

برخی مواقع شما نیاز دارید تا یک Query را بر روی یک سرور اجرا نمایید و این Query برخی اطلاعات خود را از سرور دیگری دریافت می‌نماید. در این صورت باید یک پل ارتباطی بین سرور جاری و سرور دیگر وجود داشته باشد تا بتوانید در یک Query به سرور دیگری متصل شوید و اطلاعاتی را دریافت نمایید. در حالت عادی یک Query فقط می‌تواند بر روی سرور جاری اجرا شده و اطلاعاتی را بازیابی نماید. اما اگر همین Query بخواهد به سرور دیگری متصل شود، آن سرور باید در سرور جاری بصورت Linked Server تعریف شده باشد.

به عنوان مثال:

من سروری با آدرس 192.168.0.1 دارم که دارای پایگاه داده‌ای با نام Salary می‌باشد. نام این سرور را A می‌گذارم.

همچنین من سرور دیگری با آدرس 192.168.1.100 دارم که دارای پایگاه داده‌ای با نام Accounting است. نام این سرور را B می‌گذارم.

حالا می‌خواهم در سرور A یک Query بنویسم که جدول Payment را با اتصال به سرور B به جدول Document متصل نموده و نتیجه ی JOIN این دو جدول را نمایش دهد. به عنوان مثال:

```
SELECT * FROM Payment AS pay JOIN Document AS doc
ON pay.DocumentId = doc.Id
```

این Query به هیچ عنوان اجرا نخواهد شد. زیرا نمی‌تواند جدول Document را پیدا کند. برای این منظور باید سرور B را به سرور A معرفی کنیم که این کار از طریق Linked Server انجام خواهد شد.

نحوه‌ی ایجاد یک Linked Server

بر روی سیستم من دو نسخه از SQL نصب شده است. یکی Standard Edition و دیگری Express Edition. من می‌خواهم در نسخه Standard یک Linked Server به نسخه‌ی Express ایجاد کنم. بنابراین با اتصال به نسخه Standard مراحل زیر را طی می‌کنم:

1. یک New query ایجاد می‌کنم.

2. دستورات زیر را در Query ایجاد شده می‌نویسم:

```
sp_addlinkedserver 'MyServer', '', 'SQLNCLI', '.\sqlexpress'
```

توضیحات:

sp_addlinkedserver نام رویه‌ای است که یک Linked Server را ایجاد می‌نماید.

پارامتر اول نام Linked Server را مشخص می‌نماید که جهت دسترسی به سرور دیگر مورد استفاده قرار می‌گیرد.

پارامتر دوم Product Name می‌باشد که من خالی گذاشتم.

پارامتر سوم Provider Name یا نام فراهم کننده داده‌ای است. چون من می‌خواهم به یک سرور SQL متصل شوم (SQLNCLI (Native Client) را انتخاب کردم. اگر به منبع داده‌ای دیگری مثل MySQL، Oracle، Access و ... متصل می‌شوید باید Provider

Name دیگری را انتخاب کنید.

پارامتر چهارم نام یا IP سروری است که می‌خواهیم به آن لینک شویم.

3. با فشردن F5 یا منوی Execute این Query را اجرا کنید.

با اجرای موفقیت آمیز مراحل فوق باید عنوان MyServer را در مسیر Server Objects > Linked Server مشاهده کنید. در نسخه Express پایگاه داده‌ای با نام test دارم که شامل جدولی به نام tbl می‌باشد. با نوشتن Query زیر می‌توانم محتویات این جدول را مشاهده کنم:

```
SELECT * FROM MyServer.test.dbo.tbl
```

ممکن است جهت اتصال به سرور لینک شده نیاز به نام کاربری و رمز عبور داشته باشید. جهت تعریف نام کاربری و رمز عبور برای سرور لینک شده از دستورات زیر استفاده کنید:

```
sp_addlinkedsevrlogin 'MyServer',@rmtuser='user1', @rmtpassword='abc123'
```

توضیحات:

sp_addlinkedsevrlogin نام رویه ای است که نام کاربری و رمز عبور را به یک Linked Server اضافه می‌کند.

