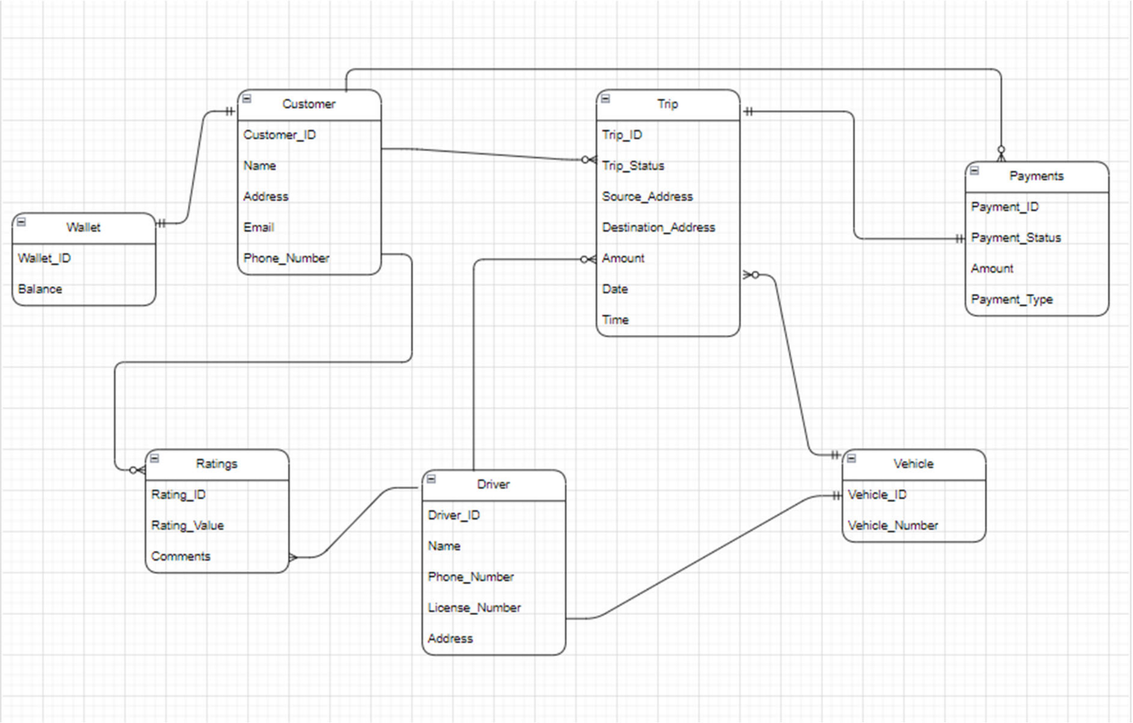
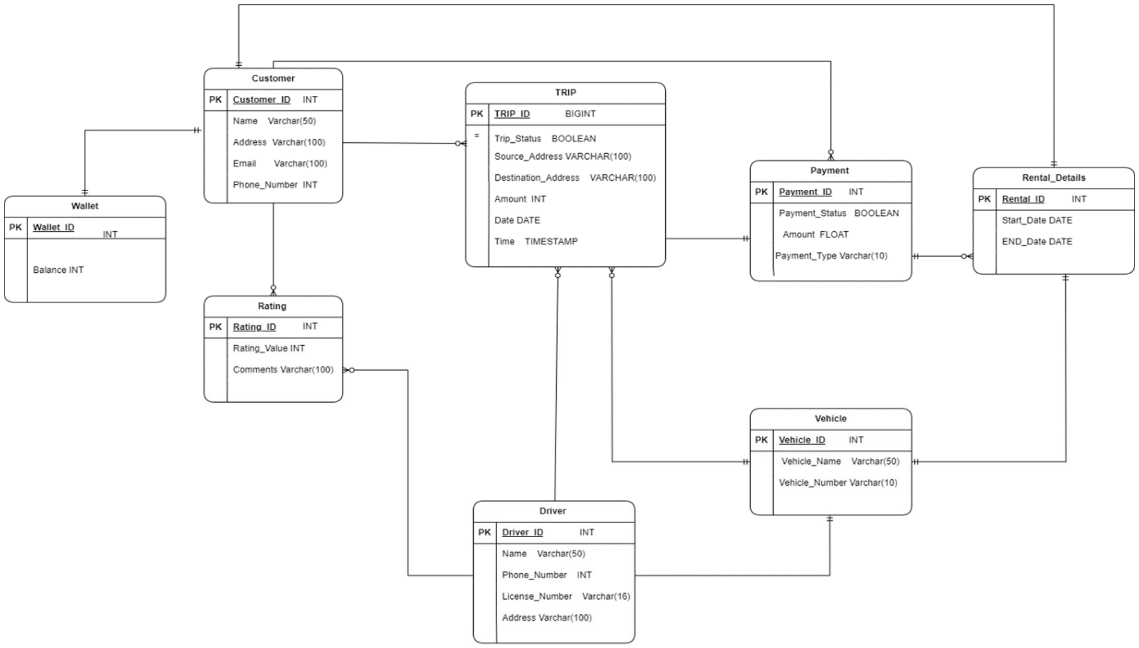
Data Modeling and E-R Diagrams

