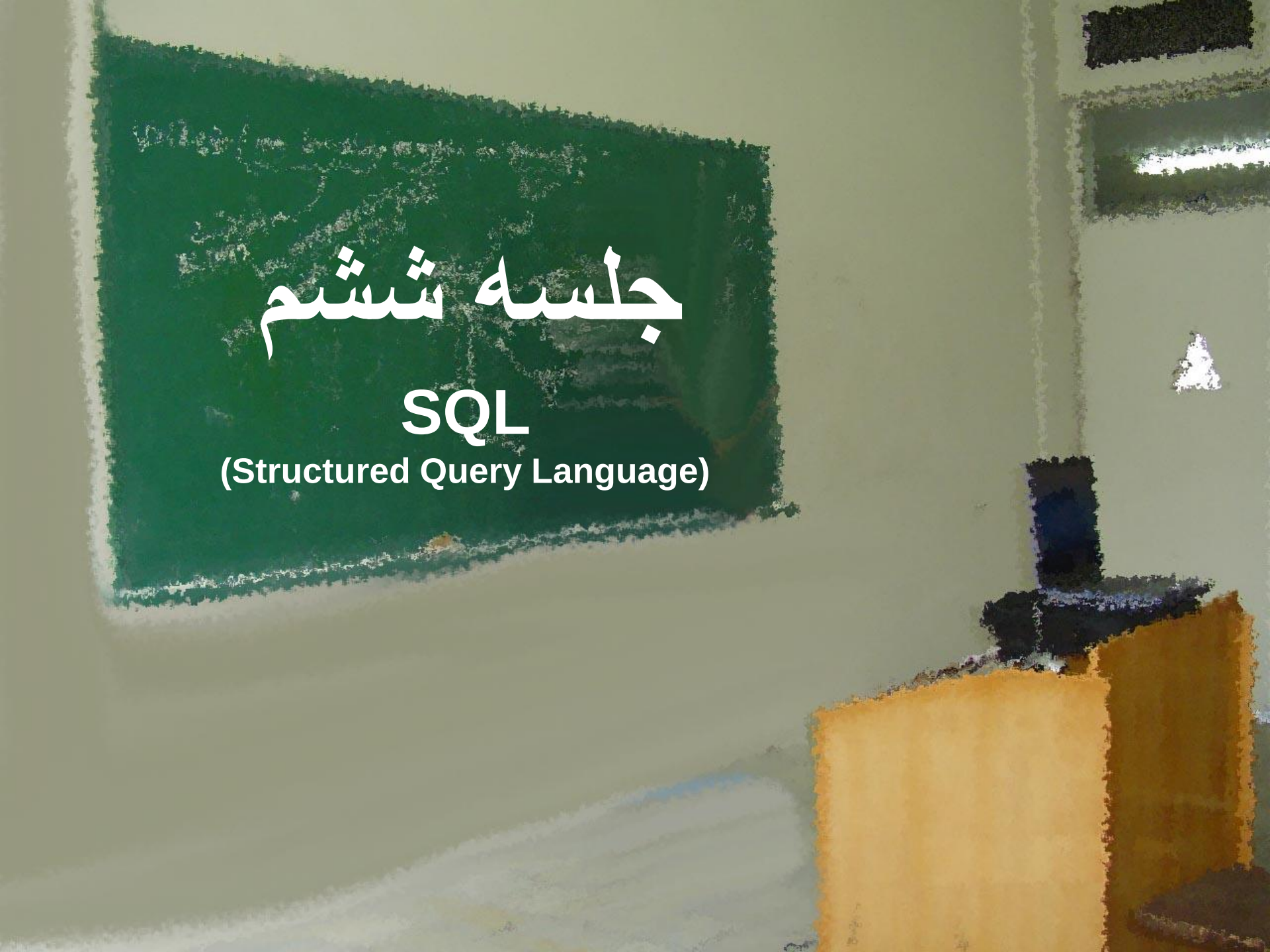
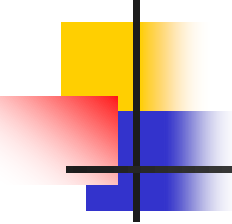




جلسه ششم

SQL

(Structured Query Language)



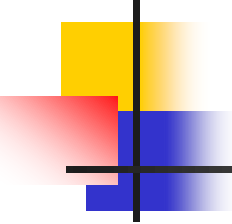
آنچه در این جلسه می خوانید:

