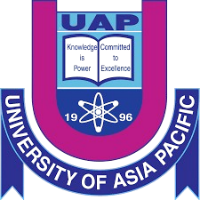
****

**UNIVERSITY OF ASIA PACIFIC**

**Department of CSE**

**Course Code:** **CSE 212**

**Course Title:** **Database Systems Lab**

**Project Name: Export and Import Management System**

**Submitted By:**

#### Md. Samimul Arefin - 23101029

#### Sanjida Siddiqa Mahima - 23101030

#### Mysha - 23101048

#### Nowshin Rahman Omi - 23101050

**Submitted to:**

**Nadeem Ahmed**

**Assistant Professor**

**Department of CSE**

**University of Asia Pacific**

**Export and Import Management System**

**1.Descripsion:**

This database management system is designed to track and manage export and import operations for an international trading company. It handles information about products, suppliers, customers, shipments, orders, inventory, and documentation required for international trade.

**Core Functionality:**

**Product Management**

* Comprehensive product catalog with detailed specifications and classifications
* Multi-supplier sourcing with price comparisons and lead time tracking
* Product certification and compliance documentation
* Country of origin tracking and rules of origin verification
* Product quality control metrics and inspection histories
* Custom tariff code assignment and duty calculation

**Supply Chain Management**

* End-to-end tracking from manufacturer to end customer
* Multi-modal transportation management (sea, air, road, rail)
* Container and pallet tracking with load optimization
* Port and customs clearance processing
* Transit time monitoring and exception alerts
* Freight cost management and carrier performance analytics

**Documentation Management**

* Automatic generation of commercial invoices and packing lists
* Certificate of origin creation and validation
* Import/export license management
* Customs declaration preparation
* Letter of credit compliance checking
* Harmonized System (HS) code verification
* Sanitary and phytosanitary certification

**Inventory Control**

* Real-time inventory visibility across global warehouses
* FIFO/LIFO/FEFO inventory valuation methods
* Lot and batch tracking with expiration date management
* Stock level alerts and automated reordering
* Warehouse capacity optimization
* Bonded warehouse handling for duty deferment
* Inventory aging analysis and obsolescence prevention

**Order Processing**

* Multi-currency order management
* International payment terms (FOB, CIF, EXW, DDP, etc.)
* Order consolidation for optimized shipping
* Split shipment handling and partial fulfillment
* Automated duty and tax calculation
* Customer-specific pricing and discount structures
* Order status tracking and notification system

**Compliance & Risk Management**

* Export control screening against denied party lists
* Embargo and sanctions compliance checking
* Import quota monitoring and license management
* Trade agreement qualification and preferential duty calculation
* Product safety and compliance documentation
* Supply chain risk assessment

**Financial Operations**

* Multi-currency invoicing and payment processing
* Exchange rate management and hedging
* Trade finance integration (letters of credit, bank guarantees)
* Landed cost calculation (product + freight + insurance + duties + taxes)
* Customs bond management
* Duty drawback and VAT recovery tracking

**Business Intelligence**

* Trade volume analytics by country, product, and customer
* Margin analysis accounting for all international trade costs
* Supply chain performance metrics and KPIs
* Compliance risk dashboards
* Market penetration and growth opportunity analysis
* Landed cost optimization recommendations
* Carrier performance and freight cost benchmarking

**2.Entities and Their Relationships**

### Entities:

**1. Products**  
  
Primary Key: product\_id  
Attributes: product\_name, unit\_price, packaging, supplier\_id  
Represents inventory items sold by the company  
  
  
**2. Suppliers**  
  
Primary Key: supplier\_id  
Attributes: contact\_person, email, phone\_number  
Represents vendors who provide products to the company  
  
  
**3. Packaging**  
  
Primary Key: packaging\_id  
Attributes: packaging\_type, unit\_capacity, weight\_capacity  
Represents different packaging options for products  
  
  
**4. Orders**  
  
Primary Key: order\_id  
Attributes: order\_date, customer\_id, product\_id, units, country\_id  
Represents customer purchases  
  
  
**5. Countries**  
Primary Key: country\_id  
Attributes: country\_name, import\_tax, export\_tax  
Represents geographical locations for shipping and regulatory purposes

**6. Warehouses**  
Primary Key: warehouse\_id  
Attributes: warehouse\_name, location, capacity, country\_id, administration\_id  
Represents physical storage locations

**7. Employees**  
  
Primary Key: employee\_id  
Attributes: employee\_name, gender, administration\_id, salary, email, phone\_number, supervisor\_id, joining\_date  
Represents staff members  
  
