



Module :
Accès aux données à l'aide de
Microsoft ADO.NET

A decorative graphic on the left side of the slide features three balloons: a green one at the top, a blue one in the middle, and a purple one at the bottom. Each balloon has a string and is surrounded by several small yellow triangles, giving the impression of a festive or celebratory theme.

Vue d'ensemble

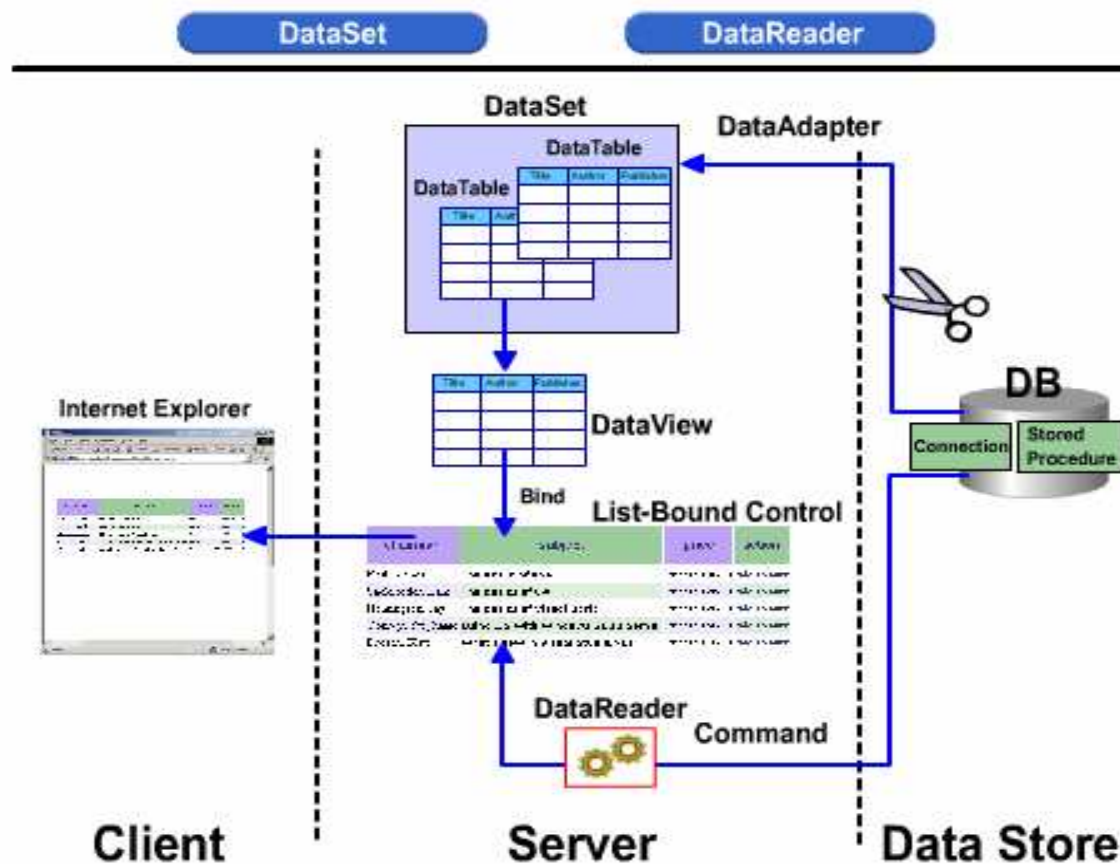
- Présentation de l'utilisation de Microsoft ADO.NET
- Connexion à une base de données
- Accès aux données avec des DataSet
- Utilisation de plusieurs tables
- Accès aux données avec des DataReader



Leçon : Présentation de l'utilisation de Microsoft ADO.NET

- Présentation multimédia : Utilisation d'ADO.NET pour accéder aux données
- Utilisation des DataSet et des DataReader
- Application pratique : Utilisation des DataSet ou des DataReader