پارامتر اول نام Linked Server می‌باشد.

پارامتر دوم نام کاربری جهت اتصال به سرور لینک شده می‌باشد.

پارامتر سوم رمز عبور جهت اتصال به سرور لینک شده می‌باشد.

نظرات خوانندگان

نویسنده: بابک جهانگیری
تاریخ: ۱۶:۵۳ ۱۳۹۳/۰۹/۱۱

آیا هنگام ارسال رمز عبور از طریق کانکشن استرینگ برای اتصال به یک سرور اس کیو ال ، چه به صورت لینک سرور و چه به صورت اتصال از طریق یک آپلیکیشن ویندوزی یا وبی داده‌های رمز عبور به صورت کد شده و ایمن ارسال می‌شوند ؟

نویسنده: محمد رجبی
تاریخ: ۹:۱ ۱۳۹۳/۰۹/۱۲

با سلام و احترام؛ ضمن سپاس از نوشتن این مطلب، نکات تکمیلی زیر برای خوانندگان ارائه می‌گردد:
پس از ایجاد Linked Server به طریقی که در مطلب فوق بدان اشاره گردید، به دو صورت می‌توان به اجرای دستورات در Server دیگر پرداخت. به شکل **Distributed Query** و یا استفاده از تابع **OPENQUERY()**.
برای مثال دو SQL Instance به نام‌های SQL1 و SQL2 موجود است که قصد داریم عملیاتی را از SQL2 با استفاده از امکان Distributed Query و OpenQuery در SQL1 انجام دهیم، برای مثال:

Distributed Query:

```
SELECT * FROM [SQL2].[test].[dbo].[emp]
```

OPENQUERY:

```
SELECT * from OPENQUERY([SQL2], 'SELECT * FROM [test].[dbo].[emp]')
```

در استفاده از OpenQuery باید جانب احتیاط را رعایت نمود، چنانچه قصد اجرای دستورات ساده ای را در Remote Server دارید، توصیه می‌شود از Distributed Query استفاده شود و اگر دستورات پیچیده و محاسباتی هستند استفاده از OPENQUERY هزینه کمتری دارد. همچنین در نظر داشته باشید چنانچه در دستورات مربوط به جدولی ستونی با دیتا تایپ XML وجود داشت قادر به اجرای دستورات به هر دو شکل نخواهید بود، در واقع کوئری‌های توزیع شده این مورد را پشتیبانی نمی‌کنند. برای مثال پس از اجرای دستورات زیر با پیغام خطا زیر مواجه می‌شوید: «Xml data type is not supported in distributed queries»

```
SELECT * FROM [نام Linked Server].[AdventureWorks].[Person].[Person]
GO
SELECT * FROM OPENQUERY([نام Linked Server ],'SELECT * FROM [AdventureWorks].[Person].[Person]')
```

همچنین در نظر داشته باشید چنانچه در دستورات مربوط به جدولی ستونی با دیتا تایپ CLR وجود داشت قادر به اجرای دستورات به شکل OpenQuery خواهید بود ولی در صورت استفاده از Distributed Query با خطایی مواجه می‌شوید با این پیغام «Objects exposing columns with CLR types are not allowed in distributed queries»

```
SELECT * FROM [نام Linked Server].[AdventureWorks].[Person].[Address]
GO
SELECT * FROM OPENQUERY([نام Linked Server ],'SELECT * FROM [AdventureWorks].[Person].[Address]')
```

برای کسب اطلاعات بیشتر به لینک زیر مراجعه فرمائید:

[Distributed query \(Four-part\) or OPENQUERY when executing linked server queries in SQL Server](#)

نویسنده: محمد طاهری
تاریخ: ۱۰:۳۸ ۱۳۹۳/۰۹/۱۵

سلام آقای جهانگیری

شما برای این کار یکسری تنظیمات را باید در Enterprise manager انجام بدهید

1- نصب Certificate در ویندوز خودتان (از طریق MMC)

2- فعال کردن Certificate در قسمت Enterprise Manager

3- پارامتر Encrypt را هم در قسمت Connection String خود لحاظ کنید.

اگر سئوالی بود بنده در خدمتم.