

National School of Business Management

CS105.3 - Database Management System

Group Assignment 16.2

(Group of 4 students)

Batch : MIS/CS/SE/PLY 16.2

**Declaration**

|  |  |  |
| --- | --- | --- |
| **Student ID** | **Student Name** | **Signature** |
| BSC-PLY-COM-16.2-**244** | Renuka, P.H. Heshan |  |
| BSC-PLY-COM-16.2**-218** | Pramuditha, P.G. shan |  |
| BSC-PLY-COM-16.2-**157** | Lekamge, W.L. Dhananja Y.S. |  |
| BSC-PLY-COM-16.2-**304** | Wickramasinghe, W.A. Chethana D. |  |

Date of Submission : ……........................................

**Table of contents**

1. **Scenario -- 04**
2. **Conceptual Design -- 07**
3. **Relational Mapping -- 08**
4. **Assumptions -- 10**
5. **Normalization -- 11**
6. **SQL -- 16**
7. **Contribution of group members -- 22**

**01.Scenario**

National School of Business Management

**CS105.3 Database Management Systems**

**Group Assignment (group of 4 students)**

**Batch: MIS/CS/SE/PLY 16.2**

**Hand out date: week of 12th December 2016**

**This assessment contributes to 25% of the module mark**

You are required to analyse the scenario given below and design and develop a DB solution to

answer the queries of interest.

**Scenario : Online Order Processing System**

A merchandiser maintains a catalog of products online. Customers can browse throw the

catalog and place orders for the items. Website needs customers to be registered before

placing any orders. Submitted orders are processed daily which involves updating the

merchandiser’s items inventory and raising the invoices for the customers. Once the customers make the payments, shipment details (delivery notes) are generated. Successful delivery of goods updates the shipment details and completes the transaction.

Incase ordered items (Customer orders) are not available in the inventory, merchandiser raises a back order (Supplier order) for its registered set of suppliers to get down the unavailable items to merchandizer warehouse. Payments are settled for the registered suppliers on a monthly basis for the goods they supplied during the month. Once the goods are received from the suppliers, customer order processing proceeds as explained before.

A system is required to support the above operations of the merchandiser and generate

necessary reports for monitoring and management of the business operations.

The system should facilitate the following reports/queries or SQLs.

a) Items report with the details of all items in the catalog.

b) Query to list down the items that has been orders during the last 3 months.

c) Query to list down the **details of the items** that has been orders during the last 3

months.

d) Query to list down the items that has **not been orders** during the last 3 months.

e) Daily report to list down the item details of customer orders placed during the day.

f) Report to List down all items in the catalog but not available in the stores.

g) Query to list down all items in the pending (not processed yet) customer orders but not

available in the stores.

h) Query to update the Customer Order ‘CO0012’ to increase the item quantity of ‘IT004’

to 16.

i) Write down the SQLs required to enter a new customer order (first insert the header

record and then insert the item detail records).

j) Write SQLS to delete the Customer order ‘CO0023’.

Please clearly specify any assumptions you make.

**Deliverables:**

1. Report including,

The problem scenario with any assumptions you made

ER diagram

Relational mapping

Normalized tables

Conclusion with a justification of your solution and limitations.

2. CD that includes

a soft copy of your report

an SQL script to generate the database with any constraints

An SQL script to insert sample data for the tables

An SQL script carrying the SQLs to generate all the required reports

**Marking Criteria:**

Following items carry a group mark.

