

Domas Budrys A5 -CSCI 5020 Screenshots

Ch14Ex1:

```
ch14ex2.sql - DO...omo-PC\domo (53))* ch14ex1.sql - DO...omo-PC\domo (56)) X
```

```
USE MyGuitarShop;

DECLARE @TotalProducts int ;
SET @TotalProducts =
    (SELECT COUNT(*) FROM Products);

IF @TotalProducts >=7
    PRINT 'The number of products is greater or equal to 7';
ELSE
    PRINT 'The number of products is less than 7';
```

100 %

Messages

The number of products is greater or equal to 7

Ch14Ex2:

```
ch14ex2.sql - DO...omo-PC\domo (53)) X ch14ex1.sql - DO...omo-PC\domo (56))
```

```
DECLARE @AvgListPrice money;

SET @TotalProducts =
    (SELECT COUNT(*) FROM Products);

SET @AvgListPrice =
    (SELECT AVG(ListPrice) FROM Products);

IF @TotalProducts >=7
    PRINT 'Total number of products =' + CONVERT(varchar, @TotalProducts)
    + ' and Average list price =' + CONVERT(varchar, @AvgListPrice);
ELSE
    PRINT 'The number of products is less than 7';
```

100 %

Messages

Total number of products =10 and Average list price =841.90

100 %

Ch15Ex2:

The screenshot displays two windows from SQL Server Enterprise Manager. The top window shows the SQL code for creating a scalar function named `fnDiscountPrice`. The function takes an integer parameter `@InputItemID` and returns a money value. The logic inside the function is to select the `ItemPrice` minus the `DiscountAmount` from the `OrderItems` table where the `ItemID` matches the input parameter.

```
CREATE FUNCTION fnDiscountPrice
(
    @InputItemID int
)
RETURNS money
AS
BEGIN
    RETURN (SELECT ItemPrice - DiscountAmount
            FROM OrderItems
            WHERE ItemID = @InputItemID)
END;
```

The bottom window shows the execution of the function. The query `SELECT dbo.fnDiscountPrice(2)` is entered in the query editor. The `Results` pane at the bottom shows a single row with the value `303.79`.

SQL Query Editor: ch15ex2.sql - DO...omo-PC\domo (57))*

```
SELECT dbo.fnDiscountPrice(2)
```

Results

	(No column name)
1	303.79

Ch15Ex4:

ch15ex4.sql - DO...omo-PC\domo (51))* × SQLQuery11.sql -...omo-PC\domo (61))*

```
CREATE PROC spInsertProducts
    @InputCategoryID int,
    @InputProductCode varchar(255),
    @InputProductName varchar(255),
    @InputListPrice money,
    @InputDiscountPercent money
AS
    IF (@InputListPrice < 0)
        PRINT ('ListPrice cannot be negative');
    ELSE IF (@InputDiscountPercent < 0) |
        PRINT ('DiscountPrice cannot be negative');
    ELSE
        INSERT INTO Products (CategoryID, ProductCode, ProductName,
                               Description, ListPrice, DiscountPercent, DateAdded)
        VALUES (@InputCategoryID, @InputProductCode, @InputProductName,
                '', @InputListPrice, @InputDiscountPercent, GETDATE());
```

100 %

Messages

Command(s) completed successfully.

100 %

✓ Query e... | DOMO-PC\SQL2012 (11.0 SP2) | domo-PC\domo (51) | MyGuitarShop | 00:00:00 | 0 rows

SQLQuery10.sql -...omo-PC\domo (60)) SQLQuery9.sql - D...omo-PC\domo (59))* X

```

USE MyGuitarShop;

EXEC spInsertProducts 2, 'CSCI', 'ComputerScience', 220.10, -12;
EXEC spInsertProducts 4, 'MATH', 'Mathematics', -4, 222;
EXEC spInsertProducts 4, 'BIO', 'Biology', 994, 222;