  
**8. Supervisors**  
  
Primary Key: supervisor\_id  
Attributes: supervisor\_name, gender, administration\_id, salary, email, phone\_number, joining\_date  
Represents employees with management responsibilities  
  
  
**9. Administration**  
  
Primary Key: administration\_id  
Attributes: administration\_name, administration\_work  
Represents organizational departments or divisions  
  
  
**10. ImportDocument**  
  
Primary Key: document\_id  
Attributes: document\_type, authority\_id, validity\_period  
Represents documentation for imported goods  
  
**11. ExportDocument**  
  
Primary Key: document\_id  
Attributes: document\_type, authority\_id, validity\_period  
Represents documentation for exported goods

**Relationships:**

### 1. Products - Suppliers (Supply)

### One-to-many: A supplier can provide multiple products

### Foreign Key: supplier\_id in Products references Suppliers

### 2. Products - Packaging (Packaging Category)

### One-to-many: A packaging type can be used for multiple products

### Foreign Key: packaging\_id in Products references Packaging

### 3. Products - Orders (Shipments)

### One-to-many: A product can appear in multiple orders

### Foreign Key: product\_id in Orders references Products

### 4. Orders - Countries (Belong)

### One-to-many: A country can be associated with multiple orders

### Foreign Key: country\_id in Orders references Countries

### 5. Countries - Warehouses (Stocks)

### One-to-many: A country can have multiple warehouses

### Foreign Key: country\_id in Warehouses references Countries

### 6. Warehouses - Administration (Inventory)

### Many-to-one: Multiple warehouses can be managed by an administration

### Foreign Key: administration\_id in Warehouses references Administration

### 7. Employees - Administration (Works)

### Many-to-one: Multiple employees can work in an administration

### Foreign Key: administration\_id in Employees references Administration

### 8. Supervisors - Administration (Works)

### Many-to-one: Multiple supervisors can work in an administration

### Foreign Key: administration\_id in Supervisors references Administration

### 9. Administration - ImportDocument (Import)

### One-to-many: An administration can be associated with multiple import documents

### Foreign Key: authority\_id in ImportDocument references Administration

### 10. Administration - ExportDocument (Export)

### One-to-many: An administration can be associated with multiple export documents

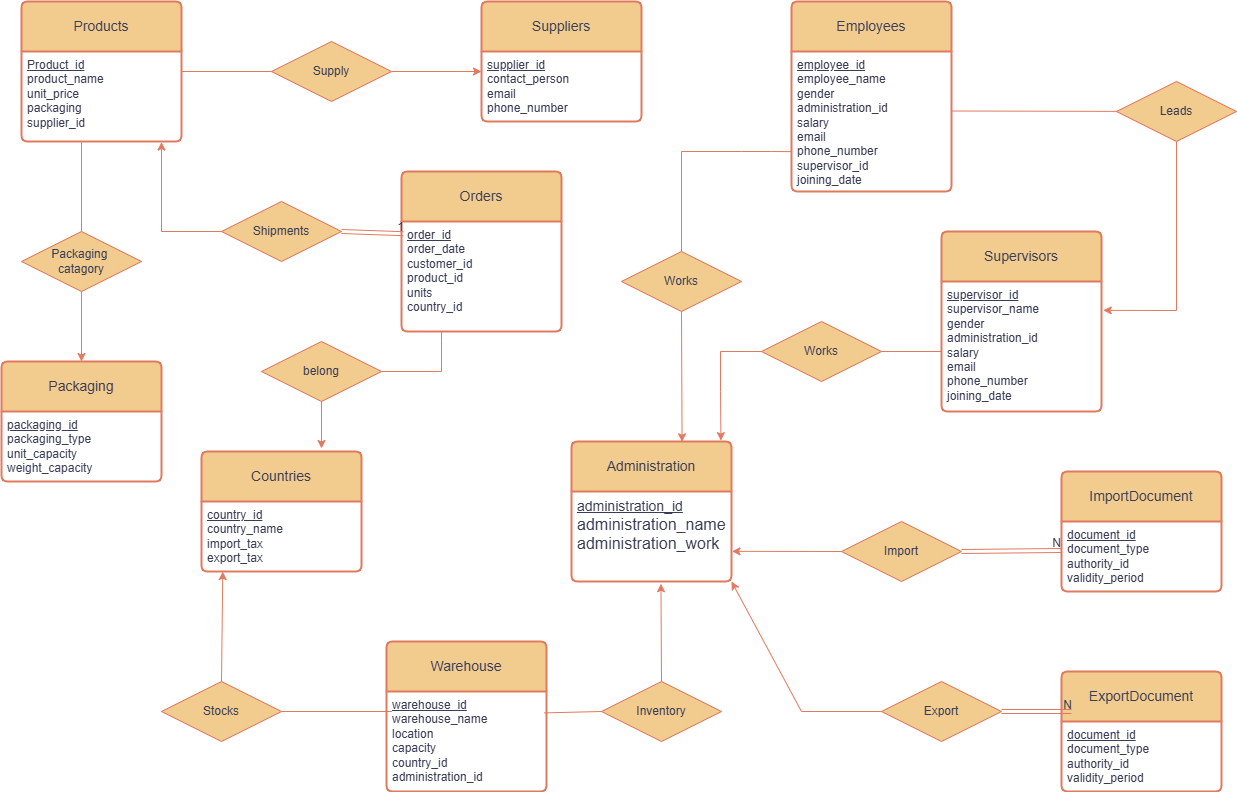
### Foreign Key: authority\_id in ExportDocument references Administration

