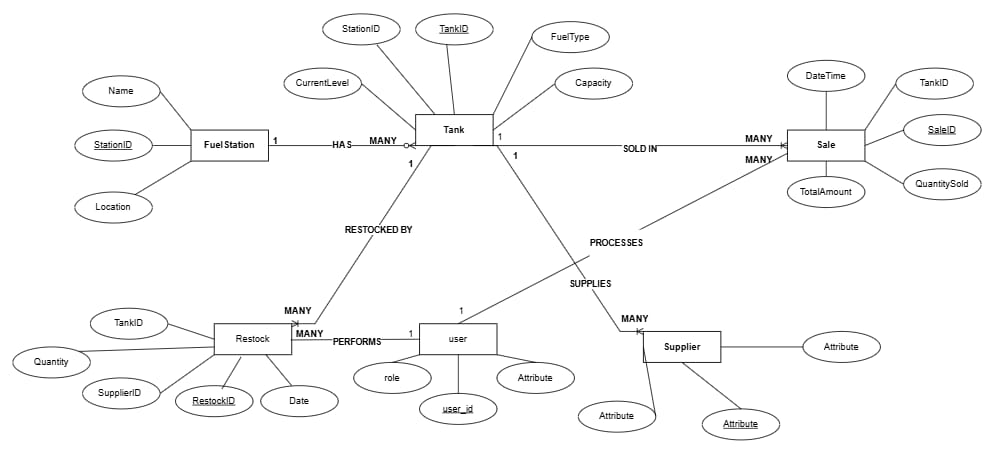
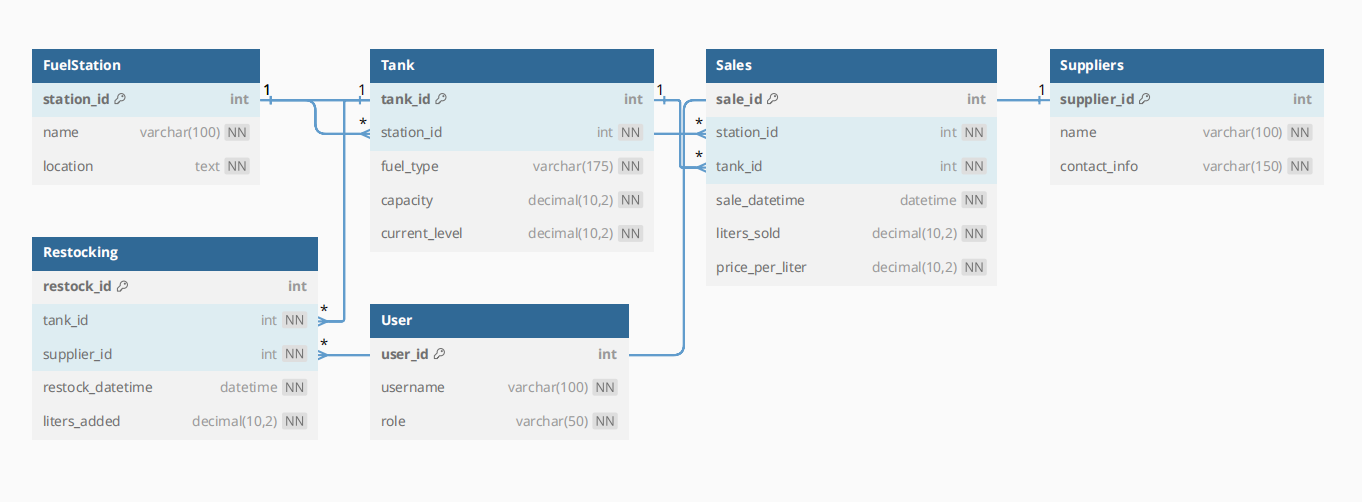
“Fuel Station”  
  
ERD (Entity Relationship Diagram)



Schema  
  


Sql - Script  
  
**1-Tables**

CREATE DATABASE DailyFuel;

