QM-7093-01 ENTERPRISE DATA SYSTEMS

CASE STUDY (CS-8) – Noah L. Schrick - 1492657

Instructor: Dr. Ismail Abdulrashid,

Instructions:

You can find the Review Questions in the textbook at the end of Chapter 8: Questions A-G of the Morgan Importing Project

- Answer questions A through G of the Morgan Importing Project found on page 450.
- Create and set up the database with the given information.
- Do not include the result table unless specifically directed to.
- Include at least one line of white space between answers.
- Submit thru Harvey drop box
- Deliverable: You are expected to submit
    - A single SQL script file (.sql) prepared and saved in SQL Server Management Studio that includes your SQL statements that answer each of the questions in order.
    - This word file that you copied all of your SQL script (no result tables) from your SQL file.
- Each query should start with a comment line that looks like the following (last character corresponding to question number):
    - /* *** CS8-MI.A *** */
    - This line should follow the SQL statement that is your answer to the particular question (e.g. B, C, D, ...)
- You should include at least one line of white space between your answer SQL statements
- Do not include the result table unless it is said so!
- Check Harvey for the due date!

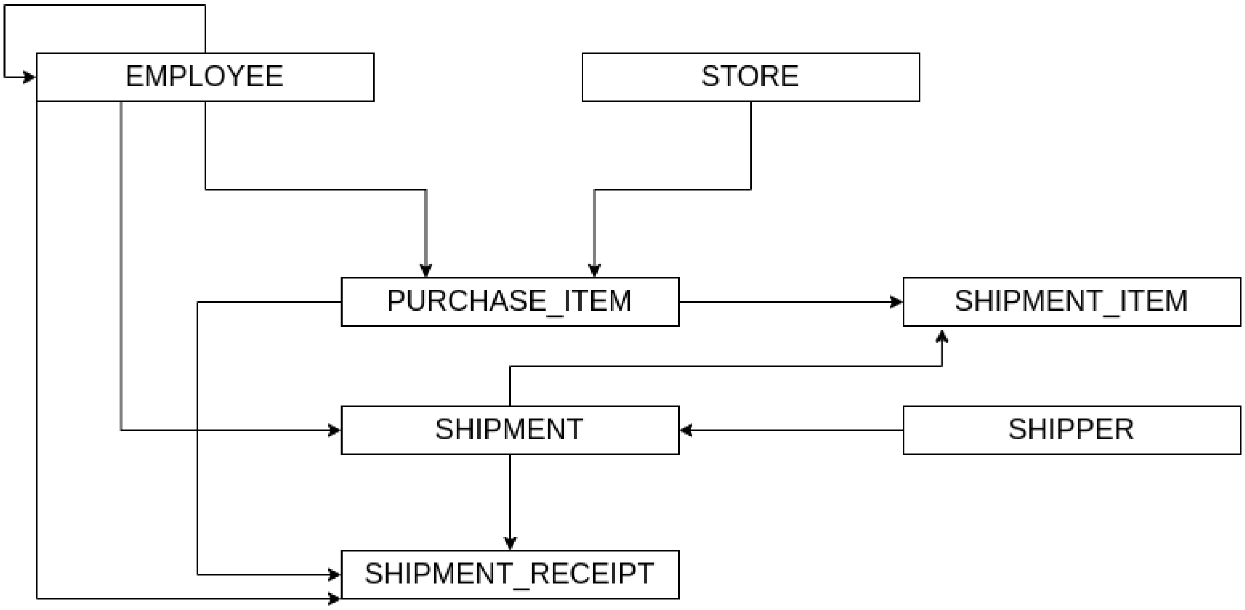
Your answer should look like this:

```
/* Your Names-Group Name */
/* *** CS1-2.17 *** */
SELECT       SKU, SKU_Description
FROM         INVENTORY;

/* *** CS1-2.18 *** */
SELECT       SKU, SKU_Description
FROM         INVENTORY;

/* *** CS1-2.19 *** */
SELECT       SKU, SKU_Description
FROM         INVENTORY;
```

Please write your solution below:

/* Noah L. Schrick */

/* *** CS8-MI.A *** */



Views and stored procedures can be incorporated into the dependency graph by creating objects for them. Each object can then be connected with directed edges to the tables shown above.