### 11. Supervisors - Employees (Leads)

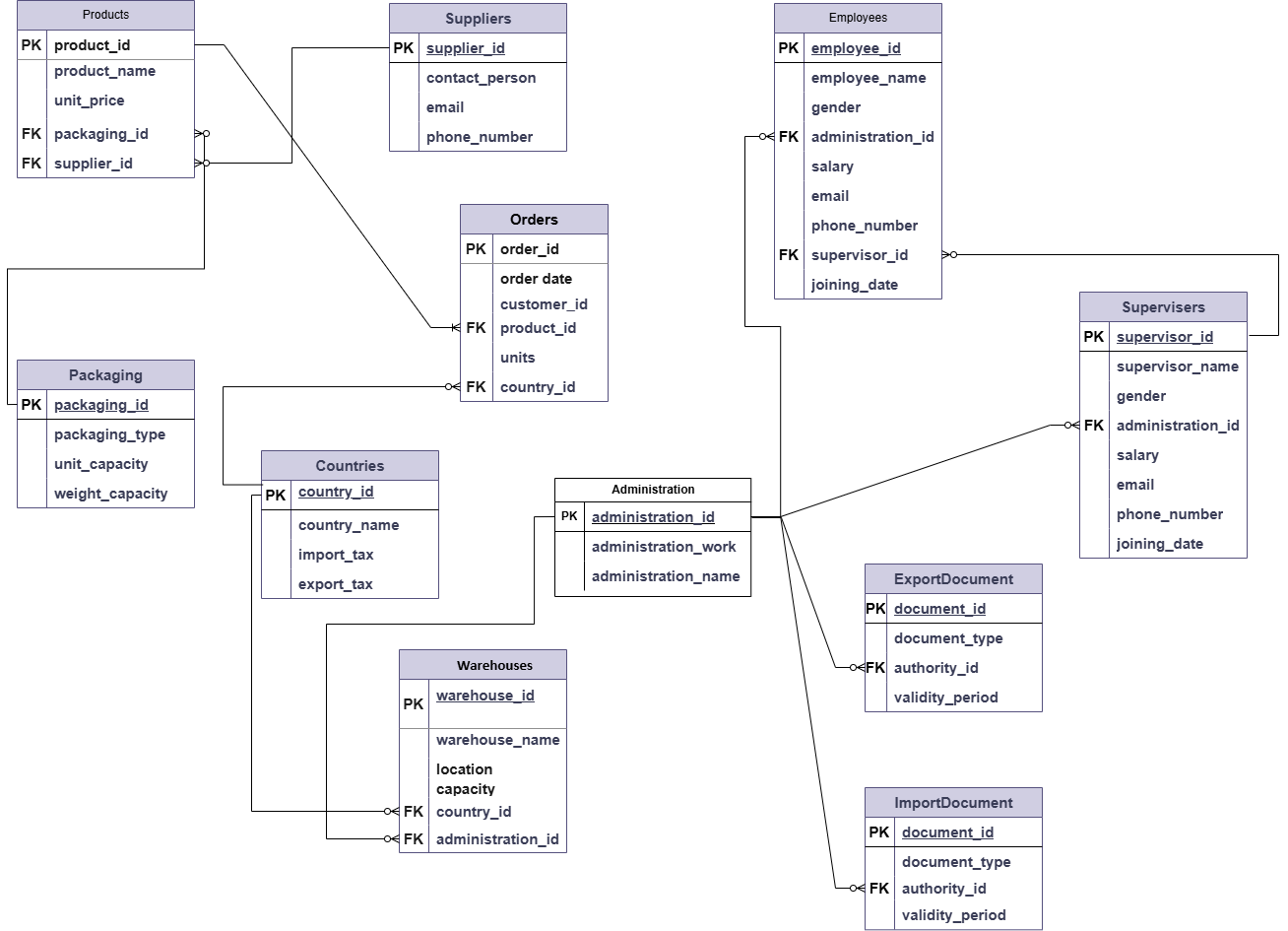
### One-to-many: A supervisor can lead multiple employees