Task 1: Create a conceptual and logical and physical data model using ER diagram for UBER (Ride sharing app).

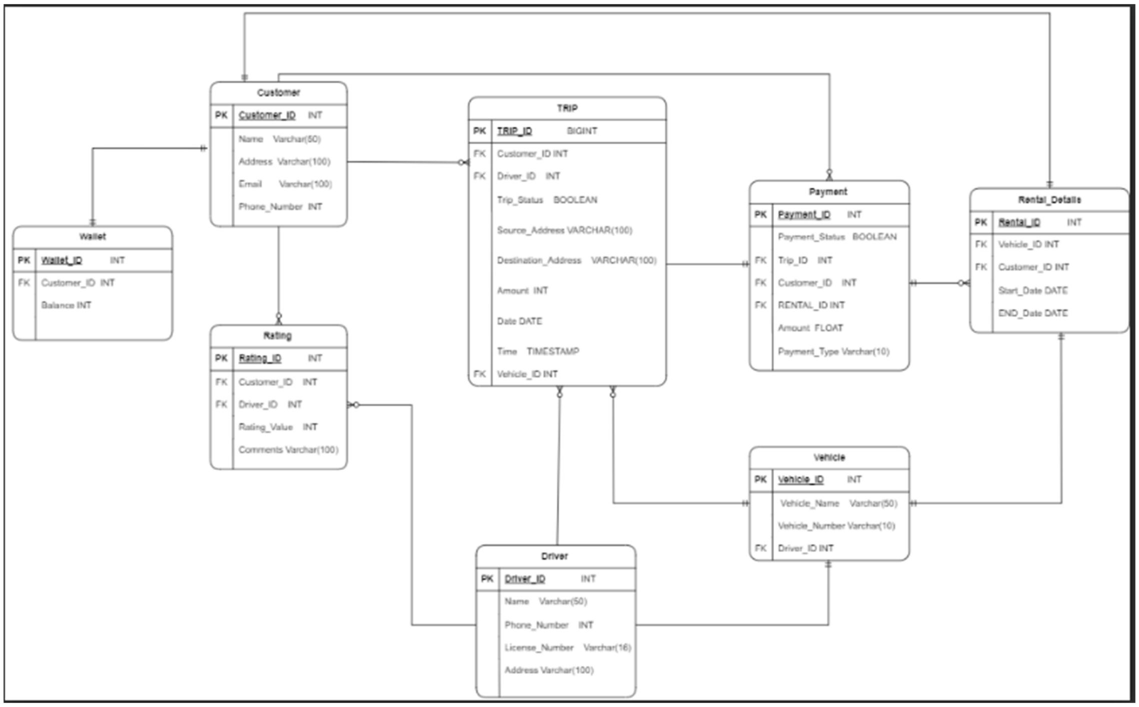
Conceptual model:



Logical model:



Physical model:



SQL Statements

Task 2: Create tables from the physical data model from above task.
(Create appropriate constraints and keys)

