

ASSIGNMENT TITLE:
Courier Management System: MSSQL ASSIGNMENT

SUBMITED BY: E.R HARISH
DATE OF SUBMISSION: 25/09/2024

GITHUB link:
https://github.com/Harish0562/Hexaware_MSSQL_Assignment.git

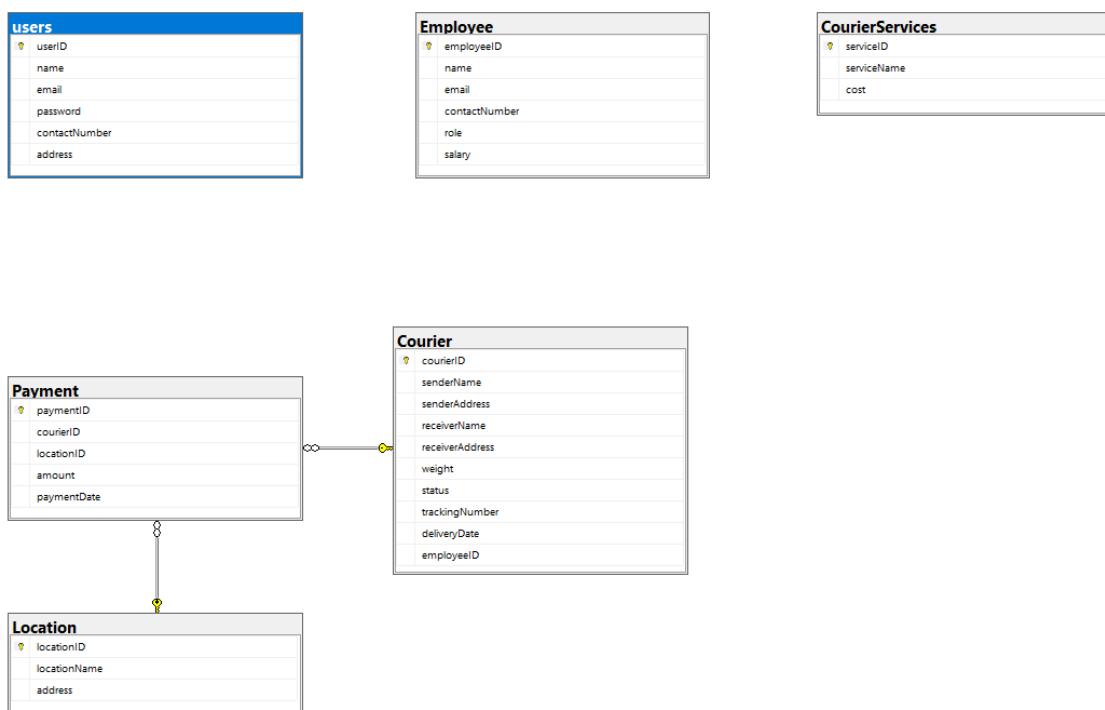
Courier Management System: MSSQL ASSIGNMENT

Task1 Database Design:

Design a SQL schema for a Courier Management System with tables for Customers, Couriers, Orders, and Parcels. Define the relationships between these tables using appropriate foreign keys.

Requirements:

- Define the Database Schema
- Create SQL tables for entities such as User, Courier, Employee, Location, Payment
- Define relationships between these tables (one-to-many, many-to-many, etc.).
- Populate Sample Data.
- Insert sample data into the tables to simulate real-world scenarios.



```
use Courier_management_assignment;
CREATE TABLE users (
    userID INT PRIMARY KEY,
    name VARCHAR(255),
    email VARCHAR(255) UNIQUE,
    password VARCHAR(255),
    contactNumber VARCHAR(20),
    address TEXT
);
```

100 % Messages
Commands completed successfully.
Completion time: 2024-09-19T14:29:38.1110562+05:30

```
SQLQuery1.sql - D...HDUG6\Harish (71)*  X
CREATE TABLE Courier (
    courierID INT PRIMARY KEY,
    senderName VARCHAR(255),
    senderAddress TEXT,
    receiverName VARCHAR(255),
    receiverAddress TEXT,
    weight DECIMAL(5, 2),
    status VARCHAR(50),
    trackingNumber VARCHAR(20) UNIQUE,
    deliveryDate DATE
);
```

100 % Messages
Commands completed successfully.
Completion time: 2024-09-19T14:32:19.1768780+05:30

```
CREATE TABLE CourierServices (
    serviceID INT PRIMARY KEY,
    serviceName VARCHAR(100),
    cost DECIMAL(8, 2)
);
```

00 % Messages
Commands completed successfully.
Completion time: 2024-09-19T14:33:03.4240172+05:30

```
SQLQuery1.sql - D...HDUG6\Harish (71)*  X
CREATE TABLE Employee (
    employeeID INT PRIMARY KEY,
    name VARCHAR(255),
    email VARCHAR(255) UNIQUE,
    contactNumber VARCHAR(20),
    role VARCHAR(50),
    salary DECIMAL(10, 2)
);
```

100 % Messages
Commands completed successfully.
Completion time: 2024-09-19T14:33:57.0141301+05:30

```
CREATE TABLE Location (
    locationID INT PRIMARY KEY,
    locationName VARCHAR(100),
    address TEXT
```

```
CREATE TABLE Payment (
    paymentID INT PRIMARY KEY,
    courierID INT,
    locationID INT,
    amount DECIMAL(10, 2),
    paymentDate DATE,
    FOREIGN KEY (courierID) REFERENCES Courier(courierID),
    FOREIGN KEY (locationID) REFERENCES Location(locationID)
);
```

100 % 
Messages
Commands completed successfully.
Completion time: 2024-09-19T14:35:37.8553859+05:30

```
INSERT INTO users(userID, name, email, password, contactNumber, address)
VALUES
(1, 'Arjun', 'arjun@gmail.com', 'password123', '123 Street, City'),
(2, 'Kaviya', 'kaviya@gmail.com', 'securepass', '9876543210', '456 Street, Town'),
(3, 'Mani', 'mani@gmail.com', 'pass123', '8765432109', '789 Street, Village'),
(4, 'Priya', 'priya@gmail.com', 'mysecurepass', '7654321098', '012 Street, County'),
(5, 'Raj', 'raj@gmail.com', 'password456', '6543210987', '345 Street, District'),
(6, 'Nisha', 'nisha@gmail.com', 'pass456', '5432109876', '678 Street, State'),
(7, 'Ganesh', 'ganesh@gmail.com', 'securepassword', '4321098765', '901 Street, Country'),
(8, 'Saranya', 'saranya@gmail.com', 'password789', '3210987654', '234 Street, Province'),
(9, 'Kumar', 'kumar@gmail.com', 'mypassword789', '2109876543', '567 Street, Region'),
(10, 'Shalini', 'shalini@gmail.com', 'passwordabc', '1098765432', '890 Street, Territory');
```

(10 rows affected)

```
SELECT * FROM Users;
```

100 %

Results Messages

	userID	name	email	password	contactNumber	address
1	1	Arjun	arjun@gmail.com	password123	1234567890	123 Street, City
2	2	Kaviya	kaviya@gmail.com	securepass	9876543210	456 Street, Town
3	3	Mani	mani@gmail.com	pass123	8765432109	789 Street, Village
4	4	Priya	priya@gmail.com	mysecurepass	7654321098	012 Street, County
5	5	Raj	raj@gmail.com	password456	6543210987	345 Street, District
6	6	Nisha	nisha@gmail.com	pass456	5432109876	678 Street, State
7	7	Ganesh	ganesh@gmail.com	securepassword	4321098765	901 Street, Country
8	8	Saranya	saranya@gmail.com	password789	3210987654	234 Street, Province
9	9	Kumar	kumar@gmail.com	mypass789	2109876543	567 Street, Region
10	10	Shalini	shalini@gmail.com	passwordabc	1098765432	890 Street, Territory

```
INSERT INTO courier (courierID, senderName, senderAddress, receiverName, receiverAddress, weight, status, trackingNumber, deliveryDate)
VALUES
(1, 'John Doe', '123 Elm Street', 'Jane Smith', '456 Oak Avenue', 2.5, 'Delivered', 'TRACK001', '2023-09-10'),
(2, 'Robert Brown', '789 Maple Road', 'Emily Davis', '101 Birch Lane', 1.2, 'In Transit', 'TRACK002', '2023-09-12'),
(3, 'David Lee', '234 Cedar Street', 'Michael Green', '567 Pine Blvd', 3.0, 'Processing', 'TRACK003', '2023-09-15'),
(4, 'Laura White', '890 Spruce Way', 'Emma Wilson', '321 Willow Court', 2.7, 'Delivered', 'TRACK004', '2023-09-11'),
(5, 'William Taylor', '654 Chestnut Circle', 'Olivia Adams', '987 Aspen Drive', 4.1, 'In Transit', 'TRACK005', '2023-09-13'),
(6, 'John Doe', '123 Elm Street', 'Michael Green', '567 Pine Blvd', 5.0, 'Processing', 'TRACK006', '2023-09-14'),
(7, 'Emily Davis', '101 Birch Lane', 'Robert Brown', '789 Maple Road', 3.5, 'Delivered', 'TRACK007', '2023-09-10'),
(8, 'Jane Smith', '456 Oak Avenue', 'William Taylor', '654 Chestnut Circle', 2.9, 'Processing', 'TRACK008', '2023-09-16'),
(9, 'Laura White', '890 Spruce Way', 'David Lee', '234 Cedar Street', 1.8, 'In Transit', 'TRACK009', '2023-09-18'),
(10, 'Emma Wilson', '321 Willow Court', 'Olivia Adams', '987 Aspen Drive', 2.0, 'Delivered', 'TRACK010', '2023-09-17');
```

91 %

Messages

(10 rows affected)

