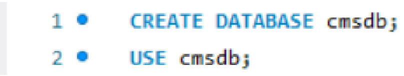
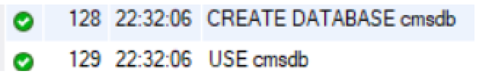
Assignment-Courier Management System

Task 1

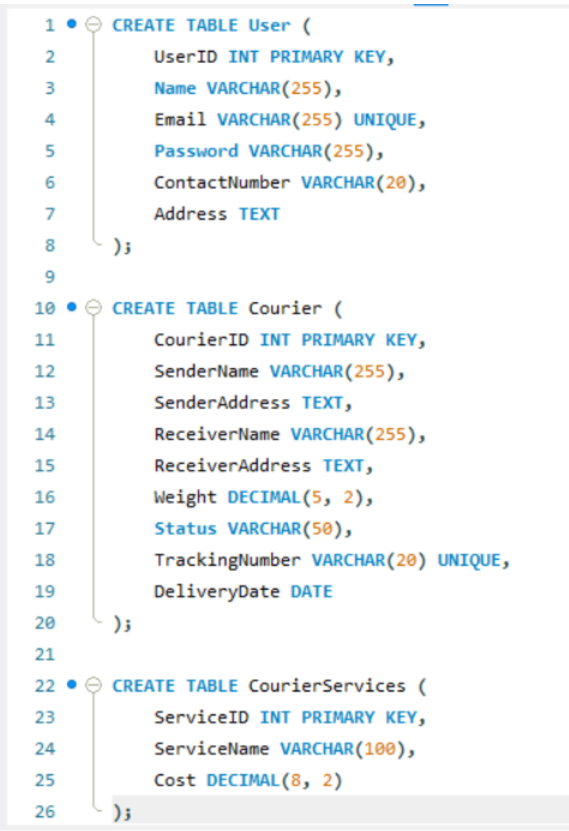
1. Database Design
2. Design a SQL schema for a Courier Management System with tables for Customers, Couriers, Orders, and Parcels. Define the relationships between these tables using appropriate foreign keys.
3. Requirements:
   • Define the Database Schema
   • Create SQL tables for entities such as 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.



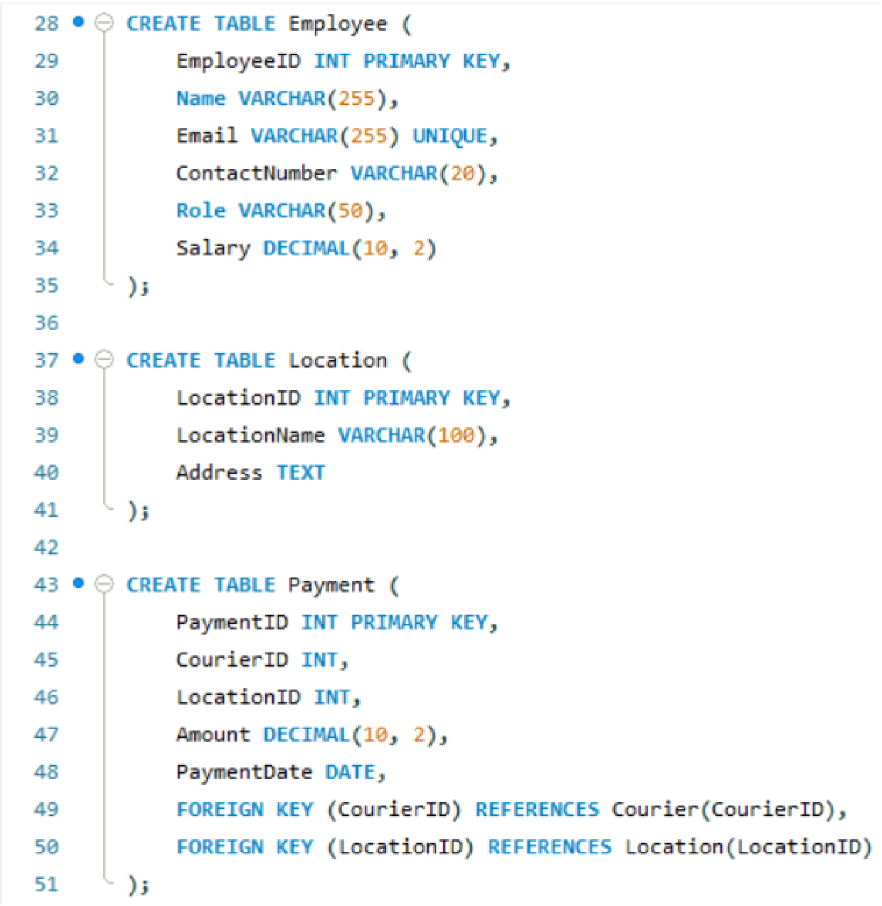
```
CREATE DATABASE cmsdb;
USE cmsdb;
```



