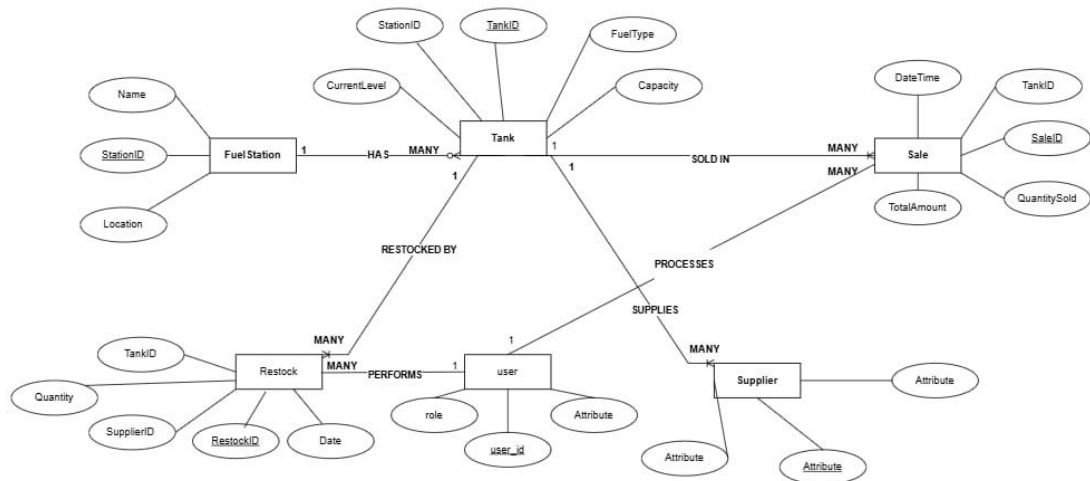
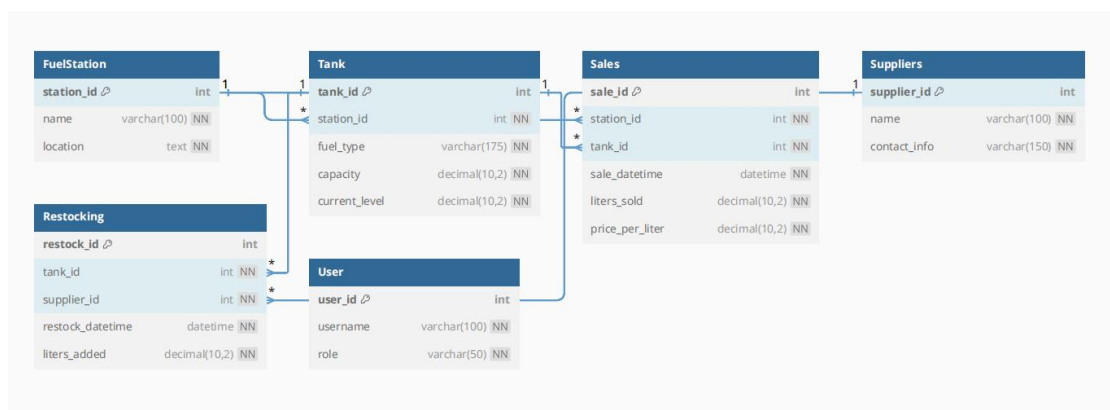


“Fuel Station”

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;

```

```
SELECT * FROM FuelStation;
SELECT * FROM Tank;
```

SELECT

82 %

	station_id	name	location
1	1	Station A	Cairo
2	2	Station B	Giza
3	3	Station C	Alexandria
4	4	Station D	Mansoura
5	5	Station E	Tanta
6	6	Station F	Aswan
7	7	Station G	Luxor
8	8	Station H	Hurghada
9	9	Station I	Port Said
10	10	Station J	Suez