### Foreign Key: supervisor\_id in Employees references Supervisors

**3. ER Diagram**

****

**4.Schema Diagram**

****

**5.SQL Code:**

**TABLE CREATE-**

CREATE DATABASE exportandimport

USE exportandimport

CREATE TABLE Suppliers (

supplier\_id INT PRIMARY KEY,

contact\_person VARCHAR(100),

email VARCHAR(100),

phone\_number VARCHAR(20)

)

-- Create Packaging table

CREATE TABLE Packaging (

packaging\_id INT PRIMARY KEY,

packaging\_type VARCHAR(50),

unit\_capacity DECIMAL(10,2),

weight\_capacity DECIMAL(10,2)

)

-- Create Products table

CREATE TABLE Products (

product\_id INT PRIMARY KEY,

product\_name VARCHAR(100),

unit\_price DECIMAL(10,2),

packaging\_id INT FOREIGN KEY

REFERENCES Packaging,

supplier\_id INT FOREIGN KEY

REFERENCES Suppliers

)

-- Create Countries table

CREATE TABLE Countries (

country\_id INT PRIMARY KEY,

country\_name VARCHAR(100),

import\_tax DECIMAL(5,2),

export\_tax DECIMAL(5,2)

)

-- Create Administration table

CREATE TABLE Administration (

administration\_id INT PRIMARY KEY,

administration\_name VARCHAR(100),

administration\_work VARCHAR(200)

)

-- Create Supervisors table

CREATE TABLE Supervisors (

supervisor\_id INT PRIMARY KEY,

supervisor\_name VARCHAR(100),

gender CHAR(1),

administration\_id INT FOREIGN KEY

REFERENCES Administration,

salary DECIMAL(10,2),

email VARCHAR(100),

phone\_number VARCHAR(20),

joining\_date DATE

)

-- Create Employees table

CREATE TABLE Employees (

employee\_id INT PRIMARY KEY,

employee\_name VARCHAR(100),

gender CHAR(1),

administration\_id INT FOREIGN KEY

REFERENCES Administration,

salary DECIMAL(10,2),

email VARCHAR(100),

phone\_number VARCHAR(20),

supervisor\_id INT FOREIGN KEY

REFERENCES Supervisors,

joining\_date DATE

)

-- Create Warehouse table

CREATE TABLE Warehouse (

warehouse\_id INT PRIMARY KEY,

warehouse\_name VARCHAR(100),

location VARCHAR(200),

capacity INT,

country\_id INT,

administration\_id INT FOREIGN KEY

REFERENCES Administration,

FOREIGN KEY (country\_id) REFERENCES Countries(country\_id)

)

-- Create ImportDocument table

CREATE TABLE ImportDocument (

document\_id INT PRIMARY KEY,

document\_type VARCHAR(50),

authority\_id INT FOREIGN KEY

REFERENCES Administration(administration\_id),

validity\_period DATE

)

-- Create ExportDocument table

CREATE TABLE ExportDocument (

document\_id INT PRIMARY KEY,

document\_type VARCHAR(50),

authority\_id INT FOREIGN KEY

REFERENCES Administration(administration\_id),

validity\_period DATE

)

-- Create Orders table

CREATE TABLE Orders (

order\_id INT PRIMARY KEY,

order\_date DATE,

customer\_id INT,

product\_id INT,

units INT,

country\_id INT,

FOREIGN KEY (product\_id) REFERENCES Products(product\_id),

FOREIGN KEY (country\_id) REFERENCES Countries(country\_id)

)

**INSERT DATA IN TABLE-**

-- Suppliers  
INSERT INTO Suppliers VALUES  
(1, 'Alice Johnson', '[alice@example.com](mailto:alice@example.com)', '1234567890'),  
(2, 'Bob Smith', '[bob@example.com](mailto:bob@example.com)', '2345678901'),  
(3, 'Carlos Ruiz', '[carlos@example.com](mailto:carlos@example.com)', '3456789012'),  
(4, 'Diana Lee', '[diana@example.com](mailto:diana@example.com)', '4567890123'),  
(5, 'Evan Thomas', '[evan@example.com](mailto:evan@example.com)', '5678901234'),  
(6, 'Fiona Zhang', '[fiona@example.com](mailto:fiona@example.com)', '6789012345'),  
(7, 'George Patel', '[george@example.com](mailto:george@example.com)', '7890123456');  
  