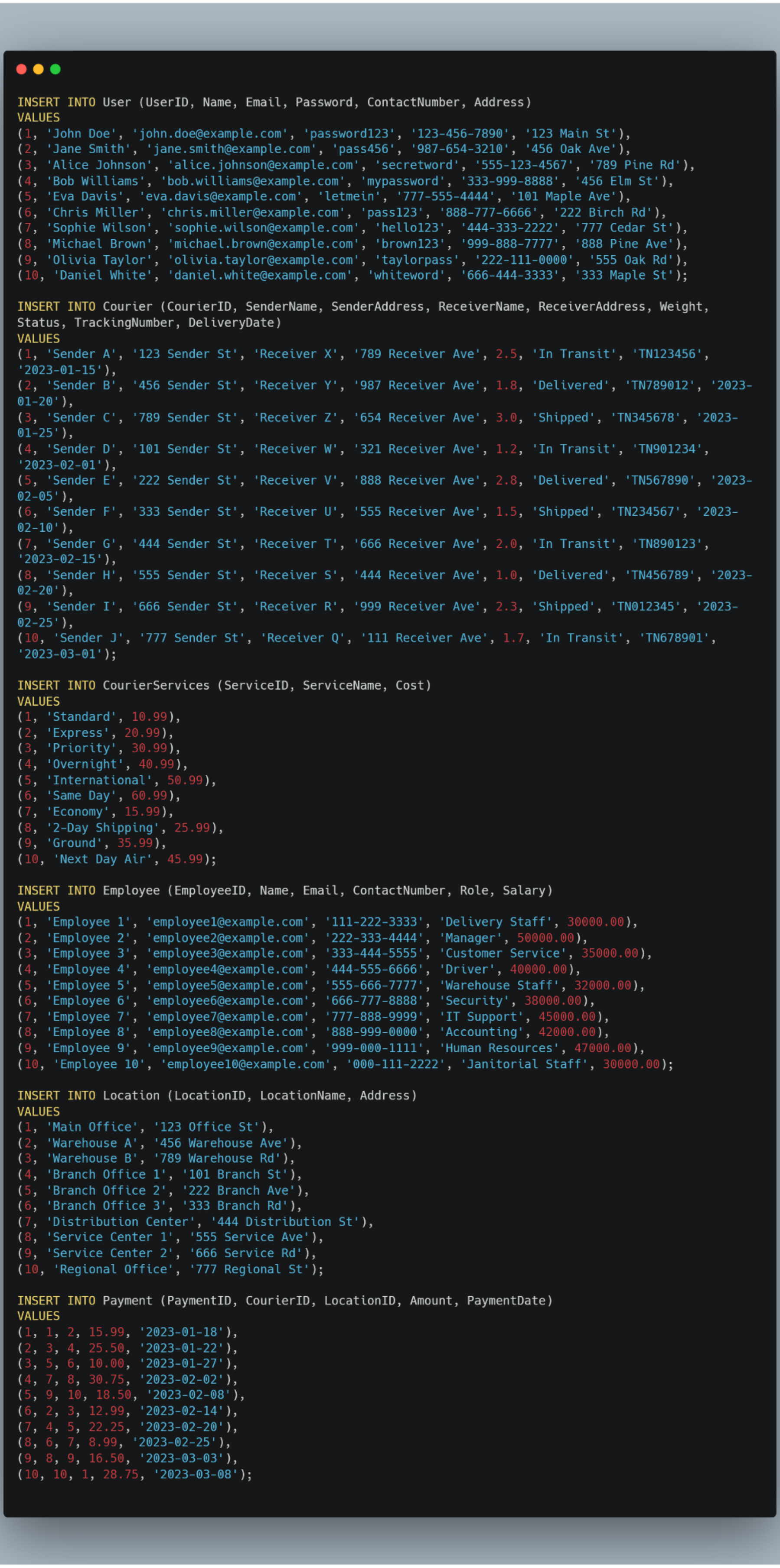
```
✓  128  22:32:06  CREATE DATABASE cmsdb
✓  129  22:32:06  USE cmsdb
```