Completion time: 2024-09-19T14:47:10.9197647+05:30

```
INSERT INTO CourierServices (serviceID, serviceName, cost)
VALUES
(1, 'Standard Delivery', 10.00),
(2, 'Express Delivery', 20.00),
(3, 'Same Day Delivery', 25.00),
(4, 'Overnight Delivery', 30.00),
(5, 'Weekend Delivery', 35.00),
(6, 'International Shipping', 50.00),
(7, 'Economy Delivery', 8.00),
(8, 'Next Day Delivery', 18.00),
(9, 'Two-Day Shipping', 22.00),
(10, 'Freight Delivery', 75.00);
```

100 %

Messages

(10 rows affected)

Completion time: 2024-09-19T14:49:21.3337186+05:30

```
INSERT INTO Employee (employeeID, name, email, contactNumber, role, salary)
VALUES
(1, 'Alice Johnson', 'alice.johnson@example.com', '555-1111', 'Courier', 35000.00),
(2, 'Bob Carter', 'bob.carter@example.com', '555-2222', 'Manager', 55000.00),
(3, 'Charlie Young', 'charlie.young@example.com', '555-3333', 'Dispatcher', 40000.00),
(4, 'Diana Scott', 'diana.scott@example.com', '555-4444', 'Courier', 35000.00),
(5, 'Ethan Ford', 'ethan.ford@example.com', '555-5555', 'Admin', 45000.00),
(6, 'Fiona Blake', 'fiona.blake@example.com', '555-6666', 'Courier', 36000.00),
(7, 'George Hall', 'george.hall@example.com', '555-7777', 'Supervisor', 48000.00),
(8, 'Helen Moore', 'helen.moore@example.com', '555-8888', 'Manager', 55000.00),
(9, 'Ian Baker', 'ian.baker@example.com', '555-9999', 'Courier', 34000.00),
(10, 'Jenny Watts', 'jenny.watts@example.com', '555-1010', 'Dispatcher', 41000.00);
```

100 %

Messages

(10 rows affected)

Completion time: 2024-09-19T14:50:09.3238262+05:30

```
INSERT INTO Location (locationID, locationName, address)
VALUES
(1, 'New York Hub', '101 Manhattan St, New York, NY'),
(2, 'Los Angeles Hub', '202 Hollywood Blvd, Los Angeles, CA'),
(3, 'Chicago Hub', '303 Windy City Rd, Chicago, IL'),
(4, 'Houston Hub', '404 Lone Star Ln, Houston, TX'),
(5, 'Phoenix Hub', '505 Desert Ave, Phoenix, AZ'),
(6, 'San Francisco Hub', '606 Bay St, San Francisco, CA'),
(7, 'Dallas Hub', '707 Big D Blvd, Dallas, TX'),
(8, 'Miami Hub', '808 Ocean Dr, Miami, FL'),
(9, 'Seattle Hub', '909 Rainy St, Seattle, WA'),
(10, 'Boston Hub', '1010 Freedom Trail, Boston, MA');
```

100 % ▶ Messages

(10 rows affected)

Completion time: 2024-09-19T14:50:44.9420766+05:30

```
INSERT INTO Payment (paymentID, courierID, locationID, amount, paymentDate)
VALUES
(1, 1, 1, 25.00, '2023-09-10'),
(2, 2, 2, 40.00, '2023-09-12'),
(3, 3, 3, 30.00, '2023-09-15'),
(4, 4, 4, 50.00, '2023-09-11'),
(5, 5, 5, 35.00, '2023-09-13'),
(6, 6, 6, 60.00, '2023-09-14'),
(7, 7, 7, 45.00, '2023-09-10'),
(8, 8, 8, 55.00, '2023-09-16'),
(9, 9, 9, 20.00, '2023-09-18'),
(10, 10, 10, 65.00, '2023-09-17');
```

100 % ▶ Messages

(10 rows affected)

Completion time: 2024-09-19T14:51:16.1129286+05:30

Task 2: Select, Where

Solve the following queries in the Schema that you have created above

1. List all customers:

```
select * from users;
```

100 %

	userID	name	email	password	contactNumber	address
1	1	Arjun	arjun@gmail.com	password123	1234567890	123 Street, City
2	2	Kaviya	kaviya@gmail.com	securepass	9876543210	456 Street, Town
3	3	Mani	mani@gmail.com	pass123	8765432109	789 Street, Village
4	4	Priya	priya@gmail.com	mysecurepass	7654321098	012 Street, County
5	5	Raj	raj@gmail.com	password456	6543210987	345 Street, District
6	6	Nisha	nisha@gmail.com	pass456	5432109876	678 Street, State
7	7	Ganesh	ganesh@gmail.com	securepassword	4321098765	901 Street, Country
8	8	Saranya	saranya@gmail.com	password789	3210987654	234 Street, Province
9	9	Kumar	kumar@gmail.com	mypass789	2109876543	567 Street, Region
10	10	Shalini	shalini@gmail.com	passwordabc	1098765432	890 Street, Territory

2. List all orders for a specific customer:

```
select * from Courier where receiverName='Michael Green' or senderName='Michael Green';
```

100 %

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate
1	3	David Lee	234 Cedar Street	Michael Green	567 Pine Blvd	3.00	Processing	TRACK003	2023-09-15
2	6	John Doe	123 Elm Street	Michael Green	567 Pine Blvd	5.00	Processing	TRACK006	2023-09-14

3. List all couriers:

```
--3)  
select * from Courier;
```

100 %

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate
1	1	John Doe	123 Elm Street	Jane Smith	456 Oak Avenue	2.50	Delivered	TRACK001	2023-09-10
2	2	Robert Brown	789 Maple Road	Emily Davis	101 Birch Lane	1.20	In Transit	TRACK002	2023-09-12
3	3	David Lee	234 Cedar Street	Michael Green	567 Pine Blvd	3.00	Processing	TRACK003	2023-09-15
4	4	Laura White	890 Spruce Way	Emma Wilson	321 Willow Court	2.70	Delivered	TRACK004	2023-09-11
5	5	William Taylor	654 Chestnut Circle	Olivia Adams	987 Aspen Drive	4.10	In Transit	TRACK005	2023-09-13
6	6	John Doe	123 Elm Street	Michael Green	567 Pine Blvd	5.00	Processing	TRACK006	2023-09-14
7	7	Emily Davis	101 Birch Lane	Robert Brown	789 Maple Road	3.50	Delivered	TRACK007	2023-09-10
8	8	Jane Smith	456 Oak Avenue	William Taylor	654 Chestnut Circle	2.90	Processing	TRACK008	2023-09-16
9	9	Laura White	890 Spruce Way	David Lee	234 Cedar Street	1.80	In Transit	TRACK009	2023-09-18
10	10	Emma Wilson	321 Willow Court	Olivia Adams	987 Aspen Drive	2.00	Delivered	TRACK010	2023-09-17

4. List all packages for a specific order:

```
--4)  
select * from Courier where courierID=5;
```

100 %

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate
1	5	William Taylor	654 Chestnut Circle	Olivia Adams	987 Aspen Drive	4.10	In Transit	TRACK005	2023-09-13

5. List all deliveries for a specific courier:

```
--5)
select courierID,trackingNumber, deliveryDate from Courier where status='Delivered';
100 %
```

	courierID	trackingNumber	deliveryDate
1	1	TRACK001	2023-09-10
2	4	TRACK004	2023-09-11
3	7	TRACK007	2023-09-10
4	10	TRACK010	2023-09-17

6. List all undelivered packages:

```
--6)
select * from Courier where status !='Delivered';
100 %
```

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate
1	2	Robert Brown	789 Maple Road	Emily Davis	101 Birch Lane	1.20	In Transit	TRACK002	2023-09-12
2	3	David Lee	234 Cedar Street	Michael Green	567 Pine Blvd	3.00	Processing	TRACK003	2023-09-15
3	5	William Taylor	654 Chestnut Circle	Olivia Adams	987 Aspen Drive	4.10	In Transit	TRACK005	2023-09-13
4	6	John Doe	123 Elm Street	Michael Green	567 Pine Blvd	5.00	Processing	TRACK006	2023-09-14
5	8	Jane Smith	456 Oak Avenue	William Taylor	654 Chestnut Circle	2.90	Processing	TRACK008	2023-09-16
6	9	Laura White	890 Spruce Way	David Lee	234 Cedar Street	1.80	In Transit	TRACK009	2023-09-18

7. List all packages that are scheduled for delivery today:

```
--7)
select * from Courier where deliveryDate='2023-09-11';
100 %
```

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate
1	4	Laura White	890 Spruce Way	Emma Wilson	321 Willow Court	2.70	Delivered	TRACK004	2023-09-11

8. List all packages with a specific status:

```
--8)
select * from Courier where status='In Transit';
100 %
```

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate
1	2	Robert Brown	789 Maple Road	Emily Davis	101 Birch Lane	1.20	In Transit	TRACK002	2023-09-12
2	5	William Taylor	654 Chestnut Circle	Olivia Adams	987 Aspen Drive	4.10	In Transit	TRACK005	2023-09-13
3	9	Laura White	890 Spruce Way	David Lee	234 Cedar Street	1.80	In Transit	TRACK009	2023-09-18

9. Calculate the total number of packages for each courier

```
--9)
SELECT courierID, COUNT(*) AS totalPackages FROM Courier GROUP BY courierID;
```

100 %

Results Messages

	courierID	totalPackages
1	1	1
2	2	1
3	3	1
4	4	1
5	5	1
6	6	1
7	7	1
8	8	1
9	9	1
10	10	1

10. Find the average delivery time for each courier

```
--10)
SELECT courierID, AVG(DATEDIFF(day, GETDATE(), DeliveryDate)) AS avgDeliveryTime FROM Courier GROUP BY courierID;
```

