This page is specific to

Microsoft Visual Studio 2010/.NET Framework 4

.NET Framework Class Library

**SqlTransaction Class**

Represents a Transact-SQL transaction to be made in a SQL Server database. This class cannot be inherited.

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifInheritance Hierarchy

[System.Object](http://msdn.microsoft.com/en-us/library/system.object.aspx)  
[System.MarshalByRefObject](http://msdn.microsoft.com/en-us/library/system.marshalbyrefobject.aspx)  
[System.Data.Common.DbTransaction](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.aspx)  
**System.Data.SqlClient.SqlTransaction**

**Namespace:** [System.Data.SqlClient](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.aspx)  
**Assembly:** System.Data (in System.Data.dll)

Description: http://i.msdn.microsoft.com/Global/Images/clear.gif

Syntax

Visual Basic

Public NotInheritable Class SqlTransaction \_

Inherits DbTransaction

C#

public sealed class SqlTransaction : DbTransaction

Visual C++

public ref class SqlTransaction sealed : public DbTransaction

F#

[<Sealed>]

type SqlTransaction =

class

inherit DbTransaction

end

The **SqlTransaction** type exposes the following members.

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifProperties

|  |  |  |
| --- | --- | --- |
|  | **Name** | **Description** |
| Description: Public property | [Connection](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqltransaction.connection.aspx) | Gets the [SqlConnection](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqlconnection.aspx) object associated with the transaction, or **nullNothingnullptr**a null reference (**Nothing** in Visual Basic) if the transaction is no longer valid. |
| Description: Protected property | [DbConnection](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.dbconnection.aspx) | Specifies the [DbConnection](http://msdn.microsoft.com/en-us/library/system.data.common.dbconnection.aspx) object associated with the transaction. (Inherited from [DbTransaction](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.aspx).) |
| Description: Public property | [IsolationLevel](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqltransaction.isolationlevel.aspx) | Specifies the [IsolationLevel](http://msdn.microsoft.com/en-us/library/system.data.isolationlevel.aspx) for this transaction. (Overrides [DbTransaction..::.IsolationLevel](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.isolationlevel.aspx).) |