```

100 %

Messages

DiscountPrice cannot be negative
ListPrice cannot be negative

(1 row(s) affected)

```

SELECT * FROM Products

```

100 %

Results Messages

	ProductID	CategoryID	ProductCode	ProductName	Description
5	5	1	washburn	Washburn D10S	The Washburn D10S acoustic guitar
6	6	1	rodriguez	Rodriguez Caballero 11	Featuring a carefully chosen, solid Ca
7	7	2	precision	Fender Precision	The Fender Precision bass guitar deli
8	8	2	hofner	Hofner Icon	With authentic details inspired by the
9	9	3	ludwig	Ludwig 5-piece Drum...	This product includes a Ludwig 5-pie
10	10	3	tama	Tama 5-Piece Drum ...	The Tama 5-piece Drum Set is the mc
11	13	4	BIO	Biology	

Ch15Ex6:

```
CREATE TRIGGER Products_UPDATE
ON Products
AFTER UPDATE
AS
BEGIN
    DECLARE @UpdatedID int
    DECLARE @InputAmount money
    DECLARE @PercentAmount money

    SET @InputAmount = (SELECT DiscountPercent FROM inserted)
    SET @UpdatedID = (SELECT ProductID FROM inserted)
    SET @PercentAmount = @InputAmount * 100

    SET NOCOUNT ON

    IF UPDATE(DiscountPercent)
        IF (@InputAmount < 0 )
            BEGIN
                ROLLBACK TRANSACTION;
            END
END
```

100 %

Messages

Command(s) completed successfully.

100 %

Query executed successfully. | DOMO-PC\SQL2012 (11.0 SP2) | domo-PC\domo (59) | MyGuitarShop | 00:00:00 | 0 rows

```
UPDATE Products
SET DiscountPercent = 110
WHERE ProductID = 13
```

00 %

Messages

Msg 50000, Level 16, State 1, Procedure Porducts_UPDATE, Line 27
DiscountAmount cannot be more than a 100
Msg 3609, Level 16, State 1, Line 1
The transaction ended in the trigger. The batch has been aborted.

```
UPDATE Products
SET DiscountPercent = -5
WHERE ProductID = 13
```

100 %

Messages

Msg 50000, Level 16, State 1, Procedure Porducts_UPDATE, Line 22
DiscountAmount cannot be negative
Msg 3609, Level 16, State 1, Line 1
The transaction ended in the trigger. The batch has been aborted.

```

UPDATE Products
SET DiscountPercent = 0.4
WHERE ProductID = 13

```

100 %

Messages

(1 row(s) affected)

```
SELECT * FROM Products
```

100 %

Results Messages

	ProductID	CategoryID	ProductCode	ProductName	Description	ListPrice	DiscountP
1	1	1	strat	Fender Stratocaster	The Fender Stratocaster is the electric guitar desig...	699.00	30.00
2	2	1	les_paul	Gibson Les Paul	This Les Paul guitar offers a carved top and humbu...	1199.00	30.00
3	3	1	sg	Gibson SG	This Gibson SG electric guitar takes the best of the...	2517.00	52.00
4	4	1	fg700s	Yamaha FG700S	The Yamaha FG700S solid top acoustic guitar has ...	489.99	38.00
5	5	1	washburn	Washburn D10S	The Washburn D10S acoustic guitar is superbly cr...	299.00	0.00
6	6	1	rodriguez	Rodriguez Caballero 11	Featuring a carefully chosen, solid Canadian cedar ...	415.00	39.00
7	7	2	precision	Fender Precision	The Fender Precision bass guitar delivers the soun...	799.99	30.00
8	8	2	hofner	Hofner Icon	With authentic details inspired by the original, the H...	499.99	25.00
9	9	3	ludwig	Ludwig 5-piece Drum...	This product includes a Ludwig 5-piece drum set a...	699.99	30.00
10	10	3	tama	Tama 5-Piece Drum ...	The Tama 5-piece Drum Set is the most affordable ...	799.99	15.00
11	13	4	BIO	Hello World		994.00	40.00

```

UPDATE Products
SET DiscountPercent = 9
WHERE ProductID = 13