100 %

Results Messages

	courierID	avgDeliveryTime
1	1	-375
2	2	-373
3	3	-370
4	4	-374
5	5	-372
6	6	-371
7	7	-375
8	8	-369
9	9	-367
10	10	-368

11. List all packages with a specific weight range:

```
--11)
SELECT * FROM Courier WHERE weight BETWEEN 2.0 AND 5.0;
```

100 %

Results Messages

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate
1	1	John Doe	123 Elm Street	Jane Smith	456 Oak Avenue	2.50	Delivered	TRACK001	2023-09-10
2	3	David Lee	234 Cedar Street	Michael Green	567 Pine Blvd	3.00	Processing	TRACK003	2023-09-15
3	4	Laura White	890 Spruce Way	Emma Wilson	321 Willow Court	2.70	Delivered	TRACK004	2023-09-11
4	5	William Taylor	654 Chestnut Circle	Olivia Adams	987 Aspen Drive	4.10	In Transit	TRACK005	2023-09-13
5	6	John Doe	123 Elm Street	Michael Green	567 Pine Blvd	5.00	Processing	TRACK006	2023-09-14
6	7	Emily Davis	101 Birch Lane	Robert Brown	789 Maple Road	3.50	Delivered	TRACK007	2023-09-10
7	8	Jane Smith	456 Oak Avenue	William Taylor	654 Chestnut Circle	2.90	Processing	TRACK008	2023-09-16
8	10	Emma Wilson	321 Willow Court	Olivia Adams	987 Aspen Drive	2.00	Delivered	TRACK010	2023-09-17

12. Retrieve employees whose names contain '%John%'

```
--12)
SELECT * FROM Employee WHERE name LIKE '%John%';
100 %
```

Results Messages

	employeeID	name	email	contactNumber	role	salary
1	1	Alice Johnson	alice.johnson@example.com	555-1111	Courier	35000.00

13. Retrieve all courier records with payments greater than \$50.

```
--13)
SELECT Courier.* FROM Courier JOIN Payment ON Courier.courierID = Payment.courierID WHERE Payment.amount > 50;
100 %
```

Results Messages

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate
1	6	John Doe	123 Elm Street	Michael Green	567 Pine Blvd	5.00	Processing	TRACK006	2023-09-14
2	8	Jane Smith	456 Oak Avenue	William Taylor	654 Chestnut Circle	2.90	Processing	TRACK008	2023-09-16
3	10	Emma Wilson	321 Willow Court	Olivia Adams	987 Aspen Drive	2.00	Delivered	TRACK010	2023-09-17

Task 3: Group by, Aggregate Functions, Having, Order By, where

14. Find the total number of couriers handled by each employee.

```
--Task3:
--14)
SELECT employeeID, COUNT(courierID) AS totalCouriers FROM Courier GROUP BY employeeID;
100 %
```

Results Messages

	employeeID	totalCouriers
1	1	1
2	2	1
3	3	1
4	4	1
5	5	1
6	6	1
7	7	1
8	8	1
9	9	1
10	10	1

15. Calculate the total revenue generated by each location.

```
--15)
SELECT locationID, SUM(amount) AS totalRevenue FROM Payment GROUP BY locationID;
```

100 %

Results Messages

	locationID	totalRevenue
1	1	25.00
2	2	40.00
3	3	30.00
4	4	50.00
5	5	35.00
6	6	60.00
7	7	45.00
8	8	55.00
9	9	20.00
10	10	65.00

16. Find the total number of couriers delivered to each location.

```
--16)
SELECT 1.locationName, COUNT(p.courierID) AS TotalCouriersDelivered FROM Location 1 JOIN Payment p ON 1.locationID = p.locationID
GROUP BY 1.locationName;
```

100 %

Results Messages

	locationName	TotalCouriersDelivered
1	Boston Hub	1
2	Chicago Hub	1
3	Dallas Hub	1
4	Houston Hub	1
5	Los Angeles Hub	1
6	Miami Hub	1
7	New York Hub	1
8	Phoenix Hub	1
9	San Francisco Hub	1
10	Seattle Hub	1

17. Find the courier with the highest average delivery time.

```
--17)
SELECT TOP 1 courierID, AVG(DATEDIFF(day, deliveryDate, GETDATE())) AS deliveryTime FROM Courier WHERE deliveryDate IS NOT NULL
GROUP BY courierID ORDER BY deliveryTime DESC;
```

100 %

Results Messages

	courierID	deliveryTime
1	1	375

18. Find Locations with Total Payments Less Than a Certain Amount.

```
--18)
SELECT 1.locationName, SUM(p.amount) AS TotalPayments FROM Location 1 JOIN Payment p ON 1.locationID = p.locationID
GROUP BY 1.locationName HAVING SUM(p.amount) < 1000;
```

100 %

Results Messages

	locationName	TotalPayments
1	Boston Hub	65.00
2	Chicago Hub	30.00
3	Dallas Hub	45.00
4	Houston Hub	50.00
5	Los Angeles Hub	40.00
6	Miami Hub	55.00
7	New York Hub	25.00
8	Phoenix Hub	35.00
9	San Francisco Hub	60.00
10	Seattle Hub	20.00

19. Calculate Total Payments per Location.

```
--19
SELECT l.locationName, SUM(p.amount) AS TotalPayments FROM Location l JOIN Payment p ON l.locationID = p.locationID
GROUP BY l.locationName;
```

100 %

Results Messages

	locationName	TotalPayments
1	Boston Hub	65.00
2	Chicago Hub	30.00
3	Dallas Hub	45.00
4	Houston Hub	50.00
5	Los Angeles Hub	40.00
6	Miami Hub	55.00
7	New York Hub	25.00
8	Phoenix Hub	35.00
9	San Francisco Hub	60.00
10	Seattle Hub	20.00

20. Retrieve couriers who have received payments totaling more than \$10 in a specific location (LocationID = 5).

```
--20
SELECT c.courierID, c.trackingNumber, SUM(p.amount) AS TotalPayments FROM Courier c JOIN Payment p ON c.courierID = p.courierID
WHERE p.locationID = 5 GROUP BY c.courierID, c.trackingNumber HAVING SUM(p.amount) > 10;
```

00 %

Results Messages

	courierID	trackingNumber	TotalPayments
1	5	TRACK005	35.00

21. Retrieve couriers who have received payments totaling more than \$10 after a certain date (PaymentDate > '2023-09-11').

```
--21
SELECT c.courierID, c.trackingNumber, SUM(p.amount) AS TotalPayments FROM Courier c JOIN Payment p ON c.courierID = p.courierID
WHERE p.paymentDate > '2023-09-11' GROUP BY c.courierID, c.trackingNumber HAVING SUM(p.amount) > 10;
```

100 %

Results Messages

	courierID	trackingNumber	TotalPayments
1	2	TRACK002	40.00
2	3	TRACK003	30.00
3	5	TRACK005	35.00
4	6	TRACK006	60.00
5	8	TRACK008	55.00
6	9	TRACK009	20.00
7	10	TRACK010	65.00

22. Retrieve locations where the total amount received is more than \$30 before a certain date (PaymentDate > '2024-09-19').

```
--22
SELECT l.locationName, SUM(p.amount) AS TotalReceived FROM Location l JOIN Payment p ON l.locationID = p.locationID
WHERE p.paymentDate < '2024-09-19' GROUP BY l.locationName HAVING SUM(p.amount) > 30;
```

100 %

Results Messages

	locationName	TotalReceived
1	Boston Hub	65.00
2	Dallas Hub	45.00
3	Houston Hub	50.00
4	Los Angeles Hub	40.00
5	Miami Hub	55.00
6	Phoenix Hub	35.00
7	San Francisco Hub	60.00

Task 4: Inner Join, Full Outer Join, Cross Join, Left Outer Join, Right Outer Join

23. Retrieve Payments with Courier Information.

```
--23)
SELECT p.paymentID, p.amount, p.paymentDate, c.courierID, c.senderName, c.receiverName, c.weight, c.status, c.trackingNumber
FROM Payment p JOIN Courier c ON p.courierID = c.courierID;
```

100 %

Results Messages

	paymentID	amount	paymentDate	courierID	senderName	receiverName	weight	status	trackingNumber
1	1	25.00	2023-09-10	1	John Doe	Jane Smith	2.50	Delivered	TRACK001
2	2	40.00	2023-09-12	2	Robert Brown	Emily Davis	1.20	In Transit	TRACK002
3	3	30.00	2023-09-15	3	David Lee	Michael Green	3.00	Processing	TRACK003
4	4	50.00	2023-09-11	4	Laura White	Emma Wilson	2.70	Delivered	TRACK004
5	5	35.00	2023-09-13	5	William Taylor	Olivia Adams	4.10	In Transit	TRACK005
6	6	60.00	2023-09-14	6	John Doe	Michael Green	5.00	Processing	TRACK006
7	7	45.00	2023-09-10	7	Emily Davis	Robert Brown	3.50	Delivered	TRACK007
8	8	55.00	2023-09-16	8	Jane Smith	William Taylor	2.90	Processing	TRACK008
9	9	20.00	2023-09-18	9	Laura White	David Lee	1.80	In Transit	TRACK009
10	10	65.00	2023-09-17	10	Emma Wilson	Olivia Adams	2.00	Delivered	TRACK010

24. Retrieve Payments with Location Information.

```
--24)
SELECT p.paymentID, p.amount, p.paymentDate, l.locationID, l.locationName, l.address
FROM Payment p JOIN Location l ON p.locationID = l.locationID;
```

