



FORUMS

TUTORIELS

MAGAZINE

FAQ

BLOGS

CHAT

NEWSLETTER

ÉTUDES

Navigation

ALM

Java

.NET

Dév. Web

EDI

Programmation

SGBD

Office

Solutions d'entreprise

Applications

Mobiles

Systèmes

.NET

Visual Studio

ASP.NET

C#

Visual Basic.NET

Windows Phone

Microsoft Azure

FORUMS.NET

FAQ.NET

TUTORIELS.NET

VIDÉOS.NET

SOURCES.NET

LIVRES.NET

OUTILS.NET

AZURE DEV CAI

Identifiant

Mot de passe

S'identifier

Aide

S'inscrire

Actions

Réseau social

Blogs

Agenda

Règles

Recherche avancée

propres questions et recevoir la newsletter

JE M'INSCRIS

Forum Dotnet Accès aux données [C#] Comment faire un simple SELECT à SQL Server ?

+ Répondre à la discussion



Discussion : **[C#] Comment faire un simple SELECT à SQL Server ?** ✓  
Sujet : *Accès aux données*

Outils de la discussion Affichage

14/07/2006, 17h25

#1

**Rodie**

En attente de confirmation mail



Inscrit en: mai 2006  
Messages: 171  
Points: 122

### [C#] Comment faire un simple SELECT à SQL Server ?

Bonjour à tous 😊

J'ai un site Web où des utilisateurs peuvent créer des comptes perso en donnant leur pseudo.  
Avant d'insérer leur pseudo, je dois vérifier qu'il n'existe pas déjà.

Code : [Sélectionner tout](#) - [Visualiser dans une fenêtre à part](#)

```
1 string command "select id_member from member where pseudo='rober"
2 SqlDataSource dataSource = new SqlDataSource(connectionString, c
3 this.mygridView.DataSource = dataSource;
4 this.mygridView.DataBind();
5 if (mygridView.Rows.Count > 0)
6 {
7     // pseudo already exists
8 }
```

C'est ce que j'ai l'habitude de faire, et je sais que c'est bien pour remplir un gridView mais nul pour tester juste l'existence d'un pseudo en base.  
Savez-vous comment je peux simplifier mon programme ?  
Mici bien 😊

Répondre avec citation 0 0

14/07/2006, 17h33

#2

**StormimOn**

Expert éminent



Développeur informatique  
Inscrit en: mai 2005  
Messages: 2 593  
Points: 7 684



Tu peux modifier ta requête ainsi

Code : [Sélectionner tout](#) - [Visualiser dans une fenêtre à part](#)

```
select count(*) from member where pseudo='robert'
```

et l'exécuter avec la méthode ExecuteScalar afin de récupérer le nombre de lignes concernées, à savoir 0 ou 1 car ton pseudo doit être unique je suppose.

Pour plus d'infos, [ExecuteScalar sur MSDN2](#) 😊

⚠ Pas de questions techniques par MP

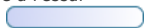
Répondre avec citation 0 0

14/07/2006, 17h39

#3

**totoche9**

Membre à l'essai



Inscrit en: septembre 2004



raté !!

donc, pourquoi ne pas faire un count et tester si tu as > 0 avant de lier ta datagrid à ta datasource.

Messages: 32  
Points: 24

Ou bien passer par un datareader et un ExecuteReader

Tu fais une première lecture. Si elle échoue c'est que ton select n'a rien ramené sinon tu lie ta datasource.

Espérant t'avoir aidé

Répondre avec citation

0

0

14/07/2006, 18h37

#4

**Rodie**

En attente de confirmation mail



Inscrit en: mai 2006  
Messages: 171  
Points: 122



Quoi raté ? ze savais que c'était nul 😞

Voila ce que j'ai fait :

Code :

[Sélectionner tout](#) - [Visualiser dans une fenêtre à part](#)

```
1 using (SqlConnection connection = new SqlConnection(connectionString)
2 {
3     connection.Open();
4     SqlCommand cmd = new SqlCommand(command, connection);
5     result = cmd.ExecuteScalar();
6     connection.Close();
7 }
8
9 if (result == null)
10 {
11     // pseudo already exists
```