```

100 %

Messages

(1 row(s) affected)

```
SELECT * FROM Products
```

100 %

Results Messages

	ProductID	CategoryID	ProductCode	ProductName	Description	ListPrice	DiscountI
1	1	1	strat	Fender Stratocaster	The Fender Stratocaster is the electric guitar desig...	699.00	30.00
2	2	1	les_paul	Gibson Les Paul	This Les Paul guitar offers a carved top and humbu...	1199.00	30.00
3	3	1	sg	Gibson SG	This Gibson SG electric guitar takes the best of the...	2517.00	52.00
4	4	1	fg700s	Yamaha FG700S	The Yamaha FG700S solid top acoustic guitar has ...	489.99	38.00
5	5	1	washburn	Washburn D10S	The Washburn D10S acoustic guitar is superbly cr...	299.00	0.00
6	6	1	rodriguez	Rodriguez Caballero 11	Featuring a carefully chosen, solid Canadian cedar ...	415.00	39.00
7	7	2	precision	Fender Precision	The Fender Precision bass guitar delivers the soun...	799.99	30.00
8	8	2	hofner	Hofner Icon	With authentic details inspired by the original, the H...	499.99	25.00
9	9	3	ludwig	Ludwig 5-piece Drum ...	This product includes a Ludwig 5-piece drum set a...	699.99	30.00
10	10	3	tama	Tama 5-Piece Drum S...	The Tama 5-piece Drum Set is the most affordable ...	799.99	15.00
11	13	4	BIO	Hello World		994.00	9.00

Ch16Ex1:

```
DECLARE ProductNameCursor CURSOR FOR
SELECT ProductID, ProductName, ListPrice
FROM Products
WHERE ListPrice > 700
ORDER BY ListPrice DESC

OPEN ProductNameCursor

FETCH NEXT FROM ProductNameCursor INTO @InputProductID, @InputProductName, @InputListPrice

WHILE (@@FETCH_STATUS = 0)
BEGIN
    PRINT(@InputProductName + ', $' + CONVERT(varchar, @InputListPrice))
    FETCH NEXT FROM ProductNameCursor INTO @InputProductID, @InputProductName, @InputListPrice
END
```

100 %

Messages

Gibson SG, \$2517.00
Gibson Les Paul, \$1199.00
Fender Precision, \$799.99
Tama 5-Piece Drum Set with Cymbals, \$799.99

Ch16Ex2:

```
ch16ex2.sql - DO...omo-PC\domo (52)  ch16ex1.sql - DO...omo-PC\domo (55)
```

```
DECLARE CustomerOrdersCursor CURSOR
FOR
SELECT LastName, AVG(ShipAmount) AS ShipAmountAvg
FROM Customers JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
GROUP BY LastName

OPEN CustomerOrdersCursor
FETCH NEXT FROM CustomerOrdersCursor

WHILE (@@FETCH_STATUS = 0)
BEGIN
    FETCH NEXT FROM CustomerOrdersCursor
END
```

100 %

Results Messages

	LastName	ShipAmountAvg
1	Albares	5.00
1	Amigon	10.00
1	Brown	10.00
1	Butt	5.00
	LastName	ShipAmountAvg

Query executed successfully. | DOMO-PC\SQL2012 (11.0 SP2) | domo-PC\domo (52) | MyGuitarShop | 00:00:00 | 35 rows

Ch17Ex1:

```
ch17ex1.sql - D:\...omo-PC\domo (56))  CreateMyGuitarSh...omo-PC\domo (59))  SQLQuery5.sql - D:\...omo-PC\domo (58))*  
BEGIN TRY  
    BEGIN TRAN  
  
        DELETE FROM Addresses  
        WHERE CustomerID = 8;  
  
        -- Must Execute this statement in order to work  
        -- Because there are CustomerID values in Orders table  
        -- DELETE FROM Orders  
        -- WHERE CustomerID = 8  
  
        DELETE FROM Customers  
        WHERE CustomerID = 8;  
  
    COMMIT TRAN;  
100 %  
Messages  
  
(2 row(s) affected)  
  
(0 row(s) affected)  
Transaction did not work. Cannot be deleted based on  
provided instructions in the question. Must uncomment  
above provided code to work.
```

