



3BM

EXPLOITATION DE MS ACCESS À PARTIR DE C#

Philippe Dekimpe

OLE DB

- OLE DB (Object Linking and Embedding Data Base) est une API développée par Microsoft permettant l'accès universel aux sources de données.
- OLE DB se sert d'interfaces COM (Component Object Model)
- Il remplace ODBC (Open Database Connectivity) et permet l'accès à des moteurs de données différents de SQL
- Installer le composant
 - **Microsoft Access Database Engine 2016 Redistributable**



CONNEXION OLEDB EN C#

- Déclarer un alias à l'espace de noms
`using System.Data.OleDb;`
- Déclarer une connexion OleDb
`OleDbConnection dbConn;`
- Déclarer l'objet contenant la chaîne de connexion
`dbConn = new OleDbConnection(@"Provider=Microsoft.ACE.OLEDB.16.0;
Data Source=C:\bus.accdb");`
- L'objet peut être déclaré comme attribut de la classe Form pour éviter de devoir le redéclarer dans chaque méthode



OUVRIR ET FERMER LA CONNEXION

```
try
{
    // open connection
    dbConn.Open();

    // next instructions ...
}
catch (Exception ex)
{
    MessageBox.Show("Failed to connect to data source " + ex.ToString());
}
finally
{
    // Disconnect Database
    dbConn.Close();
}
```



- Si vous constatez une erreur lors de la connexion OleDb
 - allez le menu Projet / Propriétés de ...
 - choisissez x64 dans Build
 - enregistrez

Application

Build*

Événements de build

Déboguer

Ressources

Services

Paramètres

Chemins des références

Configuration : (Debug) active

Plateforme : (Any CPU) active

Général

Symboles de compilation conditionnelle :

☒ Définir la constante DEBUG

☒ Définir la constante TRACE

Plateforme cible : x64



OLEDBCOMMAND : L'OBJET QUI GÈRE LA COMMANDE

- OleDbCommand command = new OleDbCommand(queryString, connection);

```
// open connection  
dbConn.Open();
```

```
// declare and set the string command  
string queryString = "SELECT * FROM vin ";
```

```
// declare OleDb command object  
OleDbCommand cmd = new OleDbCommand(queryString, dbConn);
```



OLEDBDataReader : L'OBJET QUI LIT LES DONNÉES

- Fournit un moyen de lire un flux forward-only (lecture vers l'avant uniquement) de lignes de données depuis une source de données.

```
// open connection
dbConn.Open();

// declare and set the string command
string queryString = "SELECT * FROM vin ";

// declare OleDb command object
OleDbCommand cmd = new OleDbCommand(queryString, dbConn);

// declare data reader object
OleDbDataReader dbReader = null;

// execute command
dbReader = cmd.ExecuteReader();
```



LIRE UNE LIGNE

```
// open connection
dbConn.Open();

// declare and set the string command
string queryString = "SELECT * FROM vin ";

// declare OleDb command object
OleDbCommand cmd = new OleDbCommand(queryString, dbConn);

// declare data reader object
OleDbDataReader dbReader = null;

// execute command
dbReader = cmd.ExecuteReader();

// Show first row
dbReader.Read();

// add to label
label1.Text = dbReader[0].ToString();
```



BOUCLE DE LECTURE DU RÉSULTAT DE LA COMMANDE

- A chaque passage dans le while, le méthode Read lit la ligne suivante :

```
while (dbReader.Read())
{
    string id_vin = dbReader["id_bouteille"].ToString();
    int qtt = dbReader.GetInt32(1);
    float prix = dbReader.GetFloat(2);

    // do something on data
}
```

- Le champ récupéré peut être soit :
 - désigné par son nom, la valeur est alors de type chaîne
 - désigné par sa position dans le query en partant de 0



NAVIGUER ENTRE ENREGISTREMENTS

- Déclarer un objet DataTable comme attribut de la classe
- Stocker le résultat dans une DataTable

```
// Declare DataTable as class attribute
// DataTable dt_perma = new DataTable();

// save query results in the datatable
dt_perma.Load(dbReader);

// Show first row
lb_cepage.Text = dt_perma.Rows[0][ "nom" ].ToString();

// next record
index_dt++;
if (index_dt >= 1 && index_dt < dt_perma.Rows.Count )
    lb_cepage.Text = dt_perma.Rows[index_dt][ "nom" ].ToString();
```



AFFICHER LE RÉSULTAT DANS UNE GRILLE

1. Stocker le résultat dans une DataTable
2. Afficher la DataTable dans une DataGrid

```
// save query results in the datatable  
DataTable dt = new DataTable();  
dt.Load(dbReader);
```

```
// show the datatable in the datagrid  
dataGridView1.DataSource = dt;
```



PARAMÉTRER UNE REQUÊTE

- Introduire les paramètres dans la requête :

```
string queryString = "SELECT * FROM vin_domaine where domaine like '%" +  
tb_domaine.Text + "%'";
```

Une ligne Liste Cépage Navigation Grille cépage Recherche Ajout cépage Commande

Rechercher un domaine

Bat

| | nom | domaine | millesime | quantite |
|---|-----------|------------------|-----------|----------|
| ▶ | Merlot | Chateau Batalley | 2010 | 24 |
| | Cabemet | Chateau Batalley | 2010 | 24 |
| | Grenache | Chateau Batalley | 2010 | 24 |
| | Bourgueil | Chateau Batalley | 2010 | 24 |
| | Cabemet | Chateau Batalley | 2014 | 12 |
| * | | | | |



INSÉRER UN ENREGISTREMENT

```
string queryString = "insert into cepage (nom) VALUES ('" + tb_cepage.Text + "') ";

// declare OleDb command object
OleDbCommand cmd = new OleDbCommand(queryString, dbConn);

// execute insert command
int nb_row = cmd.ExecuteNonQuery();
if (nb_row == 1)
    label.Text = "Enregistrement réussi ";
```