(۱) زبان SQL

- انواع داده ای در SQL

- انواع دستورات SQL

۱. دستورات مربوط به اسکیمای (DDL)
۲. دستورات کار با داده ها (DML)
۳. دستورات کنترلی (DCL)



آنچه در این جلسه می خوانید:

۲) تست دستورات SQL در Query Analyzer

انواع داده اي در SQL

انواع عددي

- **tinyint** : عدد صحيح ۱ بايتي مثبت (۰ تا ۲۵۵)
- **smallint** : عدد صحيح ۲ بايتي (۳۲۷۶۷ تا ۳۲۷۶۸-)
- **int** : عدد صحيح ۴ بايتي (۲۱۴۷۴۸۳۶۴۷ تا ۲۱۴۷۴۸۳۶۴۸-)
- **bigint** : عدد صحيح ۸ بايتي
- **decimal** : اعداد حقيقي از $10^{38} - 1$ تا $10^{38} - 1$
- **real** : اعداد اعشار ۴ بايتي با دقت ۷ رقم
- **float** : اعداد اعشار ۸ بايتي با دقت ۱۵ رقم

انواع کاراکتری

- **char** : رشته تا حداکثر ۸۰۰۰ کاراکتر : ذخیره با طول ثابت
- **varchar** : رشته تا حداکثر ۸۰۰۰ کاراکتر : ذخیره با طول متغیر
- **nchar** : رشته تا حداکثر ۴۰۰۰ کاراکتر: ذخیره با طول ثابت و بصورت Unicode
- **nvarchar** : رشته تا حداکثر ۴۰۰۰ کاراکتر: ذخیره با طول متغیر و بصورت Unicode

انواع دیگر

- **bit** : مقادیر منطقی ۰ و ۱
- **text** : متن شامل حداکثر ۲۱۴۷۴۸۳۶۴۷ کاراکتر
- **ntext** : متن شامل حداکثر ۱۰۷۳۷۴۱۸۲۳ کاراکتر: ذخیره Unicode
- **image** : برای ذخیره هر نوع داده با حجم زیاد تا ۲۱۴۷۴۸۳۶۴۷ بایت.
- مثلاً برای ذخیره یک فایل یا یک عکس در DB

دستورات کار با اسکیمایما

(DDL)

تعريف جدول جديد

```
CREATE TABLE "table_name"  
( "column 1" "data_type",  
  "column 2" "data_type",  
  ...  
)
```

Example1:

```
CREATE TABLE Students  
( Id int primary key,  
  Name Nvarchar(15) not null,  
  SurName Nvarchar(20) not null,  
  Field Nvarchar(10)  
)
```

Example2:

```
CREATE TABLE TelNums  
(Id int Identity,  
  TelNum Nvarchar(15),  
  Type Nvarchar(5),  
  SId int,  
  Primary Key (Id),  
  Foreign Key (SId) references Students(Id)  
)
```

از بین بردن یک جدول

DROP TABLE "table_name"

Example:

Drop TABLE Students

تغییر در اسکیمای یک جدول

Alter TABLE "table_name"
[Drop| Add | Alter] column ...

Examples:

افزودن یک فیلد جدید

- **Alter TABLE** Students **Add** BirthDate **nvarchar**(20)

تغییر در تعریف یک فیلد

- **Alter TABLE** Students **Alter column** name **nvarchar**(6) null

حذف یک فیلد

- **Alter TABLE** Students **Drop column** Field

دستورات کار با داده ها (DML)

۱- دستور Select

بازیابی داده ها از جداول

ساختار یک دستور ساده Select

جدول FROM اسامي فيلدها SELECT
WHERE شرط
projection
selection

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

مثال: لیست شماره و نام تمام دانشجویان.

Select Id,Name from S

مثال: لیست مشخصات تمام دانشجویان رشته های کامپیوتر و شیمی.

Select * from S where Field='Computer' or Field= 'Chemistry'

مثال: لیست اسامي دانشجویان کامپیوتر که شماره آنها کمتر از ۸۴۱۱۰ است.

Select Name from S where Field='Computer' and Id<=84110

مقادیر ممکن بجای اسامی فیلدها:

(۱) یک عدد ثابت

Select Name, 5 from S

Name	Unknown Column
Ali	5
Reza	5
Hassan	5

(۲) یک فرمول

Select Name, Id/10 as NewID from S

Name	NewID
Ali	8411
Reza	8412
Hassan	8413

نام گذاری یک
ستون در خروجی

(۳) ترکیب چند فیلد رشته ای

مثال: لیست نام و نام خ دانشجویان.