Utilisation d'ADO.NET pour accéder aux données



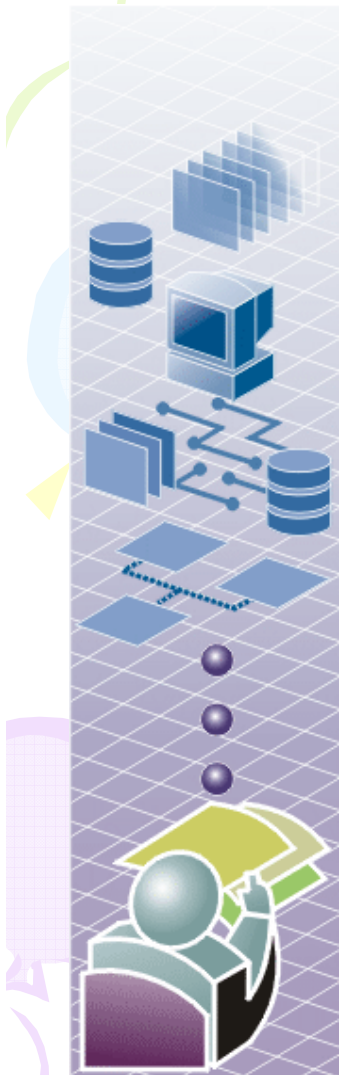


Utilisation des DataSet et des DataReader

DataSet	DataReader
Accès aux données en lecture/écriture	Accès en lecture seule
Comprend plusieurs tables provenant de bases de données distinctes	Basé sur une instruction SQL d'une seule base de données
Déconnecté	Connecté
Lié à plusieurs contrôles	Lié à un seul contrôle
Analyse des données en avant et en arrière	En avant seulement
Accès plus lent	Accès plus rapide
Prise en charge par les outils Visual Studio .NET	Codé manuellement

Application pratique : Utilisation des DataSet ou des DataReader

- Les stagiaires effectueront la tâche suivante:
 - Sélectionner le meilleur accès aux données en fonction de divers scénarios
- Durée approximative : 5 minutes

