Introduction:

E-commerce, also known as electronic commerce or internet commerce, refers to the buying and selling of goods or services using the internet, and the transfer of money and data to execute these transactions.

E-Commerce is an online store where customers can browse categories and select products of interest. The selected items may be collected in a shopping cart. the items in the shopping cart will be presented as an order if customer submit it. At that time, more information will be needed to complete the transaction. Usually, the customer will be asked to fill or select a billing address, a shipping address, a shipping option, and payment information such as credit card number. An e-mail notification is sent to the customer as soon as the order is placed.

Types of E-commerce Models:

There are four main types of ecommerce models that can describe almost every transaction that takes place between consumers and businesses.

1. Business to Consumer (B2C):

When a business sells a good or service to an individual consumer

2. Business to Business (B2B):

When a business sells a good or service to another business (e.g. A business sells software-as-a-service for other businesses to use)

3. Consumer to Consumer (C2C):

When a consumer sells a good or service to another consumer (e.g. You sell your old furniture on eBay to another consumer).

4. Consumer to Business (C2B):

When a consumer sells their own products or services to a business or organization

Software system scope:

In our scope we won't be able to cover all aspects of the system but we will cover them later. So now, we will discuss our scope showing which aspects including in our system and which is not.

Our system focused on **Business to Consumer (B2C) model** which declared before.

Aspects covered in our System Scope:

- 1- Browse categories to choose certain product.
- 2- Ordering the selected products.
- 3- Give a feedback to any product.
- 4- Return any product we don't need after ordering it at the specified time.

Aspects not covered in our System Scope:

- 1- Tracking orders.
- 2- Shipment Companies.

ERD: PRODUCT_PRICING **CATEGORY** # categoryID # priceID * categoryName * sellPrice * Description * buyPrice * Active * startDate endDate PUCHASE_ORDER **PRODUCT** FEEDBACK_REVIEW # productID # purchaseID # feedbackID * purchasedQuantity * productName * feedbackDate * Description * PurchasedDate * feedbackRate * startQuantity * feedbackComment * quantityDate availableQuantity **SUPPLIER CUSTOMER** # SupplierID # customerID ORDER_DETAILS * CompanyName * fullName * jobTitle * userName * quantity * fullName * passwrord * totalPrice * userName * E-mail * passwrord * address * E-mail gender * address phoneNumber gender postalCode phoneNumber **RETURNS** RETURNED PRODUCT # returnID * returnedQuantity **ORDERS** * returnDate # orderID * orderNumber * orderDescription *orderDate *deliveryDate * paymentMethod