Ch17Ex2:

```
DECLARE @OrderID int  
  
BEGIN TRAN  
BEGIN TRY  
  
    INSERT Orders  
    VALUES(3, GETDATE(), '10.00', '0.00', NULL, 4,  
            'American Express', '378282246310005', 04/2013, 4);  
  
    SET @OrderID = @@IDENTITY;  
  
    INSERT OrderItems  
    VALUES(@OrderID, 6, '415.00', '161.85', 1);  
  
    INSERT OrderItems  
    VALUES(@OrderID, 1, '699.00', '209.70', 1)  
%  
Messages  
  
(1 row(s) affected)  
  
(1 row(s) affected)  
  
(1 row(s) affected)
```


Ch18Ex1:

```
Ch18Ex1.sql - D:\...01m0-PC(demo (39)) x Ch18Ex2.sql - D:\...01m0-PC(demo (00))
USE MyGuitarShop;

CREATE ROLE OrderEntry;

GRANT INSERT, UPDATE
ON Orders
TO OrderEntry;

GRANT INSERT, UPDATE
ON OrderItems
TO OrderEntry;

GRANT SELECT
```

100 %

Messages

Command(s) completed successfully.

Ch18Ex2:

```
USE MyGuitarShop;

CREATE LOGIN RobertHalliday
WITH PASSWORD = 'HelloBob',
DEFAULT_DATABASE = MyGuitarShop;

CREATE USER RobertHalliday
FOR LOGIN RobertHalliday;

ALTER ROLE OrderEntry
ADD MEMBER RobertHalliday;
```

100 %

Messages

Command(s) completed successfully.

Ch18Ex3:

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the Object Explorer shows the database structure, including tables (dbo.Addresses, dbo.Administrators, etc.), users (AdminUser, guest, INFORMATION_SCHEMA, etc.), and roles (sys, Roles). The main window shows a T-SQL query in the SQL Query window, which is a loop that creates logins, users, and adds them to a role for each row in the AdminCursor.

```
SQLQuery12.sql - ...omo-PC\domo (55))* SQLQuery11.sql - ...omo-PC\domo (62))* ch18ex2.sql - DO...omo-PC\domo (61))

FETCH NEXT FROM AdminCursor INTO @InputAdminFirst, @InputAdminLast, @tempPassword

WHILE (@@FETCH_STATUS = 0)
BEGIN
    SET @CreateLogin = 'CREATE LOGIN ' + @InputAdminFirst + @InputAdminLast
    + ' WITH PASSWORD = '' + @tempPassword +
    + '', DEFAULT_DATABASE = MyGuitarShop;'

    SET @CreateUser = 'CREATE USER ' + @InputAdminFirst + @InputAdminLast
    + ' FOR LOGIN ' + @InputAdminFirst + @InputAdminLast

    SET @AlterRole = 'ALTER ROLE OrderEntry ADD MEMBER '

```

The Messages pane shows the execution results for three iterations of the loop:

```

Create Login: CREATE LOGIN AdminUser WITH PASSWORD = '6a718fbd768c2378b511f8249b54897f940e9022', DEFAL
Create User: CREATE USER AdminUser FOR LOGIN AdminUser
Add Role: ALTER ROLE OrderEntry ADD MEMBER AdminUser
-----
Create Login: CREATE LOGIN JoelMurach WITH PASSWORD = '971e95957d3b74d70d79c20c94e9cd91b85f7aae', DEF
Create User: CREATE USER JoelMurach FOR LOGIN JoelMurach
Add Role: ALTER ROLE OrderEntry ADD MEMBER JoelMurach
-----
Create Login: CREATE LOGIN MikeMurach WITH PASSWORD = '3f2975c819cefc686282456aae3a137bf896ee8', DEF
Create User: CREATE USER MikeMurach FOR LOGIN MikeMurach
Add Role: ALTER ROLE OrderEntry ADD MEMBER MikeMurach
-----

```

Ch18Ex4:

Login - New

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Connection

Server: DOMO-PC\SQL2012

Connection: domo-PC\domo

[View connection properties](#)

Progress

Ready

Script Help

Login name: RBratigan Search...