100 %

Results Messages

	paymentID	amount	paymentDate	locationID	locationName	address
1	1	25.00	2023-09-10	1	New York Hub	101 Manhattan St, New York, NY
2	2	40.00	2023-09-12	2	Los Angeles Hub	202 Hollywood Blvd, Los Angeles, CA
3	3	30.00	2023-09-15	3	Chicago Hub	303 Windy City Rd, Chicago, IL
4	4	50.00	2023-09-11	4	Houston Hub	404 Lone Star Ln, Houston, TX
5	5	35.00	2023-09-13	5	Phoenix Hub	505 Desert Ave, Phoenix, AZ
6	6	60.00	2023-09-14	6	San Francisco Hub	606 Bay St, San Francisco, CA
7	7	45.00	2023-09-10	7	Dallas Hub	707 Big D Blvd, Dallas, TX
8	8	55.00	2023-09-16	8	Miami Hub	808 Ocean Dr, Miami, FL
9	9	20.00	2023-09-18	9	Seattle Hub	909 Rainy St, Seattle, WA
10	10	65.00	2023-09-17	10	Boston Hub	1010 Freedom Trail, Boston, MA

25. Retrieve Payments with Courier and Location Information.

```
--25)
SELECT p.paymentID, p.amount, p.paymentDate, c.courierID, c.senderName, c.receiverName, l.locationID, l.locationName, l.address
FROM Payment p JOIN Courier c ON p.courierID = c.courierID JOIN Location l ON p.locationID = l.locationID;
```

100 %

Results Messages

	paymentID	amount	paymentDate	courierID	senderName	receiverName	locationID	locationName	address
1	1	25.00	2023-09-10	1	John Doe	Jane Smith	1	New York Hub	101 Manhattan St, New York, NY
2	2	40.00	2023-09-12	2	Robert Brown	Emily Davis	2	Los Angeles Hub	202 Hollywood Blvd, Los Angeles, CA
3	3	30.00	2023-09-15	3	David Lee	Michael Green	3	Chicago Hub	303 Windy City Rd, Chicago, IL
4	4	50.00	2023-09-11	4	Laura White	Emma Wilson	4	Houston Hub	404 Lone Star Ln, Houston, TX
5	5	35.00	2023-09-13	5	William Taylor	Olivia Adams	5	Phoenix Hub	505 Desert Ave, Phoenix, AZ
6	6	60.00	2023-09-14	6	John Doe	Michael Green	6	San Francisco Hub	606 Bay St, San Francisco, CA
7	7	45.00	2023-09-10	7	Emily Davis	Robert Brown	7	Dallas Hub	707 Big D Blvd, Dallas, TX
8	8	55.00	2023-09-16	8	Jane Smith	William Taylor	8	Miami Hub	808 Ocean Dr, Miami, FL
9	9	20.00	2023-09-18	9	Laura White	David Lee	9	Seattle Hub	909 Rainy St, Seattle, WA
10	10	65.00	2023-09-17	10	Emma Wilson	Olivia Adams	10	Boston Hub	1010 Freedom Trail, Boston, MA

26. List all payments with courier details.

```
--26)
SELECT p.paymentID, p.amount, p.paymentDate, c.trackingNumber, c.senderName, c.receiverName, c.status
FROM Payment p JOIN Courier c ON p.courierID = c.courierID;
```

100 %

Results Messages

	paymentID	amount	paymentDate	trackingNumber	senderName	receiverName	status
1	1	25.00	2023-09-10	TRACK001	John Doe	Jane Smith	Delivered
2	2	40.00	2023-09-12	TRACK002	Robert Brown	Emily Davis	In Transit
3	3	30.00	2023-09-15	TRACK003	David Lee	Michael Green	Processing
4	4	50.00	2023-09-11	TRACK004	Laura White	Emma Wilson	Delivered
5	5	35.00	2023-09-13	TRACK005	William Taylor	Olivia Adams	In Transit
6	6	60.00	2023-09-14	TRACK006	John Doe	Michael Green	Processing
7	7	45.00	2023-09-10	TRACK007	Emily Davis	Robert Brown	Delivered
8	8	55.00	2023-09-16	TRACK008	Jane Smith	William Taylor	Processing
9	9	20.00	2023-09-18	TRACK009	Laura White	David Lee	In Transit
10	10	65.00	2023-09-17	TRACK010	Emma Wilson	Olivia Adams	Delivered

27. Total payments received for each courier.

```
--27)
SELECT c.courierID, c.trackingNumber, SUM(p.amount) AS totalPayments
FROM Courier c JOIN Payment p ON c.courierID = p.courierID GROUP BY c.courierID, c.trackingNumber;
```

100 %

Results Messages

	courierID	trackingNumber	totalPayments
1	1	TRACK001	25.00
2	2	TRACK002	40.00
3	3	TRACK003	30.00
4	4	TRACK004	50.00
5	5	TRACK005	35.00
6	6	TRACK006	60.00
7	7	TRACK007	45.00
8	8	TRACK008	55.00
9	9	TRACK009	20.00
10	10	TRACK010	65.00

28. List payments made on a specific date.

```
--28)
SELECT * FROM Payment WHERE paymentDate = '2023-09-10';
```

100 %

Results Messages

	paymentID	courierID	locationID	amount	paymentDate
1	1	1	1	25.00	2023-09-10
2	7	7	7	45.00	2023-09-10

29. Get Courier Information for Each Payment.

```
--29)
SELECT p.paymentID, p.courierID, c.senderName, c.receiverName, c.weight, c.status, c.trackingNumber, c.deliveryDate
FROM Payment p JOIN Courier c ON p.courierID = c.courierID;
```

100 %

Results Messages

	paymentID	courierID	senderName	receiverName	weight	status	trackingNumber	deliveryDate
1	1	1	John Doe	Jane Smith	2.50	Delivered	TRACK001	2023-09-10
2	2	2	Robert Brown	Emily Davis	1.20	In Transit	TRACK002	2023-09-12
3	3	3	David Lee	Michael Green	3.00	Processing	TRACK003	2023-09-15
4	4	4	Laura White	Emma Wilson	2.70	Delivered	TRACK004	2023-09-11
5	5	5	William Taylor	Olivia Adams	4.10	In Transit	TRACK005	2023-09-13
6	6	6	John Doe	Michael Green	5.00	Processing	TRACK006	2023-09-14
7	7	7	Emily Davis	Robert Brown	3.50	Delivered	TRACK007	2023-09-10
8	8	8	Jane Smith	William Taylor	2.90	Processing	TRACK008	2023-09-16
9	9	9	Laura White	David Lee	1.80	In Transit	TRACK009	2023-09-18
10	10	10	Emma Wilson	Olivia Adams	2.00	Delivered	TRACK010	2023-09-17

30. Get Payment Details with Location.

```
--30)
SELECT p.paymentID, p.courierID, p.amount, p.paymentDate, l.locationName, l.address
FROM Payment p JOIN Location l ON p.locationID = l.locationID;
```

100 %

Results Messages

	paymentID	courierID	amount	paymentDate	locationName	address
1	1	1	25.00	2023-09-10	New York Hub	101 Manhattan St, New York, NY
2	2	2	40.00	2023-09-12	Los Angeles Hub	202 Hollywood Blvd, Los Angeles, CA
3	3	3	30.00	2023-09-15	Chicago Hub	303 Windy City Rd, Chicago, IL
4	4	4	50.00	2023-09-11	Houston Hub	404 Lone Star Ln, Houston, TX
5	5	5	35.00	2023-09-13	Phoenix Hub	505 Desert Ave, Phoenix, AZ
6	6	6	60.00	2023-09-14	San Francisco Hub	606 Bay St, San Francisco, CA
7	7	7	45.00	2023-09-10	Dallas Hub	707 Big D Blvd, Dallas, TX
8	8	8	55.00	2023-09-16	Miami Hub	808 Ocean Dr, Miami, FL
9	9	9	20.00	2023-09-18	Seattle Hub	909 Rainy St, Seattle, WA
10	10	10	65.00	2023-09-17	Boston Hub	1010 Freedom Trail, Boston, MA

31. Calculating Total Payments for Each Courier.

```
--31)
SELECT c.courierID, c.trackingNumber, SUM(p.amount) AS totalPayments
FROM Courier c JOIN Payment p ON c.courierID = p.courierID GROUP BY c.courierID, c.trackingNumber;
```

100 %

Results Messages

	courierID	trackingNumber	totalPayments
1	1	TRACK001	25.00
2	2	TRACK002	40.00
3	3	TRACK003	30.00
4	4	TRACK004	50.00
5	5	TRACK005	35.00
6	6	TRACK006	60.00
7	7	TRACK007	45.00
8	8	TRACK008	55.00
9	9	TRACK009	20.00
10	10	TRACK010	65.00

32. List Payments Within a Date Range.

```
--32)
SELECT * FROM Payment WHERE paymentDate BETWEEN '2023-09-01' AND '2023-09-30';
```

100 %

Results Messages

	paymentID	courierID	locationID	amount	paymentDate
1	1	1	1	25.00	2023-09-10
2	2	2	2	40.00	2023-09-12
3	3	3	3	30.00	2023-09-15
4	4	4	4	50.00	2023-09-11
5	5	5	5	35.00	2023-09-13
6	6	6	6	60.00	2023-09-14
7	7	7	7	45.00	2023-09-10
8	8	8	8	55.00	2023-09-16
9	9	9	9	20.00	2023-09-18
10	10	10	10	65.00	2023-09-17

33. Retrieve a list of all users and their corresponding courier records, including cases where there are no matches on either side.

```
--33)
SELECT u.userID, u.name, c.courierID, c.senderName, c.receiverName FROM Users u LEFT JOIN Courier c ON u.userID = c.courierID;
```

100 %

Results Messages

	userID	name	courierID	senderName	receiverName
1	1	Arun	1	John Doe	Jane Smith
2	2	Kavya	2	Robert Brown	Emily Davis
3	3	Mani	3	David Lee	Michael Green
4	4	Priya	4	Laura White	Emma Wilson
5	5	Raj	5	William Taylor	Olivia Adams
6	6	Nisha	6	John Doe	Michael Green
7	7	Ganesh	7	Emily Davis	Robert Brown
8	8	Saranya	8	Jane Smith	William Taylor
9	9	Kumar	9	Laura White	David Lee
10	10	Shalini	10	Emma Wilson	Olivia Adams