Merci bien à vous

Répondre avec citation

0

0

14/07/2006, 18h47

#5

**StormimOn**

Expert éminent



Développeur informatique  
Inscrit en: mai 2005  
Messages: 2 593  
Points: 7 684



Envoyé par **Rodie**

Merci bien à vous

Je suppose donc que c'est résolu ? Si c'est le cas, clique sur le bouton résolu en bas de page 😊

⚠ Pas de questions techniques par MP

Répondre avec citation

0

0

14/07/2006, 19h37

#6

**Rodie**

En attente de confirmation mail



Inscrit en: mai 2006  
Messages: 171  
Points: 122



Euh... c'est presque résolu 😊

J'ai ajouté une transaction et j'ai le code suivant :

Code :

[Sélectionner tout](#) - [Visualiser dans une fenêtre à part](#)

```
1 SqlTransaction transaction = null;
2 try
3 {
4     using (SqlConnection connection = new SqlConnection(connectionString)
5     {
6         connection.Open();
7         transaction = connection.BeginTransaction();
8         SqlCommand sqlCommand = new SqlCommand(command, connection)
9         sqlCommand.Transaction = transaction;
10        throw new Exception("not working out");
11        sqlCommand.ExecuteScalar();
12        transaction.Commit();
13    }
```

Le petit problème, c'est que lorsque je fais le **throw new Exception**, je fais bien dans le **catch** en bas mais la transaction est déjà consommée alors le **transaction.Rollback()** plante.

Ce **Rollback()** est-il complètement inutile du fait du **transaction.Commit()** au-dessus ?

Et mon code est-il du code de porc ? 🐷

Navigation

**Inscrivez-vous gratuitement**  
pour pouvoir  
participer, suivre les  
réponses en temps  
réel, voter pour les  
messages, poser vos  
propres questions et  
recevoir la newsletter

**JE M'INSCRIS**

Mici bien pour vos réponses en tout cas 😊 !!

Répondre avec citation | 0 | 0

14/07/2006, 20h07

#7

**StormimOn** ◯

Expert éminent



Développeur informatique

Inscrit en: mai 2005

Messages: 2 593

Points: 7 684



La raison est très simple, c'est à cause de ton bloc using. Quand tu sors de ce bloc, naturellement ou parce qu'une exception est levée, la méthode Dispose de l'objet, dans notre cas SqlConnection, est appelée ce qui au passage ferme la connexion et donc les transactions ne sont plus utilisables.

Essaye comme ça. C'est fait sans tester, mais ça devrait aller je pense.

Code : [Sélectionner tout](#) - [Visualiser dans une fenêtre à part](#)

```
1 bool error = false;
2 SqlTransaction transaction = null;
3 SqlConnection connection = new SqlConnection(connectionString);
4
5 try
6 {
7     connection.Open();
8     transaction = connection.BeginTransaction();
9     SqlCommand sqlCommand = new SqlCommand(command, connection);
10    sqlCommand.Transaction = transaction;
11    throw new Exception("not working out");
12    sqlCommand.ExecuteScalar();
13 }
```

Je préfère mettre dans le bloc finally tout ce qui est validation des transactions et fermeture de la connexion 😊

❗ Pas de questions techniques par MP

Répondre avec citation | 0 | 0

14/07/2006, 21h31

#8

**StormimOn** ◯

Expert éminent



Développeur informatique

Inscrit en: mai 2005

Messages: 2 593

Points: 7 684



📧 Envoyé par Rodie

*Tant que tu n'as pas fait de **transaction.Commit()**, alors la BdD (Base de Données) n'a pas été modifiée.  
Donc quel est l'intérêt de faire un **transaction.Rollback()***

On va prendre un cas tout simple. Sur une transaction tu as un traitement chargé de faire certaines opérations et pendant celui-ci une erreur survient. Cependant tu ne peux pas te permettre, pour l'intégrité du système, de conserver ce qui a été fait. Il faut donc pouvoir revenir dans un état stable et connu.

Un autre cas courant, ton commit ne passe pas il faut donc pouvoir annuler ce qui a été fait.