USE DailyFuel;

CREATE TABLE FuelStation (

station\_id INT IDENTITY(1,1) PRIMARY KEY,

name VARCHAR(100) NOT NULL UNIQUE,

location TEXT NOT NULL

);

CREATE TABLE Tank (

tank\_id INT IDENTITY(1,1) PRIMARY KEY,

station\_id INT NOT NULL FOREIGN KEY REFERENCES FuelStation(station\_id),

fuel\_type VARCHAR (175) NOT NULL,

capacity DECIMAL(10,2) NOT NULL,

current\_level DECIMAL(10,2) NOT NULL

);

CREATE TABLE Sales (

sale\_id INT IDENTITY(1,1) PRIMARY KEY,

station\_id INT NOT NULL FOREIGN KEY REFERENCES FuelStation(station\_id),

tank\_id INT NOT NULL FOREIGN KEY REFERENCES Tank (tank\_id),

sale\_datetime DATETIME NOT NULL,

liters\_sold DECIMAL(10,2) NOT NULL,

price\_per\_liter DECIMAL(10,2) NOT NULL

);

CREATE TABLE Suppliers (

supplier\_id INT IDENTITY(1,1) PRIMARY KEY,

name VARCHAR(100) NOT NULL,

contact\_info VARCHAR(150) NOT NULL

);

CREATE TABLE Restocking (

restock\_id INT IDENTITY(1,1) PRIMARY KEY,

tank\_id INT NOT NULL FOREIGN KEY REFERENCES Tank (tank\_id),

supplier\_id INT NOT NULL FOREIGN KEY REFERENCES Suppliers(supplier\_id),

restock\_datetime DATETIME NOT NULL,

liters\_added DECIMAL(10,2) NOT NULL

);

CREATE TABLE "User" (

user\_id INT IDENTITY(1,1) PRIMARY KEY,

username VARCHAR(100) NOT NULL UNIQUE,

role VARCHAR(50) NOT NULL

);

**2-Insertion Data**  
-- 1. FuelStation

INSERT INTO FuelStation (name, location) VALUES

('Station A', 'Cairo'),

('Station B', 'Giza'),

('Station C', 'Alexandria'),

('Station D', 'Mansoura'),

('Station E', 'Tanta'),

('Station F', 'Aswan'),

('Station G', 'Luxor'),

('Station H', 'Hurghada'),

('Station I', 'Port Said'),

('Station J', 'Suez');

-- 2. Tank

INSERT INTO Tank (station\_id, fuel\_type, capacity, current\_level) VALUES

(1, 'Diesel', 10000.00, 8000.00),

(1, 'Petrol 92', 8000.00, 5000.00),

(2, 'Petrol 95', 7000.00, 3000.00),

(3, 'Diesel', 12000.00, 10000.00),

(4, 'Petrol 92', 9000.00, 6000.00),

(5, 'Petrol 95', 8500.00, 4000.00),

(6, 'Diesel', 11000.00, 9000.00),

(7, 'Petrol 92', 7500.00, 5000.00),

(8, 'Petrol 95', 8000.00, 6000.00),

(9, 'Diesel', 10000.00, 300.00);

-- 3. Suppliers

INSERT INTO Suppliers (name, contact\_info) VALUES

('Supplier X', 'x@example.com'),

('Supplier Y', 'y@example.com'),

('Supplier Z', 'z@example.com'),

('Supplier Alpha', 'alpha@example.com'),

('Supplier Beta', 'beta@example.com'),

('Supplier Gamma', 'gamma@example.com'),

('Supplier Delta', 'delta@example.com'),

('Supplier Epsilon', 'epsilon@example.com'),

('Supplier Eta', 'eta@example.com'),

('Supplier Theta', 'theta@example.com');

