

**INTRODUCTION TO DATABASE SYSTEMS LAB**

**Name: Rabail Fiaz**

**Reg. No: L1F22BSCS1096**

**Section: D-16**

**Project: Fuel Pump Management System**

**Date of Submission: 19-04-2024**

**Submitted To: Sir Afham Nazir**

***Phase-01***

**Description:**

The Fuel Pump Management System streamlines gas station operations by tracking pump details, sales transactions, and customer information. It ensures fuel availability and maintains equipment through maintenance records. Employees are associated with transactions and maintenance tasks, while suppliers provide fuel and maintenance services. Fuel types are categorized for efficient management, linking pumps, tank refills, and suppliers. This integrated system enhances operational efficiency, customer service, and inventory control, optimizing the overall management of the gas station.

**Entities:**

1. **Fuel pump**

|  |  |  |
| --- | --- | --- |
| Attributes | Datatype |  |
| Pump\_ID | Int | Primary key |
| Location | Varchar |  |
| Capacity | Float |  |
| Fuel\_type | Char |  |
| Status | char |  |

1. **Transaction**

|  |  |  |
| --- | --- | --- |
| Attributes | Datatype |  |
| Transaction\_ID | Int | Primary key |
| Date time | Date |  |
| Fuel Amount | Float |  |
| Payment\_method | Char |  |

1. **Customer**

|  |  |  |
| --- | --- | --- |
| Attributes | Datatype |  |
| Customer\_ID | Int | Primary key |
| Name | char |  |
| Contact Number | int |  |
| Email | Varchar |  |
| Address | Varchar |  |

1. **Employee**

|  |  |  |
| --- | --- | --- |
| Attributes | Datatype |  |
| Employee\_ID | Int | Primary key |
| Name | char |  |
| Contact Number | int |  |
| Email | Varchar |  |
| Position | Varchar |  |

1. **Fuel Tank**

|  |  |  |
| --- | --- | --- |
| Attributes | Datatype |  |
| Tank\_ID | Int | Primary key |
| Location | char |  |
| Capacity | float |  |
| Fuel\_type | Char |  |
| Current\_level | Float |  |

1. **Payment**

|  |  |  |
| --- | --- | --- |
| Attributes | Datatype |  |
| Payment\_ID | Int | Primary key |
| Amount | float |  |
| Payment method | char |  |
| Date\_Time | DateTime |  |

1. **Fuel Type**

|  |  |  |
| --- | --- | --- |
| Attributes | Datatype |  |
| Fuel\_Type\_ID | Int | Primary key |
| Type\_Name | char |  |

1. **Tank Refill**

|  |  |  |
| --- | --- | --- |
| Attributes | Datatype |  |
| Refill\_ID | Int | Primary key |
| Date\_Time | DateTime |  |
| Refill\_Amount | Float |  |

1. **Maintenance Record**

|  |  |  |
| --- | --- | --- |
| Attributes | Datatype |  |
| Maintenance \_ID | Int | Primary key |
| Date\_Time | DateTime |  |
| Description | Varchar |  |
| Cost | Float |  |

1. **Supplier**

|  |  |  |
| --- | --- | --- |
| Attributes | Datatype |  |
| 1. Supplier \_ID | Int | Primary key |
| Name | char |  |
| Contact Number | int |  |
| Email | Varchar |  |
| Address | Varchar |  |

**Relationships**

1. **Fuel Pump and Transactions:**

One-to-Many with Transaction: Each fuel pump can have multiple transactions.

1. **Fuel Pump and Maintenance Records:**

One-to-One with Maintenance Record: Each fuel pump can have multiple maintenance records.

1. **Transaction and Fuel Pump**:

Many-to-One with Fuel Pump: Many transactions can be associated with one fuel pump.

1. **Transaction and Customer**:

Many-to-One with Customer: Many transactions can be associated with one customer.

1. **Transaction and Payment**:

One-to-One with Payment: Each transaction has one payment associated with it.

1. **Transaction and Fuel Type:**

Many-to-One with Fuel Type: Each transaction involves a specific fuel type.

1. **Customer and Transaction**:

One-to-Many with Transaction: Each customer can have multiple transactions.

1. **Employee and Transaction**:

One-to-Many with Transaction: An employee can handle multiple transactions.

1. **Employee and Maintenance Record:**

One-to-Many with Maintenance Record: An employee can be associated with multiple maintenance records.

1. **Supplier and Tank Refill:**

One-to-Many with Tank Refill: A supplier can provide fuel for multiple tank refills.

1. **Supplier and Maintenance Record:**

One-to-Many with Maintenance Record: A supplier can be associated with multiple maintenance records (e.g., supplying parts or services).

1. **Supplier and Fuel Type**:

One-to-Many with Fuel Type: A supplier can provide multiple types of fuel.

1. **Fuel Tank and Tank Refill**:

One-to-Many with Tank Refill: Each fuel tank can have multiple tank refill records.

1. **Payment and Transaction**:

One-to-One with Transaction: Each payment is associated with one transaction.

1. **Fuel Type and Fuel Pump**:

One-to-Many with Fuel Pump: Multiple fuel pumps can dispense the same type of fuel.

1. **Fuel Type and Tank Refill:**

One-to-Many with Tank Refill: The same type of fuel can be used to refill multiple tanks.

1. **Tank Refill and Fuel Tank**:

Many-to-One with Fuel Tank: Many tank refills can be associated with one fuel tank.

