemplate>
    <asp:TextBox ID="TextBox1" runat="server" Text='<%=# Bind("Reg_Id") %>'></asp:TextBox>
  </EditItemTemplate>
  <InsertItemTemplate>
    <asp:TextBox ID="TextBox1" runat="server" Text='<%=# Bind("Reg_Id") %>'></asp:TextBox>
  </InsertItemTemplate>
</asp:TemplateField>
```

Supprimez
le modèle
d'insertion
(inutile ici)



Vous constatez que la liaison de données, implicite dans un contrôle **<asp:BoundField>**, est par contre devenue explicite dans le modèle avec l'apparition de la fonction **Bind()** directement sur le contrôle concerné (Label en lecture seule ou TextBox en édition).

Pour rappel, la fonction **Bind()** est bidirectionnelle. Elle assure la synchronisation du contrôle d'affichage et de la source de données dès qu'un changement se produit sur la donnée, que ce soit à partir du contrôle d'affichage ou de la source de données elle-même.

- Modifiez tout d'abord le modèle en lecture seule : **<ItemTemplate>**. Il faudrait que la donnée liée soit Reg_nom plutôt que Reg_Id. Malheureusement l'information sur le nom de la région n'est pas renvoyée dans la requête SELECT que vous avez défini dans le contrôle de source de données dsAnnonce.
Il faudrait donc rajouter à la requête SELECT une jointure avec les tables Regions et Categories pour rapatrier également les noms de région et catégorie.
Modifiez l'attribut **SelectCommand** du contrôle **<asp:SqlDataSource>** en ce sens comme suit :

```
<asp:SqlDataSource ID="dsAnnonce" runat="server"
  ConnectionString="<%= $ ConnectionStrings:csAffairesSansRisque %>"

  SelectCommand="SELECT Annonces.Ann_Id, Annonces.Ann_Texte, Annonces.Ann_Telephone,
  Annonces.Ann_Email, Annonces.Ann_Prix, Annonces.Ann_Ville, Annonces.Ann_CodePostal,
  Ann_UserEmail, Annonces.Reg_Id, Annonces.Cat_Id, Regions.Reg_Nom, Categories.Cat_Nom FROM
  Annonces
INNER JOIN Categories ON Annonces.Cat_Id = Categories.Cat_Id
INNER JOIN Regions ON Annonces.Reg_Id = Regions.Reg_Id
  WHERE (Annonces.Ann_Id = @Ann_Id)"