L'intérêt du rollback est tout simplement de pouvoir annuler tout ce qui a été fait pendant la transaction et c'est très important de pouvoir le faire 😊

❗ Pas de questions techniques par MP

Répondre avec citation | 0 | 0

14/07/2006, 21h49

#9

**Rodie** ◯

En attente de confirmation mail



Inscrit en: mai 2006

Messages: 171

Points: 122



Oh, je viens juste de virer ma question qui est devenue stupide dès que j'ai testé mon bout de code 😊

Tu as tout à fait raison !

Le cmd.ExecuteScalar() fait son opération sur la BdD et attend le transaction.Commit() pour finaliser.

J'ai une autre tite question 😊

Je veux récupérer l'ID de la ligne que j'ai insérée (l'ID est le 1er champ de la table) Le problème, c'est que le ExecuteScalar() du code suivant retourne toujours null.

Code : [Sélectionner tout](#) - [Visualiser dans une fenêtre à part](#)

```
1 using (SqlConnection connection = new SqlConnection(connectionString))
2 {
3     connection.Open();
4     SqlCommand sqlCommand = new SqlCommand(command, connection);
5     object id = sqlCommand.ExecuteScalar();
6 }
```

Navigation

**Inscrivez-vous gratuitement**

pour pouvoir participer, suivre les réponses en temps réel, voter pour les messages, poser vos propres questions et recevoir la newsletter

**JE M'INSCRIS** ➔

C'est normal ?  
Est-ce le bon moyen pour récupérer l'ID de la ligne insérée ??

Merci énormément à toi !!!!!!!! 😊😊😊

Répondre avec citation | 0 | 0

14/07/2006, 21h58

#10

**StormimOn** ◊

Expert éminent



🇫🇷   
Développeur informatique  
Inscrit en: mai 2005  
Messages: 2 593  
Points: 7 684



Il faut procéder autrement.

Si ta requête est

Code : [Sélectionner tout](#) - [Visualiser dans une fenêtre à part](#)

```
INSERT INTO maTable VALUES(...)
```

modifie la requête ainsi

Code : [Sélectionner tout](#) - [Visualiser dans une fenêtre à part](#)

```
INSERT INTO maTable VALUES(...); SELECT @@IDENTITY
```

Tu exécutes cette requête avec un ExecuteScalar et tu pourras ainsi récupérer l'id généré automatiquement par la base.

Sinon, n'utilise pas le ExecuteScalar pour des insertions/suppressions (à part si tu veux récupérer l'id inséré, comme dans le cas présent). Utilise plutôt [ExecuteNonQuery](#).

⚠ **Pas de questions techniques par MP**

Répondre avec citation | 0 | 0

14/07/2006, 22h05

#11

**Rodie** ◊

En attente de confirmation mail



🇫🇷   
Inscrit en: mai 2006  
Messages: 171  
Points: 122



Aïe, honte à moi pour le select **SELECT @@IDENTITY** !!  
Quant au ExecuteNonQuery(), j'en avais entendu parler mais la doc dit :  
[Execute a T-SQL statement against the connection...](#)  
C'est pas dangereux le "against connection" parce que je ne vois pas ce que ça veut dire 😞

Mici beaucoup !! et bon 14 juillet !! 😊

Répondre avec citation | 0 | 0

14/07/2006, 22h07

#12

**StormimOn** ◊

Expert éminent



🇫🇷   
Développeur informatique  
Inscrit en: mai 2005  
Messages: 2 593  
Points: 7 684



🗉 Envoyé par Rodie

Quant au ExecuteNonQuery(), j'en avais entendu parler mais la doc dit :  
[Execute a T-SQL statement against the connection...](#)  
C'est pas dangereux le "against connection" parce que je ne vois pas ce que ça veut dire 😞

Non rien de dangereux 🟢  
En français on traduirait ça par

*Exécute une instruction Transact-SQL sur la connexion*

⚠ **Pas de questions techniques par MP**

Répondre avec citation | 0 | 0

14/07/2006, 22h13

#13

**Rodie** ◊

En attente de confirmation mail



🇫🇷   
Inscrit en: mai 2006



**[Résolu]**

Ah, mon anglais est pourri alors 🇫🇷  
Bon, sur ces mots, je tag ce post à résolu 😊  
Miciiiiiiii encore 😊

Navigation

**Inscrivez-vous gratuitement**

pour pouvoir participer, suivre les réponses en temps réel, voter pour les messages, poser vos propres questions et recevoir la newsletter