Select Name + ' ' + Family as CompleteName from S

CompleteName
Ali Ahmadi
Reza Rezaei
Hassan Hassani

بررسی خالی بودن یک فیلد

جدول **FROM** اسامی فیلدها **SELECT**
WHERE فیلد **IS NULL**

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Null
84130	Hassan	Hasani	Chemistry

مثال: لیست شماره و رشته دانشجویانی که رشته آنها وارد شده است.

Select Id,Field from S where not Field Is Null

~~= Null~~

مثال: لیست شماره دانشجویانی که نه رشته و نه نام خ آنها وارد نشده است.

Select Id from S where Family Is Null and Field is Null

عملگر Order By : مرتب کردن نتایج

جدول FROM اسامی فیلدها SELECT

شرط WHERE

نام فیلدها [desc | asc] ORDER BY

صعودي نزولي

رابطه S

Id	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Rezaei	Chemistry

مثال: لیست شماره و نام خ تمام دانشجویان، مرتب شده بر اساس شماره بطور صعودي.

Select Id, family from S order by Id

مثال: لیست مشخصات تمام دانشجویان مرتب بر اساس نام خ و سپس نام.

Select * from S order by family, name

مثال: لیست اسامی دانشجویان کامپیوتر بترتیب نزولي شماره آنها

Select Name from S where Field='Computer' order by Id desc

عملگر Between

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

- براي مقادير عددي و تاريخ است.
- خود مقادير ابتدا و انتهاي بازه را هم در نظر مي گيرد.

مثال: ليست نام خ تمام دانشجويان كه شماره آنها بين ۸۴۱۱۰ و ۸۴۱۲۰ است.

```
Select Id,Name from S where Id>=84110 and Id<=84120
```

يا

```
Select Id,Name from S where Id between 84110 and 84120
```

عملگر Like

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

- براي مقادير رشته اي است.
- براي يافتن مقاديري كه نه الزاما دقيقا يكسان بلكه مشابه كلمه مورد نظر باشند.
- منظور از مشابه بودن اين است كه كلمه مورد نظر جزئي از آن باشد.

مثال: ليست دانشجوياني كه نام آنها با A شروع مي شود.

Select * from S where name Like 'A%'

مثال: ليست دانشجوياني كه نام خ آنها با i تمام مي شود.

Select * from S where family Like '%i'

مثال: ليست دانشجوياني كه در نام آنها حرف A يا a وجود دارد.

Select * from S where name Like '%A%' or name like '%a%'

نکته: اگر علامت % را فراموش كنيم، عملگر Like معادل = عمل مي كند.

چند مثال دیگر:

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

مثال: لیست مشخصات تمام دانشجویان رشته کامپیوتر که نام آنها ۳ حرفی و حرف میانی آن | است.

Select * from S where Field='Computer' and name Like '_|_'

مثال: لیست مشخصات تمام دانشجویان که نام آنها با A شروع و به i ختم می شود .

Select * from S where name Like 'A%i'

مثال: لیست اسامی دانشجویانی که حرف دوم و چهارم نام آنها a و s است.

Select name from S where name like '_a_s%'

بازیابی رکوردهای تکراری

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Computer

- گاهی ممکن است خروجی یک دستور Select شامل رکوردهای تکراری باشد.
- اگر نخواهیم رکورد تکراری بازیابی شود باید از عملگر **DISTINCT** استفاده کنیم.
- عملگر **ALL** برعکس DISTINCT عمل می کند.
- اگر هیچکدام نوشته نشود، بطور پیش فرض ALL در نظر گرفته می شود.

اسامی فیلدها [DISTINCT | ALL]
SELECT
FROM جدول

مثال: لیستی از تمام رشته های دانشگاه.

Select Distinct Field from S

مثال: لیست نام و نام خ تمام دانشجویان.

Select all Name , Family from S

Select Name , Family from S

یا

عملگر UNION و UNION ALL

- اجتماع رکوردهای دو جدول را برمی گرداند.

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

رابطه T

<u>TId</u>	TName	TFamily
1	Hadi	Ahmadi
2	Karim	Hassani
3	Ali	Omidi

مثال: لیستی از نام خ تمام اساتید و دانشجویان.

