Wholesale Management System Database Project

GROUP 17

PROJECT 16

DİLAN AYRANCI: 1910205026,

MEHMET DURMUŞ: 2110205073,

SEMİH BUĞRA SEZER: 2210205343,

HASAMUDDIN AFZALI: 2210205563

Scenario

- As a wholesaler of stationery products, we need a database system to effectively manage many activities such as stocking, purchasing, selling various products and managing customer relations. The database needs of our business and the requirements of the system to be created according to these needs are detailed as follows:

- For product management, each product must have a unique ID, name, quantity and unit sales price. For each product, a record is kept of how much product will be received when the order is placed and when the quantity of the product falls below which number, a new order will be placed. Products should be divided into specific categories and each category should have a unique ID and name. In addition, the brands to which the products belong should be recorded and each brand should have a unique identity and name.

- Within the scope of purchasing and supply management, each purchase should be recorded with a unique ID and should include information such as date of purchase, quantity and delivery date. The transactions made during the procurement of products should be recorded, including the purchase price and relevant product information. The names, managers, phone numbers and addresses of suppliers should be recorded.

- In sales and customer management, sales transactions should be recorded with a unique ID; information such as date of sale, quantity and estimated delivery date should be available. Customers' phone numbers, addresses and the company or individual customer they are should be recorded. Customers can be individual or corporate.

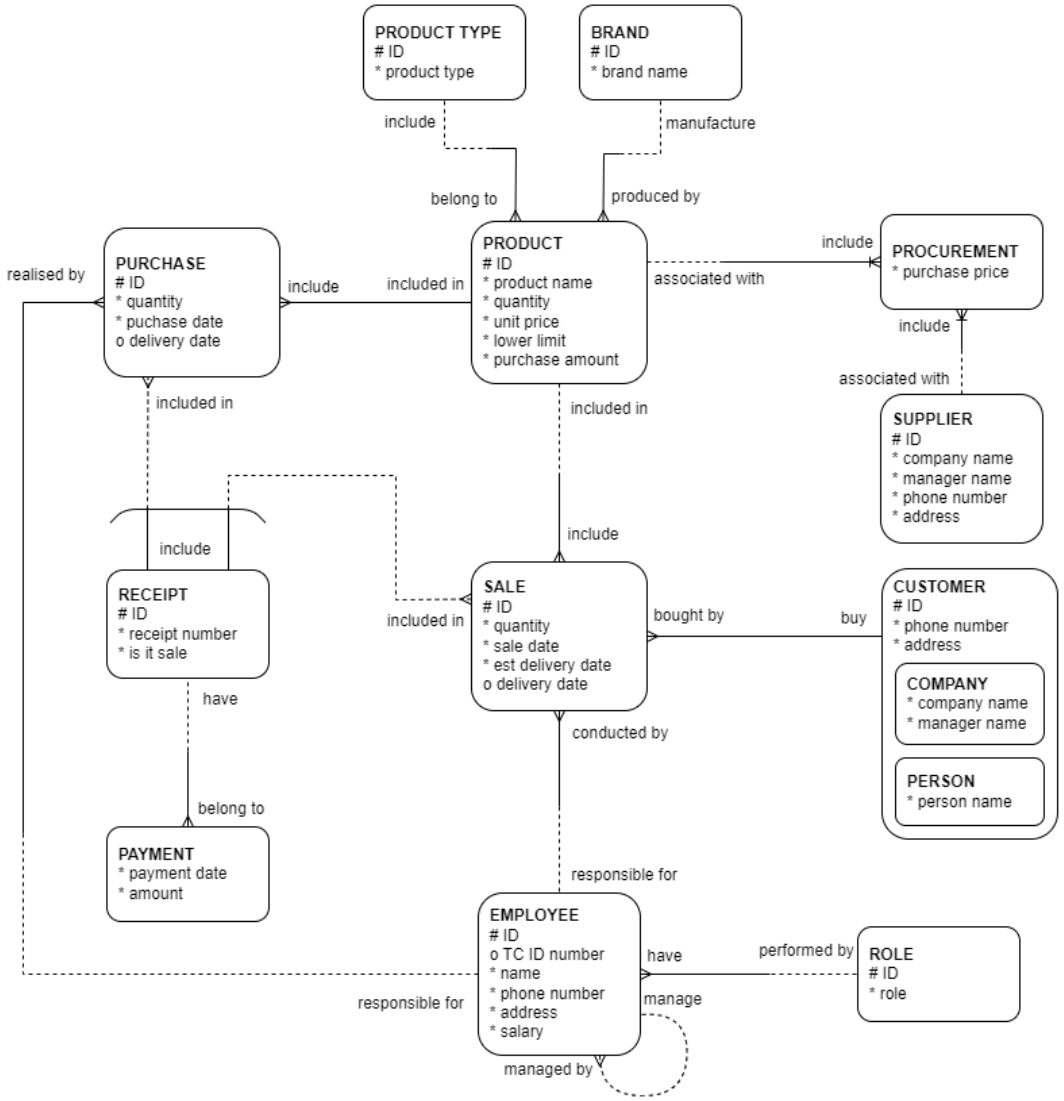
- For employees and management, employees' unique IDs, TR ID numbers, names, phone numbers and addresses should be recorded. Each employee should have a role and each role should have a unique ID and name. Employees are managed by a manager and each employee can have a manager. It should be recorded which employees made the sales and purchases.

