Soal
Case

MADYLINE

Madyline is a beauty center managed by your best friend, Mady. Mady manage all of activities that belongs to **Madyline** like **provide beauty treatment service to customer** and **purchasing the beauty product with supplier**.

To improve **Madyline's** management, Mady hired you as a database administrator are required to analyze and design a database for Bob's shop using **SQL Server Management Studio**.

Every employee that hired by **Madyline** have a task to **serve a customer who wants to do treatment** and **purchase the beauty product from supplier**. Every employee must be following the procedures, which are:

- Every **employee** hired must have a personal information like **name**, **gender**, **date of birth**, **phone number**, **email**, and **salary**. Every **employee** has an identification number with the following format:

"EPYXXX" X => number between 0 – 9

- The employee can purchase the beauty product with a supplier.
- Every **purchase transaction** made with the supplier have all the information about **supplier**, **employee, purchase transaction date, beauty products purchased, payment type, and the quantity of each product**. Every **purchase transaction** has an identification number with the following format:

"PCTXXX" X => number between 0 – 9

- Every beauty **product** purchased from supplier have its own **name** and **price**. Every **product** has an identification number with the following format:

"PDTXXX" X => number between 0 – 9

- Staff can also serve a customer who wants to do treatment.

- Every **service transaction** made by the customer have all the information about **customer**, **employee**, **room theme**, **service transaction date,** and the **treatments**. Every **service transaction** has an identification number with the following format:

"SVTXXX" X => number between 0 – 9

- Every **room theme** provided by **Madyline** has its own **name**. Every **room theme** has an identification number with the following format:

"ROTXXX" X => number between 0 – 9

- Every **treatment** provided by **Madyline** have its own **name**, and **price**. Every **treatment** has an identification number with the following format:

"TRMXXX" X => number between 0 – 9

Every customer that wants to have treatment at **Madyline** must be following the **service transaction procedures**, those are:

- Every **customer** that wants to do treatment must already completed personal information like **name, gender**, **phone number**, **email**, and **address**. Every **customer** has an identification number with the following format:

"CTRXXX" X => number between 0 – 9

- Customer can take **more than one treatment** in every transaction.

Every supplier that wants to sell their beauty product must be following the **purchase transaction procedures**, those are:

- Every **supplier** that wants to sell their beauty product must already completed personal information like **name**, **address**, **phone number**, and **email**. Every **supplier** has an identification number with the following format:

"SPRXXX" X => number between 0 – 9

- Supplier can sell **more than one beauty product** in every transaction.

Notes:

- The customer gender must be either “Male” or “Female” (without quote).
- The length of customer name must be equal or more than 3 characters.
- The length of employee phone must be equal to 12 digits.
- The room theme must end with ‘ Suite’ (without quote).
- The treatment price must be between 250000 and 20000000.
- The supplier email must contain ‘@’ symbol (without quote).
- The year of the service transaction date must be in 2021.

Now **Madyline** still using manual management system to maintain the **service transactions** and **purchase transactions**. You as her precious friend wants to help **Madyline** to create a database system that can store data and maintain the **service transactions** and **purchase transactions**. The tasks that you must do are:

a. Create **Entity Relationship Diagram** to maintain **service transactions** and **purchase transactions**.
b. Create a database system using **DDL syntax** that relevant with **service transactions** and **purchase transactions**.
c. Create query using **DML syntax** to fill the tables in database systems with data based on the following conditions:
    - **Master** table must be filled with more than or equals 10 data.
    - **Transaction** table must be filled with more than or equals 15 data.
    - **Transaction detail** table must be filled with more than or equals 25 data.
d. Create query using **DML syntax** to **simulate** the transactions process for **service transactions** and **purchase transactions**.

    Note: DML syntax to **fill database** and DML syntax to **simulate** the **transactions process** should be a **different query**.
e. To support database management process in **Madyline**, Mady asked you to provide some query that resulting important data. The requirements that asked from her are:
    1. Display EmployeeName, SupplierID, and Purchase Transaction Quantity (obtained from the sum of quantity), and PaymentType for every purchase transaction which the transaction happened on the 25th day and the id of the supplier is ‘SPR008’.

2. Display CustomerID, CustomerName, ServiceTransactionDate and Total Treatment (obtained from the count of the treatment) for every customer whose gender is female and the Total Treatment is greater than 2.

3. Display SupplierName, Purchase Transaction Date (obtained from purchase date in 'yyyy.mm.dd'), Total Product Type (obtained from the count of product purchased), and Total Quantity (obtained from the sum of quantity) for every purchase transaction which the transaction happened on the 9th month and Total Quantity is greater than 200.

4. Display EmployeeName, Employee Gender (obtained the first character of employee's gender), CustomerName, Total Service Transaction (obtained from the count of service transaction) for every service transaction that happened on even day and the average of employee salary is less than 5000000.

5. Display CustomerName (obtained from customer's name in uppercase format), CustomerGender, RoomThemeName, and Service Transaction Date (obtained from service date in 'yyyy.mm.dd') for every service transaction that happened on the 25th day and the salary of employee is less than the average of all employee's salary.
   (alias subquery)

6. Display SupplierName, PurchaseDate (obtained from purchases date in 'Mon dd, yyyy' format), and ProductName (obtained from the product's name in lowercase format) for every purchase transaction with product price is greater than the average price of all products and the supplier's name ends with ' Distribution'.
   (alias subquery)

7. Display Total Purchase Transaction (obtained from count of purchase transactions and ended with ' Transaction(s)'), SupplierName, Employee Name (obtained from employee's name from the first character until a character before space), and PurchaseTransactionDate for every purchase transaction with the quantity is lower than the average quantity of all purchase transaction and the purchase transaction happened on Wednesday.
   (alias subquery)

8. Display SupplierName, Purchase Transaction Date (obtained from purchases date in 'mm/dd/yyyy' format), ProductName, Product Number (obtained from ProductID by replacing the first three characters with 'PR'), and Total Purchase Transaction (obtained from the count of the transaction) for every purchase transaction where the quantity is lower than the sum of all quantity that have ProductID is 'PDT005'. Sort the result by supplier name in descending order.
   (alias subquery)

9. Create a view named '**ViewPurchaseTransaction**' to display EmployeeName, SupplierName, Total Purchase Transaction (obtained from the count of purchase transaction), and Maximum Purchase (obtained from the maximum of quantity) for every employee whose name contains 'd' character and the Total Purchase Transaction is greater than 1.

10. Create a view named '**ViewEmployeeSalaryinRoomDetail**' to display EmployeeName. EmployeeSalary, ServiceTransactionID, Total Treatment Transaction (obtained from the count of treatment) for every transaction that happened in room 'ROT001' and average of employee salary is less than 4000000.

File that must be collected:

1. Entity Relationship Diagram (.vpp, .png)
2. Query to create the database system. (.sql)
3. Query to insert data into tables. (.sql)
4. Query to simulate the transactions processes. (.sql)
5. Query to answer the 10 cases. (.sql)