NORTH SOUTH UNIVERSITY

Database Management System Lab-CSE311L

Database Design

Section:06

Group no: 08

Member One
Shahariar Ifti
2012632042

Member Two
Noman Saffat Sajid
2131578642

Scope:

The browsing and ordering module offers functionality to effectively store, retrieve and update products and order information for seamless customer client experience. This facility is provided using the safe, manageable and scalable database system technology.

Task 1:

Browsing And Ordering Module Entities:
Orders, Payments, Shipping_Methods, Shipping_carriers, Products, Categories, Brands, Reviews, Ratings, Supplier, Inventory, Inventory_Audit, Customer, Address
Total Number of Entities: 14

Task 2:

Many to Many:

1. Products and Brands
2. Products and Supplier
3. Categories and Brands

Many to One:

1. Orders and Shipping_Methods
2. Orders and Shipping_Carriers
3. Products and Categories
4. Customer and Address
5. Order and Address

One to Many

1. Shipping methods and Shipping_Carriers
2. Products and Reviews
3. Customer and payment

One to One:

1. Orders and Payments
2. Product and Inventory_Audit

Weak Entity:

1. Products and Ratings
2. Products and Inventory

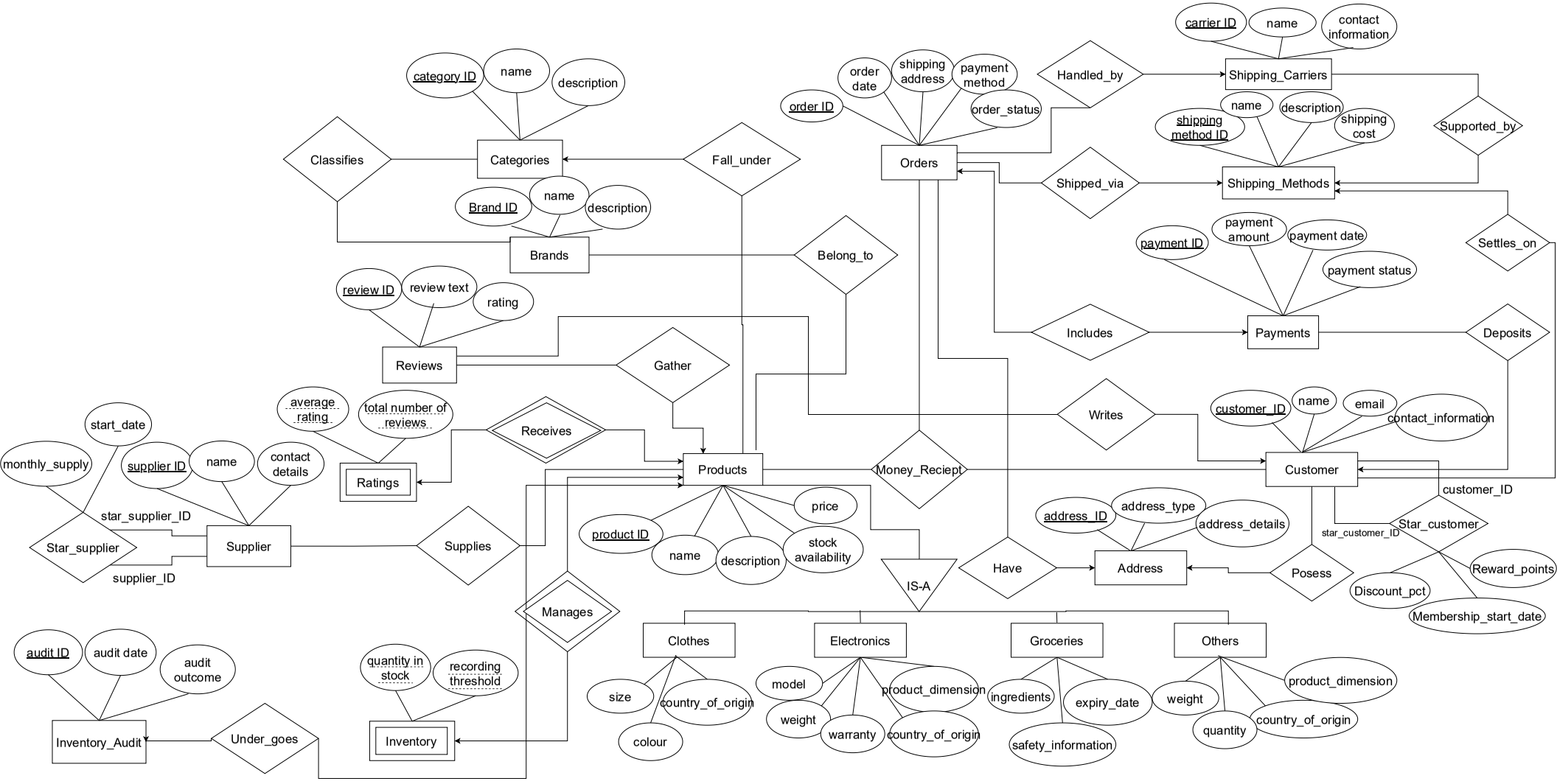
Roles:

1. Customer (Star_customer)
2. Supplier (Star_supplier)

Ternary:

1. Products, Orders, Customers

Task3:


carrier ID
name
contact information
Shipping_Carriers
Handled_by
order date
shipping address
payment method
order ID
order_status
Orders
shipping method ID
name
description
shipping cost
Shipping_Methods
Supported_by
category ID
name
description
Categories
Classifies
Fall_under
Shipped_via
Brand ID
name
description
Brands
Belong_to
payment ID
payment amount
payment date
payment status
Settles_on
review ID
review text
rating
Reviews
Gather
Includes
Payments
Deposits
average rating
total number of reviews
Receives
Writes
customer_ID
name
email
contact_information
start_date
monthly_supply
supplier ID
name
contact details
Ratings
Products
Money_Reciept
Customer
customer_ID
star_supplier_ID
Star_supplier
Supplier
Supplies
supplier_ID
price
product ID
name
description
stock availability
address_ID
address_type
address_details
star_customer_ID
Star_customer
Reward_points
Discount_pct
Membership_start_date
IS-A
Have
Address
Posess
Manages
Clothes
Electronics
Groceries
Others
audit ID
audit date
audit outcome
quantity in stock
recording threshold
size
country_of_origin
colour
model
weight
warranty
product_dimension
country_of_origin
ingredients
expiry_date
safety_information
weight
quantity
product_dimension
country_of_origin
Inventory_Audit
Under_goes
Inventory

Relational Schema:

1. Customer(customer_ID, address_ID, shipping_ID, name, email, contact_information)
2. Address(address_ID, address_type, address_details)
3. Product(product_ID, name, description, stock_availability, price)
4. Order(order_ID, order_date, shipping_address, payment_method, order_status, carrier_ID, shipping_method_ID, address_ID)
5. Money_Receipt(customer_ID, product_ID, order_ID)
6. Star_Customer(customer_ID, star_customer_ID, reward_points, discount_pct, membership_start_date)
7. Payment(payment_ID, order_ID, customer_ID, payment_amount, payment_date, payment_status)
8. Shipping_Method(shipping_method_ID, name, description, shipping_cost)
9. Shipping_Carriers(carrier_ID, shipping_method_ID, name, contact_information)
10. Categories(category_ID, name, description, product_ID)
11. Brands(Brand_ID, name , description)
12. Classifies(category_ID, brand_ID)
13. Belong_To(brand_ID, product_ID)
14. Review(review_ID, review_text, rating, product_ID, customer_ID)
15. Ratings(product_ID, average_rating, total_number_of_reviews)
16. Supplier(supplier_ID, name, contact_details)
17. Star_Supplier(supplier_ID, star_supplier_ID, monthly_supply, start_date)
18. Supplies(supplier_ID, product_ID)
19. Inventory(product_ID, quantity_in_stock, reordering_threshold)
20. Inventory_Audit(audit_ID, product_ID, audit_date, audit_outcome)

21. Clothes(product_ID, size, colour, country_of_origin)
22. Electronics(product_ID, model, weight, warranty, product_dimension, country_of_origin)
23. Groceries(product_ID, ingredients, safety_information, expiry_date)
24. Others(product_ID, weight, quantity, country_of_origin, product_dimension)

SQL Scripts:

1. Create TABLE Customer(customer_ID int(20) PRIMARY KEY, address_ID int(20), shipping_method_ID int(20), name varchar(20), contact_information varchar(20), FOREIGN KEY(address_ID) REFERENCES address(address_ID), FOREIGN KEY(shipping_method_ID) REFERENCES shipping_method(shipping_method_ID))
2. Create TABLE Address(address_ID INT(20) PRIMARY KEY, address_type varchar(20), address_details varchar(20))
3. Create TABLE Products(product_ID INT(20) PRIMARY KEY, name varchar(20) , description varchar(20), stock_availability int(20), price float(10))
4. Create TABLE Orders(order_ID int(20) PRIMARY KEY, order_date date, shipping_address varchar(20), payment_method varchar(10), order_status varchar(10), carrier_ID int(20), shipping_method_ID int(20), address_ID int(20), FOREIGN KEY(shipping_method_ID) REFERENCES shipping_method(shipping_method_ID), FOREIGN KEY(carrier_ID) REFERENCES shipping_carriers(carrier_ID), FOREIGN KEY(address_ID) REFERENCES address(address_ID))
5. Create table Money_Reciept(customer_ID int(20), product_ID int(20), order_ID int(20), PRIMARY KEY(customer_ID, product_ID, order_ID), FOREIGN KEY(customer_ID) REFERENCES customer(customer_ID), FOREIGN KEY(product_ID) REFERENCES products(product_ID), FOREIGN KEY(order_ID) REFERENCES orders(order_ID))
6. CREATE TABLE Star_Customer(star_customer_ID int(20), customer_ID int(20), reward_points int(10), discount_pct int(5), membership_start_date date, PRIMARY KEY(customer_ID, star_customer_ID, reward_points, discount_pct, membership_start_date), FOREIGN KEY(customer_ID) REFERENCES customer(customer_ID))