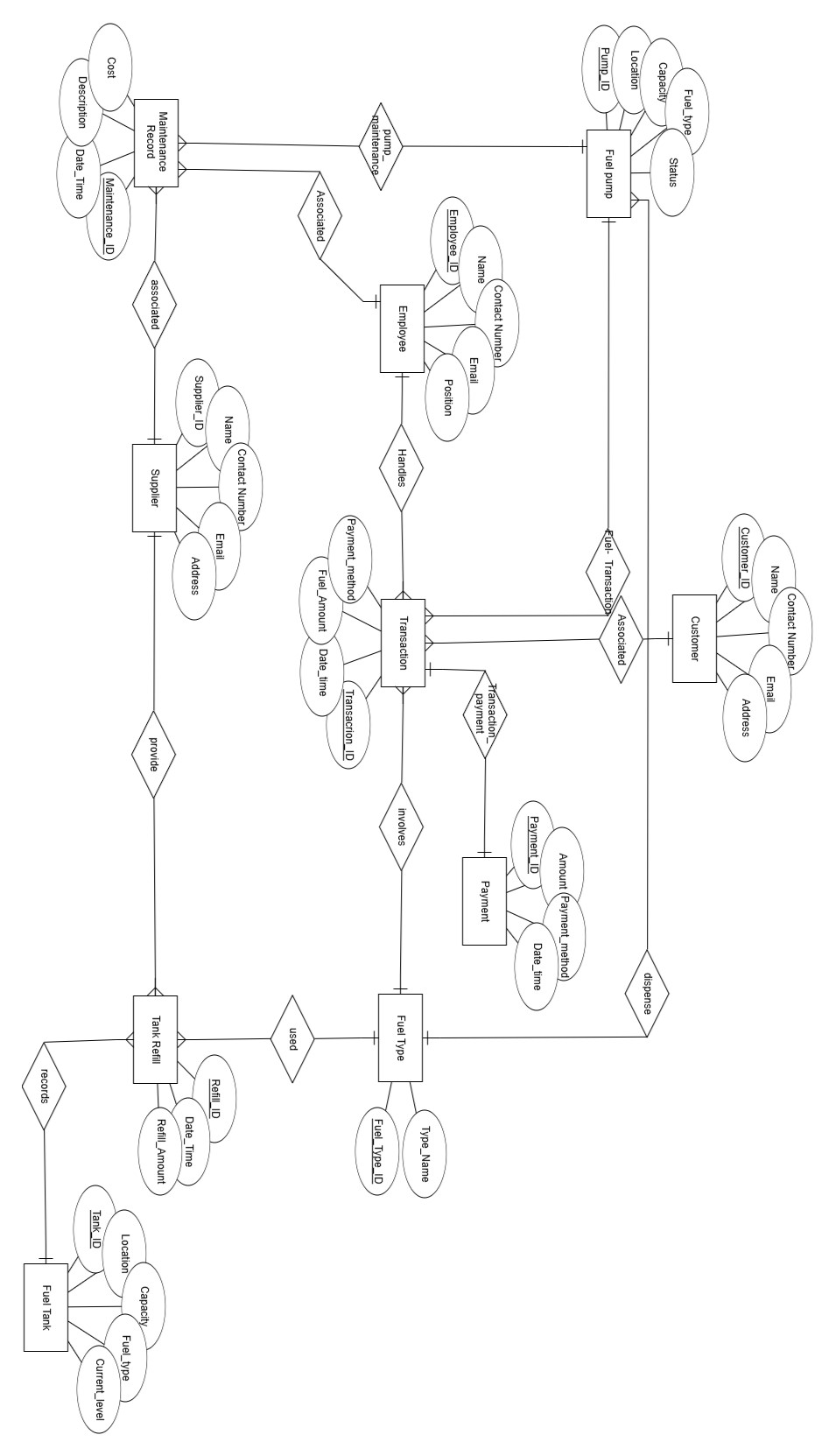
1. **Maintenance Record and Fuel Pump**:

Many-to-One with Fuel Pump: Many maintenance records can be associated with one fuel pump.

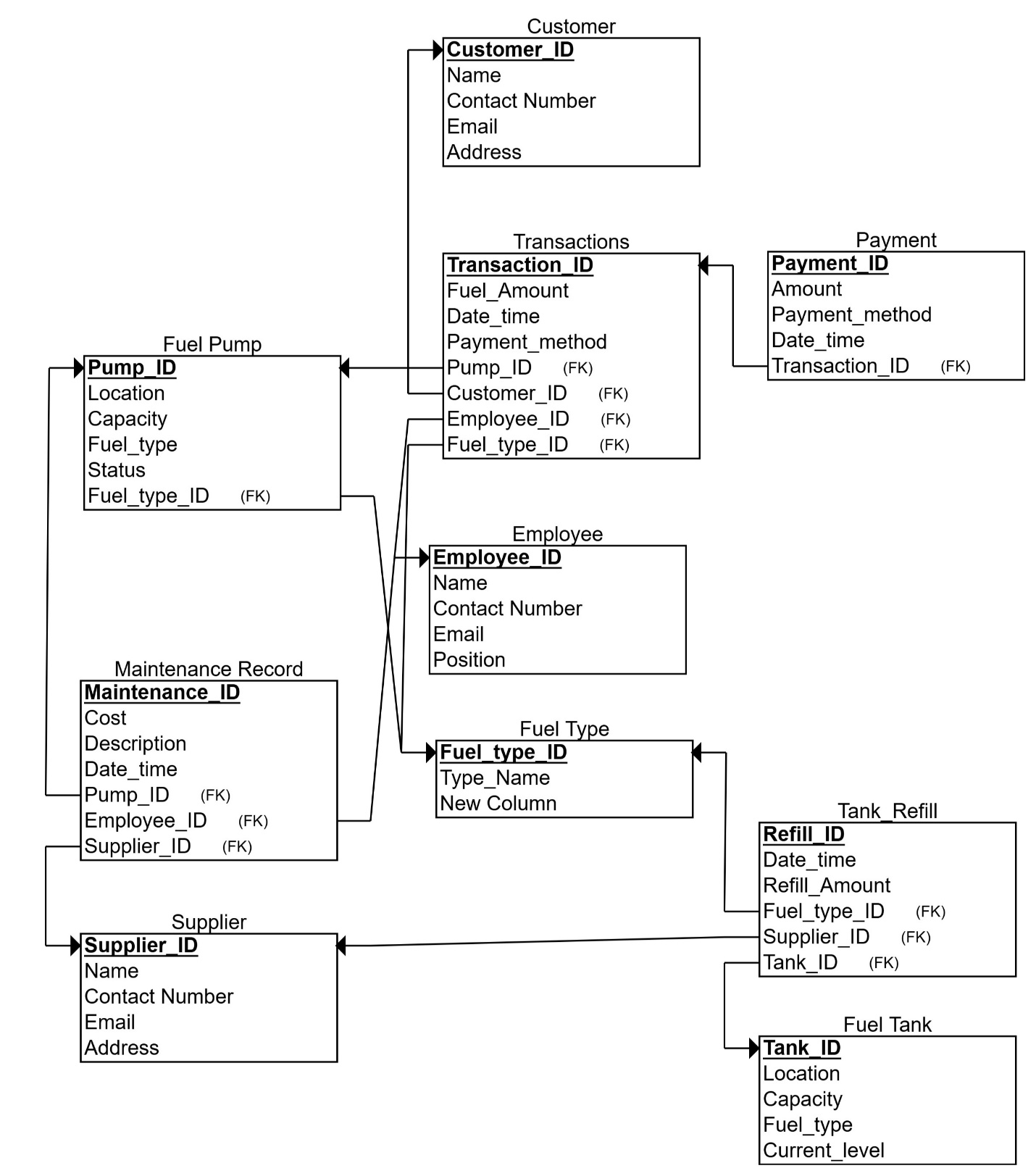
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