SQL Commands:

```
1 CREATE TABLE CUSTOMER
 3
       CustomerID INT NOT NULL,
 4
       FullName VARCHAR (30) NOT NULL,
 5
       UserName VARCHAR (15) NOT NULL,
       Password VARCHAR (15) NOT NULL,
 6
 7
       Bdate DATE,
       Email VARCHAR (30) NOT NULL,
 8
 9
       Address VARCHAR (50) NOT NULL,
       Gender CHAR,
10
11
      PhoneNumber INT NOT NULL,
12
      PostalCode CHAR(5) NOT NULL,
13
      PRIMARY KEY (CustomerID),
14
      UNIQUE (UserName)
15 );
16 CREATE TABLE SUPPLIER
17 | (
18
       SupplierID INT
                                 NOT NULL,
19
       CompanyName VARCHAR (15),
      JobTitle VARCHAR(15) NOT NULL,
20
21
      FullName VARCHAR (30) NOT NULL,
22
      UserName VARCHAR(15) NOT NULL,
23
                  VARCHAR (15) NOT NULL,
      Password
       Bdate DATE,
24
       Email
25
                  VARCHAR (30) NOT NULL,
26
      Address VARCHAR (40) NOT NULL,
27
       Gender
                  CHAR,
      PhoneNumber INT
28
                             NOT NULL,
29
      PRIMARY KEY(SupplierID),
30l
      UNIQUE (UserName)
31 );
```

```
33 CREATE TABLE CATEGORY
34 (
35
       CategoryID
                       INT
                                       NOT NULL,
      CategoryName VARCHAR (15)
36
                                       NOT NULL,
       Description VARCHAR2 (1000),
37
38
      Active
                       CHAR
                                       NOT NULL,
39
       PRIMARY KEY (CategoryID),
40
      UNIQUE (CategoryName)
41 );
42 CREATE TABLE PRODUCT
43 (
44
      ProductID
                         INT
                                     NOT NULL ,
                       VARCHAR (15) NOT NULL,
45
      ProductName
46
      CategoryID
                                      NOT NULL,
                         INT
                        VARCHAR2 (1000) ,
47
      Description
      QantityDate
48
                        DATE
                                     NOT NULL,
      StartQuantity INT
49
                                    NOT NULL,
      AvailableQuantity INT
50
                                 NOT NULL,
51
      PRIMARY KEY (ProductID),
52
      UNIQUE (ProductName) ,
53
      FOREIGN KEY (CategoryID) REFERENCES CATEGORY (CategoryID)
54 );
56 CREATE TABLE PRODUCT PRICING
      ProductID
57 (
                          INT
                                  NOT NULL ,
58
      PriceID
                          INT
                                  NOT NULL,
                         FLOAT NOT NULL,
      SellPrice
59
                          FLOAT NOT NULL,
60
      BuyPrice
61
      StartDate
                          DATE NOT NULL,
62
      EndDate
                               NOT NULL ,
                          DATE
63
      PRIMARY KEY (PriceID),
      FOREIGN KEY (ProductID) REFERENCES PRODUCT (ProductID)
64
65);
```

```
66 CREATE TABLE ORDERS
67 (
68
       OrderID
                           INT
                                     NOT NULL,
69
       OrderNumber
                           INT
                                      NOT NULL,
70
                          VARCHAR2 (1000),
       Decription
71
       OrderDate
                          DATE
                                      NOT NULL ,
72
       DeliveryDate
                          DATE
                                      NOT NULL,
       PaymentMethod VARCHAR (15) NOT NULL,
73
74
       CustomerID
                           INT
                                      NOT NULL,
75
       PRIMARY KEY (OrderID),
76
       FOREIGN KEY (CustomerID) REFERENCES CUSTOMER (CustomerID)
77 );
78 CREATE TABLE ORDER DETAILS
79
80 (
       DetailsID
                           INT
81
                                         NOT NULL,
82
       OrderID
                           INT
                                        NOT NULL,
83
       ProductID
                                         NOT NULL ,
                           INT
                                            NOT NULL ,
84
       Quantity
                           INT
85
       TotalPrice
                           FLOAT
                                        NOT NULL,
86
       PRIMARY KEY(DetailsID),
87
       FOREIGN KEY (OrderID) REFERENCES ORDERS (OrderID),
       FOREIGN KEY (ProductID) REFERENCES PRODUCT (ProductID)
88
89 );
90 CREATE TABLE PURCHASE ORDER
91
92
       PurchaseID
                          INT NOT NULL,
       PurchasedQuantity INT
93
                                     NOT NULL,
       PurchasedDate DATE
94
                                     NOT NULL ,
95
       SupplierID
                          INT
                                    NOT NULL,
96
       ProductID
                          INT
                                     NOT NULL .
97
       PRIMARY KEY (PurchaseID),
       FOREIGN KEY (SupplierID) REFERENCES SUPPLIER (SupplierID),
98
99
       FOREIGN KEY (ProductID) REFERENCES PRODUCT (ProductID)
100 :
```

```
101 CREATE TABLE FEEDBACK REVIEWS
102 (
                INT
103 FeedbackID
                                   NOT NULL,
      FeedbackDate
                        DATE
104
                                   NOT NULL,
      FeedbackComment VARCHAR2 (1000),
105
                   INT
106
    ProductID
                              NOT NULL ,
     CustomerID
107
                        INT NOT NULL,
    PRIMARY KEY (FeedbackID),
FOREIGN KEY (CustomerID) REFERENCES CUSTOMER (CustomerID),
108
109
FOREIGN KEY (ProductID) REFERENCES PRODUCT (ProductID)
111
112 );
113 CREATE TABLE RETURNS
114 (
ReturnID INT NOT NULL,
ReturnDate DATE NOT NULL,
117 PRIMARY KEY (ReturnID)
118 );
119 CREATE TABLE RETURNED PRODUCT
120 ( ReturnID
                       INT NOT NULL,
121 DetailsID
                 INT NOT NULL,
122 ReturnedQuantity INT NOT NULL,
123 PRIMARY KEY (ReturnID, DetailsID),
124 FOREIGN KEY (ReturnID) REFERENCES RETURNS (ReturnID),
125 FOREIGN KEY (DetailsID) REFERENCES ORDER DETAILS (DetailsID)
126
127 | ;
```