Ans:

```
create database cardriver

use cardriver;

CREATE TABLE Customer (

   Customer_ID INT PRIMARY KEY,

   Name VARCHAR(255),

   Address VARCHAR(150),

   Email VARCHAR(255),

   Phone_No VARCHAR(20)

);

CREATE TABLE Driver (

   DriverID INT PRIMARY KEY,

   Name Varchar(20),

   Phone_Number int check(Phone_number >= 1000000000 and Phone_number < 999999999),

   LicenseNumber VARCHAR(50),

   Address Varchar(50)

);
```

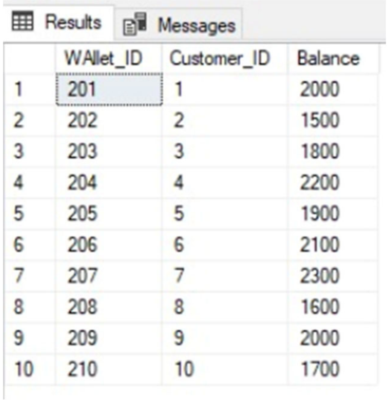
```
alter table Driver
add Phone_Number BIGINT;
CREATE TABLE Wallet(
WAllet_ID int,
Customer_ID int,
Balance int,
FOREIGN KEY (Customer_ID) REFERENCES Customer(Customer_ID)
);
CREATE TABLE Vehicle (
    Vehicle_ID INT PRIMARY KEY,
    Vehicle_Name varchar(10),
    Vehicle_Number varchar(10),
    Driver_ID int,
    FOREIGN KEY (Driver_ID) REFERENCES Driver(DriverID)
);
CREATE TABLE Trip (
    Trip_ID INT PRIMARY KEY,
    Customer_ID INT,
    DriverID INT,
    Trip_Status BIT,
    Source_Address Varchar(50),
    Destination_Address Varchar(50),
    Amount int,
    Date date,
    Vehicle_ID int,
    FOREIGN KEY (Vehicle_ID) REFERENCES Vehicle(Vehicle_ID),
    FOREIGN KEY (DriverID) REFERENCES Driver(DriverID)
);
CREATE TABLE Payment (
    PaymentID INT PRIMARY KEY,
    Payment_Status BIT,
    Trip_ID INT,
    Customer_ID INT,
    Rental_ID int,
    Amount Float,
    PaymentType VARCHAR(50),
    FOREIGN KEY (Trip_ID) REFERENCES Trip(Trip_ID),
    FOREIGN KEY (Rental_ID) REFERENCES Rental_Details(Rental_ID),
    FOREIGN KEY (Customer_ID) REFERENCES Customer(Customer_ID)
);
CREATE TABLE Rating (
    Rating_ID INT PRIMARY KEY,
    Customer_ID INT,
    DriverID INT,
    Rating_Value INT,
    Comments VARCHAR(100),
    FOREIGN KEY (Customer_id) REFERENCES Customer(Customer_id),
    FOREIGN KEY (DriverID) REFERENCES Driver(DriverID)
);
CREATE TABLE Rental_Details (
    Rental_ID INT PRIMARY KEY,
    Vehicle_ID INT,
    Customer_id INT,
    Start_Date DATE,
    End_Date DATE,
    FOREIGN KEY (Vehicle_ID) REFERENCES Vehicle(Vehicle_ID),
    FOREIGN KEY (Customer_id) REFERENCES Customer(Customer_id)
);
```

Task 3: Populate all the tables created with sample data.