  ...
```

```
</asp:SqlDataSource>
```

- Vous pouvez maintenant changer la liaison dans le contrôle **<asp:Label>** du modèle **<ItemTemplate>** pour qu'elle lie la propriété **Text** du contrôle à la colonne **Reg_Nom**.

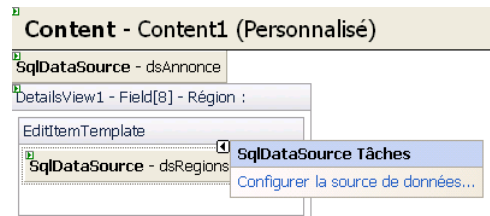
```
<ItemTemplate>
  <asp:Label ID="Label1" runat="server" Text='<%= Bind("Reg_Nom") %>'></asp:Label>
</ItemTemplate>
```

- Modifiez maintenant le modèle en édition : **<EditItemTemplate>**. L'idéal serait de proposer à l'administrateur une liste déroulante avec tous les noms des régions existantes... Pour réaliser cela, il va vous falloir déclarer une autre source de données avec une requête **SELECT** sur la table **Regions** de façon à rapatrier toutes les régions et dessiner une liste déroulante liée à ce contrôle de données.
- Basculez la page **DetailsAnnonce.aspx** en mode **Design**.
- Affichez la balise active du contrôle **DetailsView1** > **Modifier les modèles...**
- Dans la balise active, sélectionnez Afficher : **EditItemTemplate**.

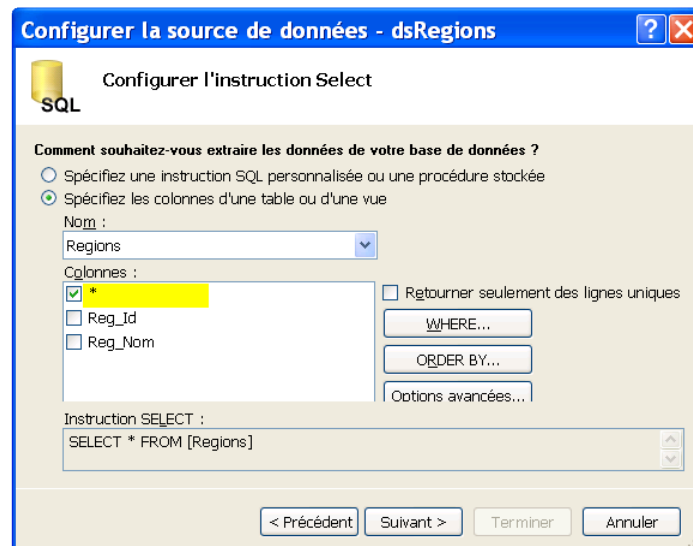
Le modèle contient une Textbox par défaut.



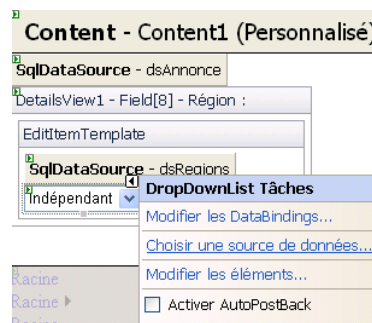
- Supprimez le contrôle **TextBox**.
- Faites un glisser-déplacer de la **Boîte à outils** > rubrique **Données** > **SqlDataSource** sur le modèle.
- Renommez le contrôle **dsRegions**.
- Dans la balise active du contrôle **dsRegions**, sélectionnez **Configurer la source de données...**



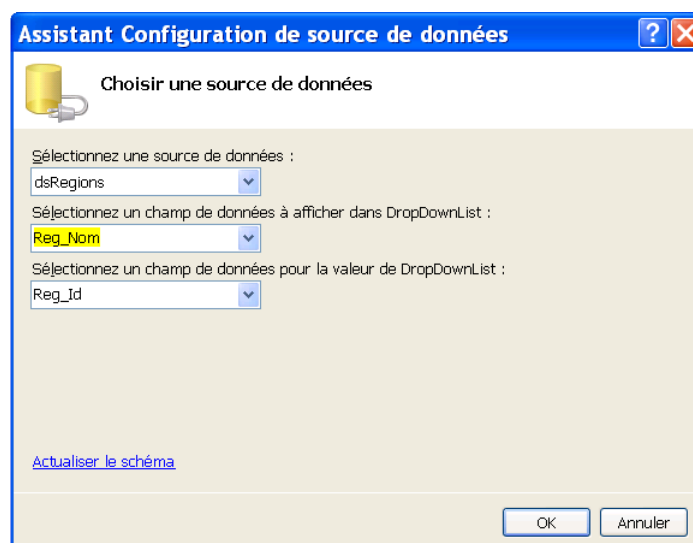
- Sélectionnez ensuite la connexion **csAffairesSansRisque** et cliquez sur **Suivant**.
- Dans la fenêtre **Configurer l'instruction Select**, sélectionnez la table **Regions** et toutes les colonnes *.



- Cliquez sur **Suivant** puis **Terminer**.
- Faites un glisser déplacer d'un contrôle **DropDownList** sur le modèle.
- Dans sa balise active, cliquez sur **Choisir une source de données...** pour le lier à la source **dsRegions** créée précédemment.

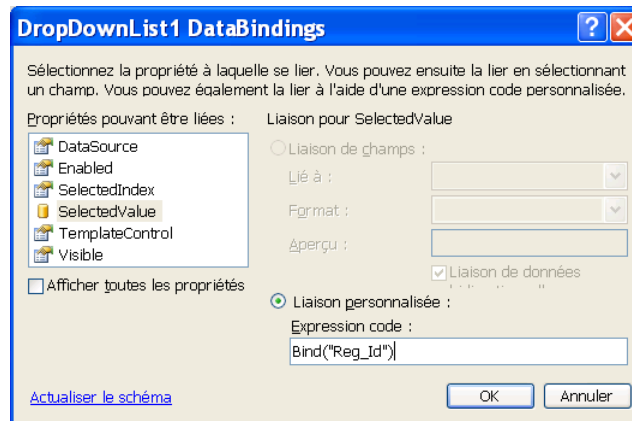


- Dans la fenêtre **Choisir une source de données**, sélectionnez **dsRegions** et paramétrer la valeur affichée dans la liste sur **Reg_Nom** et la valeur à considérer dans la source de données **Reg_Id**.



- Cliquez sur **OK**.

- Dans la balise active de la DropDownList, cliquez sur **Modifier les Databindings...** et ajoutez **Bind(« Reg_Id »)** pour la propriété **SelectedValue** :



Ici, vous indiquez à ASP.NET que la liaison devra se faire par rapport à la valeur sélectionnée dans la liste déroulante (**SelectedValue** c'est-à-dire **Reg_Id** qui est le **DataValueField** dans la DropDownList) et que le champ à mettre à jour dans la base de données globale (**dsAnnonce**) sera **Reg_Id** (fonction **Bind()**).

- Cliquez sur **OK**.
- Cliquez sur la balise active > **Terminer la modification du modèle**.
- Basculez en mode **Source** pour voir le modèle **<EditItemTemplate>** généré :