34. Retrieve a list of all couriers and their corresponding services, including cases where there are no matches on either side.

```
--34)
SELECT c.courierID, c.trackingNumber, s.serviceName, s.cost
FROM Courier c LEFT JOIN CourierServices s ON c.courierID = s.serviceID;
```

100 %

Results Messages

	courierID	trackingNumber	serviceName	cost
1	1	TRACK001	Standard Delivery	10.00
2	2	TRACK002	Express Delivery	20.00
3	3	TRACK003	Same Day Delivery	25.00
4	4	TRACK004	Overnight Delivery	30.00
5	5	TRACK005	Weekend Delivery	35.00
6	6	TRACK006	International Shipping	50.00
7	7	TRACK007	Economy Delivery	8.00
8	8	TRACK008	Next Day Delivery	18.00
9	9	TRACK009	Two-Day Shipping	22.00
10	10	TRACK010	Freight Delivery	75.00

35. Retrieve a list of all employees and their corresponding payments, including cases where there are no matches on either side.

```
--35
SELECT e.employeeID, e.name, p.paymentID, p.amount FROM Employee e LEFT JOIN Payment p ON e.employeeID = p.courierID;
```

	employeeID	name	paymentID	amount
1	1	Alice Johnson	1	25.00
2	2	Bob Carter	2	40.00
3	3	Charlie Young	3	30.00
4	4	Diana Scott	4	50.00
5	5	Ethan Ford	5	35.00
6	6	Fiona Blake	6	60.00
7	7	George Hall	7	45.00
8	8	Helen Moore	8	55.00
9	9	Ian Baker	9	20.00
10	10	Jenny Watts	10	65.00

36. List all users and all courier services, showing all possible combinations.

```
--36
SELECT u.name AS userName, s.serviceName FROM Users u CROSS JOIN CourierServices s;
```

	userName	serviceName
1	Arjun	Standard Delivery
2	Kaviya	Standard Delivery
3	Mani	Standard Delivery
4	Priya	Standard Delivery
5	Raj	Standard Delivery
6	Nisha	Standard Delivery
7	Ganesh	Standard Delivery
8	Saranya	Standard Delivery
9	Kumar	Standard Delivery
10	Shalini	Standard Delivery
11	Arjun	Express Delivery
12	Kaviya	Express Delivery
13	Mani	Express Delivery
14	Priya	Express Delivery
15	Raj	Express Delivery
16	Nisha	Express Delivery
17	Ganesh	Express Delivery
18	Saranya	Express Delivery
19	Kumar	Express Delivery
20	Shalini	Express Delivery
21	Arjun	Same Day Deliv...
22	Kaviya	Same Day Deliv...
23	Mani	Same Day Deliv...
24	Priya	Same Day Deliv...
25	Raj	Same Day Deliv...

	userName	serviceName
26	Nisha	Same Day Deliv...
27	Ganesh	Same Day Deliv...
28	Saranya	Same Day Deliv...
29	Kumar	Same Day Deliv...
30	Shalini	Same Day Deliv...
31	Arjun	Overnight Deliv...
32	Kaviya	Overnight Deliv...
33	Mani	Overnight Deliv...
34	Priya	Overnight Deliv...
35	Raj	Overnight Deliv...
36	Nisha	Overnight Deliv...
37	Ganesh	Overnight Deliv...
38	Saranya	Overnight Deliv...
39	Kumar	Overnight Deliv...
40	Shalini	Overnight Deliv...
41	Arjun	Weekend Deliv...
42	Kaviya	Weekend Deliv...
43	Mani	Weekend Deliv...
44	Priya	Weekend Deliv...
45	Raj	Weekend Deliv...
46	Nisha	Weekend Deliv...
47	Ganesh	Weekend Deliv...
48	Saranya	Weekend Deliv...
49	Kumar	Weekend Deliv...
50	Shalini	Weekend Deliv...

	userName	serviceName
51	Arjun	International Sh...
52	Kaviya	International Sh...
53	Mani	International Sh...
54	Priya	International Sh...
55	Raj	International Sh...
56	Nisha	International Sh...
57	Ganesh	International Sh...
58	Saranya	International Sh...
59	Kumar	International Sh...
60	Shalini	International Sh...
61	Arjun	Economy Delivery
62	Kaviya	Economy Delivery
63	Mani	Economy Delivery
64	Priya	Economy Delivery
65	Raj	Economy Delivery
66	Nisha	Economy Delivery
67	Ganesh	Economy Delivery
68	Saranya	Economy Delivery
69	Kumar	Economy Delivery
70	Shalini	Economy Delivery
71	Arjun	Next Day Delivery
72	Kaviya	Next Day Delivery
73	Mani	Next Day Delivery
74	Priya	Next Day Delivery
75	Raj	Next Day Delivery

	userName	serviceName
76	Nisha	Next Day Delivery
77	Ganesh	Next Day Delivery
78	Saranya	Next Day Delivery
79	Kumar	Next Day Delivery
80	Shalini	Next Day Delivery
81	Arjun	Two-Day Shippi...
82	Kaviya	Two-Day Shippi...
83	Mani	Two-Day Shippi...
84	Priya	Two-Day Shippi...
85	Raj	Two-Day Shippi...
86	Nisha	Two-Day Shippi...
87	Ganesh	Two-Day Shippi...
88	Saranya	Two-Day Shippi...
89	Kumar	Two-Day Shippi...
90	Shalini	Two-Day Shippi...
91	Arjun	Freight Delivery
92	Kaviya	Freight Delivery
93	Mani	Freight Delivery
94	Priya	Freight Delivery
95	Raj	Freight Delivery
96	Nisha	Freight Delivery
97	Ganesh	Freight Delivery
98	Saranya	Freight Delivery
99	Kumar	Freight Delivery
100	Shalini	Freight Delivery

37. List all employees and all locations, showing all possible combinations.

```
--37)
SELECT e.name AS employeeName, l.locationName FROM Employee e CROSS JOIN Location l;
```

	employeeName	locationName
1	Alice Johnson	New York Hub
2	Bob Carter	New York Hub
3	Charlie Young	New York Hub
4	Diana Scott	New York Hub
5	Ethan Ford	New York Hub
6	Fiona Blake	New York Hub
7	George Hall	New York Hub
8	Helen Moore	New York Hub
9	Ian Baker	New York Hub
10	Jenny Watts	New York Hub
11	Alice Johnson	Los Angeles Hub
12	Bob Carter	Los Angeles Hub
13	Charlie Young	Los Angeles Hub
14	Diana Scott	Los Angeles Hub
15	Ethan Ford	Los Angeles Hub
16	Fiona Blake	Los Angeles Hub
17	George Hall	Los Angeles Hub
18	Helen Moore	Los Angeles Hub
19	Ian Baker	Los Angeles Hub
20	Jenny Watts	Los Angeles Hub
21	Alice Johnson	Chicago Hub
22	Bob Carter	Chicago Hub
23	Charlie Young	Chicago Hub
24	Diana Scott	Chicago Hub
25	Ethan Ford	Chicago Hub

	employeeName	locationName
26	Fiona Blake	Chicago Hub
27	George Hall	Chicago Hub
28	Helen Moore	Chicago Hub
29	Ian Baker	Chicago Hub
30	Jenny Watts	Chicago Hub
31	Alice Johnson	Houston Hub
32	Bob Carter	Houston Hub
33	Charlie Young	Houston Hub
34	Diana Scott	Houston Hub
35	Ethan Ford	Houston Hub
36	Fiona Blake	Houston Hub
37	George Hall	Houston Hub
38	Helen Moore	Houston Hub
39	Ian Baker	Houston Hub
40	Jenny Watts	Houston Hub
41	Alice Johnson	Phoenix Hub
42	Bob Carter	Phoenix Hub
43	Charlie Young	Phoenix Hub
44	Diana Scott	Phoenix Hub
45	Ethan Ford	Phoenix Hub
46	Fiona Blake	Phoenix Hub
47	George Hall	Phoenix Hub
48	Helen Moore	Phoenix Hub
49	Ian Baker	Phoenix Hub
50	Jenny Watts	Phoenix Hub

	employeeName	locationName
51	Alice Johnson	San Francisco ...
52	Bob Carter	San Francisco ...
53	Charlie Young	San Francisco ...
54	Diana Scott	San Francisco ...
55	Ethan Ford	San Francisco ...
56	Fiona Blake	San Francisco ...
57	George Hall	San Francisco ...
58	Helen Moore	San Francisco ...
59	Ian Baker	San Francisco ...
60	Jenny Watts	San Francisco ...
61	Alice Johnson	Dallas Hub
62	Bob Carter	Dallas Hub
63	Charlie Young	Dallas Hub
64	Diana Scott	Dallas Hub
65	Ethan Ford	Dallas Hub
66	Fiona Blake	Dallas Hub
67	George Hall	Dallas Hub
68	Helen Moore	Dallas Hub
69	Ian Baker	Dallas Hub
70	Jenny Watts	Dallas Hub
71	Alice Johnson	Miami Hub
72	Bob Carter	Miami Hub
73	Charlie Young	Miami Hub
74	Diana Scott	Miami Hub
75	Ethan Ford	Miami Hub

	employeeName	locationName
76	Fiona Blake	Miami Hub
77	George Hall	Miami Hub
78	Helen Moore	Miami Hub
79	Ian Baker	Miami Hub
80	Jenny Watts	Miami Hub
81	Alice Johnson	Seattle Hub
82	Bob Carter	Seattle Hub
83	Charlie Young	Seattle Hub
84	Diana Scott	Seattle Hub
85	Ethan Ford	Seattle Hub
86	Fiona Blake	Seattle Hub
87	George Hall	Seattle Hub
88	Helen Moore	Seattle Hub
89	Ian Baker	Seattle Hub
90	Jenny Watts	Seattle Hub
91	Alice Johnson	Boston Hub
92	Bob Carter	Boston Hub
93	Charlie Young	Boston Hub
94	Diana Scott	Boston Hub
95	Ethan Ford	Boston Hub
96	Fiona Blake	Boston Hub
97	George Hall	Boston Hub
98	Helen Moore	Boston Hub
99	Ian Baker	Boston Hub
100	Jenny Watts	Boston Hub

