



­

Assignment Document:

MERGE Sql Server

Version: < Sql Server 2008>/ASSIGNMENT/xxxx/x.x

Date: 03-08-2011

Cognizant

500 Glen Pointe Center West

Teaneck, NJ 07666

Ph: 201-801-0233  
[www.cognizant.com](http://www.cognizant.com)

SET NOCOUNT ON;

* Create Target Table
* Create Source Table
* Update and Insert into Customers

CREATE TABLE Customers (LastName VARCHAR(50), FirstName VARCHAR(50));

INSERT INTO Customers VALUES ('Doe', 'Jane');

select \* from Customers

CREATE TABLE NewCustomers(LastName VARCHAR(50), FirstName VARCHAR(50));

INSERT INTO NewCustomers VALUES ('Doe', 'John');

INSERT INTO NewCustomers VALUES ('Smith', 'Doris');

select \* from NewCustomers

MERGE Customers AS C

USING NewCustomers AS NC

ON C.LastName = NC. LastName

WHEN MATCHED THEN

UPDATE SET C.FirstName = NC.FirstName

WHEN NOT MATCHED THEN

INSERT (LastName, FirstName) VALUES (NC.LastName,NC.FirstName);

Example 2:

CREATE TABLE Students (LastName VARCHAR(50),

FirstName VARCHAR(50),

Address VARCHAR(100),

Age INT);

INSERT INTO Students VALUES ('Fritz', 'David', '181 Kline Street', 14)

,('Reese', 'Paul' , '4429 South Union', 14)

,('Brown', 'Jake' , '5401 Washington Ave',14);

CREATE TABLE NewYearRoster(LastName VARCHAR(50),

FirstName VARCHAR(50),

Address VARCHAR(100),

Age INT);

INSERT INTO NewYearRoster

VALUES ('Fritz', 'David', '181 Kline Street', 15)

,('Reese', 'Paul', '1950 Grandview Place', 15)

,('Adam', 'Wilbur', '4231 W. 93rd', 15);

select \* from NewYearRoster

select \* from Students

MERGE Students AS T

USING NewYearRoster AS S

ON S.LastName = T. LastName and

S.FirstName = T.FirstName

WHEN MATCHED THEN

UPDATE SET T.Address = S.Address,

T.Age = S.Age

WHEN NOT MATCHED THEN

INSERT (LastName,

FirstName,

Address,

Age)

VALUES (S.LastName,

S.FirstName,

S.Address,

S.Age)

WHEN NOT MATCHED BY SOURCE THEN

DELETE;

select \* from Students

Example 3:

MERGE Students AS T

USING NewYearRoster AS S

ON S.LastName = T. LastName and

S.FirstName = T.FirstName

WHEN MATCHED THEN

UPDATE SET T.Address = S.Address,

T.Age = S.Age

WHEN NOT MATCHED THEN

INSERT (LastName,

FirstName,

Address,

Age)

VALUES (S.LastName,

S.FirstName,

S.Address,

S.Age)

WHEN NOT MATCHED BY SOURCE THEN

DELETE

OUTPUT $action

, Inserted.LastName, Inserted.FirstName

, Deleted.LastName, Deleted.FirstName;

Example 4:

MERGE TOP (2) Students AS T

USING NewYearRoster AS S

ON S.LastName = T. LastName and

S.FirstName = T.FirstName

WHEN NOT MATCHED THEN

INSERT (LastName,

FirstName,

Address,

Age)

VALUES (S.LastName,

S.FirstName,

S.Address,

S.Age)

WHEN NOT MATCHED BY SOURCE THEN

DELETE

WHEN MATCHED THEN

UPDATE SET T.Address = S.Address,

T.Age = S.Age

OUTPUT $action

, Inserted.LastName, Inserted.FirstName

, Deleted.LastName, Deleted.FirstName;

Example 5:

DECLARE @Source TABLE

(

a INT,

b INT

)

INSERT @Source

SELECT 11, 21 UNION ALL

SELECT 12, 22

DECLARE @PrimaryTarget TABLE

(

a INT

)

DECLARE @SecondaryTarget TABLE

(

b INT

)

MERGE @PrimaryTarget AS pt

USING @Source AS s ON 1 = 1

WHEN NOT MATCHED BY TARGET

THEN INSERT (

a

)

VALUES (

s.a

)

OUTPUT s.b

INTO @SecondaryTarget;

SELECT 'Source' AS TableName,

a,

b

FROM @Source

SELECT 'PrimaryTarget' AS TableName,

a

FROM @PrimaryTarget

SELECT 'SecondaryTarget' AS TableName,

b

FROM @SecondaryTarget

Example 6:

CREATE TABLE Products ( ProductID INT PRIMARY KEY, ProductName VARCHAR(100), Rate MONEY )

GO

--Insert records into target table

INSERT INTO Products VALUES (1, 'Tea', 10.00), (2, 'Coffee', 20.00), (3, 'Muffin', 30.00), (4, 'Biscuit', 40.00)

GO

CREATE TABLE UpdatedProducts ( ProductID INT PRIMARY KEY, ProductName VARCHAR(100), Rate MONEY )

GO

--Insert records into source table

INSERT INTO UpdatedProducts VALUES (1, 'Tea', 10.00), (2, 'Coffee', 25.00), (3, 'Muffin', 35.00), (5, 'Pizza', 60.00)

GO

SELECT \* FROM Products

SELECT \* FROM UpdatedProducts

GO

MERGE Products AS TARGET USING UpdatedProducts

AS SOURCE ON (TARGET.ProductID = SOURCE.ProductID)

WHEN MATCHED AND

TARGET.ProductName <> SOURCE.ProductName

OR TARGET.Rate <> SOURCE.Rate THEN

UPDATE SET TARGET.ProductName = SOURCE.ProductName, TARGET.Rate = SOURCE.Rate

WHEN NOT MATCHED BY TARGET THEN

INSERT (ProductID, ProductName, Rate) VALUES (SOURCE.ProductID, SOURCE.ProductName, SOURCE.Rate)

WHEN NOT MATCHED BY SOURCE THEN DELETE

OUTPUT $action,

DELETED.ProductID AS TargetProductID,

DELETED.ProductName AS TargetProductName,

DELETED.Rate AS TargetRate,

INSERTED.ProductID AS SourceProductID, INSERTED.ProductName AS SourceProductName,

INSERTED.Rate AS SourceRate;

SELECT @@ROWCOUNT;