-- Packaging  
INSERT INTO Packaging VALUES  
(1, 'Box', 10.00, 50.00),  
(2, 'Bag', 5.00, 20.00),  
(3, 'Crate', 15.00, 100.00),  
(4, 'Envelope', 1.00, 1.00),  
(5, 'Tube', 2.50, 5.00),  
(6, 'Canister', 12.00, 60.00),  
(7, 'Bottle', 0.75, 1.50),  
(8, 'Drum', 50.00, 200.00);  
  
-- Products  
INSERT INTO Products VALUES  
(1, 'Laptop', 1500.00, 1, 1),  
(2, 'Phone', 800.00, 2, 2),  
(3, 'Tablet', 600.00, 3, 3),  
(4, 'Monitor', 300.00, 4, 4),  
(5, 'Keyboard', 50.00, 5, 5),  
(6, 'Mouse', 25.00, 6, 6),  
(7, 'Headphones', 75.00, 7, 7),  
(8, 'Webcam', 90.00, 8, 1);  
  
-- Countries  
INSERT INTO Countries VALUES  
(1, 'USA', 5.00, 3.00),  
(2, 'Germany', 6.00, 4.00),  
(3, 'India', 7.00, 5.00),  
(4, 'Brazil', 8.00, 6.00),  
(5, 'Canada', 4.00, 2.00),  
(6, 'Japan', 3.50, 2.50),  
(7, 'Australia', 5.50, 3.50),  
(8, 'UK', 4.25, 2.75);  
  
-- Administration  
INSERT INTO Administration VALUES  
(1, 'Logistics', 'Manage logistics and operations'),  
(2, 'HR', 'Handle human resources'),  
(3, 'Finance', 'Manage company finances'),  
(4, 'Procurement', 'Handle supplier relations'),  
(5, 'IT', 'Maintain IT systems'),  
(6, 'Legal', 'Ensure compliance'),  
(7, 'Marketing', 'Brand and promote'),  
(8, 'Operations', 'Oversee operations');  
  
-- Supervisors  
INSERT INTO Supervisors VALUES  
(1, 'John Doe', 'M', 1, 75000.00, '[john@example.com](mailto:john@example.com)', '3456789012', '2020-01-15'),  
(2, 'Jane Smith', 'F', 2, 72000.00, '[jane@example.com](mailto:jane@example.com)', '4567890123', '2021-03-20'),  
(3, 'Mike Evans', 'M', 3, 71000.00, '[mike@example.com](mailto:mike@example.com)', '5678901234', '2021-08-10'),  
(4, 'Linda Kim', 'F', 4, 76000.00, '[linda@example.com](mailto:linda@example.com)', '6789012345', '2020-11-05'),  
(5, 'Sarah Connor', 'F', 5, 74000.00, '[sarah@example.com](mailto:sarah@example.com)', '7890123456', '2019-06-01'),  
(6, 'Robert Paul', 'M', 6, 73000.00, '[robert@example.com](mailto:robert@example.com)', '8901234567', '2018-09-14'),  
(7, 'Aisha Ahmed', 'F', 7, 72500.00, '[aisha@example.com](mailto:aisha@example.com)', '9012345678', '2022-01-10');  
  
-- Employees  
INSERT INTO Employees VALUES  
(1, 'Tom Hardy', 'M', 1, 50000.00, '[tom@example.com](mailto:tom@example.com)', '5678901234', 1, '2022-06-10'),  
(2, 'Emily Clark', 'F', 2, 52000.00, '[emily@example.com](mailto:emily@example.com)', '6789012345', 2, '2023-02-05'),  
(3, 'Rahul Mehta', 'M', 3, 48000.00, '[rahul@example.com](mailto:rahul@example.com)', '7890123456', 3, '2022-09-12'),  
(4, 'Sara Gomez', 'F', 4, 49000.00, '[sara@example.com](mailto:sara@example.com)', '8901234567', 4, '2023-01-10'),  
(5, 'Brian Lee', 'M', 5, 46000.00, '[brian@example.com](mailto:brian@example.com)', '9012345678', 5, '2021-05-22'),  
(6, 'Nina Patel', 'F', 6, 47000.00, '[nina@example.com](mailto:nina@example.com)', '0123456789', 6, '2020-03-15'),  
(7, 'Jason Roy', 'M', 7, 51000.00, '[jason@example.com](mailto:jason@example.com)', '1234567890', 7, '2021-07-19');  
  