**JE M'INSCRIS** ➡

Messages: 171  
Points: 122

Répondre avec citation | 0 | 0

14/07/2006, 22h22

#14

Nip

Rédacteur



Inscrit en: juin 2004  
Messages: 963  
Points: 1 044



Juste une précision: dans ton cas utilise SCOPE\_IDENTITY() plutôt que @@IDENTITY.  
Pourquoi? Parce que @@IDENTITY retourne la dernière valeur IDENTITY générée pendant ta connexion; SCOPE\_IDENTITY() retourne la dernière valeur IDENTITY générée dans le scope (portée) de ta procédure.  
Enorme différence lors d'accès concurrents 😊.

Répondre avec citation | 0 | 0

16/07/2006, 11h11

#15

Rodie

En attente de confirmation mail



Inscrit en: mai 2006  
Messages: 171  
Points: 122



Ah merci bien 😊  
Les différences SCOPE\_IDENTITY, IDENT\_CURRENT et @@IDENTITY sont ici :  
<http://msdn2.microsoft.com/fr-fr/library/ms190315.aspx>

Répondre avec citation | 0 | 0

30/10/2006, 19h04

#16

dymezac

Membre régulier



Inscrit en: janvier 2006  
Messages: 224  
Points: 99



bonjour,  
  
je sais que ce post date un peu mais je me demandais pour quelle raison faire une transaction pour une simple requête SELECT ?  
  
(en fait je suis en train d'apprendre les transactions)  
  
d'avance merci pour vos éclaircissements

Répondre avec citation | 0 | 0

+ Répondre à la discussion

Cette discussion est résolue.

Actualités FAQs TUTORIELS SOURCES LIVRES OUTILS AZURE DEV  
.NET .NET .NET .NET .NET CAMP

Forum Dotnet Accès aux données [C#] Comment faire un simple SELECT à SQL Server ?

#### Développeur .net et web H/F

↳ ORSYS - Ile de France - Puteaux (92800)

#### Chef de Projet .Net H/F

↳ MICHAEL PAGE - Lorraine - Metz (57000)

#### Développeur DOT.NET C# Luxembourg

↳ EKXEL IT SERVICES & FINANCIAL ENGINEERING - Luxembourg [Voir plus d'offres](#)

« Discussion précédente | Discussion suivante »

Navigation

**Inscrivez-vous gratuitement**  
pour pouvoir participer, suivre les réponses en temps réel, voter pour les messages, poser vos propres questions et recevoir la newsletter

**JE M'INSCRIS**

Quels langages en 2019 selon l'IEEE  
hon leader pour la troisième année  
écutive, il s'impose dans tous les  
aines dans lesquels il est utilisé, du  
loppement web à l'embarqué

Quels sont vos langages de  
programmation préférés pour le  
développement web en 2019 ? Et  
pourquoi ?

**l'état de l'écosystème des  
loppeurs en 2019 : l'évolution du  
le du code, d'après une enquête  
êe par JetBrains**

**Publication de la page des meilleurs  
cours et tutoriels pour apprendre Visual  
Studio, par l'équipe de rédaction**

Navigation

**Inscrivez-vous  
gratuitement**  
pour pouvoir  
participer, suivre les  
réponses en temps  
réel, voter pour les  
messages, poser vos  
propres questions et  
recevoir la newsletter

**JE M'INSCRIS**

Discussions similaires

[c#][SQL Server 2005] Faire un simple select  
Par skystef dans le forum Accès aux données

Réponses: 1  
Dernier message: 19/12/2006, 07h38

comment faire un insert, + select + update dans la meme requete  
Par jam92400 dans le forum Requêtes et SQL.

Réponses: 6  
Dernier message: 27/06/2006, 19h33

Comment faire plus simple  
Par siaoly dans le forum Langage

Réponses: 5  
Dernier message: 11/06/2006, 23h56

Comment faire une simple boucle de décrementation  
Par marcus333 dans le forum Langage

Réponses: 5  
Dernier message: 05/02/2006, 14h24

Comment faire plus simple  
Par mobscene dans le forum Langage

Réponses: 5  
Dernier message: 26/07/2005, 05h12

Partager

Nous contacter

Developpez.com

Haut de page