***Phase-02***

**ERD:**



**RELATIONAL SCHEMA:**



***Phase-03***

**“DDL”**

1. **Insertion:**

create database project;

use project;

create table Fuel\_pump(

Pump\_ID int(5),

Location varchar (20),

Capacity float(20),

Fuel\_type char(10),

status char (20)

);

create table Transaction(

Transaction\_ID int(5),

Date\_time Date,

Fuel\_Amount float(20),

Payment\_method char(10)

);

create table customer(

Customer\_ID int(5),

Name char(20),

Contact\_Number int(50),

Email varchar(25),

Address varchar(20)

);

create table Employee(

Employee\_ID int(5),

Name char(20),

Contact\_Number int(50),

Email varchar(25),

Position varchar(10)

);

create table Fuel\_Tank(

Tank\_ID int(5),

Location char(20),

Capacity float(20),

Fuel\_Type char(10),

Current\_level float(20)

);

create table Payment(

Payment\_ID int(5),

Amount float(20),

Payment\_method char(10),

Date\_time datetime

);

create table Fuel\_type(

Fuel\_type\_ID int(5),

Type\_name char(10)

);

create table Tank\_Refill(

Refill\_ID int(5),

Date\_time datetime,

Refill\_Amount float(10)

);

create table Maintenance\_Record(

Maintenance\_ID int(5),

Date\_time datetime,

Description varchar(20),

Cost float(20)

);

create table Supplier(

Supplier\_ID int(5),

Name char(10),

Contact\_Number int(50),

Email varchar(25),

Address varchar(20)

);

insert into Fuel\_pump (Pump\_ID, Location, Capacity, Fuel\_type, Status) values

(1, 'Location\_1', 892.95, 'Diesel', 'Inactive'),

(2, 'Location\_2', 695.61, 'Petrol', 'Inactive'),

(3, 'Location\_3', 704.34, 'Petrol', 'Inactive'),

(4, 'Location\_4', 701.52, 'Gas', 'Inactive'),

(5, 'Location\_5', 828.62, 'Diesel', 'Active'),

(6, 'Location\_6', 544.03, 'Gas', 'Active'),

(7, 'Location\_7', 952.81, 'Gas', 'Active'),

(8, 'Location\_8', 615.62, 'Petrol', 'Active'),

(9, 'Location\_9', 584.34, 'Petrol', 'Inactive'),

(10, 'Location\_10', 831.40, 'Diesel', 'Inactive');

select\*from Fuel\_pump;

insert into Transaction (Transaction\_ID, Date\_time, Fuel\_Amount, Payment\_method) values

(1, '2023-01-22', 86.40, 'Cash'),

(2, '2023-09-01', 96.64, 'Cash'),

(3, '2023-11-23', 82.48, 'Card'),

(4, '2023-10-09', 65.46, 'Cash'),

(5, '2023-05-10', 17.39, 'Card'),

(6, '2023-08-07', 82.46, 'Cash'),

(7, '2023-07-01', 41.96, 'Cash'),

(8, '2023-10-25', 31.30, 'Card'),

(9, '2023-08-08', 45.02, 'Cash'),

(10, '2023-10-02', 61.36, 'Card');

select\*from Transaction;

insert into Customer (Customer\_ID, Name, Contact\_Number, Email, Address) values

(1, 'Customer\_1', 619993791, 'email1@example.com', 'Address\_1'),

(2, 'Customer\_2', 356731234, 'email2@example.com', 'Address\_2'),

(3, 'Customer\_3', 578755580, 'email3@example.com', 'Address\_3'),

(4, 'Customer\_4', 626374693, 'email4@example.com', 'Address\_4'),

(5, 'Customer\_5', 722520902, 'email5@example.com', 'Address\_5'),

(6, 'Customer\_6', 343480754, 'email6@example.com', 'Address\_6'),

(7, 'Customer\_7', 833398957, 'email7@example.com', 'Address\_7'),

(8, 'Customer\_8', 430518438, 'email8@example.com', 'Address\_8'),

(9, 'Customer\_9', 732227830, 'email9@example.com', 'Address\_9'),

(10, 'Customer\_10',436676712, 'email10@example.com', 'Address\_10');

select\*from customer;

insert into Employee (Employee\_ID, Name, Contact\_Number, Email, Position) values

(1, 'Employee\_1', 270863327, 'email1@company.com', 'Clerk'),

(2, 'Employee\_2', 133568672, 'email2@company.com', 'Manager'),

(3, 'Employee\_3', 191005173, 'email3@company.com', 'Clerk'),

(4, 'Employee\_4', 151017849, 'email4@company.com', 'Attendant'),

(5, 'Employee\_5', 254715837, 'email5@company.com', 'Clerk'),

(6, 'Employee\_6', 130657187, 'email6@company.com', 'Attendant'),

(7, 'Employee\_7', 030846552, 'email7@company.com', 'Manager'),

(8, 'Employee\_8',857032973, 'email8@company.com', 'Manager'),

(9, 'Employee\_9', 856267391, 'email9@company.com', 'Clerk'),

(10, 'Employee\_10', 933766050, 'email10@company.com', 'Manager');

select\*from Employee;

insert into Fuel\_Tank (Tank\_ID, Location, Capacity, Fuel\_Type, Current\_level) values

(1, 'Location\_1', 4851.44, 'Petrol', 2110.15),

(2, 'Location\_2', 4033.89, 'Gas', 1053.55),

(3, 'Location\_3', 2924.48, 'Diesel', 456.83),

(4, 'Location\_4', 1190.37, 'Diesel', 509.62),

(5, 'Location\_5', 3762.69, 'Gas', 1859.54),

(6, 'Location\_6', 4126.03, 'Petrol', 1933.62),

(7, 'Location\_7', 2455.72, 'Petrol', 2425.74),

(8, 'Location\_8', 2843.82, 'Diesel', 1964.91),

(9, 'Location\_9', 2781.35, 'Gas', 1488.78),

(10, 'Location\_10', 2673.50, 'Petrol', 493.72);

select\*from Fuel\_Tank;

insert into Payment (Payment\_ID, Amount, Payment\_method, Date\_time) values

(1, 203.72, 'Cash', '2023-04-19 08:17:25'),

(2, 255.48, 'Card', '2023-06-08 14:43:15'),

(3, 73.98, 'Cash', '2023-10-28 05:01:11'),

(4, 276.20, 'Cash', '2023-02-13 10:30:47'),

(5, 475.72, 'Card', '2023-01-27 03:04:48'),

(6, 109.80, 'Card', '2023-11-03 11:32:01'),

(7, 438.24, 'Card', '2023-03-18 02:09:54'),

(8, 64.86, 'Cash', '2023-07-13 00:06:52'),

(9, 390.33, 'Cash', '2023-04-11 23:19:07'),

(10, 199.71, 'Card', '2023-09-16 00:51:57');