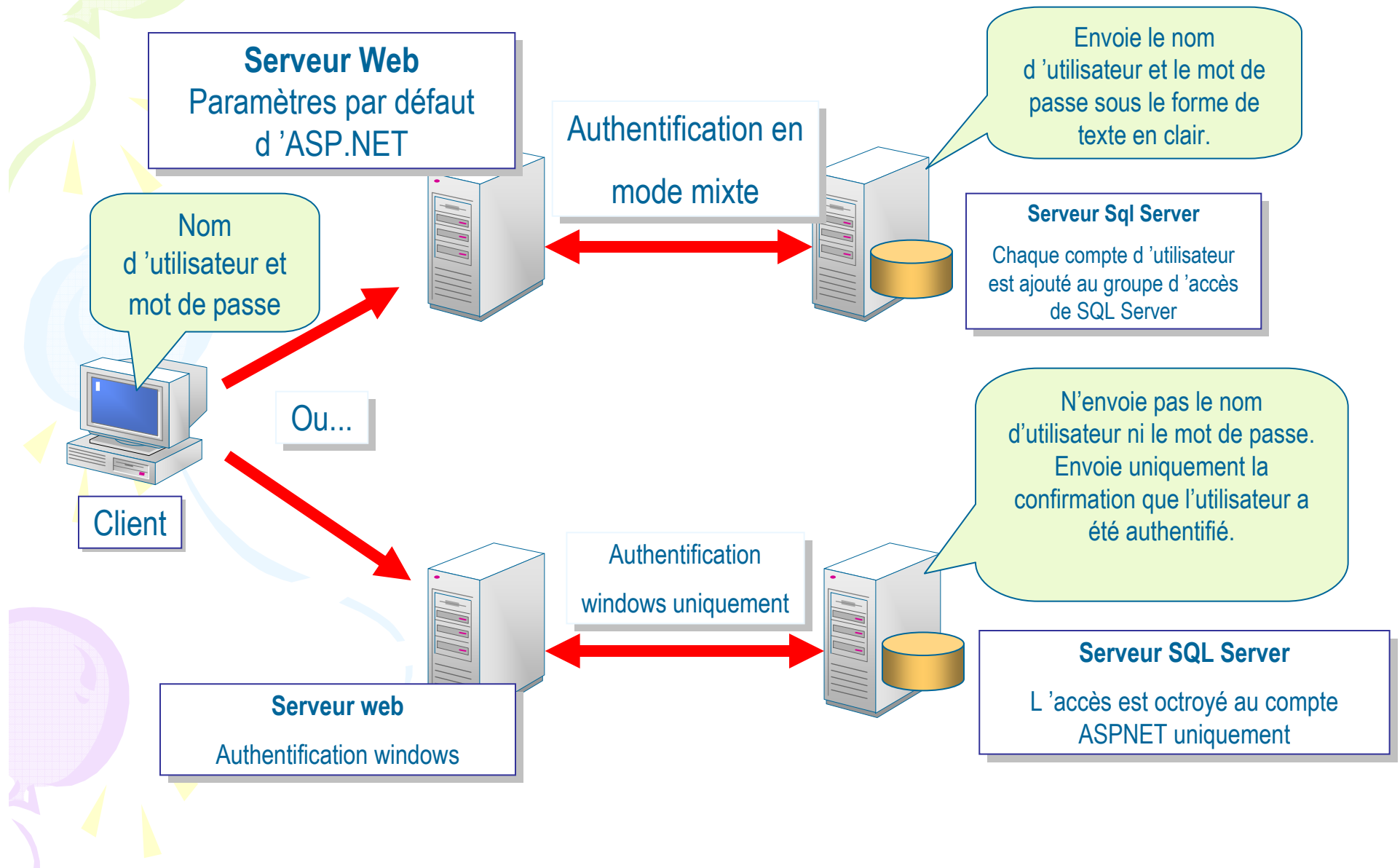




Leçon : Connexion à une base de données

- Sécurité SQL Server
- Création de la connexion
- Démonstration : Définition de la sécurité pour SQL Server

Sécurité SQL Server



Serveur Web
Paramètres par défaut
d'ASP.NET

Authentification en
mode mixte

Envoie le nom
d'utilisateur et le mot de
passe sous la forme de
texte en clair.

Serveur Sql Server
Chaque compte d'utilisateur
est ajouté au groupe d'accès
de SQL Server

Nom
d'utilisateur et
mot de passe

Ou...

N'envoie pas le nom
d'utilisateur ni le mot de passe.
Envoie uniquement la
confirmation que l'utilisateur a
été authentifié.

Client

Authentification
windows uniquement

Serveur SQL Server
L'accès est octroyé au compte
ASPNET uniquement

Serveur web
Authentification windows



Création de la connexion

- Utilisation de SqlConnection

```
Dim strConn As String = "data source=localhost; " & _  
    "initial catalog=northwind; integrated security=true"  
Dim conn As New SqlConnection(strConn)
```

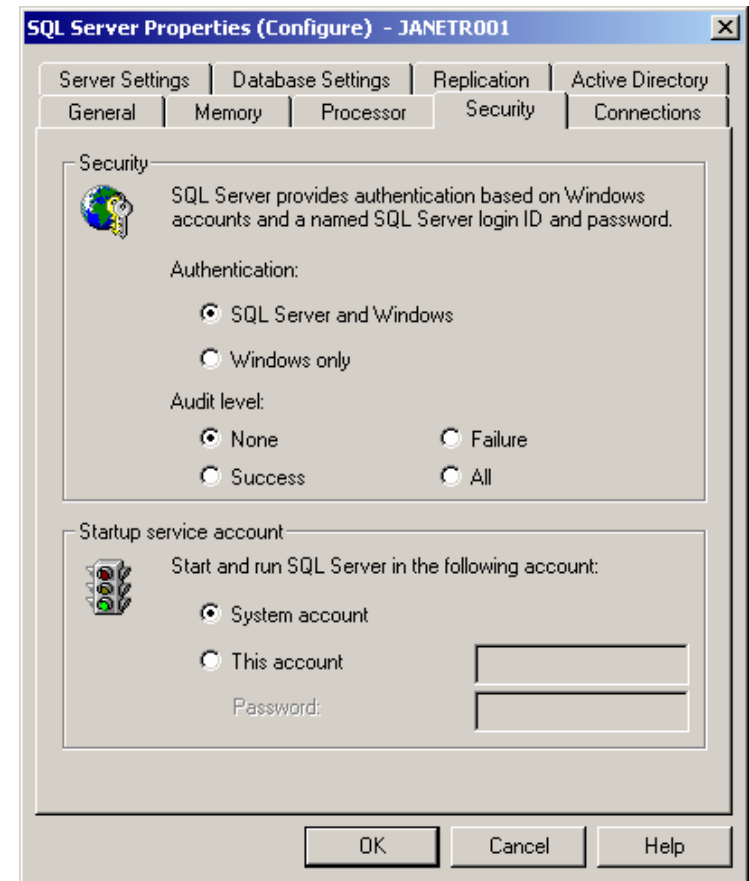
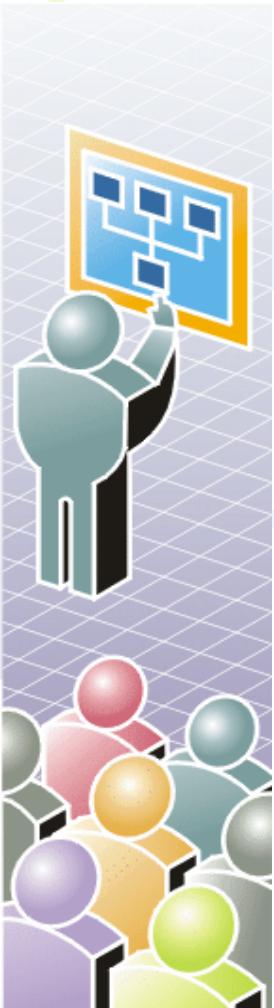