7. CREATE table Payments(payment_ID int(20) PRIMARY KEY, order_ID int(20), customer_ID int(20), payment_amount float(20), payment_date date, payment_status varchar(10), FOREIGN KEY(customer_ID) REFERENCES customer(customer_ID), FOREIGN KEY(order_ID) REFERENCES orders(order_ID))

8. Create TABLE Shipping_Method(shipping_method_ID INT(20) PRIMARY KEY, name varchar(20), description varchar(20), shipping_cost float(20))

9. Create TABLE Shipping_Carriers(carrier_ID int(20) PRIMARY KEY, shipping_method_ID INT(20), name varchar(20), contact_information varchar(20), FOREIGN KEY(shipping_method_ID) REFERENCES shipping_method(shipping_method_ID))

10. CREATE TABLE Categories(category_ID int(20) PRIMARY KEY, name varchar(20), description varchar(20), product_ID int(20), FOREIGN KEY(product_ID) REFERENCES products(product_ID))

11. Create TABLE Brands(brand_ID INT(20) PRIMARY KEY , name varchar(20) , description varchar(20))

12. CREATE TABLE Classifies(category_ID int(20), brand_ID int(20), PRIMARY KEY(category_ID, brand_ID), FOREIGN KEY(category_ID) REFERENCES categories(category_ID), FOREIGN KEY(brand_ID) REFERENCES brands(brand_ID))

13. Create TABLE Belong_To(brand_ID INT(20), product_ID INT(20), PRIMARY KEY(brand_ID, product_ID), FOREIGN KEY(brand_ID) REFERENCES brands(brand_ID), FOREIGN KEY(product_ID) REFERENCES products(product_ID))

14. Create TABLE Review(review_ID INT(20), review_text varchar(50) , rating int(5), product_ID int(20), customer_ID int(20), FOREIGN KEY(product_ID) REFERENCES products(product_ID), FOREIGN KEY(customer_ID) REFERENCES customer(customer_ID))

15. CREATE TABLE Ratings (product_ID INT(20), average_rating INT(5), total_number_of_reviews INT(20), PRIMARY KEY(product_ID, average_rating, total_number_of_reviews), FOREIGN KEY(product_ID) REFERENCES products(product_ID));

16. CREATE TABLE Supplier (supplier_ID INT(20) PRIMARY KEY, name VARCHAR(30), contact_details VARCHAR(50));

17. CREATE TABLE Star_Supplier (supplier_ID INT(20), star_supplier_ID INT(20), monthly_supply INT(20), start_date DATE, PRIMARY KEY(supplier_ID, star_supplier_ID, monthly_supply, start_date), FOREIGN KEY (supplier_ID) REFERENCES Supplier(supplier_ID));

18. CREATE TABLE Supplies (supplier_ID INT(20), product_ID INT(20), PRIMARY KEY(supplier_ID, product_ID), FOREIGN KEY (supplier_ID) REFERENCES Supplier(supplier_ID), FOREIGN KEY (product_ID) REFERENCES Products(product_ID));

19. CREATE TABLE Inventory (product_ID INT(20), quantity_in_stock INT(20), reordering_threshold INT(20), PRIMARY KEY(product_ID, quantity_in_stock, reordering_threshold), FOREIGN KEY (product_ID) REFERENCES Products(product_ID)

    );

20. CREATE TABLE Inventory_Audit (audit_ID INT(20) PRIMARY KEY, product_ID INT(20), audit_date DATE, audit_outcome VARCHAR(50), FOREIGN KEY (product_ID) REFERENCES Products(product_ID));

21. CREATE TABLE Clothes (product_ID INT(20), size VARCHAR(10), colour VARCHAR(20), country_of_origin VARCHAR(50), FOREIGN KEY (product_ID) REFERENCES Products(product_ID));

22. CREATE TABLE Electronics (product_ID INT(20), model VARCHAR(50), weight float(20), warranty varchar(20), product_dimension VARCHAR(50), country_of_origin VARCHAR(50), FOREIGN KEY (product_ID) REFERENCES Products(product_ID));

23. CREATE TABLE Groceries (product_ID INT(20), ingredients varchar(20), safety_information varchar(20), expiry_date DATE, FOREIGN KEY (product_ID) REFERENCES Products(product_ID));

24. CREATE TABLE Others (product_ID INT(20), weight float(20), quantity INT(20), country_of_origin VARCHAR(50), product_dimension VARCHAR(50), FOREIGN KEY (product_ID) REFERENCES Products(product_ID));