☐ Windows authentication

☒ SQL Server authentication

Password:

Confirm password:

☐ Specify old password

Old password:

☒ Enforce password policy

☐ Enforce password expiration

☐ User must change password at next login

☐ Mapped to certificate

☐ Mapped to asymmetric key

☒ Map to Credential

Add

Mapped Credentials

Credential	Provider
------------	----------

Remove

Default database: MyGuitarShop

Default language: <default>

OK Cancel

Login - New

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Connection

Server: DOMO-PC\SQL2012

Connection: domo-PC\domo

[View connection properties](#)

Progress

Ready

Script Help

Users mapped to this login:

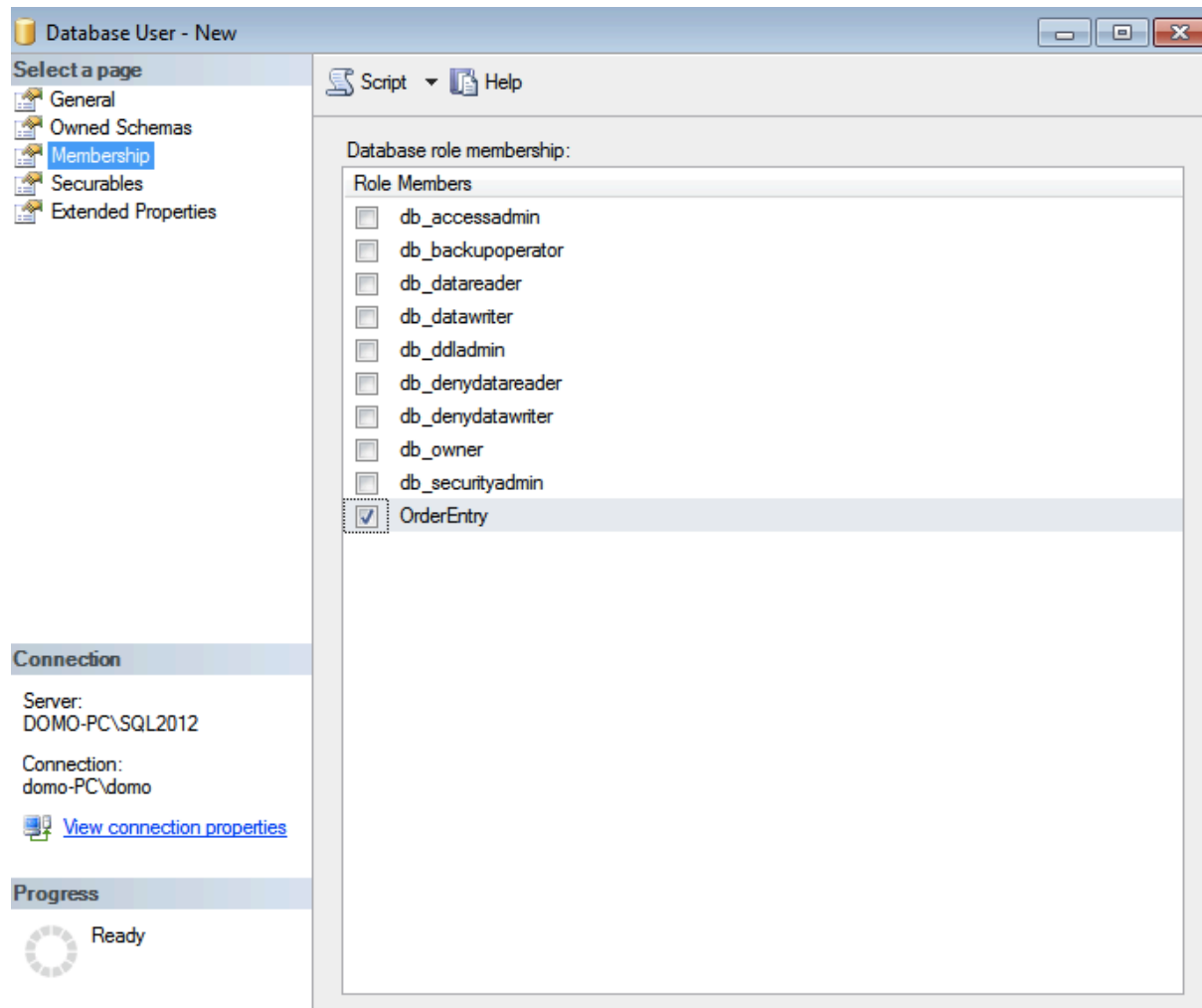
Map	Database	User	Default Schema
<input type="checkbox"/>	AdventureWorksDW20...		
<input type="checkbox"/>	master		
<input type="checkbox"/>	model		
<input type="checkbox"/>	msdb		
<input checked="" type="checkbox"/>	MyGuitarShop	RBratigan	
<input type="checkbox"/>	ReportServer\$SQL2012		
<input type="checkbox"/>	ReportServer\$SQL201...		
<input type="checkbox"/>	tempdb		

