



Forum: WD 11

Topic: [Résolu] Connexion SQL Server 2000 & WinDev 11

Subject: Re: Connexion SQL Server 2000 & WinDev 11

Posté par: drcharly93

Contribution le : 11/1/2009 16:38:30

Bonjour Unizuka,

Voilà la technique que j'utilise

1 - les paramètres de connexion

tous les paramètres de connexion sont stockés dans un fichier ini.

Cet technique facilite la portabilité de l'application, puisqu'il n'y aura pas besoin de recompiler l'application avec les nouveaux paramètres puisqu'ils seront enregistrés dans un fichier ini : **<Nom de l'application>.ini**

2 - Chargement des paramètres de connexion

dans la déclaration globale de mon application je créer une structure permettant de manipuler mes paramètres

Au début de l'application création d'une structure adaptée

//Initialisation de <MonProjet> StructureCnx est une Structure NumConnexion est un entier
NomSource est une chaîne Utilisateur est une chaîne MotDePasse est une chaîne
NomDataBase est une chaîne TypeBase est une chaîne Provider est une chaîne Infos est
une chaîne FIN gstCnx est une StructureCnx

Ensuite je créer une procédure pour charger les paramètres de connexion

```
PROCEDURE proLectureFichierIni() sFichierIni est une chaîne sFichierIni =  
ComplèteRep(fRepExe()) + ProjetInfo(piNomProjet) + ".ini" gstCnx:NomSource =  
INILit("Connexion","NomSource","",sFichierIni) SI gstCnx:NomSource = "" ALORS  
INIEcrit("Connexion","NomSource","",sFichierIni) gstCnx:Utilisateur =  
INILit("Connexion","Utilisateur","",sFichierIni) SI gstCnx:Utilisateur = "" ALORS  
INIEcrit("Connexion","Utilisateur","",sFichierIni) gstCnx:MotDePasse =  
INILit("Connexion","MotDePasse","",sFichierIni) SI gstCnx:MotDePasse = "" ALORS  
INIEcrit("Connexion","MotDePasse","",sFichierIni) gstCnx:NomDataBase=  
INILit("Connexion","NomDataBase","",sFichierIni) SI gstCnx:NomDataBase = "" ALORS  
INIEcrit("Connexion","NomDataBase","",sFichierIni) gstCnx:TypeBase =  
INILit("Connexion","TypeBase","",sFichierIni) SI gstCnx:TypeBase = "" ALORS  
INIEcrit("Connexion","TypeBase","",sFichierIni) gstCnx:Provider =  
INILit("Connexion","PROVIDER","",sFichierIni) SI gstCnx:Provider = "" ALORS  
INIEcrit("Connexion","Provider","ODBC",sFichierIni)
```

Je créer ensuite une procédure proConnexionFerme

```
PROCEDURE ConnexionFerme() SQLDéconnecte()
```

Puis une procédure proConnexionOuvre

```
PROCEDURE ConnexionOuvre() gstCnx:NumConnexion = SQLConnecte( gstCnx:NomSource,  
gstCnx:Utilisateur, gstCnx:MotDePasse, gstCnx:NomDataBase, gstCnx:TypeBase, gstCnx:Provider,  
gstCnx:Infos ) SI gstCnx:NumConnexion = 0 ALORS SQLInfoGene() Erreur(SQL.Erreur + RC +  
SQL.MesErreur) RENVOYER Faux FIN RENVOYER Vrai
```

Exemple de paramètre à une base SQLServeur
Citation :

```
[Connexion]
Rem Nom du serveur
NomSource=TSE
Rem Nom de l'Utilisateur
Utilisateur=sa
Rem Mot de passe
MotDePasse=MotDePasse
Rem Nom de la base de donnees dans SQLServer
NomDataBase=WDFORGE_SQL
Rem OLEDB acces OLEDB
TypeBase=OLEDB
Rem SQLOLEDB nom du provider dans WinDev
Provider=SQLOLEDB
```

Exemple de code avec exécution de code SQL et parcours

```
sSql est une Chaine Rs est une surce de donnees //Connexion à la base SI PAS
proConnexionOuvre() Alors SqlInfoGene( gstCnx:NomSource ) Erreur( Sql.MesErreur )
Renvoyer Faux) Fin sSql = "SELECT RE_No, RE_Nom FROM F_REPRESEN" //Exécution de la
requête SI PAS SQLExec( sSql, "Rs" ) ALORS SQLInfoGene( "Rs" ) Erreur( "Erreur dans la
requete : " + Sql , Sql.MesErreur ) RENVOYER Faux FIN //Chargement de la requête
SQLPremier( Rs ) //Très important SI Sql.NbLig = 0 ALORS //Récupère le nombre de ligne
RENVOYER Faux FIN //Parcours des enregistrements TANTQUE PAS SQL.EnDehors //
SQLCol( Rs, 1 ) //Récupère la colonne1 : RE_No //SQLCol( Rs, 2 ) //Récupère la colonne2 :
RE_Nom SQLSuivant( Rs ) FIN proConnexionFerme()
```

Code saisie à la volée pouvant contenir des erreurs