38. Retrieve a list of couriers and their corresponding sender information (if available).

--38)
`SELECT c.courierID, c.senderName, c.senderAddress FROM Courier c;`

100 %

Results Messages

	courierID	senderName	senderAddress
1	1	John Doe	123 Elm Street
2	2	Robert Brown	789 Maple Road
3	3	David Lee	234 Cedar Street
4	4	Laura White	890 Spruce Way
5	5	William Taylor	654 Chestnut Circle
6	6	John Doe	123 Elm Street
7	7	Emily Davis	101 Birch Lane
8	8	Jane Smith	456 Oak Avenue
9	9	Laura White	890 Spruce Way
10	10	Emma Wilson	321 Willow Court

39. Retrieve a list of couriers and their corresponding receiver information (if available).

```
--39)
SELECT c.courierID, c.receiverName, c.receiverAddress FROM Courier c;
```

100 %

Results Messages

	courierID	receiverName	receiverAddress
1	1	Jane Smith	456 Oak Avenue
2	2	Emily Davis	101 Birch Lane
3	3	Michael Green	567 Pine Blvd
4	4	Emma Wilson	321 Willow Court
5	5	Olivia Adams	987 Aspen Drive
6	6	Michael Green	567 Pine Blvd
7	7	Robert Brown	789 Maple Road
8	8	William Taylor	654 Chestnut Circle
9	9	David Lee	234 Cedar Street
10	10	Olivia Adams	987 Aspen Drive

40. Retrieve a list of couriers along with the courier service details (if available).

```
--40)
SELECT c.courierID, c.trackingNumber, s.serviceName, s.cost
FROM Courier c LEFT JOIN CourierServices s ON c.courierID = s.serviceID;
```

100 %

Results Messages

	courierID	trackingNumber	serviceName	cost
1	1	TRACK001	Standard Delivery	10.00
2	2	TRACK002	Express Delivery	20.00
3	3	TRACK003	Same Day Delivery	25.00
4	4	TRACK004	Overnight Delivery	30.00
5	5	TRACK005	Weekend Delivery	35.00
6	6	TRACK006	International Shipping	50.00
7	7	TRACK007	Economy Delivery	8.00
8	8	TRACK008	Next Day Delivery	18.00
9	9	TRACK009	Two-Day Shipping	22.00
10	10	TRACK010	Freight Delivery	75.00

41. Retrieve a list of employees and the number of couriers assigned to each employee.

```
--41)
SELECT e.employeeID, e.name, COUNT(c.courierID) AS totalCouriers
FROM Employee e LEFT JOIN Courier c ON e.employeeID = c.employeeID GROUP BY e.employeeID, e.name;
```

100 %

Results Messages

	employeeID	name	totalCouriers
1	1	Alice Johnson	1
2	2	Bob Carter	1
3	3	Charlie Young	1
4	4	Diana Scott	1
5	5	Ethan Ford	1
6	6	Fiona Blake	1
7	7	George Hall	1
8	8	Helen Moore	1
9	9	Ian Baker	1
10	10	Jenny Watts	1

42. Retrieve a list of locations and the total payment amount received at each location.

```
--42)
SELECT l.locationName, SUM(p.amount) AS totalAmount
FROM Location l JOIN Payment p ON l.locationID = p.locationID GROUP BY l.locationName;
```

100 %

Results Messages

	locationName	totalAmount
1	Boston Hub	65.00
2	Chicago Hub	30.00
3	Dallas Hub	45.00
4	Houston Hub	50.00
5	Los Angeles Hub	40.00
6	Miami Hub	55.00
7	New York Hub	25.00
8	Phoenix Hub	35.00
9	San Francisco Hub	60.00
10	Seattle Hub	20.00

43. Retrieve all couriers sent by the same sender (based on SenderName).

```
--43)
SELECT c1.courierID, c1.senderName, c1.senderAddress, c1.receiverName, c1.receiverAddress, c1.weight, c1.status,
c1.trackingNumber, c1.deliveryDate, c1.employeeID
FROM Courier c1 INNER JOIN Courier c2 ON c1.senderName = c2.senderName
WHERE c1.courierID <> c2.courierID;
```

100 %

Results Messages

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate	employeeID
1	1	John Doe	123 Elm Street	Jane Smith	456 Oak Avenue	2.50	Delivered	TRACK001	2023-09-10	1
2	4	Laura White	890 Spruce Way	Emma Wilson	321 Willow Court	2.70	Delivered	TRACK004	2023-09-11	4
3	6	John Doe	123 Elm Street	Michael Green	567 Pine Blvd	5.00	Processing	TRACK006	2023-09-14	6
4	9	Laura White	890 Spruce Way	David Lee	234 Cedar Street	1.80	In Transit	TRACK009	2023-09-18	9

44. List all employees who share the same role.

```
--44)
SELECT name, role FROM Employee GROUP BY role, name HAVING COUNT(*) > 1;
```

100 %

Results Messages

name	role
------	------

45. Retrieve all payments made for couriers sent from the same location.

```
--45)
SELECT p.paymentID, p.amount, c.senderAddress
FROM Payment p JOIN Courier c ON p.courierID = c.courierID WHERE c.senderAddress LIKE '123 Elm Street';
```

100 %

Results Messages

paymentID	amount	senderAddress
1	25.00	123 Elm Street
2	60.00	123 Elm Street

46. Retrieve all couriers sent from the same location (based on SenderAddress).

```
--46)
SELECT * FROM Courier WHERE senderAddress LIKE '123 Elm Street';
```

100 %

Results Messages

courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate	employeeID
1	John Doe	123 Elm Street	Jane Smith	456 Oak Avenue	2.50	Delivered	TRACK001	2023-09-10	1
2	John Doe	123 Elm Street	Michael Green	567 Pine Blvd	5.00	Processing	TRACK006	2023-09-14	6

47. List employees and the number of couriers they have delivered.

```
--47)
SELECT e.employeeID, e.name, COUNT(c.courierID) AS totalDelivered
FROM Employee e JOIN Courier c ON e.employeeID = c.employeeID WHERE c.status = 'Delivered' GROUP BY e.employeeID, e.name;
```

100 %

Results Messages

employeeID	name	totalDelivered
1	Alice Johnson	1
2	Diana Scott	1
3	George Hall	1
4	Jenny Watts	1

48. Find couriers that were paid an amount greater than the cost of their respective courier services.

```
--48)
SELECT c.courierID, c.trackingNumber, p.amount, s.cost
FROM Courier c
JOIN Payment p ON c.courierID = p.courierID
JOIN CourierServices s ON c.courierID = s.serviceID WHERE p.amount > s.cost;
```

100 %

Results Messages

courierID	trackingNumber	amount	cost
1	TRACK001	25.00	10.00
2	TRACK002	40.00	20.00
3	TRACK003	30.00	25.00
4	TRACK004	50.00	30.00
5	TRACK006	60.00	50.00
6	TRACK007	45.00	8.00
7	TRACK008	55.00	18.00

Scope: Inner Queries, Non Equi Joins, Equi joins, Exist, Any, All

49. Find couriers that have a weight greater than the average weight of all couriers.

```
--49)
SELECT * Courier WHERE weight > (SELECT AVG(weight) FROM Courier);
```

100 %

Results Messages

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate	employeeID
1	3	David Lee	234 Cedar Street	Michael Green	567 Pine Blvd	3.00	Processing	TRACK003	2023-09-15	3
2	5	William Taylor	654 Chestnut Circle	Olivia Adams	987 Aspen Drive	4.10	In Transit	TRACK005	2023-09-13	5
3	6	John Doe	123 Elm Street	Michael Green	567 Pine Blvd	5.00	Processing	TRACK006	2023-09-14	6
4	7	Emily Davis	101 Birch Lane	Robert Brown	789 Maple Road	3.50	Delivered	TRACK007	2023-09-10	7
5	8	Jane Smith	456 Oak Avenue	William Taylor	654 Chestnut Circle	2.90	Processing	TRACK008	2023-09-16	8

50. Find the names of all employees who have a salary greater than the average salary.

```
--50)
SELECT name, salary FROM Employee WHERE salary > (SELECT AVG(salary) FROM Employee);
```

100 %

Results Messages

	name	salary
1	Bob Carter	55000.00
2	Ethan Ford	45000.00
3	George Hall	48000.00
4	Helen Moore	55000.00

51. Find the total cost of all courier services where the cost is less than the maximum cost.

```
--51)
SELECT SUM(cost) AS totalCost FROM CourierServices WHERE cost < (SELECT MAX(cost) FROM CourierServices);
```

100 %

Results Messages

	totalCost
1	218.00

52. Find all couriers that have been paid for.

```
--52)
SELECT DISTINCT c.courierID, c.trackingNumber FROM Courier c JOIN Payment p ON c.courierID = p.courierID;
```

100 %

Results Messages

	courierID	trackingNumber
1	1	TRACK001
2	2	TRACK002
3	3	TRACK003
4	4	TRACK004
5	5	TRACK005
6	6	TRACK006
7	7	TRACK007
8	8	TRACK008
9	9	TRACK009
10	10	TRACK010

53. Find the locations where the maximum payment amount was made.

```
--53)
SELECT locationID, MAX(amount) AS maxPayment FROM Payment GROUP BY locationID;
```