select\*from Payment;

insert into Fuel\_type (Fuel\_type\_ID, Type\_name) values

(1, 'Gas'),

(2, 'Petrol'),

(3, 'Petrol'),

(4, 'Petrol'),

(5, 'Diesel'),

(6, 'Gas'),

(7, 'Gas'),

(8, 'Diesel'),

(9, 'Diesel'),

(10, 'Gas');

select\*from Fuel\_type;

insert into Tank\_Refill (Refill\_ID, Date\_time, Refill\_Amount) values

(1, '2023-11-25 19:00:37', 117.12),

(2, '2023-04-04 12:49:12', 567.15),

(3, '2023-08-12 10:21:25', 527.34),

(4, '2023-01-31 03:57:15', 406.93),

(5, '2023-08-23 01:37:21', 433.05),

(6, '2023-04-15 10:46:04', 468.33),

(7, '2023-01-14 22:28:51', 798.16),

(8, '2023-07-25 13:23:40', 112.49),

(9, '2023-06-04 17:25:58', 766.26),

(10, '2023-03-09 16:08:47', 637.31);

select\*from Tank\_Refill;

insert into Maintenance\_Record (Maintenance\_ID, Date\_time, Description, Cost) values

(1, '2023-09-01 19:00:02', 'Description\_1', 909.85),

(2, '2023-11-22 04:33:49', 'Description\_2', 310.92),

(3, '2023-07-19 02:45:46', 'Description\_3', 495.51),

(4, '2023-05-18 07:44:41', 'Description\_4', 678.57),

(5, '2023-04-27 21:28:46', 'Description\_5', 586.72),

(6, '2023-09-14 11:32:51', 'Description\_6', 864.85),

(7, '2023-12-19 15:29:04', 'Description\_7', 498.55),

(8, '2023-06-19 13:28:17', 'Description\_8', 273.12),

(9, '2023-11-30 06:06:25', 'Description\_9', 364.54),

(10, '2023-04-18 15:47:51', 'Description\_10', 831.34);

select\*from Maintenance\_Record;

insert into Supplier (Supplier\_ID, Name, Contact\_Number, Email, Address) values

(1, 'Supplier\_1', 518149973, 'supplier1@mail.com', 'Address\_1'),

(2, 'Supplier\_2', 745876443, 'supplier2@mail.com', 'Address\_2'),

(3, 'Supplier\_3', 209385859, 'supplier3@mail.com', 'Address\_3'),

(4, 'Supplier\_4', 755352322, 'supplier4@mail.com', 'Address\_4'),

(5, 'Supplier\_5', 839647875, 'supplier5@mail.com', 'Address\_5'),

(6, 'Supplier\_6', 608036272, 'supplier6@mail.com', 'Address\_6'),

(7, 'Supplier\_7', 890998849, 'supplier7@mail.com', 'Address\_7'),

(8, 'Supplier\_8', 421819594, 'supplier8@mail.com', 'Address\_8'),

(9, 'Supplier\_9', 958091141, 'supplier9@mail.com', 'Address\_9'),

(10, 'Suppler\_10', 66812936, 'supplier10@mail.com', 'Address\_10');

select\*from Supplier;

1. **Primary keys:**

alter table Fuel\_pump

add primary key (Pump\_ID);

alter table Transaction

add primary key (Transaction\_ID);

alter table customer

add primary key (customer\_ID);

alter table Employee

add primary key (Employee\_ID);

alter table Fuel\_Tank

add primary key (Tank\_ID);

alter table Payment

add primary key (Payment\_ID);

alter table Fuel\_type

add primary key (Fuel\_type\_ID);

alter table Tank\_Refill

add primary key (Refill\_ID);

alter table Maintenance\_Record

add primary key (Maintenance\_ID);

alter table Supplier

add primary key (Supplier\_ID);

1. **Foriegn keys:**

-- Transaction and Fuel Pump: Many-to-One with Fuel Pump

alter table Transaction

add foreign key (Pump\_ID) references Fuel\_pump(Pump\_ID);

-- Transaction and Customer: Many-to-One with Customer

alter table Transaction

add foreign key (Customer\_ID) references Customer(Customer\_ID);

-- Transaction and Payment: One-to-One with Payment

alter table Payment

add foreign key (Transaction\_ID) references Transaction(Transaction\_ID);

-- Transaction and Fuel Type: Many-to-One with Fuel Type

alter table Transaction

add foreign key (Fuel\_Type\_ID) references Fuel\_type(Fuel\_type\_ID);

-- Employee and Transaction: One-to-Many with Transaction

ALTER TABLE Transaction

ADD Employee\_ID INT(5),

ADD FOREIGN KEY (Employee\_ID) REFERENCES Employee(Employee\_ID);

-- Employee and Maintenance Record: One-to-Many with Maintenance Record

alter table Maintenance\_Record

ADD M\_Employee\_ID INT(5),

add foreign key (M\_Employee\_ID) references Employee(Employee\_ID);

-- Supplier and Tank Refill: One-to-Many with Tank Refill

alter table Tank\_Refill

add foreign key (Supplier\_ID) references Supplier(Supplier\_ID);

-- Supplier and Maintenance Record: One-to-Many with Maintenance Record

ALTER TABLE Maintenance\_Record

ADD M\_Supplier\_ID INT(5),

ADD FOREIGN KEY (M\_Supplier\_ID) REFERENCES Supplier(Supplier\_ID);

-- Supplier and Fuel Type: One-to-Many with Fuel Type

alter table Fuel\_type

ADD Supplier\_ID INT(5),

add foreign key (Supplier\_ID) references Supplier(Supplier\_ID);

-- Fuel Tank and Tank Refill: One-to-Many with Tank Refill

alter table Tank\_Refill

add foreign key (Tank\_ID) references Fuel\_Tank(Tank\_ID);

-- Fuel Type and Fuel Pump: One-to-Many with Fuel Pump

alter table Fuel\_pump

ADD Fuel\_type\_ID INT(5),

add foreign key (Fuel\_type\_ID) references Fuel\_type(Fuel\_type\_ID);

-- Fuel Type and Tank Refill: One-to-Many with Tank Refill

alter table Tank\_Refill

ADD Fuel\_type\_ID INT(5),

add foreign key (Fuel\_type\_ID) references Fuel\_type(Fuel\_type\_ID);