- Définition des paramètres de la chaîne de connexion

- Connection timeout
 - Data source
 - Initial catalog
 - Integrated security
 - Password
 - Persist security info
 - Provider
 - User ID
- 

Démonstration : Définition de la sécurité pour SQL Server

- Ouverture de SQL Server Enterprise Manager
- Définition du mode d'authentification
- Vérification avec la sécurité intégrée
- Vérification avec sécurité en mode mixte





Leçon : Accès aux données avec des DataSet

- Création d'un DataAdapter
- Création d'un DataSet
- Démonstration : Utilisation d'un DataSet par programmation
- Utilisation d'un DataView
- Application pratique : Organisation de code pour créer un DataSet
- Liaison d'un DataSet à un contrôle de liste
- Application pratique dirigée : Affichage d'un DataSet
- Gestion des erreurs



Création d'un DataAdapter

- Stockage de la requête dans un

```
Dim da As New SqlDataAdapter ("select * from Authors", conn)
```

- 
- Le constructeur DataAdapter définit la propriété SelectCommand

```
da.SelectCommand.CommandText  
da.SelectCommand.Connection
```

- 
- Définition des propriétés InsertCommand, UpdateCommand et DeleteCommand si nécessaire



Création d'un DataSet

- Création et remplissage d'un DataSet avec des DataTable
 - La méthode Fill exécute la propriété SelectCommand

```
Dim ds As New DataSet()  
da.Fill(ds, "Authors")
```

```
ds.Tables("Authors").Rows.Count
```

```
Dim r As DataRow  
Dim str As String  
For Each r in _  
    ds.Tables("Authors").Rows  
    str &= r(2)  
    str &= r("au_lname")  
Next
```

Démonstration : Utilisation d'un DataSet par programmation



- Création d'une connexion
- Création d'un DataAdapter
- création d'un DataSet
- lecture de données d'un DataSet par programmation



Utilisation d'un DataView

- Un objet DataView peut être personnalisé pour présenter un sous-ensemble de données d'un DataTable
- La propriété DefaultView retourne la vue par défaut Dataview de la table

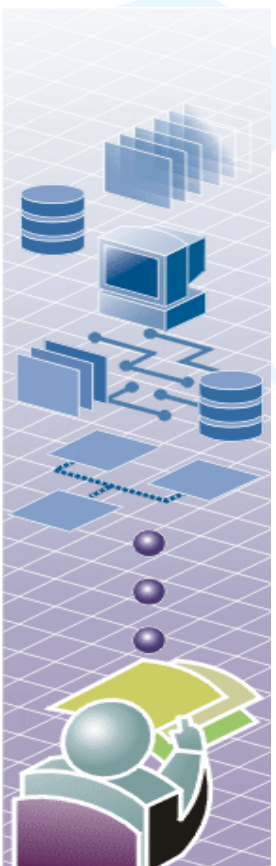
```
Dim dv As DataView = ds.Tables("Authors").DefaultView
```

- Définition d'une autre vue d'un DataSet

```
Dim dv As New DataView (ds.Tables("Authors"))  
dv.RowFilter = "state = 'CA'"
```



Application pratique : Organisation de code pour créer un Dataset

- 
- Les stagiaires devront:
 - Réorganiser des lignes de code pour créer un DataSet
 - Durée approximative : 5 minutes

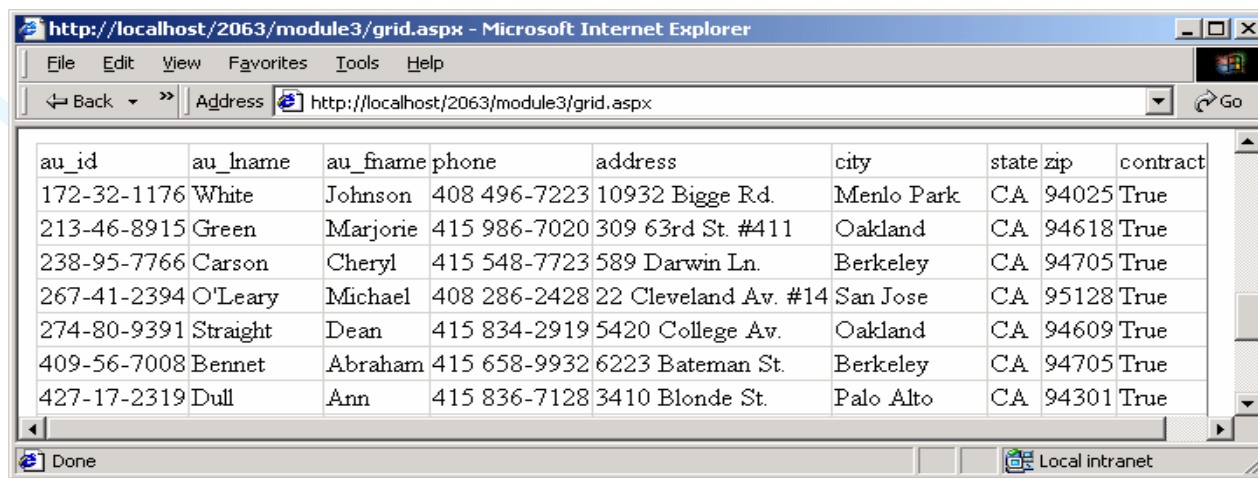
Liaison d'un DataSet à un contrôle de liste