Select Family from S UNION Select Family From T

Select Family from S UNION ALL Select Family From T

دستور اول دقیقاً اجتماع است یعنی رکورد های تکراری یکبار در خروجی می آیند. اما در دومی تکرار می شوند.

مثال: لیست مشخصات تمام دانشجویان رشته های کامپیوتر و شیمی.

Select * from S where Field='Computer' UNION

Select * from S where Field= 'Chemistry'

عملگر IN

- عضویت یک مقدار را در یک مجموعه بررسی می کند.

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

رابطه T

<u>TId</u>	TName	TFamily
1	Hadi	Ahmadi
2	Karim	Hassani
3	Ali	Omidi

رابطه STC (جدول نمرات)

<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

حالت اول) مجموعه ثابت و مشخص است.

مثال: لیست مشخصات تمام دانشجویان رشته های کامپیوتر، ریاضی و شیمی.

Select * from S where Field IN ('Computer' , 'Math' , 'Chemistry')

عملگر IN

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

رابطه T

<u>TId</u>	TName	TFamily
1	Hadi	Ahmadi
2	Karim	Hassani
3	Ali	Omidi

رابطه STC (جدول نمرات)

<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

حالت دوم مجموعه ثابت و مشخص نیست. (Nested Query)

مثال: لیست مشخصات تمام دانشجویان که تاکنون درس گرفته اند.

(باید Id آنها عضو مجموعه SId های موجود در STC باشد)

SELECT * from S where Id IN (SELECT SId from STC)

مجموعه SId های موجود در STC

~~SELECT * from S where Id IN (SELECT * from STC)~~

Nested Query یک روش پاسخ به پرس و جوهایی است که به بیش از یک جدول نیاز دارند

یک مثال دیگر

رابطه S

<u>Id</u>	Name	Family	Field
84110	Ali	Ahmadi	Computer
84120	Reza	Rezaei	Math
84130	Hassan	Hasani	Chemistry

رابطه C

<u>CId</u>	CName	Units
01	DB	3
02	OS	3

رابطه STC (جدول نمرات)

<u>SId</u>	<u>TId</u>	<u>CId</u>	mark
84110	2	01	12
84110	1	03	8
84130	2	02	18

مثال: لیست مشخصات دانشجویانی که تاکنون درس ۳ واحدی نگرفته اند.
(باید Id آنها عضو مجموعه SId های موجود در STC که درس ۳ واحدی گرفتند نباشد)

```
SELECT * from S where Id NOT IN
  (SELECT SId from STC Where CID IN
    (SELECT CId from C Where Units=3)
  )
```