Created ERD for all tables: SYSTEM.PRODUCT_PRICING SYSTEM.CATEGORY PRODUCTID NUMBER (*,0) CATEGORYID NUMBER (*,0) PRICEID NUMBER (*,0) VARCHAR2 (15 BYTE) CATEGORYNAME * SELLPRICE FLOAT (126) DESCRIPTION VARCHAR2 (1000 BYTE) BUYPRICE FLOAT (126) CHAR (1 BYTE) ACTIVE * STARTDATE DATE DATE ENDDATE 🖙 CATEGORY_PK (CATEGORYID) CATEGORY_CATEGORYNAME_UN (CATEGORYNAME) > PRODUCT_PRICING_PK (PRICEID) 🧏 SYS_C007080 (PRODUCTID) SYSTEM.FEEDBACK_REVIEWS SYSTEM.PRODUCT SYSTEM.PURCHASE_ORDER FEEDBACKID NUMBER (*,0) PRODUCTID NUMBER (*,0) * PURCHASEID NUMBER (*,0) FEEDBACKDATE DATE * PURCHASEDQUANTITY NUMBER (*,0) PRODUCTNAME VARCHAR2 (15 BYTE) FEEDBACKCOMMENT VARCHAR2 (1000 BYTE) CATEGORYID NUMBER (*,0) * PURCHASEDDATE DATE PRODUCTID NUMBER (*,0) SUPPLIERID DESCRIPTION VARCHAR2 (1000 BYTE) NUMBER (*,0) CUSTOMERID NUMBER (*,0) QANTITYDATE DATE * PRODUCTID NUMBER (*,0) FEEDBACK_REVIEWS_PK (FEEDBACKID) STARTQUANTITY NUMBER (*,0) PURCHASE_ORDER_PK (PURCHASEID) AVAILABLEQUANTITY NUMBER (*,0) 🛂 SYS_C007109 (CUSTOMERID) SYS_CO07102 (SUPPLIERID) SYS_COO7110 (PRODUCTID) PRODUCT_PK (PRODUCTID) 😼 SYS_COO7103 (PRODUCTID) PRODUCT_PRODUCTNAME_UN (PRODUCTNAME) 🛂 SYS_C007072 (CATEGORYID) SYSTEM.CUSTOMER P * CUSTOMERID NUMBER (*.0) VARCHAR2 (30 BYTE) * FULLNAME U * USERNAME VARCHAR2 (15 BYTE) PASSWORD VARCHAR2 (15 BYTE) SYSTEM.ORDER_DETAILS BDATE DATE DETAILSID NUMBER (*,0) SYSTEM.SUPPLIER * EMAIL VARCHAR2 (30 BYTE) NUMBER (*,0) ORDERID NUMBER (*,0) * ADDRESS VARCHAR2 (50 BYTE) SUPPLIERID PRODUCTID NUMBER (*,0) COMPANYNAME VARCHAR2 (15 BYTE) GENDER CHAR (1 BYTE) NUMBER (*,0) QUANTITY PHONENUMBER NUMBER (*,0) JOBTITLE VARCHAR2 (15 BYTE) TOTALPRICE FLOAT (126) * POSTALCODE CHAR (5 BYTE) FULLNAME VARCHAR2 (30 BYTE) ORDER_DETAILS_PK (DETAILSID) * USERNAME VARCHAR2 (15 BYTE) CUSTOMER_PK (CUSTOMERID) PASSWORD VARCHAR2 (15 BYTE) SYS_C007141 (ORDERID) CUSTOMER_USERNAME_UN (USERNAME) BDATE DATE 🛐 SYS_C007142 (PRODUCTID) **EMAIL** VARCHAR2 (30 BYTE) ADDRESS VARCHAR2 (40 BYTE) GENDER CHAR (1 BYTE) PHONENUMBER NUMBER (*,0) SYSTEM.ORDERS 🖙 SUPPLIER_PK (SUPPLIERID) ORDERID NUMBER (*,0) SUPPLIER_USERNAME_UN (USERNAME) ORDERNUMBER SYSTEM.RETURNED_PRODUCT NUMBER (*,0) VARCHAR2 (1000 BYTE) DECRIPTION PF* RETURNID NUMBER (*,0) ORDERDATE DATE NUMBER (*,0) SYSTEM.RETURNS DELIVERYDATE DATE RETURNEDQUANTITY NUMBER (*,0) RETURNID NUMBER (*.0) PAYMENTMETHOD VARCHAR2 (15 BYTE) RETURNED_PRODUCT_PK (RETURNID, DETAILSID) RETURNDATE CUSTOMERID NUMBER (*,0) RETURNS_PK (RETURNID) SYS_CO07147 (RETURNID) ORDERS_PK (ORDERID) 🕏 SYS_C007148 (DETAILSID) 🧣 SYS_C007088 (CUSTOMERID)

ERD Explanation:

Relationships between entities:

1- Category to Product:

one to many. we have several products in category but each product lies in one category.

2- Product to product_pricing:

one to many. price of each product varies by the time.

3- Product to Feedback Review:

one to many. Every product has many feedbacks but every feedback has a certain product.

4- Customer to Orders:

one to many. Customer may order more than one time but every order has one customer only.

5- Customer to Feedback-Review:

one to many. Customer may give many feedbacks to products but every feedback on this product is gave by this customer only. No one has the same feedback for sure.

6- Product to Order:

many to many. Customer may order many products at the same time, so in this order we have many products. And every product is ordered by many customers. So, we created a new entity which is called "Order_Details". It will inform us the quantity of products ordered by the same customer and the total price for these products.

7- Returns to Order_Details:

many to many. Products in the order may be returned or part of them. So, we created a new entity to know which products will be returned which is called "**Returned_Product**". It will inform us the quantity of the products that will be returned.

8- Supplier to Purchase_Order:

one to many. Supplier can supply many products

9- Product to Purchase_Order:

one to many. Products can be