-- 4. Users

INSERT INTO "User" (username, role) VALUES

('admin', 'Admin'),

('manager1', 'Manager'),

('manager2', 'Manager'),

('cashier1', 'Cashier'),

('cashier2', 'Cashier'),

('supervisor1', 'Supervisor'),

('auditor1', 'Auditor'),

('operator1', 'Operator'),

('operator2', 'Operator'),

('guest', 'Guest');

-- 5. Sales

INSERT INTO Sales (station\_id, tank\_id, sale\_datetime, liters\_sold, price\_per\_liter) VALUES

(1, 1, GETDATE(), 100.00, 8.50),

(1, 2, GETDATE(), 150.00, 9.25),

(2, 3, GETDATE(), 200.00, 10.00),

(3, 4, GETDATE(), 120.00, 8.75),

(4, 5, GETDATE(), 180.00, 9.50),

(5, 6, GETDATE(), 220.00, 10.20),

(6, 7, GETDATE(), 160.00, 8.60),

(7, 8, GETDATE(), 190.00, 9.30),

(8, 9, GETDATE(), 210.00, 10.10),

(9, 10, GETDATE(), 170.00, 8.80);

-- 6. Restocking

INSERT INTO Restocking (tank\_id, supplier\_id, restock\_datetime, liters\_added) VALUES

(1, 1, GETDATE(), 2000.00),

(2, 2, GETDATE(), 1500.00),

(3, 3, GETDATE(), 1800.00),

(4, 4, GETDATE(), 2200.00),

(5, 5, GETDATE(), 1700.00),

(6, 6, GETDATE(), 1600.00),

(7, 7, GETDATE(), 2000.00),

(8, 8, GETDATE(), 1500.00),

(9, 9, GETDATE(), 1800.00),

(10, 10, GETDATE(), 2200.00);

3-Queries

SELECT \* FROM FuelStation;

SELECT \* FROM Tank;

SELECT

T.tank\_id,

F.name AS station\_name,

T.fuel\_type,

T.capacity,

T.current\_level

FROM Tank T

JOIN FuelStation F ON T.station\_id = F.station\_id;

SELECT

tank\_id,

fuel\_type,

current\_level,

capacity,

(current\_level / capacity) \* 100 AS percent\_full

FROM Tank

WHERE current\_level < (capacity \* 0.2);

SELECT

F.name AS station\_name,

SUM(S.liters\_sold \* S.price\_per\_liter) AS total\_sales

FROM Sales S

JOIN FuelStation F ON S.station\_id = F.station\_id

GROUP BY F.name;

SELECT

CONVERT(DATE, sale\_datetime) AS sale\_date,

T.fuel\_type,

SUM(liters\_sold) AS total\_liters\_sold

FROM Sales S

JOIN Tank T ON S.tank\_id = T.tank\_id

GROUP BY CONVERT(DATE, sale\_datetime), T.fuel\_type

ORDER BY sale\_date;

SELECT

R.tank\_id,

T.fuel\_type,

SUM(R.liters\_added) AS total\_restocked

FROM Restocking R

JOIN Tank T ON R.tank\_id = T.tank\_id

GROUP BY R.tank\_id, T.fuel\_type;

SELECT

R.tank\_id,

MAX(R.restock\_datetime) AS last\_restock

FROM Restocking R

GROUP BY R.tank\_id;

SELECT \* FROM "User"

ORDER BY role;

-- ================================

-- DDL – Data Definition Language

-- ================================

ALTER TABLE Suppliers ADD is\_active BIT DEFAULT 1;

ALTER TABLE Tank ALTER COLUMN fuel\_type VARCHAR(100);

-- ================================

-- DML – Data Manipulation Language

-- ================================

UPDATE Tank SET current\_level = current\_level - 100 WHERE tank\_id = 1;

INSERT INTO Sales (station\_id, tank\_id, sale\_datetime, liters\_sold, price\_per\_liter)

VALUES (1, 1, GETDATE(), 120, 8.75);

DELETE FROM Restocking WHERE restock\_id = 1;

SELECT name, location FROM FuelStation;

