

## TP 8 : L'accès aux données

Dans une application ASP.NET on peut toujours utiliser le Framework ADO.NET pour l'exploitation des bases de données. Il faut alors une connexion, des commandes et des requêtes. (Voir cours ADO.NET)

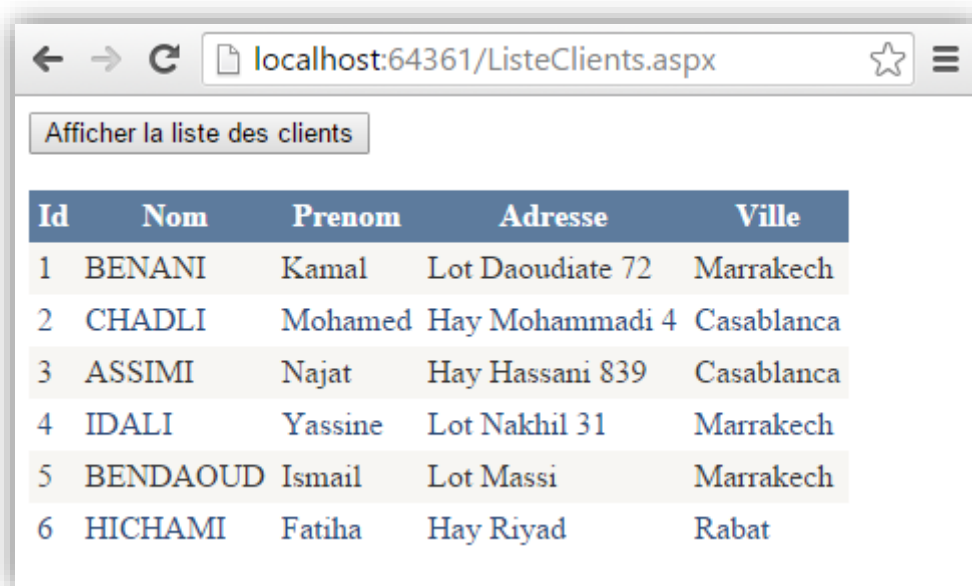
### Atelier 1 : Affichage des données dans un contrôle GridView

Nous allons commencer par mettre en œuvre une page web qui affiche une liste des clients dans un GridView en utilisant le mode connecté du Framework ADO.NET.

Il nous faut alors une source de données. Pour cela, nous utiliserons une base de données 'gestionCmd' avec la table :

Clients (Id, Nom, Prenom, Adresse, Ville).

1. Créez un projet ASP.NET et une page Web ListeClients.aspx
2. Ajoutez un contrôle GridView et un bouton à la page « ListeClients.aspx ».



3. En haut de la page de code behind (ListeClients.aspx.vb), ajouter l'espace de nom SqlClient de la manière suivante :

```
using System.Data;
using System.Data.SqlClient;
```

4. Ajouter dans l'événement « Button1\_Click » le code qui va lier la GridView à une base de données :

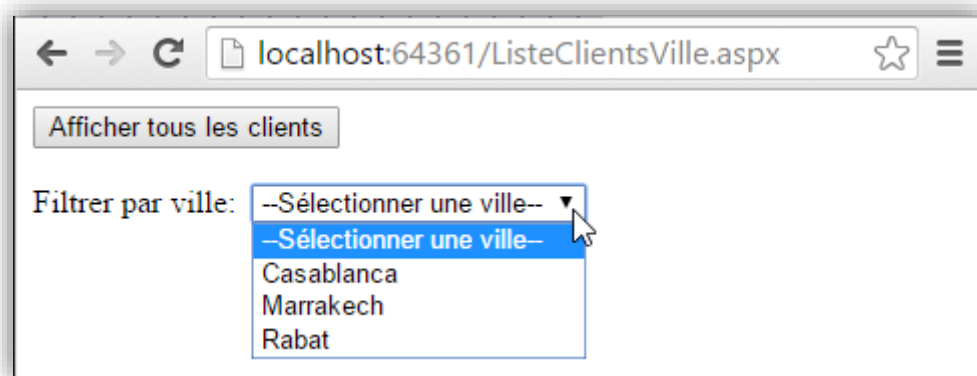
```
protected void Button1_Click(...) {
    SqlConnection cn = new
SqlConnection(@"server=localhost\squlexpress;database=gestionCmd;Integrated
Security=True");

    SqlCommand cmd = new SqlCommand("SELECT * FROM Clients", cn);
    cn.Open();
    GridView1.DataSource = cmd.ExecuteReader();
    GridView1.DataBind();
    cn.Close();
}
```

5. Compilez la page et testez son fonctionnement.
6. Remarquez que si la table Client est vide, le GridView ne sera pas affiché.