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifMethods

|  |  |  |
| --- | --- | --- |
|  | **Name** | **Description** |
| Description: Public method | [Commit](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqltransaction.commit.aspx) | Commits the database transaction. (Overrides [DbTransaction..::.Commit()()()](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.commit.aspx).) |
| Description: Public method | [CreateObjRef](http://msdn.microsoft.com/en-us/library/system.marshalbyrefobject.createobjref.aspx) | Creates an object that contains all the relevant information required to generate a proxy used to communicate with a remote object. (Inherited from [MarshalByRefObject](http://msdn.microsoft.com/en-us/library/system.marshalbyrefobject.aspx).) |
| Description: Public method | [Dispose()()()](http://msdn.microsoft.com/en-us/library/bf2cw321.aspx) | Releases the unmanaged resources used by the [DbTransaction](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.aspx). (Inherited from [DbTransaction](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.aspx).) |
| Description: Protected method | [Dispose(Boolean)](http://msdn.microsoft.com/en-us/library/ms224592.aspx) | Releases the unmanaged resources used by the [DbTransaction](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.aspx) and optionally releases the managed resources. (Inherited from [DbTransaction](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.aspx).) |
| Description: Public method | [Equals(Object)](http://msdn.microsoft.com/en-us/library/bsc2ak47.aspx) | Determines whether the specified [Object](http://msdn.microsoft.com/en-us/library/system.object.aspx) is equal to the current [Object](http://msdn.microsoft.com/en-us/library/system.object.aspx). (Inherited from [Object](http://msdn.microsoft.com/en-us/library/system.object.aspx).) |
| Description: Protected method | [Finalize](http://msdn.microsoft.com/en-us/library/system.object.finalize.aspx) | Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from [Object](http://msdn.microsoft.com/en-us/library/system.object.aspx).) |
| Description: Public method | [GetHashCode](http://msdn.microsoft.com/en-us/library/system.object.gethashcode.aspx) | Serves as a hash function for a particular type. (Inherited from [Object](http://msdn.microsoft.com/en-us/library/system.object.aspx).) |
| Description: Public method | [GetLifetimeService](http://msdn.microsoft.com/en-us/library/system.marshalbyrefobject.getlifetimeservice.aspx) | Retrieves the current lifetime service object that controls the lifetime policy for this instance. (Inherited from [MarshalByRefObject](http://msdn.microsoft.com/en-us/library/system.marshalbyrefobject.aspx).) |
| Description: Public method | [GetType](http://msdn.microsoft.com/en-us/library/system.object.gettype.aspx) | Gets the [Type](http://msdn.microsoft.com/en-us/library/system.type.aspx) of the current instance. (Inherited from [Object](http://msdn.microsoft.com/en-us/library/system.object.aspx).) |
| Description: Public method | [InitializeLifetimeService](http://msdn.microsoft.com/en-us/library/system.marshalbyrefobject.initializelifetimeservice.aspx) | Obtains a lifetime service object to control the lifetime policy for this instance. (Inherited from [MarshalByRefObject](http://msdn.microsoft.com/en-us/library/system.marshalbyrefobject.aspx).) |
| Description: Protected method | [MemberwiseClone()()()](http://msdn.microsoft.com/en-us/library/system.object.memberwiseclone.aspx) | Creates a shallow copy of the current [Object](http://msdn.microsoft.com/en-us/library/system.object.aspx). (Inherited from [Object](http://msdn.microsoft.com/en-us/library/system.object.aspx).) |
| Description: Protected method | [MemberwiseClone(Boolean)](http://msdn.microsoft.com/en-us/library/ms131262.aspx) | Creates a shallow copy of the current [MarshalByRefObject](http://msdn.microsoft.com/en-us/library/system.marshalbyrefobject.aspx) object. (Inherited from [MarshalByRefObject](http://msdn.microsoft.com/en-us/library/system.marshalbyrefobject.aspx).) |
| Description: Public method | [Rollback()()()](http://msdn.microsoft.com/en-us/library/zayx5s0h.aspx) | Rolls back a transaction from a pending state. (Overrides [DbTransaction..::.Rollback()()()](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.rollback.aspx).) |
| Description: Public method | [Rollback(String)](http://msdn.microsoft.com/en-us/library/4ws6y4dy.aspx) | Rolls back a transaction from a pending state, and specifies the transaction or savepoint name. |
| Description: Public method | [Save](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqltransaction.save.aspx) | Creates a savepoint in the transaction that can be used to roll back a part of the transaction, and specifies the savepoint name. |
| Description: Public method | [ToString](http://msdn.microsoft.com/en-us/library/system.object.tostring.aspx) | Returns a string that represents the current object. (Inherited from [Object](http://msdn.microsoft.com/en-us/library/system.object.aspx).) |

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifExplicit Interface Implementations

|  |  |  |
| --- | --- | --- |
|  | **Name** | **Description** |
| Description: Explicit interface implemetationDescription: Private property | [IDbTransaction..::.Connection](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqltransaction.aspx#Data#IDbTransaction#Connection) | Gets the [DbConnection](http://msdn.microsoft.com/en-us/library/system.data.common.dbconnection.aspx) object associated with the transaction, or a null reference if the transaction is no longer valid. (Inherited from [DbTransaction](http://msdn.microsoft.com/en-us/library/system.data.common.dbtransaction.aspx).) |

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifRemarks

The application creates a **SqlTransaction** object by calling [BeginTransaction](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqlconnection.begintransaction.aspx) on the [SqlConnection](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqlconnection.aspx) object. All subsequent operations associated with the transaction (for example, committing or aborting the transaction), are performed on the **SqlTransaction** object.

|  |
| --- |
| **Description: NoteNote** |
| **Try**/**Catch** exception handling should always be used when committing or rolling back a **SqlTransaction**. Both [Commit](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqltransaction.commit.aspx) and [Rollback](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqltransaction.rollback.aspx) generate an [InvalidOperationException](http://msdn.microsoft.com/en-us/library/system.invalidoperationexception.aspx) if the connection is terminated or if the transaction has already been rolled back on the server. |

For more information on SQL Server transactions, see "Explicit Transactions" and "Coding Efficient Transactions" in SQL Server 2005 Books Online.