-- Maintenance Record and Fuel Pump: Many-to-One with Fuel Pump

alter table Maintenance\_Record

add foreign key (Pump\_ID) references Fuel\_pump(Pump\_ID);

1. **Updation:**

update Transaction

set Fuel\_Amount = 150.00

where Transaction\_ID = 1;

select\* from Transaction;

update Maintenance\_Record

set Cost = 510.5

where Maintenance\_ID = 4;

select\* from Maintenance\_Record;

update Employee

set Position = 'HR'

where Employee\_ID = 2;

select\* from Employee;

1. **Changing datatypes:**

-- Change Supplier\_ID from INT(5) to INT(10)

alter table Fuel\_type

modify Supplier\_ID int(10);

-- Change Contact\_Number from BIGINT(15) to VARCHAR(20)

alter table Supplier

modify Contact\_Number varchar(20);

-- Change Capacity from FLOAT(20) to DECIMAL(10,2)

alter table Fuel\_pump

modify Status varchar(50);

-- Change Current\_level from FLOAT(20) to DECIMAL(8,2)

alter table Fuel\_Tank

modify Location varchar(50);

-- Change Amount from FLOAT(20) to DECIMAL(10,2)

alter table Payment

modify Payment\_method varchar(50);

-- Change Refill\_Amount from FLOAT(10) to DECIMAL(8,2)

alter table Tank\_Refill

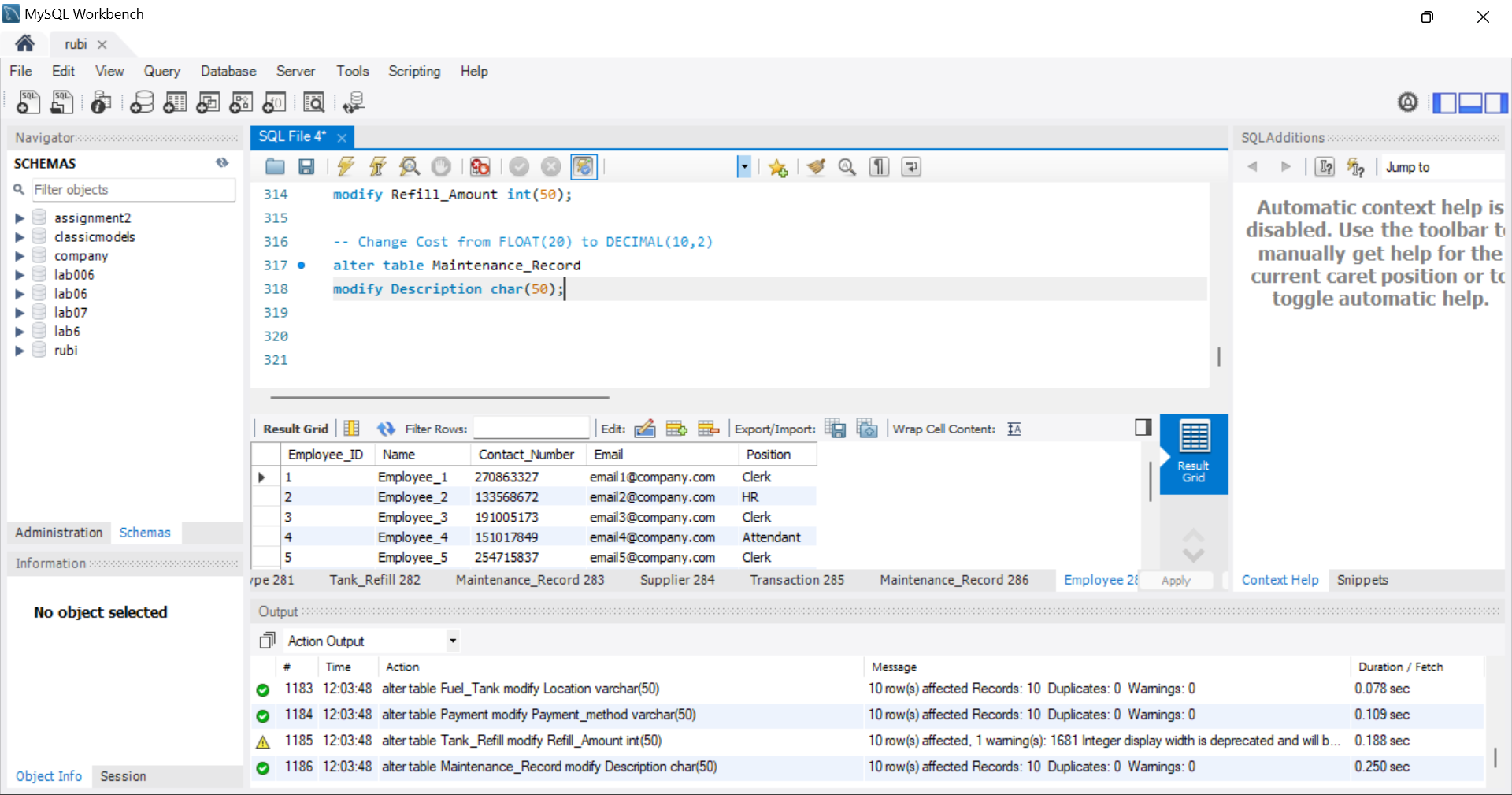
modify Refill\_Amount int(50);

-- Change Cost from FLOAT(20) to DECIMAL(10,2)

alter table Maintenance\_Record

modify Description char(50);

**output:**



1. **Rename table:**

alter table Fuel\_Tank rename to FuelTank;

alter table Transaction rename to TransactionRecord;

alter table Payment rename to PaymentInfo;

alter table Tank\_Refill rename to TankRefill;

alter table Maintenance\_Record rename to MaintenanceRecord;

alter table Employee rename to EmployeeInfo;