□ Création du contrôle

```
<asp:DataGrid id="dg" runat="server" />
```

□ Liaison à un DataSet ou à un DataView

```
dg.DataSource = ds  
dg.DataMember = "Authors"  
dg.DataBind()
```



The screenshot shows a web browser window displaying a data grid. The grid contains the following data:

au_id	au_lname	au_fname	phone	address	city	state	zip	contract
172-32-1176	White	Johnson	408 496-7223	10932 Bigge Rd.	Menlo Park	CA	94025	True
213-46-8915	Green	Marjorie	415 986-7020	309 63rd St. #411	Oakland	CA	94618	True
238-95-7766	Carson	Cheryl	415 548-7723	589 Darwin Ln.	Berkeley	CA	94705	True
267-41-2394	O'Leary	Michael	408 286-2428	22 Cleveland Av. #14	San Jose	CA	95128	True
274-80-9391	Straight	Dean	415 834-2919	5420 College Av.	Oakland	CA	94609	True
409-56-7008	Bennet	Abraham	415 658-9932	6223 Bateman St.	Berkeley	CA	94705	True
427-17-2319	Dull	Ann	415 836-7128	3410 Blonde St.	Palo Alto	CA	94301	True



Application pratique dirigée : Affichage d'un DataSet



- Création d'une connexion
- Création d'un DataAdapter
- Création d'un DataSet
- Création d'un DataView
- Liaison du DataSet et du DataView à des contrôles DataGridView



Gestion des erreurs

- La connexion ne s'ouvre
 - La chaîne de connexion n'est pas valide
 - Le serveur ou la base de données est introuvable
 - La connexion a échoué
- DataAdapter ne peut pas créer un DataSet
 - La syntaxe SQL n'est pas valide
 - Le nom de table ou de champ n'est pas valide



Leçon: Utilisation de plusieurs tables

- Stockage de plusieurs tables
- Création de relations
- Navigation par programmation entre des tables à l'aide des relations
- Navigation par visuelle entre des tables à l'aide des relations
- Application pratique dirigée :
Affichage de données issues de plusieurs tables

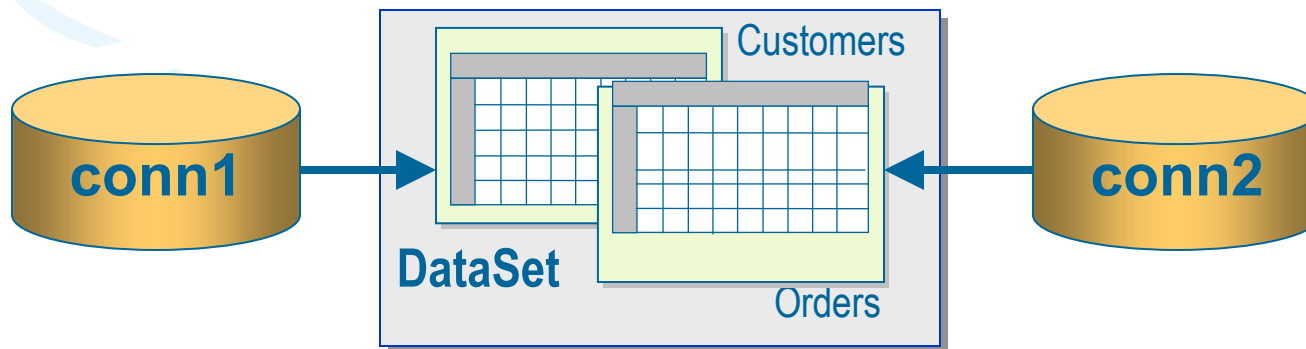
Stockage de plusieurs tables

□ Ajout de la première table

```
daCustomers = New SqlDataAdapter _  
    ("select * from Customers", conn1)  
daCustomers.Fill(ds, "Customers")
```