```
<EditItemTemplate>

<asp:SqlDataSource ID="dsRegions" runat="server"
    ConnectionString="<%= $ ConnectionStrings:csAffairesSansRisque %>"
    SelectCommand="SELECT * FROM [Regions]">
</asp:SqlDataSource>

<asp:DropDownList ID="DropDownList1" runat="server"
    DataSourceID="dsRegions" DataTextField="Reg_Nom"
    DataValueField="Reg_Id" SelectedValue="<%= # Bind("Reg_Id") %>">
</asp:DropDownList>

</EditItemTemplate>
```

- Sauvegardez la page **DetailsAnnonce.aspx**.
- Exécutez **AdministrerAnnonces.aspx** dans VOTRE NAVIGATEUR et sélectionnez une ligne puis **Détails...** pour exécuter la page **DetailsAnnonce.aspx** et valider le fonctionnement des modèles

En mode lecture seule :

Id de l'annonce : 4
 Texte de l'annonce : Vend chaton magnifique
 N° de téléphone : 0129384779
 Adresse de messagerie : fred@hotmail.com
 Montant : 50€
 Ville : Région Paris
 CodePostal : 76399
 @ Utilisateur du site :
 Région : Alsace
 Catégorie : 7
 Modifier Supprimer

C'est bien le champ Reg_Nom qui apparaît.

En mode édition :

Id de l'annonce :	Bourgogne	
Texte de l'annonce :	Bretagne	
N° de téléphone :	Centre	
Adresse de messagerie :	Champagne	
Montant :	Franche Comte	
Ville :	Languedoc Roussillon	
CodePostal :	Limousin	
@ Utilisateur du site :	Lorraine	
Région :	Midi Pyrenees	
Catégorie :	Nord	
	Pays de Loire	
	Picardie	
	Poitou Charente	
	Provence Cote d'Azur	
	Ile de France	
	Rhone Alpes	
	Haute Normandie	
	Alsace	

Mettre à jour Annuler

Vous obtenez une liste déroulante à la place d'une zone de texte.

- Changez de région et cliquez sur **Mettre à jour** pour valider que la mise à jour fonctionne via le champ **Reg_Id** sous-jacent. La nouvelle région doit apparaître au retour en mode lecture de l'annonce.

2. Maintenant re-déroulez toute la procédure du point 1 pour la Catégorie.



Pour les plus flemmards d'entre vous ☺...un simple copier coller des modèles dans le source devrait faire l'affaire avec les deux ou trois retouches qui s'imposent pour changer les noms (attention aux contrôles qui doivent avoir un Id unique notamment)