Description: http://i.msdn.microsoft.com/Global/Images/clear.gifExamples

The following example creates a [SqlConnection](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqlconnection.aspx) and a **SqlTransaction**. It also demonstrates how to use the [BeginTransaction](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqlconnection.begintransaction.aspx), [Commit](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqltransaction.commit.aspx), and [Rollback](http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqltransaction.rollback.aspx) methods. The transaction is rolled back on any error. **Try**/**Catch** error handling is used to handle any errors when attempting to commit or roll back the transaction.

Visual Basic

Private Sub ExecuteSqlTransaction(ByVal connectionString As String)

Using connection As New SqlConnection(connectionString)

connection.Open()

Dim command As SqlCommand = connection.CreateCommand()

Dim transaction As SqlTransaction

' Start a local transaction

transaction = connection.BeginTransaction("SampleTransaction")

' Must assign both transaction object and connection

' to Command object for a pending local transaction.

command.Connection = connection

command.Transaction = transaction

Try

command.CommandText = \_

"Insert into Region (RegionID, RegionDescription) VALUES (100, 'Description')"

command.ExecuteNonQuery()

command.CommandText = \_

"Insert into Region (RegionID, RegionDescription) VALUES (101, 'Description')"

command.ExecuteNonQuery()

' Attempt to commit the transaction.

transaction.Commit()

Console.WriteLine("Both records are written to database.")

Catch ex As Exception

Console.WriteLine("Commit Exception Type: {0}", ex.GetType())

Console.WriteLine(" Message: {0}", ex.Message)

' Attempt to roll back the transaction.

Try

transaction.Rollback()

Catch ex2 As Exception

' This catch block will handle any errors that may have occurred

' on the server that would cause the rollback to fail, such as

' a closed connection.

Console.WriteLine("Rollback Exception Type: {0}", ex2.GetType())

Console.WriteLine(" Message: {0}", ex2.Message)

End Try

End Try

End Using

End Sub

C#

private static void ExecuteSqlTransaction(string connectionString)

{

using (SqlConnection connection = new SqlConnection(connectionString))

{

connection.Open();

SqlCommand command = connection.CreateCommand();

SqlTransaction transaction;

// Start a local transaction.

transaction = connection.BeginTransaction("SampleTransaction");

// Must assign both transaction object and connection

// to Command object for a pending local transaction

command.Connection = connection;

command.Transaction = transaction;

try

{

command.CommandText =

"Insert into Region (RegionID, RegionDescription) VALUES (100, 'Description')";

command.ExecuteNonQuery();

command.CommandText =

"Insert into Region (RegionID, RegionDescription) VALUES (101, 'Description')";

command.ExecuteNonQuery();

// Attempt to commit the transaction.

transaction.Commit();

Console.WriteLine("Both records are written to database.");

}

catch (Exception ex)

{

Console.WriteLine("Commit Exception Type: {0}", ex.GetType());

Console.WriteLine(" Message: {0}", ex.Message);

// Attempt to roll back the transaction.

try

{

transaction.Rollback();

}

catch (Exception ex2)

{

// This catch block will handle any errors that may have occurred

// on the server that would cause the rollback to fail, such as

// a closed connection.

Console.WriteLine("Rollback Exception Type: {0}", ex2.GetType());

Console.WriteLine(" Message: {0}", ex2.Message);

}

}

}

}