



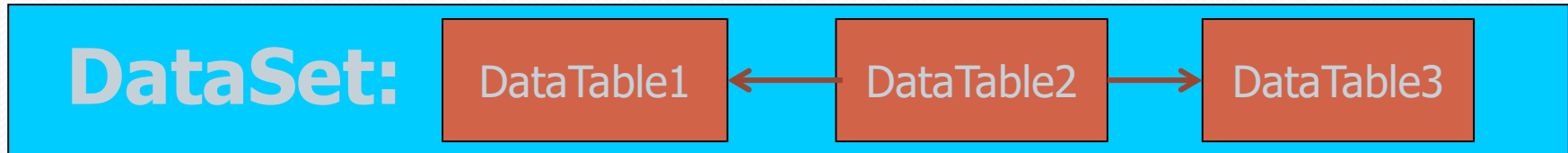
# ADO.NET: Par programmation



# ADO.NET par programmation

- Jusqu'à présent, nous avons utilisé le mode déconnecté en utilisant des assistants mais il est possible d'utiliser le mode déconnecté par programmation.
- Trois objets de base vont être utilisés:
  - Le **Dataset**
  - Les **DataTables** à l'intérieur du **DataSet**
  - Le **DataAdapter**

# L'objet DataSet (1)

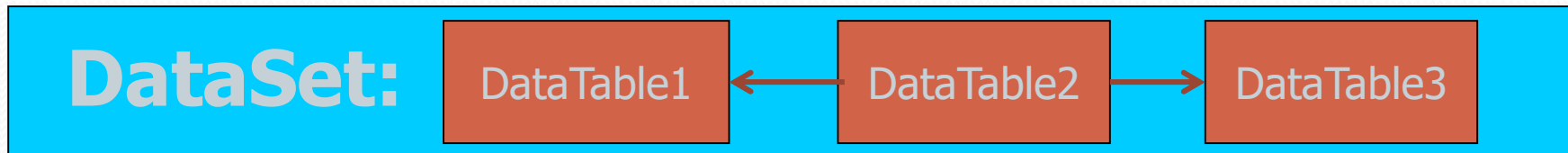


- On crée un nouveau *DataSet* vide de la manière suivante:

```
DataSet monDataSet = new DataSet();
```



# L'objet DataSet (2)



- Éventuellement, ce *DataSet* va contenir des *DataTables*. Plusieurs méthodes ont été définies sur les *DataTables*.

- *Exples:*

Pour vérifier si une *DataTable* existe dans le *DataSet*  
**if (monDataSet.Contains("nomDataTable"))**

Pour supprimer une *DataTable* dans le *DataSet*  
**monDataSet.Remove("nomDataTable")**

# L'objet *DataAdapter* (1)

- Permet de faire le lien entre la base de données réelle et le *DataSet* (situé en mémoire)
- Le *TableAdapter* et le *DataAdapter* sont des objets similaires.
- *ADO.NET* utilise un *TableAdapter* lorsque le *DataSet* est créé à l'aide des assistants.
- Lorsque le *DataSet* est créé par programmation, on utilise habituellement un *DataAdapter*.



# L'objet *DataAdapter* (2)

- Création d'un *DataAdapter* accompagnée d'une requête SELECT.  
**String maRequeteSELECT = "SELECT empNo, empNom  
FROM employe";**  
**SqlDataAdapter monDataAdapter = new  
SqlDataAdapter(maRequeteSELECT, maConnexion);**
- Permet d'exécuter la requête SELECT, de créer et de remplir une *DataTable* dans le *DataSet*  
**monDataAdapter.Fill(monDataSet,"NoNomEmployes" );**

A la suite de l'exécution de cette instruction, la *DataTable* **NoNomEmployes** va exister dans le *DataSet* et va contenir le nom et le prénom de tous les employés.

# L'objet *DataAdapter*(3)

- Il est possible d'utiliser un *DataAdapter* pour insérer, mettre à jour ou supprimer un enregistrement.
- Permet d'insérer un nouvel enregistrement dans la base de données.

```
String maRequeteINSERT = "INSERT ...";  
monDataAdapter.InsertCommand = new  
SqlCommand(maRequeteInsert, maConnexion);  
monDataAdapter.InsertCommand.ExecuteNonQuery();
```

- Même principe pour la mise à jour et la suppression.

```
String maRequeteUPDATE = "UPDATE ...";  
monDataAdapter.UpdateCommand = new  
SqlCommand(maRequeteUpdate, maConnexion);  
monDataAdapter.UpdateCommand.ExecuteNonQuery();
```



# L'objet DataTable (1)

- Constitué d'une collection de *DataRow* et d'une collection de *DataColumn*.
- On peut avoir accès aux données d'une *DataTable*.
- Déterminer le nombre de lignes (ou d'enregistrements) et le nombre de colonnes (ou de champs) dans la *DataTable*.

```
int nbLignes = monDataSet.Tables["noNomEmployes"].Rows.Count;  
int nbColonnes = monDataSet.Tables["noNomEmploye"].Columns.Count
```



# L'objet DataTable (2)

- Accès à la ligne no 1 dans la *DataTable* (cette ligne est de type *DataRow*).

```
DataRow maLigne1 = monDataSet.Tables["noNomEmployes"].Rows[1];
```

- Accès à la colonne no 1 dans la *DataTable* (cette ligne est de type *DataColumn*).

```
DataColumn maColonne1 =  
monDataSet.Tables["noNomEmployes"].Columns[1];
```

- Accès au nom de l'employé (*empNom*) sur la ligne no 1 dans la *DataTable*.

```
string nomLigne1 =  
monDataSet.Tables["noNomEmployes"].Rows[1]["empNom"];
```

# L'objet DataTable (3)

- Parcourir la *DataTable* et afficher son contenu:

```
DataTable maDataTable = monDataSet.Tables["noNomEmployes"];  
foreach (DataRow maLigne in maDataTable.Rows)  
    foreach (DataColumn maColonne in maDataTable.Columns )  
        Console.WriteLine(maLigne[maColonne].ToString());
```



# L'objet DataTable (4)

- Il est également possible de supprimer une ligne dans une **DataTable**.

```
monDataSet.Tables["noNomsEmploye"].Rows[1].Delete();
```

- Il est également possible d'insérer une ligne dans une **DataTable**.

```
DataRow maLigne =  
    monDataSet.Tables["noNomEmployes"].NewRow();  
maLigne["empNo"] = 5000;  
maLigne["empNom"] = "Joseph";  
monDataSet.Tables["noNomsEmploye"].Rows.Add(maLigne);
```