```
SELECT * FROM FuelStation;
SELECT * FROM Tank;
```

SELECT

82 %

	tank_id	station_id	fuel_type	capacity	current_level
1	1	1	Diesel	10000.00	8000.00
2	2	1	Petrol 92	8000.00	5000.00
3	3	2	Petrol 95	7000.00	3000.00
4	4	3	Diesel	12000.00	10000.00
5	5	4	Petrol 92	9000.00	6000.00
6	6	5	Petrol 95	8500.00	4000.00
7	7	6	Diesel	11000.00	9000.00
8	8	7	Petrol 92	7500.00	5000.00
9	9	8	Petrol 95	8000.00	6000.00
10	10	9	Diesel	10000.00	7000.00
11	11	1	Diesel	10000.00	8000.00
12	12	1	Petrol 92	8000.00	5000.00
13	13	2	Petrol 95	7000.00	3000.00
14	14	3	Diesel	12000.00	10000.00
15	15	4	Petrol 92	9000.00	6000.00
16	16	5	Petrol 95	8500.00	4000.00
17	17	6	Diesel	11000.00	9000.00
18	18	7	Petrol 92	7500.00	5000.00
19	19	8	Petrol 95	8000.00	6000.00
20	20	9	Diesel	10000.00	0.00
21	21	1	Diesel	10000.00	8000.00
22	22	1	Petrol 92	8000.00	5000.00
23	23	2	Petrol 95	7000.00	3000.00
24	24	3	Diesel	12000.00	10000.00

Query executed successfully.

```

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);

```

82 %

Results Messages

	tank_id	station_name	fuel_type	capacity	current_level
1	1	Station A	Diesel	10000.00	8000.00
2	2	Station A	Petrol 92	8000.00	5000.00
3	3	Station B	Petrol 95	7000.00	3000.00
4	4	Station C	Diesel	12000.00	10000.00
5	5	Station D	Petrol 92	9000.00	6000.00
6	6	Station E	Petrol 95	8500.00	4000.00
7	7	Station F	Diesel	11000.00	9000.00
8	8	Station G	Petrol 92	7500.00	5000.00
9	9	Station H	Petrol 95	8000.00	6000.00
10	10	Station I	Diesel	10000.00	7000.00
11	11	Station A	Diesel	10000.00	8000.00
12	12	Station A	Petrol 92	8000.00	5000.00
13	13	Station B	Petrol 95	7000.00	3000.00
14	14	Station C	Diesel	12000.00	10000.00

	tank_id	fuel_type	current_level	capacity	percent_full
1	20	Diesel	0.00	10000.00	0.00000000000000
2	30	Diesel	300.00	10000.00	3.00000000000000
3	40	Diesel	300.00	10000.00	3.00000000000000

✓ Query executed successfully.

```

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(liters_sold) AS total_liters_sold

```

75 %

Results Messages

	station_name	total_sales
1	Station A	4475.0000
2	Station B	4000.0000
3	Station C	2100.0000
4	Station D	3420.0000
5	Station E	4488.0000
6	Station F	2752.0000
7	Station G	3534.0000
8	Station H	4242.0000
9	Station I	2992.0000

	sale_date	fuel_type	total_liters_sold
1	2025-05-06	Diesel	550.00
2	2025-05-06	Petrol 92	520.00
3	2025-05-06	Petrol 95	630.00
4	2025-05-07	Diesel	550.00
5	2025-05-07	Petrol 92	520.00
6	2025-05-07	Petrol 95	630.00

✓ Query executed successfully.

```

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;

```

160 %

Messages

(1 row affected)

(1 row affected)

Completion time: 2025-05-07T01:14:25.2281242+03:00


```
BEGIN TRANSACTION;
UPDATE Tank SET current_level = current_level + 500 WHERE tank_id = 5;
ROLLBACK;
```

160 %
Messages

(1 row affected)

Completion time: 2025-05-07T01:15:57.9029284+03:00

Answer sql Question

```
--Answer The Questions
--1- Retrieve a list of all records from a main table in this project.
SELECT * FROM Sales;
--2- How many entries are related to a certain condition? (Use COUNT or WHERE)
SELECT COUNT(*) AS GasolineSalesCount
FROM Sales
JOIN Tank ON Sales.tank_id = Tank.tank_id
WHERE Tank.fuel_type = 'Gasoline';
--3- Use JOIN to relate at least two tables and show combined data.
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;
--4- Write a DCL query to grant or revoke access to a table.
GRANT SELECT ON Sales TO [cashier1];
--OR--
REVOKE SELECT ON Sales FROM [cashier1];
--5- Show how to use BEGIN TRANSACTION and COMMIT for a key action.
BEGIN TRANSACTION;

INSERT INTO Sales (station_id, tank_id, sale_datetime, liters_sold, price_per_liter)
VALUES (1, 2, GETDATE(), 50.0, 9.5);

UPDATE Tank
SET current_level = current_level - 50.0
WHERE tank_id = 2;

COMMIT;
```

Name	ID	Grade
Mohamed Ahmed	692300194	
Ahmed Maged	692300194	

Ziad Nasr	692300194	
Ahmed Mohamed	692300194	