/* *** CS8-MI.B *** */

1.) Create a new table called ITEM, with ItemID as the primary key, and columns of ItemDescription and Category.

2.) Alter views, triggers, and constraints as needed.

3.) Copy PurchaseItemID from PURCHASE_ITEM to ITEM, changing the column name to ItemID. Copy ItemDescription and Category from PURCHASE_ITEM to ITEM.

```
/* *** CS8-MI.C *** */
CREATE TABLE ITEM (
        ItemID INT IDENTITY(500,5),
        ItemDescription  CHAR(60),
        Category CHAR(25),
        PRIMARY KEY(ItemID)
);

INSERT INTO ITEM (ItemID, ItemDescription, Category)
        SELECT PurchaseItemID, ItemDescription, Categeory
                FROM PURCHASE_ITEM
;
```

```
/* *** CS8-MI.D *** */
```

1.) Create a new table called SHIPMENT_LINE_ITEM.

2.) Alter views, triggers, and constraints as needed.

    2a) Drop foreign key constraints.

3.) Copy all data from SHIPMENT_ITEM to SHIPMENT_LINE_ITEM.

4.) Add foreign key constraints.

5.) Drop SHIPMENT_ITEM.

```
/* *** CS8 – MI.E *** */

CREATE TABLE SHIPMENT_LINE_ITEM(

ShipmentID INT FOREIGN KEY REFERENCES SHIPMENT(ShipmentID),

ShipmentLineNumber INT,

ItemID INT FOREIGN KEY REFERENCES ITEM(ItemID),

InsuredValue INT,

PRIMARY KEY(ShipmentID, ShipmentItemID)

)


INSERT INTO SHIPMENT_LINE_ITEM (ShipmentID, ShipmentLineNumber, ItemID, InsuredValue)

        SELECT ShipmentID, ShipmentItemID, PurchaseItemID, InsuredValue

        FROM SHIPMENT_ITEM;


DROP TABLE SHIPMENT_ITEM;


/* *** CS8-MI.F ***  */
```

1.) Create new tables called INVOICE and INVOICE_LINE_ITEM.

2.) Alter views, triggers, and constraints as needed.

    2a) Drop foreign key constraints.

3.) Copy PurchaseItemID, InvoiceDate, StoreID, PurchasingAgentID, SubtotalUSD, TaxUSD, and TotalUSED from PURCHASE_ITEM to INVOICE.

4.) Copy PurchaseItemID, Quantity, UnitPriceUSD, ExtendedPriceUSD from PURCHASE_ITEM to INVOICE_LINE_ITEM.

4.) Add foreign key constraints.

5.) Drop PURCHASE_ITEM

```
/* *** CS8-MI.G *** */
CREATE TABLE INVOICE(
InvoiceNumber INT IDENTITY(1,1),
InvoiceDate DATE,
StoreID INT FOREIGN KEY REFERENCES STORE(StoreID),
PurchasingAgentID FOREIGN KEY REFERENCES EMPLOYEE(EmployeeID),
SubtotalUSD INT,
TaxUSD INT,
TotalUSD INT
PRIMARY KEY(InvoiceNumber)
)

CREATE TABLE INVOICE_LINE_ITEM(
LineNumber INT IDENTITY(1,1),
InvoiceNumber INT FOREIGN KEY REFERENCES INVOICE(InvoiceNumber),
ItemID INT FOREIGN KEY REFERENCES ITEM(ItemID),
Quantity INT,
UnitPriceUSD INT,
ExtendedPriceUSD INT,
PRIMARY KEY(InvoiceNumber, LineNumber)
);

INSERT INTO INVOICE(InvoiceNumber, InvoiceDate, StoreID, PurchasingAgentID, SubtotalUSD, TaxUSD, TotalUSD)
	SELECT PurchaseDate, StoreID, PurchasingAgentID, PriceUSD
		FROM PURCHASE_ITEM
;

INSERT INTO INVOICE_LINE_NUMBER(ItemID)
	SELECT ItemID
		FROM ITEM
;

DROP TABLE PURCHASE_ITEM;
```