- In the invoice and payments section, each sales transaction should have a unique invoice ID, invoice number and sales status. For invoice payments, the payment date and amount should be recorded.

- Based on the information recorded in the database, various reports will be generated to improve the performance and efficiency of our business. With these reports, information about the receivables, payables and monthly profit status of the business will be presented.

ERD

- **Notes for procedural business rules:**
  - If the required quantity of a product is not in stock, the estimated delivery date must be determined according to the time it takes to obtain the required stock.
  - The receipt numbers of the sales made must be generated by the program.



Matrix Diagram

	Product	Product Type	Brand	Procure-ment	Supplier	Customer	Sale	Purchase	Receipt	Payment	Employee	Role
Product		belong to	produced by	Associated with			included in	included in				
Product Type	include											
Brand	manufac-ture											
Procure-ment	include				include							
Supplier				Associated with								
Customer							buy					
Sale	include					bought by			included in		conducted by	
Purchase	include								included in		realised by	
Receipt							include	include		Have		
Payment									belong to			
Employee							responsible for	responsible for			manage / managed by	have
Role											Performed by	

Table Instance Charts

PRODUCTS

	ID	PRODUCT_NAME	PRODUCT_TYPE	BRAND	QUANTİTY	UNİT_PRİCE	LOWER_LİMİT	PURCHASE_AMOUNT
Key Type	primary key		foreign key	foreign key				
Null	no	no	no	no	no	no	no	no
Unique	yes	no	no	no	no	no	no	no
FK Column			Product Type	Brand				
Data Type	number	varchar2	number	number	number	number	number	number
Lenght	6	50	6	6	6	7,2	3	5

Table Instance Charts

Product Type

	ID	PRODUCT TYPE
Key Type	primary key	
Null	no	no
Unique	yes	yes
FK Column		
Data Type	number	varchar2
Lenght	6	50

Table Instance Charts

Brand

	ID	Brand Name
Key Type	primary key	
Null	no	no
Unique	yes	yes
FK Column		
Data Type	number	varchar2
Lenght	6	50

Table Instance Charts

Procurement

	Product ID	Supplier ID	Purchase Price
Key Type	foreign key	foreign key	
Null	no	no	no
Unique	no	no	no
FK Column			
Data Type	number	number	number
Lenght	6	6	7,2

Table Instance Charts

Supplier

	ID	Company Name	Manager Name	Phone Number	Address
Key Type	primary key				
Null	no	no	no	no	no
Unique	yes	yes	no	yes	yes
FK Column					
Data Type	number	varchar2	varchar2	varchar2	varchar2
Lenght	6	75	50	13	100