```
<asp:TemplateField HeaderText="Catégorie :" SortExpression="Cat_Id">
  <ItemTemplate>
    <asp:Label ID="Label2" runat="server" Text='<%= Bind("Cat_Nom") %>'></asp:Label>
  </ItemTemplate>
  <EditItemTemplate>
    <asp:SqlDataSource ID="dsCategories" runat="server"
      ConnectionString="<%= $ ConnectionStrings:csAffairesSansRisque %>"
      SelectCommand="SELECT * FROM [Categories]"></asp:SqlDataSource>
    <asp:DropDownList ID="DropDownList2" runat="server"
      DataSourceID="dsCategories" DataTextField="Cat_Nom"
      DataValueField="Cat_Id" SelectedValue='<%= Bind("Cat_Id") %>'>
    </asp:DropDownList>
  </EditItemTemplate>
</asp:TemplateField>
```

- Testez en exécution le bon fonctionnement de la catégorie.

3. Pour terminer, nous allons changer également le bas de page du contrôle **DetailsView** pour ajouter un bouton (de type lien hypertexte) de retour vers la page **AdministrerAnnonces.aspx**.



Evidemment, vous pourriez ajouter ce bouton n'importe où sur la page **DetailsAnnonce.aspx**. Mais histoire de se compliquer un peu la vie ☺, nous allons ici le positionner juste à côté des boutons de commande à l'intérieur du contrôle **DetailsView**...

- Reprenez les étapes précédentes de manière similaire. Commencez par convertir en modèle le champ **CommandField** qui contient les boutons de commande.





AGIL.COM
Gestion de la relation client

Plan du site | Contactez-nous | Ajoutez aux favoris
 Spécialistes des technos .NET
 MSDN Regional Director depuis 10 ans



Espace membre

Nos Annonces

Consultez toutes nos annonces et passez vos annonces en ligne

- [+ Consulter nos annonces](#)
- [• Diffuser une annonce](#)
- [• Administrer les annonces](#)
- [• Qui sommes nous ?](#)

Id de l'annonce : 6

Texte de l'annonce : Recherche à apprendre ASP.NET

N° de téléphone : 0448938991

Adresse de messagerie : estelle@hotmail.com

Montant : pour rien

Ville : Aix-en-Provence

CodePostal : 13800

@ Utilisateur du site :

Région : Midi Pyrenees

Catégorie : Informatique

[Modifier](#) | [Supprimer](#) | [Retourner à la liste des annonces](#)

[CONSULTER NOS ANNONCES +](#)
[DIFFUSER UNE ANNONCE](#)
[ADMINISTRER LES ANNONCES](#)
[QUI SOMMES NOUS ?](#)







AffairesSansRisque | Adresse | Tél : 0 800 888 888 | Fax : 0 111 111 999



HyperLink1 DataBindings

Sélectionnez la propriété à laquelle se lier. Vous pouvez ensuite la lier en sélectionnant un champ. Vous pouvez également la lier à l'aide d'une expression code personnalisée.

Propriétés pouvant être liées : Liaison pour NavigateurUrl

 Enabled
 ImageUrl
 **NavigateUrl**
 TemplateUrl
 Text
 Visible

☐ Afficher toutes les propriétés

☐ Liaison de champs :
Lié à :
Format :
Aperçu :
☐ Liaison de données

☒ Liaison personnalisée :
Expression code :

[Actualiser le schéma](#)



L'expression de liaison est plus complexe : vous liez la valeur de la propriété **NavigateUrl** du contrôle à la colonne **Reg_Id**. Cependant ce **Reg_Id** n'est pas évalué tel quel ; il n'est en fait qu'un paramètre (représenté par {0}) de la chaîne de caractères globale représentant l'url. La propriété **NavigateUrl** vaut « AdministrerAnnonces.aspx ?Reg_Id=<valeur du reg_id de l'annonce en cours dans la source de données définie par le SqlDataSource > »

- Resterait enfin à interpréter le paramètre **Reg_Id** dans le chargement de la page AdministrerAnnonces.aspx pour positionner la liste déroulante sur la région voulue.

4.2 Diffuser une annonce (FormView + utilisation d'une procédure stockée)

Déroulement de l'exercice :

1. Affichez la page **DiffuserAnnonce.aspx**.
2. Ajoutez un contrôle Source de données pointant sur la table **Annonces** et nommez le **dsAnnonces**.
3. Configurez la source de données de façon à récupérer la procédure stockée **DiffuserAnnonce** pour la requête INSERT dans la table Annonces.



Attention, le bouton **Suivant** reste griser si vous ne définissez pas une requête SELECT dans l'onglet SELECT (pourtant inutile ici puisque nous n'allons utiliser que la requête d'insertion sur la base). Il faut savoir également que sans requête SELECT, lorsque vous dessinerez le contrôle d'affichage FormView sur la page et que vous le lierez à votre source de données, la génération automatique des colonnes du formulaire n'aura pas lu en mode Design (pour la bonne raison que le Designer se repose sur les champs retournés par la requête SELECT pour en déduire les champs à afficher)...

- Basculez la page en mode Source pour voir le contrôle **SqlDataSource** généré :

La commande INSERT est de type **StoredProcedure** et celle-ci s'appelle **DiffuserAnnonce**.