100 %

Results Messages

	locationID	maxPayment
1	1	25.00
2	2	40.00
3	3	30.00
4	4	50.00
5	5	35.00
6	6	60.00
7	7	45.00
8	8	55.00
9	9	20.00
10	10	65.00

54. Find all couriers whose weight is greater than the weight of all couriers sent by a specific sender (e.g., 'SenderName').

```
--54)
SELECT * FROM Courier c1
WHERE c1.weight > (SELECT MAX(c2.weight) FROM Courier c2 WHERE c2.senderName = 'Laura White');
```

100 %

Results Messages

	courierID	senderName	senderAddress	receiverName	receiverAddress	weight	status	trackingNumber	deliveryDate	employeeID
1	3	David Lee	234 Cedar Street	Michael Green	567 Pine Blvd	3.00	Processing	TRACK003	2023-09-15	3
2	5	William Taylor	654 Chestnut Circle	Olivia Adams	987 Aspen Drive	4.10	In Transit	TRACK005	2023-09-13	5
3	6	John Doe	123 Elm Street	Michael Green	567 Pine Blvd	5.00	Processing	TRACK006	2023-09-14	6
4	7	Emily Davis	101 Birch Lane	Robert Brown	789 Maple Road	3.50	Delivered	TRACK007	2023-09-10	7
5	8	Jane Smith	456 Oak Avenue	William Taylor	654 Chestnut Circle	2.90	Processing	TRACK008	2023-09-16	8

CODE:

```
use Courier_management_assignment;
```

```
CREATE TABLE users (
```

```
    userID INT PRIMARY KEY,
```

```
    name VARCHAR(255),
```

```
    email VARCHAR(255) UNIQUE,
```

```
    password VARCHAR(255),
```

```
    contactNumber VARCHAR(20),
```

```
    address TEXT
```

```
);
```

```
CREATE TABLE Courier (
```

```
    courierID INT PRIMARY KEY,
```

```
    senderName VARCHAR(255),
```

```
    senderAddress TEXT,
```

```
    receiverName VARCHAR(255),
```

```
    receiverAddress TEXT,
```

```
    weight DECIMAL(5, 2),
```

```
    status VARCHAR(50),
```

```
    trackingNumber VARCHAR(20) UNIQUE,
```

```
    deliveryDate DATE
```

```
);
```

```
ALTER TABLE Courier ADD employeeID INT;
```

```
UPDATE Courier SET employeeID = 1 WHERE courierID = 1;
```

```
UPDATE Courier SET employeeID = 2 WHERE courierID = 2;
```

```
UPDATE Courier SET employeeID = 3 WHERE courierID = 3;
```

```
UPDATE Courier SET employeeID = 4 WHERE courierID = 4;
```

```
UPDATE Courier SET employeeID = 5 WHERE courierID = 5;  
UPDATE Courier SET employeeID = 6 WHERE courierID = 6;  
UPDATE Courier SET employeeID = 7 WHERE courierID = 7;  
UPDATE Courier SET employeeID = 8 WHERE courierID = 8;  
UPDATE Courier SET employeeID = 9 WHERE courierID = 9;  
UPDATE Courier SET employeeID = 10 WHERE courierID = 10;
```

```
select * from Courier;
```

```
CREATE TABLE CourierServices (  
    serviceID INT PRIMARY KEY,  
    serviceName VARCHAR(100),  
    cost DECIMAL(8, 2)  
)
```

```
CREATE TABLE Employee (  
    employeeID INT PRIMARY KEY,  
    name VARCHAR(255),  
    email VARCHAR(255) UNIQUE,  
    contactNumber VARCHAR(20),  
    role VARCHAR(50),  
    salary DECIMAL(10, 2)  
)
```

```
CREATE TABLE Location (  
    locationID INT PRIMARY KEY,  
    locationName VARCHAR(100),  
    address TEXT
```

);

```
CREATE TABLE Payment (
    paymentID INT PRIMARY KEY,
    courierID INT,
    locationID INT,
    amount DECIMAL(10, 2),
    paymentDate DATE,
    FOREIGN KEY (courierID) REFERENCES Courier(courierID),
    FOREIGN KEY (locationID) REFERENCES Location(locationID)
);
```

```
INSERT INTO users(userID, name, email, password, contactNumber, address)
VALUES
(1, 'Arjun', 'arjun@gmail.com', 'password123', '1234567890', '123 Street, City'),
(2, 'Kaviya', 'kaviya@gmail.com', 'securepass', '9876543210', '456 Street, Town'),
(3, 'Mani', 'mani@gmail.com', 'pass123', '8765432109', '789 Street, Village'),
(4, 'Priya', 'priya@gmail.com', 'mysecurepass', '7654321098', '012 Street, County'),
(5, 'Raj', 'raj@gmail.com', 'password456', '6543210987', '345 Street, District'),
(6, 'Nisha', 'nisha@gmail.com', 'pass456', '5432109876', '678 Street, State'),
(7, 'Ganesh', 'ganesh@gmail.com', 'securepassword', '4321098765', '901 Street, Country'),
(8, 'Saranya', 'saranya@gmail.com', 'password789', '3210987654', '234 Street, Province'),
(9, 'Kumar', 'kumar@gmail.com', 'mypass789', '2109876543', '567 Street, Region'),
(10, 'Shalini', 'shalini@gmail.com', 'passwordabc', '1098765432', '890 Street, Territory');
```

SELECT * FROM Users;

INSERT INTO Courier (courierID, senderName, senderAddress, receiverName, receiverAddress, weight, status, trackingNumber, deliveryDate)

VALUES

(1, 'John Doe', '123 Elm Street', 'Jane Smith', '456 Oak Avenue', 2.5, 'Delivered', 'TRACK001', '2023-09-10'),

(2, 'Robert Brown', '789 Maple Road', 'Emily Davis', '101 Birch Lane', 1.2, 'In Transit', 'TRACK002', '2024-09-19'),

(3, 'David Lee', '234 Cedar Street', 'Michael Green', '567 Pine Blvd', 3.0, 'Processing', 'TRACK003', '2023-09-15'),

(4, 'Laura White', '890 Spruce Way', 'Emma Wilson', '321 Willow Court', 2.7, 'Delivered', 'TRACK004', '2023-09-11'),

(5, 'William Taylor', '654 Chestnut Circle', 'Olivia Adams', '987 Aspen Drive', 4.1, 'In Transit', 'TRACK005', '2023-09-13'),

(6, 'John Doe', '123 Elm Street', 'Michael Green', '567 Pine Blvd', 5.0, 'Processing', 'TRACK006', '2023-09-14'),

(7, 'Emily Davis', '101 Birch Lane', 'Robert Brown', '789 Maple Road', 3.5, 'Delivered', 'TRACK007', '2023-09-10'),

(8, 'Jane Smith', '456 Oak Avenue', 'William Taylor', '654 Chestnut Circle', 2.9, 'Processing', 'TRACK008', '2023-09-16'),

(9, 'Laura White', '890 Spruce Way', 'David Lee', '234 Cedar Street', 1.8, 'In Transit', 'TRACK009', '2023-09-18'),

(10, 'Emma Wilson', '321 Willow Court', 'Olivia Adams', '987 Aspen Drive', 2.0, 'Delivered', 'TRACK010', '2023-09-17');

INSERT INTO CourierServices (serviceID, serviceName, cost)

VALUES

(1, 'Standard Delivery', 10.00),

(2, 'Express Delivery', 20.00),

(3, 'Same Day Delivery', 25.00),

(4, 'Overnight Delivery', 30.00),

(5, 'Weekend Delivery', 35.00),

(6, 'International Shipping', 50.00),

(7, 'Economy Delivery', 8.00),

(8, 'Next Day Delivery', 18.00),
(9, 'Two-Day Shipping', 22.00),
(10, 'Freight Delivery', 75.00);

INSERT INTO Employee (employeeID, name, email, contactNumber, role, salary)
VALUES

(1, 'Alice Johnson', 'alice.johnson@example.com', '555-1111', 'Courier', 35000.00),
(2, 'Bob Carter', 'bob.carter@example.com', '555-2222', 'Manager', 55000.00),
(3, 'Charlie Young', 'charlie.young@example.com', '555-3333', 'Dispatcher', 40000.00),
(4, 'Diana Scott', 'diana.scott@example.com', '555-4444', 'Courier', 35000.00),
(5, 'Ethan Ford', 'ethan.ford@example.com', '555-5555', 'Admin', 45000.00),
(6, 'Fiona Blake', 'fiona.blake@example.com', '555-6666', 'Courier', 36000.00),
(7, 'George Hall', 'george.hall@example.com', '555-7777', 'Supervisor', 48000.00),
(8, 'Helen Moore', 'helen.moore@example.com', '555-8888', 'Manager', 55000.00),
(9, 'Ian Baker', 'ian.baker@example.com', '555-9999', 'Courier', 34000.00),
(10, 'Jenny Watts', 'jenny.watts@example.com', '555-1010', 'Dispatcher', 41000.00);

INSERT INTO Location (locationID, locationName, address)
VALUES

(1, 'New York Hub', '101 Manhattan St, New York, NY'),
(2, 'Los Angeles Hub', '202 Hollywood Blvd, Los Angeles, CA'),
(3, 'Chicago Hub', '303 Windy City Rd, Chicago, IL'),
(4, 'Houston Hub', '404 Lone Star Ln, Houston, TX'),
(5, 'Phoenix Hub', '505 Desert Ave, Phoenix, AZ'),
(6, 'San Francisco Hub', '606 Bay St, San Francisco, CA'),
(7, 'Dallas Hub', '707 Big D Blvd, Dallas, TX'),
(8, 'Miami Hub', '808 Ocean Dr, Miami, FL'),
(9, 'Seattle Hub', '909 Rainy St, Seattle, WA');