## Atelier 2 : Filtrage de la liste des données avec un DropDownList

On souhaite maintenant filtrer la liste des clients par ville. Pour cela nous allons utiliser le contrôle DropDownList.



1. Ajouter un contrôle DropDownList drpVille.
2. Remplir le DropDownList avec la liste des clients. Ajouter le code suivant dans l'événement Page\_Load

```
protected void Page_Load(object sender, EventArgs e) {
    if (!IsPostBack)
    {
        drpVille.AutoPostBack = true;
        SqlConnection cn = new SqlConnection("Votre chaine de connexion");
        SqlCommand cmd = new SqlCommand("SELECT DISTINCT Ville FROM Clients", cn);
        cn.Open();
    }
}
```

```

drpVille.DataSource = cmd.ExecuteReader();
drpVille.DataTextField = "Ville";
drpVille.DataBind();
drpVille.Items.Insert(0, "--Sélectionner une ville--");
cn.Close();
}
}

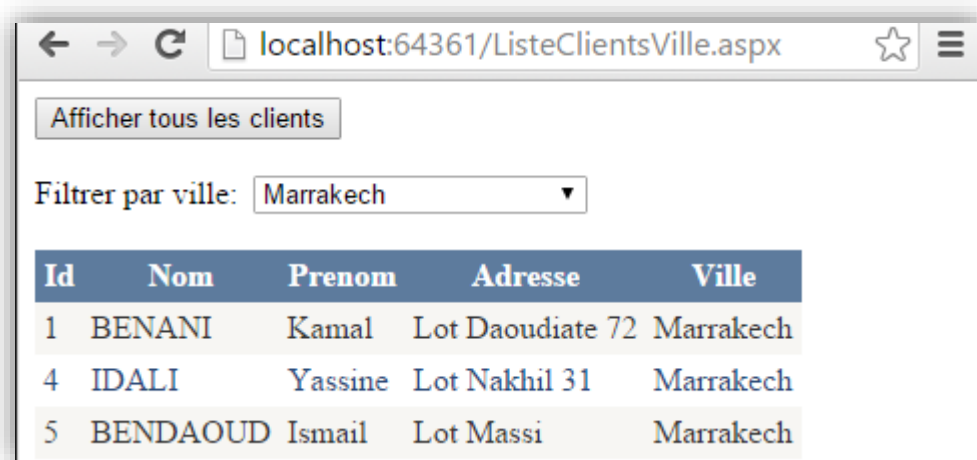
```

3. Double cliquer sur la DropDownList et ajoutez le code suivant :

```

protected void DropDownList1_SelectedIndexChanged(...) {
    SqlConnection cn = new SqlConnection("Votre chaine de connexion");
    string sql = "SELECT * FROM Clients WHERE Ville=@ville";
    SqlCommand cmd = new SqlCommand(sql, cn);
    cmd.Parameters.AddWithValue("@ville", drpVille.SelectedItem.Value);
    cn.Open();
    GridView1.DataSource = cmd.ExecuteReader();
    GridView1.DataBind();
    cn.Close();
}

```



## Exercice 1 : Gestion des clients

Créer une page qui permet la gestion de la table Client avec les opérations de recherche, d'ajout, de suppression et de modification en utilisant le mode connecté.

## Exercice 2 :

L'objectif de cet exercice est de mettre en place un ensemble d'espaces différents assez classiques à savoir un système que l'on retrouve sur la majorité des sites Internet.

Soit à développer une application web pour la gestion des ventes.

Le schéma relationnel de l'application est le suivant :

Client (NumClient, Nom, Adresse, Ville, Tel, Login, Pass)

Commande (NumCmd, dateCmd)

Contenir(NumCmd, NumArticle, quantité)

Article (NumArticle, Designation, Pu, catégorie)

L'application comporte trois espaces :

**Espace Visiteur :**

1. Créer une page qui affiche la liste des articles par catégorie.
2. Créer une page d'inscription pour un client. Dans cette page l'utilisateur doit fournir les informations suivantes relatives au client : Nom(\*), Prénom, Adresse, Ville, Tél (obligatoire et valide), Login et Mot de passe (doit contenir des lettres et des chiffres).
  - Le Numéro client est généré automatiquement
  - La ville sera sélectionnée dans une liste déroulante
3. Créer une page d'authentification pour les clients. Après l'authentification rediriger l'utilisateur vers une page qui affiche ses informations personnelles.

**Espace Client**

4. Créer une page permettant à un client de modifier ses informations personnelles.
5. Créer une page qui affiche les commandes du client connecté.
6. Créer une page de déconnexion d'un client.

**Espace ADMIN :**

7. Créer une page admin.aspx contenant le menu suivant :
  - Gestion des clients
  - Gestion des articles
  - Consultation des commandes

Remarque : L'accès aux pages suivantes ainsi que la page admin.aspx est réservé pour les administrateurs (Créer une table Admin ).