```
<asp:SqlDataSource ID="dsAnnonce" runat="server"
    ConnectionString="<%%$ ConnectionStrings:csAffairesSansRisque %>"
    InsertCommand="DiffuserAnnonce"
    InsertCommandType="StoredProcedure"
    SelectCommand="Select * from [Annonces]">
    <InsertParameters>
        <asp:Parameter Name="Ann_Texte" Type="String" />
        <asp:Parameter Name="Ann_Telephone" Type="String" />
        <asp:Parameter Name="Ann_Email" Type="String" />
        <asp:Parameter Name="Ann_Prix" Type="String" />
        <asp:Parameter Name="Ann_Ville" Type="String" />
        <asp:Parameter Name="Ann_CodePostal" Type="String" />
        <asp:Parameter Name="Ann_UserEmail" Type="String" />
        <asp:Parameter Name="Reg_Id" Type="Int32" />
        <asp:Parameter Name="Cat_Id" Type="Int32" />
    </InsertParameters>
</asp:SqlDataSource>
```

Collection des paramètres attendus par la procédure stockée.

4. Dessinez un contrôle FormView sur la page :

- Basculez la page en mode Design et faites un glisser déplacer de la **Boîte à outils** > rubrique **Données** > contrôle **FormView** sur **Content1**.
- Configurez la source de données du contrôle sur **dsAnnonce** créé précédemment.

Pour mémo : la génération des éléments n'a pas lieu ici sans la requête SELECT dans dsAnnonce.

- Paramétrez la propriété **DefaultMode** du contrôle **FormView** sur **Insert** pour charger le formulaire en mode d'insertion au lancement de la page.

Le Designer vous montre maintenant le formulaire en mode d'insertion.

Content - Content1 (Personnalisé)

SqlDataSource - dsAnnonce

Ann_Texte: Lié aux données

Ann_Telephone: Lié aux données

Ann_Email: Lié aux données

Ann_Prix: Lié aux données

Ann_Ville: Lié aux données

Ann_CodePostal: Lié aux données

Ann_UserEmail: Lié aux données

Reg_Id: Lié aux données

Cat_Id: Lié aux données

[Insert](#) [Cancel](#)

- Basculez la page en mode **Source** pour observer le flux du contrôle **FormView** (ne gardez que le modèle **<InsertTemplate>** puisqu'on n'utilise que le mode insertion) :