□ Ajout de tables consécutives

```
daOrders = New SqlDataAdapter _  
    ("select * from Orders", conn2)  
daOrders.Fill(ds, "Orders")
```



Création de relations

Identification de la colonne parente

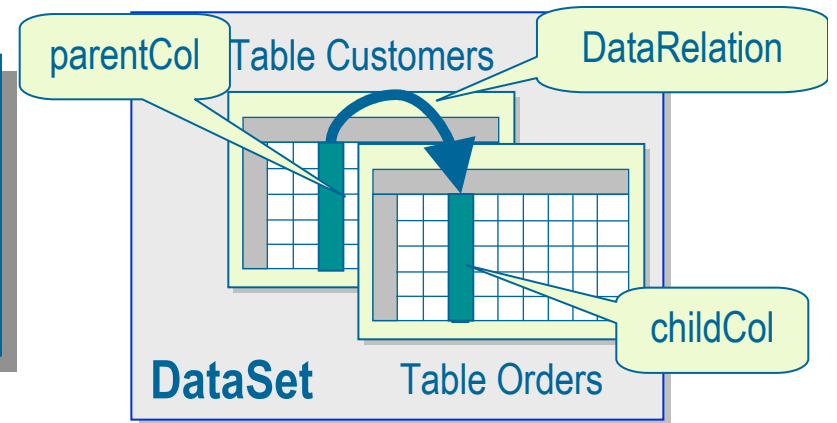
```
Dim parentCol As DataColumn = _  
    ds.Tables("Customers").Columns("CustomerID")
```

Identification de la colonne enfant

```
Dim childCol As DataColumn = _  
    ds.Tables("Orders").Columns("CustomerID")
```

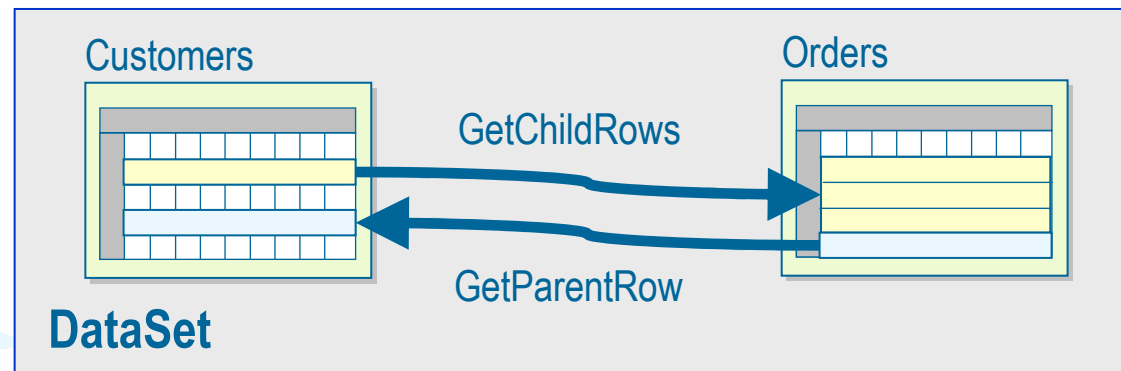
Création de DataRelation

```
Dim dr As New DataRelation _  
    ("name", parentCol, _  
    childCol)  
ds.DataRelations.Add(dr)
```



Navigation par programmation entre des tables à l'aide des relations

```
ds.Tables(index).Rows(index).GetChildRows("relation")  
ds.Tables(index).Rows(index).GetParentRow("relation")
```