شماره دانشجویانی
که تاکنون درس
۳ واحدی گرفته
اند

تست دستورات SQL در Query Analyzer



Internet

Internet Explorer



E-mail

Outlook Express



RealPlayer



Microsoft Office Pow
2003



Adobe Reader 8



jetAudio



Microsoft Office Wor



Paint

All Programs

start



EzAudio



Games



Intel Network Adapters



jetAudio



LiveUpdate Notice



Microsoft Office



Microsoft SQL Server



Microsoft SQL Server - Switch



Microsoft Visual Studio .NET 2003



Norton AntiVirus



Real



Realtek Sound Manager



Startup



Adobe Reader 8



Internet Explorer



MSN



Outlook Express



RealPlayer



Remote Assistance



Windows Media Player



Windows Messenger



Books Online



Client Network Utility



Configure SQL XML Support in IIS



Enterprise Manager



Import and Export Data



Profiler



Query Analyzer

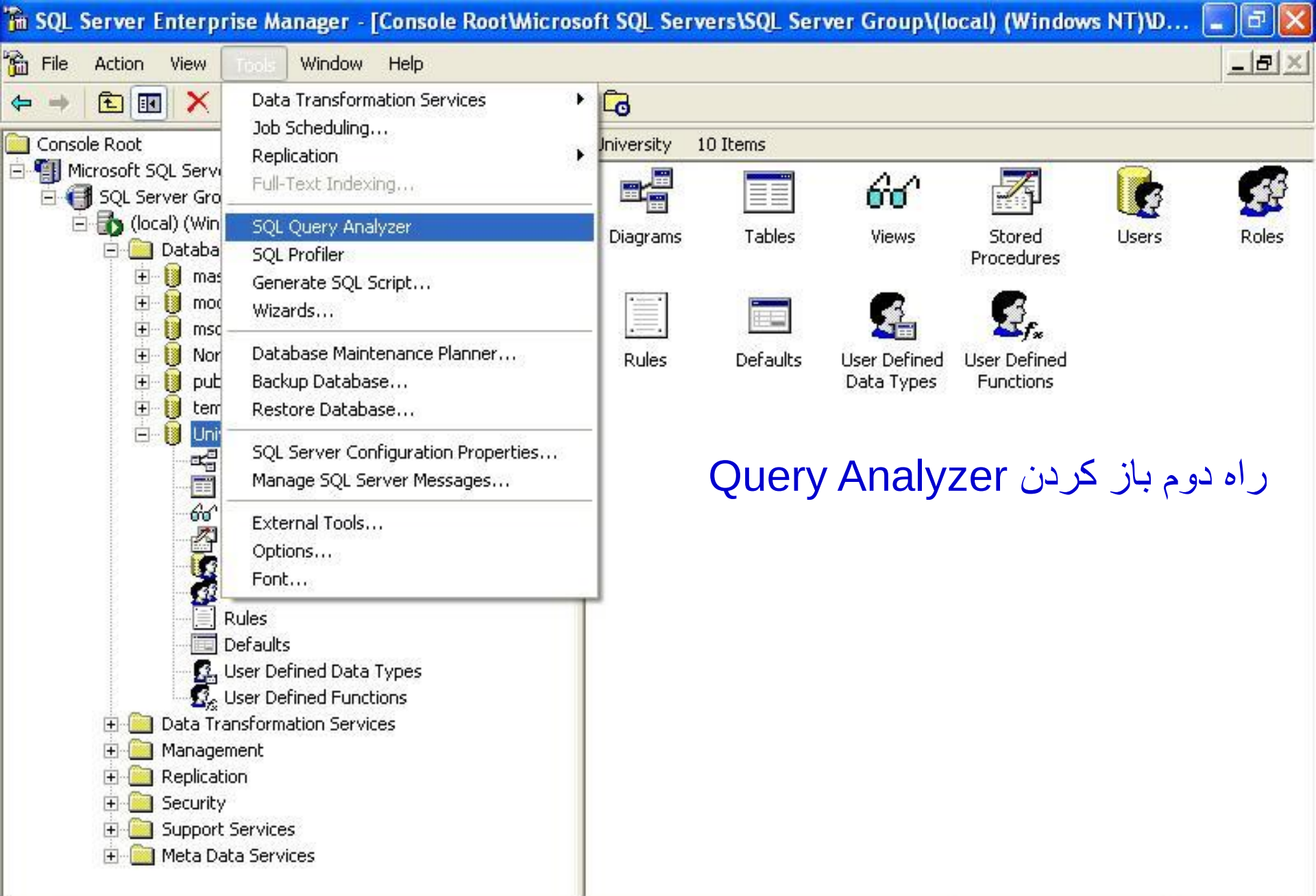


Server Network Utility