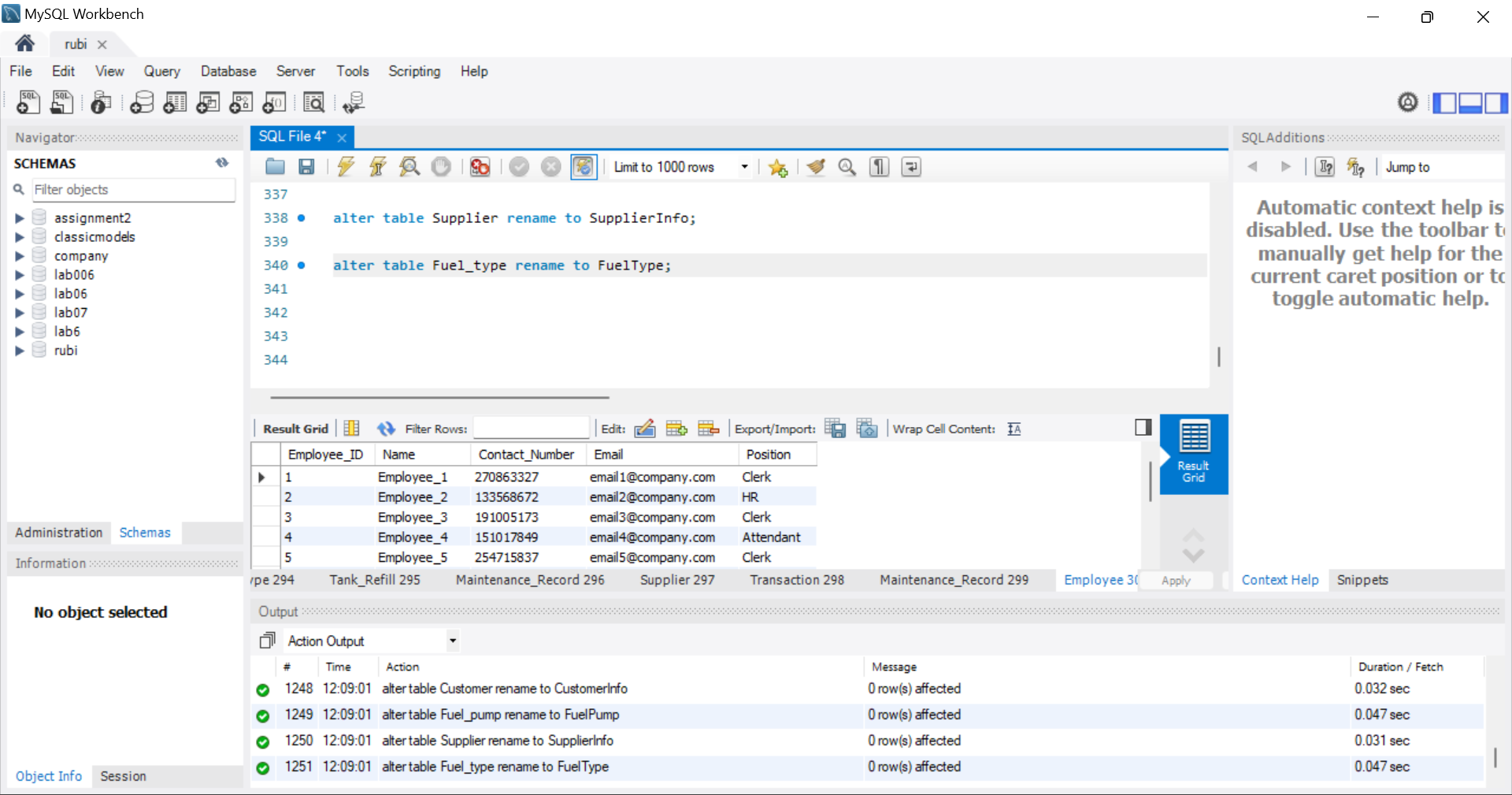
alter table Customer rename to CustomerInfo;

alter table Fuel\_pump rename to FuelPump;

alter table Supplier rename to SupplierInfo;

alter table Fuel\_type rename to FuelType;

**output:**

****

1. **Drop columns:**

alter table FuelType

drop column Supplier\_ID;

alter table SupplierInfo

drop column Contact\_Number;

alter table FuelPump

drop column Status;

alter table CustomerInfo

drop column Email;

alter table EmployeeInfo

drop column Contact\_Number;

alter table FuelTank

drop column Current\_level;

alter table TransactionRecord

drop column Fuel\_Amount,

drop column M\_Employee\_ID;

alter table PaymentInfo

drop column Payment\_method;

alter table TankRefill

drop column Date\_time;

alter table MaintenanceRecord

drop column Supplier\_ID;

**“DML”**

-- 1. Select FuelType with a specific ID:

SELECT \* FROM FuelType WHERE Fuel\_type\_ID = 1;

-- 2. Select SupplierInfo where the name is not 'Supplier\_5':

SELECT \* FROM SupplierInfo WHERE Name <> 'Supplier\_5';

-- 3. Select FuelPump with a capacity greater than 800:

SELECT \* FROM FuelPump WHERE Capacity > 800;

-- 4. Select CustomerInfo where the Contact\_Number is less than 500000000:

SELECT \* FROM CustomerInfo WHERE Contact\_Number < 500000000;

-- 5. Select EmployeeInfo where the position is 'Manager':

SELECT \* FROM EmployeeInfo WHERE Position = 'Manager';

-- 6. Select FuelTank where the capacity is between 3000 and 5000:

SELECT \* FROM FuelTank WHERE Capacity BETWEEN 3000 AND 5000;

-- 7. Select TransactionRecord where the payment method is 'Card':

SELECT \* FROM TransactionRecord WHERE Payment\_method = 'Card';

-- 8. Select PaymentInfo where the amount is greater than or equal to 200:

SELECT \* FROM PaymentInfo WHERE Amount >= 200;

-- 9. Select TankRefill where the Refill\_Amount is less than or equal to 500:

SELECT \* FROM TankRefill WHERE Refill\_Amount <= 500;

-- 10. Select MaintenanceRecord where the cost is not equal to 600:

SELECT \* FROM MaintenanceRecord WHERE Cost <> 600;

-- 11. Select FuelType where the Type\_name starts with 'P':

SELECT \* FROM FuelType WHERE Type\_name LIKE 'P%';

-- 12. Select SupplierInfo where the address contains 'Address':

SELECT \* FROM SupplierInfo WHERE Address LIKE '%Address%';

-- 13. Select FuelPump where the location ends with '3':

SELECT \* FROM FuelPump WHERE Location LIKE '%3';

-- 14. Select CustomerInfo where the name includes 'Customer':

SELECT \* FROM CustomerInfo WHERE Name LIKE '%Customer%';

-- 15. Select EmployeeInfo where the email ends with 'company.com':

SELECT \* FROM EmployeeInfo WHERE Email LIKE '%@company.com';

-- 16. Select FuelTank where the location starts with 'Location\_1':

SELECT \* FROM FuelTank WHERE Location LIKE 'Location\_1%';

-- 17. Select TransactionRecord where the date is in the year 2023:

SELECT \* FROM TransactionRecord WHERE YEAR(Date\_time) = 2023;

-- 18. Select PaymentInfo where the datetime is after '2023-06-01 00:00:00':

SELECT \* FROM PaymentInfo WHERE Date\_time > '2023-06-01 00:00:00';