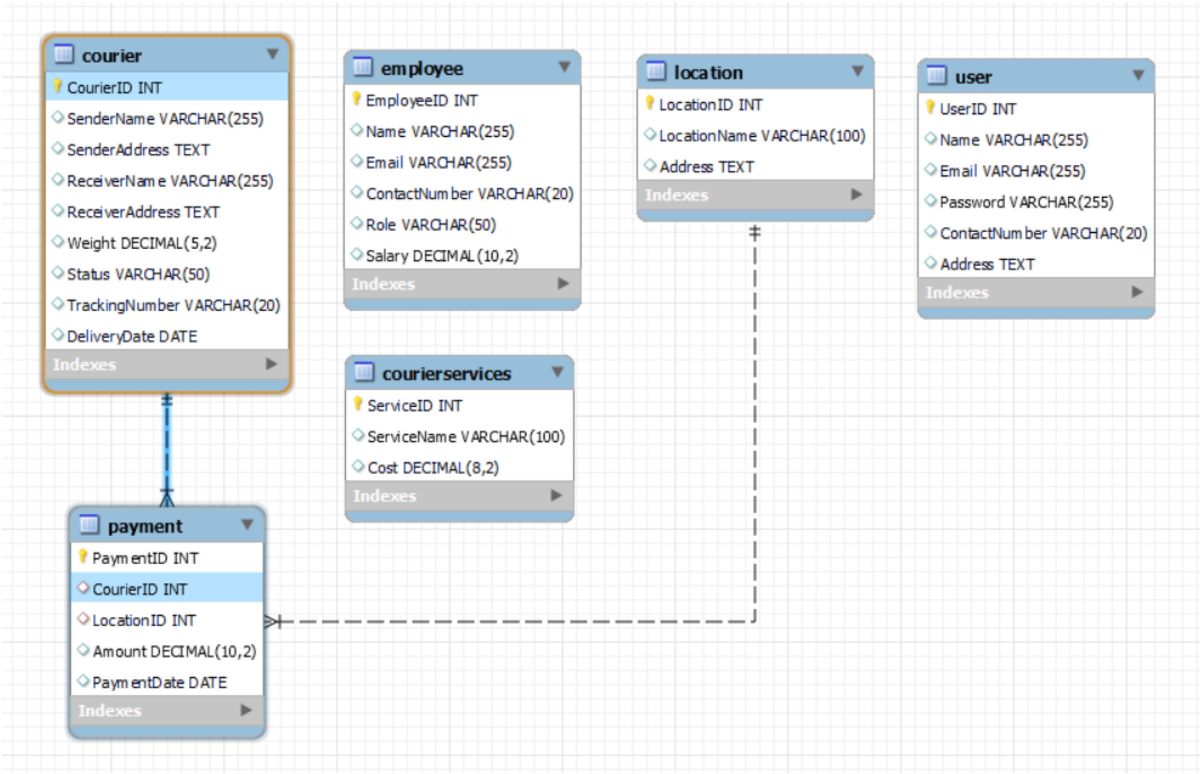
```
CREATE TABLE User (
    UserID INT PRIMARY KEY,
    Name VARCHAR(255),
    Email VARCHAR(255) UNIQUE,
    Password VARCHAR(255),
    ContactNumber VARCHAR(20),
    Address TEXT
);

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
);

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)
);
```