-- ================================

-- DCL – Data Control Language

-- ================================

GRANT SELECT ON FuelStation TO guest;

REVOKE INSERT, UPDATE, DELETE ON Sales FROM guest;

-- ================================

-- TCL – Transaction Control Language

-- ================================

BEGIN TRANSACTION;

INSERT INTO Sales (station\_id, tank\_id, sale\_datetime, liters\_sold, price\_per\_liter)

VALUES (2, 3, GETDATE(), 100, 9.50);

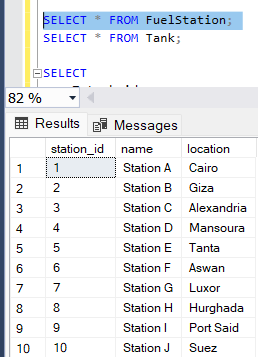
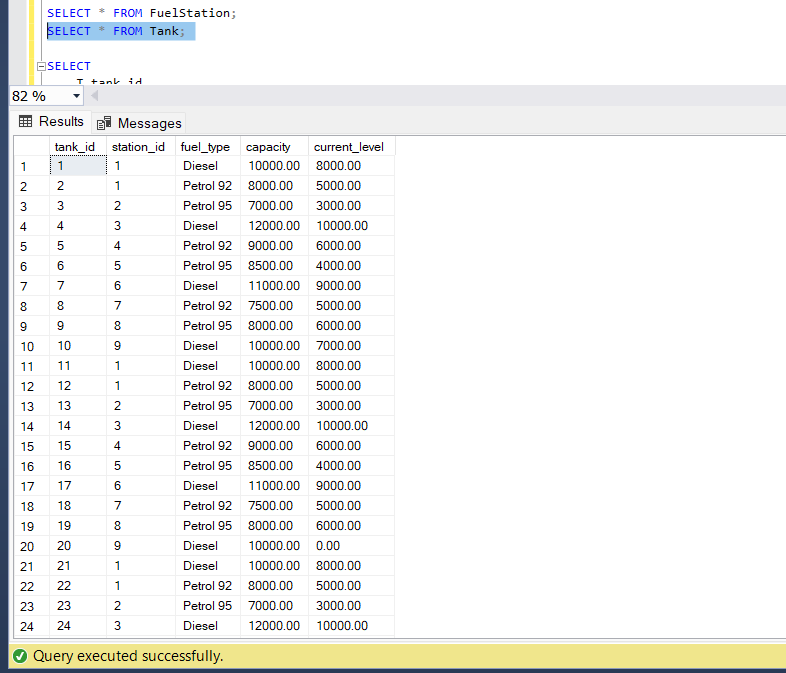
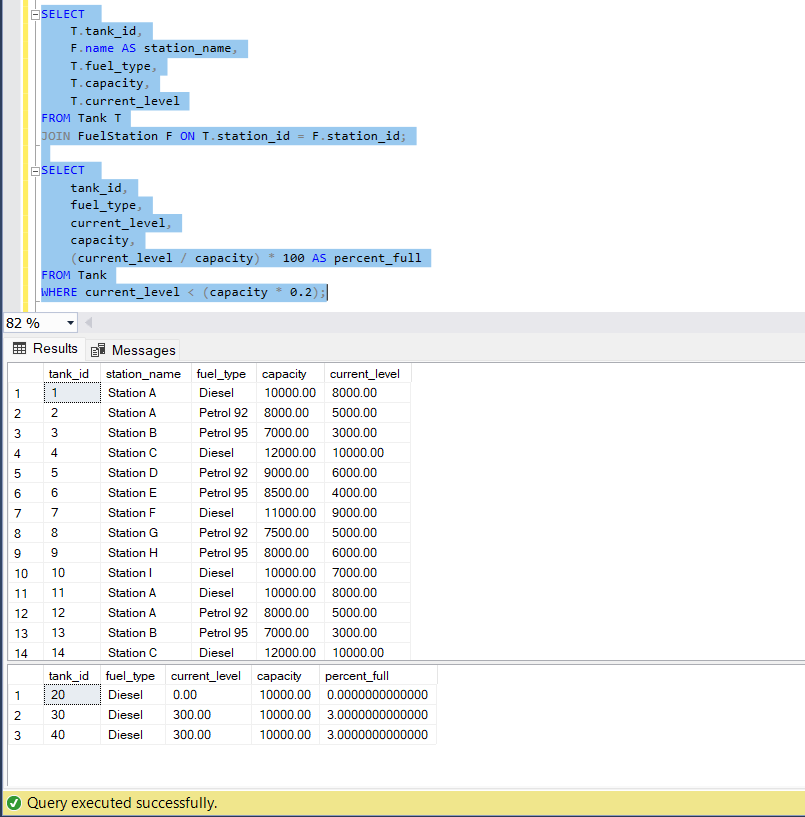
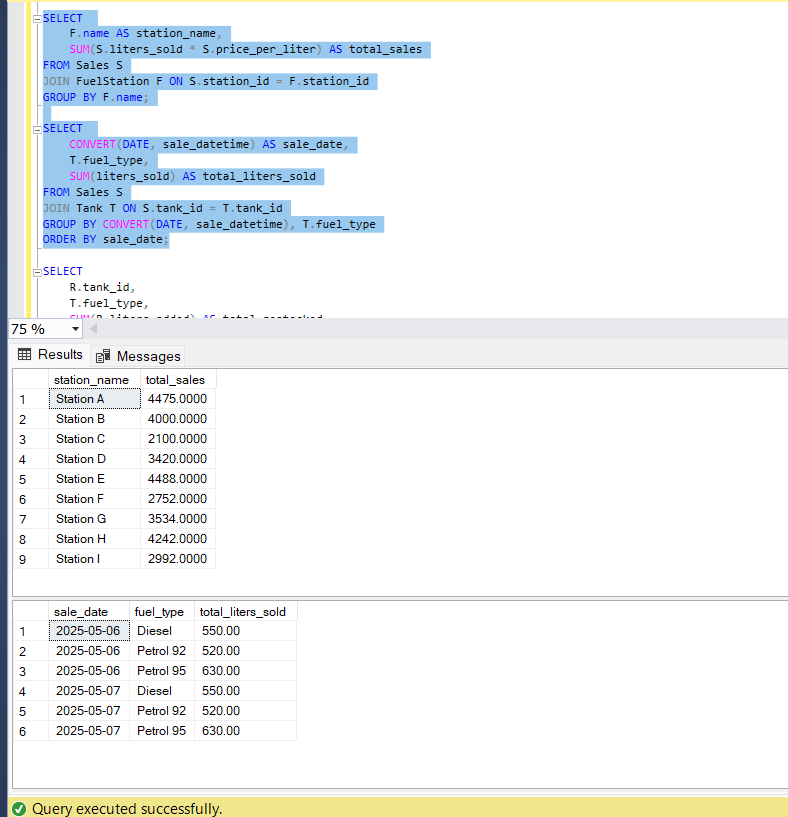
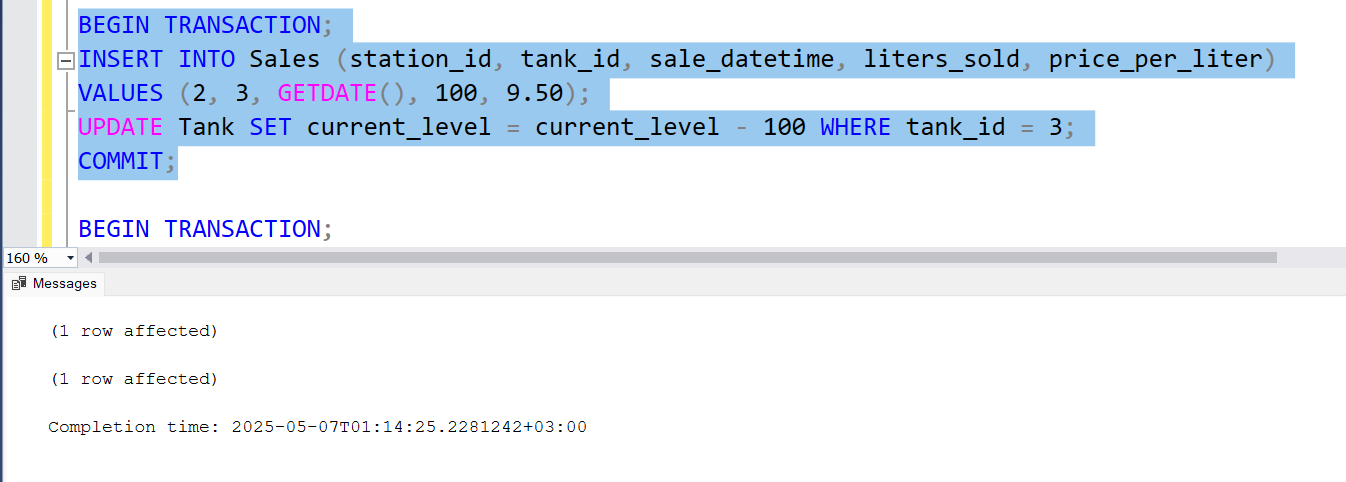
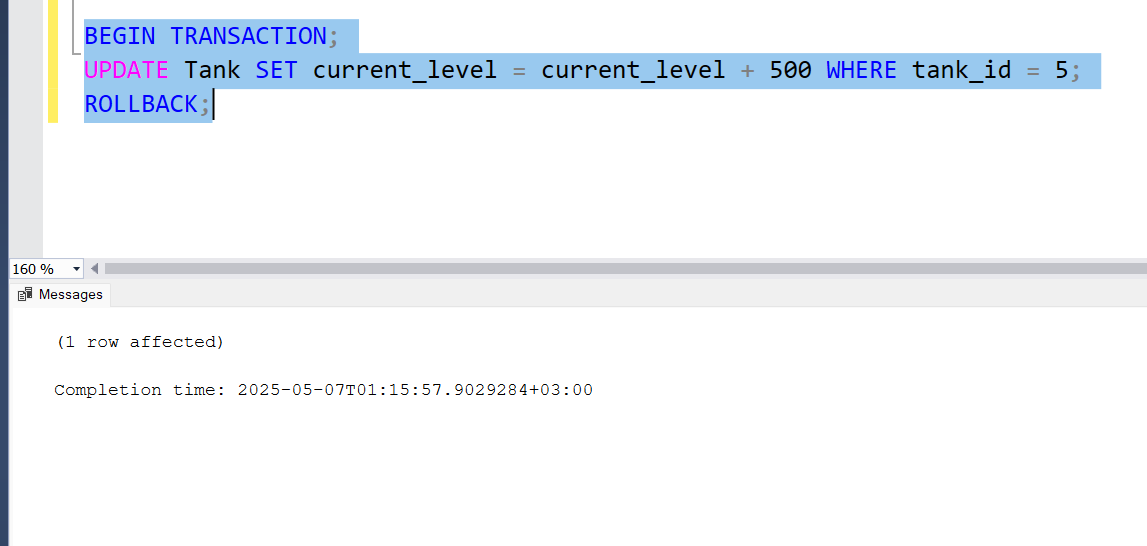
UPDATE Tank SET current\_level = current\_level - 100 WHERE tank\_id = 3;

COMMIT;

BEGIN TRANSACTION;

DELETE FROM Tank WHERE tank\_id = 5;

ROLLBACK;

Answer sql Question  
  


|  |  |  |
| --- | --- | --- |
| Name | ID | Grade |
| Mohamed Ahmed | 692300194 |  |
| Ahmed Maged | 692300194 |  |
| Ziad Nasr | 692300194 |  |
| Ahmed Mohamed | 692300194 |  |