```
-- MERGE (4 points)

 /*
   Given two sets of tables as defined below, build a data pipeline
   using the SQL MERGE function and data refresh jobs to synchronize
   the data stored in the destination with the data stored in the source
   on a schedule. Keep an audit trail of what's changed in the destination
   set of tables using the SQL OUTPUT command. The two data sets exist
   in two different databases.

   Create a stored procedure containing the SQL code. The data pipeline
   will be based on the stored procedure.

   Regarding the UPDATE command, in the destination data set,
   the customer id, order id, order date, and item id will not
   change. INSERT and DELETE may affect all columns of the
   destination data set.

   Think about:
   1) How to prepare the data format at the source for data synch
   2) Where to put the audit table?
   3) In which database to create the stored procedure?
 */

 /*
 Steps:
 1) Create a source database and a destination database
 2) Create a stored procedure in your database containing MERGE and OUTPUT
 3) Use Azure Data Studio to establish the data pipeline
 4) Set up a SQL job in Azure Data Studio to synch data (required). For
    this job, you can just use one of the existing schedules.
 5) Set up a Jupyter Notebook job in Azure Data Studio to synch data.
    If you encounter an issue when setting up the job on your computer,
    you can skip this.
 */

 /*
 Submit:
 1) SQL code
 2) Screenshots of the configured data synch job(s)
 */

/*
Helpful Videos for this lab:

Install and Use Azure Data Studio
https://www.youtube.com/watch?v=YRK0hOssrfI&list=PL3gqFGmaw_0HgOyn_BuC3qZ87dalVF2fK&index
=7

How to Install SQL Server Agent Extension for Azure Data Studio
https://www.youtube.com/watch?v=cBFdoe2Zk6w&list=PL-zncNSJGgbjuUF-
OyzCR6eKs4o_Fq7zj&index=19
```

```
How to Create SQL Job in Azure Data Studio
https://www.youtube.com/watch?v=gYzG3XWskSY&list=PL-zncNSJGgbjuUF-OyzCR6eKs4o_Fq7zj&index=23

Jupyter Notebook and SQL
https://www.youtube.com/watch?v=6WO-zo6XR0k&list=PL-zncNSJGgbjuUF-OyzCR6eKs4o_Fq7zj&index=27

How to Create Jupyter Notebook Job in Azure Data Studio
https://www.youtube.com/watch?v=msUKon2FTtY&list=PL-zncNSJGgbjuUF-OyzCR6eKs4o_Fq7zj&index=22

*/

/* Source */

create table SaleOrder
(OrderID int identity primary key,
 OrderDate date,
 CustomerID int);

create table OrderItem
(ItemID int,
 OrderID int references SaleOrder(OrderID),
 Quantity int,
 UnitPrice money,
 LastModified datetime default getdate()
 primary key (OrderID, ItemID));


/* Destination */

create table ItemsReport
(CustomerID int,
 OrderDate date,
 OrderID int,
 ItemID int,
 Quantity int,
 UnitPrice money,
 LastModified datetime
 primary key(OrderID, ItemID));

CREATE TABLE ItemsAudit
 (
  Audit_PK INT  IDENTITY(1,1) NOT NULL
  ,CustomerID int
  ,OrderDate date
  ,OrderID int
  ,ItemID int
  ,OldQuantity int
  ,NewQuantity int
  ,OldUnitPrice money
  ,NewUnitPrice money
  ,NewLastModified datetime
  ,OldLastModified datetime
  ,[Action] CHAR(6) NULL
  ,ActionTime DATETIME DEFAULT GETDATE()
 );
```

```
--TRIGGER (4 points)

/*
   Given two sets of tables as defined below, build a data pipeline
   using the SQL triggers to synchronize the destination data with
   the source data in an event-driven and real-time manner.
   The two data sets exist in two different databases. Reuse the
   two databases created in Part 1.

   For simplicity, let's assume:
   1) All customers already exist in the destination table
   2) No new customer will be added to the destination table
   3) No existing customer will be deleted from the destination
      table

   Please keep in mind, in the source table, we have raw data
   (data without aggregation). In the destination table,
   we have aggregated data. TotalPurchase is the sum of
   OrderValue. NumberOfOrders is the count of orders.

   The audit table stores the order changes that caused
   TotalPurchase and NumberOfOrders to change. "Action" in
   the audit table indicates whether INSERT, UPDATE, or DELETE
   that caused the destination data to change.

   Think about:
   1) Where to put the audit table?
   2) How many trigger(s) de we need to create?
   3) In which database to create the trigger(s)?

   Submit the SQL code.
*/

/* Source */

CREATE TABLE CustomerOrder
(OrderID INT IDENTITY PRIMARY KEY,
 CustomerID INT NOT NULL,
 OrderDate DATETIME DEFAULT getdate(),
 OrderValue MONEY NOT NULL);


/* Destination */

CREATE TABLE CustomerReport
(CustomerID INT PRIMARY KEY,
 LastName VARCHAR(50),
 FirstName VARCHAR(50),
 Email VARCHAR(50),
 Phone VARCHAR(20),
 TotalPurchase MONEY,
 NumberOfOrders INT,
 ModifiedDate DATETIME DEFAULT getdate());
```

```
CREATE TABLE AuditCustomer
 (
  Audit_PK  INT  IDENTITY(1,1) PRIMARY KEY
  ,CustomerID  INT  NOT NULL
  ,OrderID INT
  ,OrderDate DATETIME
  ,OrderValue MONEY
  ,ModifiedDate DATETIME
  ,[Action] CHAR(6)
 );
```