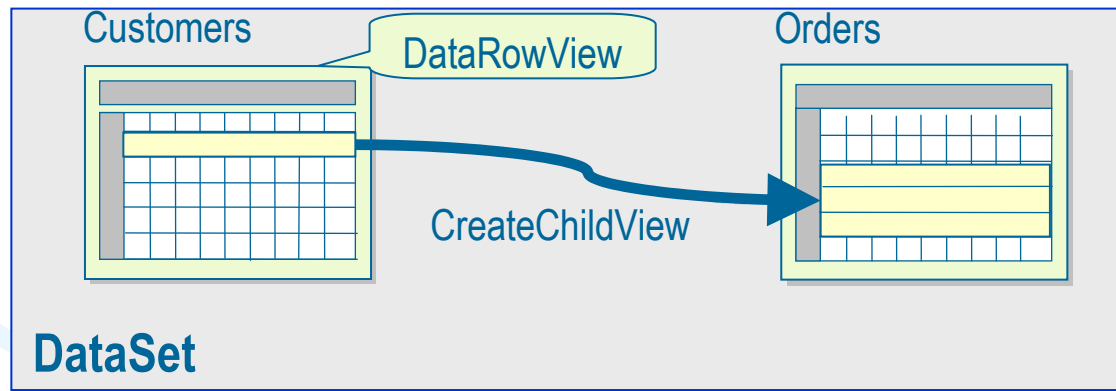




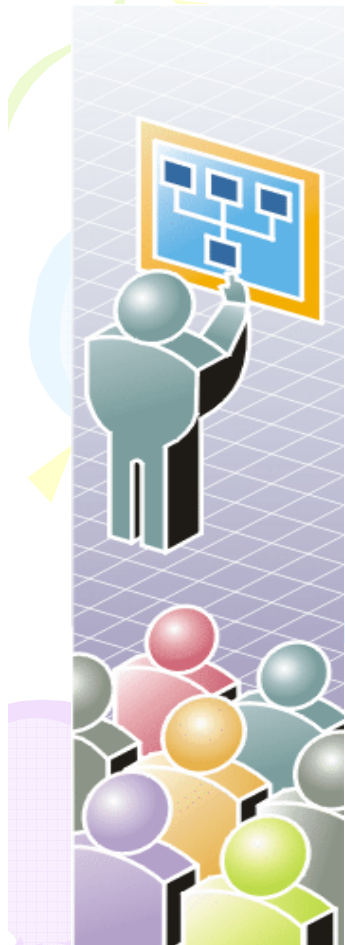
Navigation par visuelle entre des tables à l'aide des relations


```
Dim tableView As DataView
Dim currentRowView As DataRowView

tableView = New DataView(ds.Tables("Customers"))
currentRowView = tableView(dgCustomers.SelectedIndex)
dgChild.DataSource = currentRowView.CreateChildView("CustOrders")
```



Application pratique dirigée : Affichage de donnée issues de plusieurs tables



- Par programmation:
 - Création d 'un **DataSet**
 - Création d 'un **DataRelation**
 - Affichage des enregistrements enfants à l 'aide de DataRelation
- visuellement:
 - Appel de **CreateChildView**



Leçon : Accès aux données avec des DataReader

- Présentation d'un DataReader
- Création d'un DataReader
- Lecture de données à partir d'un DataReader
- Liaison d'un DataReader à un contrôle de liste
- Application pratique : Organisation de code pour créer un DataReader
- Démonstration : Affichage de données à l'aide des **DataReader**



Présentation d'un DataReader

- En avant seulement en lecture seule
- Accès rapide aux données
- Connecté à une source de données
- Gestion manuelle de la connexion
- Gestion manuelle des données ou liaison à un contrôle de liste
- Moins de ressources serveur utilisées



Création d'un DataReader

- Pour utiliser un DataReader:
 - Créez et ouvrez la connexion à la base de données
 - Créez un objet **Command**
 - Créez l'objet **DataReader** à partir de l'objet **Command**
 - Appelez la méthode ExecuteReader
 - Utilisez l'objet DataReader
 - Fermez l'objet DataReader
 - fermez l'objet **Connection**
- Utilisation de la gestion des erreurs



Lecture de données à partir d'un DataReader

- Appel de Read pour chaque enregistrement
 - Retourne Null lorsqu'il n'y a plus d'enregistrement
- Accès aux champs
 - Le paramètre est la position ordinaire ou le nom du champ
 - les fonctions **Get** offrent de meilleures performances

```
Do While myReader.Read()  
    str &= myReader(1)  
    str &= myReader("field")  
    str &= myReader.GetDateTime(2)  
Loop
```

Three decorative balloons are positioned on the left side of the slide. The top balloon is green, the middle one is light blue, and the bottom one is purple. Each balloon has a string and several small yellow triangular flags attached to it. The balloons are arranged in a vertical line, with the green one at the top, the blue one in the middle, and the purple one at the bottom.