Ans:

```
-- Inserting data into the Customer table
INSERT INTO Customer (Customer_ID, Name, Address, Email, Phone_No)
VALUES
(1, 'Ramesh Kumar', '123 ABC Street, Mumbai', 'ramesh@example.com', '9876543210'),
(2, 'Priya Patel', '456 XYZ Road, Delhi', 'priya@example.com', '8765432109'),
(3, 'Suresh Singh', '789 PQR Lane, Bangalore', 'suresh@example.com', '7654321098'),
(4, 'Anita Gupta', '321 LMN Avenue, Kolkata', 'anita@example.com', '6543210987'),
(5, 'Rajesh Sharma', '654 DEF Colony, Chennai', 'rajesh@example.com', '5432109876'),
(6, 'Meera Verma', '987 GHI Street, Hyderabad', 'meera@example.com', '4321098765'),
(7, 'Amit Shah', '234 JKL Road, Pune', 'amit@example.com', '3210987654'),
(8, 'Sunita Reddy', '567 UVW Lane, Jaipur', 'sunita@example.com', '2109876543'),
(9, 'Vivek Mishra', '890 OPQ Colony, Ahmedabad', 'vivek@example.com', '1098765432'),
(10, 'Pooja Choudhary', '123 RST Avenue, Lucknow', 'pooja@example.com', '0987654321');
-- Inserting data into the Driver table
INSERT INTO Driver (DriverID, Name, Phone_Number, LicenseNumber, Address)
VALUES
(11, 'Raj Kumar', 9876543210, 'DL1234', '123 ABC Street, Mumbai'),
(12, 'Aarti Singh', 8765432109, 'MH5678', '456 XYZ Road, Delhi'),
(13, 'Vikram Verma', 7654321098, 'KA91011', '789 PQR Lane, Bangalore'),
(14, 'Neha Gupta', 6543210987, 'WB1213', '321 LMN Avenue, Kolkata'),
(15, 'Sanjay Sharma', 5432109876, 'TN1415', '654 DEF Colony, Chennai'),
(16, 'Divya Reddy', 4321098765, 'TS1617', '987 GHI Street, Hyderabad'),
(17, 'Ravi Patel', 3210987654, 'MH1819', '234 JKL Road, Pune'),
(18, 'Kavita Singh', 2109876543, 'RJ2021', '567 UVW Lane, Jaipur'),
(19, 'Rajesh Tiwari', 1098765432, 'GJ2223', '890 OPQ Colony, Ahmedabad'),
(20, 'Anil Kumar', 1987654321, 'UP2425', '123 RST Avenue, Lucknow');
-- Inserting data into the Vehicle table
INSERT INTO Vehicle (Vehicle_ID, Vehicle_Name, Vehicle_Number, Driver_ID)
VALUES
(21, 'Car', 'MH01AB1234', 11),
(22, 'SUV', 'DL02CD5678', 12),
(23, 'Van', 'KA03EF9101', 13),
(24, 'Truck', 'WB04GH1213', 14),
(25, 'Motorcycle', 'TN05IJ1415',15),
(26, 'Auto', 'TS06KL1617', 16),
(27, 'Tempo', 'MH07MN1819', 17),
(28, 'Bus', 'RJ08OP2021',18),
(29, 'Cycle', 'GJ09QR2223', 19),
(30, 'Bicycle', 'UP10ST2425', 20);

-- Inserting data into the Trip table

INSERT INTO Trip (Trip_ID, Customer_ID, DriverID, Trip_Status, Source_Address, Destination_Address, Amount, Date, Vehicle_ID)

VALUES

(41, 1, 11, 1, '123 ABC Street, Mumbai', '456 XYZ Road, Delhi', 500, '2024-02-09', 21),

(42, 2, 12, 1, '456 XYZ Road, Delhi', '789 PQR Lane, Bangalore', 750, '2024-02-10', 22),

(43, 3, 13, 0, '789 PQR Lane, Bangalore', '321 LMN Avenue, Kolkata', 600, '2024-02-11', 23),

(44, 4, 14, 1, '321 LMN Avenue, Kolkata', '654 DEF Colony, Chennai', 700, '2024-02-12', 24),

(45, 5, 15, 1, '654 DEF Colony, Chennai', '987 GHI Street, Hyderabad', 550, '2024-02-13', 25),

(46, 6, 16, 1, '987 GHI Street, Hyderabad', '234 JKL Road, Pune', 800, '2024-02-14', 26),

(47, 7, 17, 1, '234 JKL Road, Pune', '567 UVW Lane, Jaipur', 900, '2024-02-15', 27),

(48, 8, 18, 0, '567 UVW Lane, Jaipur', '890 OPQ Colony, Ahmedabad', 650, '2024-02-16', 28),

(49, 9, 19, 1, '890 OPQ Colony, Ahmedabad', '123 RST Avenue, Lucknow', 850, '2024-02-17', 29),

(50, 10, 20, 1, '123 RST Avenue, Lucknow', '123 ABC Street, Mumbai', 700, '2024-02-18', 30);

-- Inserting data into the Payment table

INSERT INTO Payment (PaymentID, Payment_Status, Trip_ID, Customer_ID, Rental_ID, Amount, PaymentType)

VALUES

(101, 1, 41, 1, 51, 500, 'Card'),

(102, 1, 42, 2, 52, 750, 'Cash'),

(103, 0, 43, 3, 53, 600, 'UPI'),

(104, 1, 44, 4, 54, 700, 'Cash'),

(105, 1, 45, 5, 55, 550, 'Card'),

(106, 1, 46, 6, 56, 800, 'Cash'),

(107, 1, 47, 7, 57, 900, 'Card'),

(108, 0, 48, 8, 58, 650, 'UPI'),

(109, 1, 49, 9, 59, 850, 'Cash'),

(110, 1, 50, 10, 60, 700, 'Card');

-- Inserting data into the Wallet table

INSERT INTO Wallet (Wallet_ID, Customer_id, Balance)

VALUES

(201, 1, 2000),

(202, 2, 1500),
(203, 3, 1800),
(204, 4, 2200),
(205, 5, 1900),
(206, 6, 2100),
(207, 7, 2300),
(208, 8, 1600),
(209, 9, 2000),
(210, 10, 1700);
-- Inserting data into the Rating table
INSERT INTO Rating (Rating_ID, Customer_ID, DriverID, Rating_Value, Comments)
VALUES
(31, 1, 11, 4, 'Great service'),
(32, 2, 12, 5, 'Very polite and helpful'),
(33, 3, 13, 3, 'Average experience'),
(34, 4, 14, 4, 'Good driving skills'),
(35, 5, 15, 4, 'On-time arrival'),
(36, 6, 16, 5, 'Excellent service'),
(37, 7, 17, 4, 'Clean vehicle'),
(38, 8, 18, 3, 'Average ride'),
(39, 9, 19, 5, 'Friendly driver'),
(40,10, 20, 4, 'Smooth ride');
-- Inserting data into the Rental_Details table
INSERT INTO Rental_Details (Rental_ID, Vehicle_ID, Customer_id, Start_Date, End_Date)
VALUES
(51, 21, 1, '2024-02-09', '2024-02-10'),
(52, 22, 2, '2024-02-10', '2024-02-11'),
(53, 23, 3, '2024-02-11', '2024-02-12'),
(54, 24, 4, '2024-02-12', '2024-02-13'),
(55, 25, 5, '2024-02-13', '2024-02-14'),
(56, 26, 6, '2024-02-14', '2024-02-15'),
(57, 27, 7, '2024-02-15', '2024-02-16'),
(58, 28, 8, '2024-02-16', '2024-02-17'),
(59, 29, 9, '2024-02-17', '2024-02-18'),
(60, 30, 10, '2024-02-18', '2024-02-19');
```