☐ Guest account enabled for: MyGuitarShop

Database role membership for: MyGuitarShop

<input type="checkbox"/>	db_accessadmin
<input type="checkbox"/>	db_backupoperator
<input type="checkbox"/>	db_datareader
<input type="checkbox"/>	db_datawriter
<input type="checkbox"/>	db_ddladmin
<input type="checkbox"/>	db_denydatareader
<input type="checkbox"/>	db_denydatawriter
<input type="checkbox"/>	db_owner
<input type="checkbox"/>	db_securityadmin
<input checked="" type="checkbox"/>	OrderEntry
<input checked="" type="checkbox"/>	public

OK Cancel



Ch19Ex1:

ch19ex1.sql - DO...omo-PC\domo (53)) X CustomerAddresses.xml XML_F52E2B61-18A...00805F49916B5.xml

```
USE MyGuitarShop;  
  
SELECT C.FirstName, C.LastName, A.Line1, A.Line2, A.City, A.State, A.ZipCode  
FROM Customers AS C  
JOIN Addresses AS A  
ON C.CustomerID = A.CustomerID  
FOR XML PATH ('Customer'), ROOT ('Customers')
```

100 %

Results Messages

XML_F52E2B61-18A1-11d1-B105-00805F49916B

1 <Customers><Customer><FirstName>Allan</First Name...

Query executed successfully. DOMO-PC\SQL2012 (11.0 SP2) | domo-PC\domo (53) | MyGuitarShop | 00:00:00 | 1 rows

```
<Customers>  
<Customer>  
  <FirstName>Allan</FirstName>  
  <LastName>Sherwood</LastName>  
  <Line1>100 East Ridgewood Ave.</Line1>  
  <Line2></Line2>  
  <City>Paramus</City>  
  <State>NJ</State>  
  <ZipCode>07652</ZipCode>  
</Customer>  
<Customer>  
  <FirstName>Allan</FirstName>  
  <LastName>Sherwood</LastName>  
  <Line1>21 Rosewood Rd.</Line1>  
  <Line2></Line2>  
  <City>Woodcliff Lake</City>  
  <State>NJ</State>  
  <ZipCode>07677</ZipCode>  
</Customer>  
<Customer>  
  <FirstName>Barry</FirstName>  
  <LastName>Zimmer</LastName>  
  <Line1>16285 Wendell St.</Line1>  
  <Line2></Line2>  
  <City>Omaha</City>  
  <State>NE</State>  
  <ZipCode>68135</ZipCode>  
</Customer>  
<Customer>  
  <FirstName>Christine</FirstName>
```