Table Instance Charts

Customer

	ID	Company Name	Manager Name	Person Name	Phone Number	Address
Key Type	primary key					
Null	no	yes	yes	yes	no	no
Unique	yes	yes	no	no	yes	no
FK Column						
Data Type	number	varchar2	varchar2	varchar2	varchar2	varchar2
Lenght	6	75	50	50	13	100

Table Instance Charts

Sale

	ID	Product ID	Customer ID	Receipt ID	Employee ID	Quantity	Sale Date	Est Delivery Date	Delivery Date
Key Type	primary key	foreign key	foreign key	foreign key	foreign key				
Null	no	no	no	no	no	no	no	no	yes
Unique	yes	no	no	no	no	no	no	no	no
FK Column		Product	Customer	Receipt	Employee				
Data Type	number	number	number	number	number	number	date	date	date
Lenght	6	6	6	8	6	5			

Table Instance Charts

Purchase

	ID	Product ID	Receipt ID	Employee ID	Quantity	Purchase Date	Delivery Date
Key Type	primary key	foreign key	foreign key	foreign key			
Null	no	no	yes	no	no	no	yes
Unique	yes	no	no	no	no	no	no
FK Column		Product	Receipt	Employee			
Data Type	number	number	number	number	number	date	date
Lenght	6	6	6	6	5		

Table Instance Charts

Receipt

	ID	Receipt Number	Is It Sale
Key Type	primary key		
Null	no	no	no
Unique	yes	no	no
FK Column			
Data Type	number	number	number
Lenght	6	8	1

Table Instance Charts

Payment

	Receipt ID	Payment Date	Amount
Key Type	foreign key		
Null	no	no	no
Unique	yes	no	no
FK Column	Receipt		
Data Type	number	date	number
Lenght	6		9,2

Table Instance Charts

Employee

	ID	TC ID Number	Name	Phone Number	Address	Role ID	Managed By	Salary
Key Type	primary key					foreign key	foreign key	
Null	no	yes	no	no	no	no	yes	no
Unique	yes	yes	no	no	no	no	no	no
FK Column						Role	Employee	
Data Type	number	varchar2	varchar2	varchar2	varchar2	number	number	number
Lenght	6	11	50	13	100	6	6	8,2

Table Instance Charts

Role

	ID	Role
Key Type	primary key	
Null	no	no
Unique	yes	yes
FK Column		
Data Type	number	varchar2
Lenght	6	20

Create Table Codes

Product Type

```
CREATE TABLE ProductTypes(
    ID NUMBER(6) PRIMARY KEY,
    product_type VARCHAR2(50) UNIQUE NOT NULL
)
```

Create Table Codes

Suppliers

```
CREATE TABLE Suppliers (
    ID NUMBER(6) PRIMARY KEY,
    company_name VARCHAR2(75) UNIQUE NOT NULL,
    manager_name VARCHAR2(50) NOT NULL,
    phone_number VARCHAR2(13) UNIQUE NOT NULL,
    address VARCHAR2(100) NOT NULL
)
```

Create Table Codes

Brands

```
CREATE TABLE Brands (
    ID NUMBER(6) PRIMARY KEY,
    brand_name VARCHAR2(50) UNIQUE NOT NULL
)
```

Create Table Codes

Customers

```
CREATE TABLE Customers (
    ID NUMBER(6) PRIMARY KEY,
    company_name VARCHAR2(50) UNIQUE,
    manager_name VARCHAR2(50),
    person_name VARCHAR2(50),
    phone_number VARCHAR2(13) NOT NULL,
    address VARCHAR2(100) NOT NULL
)
```

Create Table Codes

Roles

```
CREATE TABLE Roles (
    ID NUMBER(6) PRIMARY KEY,
    role VARCHAR2(20) UNIQUE NOT NULL
)
```

Create Table Codes

Receipts

```
CREATE TABLE Receipts (
        ID NUMBER(6) PRIMARY KEY,
    receipt_number NUMBER(8) NOT NULL,
    is_it_sale NUMBER(1) CHECK (is_it_sale IN (0, 1)) NOT NULL
)
```