Lecture de données à partir d'un DataReader

- Fermeture du DataReader
- Fermeture de la connexion

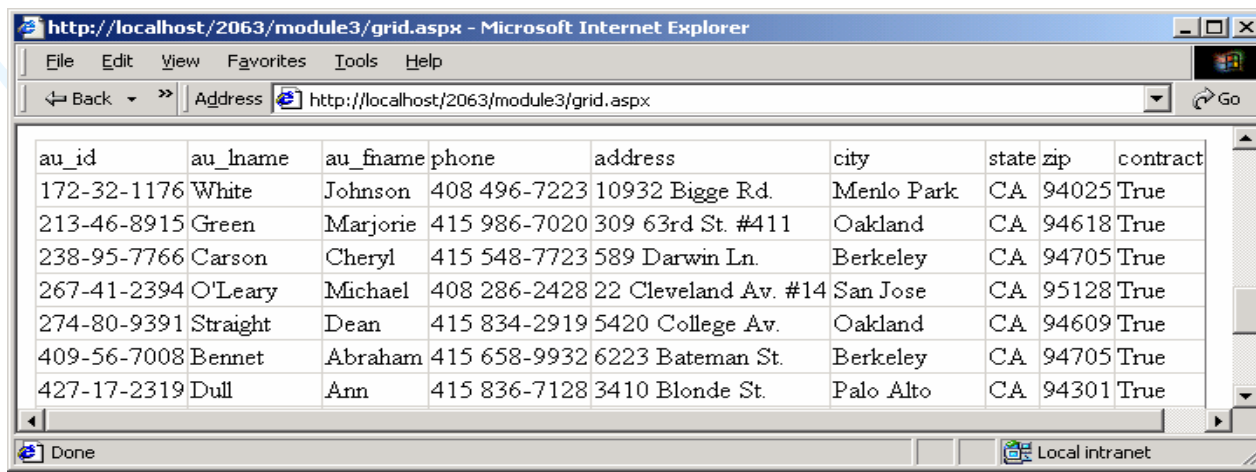
Liaison d'un DataReader à un contrôle de liste

□ Création du contrôle

```
<asp:DataGrid id="dgAuthors" runat="server" />
```

□ Liaison à un DataReader

```
dgAuthors.DataSource = dr  
dgAuthors.DataBind()
```

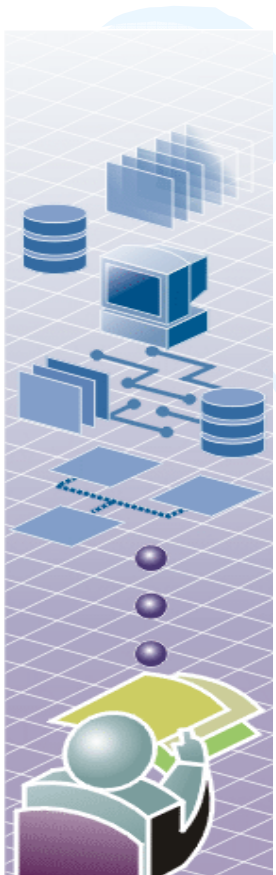


The screenshot shows a web browser window displaying a data grid. The grid contains the following data:

au_id	au_lname	au_fname	phone	address	city	state	zip	contract
172-32-1176	White	Johnson	408 496-7223	10932 Bigge Rd.	Menlo Park	CA	94025	True
213-46-8915	Green	Marjorie	415 986-7020	309 63rd St. #411	Oakland	CA	94618	True
238-95-7766	Carson	Cheryl	415 548-7723	589 Darwin Ln.	Berkeley	CA	94705	True
267-41-2394	O'Leary	Michael	408 286-2428	22 Cleveland Av. #14	San Jose	CA	95128	True
274-80-9391	Straight	Dean	415 834-2919	5420 College Av.	Oakland	CA	94609	True
409-56-7008	Bennet	Abraham	415 658-9932	6223 Bateman St.	Berkeley	CA	94705	True
427-17-2319	Dull	Ann	415 836-7128	3410 Blonde St.	Palo Alto	CA	94301	True



Application pratique : Organisation de code pour créer un DataReader



- Les stagiaires devront:
 - Réorganiser des lignes de code pour créer un DataReader
- Durée approximative : 5 minutes



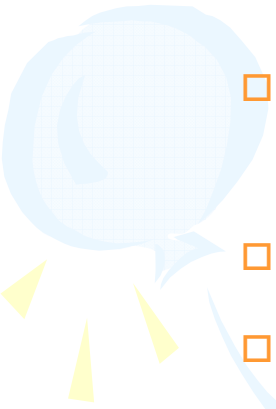
Démonstration : Affichage de données à l'aide des DataReader

- Création d'un SqlConnection
- Création d'un DataReader
- Liaison du DataReader à un ListBox
- Création des éléments du ListBox à partir des données fournies par le DataReader





Contrôle des acquis

- 
- Présentation de l'utilisation de Microsoft ADO.NET
 - Connexion à une base de données
 - Accès aux données avec des DataSet
 - Utilisation de plusieurs tables
 - Accès aux données avec des DataReader
- 