```
<asp:FormView ID="FormView1" runat="server" DataKeyNames="Ann_Id" DataSourceID="dsAnnonce"
DefaultMode="Insert">
  <InsertItemTemplate>
    Ann_Texte:
    <asp:TextBox ID="Ann_TexteTextBox" runat="server" Text='<%= Bind("Ann_Texte") %>'>
    </asp:TextBox><br />
    Ann_Telephone:
    <asp:TextBox ID="Ann_TelephoneTextBox" runat="server"
      Text='<%= Bind("Ann_Telephone") %>'>
    </asp:TextBox><br />
    Ann_Email:
    <asp:TextBox ID="Ann_EmailTextBox" runat="server" Text='<%= Bind("Ann_Email") %>'>
    </asp:TextBox><br />
    Ann_Prix:
    <asp:TextBox ID="Ann_PrixTextBox" runat="server" Text='<%= Bind("Ann_Prix") %>'>
    </asp:TextBox><br />
    Ann_Ville:
    <asp:TextBox ID="Ann_VilleTextBox" runat="server" Text='<%= Bind("Ann_Ville") %>'>
    </asp:TextBox><br />
    Ann_CodePostal:
    <asp:TextBox ID="Ann_CodePostalTextBox" runat="server"
      Text='<%= Bind("Ann_CodePostal") %>'>
    </asp:TextBox><br />
    Ann_UserEmail:
    <asp:TextBox ID="Ann_UserEmailTextBox" runat="server"
      Text='<%= Bind("Ann_UserEmail") %>'>
    </asp:TextBox><br />
    Reg_Id:
    <asp:TextBox ID="Reg_IdTextBox" runat="server" Text='<%= Bind("Reg_Id") %>'>
    </asp:TextBox><br />
    Cat_Id:
    <asp:TextBox ID="Cat_IdTextBox" runat="server" Text='<%= Bind("Cat_Id") %>'>
    </asp:TextBox><br />
    <asp:LinkButton ID="InsertButton" runat="server" CausesValidation="True"
      CommandName="Insert" Text="Insert">
    </asp:LinkButton>
    <asp:LinkButton ID="InsertCancelButton" runat="server"
      CausesValidation="False" CommandName="Cancel" Text="Cancel">
    </asp:LinkButton>
  </InsertItemTemplate>
</asp:FormView>
```

5. Sauvegardez la page **DiffuserAnnonce.aspx**.
6. Exécutez la page **DiffuserAnnonce.aspx** pour tester le formulaire.



Evidemment le rendu n'est pas terrible. Justement l'intérêt du contrôle FormView réside dans le fait que vous êtes totalement libre de personnaliser le formulaire comme vous le souhaitez à l'intérieur de ces modèles.

Voici un exemple avec une structure en tableau <table> HTML (...oui, c'est vrai c'est terrifiant !! Vous comprenez pourquoi il vaut beaucoup mieux travailler avec les balises <div>)

```
<asp:FormView ID="FormView1" runat="server" DataKeyNames="Ann_Id" DataSourceID="dsAnnonce"
DefaultMode="Insert" >
  <InsertItemTemplate>
    <table border="0" cellpadding="0" cellspacing="5" >
      <tr>
        <td style="height: 78px" valign="top" width="150">
          <h3>Texte de l'annonce (250 caractères max)</h3>
        </td>
        <td colspan="3" style="height: 78px" valign="top" >
          <asp:TextBox ID="TextBox1" runat="server" Text='<%= Bind("Ann_Texte") %>'
            TextMode="MultiLine" MaxLength="250" Height="132px" Width="350"
            EnableTheming="False"></asp:TextBox></td>
      </tr>
      <tr>
        <td valign="top" width="150">
          <h3>Catégorie </h3>
        </td>
        <td valign="top" width="200">
          <asp:DropDownList ID="DropDownList1" runat="server" DataSourceID="dsCategories"
            DataTextField="Cat_Nom" DataValueField="Cat_Id"
            SelectedValue='<%= Bind("Cat_Id") %>'>
          </asp:DropDownList></td>
        <td valign="top" width="60">
          <h3>Région</h3>
        </td>
        <td valign="top" width="150">
          <asp:DropDownList ID="DropDownList3" runat="server" DataSourceID="dsRegions"
            DataTextField="Reg_Nom" DataValueField="Reg_Id"
            SelectedValue='<%= Bind("Reg_Id") %>'>
          </asp:DropDownList></td>
      </tr>
      <tr>
        <td valign="top" width="150">
          <h3>Prix (proposé)</h3>
        </td>
        <td colspan="3">
          <asp:TextBox ID="TextBox2" runat="server" Text='<%= Bind("Ann_Prix") %>'
            TextMode="SingleLine" MaxLength="10" Height="20px" Width="150px"
            EnableTheming="False"></asp:TextBox>
        </td>
      </tr>
    </table>
    <asp:Button ID="InsertButton" runat="server" Text="Insérer" />
    <asp:Button ID="CancelButton" runat="server" Text="Annuler" />
  </InsertItemTemplate>
</asp:FormView>
```


[illegible]

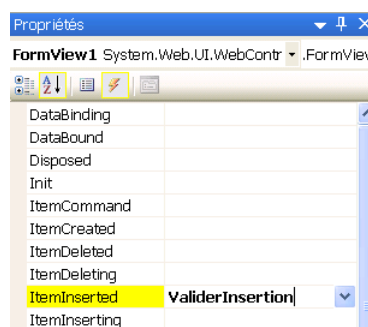
```
<br />
</InsertItemTemplate>
</asp:FormView>
```

Vous obtenez ceci :

7. Il reste une dernière étape pour afficher un message sympathique à l'utilisateur lorsque l'annonce est validée. Pour cela construisez une nouvelle page **DiffuserAnnonceTerminee.aspx** :
 - Glissez sur la page un contrôle Label dans une balise `<h1>`.