100 %

```
ch19ex1.sql - DO...omo-PC\domo (53)  CustomerAddresses.xsd  XML_F52E2B61-18A...00805F49916B8.xml
<xsd:schema targetNamespace="CustomerAddresses" xmlns:schema="CustomerAddresses" xmlns:xsd="http://w
<xsd:import namespace="http://schemas.microsoft.com/sqlserver/2004/sqltypes" schemaLocation="http:
<xsd:element name="C">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="schema:A" minOccurs="0" maxOccurs="unbounded" />
    </xsd:sequence>
    <xsd:attribute name="FirstName" use="required">
      <xsd:simpleType>
        <xsd:restriction base="sqltypes:varchar" sqltypes:localeId="1033" sqltypes:sqlCompareOptio
          <xsd:maxLength value="60" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="LastName" use="required">
      <xsd:simpleType>
        <xsd:restriction base="sqltypes:varchar" sqltypes:localeId="1033" sqltypes:sqlCompareOptio
          <xsd:maxLength value="60" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
  </xsd:complexType>
</xsd:element>
<xsd:element name="A">
  <xsd:complexType>
    <xsd:attribute name="Line1" use="required">
      <xsd:simpleType>
        <xsd:restriction base="sqltypes:varchar" sqltypes:localeId="1033" sqltypes:sqlCompareOptio
          <xsd:maxLength value="60" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
  </xsd:complexType>
</xsd:element>
```

Ch19Ex2:

```
NewCustomers.xml  ch19ex2.sql - DO...omo-PC\domo (52))*  ch19ex1.sql - DO...omo-PC\domo (53))
EXEC sp_xml_preparedocument @hdoc OUTPUT, @x

INSERT INTO Customers(EmailAddress, Password, FirstName, LastName)
SELECT *
FROM OPENXML(@hdoc, '/NewCustomers/Customer', 1)
WITH(
    EmailAddress varchar(255),
    Password varchar(60),
    FirstName varchar(60),
    LastName varchar(60)
)

EXEC sp_xml_removedocument @hdoc
```

100 %

Messages

(3 row(s) affected)

SQLQuery3.sql - D...omo-PC\domo (56))* NewCustomers.xml ch19ex2.sql - DO...omo-PC\domo (52))*

```
SELECT * FROM Customers
```

100 %

Results Messages

	CustomerID	EmailAddress	Password	FirstName	LastName	Shipping
481	481	nickolas_juvera@cox.net	77f602baf770e3618da53c0883d7c9bc82eca1f6	Nickolas	Juvera	507
482	482	gary_nunlee@nunlee.org	915e9617c8f0fb88e182a870a79129ae61c7efec	Gary	Nunlee	508
483	483	diane@cox.net	fba3f2a325a4e4eaf5992cbcc910ef91ecee3191	Diane	Devreese	510
484	484	roslyn.chavous@chavous....	3bf8ab2bdbc781f2000472c4fd9d86d6433ccd35	Roslyn	Chavous	511
485	485	glory@yahoo.com	31c689d737359ce214d095f5b0bcf6de82e57f3d	Glory	Schieler	512
486	493	izzychan@yahoo.com	NULL	Isabella	Chan	NULL
487	494	johnprine@gmail.com	NULL	John	Prine	NULL
488	495	kathykitchen@sboglobal.net	NULL	Kathy	Kitchen	NULL

Query executed successfully. | DOMO-PC\SQL2012 (11.0 SP2) | domo-PC\domo (56) | MyGuitarShop | 00:00:00 | 488 rows

Ch19Ex3:

SQLQuery4.sql - D:\omo-PC\domo (56))* x NewCustomers.xml ch19ex2.sql - DO...omo-PC\domo (52))

```
USE MyGuitarShop;

DECLARE @x xml

SELECT @x=C
FROM OPENROWSET (BULK 'C:\Users\domo\Desktop\Exercise starts\NewCustomers.xml', SINGLE_BLOB) as

DECLARE @hdoc int

EXEC sp_xml_preparedocument @hdoc OUTPUT, @x

SELECT *
FROM OPENXML (@hdoc, '/NewCustomers/Customers', 1)
```

100 %

Results Messages

	EmailAddress	Passwrod	FirstName	LastName
1	izzychan@yahoo.com	NULL	Isabella	Chan
2	johnprine@gmail.com	NULL	John	Prine
3	kathykitchen@sbcglobal.net	NULL	Kathy	Kitchen

Query executed successfully. | DOMO-PC\SQL2012 (11.0 SP2) | domo-PC\domo (56) | MyGuitarShop | 00:00:00 | 3 rows