-- Warehouse  
INSERT INTO Warehouse VALUES  
(1, 'Central Warehouse', 'New York', 1000, 1, 1),  
(2, 'Europe Warehouse', 'Berlin', 800, 2, 2),  
(3, 'Asia Hub', 'Mumbai', 1200, 3, 3),  
(4, 'South America Depot', 'São Paulo', 700, 4, 4),  
(5, 'North Hub', 'Toronto', 900, 5, 5),  
(6, 'Pacific Storage', 'Tokyo', 950, 6, 6),  
(7, 'Southern Stock', 'Sydney', 850, 7, 7);  
  
-- ImportDocument  
INSERT INTO ImportDocument VALUES  
(1, 'Import Permit', 1, '2025-12-31'),  
(2, 'Customs Clearance', 2, '2025-06-30'),  
(3, 'Sanitary Certificate', 3, '2025-10-15'),  
(4, 'Safety Approval', 4, '2025-08-01'),  
(5, 'Quality Check', 5, '2025-11-15'),  
(6, 'Shipment Notice', 6, '2025-09-30'),  
(7, 'Product Registration', 7, '2025-07-01');  
  
-- ExportDocument  
INSERT INTO ExportDocument VALUES  
(1, 'Export Permit', 1, '2026-01-15'),  
(2, 'Shipping Certificate', 2, '2025-11-20'),  
(3, 'Destination Approval', 3, '2026-03-31'),  
(4, 'Transport Insurance', 4, '2025-12-01'),  
(5, 'Label Verification', 5, '2026-02-20'),  
(6, 'Packaging Cert.', 6, '2026-04-10'),  
(7, 'Environmental Doc', 7, '2026-05-05');  
  
-- Orders  
INSERT INTO Orders VALUES  
(1, '2025-05-01', 101, 1, 10, 1),  
(2, '2025-05-02', 102, 2, 5, 2),  
(3, '2025-05-03', 103, 3, 7, 3),  
(4, '2025-05-04', 104, 4, 8, 4),  
(5, '2025-05-05', 105, 5, 20, 5),  
(6, '2025-05-06', 106, 6, 15, 6),  
(7, '2025-05-07', 107, 7, 12, 7),  
(8, '2025-05-08', 108, 8, 9, 8);

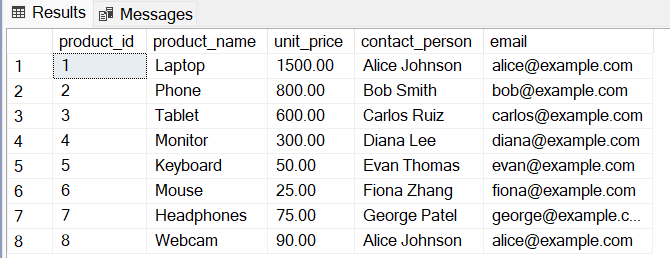
**6. SQL QUERY-**

--List all products with their supplier details

SELECT p.product\_id, p.product\_name, p.unit\_price, s.contact\_person, s.email

FROM Products p, Suppliers s

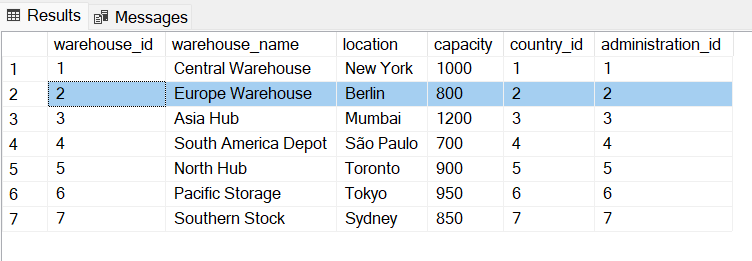
where p.supplier\_id = s.supplier\_id;

****

---Find all warehouses with their locations and capacities:

SELECT \*

FROM Warehouse

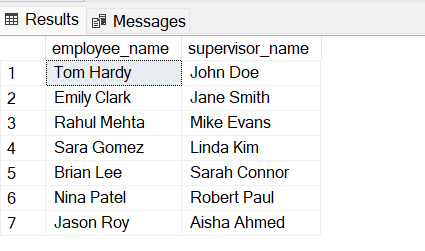
****

--List all employees with their supervisors

SELECT e.employee\_name, s.supervisor\_name

FROM Employees e,Supervisors s

where e.supervisor\_id = s.supervisor\_id



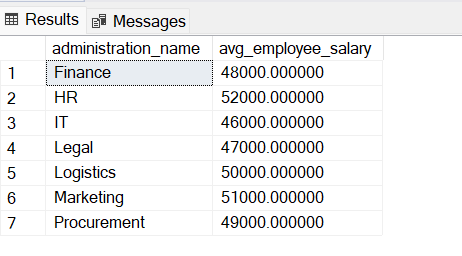
--Calculate the average salary by administration department