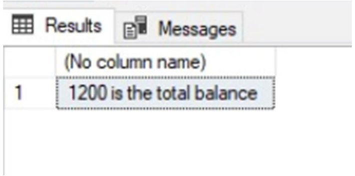
Task 4: Write a stored routine to update the wallet balance of a user.
(2 incoming argument i.e. user and amount. (Print or return) updated wallet balance)



	WAllet_ID	Customer_ID	Balance
1	201	1	2000
2	202	2	1500
3	203	3	1800
4	204	4	2200
5	205	5	1900
6	206	6	2100
7	207	7	2300
8	208	8	1600
9	209	9	2000
10	210	10	1700

```
DECLARE @input INT = 3;
DECLARE @factorialResult INT;
SET @factorialResult = dbo.CalculateFactorial(@input);
SELECT @factorialResult AS FactorialResult;
SELECT
    'Factorial of 5 is ' + CAST(dbo.CalculateFactorial(5) AS VARCHAR) AS Result1,
    'Factorial of 0 is ' + CAST(dbo.CalculateFactorial(0) AS VARCHAR) AS Result2,
    'Factorial of 10 is ' + CAST(dbo.CalculateFactorial(10) AS VARCHAR) AS Result3;
select * from Wallet;
go
CREATE PROCEDURE UpdateWallet6(
    @user int,
    @amt int
)
AS
BEGIN
    -- Update Column1 by adding the specified value
    UPDATE wallet
    SET Balance = Balance - @amt
    WHERE Customer_ID = @user
            select  cast(balance as varchar(10)) +' is the total balance ' from wallet where Customer_ID= @user
END;
go
```

```
EXEC dbo.UpdateWallet6 @user =1,@amt=200;

select * from wallet
```