Create Table Codes

Payments

```
CREATE TABLE Payments (
    receipt_id NUMBER(8) NOT NULL,
    payment_date DATE NOT NULL,
    amount NUMBER(7,2) NOT NULL,
    CONSTRAINT fk_receipt_id FOREIGN KEY (receipt_id) REFERENCES Receipts(ID)
)
```

Create Table Codes

Products

```
CREATE TABLE Products (
    ID NUMBER(6) PRIMARY KEY,
    product_name VARCHAR2(50) NOT NULL,
    product_type NUMBER(6) NOT NULL,
    brand NUMBER(6)  NOT NULL,
    quantity NUMBER(6) NOT NULL,
    unit_price NUMBER(7,2) NOT NULL,
    lower_limit NUMBER(3) NOT NULL,
    purchase_amount NUMBER(5) NOT NULL,
    CONSTRAINT fk_product_type FOREIGN KEY (product_type) REFERENCES ProductTypes(ID),
    CONSTRAINT fk_brand FOREIGN KEY (brand) REFERENCES Brands(ID)
)
```

Create Table Codes

Procurements

```
CREATE TABLE Procurements (
    product_id NUMBER(6) NOT NULL,
    supplier_id NUMBER(6) NOT NULL,
    purchase_price NUMBER(7,2) NOT NULL,
    CONSTRAINT fk_product_id FOREIGN KEY (product_id) REFERENCES Products(ID),
    CONSTRAINT fk_supplier_id FOREIGN KEY (supplier_id) REFERENCES Suppliers(ID)
)
```

Create Table Codes

Employees

```
CREATE TABLE Employees (
    ID NUMBER(6) PRIMARY KEY,
    tc_id_number VARCHAR2(11) UNIQUE,
    name VARCHAR2(50) NOT NULL,
    phone_number VARCHAR2(13) UNIQUE NOT NULL,
    address VARCHAR2(100) NOT NULL,
    salary NUMBER(6) NOT NULL,
    hire_date DATE() NOT NULL,
    role_id NUMBER(6) NOT NULL,
    managed_by NUMBER(6),
    CONSTRAINT fk_role_id FOREIGN KEY (role_id) REFERENCES Roles(ID),
    CONSTRAINT fk_managed_by FOREIGN KEY (managed_by) REFERENCES Employees(ID)
)
```

Create Table Codes

Sales

```
CREATE TABLE Sales(
    ID NUMBER(6) PRIMARY KEY,
    product_id NUMBER(6) NOT NULL,
    customer_id NUMBER(6) NOT NULL,
    receipt_id NUMBER(8),
    quantity NUMBER(5) NOT NULL,
    employee_id NUMBER(6) NOT NULL,
    sale_date DATE NOT NULL,
    est_delivery_date DATE NOT NULL,
    delivery_date DATE,
    CONSTRAINT fk_sale_product_id FOREIGN KEY (product_id) REFERENCES Products(ID),
    CONSTRAINT fk_customer_id FOREIGN KEY (customer_id) REFERENCES Customers(ID),
    CONSTRAINT fk_employee_id FOREIGN KEY (employee_id) REFERENCES Employees(ID),
    CONSTRAINT fk_sale_receipt_id FOREIGN KEY (receipt_id) REFERENCES Receipts(ID)
)
```

Create Table Codes

Purchases

```
CREATE TABLE Purchases(
    ID NUMBER(6) PRIMARY KEY,
    product_id NUMBER(6) NOT NULL,
    receipt_id NUMBER(8),
    employee_id NUMBER(6) NOT NULL,
    quantity NUMBER(5) NOT NULL,
    purchase_date DATE NOT NULL,
    delivery_date DATE,
    CONSTRAINT fk_purchase_product_id FOREIGN KEY (product_id) REFERENCES Products(ID),
    CONSTRAINT fk_purchase_receipt_id FOREIGN KEY (receipt_id) REFERENCES Receipts(ID),
    CONSTRAINT fk_purchase_employee_id FOREIGN KEY (employee_id) REFERENCES Employees(ID)
)
```