(10, 'Boston Hub', '1010 Freedom Trail, Boston, MA');

INSERT INTO Payment (paymentID, courierID, locationID, amount, paymentDate)

VALUES

(1, 1, 1, 25.00, '2023-09-10'),

(2, 2, 2, 40.00, '2023-09-12'),

(3, 3, 3, 30.00, '2023-09-15'),

(4, 4, 4, 50.00, '2023-09-11'),

(5, 5, 5, 35.00, '2023-09-13'),

(6, 6, 6, 60.00, '2023-09-14'),

(7, 7, 7, 45.00, '2023-09-10'),

(8, 8, 8, 55.00, '2023-09-16'),

(9, 9, 9, 20.00, '2023-09-18'),

(10, 10, 10, 65.00, '2023-09-17');

--Task2:

--1)

select * from users;

--2)

select * from Courier where receiverName='Michael Green' or senderName='Michael Green';

--3)

select * from Courier;

--4)

select * from Courier where courierID=5;

--5)

select courierID,trackingNumber, deliveryDate from Courier where status='Delivered';

--6)

select * from Courier where status !='Delivered';

--7)

```
select * from Courier where deliveryDate='2023-09-11';  
--8)  
select * from Courier where status='In Transit';  
--9)  
SELECT courierID, COUNT(*) AS totalPackages FROM Courier GROUP BY courierID;  
--10)  
SELECT courierID, AVG(DATEDIFF(day, GETDATE(), DeliveryDate)) AS avgDeliveryTime FROM Courier GROUP BY courierID;  
--11)  
SELECT * FROM Courier WHERE weight BETWEEN 2.0 AND 5.0;  
--12)  
SELECT * FROM Employee WHERE name LIKE '%John%';  
--13)  
SELECT Courier.* FROM Courier JOIN Payment ON Courier.courierID =  
Payment.courierID WHERE Payment.amount > 50;
```

--Task3:

```
--14)  
SELECT employeeID, COUNT(courierID) AS totalCouriers FROM Courier GROUP BY  
employeeID;  
--15)  
SELECT locationID, SUM(amount) AS totalRevenue FROM Payment GROUP BY  
locationID;  
--16)  
SELECT l.locationName, COUNT(p.courierID) AS TotalCouriersDelivered FROM Location  
l JOIN Payment p ON l.locationID = p.locationID  
GROUP BY l.locationName;  
--17)  
SELECT TOP 1 courierID, AVG(DATEDIFF(day, deliveryDate, GETDATE())) AS  
deliveryTime FROM Courier WHERE deliveryDate IS NOT NULL  
GROUP BY courierID ORDER BY deliveryTime DESC;
```

--18)

```
SELECT l.locationName, SUM(p.amount) AS TotalPayments FROM Location l JOIN
Payment p ON l.locationID = p.locationID
GROUP BY l.locationName HAVING SUM(p.amount) < 1000;
```

--19)

```
SELECT l.locationName, SUM(p.amount) AS TotalPayments FROM Location l JOIN
Payment p ON l.locationID = p.locationID
GROUP BY l.locationName;
```

--20)

```
SELECT c.courierID, c.trackingNumber, SUM(p.amount) AS TotalPayments FROM Courier
c JOIN Payment p ON c.courierID = p.courierID
WHERE p.locationID = 5 GROUP BY c.courierID, c.trackingNumber HAVING
SUM(p.amount) > 10;
```

--21)

```
SELECT c.courierID, c.trackingNumber, SUM(p.amount) AS TotalPayments FROM Courier
c JOIN Payment p ON c.courierID = p.courierID
WHERE p.paymentDate > '2023-09-11' GROUP BY c.courierID, c.trackingNumber
HAVING SUM(p.amount) > 10;
```

--22)

```
SELECT l.locationName, SUM(p.amount) AS TotalReceived FROM Location l JOIN
Payment p ON l.locationID = p.locationID
WHERE p.paymentDate < '2024-09-19' GROUP BY l.locationName HAVING
SUM(p.amount) > 30;
```

-- ***** TASK 4*****

--23)

```
SELECT p.paymentID, p.amount, p.paymentDate, c.courierID, c.senderName,
c.receiverName, c.weight, c.status, c.trackingNumber
FROM Payment p JOIN Courier c ON p.courierID = c.courierID;
```

--24)

```
SELECT p.paymentID, p.amount, p.paymentDate, l.locationID, l.locationName, l.address
FROM Payment p JOIN Location l ON p.locationID = l.locationID;
```

--25)

```
SELECT p.paymentID, p.amount, p.paymentDate, c.courierID, c.senderName,  
c.receiverName, l.locationID, l.locationName, l.address
```

```
FROM Payment p JOIN Courier c ON p.courierID = c.courierID JOIN Location l ON  
p.locationID = l.locationID;
```

--26)

```
SELECT p.paymentID, p.amount, p.paymentDate, c.trackingNumber, c.senderName,  
c.receiverName, c.status
```

```
FROM Payment p JOIN Courier c ON p.courierID = c.courierID;
```

--27)

```
SELECT c.courierID, c.trackingNumber, SUM(p.amount) AS totalPayments
```

```
FROM Courier c JOIN Payment p ON c.courierID = p.courierID GROUP BY c.courierID,  
c.trackingNumber;
```

--28)

```
SELECT * FROM Payment WHERE paymentDate = '2023-09-10';
```

--29)

```
SELECT p.paymentID, p.courierID, c.senderName, c.receiverName, c.weight, c.status,  
c.trackingNumber, c.deliveryDate
```

```
FROM Payment p JOIN Courier c ON p.courierID = c.courierID;
```

--30)

```
SELECT p.paymentID, p.courierID, p.amount, p.paymentDate, l.locationName, l.address
```

```
FROM Payment p JOIN Location l ON p.locationID = l.locationID;
```

--31)

```
SELECT c.courierID, c.trackingNumber, SUM(p.amount) AS totalPayments
```

```
FROM Courier c JOIN Payment p ON c.courierID = p.courierID GROUP BY c.courierID,  
c.trackingNumber;
```

--32)

```
SELECT * FROM Payment WHERE paymentDate BETWEEN '2023-09-01' AND '2023-09-  
30';
```

--33)

```
SELECT u.userID, u.name, c.courierID, c.senderName, c.receiverName FROM Users u  
LEFT JOIN Courier c ON u.userID = c.courierID;
```

--34)

```
SELECT c.courierID, c.trackingNumber, s.serviceName, s.cost
FROM Courier c LEFT JOIN CourierServices s ON c.courierID = s.serviceID;
```

--35)

```
SELECT e.employeeID, e.name, p.paymentID, p.amount FROM Employee e LEFT JOIN
Payment p ON e.employeeID = p.courierID;
```

--36)

```
SELECT u.name AS userName, s.serviceName FROM Users u CROSS JOIN
CourierServices s;
```

--37)

```
SELECT e.name AS employeeName, l.locationName FROM Employee e CROSS JOIN
Location l;
```

--38)

```
SELECT c.courierID, c.senderName, c.senderAddress FROM Courier c;
```

--39)

```
SELECT c.courierID, c.receiverName, c.receiverAddress FROM Courier c;
```

--40)

```
SELECT c.courierID, c.trackingNumber, s.serviceName, s.cost
FROM Courier c LEFT JOIN CourierServices s ON c.courierID = s.serviceID;
```

--41)

```
SELECT e.employeeID, e.name, COUNT(c.courierID) AS totalCouriers
FROM Employee e LEFT JOIN Courier c ON e.employeeID = c.employeeID GROUP BY
e.employeeID, e.name;
```

--42)

```
SELECT l.locationName, SUM(p.amount) AS totalAmount
FROM Location l JOIN Payment p ON l.locationID = p.locationID GROUP BY
l.locationName;
```

--43)

```
SELECT
c1.courierID,c1.senderName,c1.senderAddress,c1.receiverName,c1.receiverAddress,c1.weig
ht,c1.status,
c1.trackingNumber,c1.deliveryDate,c1.employeeID
```

```
FROM Courier c1 INNER JOIN Courier c2 ON c1.senderName = c2.senderName
WHERE c1.courierID <> c2.courierID;
--44)

SELECT name, role FROM Employee GROUP BY role, name HAVING COUNT(*) > 1;
--45)

SELECT p.paymentID, p.amount, c.senderAddress
FROM Payment p JOIN Courier c ON p.courierID = c.courierID WHERE c.senderAddress
LIKE '123 Elm Street';

--46)

SELECT * FROM Courier WHERE senderAddress LIKE '123 Elm Street';

--47)

SELECT e.employeeID, e.name, COUNT(c.courierID) AS totalDelivered
FROM Employee e JOIN Courier c ON e.employeeID = c.employeeID WHERE c.status =
'Delivered' GROUP BY e.employeeID, e.name;

--48)

SELECT c.courierID, c.trackingNumber, p.amount, s.cost
FROM Courier c
JOIN Payment p ON c.courierID = p.courierID
JOIN CourierServices s ON c.courierID = s.serviceID WHERE p.amount > s.cost;

--****scope****

--49)

SELECT * FROM Courier WHERE weight > (SELECT AVG(weight) FROM Courier);

--50)

SELECT name, salary FROM Employee WHERE salary > (SELECT AVG(salary) FROM Employee);

--51)

SELECT SUM(cost) AS totalCost FROM CourierServices WHERE cost < (SELECT
MAX(cost) FROM CourierServices);

--52)
```

```
SELECT DISTINCT c.courierID, c.trackingNumber FROM Courier c JOIN Payment p ON c.courierID = p.courierID;
```

--53)

```
SELECT locationID, MAX(amount) AS maxPayment FROM Payment GROUP BY locationID;
```

--54)

```
SELECT * FROM Courier c1
```

```
WHERE c1.weight > (SELECT MAX(c2.weight) FROM Courier c2 WHERE c2.senderName = 'Laura White');
```