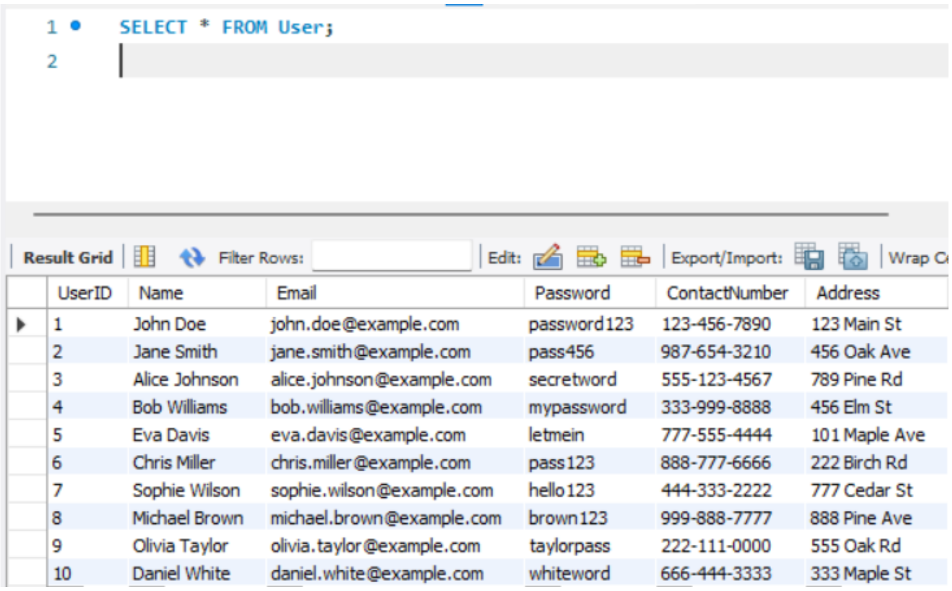


```sql
INSERT INTO User (UserID, Name, Email, Password, ContactNumber, Address)
VALUES
(1, 'John Doe', 'john.doe@example.com', 'password123', '123-456-7890', '123 Main St'),
(2, 'Jane Smith', 'jane.smith@example.com', 'pass456', '987-654-3210', '456 Oak Ave'),
(3, 'Alice Johnson', 'alice.johnson@example.com', 'secretword', '555-123-4567', '789 Pine Rd'),
(4, 'Bob Williams', 'bob.williams@example.com', 'mypassword', '333-999-8888', '456 Elm St'),
(5, 'Eva Davis', 'eva.davis@example.com', 'letmein', '777-555-4444', '101 Maple Ave'),
(6, 'Chris Miller', 'chris.miller@example.com', 'pass123', '888-777-6666', '222 Birch Rd'),
(7, 'Sophie Wilson', 'sophie.wilson@example.com', 'hello123', '444-333-2222', '777 Cedar St'),
(8, 'Michael Brown', 'michael.brown@example.com', 'brown123', '999-888-7777', '888 Pine Ave'),
(9, 'Olivia Taylor', 'olivia.taylor@example.com', 'taylorpass', '222-111-0000', '555 Oak Rd'),
(10, 'Daniel White', 'daniel.white@example.com', 'whiteword', '666-444-3333', '333 Maple St');

INSERT INTO Courier (CourierID, SenderName, SenderAddress, ReceiverName, ReceiverAddress, Weight,
Status, TrackingNumber, DeliveryDate)
VALUES
(1, 'Sender A', '123 Sender St', 'Receiver X', '789 Receiver Ave', 2.5, 'In Transit', 'TN123456',
'2023-01-15'),
(2, 'Sender B', '456 Sender St', 'Receiver Y', '987 Receiver Ave', 1.8, 'Delivered', 'TN789012', '2023-
01-20'),
(3, 'Sender C', '789 Sender St', 'Receiver Z', '654 Receiver Ave', 3.0, 'Shipped', 'TN345678', '2023-
01-25'),
(4, 'Sender D', '101 Sender St', 'Receiver W', '321 Receiver Ave', 1.2, 'In Transit', 'TN901234',
'2023-02-01'),
(5, 'Sender E', '222 Sender St', 'Receiver V', '888 Receiver Ave', 2.8, 'Delivered', 'TN567890', '2023-
02-05'),
(6, 'Sender F', '333 Sender St', 'Receiver U', '555 Receiver Ave', 1.5, 'Shipped', 'TN234567', '2023-
02-10'),
(7, 'Sender G', '444 Sender St', 'Receiver T', '666 Receiver Ave', 2.0, 'In Transit', 'TN890123',
'2023-02-15'),
(8, 'Sender H', '555 Sender St', 'Receiver S', '444 Receiver Ave', 1.0, 'Delivered', 'TN456789', '2023-
02-20'),
(9, 'Sender I', '666 Sender St', 'Receiver R', '999 Receiver Ave', 2.3, 'Shipped', 'TN012345', '2023-
02-25'),
(10, 'Sender J', '777 Sender St', 'Receiver Q', '111 Receiver Ave', 1.7, 'In Transit', 'TN678901',
'2023-03-01');

INSERT INTO CourierServices (ServiceID, ServiceName, Cost)
VALUES
(1, 'Standard', 10.99),
(2, 'Express', 20.99),
(3, 'Priority', 30.99),
(4, 'Overnight', 40.99),
(5, 'International', 50.99),
(6, 'Same Day', 60.99),
(7, 'Economy', 15.99),
(8, '2-Day Shipping', 25.99),
(9, 'Ground', 35.99),
(10, 'Next Day Air', 45.99);

INSERT INTO Employee (EmployeeID, Name, Email, ContactNumber, Role, Salary)
VALUES
(1, 'Employee 1', 'employee1@example.com', '111-222-3333', 'Delivery Staff', 30000.00),
(2, 'Employee 2', 'employee2@example.com', '222-333-4444', 'Manager', 50000.00),
(3, 'Employee 3', 'employee3@example.com', '333-444-5555', 'Customer Service', 35000.00),
(4, 'Employee 4', 'employee4@example.com', '444-555-6666', 'Driver', 40000.00),
(5, 'Employee 5', 'employee5@example.com', '555-666-7777', 'Warehouse Staff', 32000.00),
(6, 'Employee 6', 'employee6@example.com', '666-777-8888', 'Security', 38000.00),
(7, 'Employee 7', 'employee7@example.com', '777-888-9999', 'IT Support', 45000.00),
(8, 'Employee 8', 'employee8@example.com', '888-999-0000', 'Accounting', 42000.00),
(9, 'Employee 9', 'employee9@example.com', '999-000-1111', 'Human Resources', 47000.00),
(10, 'Employee 10', 'employee10@example.com', '000-111-2222', 'Janitorial Staff', 30000.00);

INSERT INTO Location (LocationID, LocationName, Address)
VALUES
(1, 'Main Office', '123 Office St'),
(2, 'Warehouse A', '456 Warehouse Ave'),
(3, 'Warehouse B', '789 Warehouse Rd'),
(4, 'Branch Office 1', '101 Branch St'),
(5, 'Branch Office 2', '222 Branch Ave'),
(6, 'Branch Office 3', '333 Branch Rd'),
(7, 'Distribution Center', '444 Distribution St'),
(8, 'Service Center 1', '555 Service Ave'),
(9, 'Service Center 2', '666 Service Rd'),
(10, 'Regional Office', '777 Regional St');

INSERT INTO Payment (PaymentID, CourierID, LocationID, Amount, PaymentDate)
VALUES
(1, 1, 2, 15.99, '2023-01-18'),
(2, 3, 4, 25.50, '2023-01-22'),
(3, 5, 6, 10.00, '2023-01-27'),
(4, 7, 8, 30.75, '2023-02-02'),
(5, 9, 10, 18.50, '2023-02-08'),
(6, 2, 3, 12.99, '2023-02-14'),
(7, 4, 5, 22.25, '2023-02-20'),
(8, 6, 7, 8.99, '2023-02-25'),
(9, 8, 9, 16.50, '2023-03-03'),
(10, 10, 1, 28.75, '2023-03-08');
```