SELECT a.administration\_name, AVG(e.salary) AS avg\_employee\_salary

FROM Employees e,Administration a

where e.administration\_id = a.administration\_id

GROUP BY a.administration\_name;

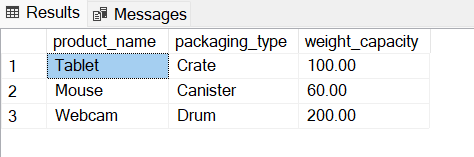


--Find products that are packaged in containers weighing more than 50 units:

SELECT p.product\_name, pk.packaging\_type, pk.weight\_capacity

FROM Products p,Packaging pk

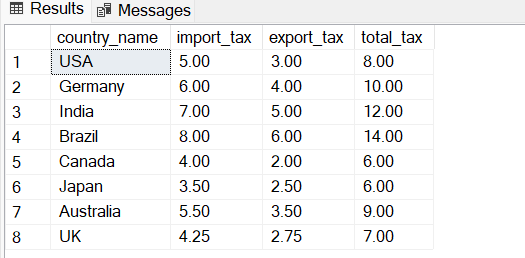
where p.packaging\_id = pk.packaging\_id and pk.weight\_capacity > 50;



--Calculate the total tax for each country's import and export

SELECT country\_name, import\_tax, export\_tax, (import\_tax + export\_tax) AS total\_tax

FROM Countries

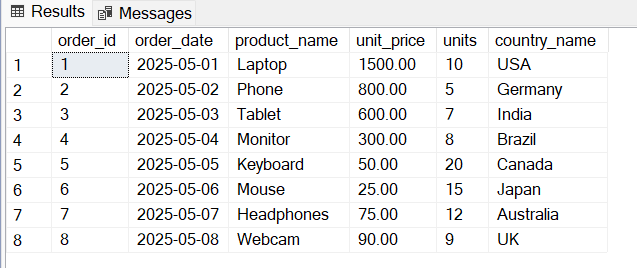


---Find all orders with their product details and customer countries:

SELECT o.order\_id, o.order\_date, p.product\_name, p.unit\_price, o.units, c.country\_name

FROM Orders o JOIN Products p ON o.product\_id = p.product\_id

JOIN Countries c ON o.country\_id = c.country\_id;



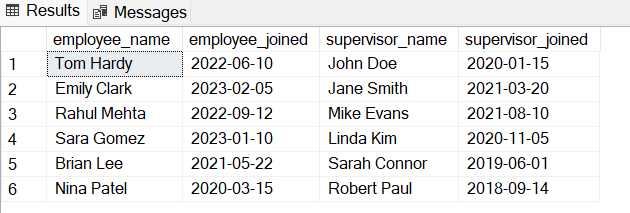
--List all employees who joined after their supervisors:

SELECT e.employee\_name, e.joining\_date AS employee\_joined,

s.supervisor\_name, s.joining\_date AS supervisor\_joined

FROM Employees e,Supervisors s

where e.supervisor\_id = s.supervisor\_id and e.joining\_date > s.joining\_date;



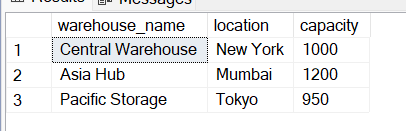
--Find warehouses that have capacity greater than the average capacity

SELECT w.warehouse\_name, w.location, w.capacity

FROM Warehouse w

WHERE w.capacity >

(SELECT AVG(capacity) FROM Warehouse);



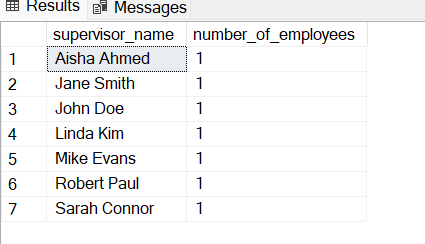
--Count the number of employees per supervisor:

SELECT s.supervisor\_name, COUNT(e.employee\_id) AS number\_of\_employees

FROM Supervisors s,Employees e Where s.supervisor\_id = e.supervisor\_id

GROUP BY s.supervisor\_name

ORDER BY number\_of\_employees DESC;