1. Quality of the documentation – 10 Marks

2. Introduction to the problem and assumptions – 10 marks

Following items are assessed on individual student basis

3. ER Diagram - 20 Marks

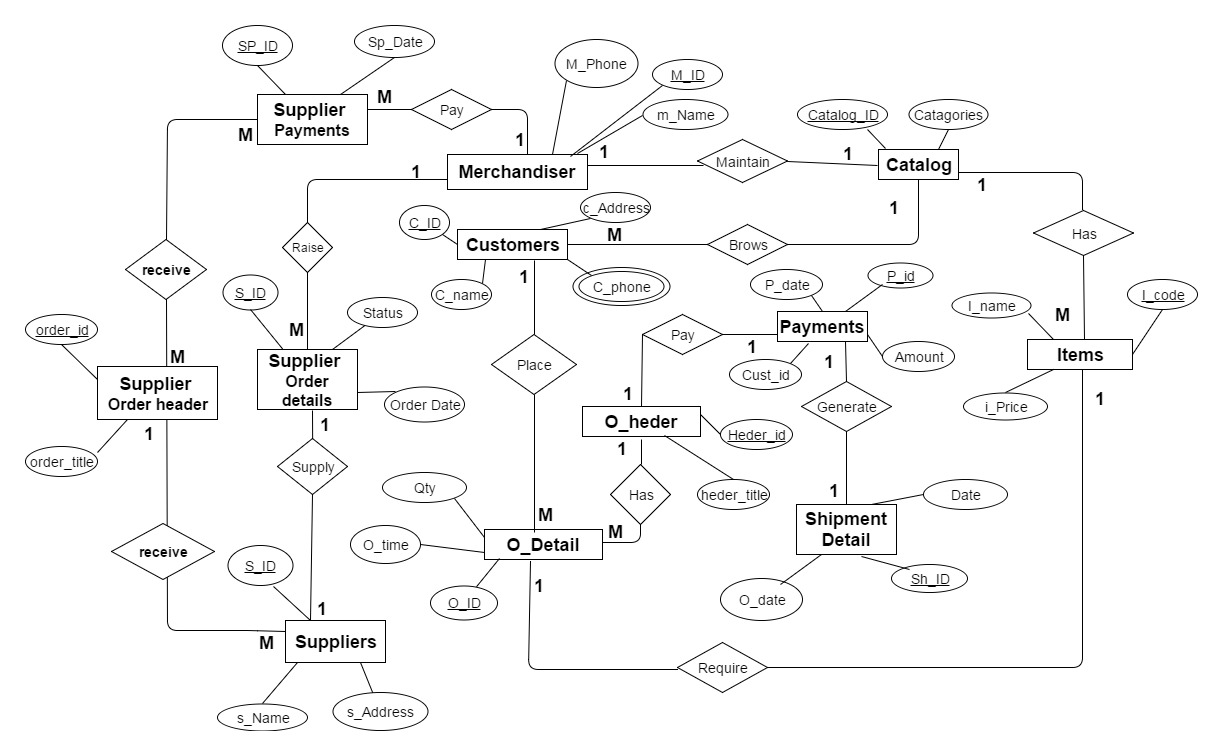
4. Normalization - 10 Marks

5. SQLs to create the database and constraints - 20 Marks

6. SQLs to load the sample data – 10 Marks

7. SQLs to generate the reports – 20 Mark

**2. Conceptual Design- ER Diagram**

****

**3.Relational Mapping**

Supplier\_payment (SS\_id , Date , M\_id)

Merchandiser (m\_id , M\_name , m\_place)

Catalog (catalog\_id , catagories , m\_id)

Customer (c\_id , c\_name, c\_address , catlog\_id)

Payments (p\_id , p\_date , amount , cust\_id)

Items (i\_code , i\_name ,i\_ price , cost\_id)

Supplier order detail (s\_id , status , order\_date , m\_id)

Supplier order heder (order\_id , order\_title , s\_id)

Order header (heder\_id , heder\_title , p\_id)

Order detail (o\_id , o\_time , qty , i\_code , c\_id , h\_id)

Suppliers (s\_id , s\_name , s\_address , s\_id , order\_id)

Shipment detail (sh\_id , date , o\_date , p\_id)

C\_phone ( cust\_id , c\_phone )

Receive ( so\_id , order\_id )

**4.Assumptions**

**General assumptions**

* Assuming that one supplier can supply one order detail list,
* Assuming that customer can browse only one catalog,
* Assuming that one order header can have more than one details,
* Assuming that supplies can receive only one order header,
* Assuming that merchandiser only can maintain one catalog,

# **5.Normalization**

* Supplier\_payment (SS\_id , Date , M\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘SS\_id’.

\* Therefore the relation ‘Supplier\_payment’ is in 3NF.

* Merchandiser (m\_id , M\_name , m\_place)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘m\_id ’.

\* Therefore the relation ‘Merchandiser’ is in 3NF.

* Catalog (catalog\_id , catagories , m\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘catalog\_id ’.

\* Therefore the relation ‘Catalog’ is in 3NF.

* Customer (c\_id , c\_name, c\_address , catlog\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘c\_id’.

\* Therefore the relation ‘Customer’ is in 3NF.

* Payments (p\_id , p\_date , amount , cust\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘p\_id.

\* Therefore the relation ‘Payments’ is in 3NF.

* Items (i\_code , i\_name ,i\_ price , cost\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘i\_code’.

\* Therefore the relation ‘Items’ is in 3NF.

* Supplier order detail (s\_id , status , order\_date , m\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘s\_id’.

\* Therefore the relation ‘Supplier order detail’ is in 3NF.

* Supplier order heder (order\_id , order\_title , s\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘order\_id’.

\* Therefore the relation ‘Supplier order heder’ is in 3NF.

* Order header (heder\_id , heder\_title , p\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘heder\_id’.

\* Therefore the relation ‘Order header’ is in 3NF.

* Order detail (o\_id , o\_time , qty , i\_code , c\_id , h\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘o\_id’.

\* Therefore the relation ‘Order detail’ is in 3NF.

* Suppliers (s\_id , s\_name , s\_address , s\_id , order\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘s\_id’.

\* Therefore the relation ‘Suppliers’ is in 3NF.

* Shipment detail (sh\_id , date , o\_date , p\_id)

\* Every attribute in this relation is atomic.

\* Every non-primary key attribute is fully dependent on the primary key ‘sh\_id’.

\* Therefore the relation ‘Shipment detail ‘ is in 3NF.

* C\_phone ( cust\_id , c\_phone )

\* Every attribute in this relation is atomic.

* Receive ( so\_id , order\_id )

\* Every attribute in this relation is atomic.