-- 19. Select TankRefill where the Refill\_Amount is between 400 and 800:

SELECT \* FROM TankRefill WHERE Refill\_Amount BETWEEN 400 AND 800;

-- 20. Select MaintenanceRecord where the description includes 'Description':

SELECT \* FROM MaintenanceRecord WHERE Description LIKE '%Description%';

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

***Phase-04***

**“Joins”**

1. **Inner join:**

SELECT Transaction.Transaction\_ID,Customer.Name AS Customer\_Name, Employee.Name AS Employee\_Name, Transaction.Fuel\_Amount

FROM Transaction

INNER JOIN Customer ON Transaction.Customer\_ID = Customer.Customer\_ID

INNER JOIN Employee ON Transaction.Employee\_ID = Employee.Employee\_ID;

SELECT Tank\_Refill.Refill\_ID,Supplier.Name AS Supplier\_Name, Fuel\_Tank.Location AS Tank\_Location, Tank\_Refill.Refill\_Amount

FROM Tank\_Refill

INNER JOIN Supplier ON Tank\_Refill.Supplier\_ID = Supplier.Supplier\_ID

INNER JOIN Fuel\_Tank ON Tank\_Refill.Tank\_ID = Fuel\_Tank.Tank\_ID;

SELECT Payment.Payment\_ID,Payment.Amount,Transaction.Fuel\_Amount,Customer.Name AS Customer\_Name

FROM Payment

INNER JOIN Transaction ON Payment.Transaction\_ID = Transaction.Transaction\_ID

INNER JOIN Customer ON Transaction.Customer\_ID = Customer.Customer\_ID;

SELECT Maintenance\_Record.Maintenance\_ID,Employee.Name AS Employee\_Name, Fuel\_pump.Location AS Pump\_Location,Maintenance\_Record.Description

FROM Maintenance\_Record

INNER JOIN Employee ON Maintenance\_Record.M\_Employee\_ID = Employee.Employee\_ID

INNER JOIN Fuel\_pump ON Maintenance\_Record.Pump\_ID = Fuel\_pump.Pump\_ID;

SELECT Fuel\_type.Type\_name,Supplier.Name AS Supplier\_Name

FROM Fuel\_type

INNER JOIN Supplier ON Fuel\_type.Supplier\_ID = Supplier.Supplier\_ID;

1. **left outer join:**

SELECT Customer.Name AS Customer\_Name,Transaction.Transaction\_ID,Transaction.Fuel\_Amount

FROM Customer

LEFT JOIN Transaction ON Customer.Customer\_ID = Transaction.Customer\_ID;

SELECT Fuel\_pump.Location AS Pump\_Location,Maintenance\_Record.Maintenance\_ID,Maintenance\_Record.Description

FROM Fuel\_pump

LEFT JOIN Maintenance\_Record ON Fuel\_pump.Pump\_ID = Maintenance\_Record.Pump\_ID;

SELECT Supplier.Name AS Supplier\_Name,Tank\_Refill.Refill\_ID,Tank\_Refill.Refill\_Amount

FROM Supplier

LEFT JOIN Tank\_Refill ON Supplier.Supplier\_ID = Tank\_Refill.Supplier\_ID;

SELECT Employee.Name AS Employee\_Name,Transaction.Transaction\_ID,Transaction.Fuel\_Amount

FROM Employee

LEFT JOIN Transaction ON Employee.Employee\_ID = Transaction.Employee\_ID;

SELECT Fuel\_Tank.Location AS Tank\_Location,Tank\_Refill.Refill\_ID,Tank\_Refill.Refill\_Amount

FROM Fuel\_Tank

LEFT JOIN Tank\_Refill ON Fuel\_Tank.Tank\_ID = Tank\_Refill.Tank\_ID;

1. **right outer join:**

SELECT Transaction.Transaction\_ID,Transaction.Fuel\_Amount,Customer.Name AS Customer\_Name

FROM Transaction

RIGHT JOIN Customer ON Transaction.Customer\_ID = Customer.Customer\_ID;

SELECT Fuel\_pump.Location AS Pump\_Location,Maintenance\_Record.Maintenance\_ID,Maintenance\_Record.Description

FROM Maintenance\_Record

RIGHT JOIN Fuel\_pump ON Maintenance\_Record.Pump\_ID = Fuel\_pump.Pump\_ID;

SELECT Tank\_Refill.Refill\_ID,Tank\_Refill.Refill\_Amount,Supplier.Name AS Supplier\_Name

FROM Tank\_Refill

RIGHT JOIN Supplier ON Tank\_Refill.Supplier\_ID = Supplier.Supplier\_ID;

SELECT Maintenance\_Record.Maintenance\_ID,Maintenance\_Record.Description,Employee.Name AS Employee\_Name

FROM Maintenance\_Record

RIGHT JOIN Employee ON Maintenance\_Record.M\_Employee\_ID = Employee.Employee\_ID;

SELECT Transaction.Transaction\_ID, Transaction.Fuel\_Amount,Fuel\_pump.Location AS Pump\_Location

FROM Transaction

RIGHT JOIN Fuel\_pump ON Transaction.Pump\_ID = Fuel\_pump.Pump\_ID;

1. **natural join:**

SELECT Supplier.Name AS Supplier\_Name,Tank\_Refill.Refill\_ID,Tank\_Refill.Refill\_Amount

FROM Supplier

NATURAL JOIN Tank\_Refill;

SELECT Maintenance\_Record.Maintenance\_ID,Maintenance\_Record.Description,Supplier.Name AS Supplier\_Name

FROM Maintenance\_Record

NATURAL JOIN Supplier;

SELECT Employee.Name AS Employee\_Name, Transaction.Transaction\_ID,Transaction.Fuel\_Amount

FROM Employee

NATURAL JOIN Transaction;

SELECT Transaction.Transaction\_ID,Customer.Name AS Customer\_Name, Transaction.Fuel\_Amount

FROM Transaction

NATURAL JOIN Customer;

SELECT Fuel\_type.Type\_name,Supplier.Name AS Supplier\_Name

FROM Fuel\_type

NATURAL JOIN Supplier;

1. **Self join:**

SELECT C1.Customer\_ID AS Customer1\_ID,C1.Name AS Customer1\_Name,C2.Customer\_ID AS Customer2\_ID,C2.Name AS Customer2\_Name,C1.Address

FROM Customer C1

JOIN Customer C2 ON C1.Address = C2.Address AND C1.Customer\_ID < C2.Customer\_ID;

SELECT E1.Employee\_ID AS Employee1\_ID,E1.Name AS Employee1\_Name,E2.Employee\_ID AS Employee2\_ID,E2.Name AS Employee2\_Name,E1.Contact\_Number