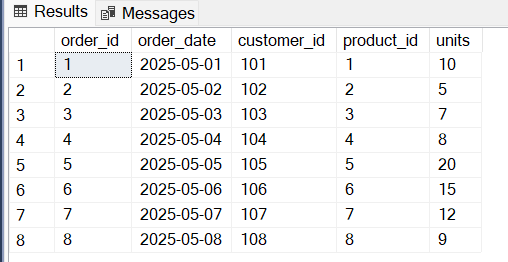
--Find all orders placed in May 2025

SELECT order\_id, order\_date, customer\_id, product\_id, units

FROM Orders

WHERE order\_date BETWEEN '2025-05-01' AND '2025-05-31'

ORDER BY order\_date;



--Find all people (employees and supervisors) who work in the Logistics department:

SELECT e.employee\_name AS name, 'Employee' AS role, e.email, e.phone\_number

FROM Employees e

JOIN Administration a ON e.administration\_id = a.administration\_id

WHERE a.administration\_name = 'Logistics'

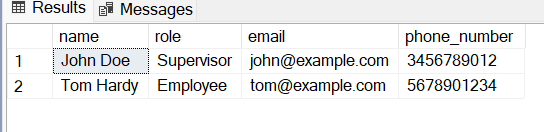
UNION

SELECT s.supervisor\_name AS name, 'Supervisor' AS role, s.email, s.phone\_number

FROM Supervisors s

JOIN Administration a ON s.administration\_id = a.administration\_id

WHERE a.administration\_name = 'Logistics'



---List employees who aren't supervisors based on contact information

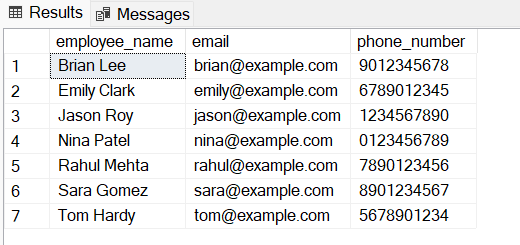
SELECT employee\_name , email, phone\_number

FROM Employees

EXCEPT

SELECT supervisor\_name, email, phone\_number

FROM Supervisors



---Find employees and supervisors who share the same administration department

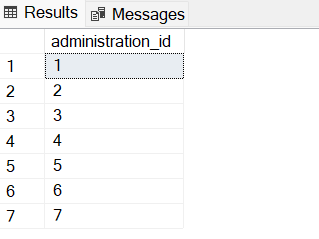
SELECT administration\_id

FROM Employees

INTERSECT

SELECT administration\_id

FROM Supervisors;



--Add a new supplier:

INSERT INTO Suppliers VALUES

(8, 'mahima', '23101030@uap-bd.edu', '01537479');

**7 .CEP Mapping**

Our project is a solution to a complex engineering problem because it can’t be resolved without in depth engineering knowledge. There is no obvious solution to and requires some amount of abstract thinking depending on the Database model. It also involves a diverse group of stakeholders with widely varying needs.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **K’s** | **Attributes** | **How K’s are addressed though our project** | **CO** | **PO** |
| **K3** | Engineering Fundamentals | Our Project involves a systematic formulation of engineering fundamentals like programming code, designing and problem solving. | CO1, CO2, CO3 | PO1, PO2 |
| **K5** | Engineering Design | We’ve made Entity-Relationship (ER) Diagram and Schema Diagram which are part of Engineering design. | CO3, CO4 | PO3, PO2 |
| K6 | Engineering Practice | We’ve implemented our design in MySQL. We created tables, inserted data with DDL and ran queries with DML. We’ve used MySQL workbench software as a tool for implementing our design. | CO1, CO2, CO5 | PO1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **P1** | Depth of knowledge required | The project requires us to study one of the fundamentals of engineering which is Database Management System design **(K3)**. It also requires designing ER and Schema diagram**(k5)** and implementing the design **(K6)**. | CO3,  CO4  CO5 | PO1,  PO2,  PO3,  PO7 |
| **P3** | Depth of analysis required | The designing and implementation differ for various kinds of businesses, so some amount of abstract thinking is necessary to formulate and analyze suitable models. | CO3,  CO5 | PO3 |
| **P7** | Interdependence | Our database has some subproblems or sections. Some of the table are related to orders and shipment of customers, some of the tables are related to orders and shipment of customers, some of the table are related to inventory and storing, some of the table are related to employees and supervisors of different administration, some table are for taxes and transactions. | CO3,  CO4  CO5 | PO2,  PO7 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A’s** | **Attributes** | **How A’s are addressed though are project** | **CO** | **PO** |
| **A1** | Range of resources | We needed various kinds of resources for this project such as Developers (US), Servicers and HDD Storage System (Laptop /Desktop), information about customers and trade licenses. | CO1 | P05 |
| **A5** | Familiarity | We had to get familiar with trade, economics and business operations besides engineering knowledge. | CO8,  C09 | P011 |

**END**