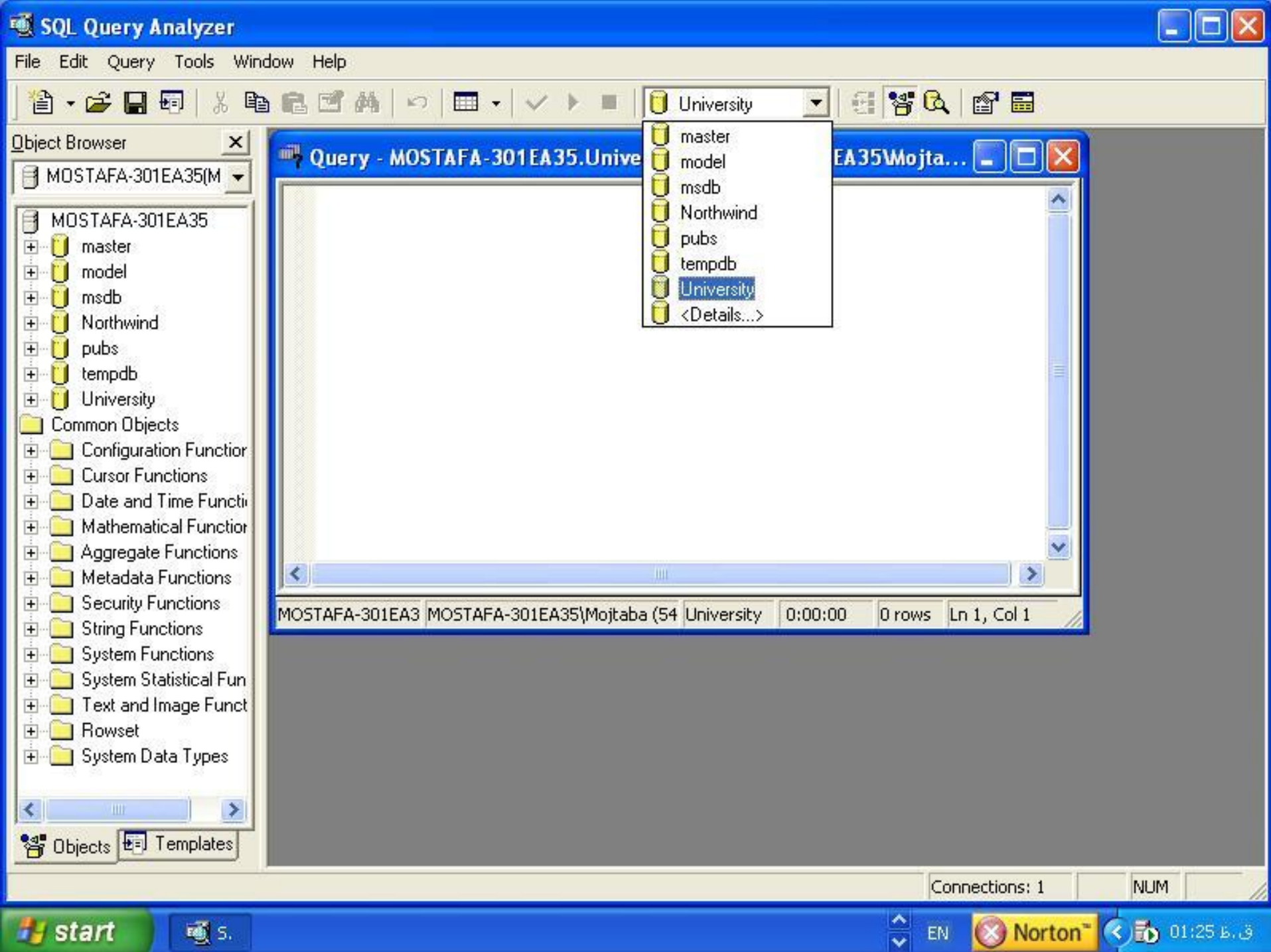


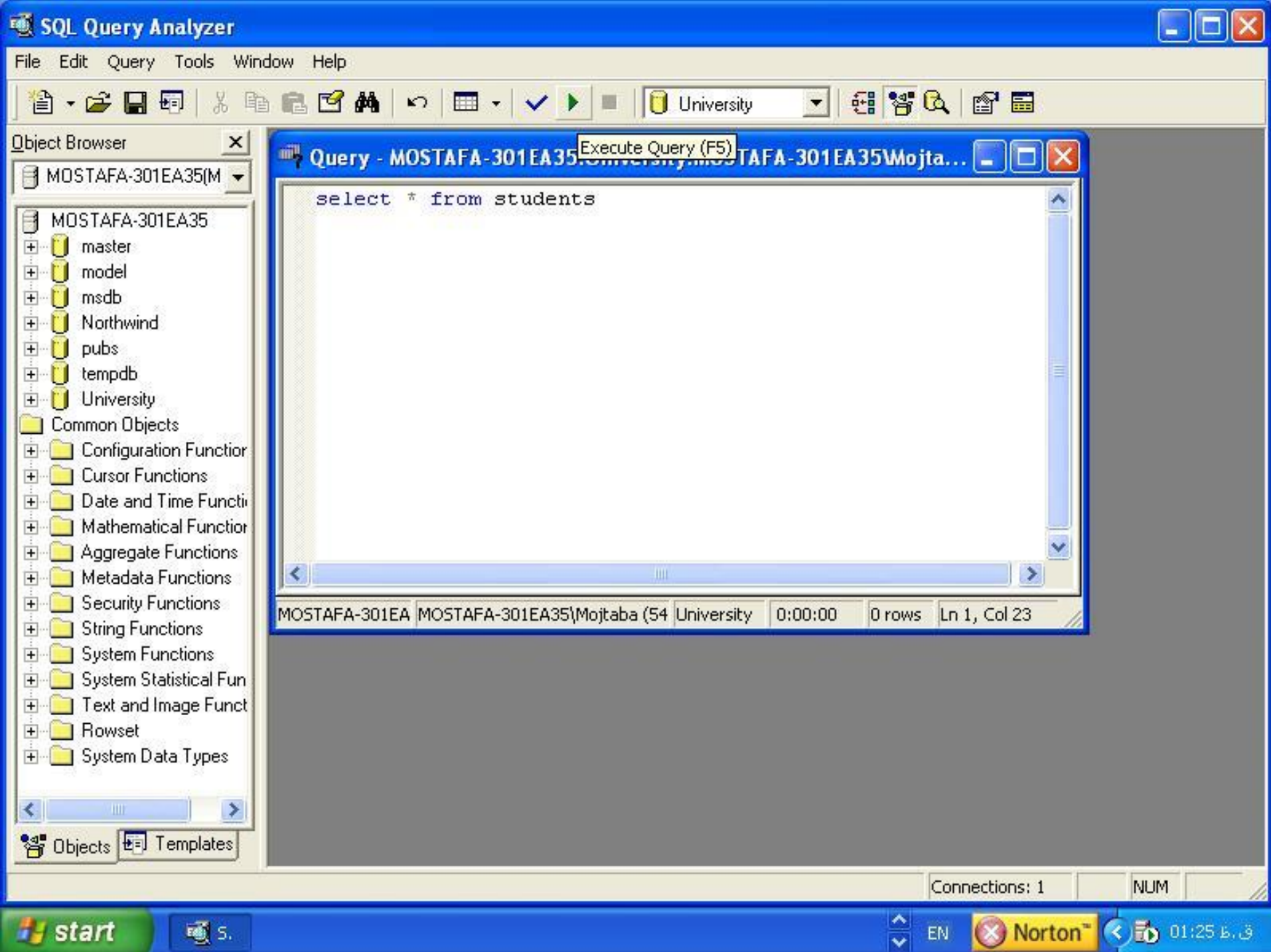
Service Manager

EN



راه دوم باز کردن Query Analyzer





SQL Query Analyzer

File Edit Query Tools Window Help

University

Object Browser

- MOSTAFA-301EA35(M)
- MOSTAFA-301EA35
 - master
 - model
 - msdb
 - Northwind
 - pubs
 - tempdb
 - University
- Common Objects
 - Configuration Function
 - Cursor Functions
 - Date and Time Function
 - Mathematical Function
 - Aggregate Functions
 - Metadata Functions
 - Security Functions
 - String Functions
 - System Functions
 - System Statistical Function
 - Text and Image Function
 - Rowset
 - System Data Types

Query - MOSTAFA-301EA35.University.MOSTAFA-301EA35Mojta...

```
select * from students
```

	Id	Name	family
1	84110	Ali	Ahmadi
2	84120	Reza	Rezaei
3	84130	Hassan	hassani
4	84140	mahdi	Mahdavi
5	84150	Amir	Amiri

Grids Messages

MOSTAFA-301EA MOSTAFA-301EA35\Mojtaba (54 University 0:00:00 5 rows Ln 1, Col 23

Connections: 1 NUM

SQL Query Analyzer

File Edit Query Tools Window Help

University

Object Browser

- MOSTAFA-301EA35(M)
- MOSTAFA-301EA35
 - master
 - model
 - msdb
 - Northwind
 - pubs
 - tempdb
 - University
- Common Objects
 - Configuration Function
 - Cursor Functions
 - Date and Time Function
 - Mathematical Function
 - Aggregate Functions
 - Metadata Functions
 - Security Functions
 - String Functions
 - System Functions
 - System Statistical Function
 - Text and Image Function
 - Rowset
 - System Data Types

Query - MOSTAFA-301EA35.University.MOSTAFA-301EA35Mojta...

```
select * from students where name like 'A%'
and Id between 84110 and 84170
```

	Id	Name	family
1	84110	Ali	Ahmadi
2	84150	Amir	Amiri

Grids Messages

MOSTAFA-301EA3 MOSTAFA-301EA35\Mojtaba (54 University 0:00:00 2 rows Ln 2, Col 2

Connections: 1 NUM