FROM Employee E1

JOIN Employee E2 ON E1.Contact\_Number = E2.Contact\_Number AND E1.Employee\_ID < E2.Employee\_ID;

SELECT T1.Transaction\_ID AS Transaction1\_ID,T2.Transaction\_ID AS Transaction2\_ID,T1.Date\_time

FROM Transaction T1

JOIN Transaction T2 ON T1.Date\_time = T2.Date\_time AND T1.Transaction\_ID < T2.Transaction\_ID;

SELECT FT1.Tank\_ID AS Tank1\_ID,FT2.Tank\_ID AS Tank2\_ID,FT1.Capacity

FROM Fuel\_Tank FT1

JOIN Fuel\_Tank FT2 ON FT1.Capacity = FT2.Capacity AND FT1.Tank\_ID < FT2.Tank\_ID;

SELECT FP1.Pump\_ID AS Pump1\_ID,FP2.Pump\_ID AS Pump2\_ID,FP1.Location

FROM Fuel\_pump FP1

JOIN Fuel\_pump FP2 ON FP1.Location = FP2.Location AND FP1.Pump\_ID < FP2.Pump\_ID;

* **Nested Queries:**

1. Select FuelType with a specific ID:

SELECT \* FROM Fuel\_type WHERE Fuel\_type\_ID = (SELECT Fuel\_type\_ID FROM Fuel\_type WHERE Fuel\_type\_ID = 1);

1. Select SupplierInfo where the name is not 'Supplier\_5':

SELECT \* FROM Supplier WHERE Name IN (SELECT Name FROM Supplier WHERE Name <> 'Supplier\_5');

1. Select FuelPump with a capacity greater than 800:

SELECT \* FROM Fuel\_pump WHERE Capacity > (SELECT MIN(Capacity) FROM Fuel\_pump WHERE Capacity > 800);

1. Select CustomerInfo where the Contact\_Number is less than 500000000:

SELECT \* FROM Customer WHERE Contact\_Number < (SELECT MAX(Contact\_Number) FROM Customer WHERE Contact\_Number < 500000000);

1. Select EmployeeInfo where the position is 'Manager':

SELECT \* FROM Employee WHERE Position IN (SELECT Position FROM Employee WHERE Position = 'Manager');

1. Select FuelTank where the capacity is between 3000 and 5000:

SELECT \* FROM Fuel\_Tank WHERE Capacity BETWEEN (SELECT MIN(Capacity) FROM Fuel\_Tank WHERE Capacity BETWEEN 3000 AND 5000) AND (SELECT MAX(Capacity) FROM Fuel\_Tank WHERE Capacity BETWEEN 3000 AND 5000);

1. Select TransactionRecord where the transaction has a corresponding payment in PaymentInfo:

SELECT \* FROM Transaction t1 WHERE EXISTS (SELECT \* FROM Payment p1 WHERE p1.Transaction\_ID = t1.Transaction\_ID);

1. Select PaymentInfo where the amount is greater than or equal to 200:

SELECT \* FROM Payment WHERE Amount >= (SELECT MIN(Amount) FROM Payment WHERE Amount >= 200);

1. Select TankRefill where the Refill\_Amount is less than or equal to 500:

SELECT \* FROM Tank\_Refill WHERE Refill\_Amount <= (SELECT MAX(Refill\_Amount) FROM Tank\_Refill WHERE Refill\_Amount <= 500);

1. Select MaintenanceRecord where the cost is not equal to 600:

SELECT \* FROM Maintenance\_Record WHERE Cost <> (SELECT Cost FROM Maintenance\_Record WHERE Cost = 600 LIMIT 1);

* **Correlated Subqueries:**

1. Select FuelType with a specific ID:

SELECT \* FROM Fuel\_type t1 WHERE EXISTS (SELECT \* FROM Fuel\_type t2 WHERE t1.Fuel\_type\_ID = t2.Fuel\_type\_ID AND t2.Fuel\_type\_ID = 1);

2. Select SupplierInfo where the name is not 'Supplier\_5':

SELECT \* FROM Supplier s1 WHERE NOT EXISTS (SELECT \* FROM Supplier s2 WHERE s2.Name = 'Supplier\_5' AND s1.Supplier\_ID = s2.Supplier\_ID);

3. Select FuelPump with a capacity greater than 800:

SELECT \* FROM Fuel\_pump f1 WHERE EXISTS (SELECT \* FROM Fuel\_pump f2 WHERE f1.Pump\_ID = f2.Pump\_ID AND f2.Capacity > 800);

4. Select CustomerInfo where the Contact\_Number is less than 500000000:

SELECT \* FROM Customer c1 WHERE EXISTS (SELECT \* FROM Customer c2 WHERE c1.Customer\_ID = c2.Customer\_ID AND c2.Contact\_Number < 500000000);

5. Select EmployeeInfo where the position is 'Manager':

SELECT \* FROM Employee e1 WHERE EXISTS (SELECT \* FROM Employee e2 WHERE e1.Employee\_ID = e2.Employee\_ID AND e2.Position = 'Manager');

6. Select FuelTank where the capacity is between 3000 and 5000:

SELECT \* FROM Fuel\_Tank ft1 WHERE EXISTS (SELECT \* FROM Fuel\_Tank ft2 WHERE ft1.Tank\_ID = ft2.Tank\_ID AND ft2.Capacity BETWEEN 3000 AND 5000);

7. Select TransactionRecord where the transaction has a corresponding payment in PaymentInfo:

SELECT \* FROM Transaction tr1 WHERE EXISTS (SELECT \* FROM Payment p1 WHERE p1.Transaction\_ID = tr1.Transaction\_ID);

8. Select PaymentInfo where the amount is greater than or equal to 200:

SELECT \* FROM Payment p1 WHERE EXISTS (SELECT \* FROM Payment p2 WHERE p1.Payment\_ID = p2.Payment\_ID AND p2.Amount >= 200);

9. Select TankRefill where the Refill\_Amount is less than or equal to 500:

SELECT \* FROM Tank\_Refill tr1 WHERE EXISTS (SELECT \* FROM Tank\_Refill tr2 WHERE tr1.Refill\_ID = tr2.Refill\_ID AND tr2.Refill\_Amount <= 500);

10. Select MaintenanceRecord where the cost is not equal to 600:

SELECT \* FROM Maintenance\_Record mr1 WHERE EXISTS (SELECT \* FROM Maintenance\_Record mr2 WHERE mr1.Maintenance\_ID = mr2.Maintenance\_ID AND mr2.Cost <> 600);

***THE END***

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*