**6.SQL**

**SQL Statements for database creation & constrains**

1. CREATE TABLE Customer

(

C\_Id INT NOT NULL PRIMARY KEY,

C\_name VARCHAR (15) NOT NULL,

Address VARCHAR (50) NOT NUL,

C\_phone CHAR (10), NOT NUL,

Catalog\_Id INT,

CONSTRAINT fk\_CatCustomer FOREIGN KEY (Catalog\_Id)  
REFERENCES Catalog(Catalog\_Id),

)

1. CREATE TABLE O\_details

(

O\_Id int NOT NULL PRIMARY KEY,

O\_time TIME,

Qty INT,

C\_Id int,  
CONSTRAINT fk\_CusOrders FOREIGN KEY (C\_Id)  
REFERENCES Customer (C\_Id),

)

1. CREATE TABLE Merchandiser

(

M\_Id int NOT NULL PRIMARY KEY,

M\_Phone CHAR (10),

M\_Name VARCHAR (30),

)

1. CREATE TABLE Payment

(

P\_Id int NOT NULL PRIMARY KEY,

P\_Date DATE,

Cust\_Id INT ,

Amount INT,

Sh\_Id int ,

CONSTRAINT fk\_Shipment\_Payment FOREIGN KEY (Sh\_Id)  
REFERENCES Shipment\_Details (Sh\_Id),

)

1. CREATE TABLE Supplier

(

S\_Id int NOT NULL PRIMARY KEY,

S\_name VARCHAR (30) NOT NULL,

S\_Address VARCHAR (50) NOT NUL,

)

1. CREATE TABLE Supplier\_Order\_details

(

S\_Id int NOT NULL PRIMARY KEY,

Order\_Date DATE,

Status VARCHAR(30),

S\_Id int ,

m\_Id int,

CONSTRAINT fk\_SupOrders FOREIGN KEY (S\_Id )  
REFERENCES Suppliers (S\_Id ),

CONSTRAINT fk\_Merch\_SupOrders FOREIGN KEY (m\_Id )  
REFERENCES Merchandiser (m\_Id ),

)

1. CREATE TABLE Supplier\_Payment

(

SP\_Id int NOT NULL PRIMARY KEY,

Sp\_Date DATE ,

order\_Id int ,

M\_Id int,

CONSTRAINT fk\_SupOrders FOREIGN KEY (order\_Id )  
Supplier\_order\_heder(order\_Id ),

CONSTRAINT fk\_Merch\_ SupPayment FOREIGN KEY (M\_Id )

REFERENCES Merchandiser (M\_Id )

)

1. CREATE TABLE Catalog

(

Catalog\_Id int NOT NULL PRIMARY KEY,

Categories VARCHAR (15) NOT NULL,

M\_Id int,

CONSTRAINT fk\_Merch\_Catelog FOREIGN KEY (m\_Id )  
REFERENCES Merchandiser (m\_Id )  
)

1. CREATE TABLE Item

(

I\_Code int NOT NULL PRIMARY KEY,

I\_Name VARCHAR (15) NOT NULL,

I\_Price MONEY,

Cat\_Id int,

O\_ID int,

CONSTRAINT fk\_Item\_Catelog FOREIGN KEY (Catalog\_Id)  
REFERENCES Catalog (Catalog\_Id),

CONSTRAINT fk\_OrderDetails\_Items FOREIGN KEY (O\_ID)  
REFERENCES O\_Detail(O\_ID),  
)

1. CREATE TABLE Shipment\_Details

(

Sh ID INT NOT NULL PRIMARY KEY,

Ship\_Date DATE NOT NULL,

O Date DATE,

Item\_Code int ,

CONSTRAINT fk\_Payments\_Shipmentdetail FOREIGN KEY (P\_ID )  
REFERENCES Payments (P\_ID),

)

1. CREATE TABLE O\_Heder

(

Heder\_Id int NOT NULL PRIMARY KEY,

Heder Title VARCHAR (15) NOT NULL,

P\_Id int,

O\_Id int,

CONSTRAINT fk\_payments\_Oheder FOREIGN KEY (P\_ID )  
REFERENCES payments (P\_ID),

CONSTRAINT fk\_ODetails \_Oheder FOREIGN KEY (O\_ID )  
REFERENCES O\_Detail(O\_ID),

)

1. CREATE TABLE Supplier\_Order\_Heder

(

Order\_Id int NOT NULL PRIMARY KEY,

Order\_Title VARCHAR (15) NOT NULL,

SP\_Id int,

O\_Id int,

CONSTRAINT fk\_Suppliers\_SupOrderHeder FOREIGN KEY (S\_ID )  
REFERENCES Suppliers (S\_ID),

CONSTRAINT fk\_ SupplierPayements \_ SupOrderHeder FOREIGN KEY (SP\_ID )  
REFERENCES Suppliers\_Payements (SP\_ID),

)

***7. Contribution of group members ,***

**Lekamge , W.L. Dhananja Y.S. & Wickramasinghe, W.A. Chethana D. -:**

* Conceptual Design - ER diagram
* Relational mapping
* Assumptions

**Renuka, P.H. Heshan & Pramuditha, P.G. shan -:**

* Normalization
* SQL