```
<h1><asp:Label ID="Label1" runat="server" Text="Label" EnableTheming="false"></asp:Label></h1>
```

8. Revenez sur la page `DiffuserAnnonce.aspx` pour brancher l'appel à la page `DiffuserAnnonceTerminee.aspx` lorsqu'une annonce est insérée dans la base :
 - Ouvrez **DiffuserAnnonce.aspx** en mode **Design**.
 - Affichez la **fenêtre de propriétés** du contrôle **FormView** (F4 ou clic droit > Propriétés).
 - Cliquez sur l'icône  dans la barre d'icônes pour faire apparaître les événements disponibles pour ce contrôle.
 - Lorsqu'une annonce a été insérée dans la base, ASP.NET déclenche l'évènement **ItemInserted**. Indiquez un nom pour la procédure que vous allez définir en réponse à cet évènement : **ValiderInsertion**.



Dans le source de la page, l'attribut **OnItemInserted="ValiderInsertion"** a été ajouté au contrôle **FormView1**.


- Affichez la fenêtre de code de la page en double cliquant sur **ValiderInsertion** directement dans la fenêtre de propriétés.
- Saisissez le code suivant :

Code en C#

```
protected void ValiderInsertion(object sender, FormViewInsertedEventArgs e)
{
    if (e.Exception != null)
    {
        Server.Transfer("DiffuserAnnonceTerminee.aspx?Status=0");
    }
    else
    {
        Server.Transfer("DiffuserAnnonceTerminee.aspx?Status=1");
    }
}
```



Ce code récupère le paramètre **e** de type **FormViewInsertedEventArgs** qui contient des informations sur l'exception en cas d'erreur lors de l'insertion. La fonction `Transfer` de l'objet `Server`, transfère le pointeur de page de ASP.NET sur la page **DiffuserAnnonceTerminee.aspx** en passant en paramètre le statut de l'opération d'insertion : 1 si tout s'est bien passée, 0 sinon.

9. Affichez un message approprié dans le label de la page **DiffuserAnnonceTerminee.aspx** en fonction du statut :
 - Ouvrez le code de la page **DiffuserAnnonceTerminee.aspx** (icône  de la barre d'icône de l'explorateur de solutions)
 - Ajoutez le code suivant à la procédure de chargement de la page web :

Code en C#

```
protected void Page_Load(object sender, EventArgs e)
{
    string IdRegion = Request.QueryString["Status"];

    if( IdRegion == "0" )
    {
        Label1.Text = "Erreur dans l'ajout de votre annonce.  
Recommencez ou contactez-nous au 0.000.000.000";
    }
    Else
    {
        Label1.Text = "Votre annonce sera diffusée sous 24h.  
Vous recevrez un message de confirmation. Merci de votre confiance";
    }
}
```

10. Exécutez la page **DiffuserAnnonce.aspx** pour tester le renvoi sur la page **DiffuserAnnonceTerminee.aspx**.



A VOIR

1. Créer la page **ConsulterAnnoncesJournaux.aspx** de type "Web Form" en l'associant au fichier maître. Ouvrir cette page mode Design.