	(No column name)
1	1200 is the total balance

Task 5: Write the correct SQL execution order for the following SELECT query.

```
SELECT DISTINCT column, AGG_FUNC (column_or_expression), ...
FROM mytable
JOIN another_table ON mytable.column = another_table.column
WHERE constraint_expression
GROUP BY column
HAVING constraint_expression
ORDER BY column ASC/DESC
LIMIT count OFFSET COUNT;
```

Ans : The order of execution for the query is

1. FROM and JOINs are executed simultaneously
2. WHERE
3. GROUP BY
4. HAVING
5. SELECT
6. DISTINCT

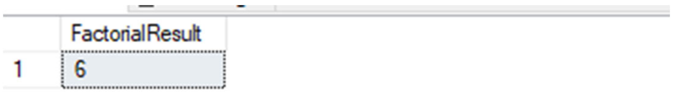
Functions and Procedures

Task 6: Write a simple function to calculate factorial of any number. (function accepts an IN parameter and returns the factorial of that).

Ans:

```
CREATE FUNCTION dbo.Cf (@n INT)

RETURNS INT

AS

BEGIN

  DECLARE @result INT = 1;

  DECLARE @i INT = 1;

  WHILE @i <= @n

  BEGIN

    SET @result = @result * @i;

    SET @i = @i + 1;
    END
    RETURN @result;
END;
GO
DECLARE @input INT = 3;
DECLARE @factorialResult INT;
SET @factorialResult = dbo.Cf(@input);
SELECT @factorialResult AS FactorialResult;
```

Output:



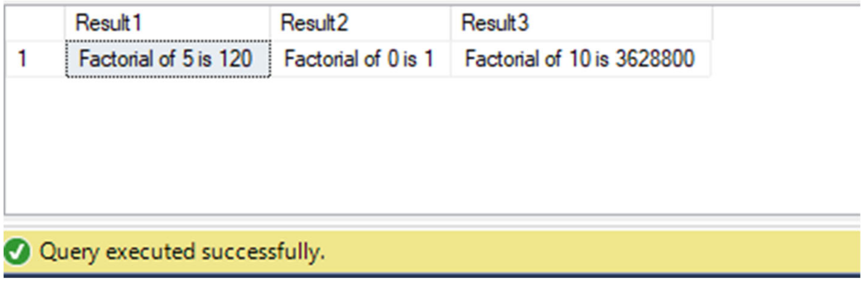
	FactorialResult
1	6

Task 7: Execute the function with multiple values as a test case

Ans:

```
SELECT
    'Factorial of 5 is ' + CAST(dbo.Cf(5) AS VARCHAR) AS Result1,
    'Factorial of 0 is ' + CAST(dbo.Cf(0) AS VARCHAR) AS Result2,
    'Factorial of 10 is ' + CAST(dbo.Cf(10) AS VARCHAR) AS Result3;
```

Output:



	Result1	Result2	Result3
1	Factorial of 5 is 120	Factorial of 0 is 1	Factorial of 10 is 3628800

Query executed successfully.

MCQs:

1.) What is a tuple equivalent to in SQL?
Ans- A row in the table

2.) How many NULL value that Unique key can have?
Ans- 1

3.) Which Join is used to get only match tuples?
Ans- Inner join

4.) Which is the kind of Aggregate function?
Ans- MIN

5.) Which are the TRANSACTION control commands?

- a.) Commit
- b.)Savepoint
- c.) Rollback
- d.) All the above

Ans- All the above

6.) When a program is abnormally terminated in a transaction,which of the following command occurs?
Ans- Rollback

7.) What is wrong with the following query?

Select V_ID,P_ID,P_DESC,P_RATE rate FROM TABLE1 GROUP BY V_ID

- a.) No Aggregate function is used
- b.)No where cluase is specified
- c.)Alias is only for one column
- d.) Nothing is wrong

Ans- No Aggregate function is used