Task 2’

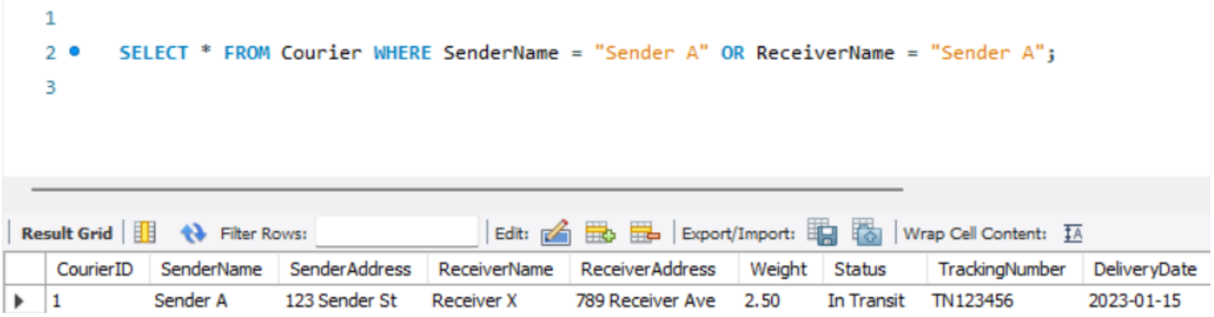
1. List all customers.



```
SELECT * FROM User;
```

UserID	Name	Email	Password	ContactNumber	Address
1	John Doe	john.doe@example.com	password123	123-456-7890	123 Main St
2	Jane Smith	jane.smith@example.com	pass456	987-654-3210	456 Oak Ave
3	Alice Johnson	alice.johnson@example.com	secretword	555-123-4567	789 Pine Rd
4	Bob Williams	bob.williams@example.com	mypassword	333-999-8888	456 Elm St
5	Eva Davis	eva.davis@example.com	letmein	777-555-4444	101 Maple Ave
6	Chris Miller	chris.miller@example.com	pass123	888-777-6666	222 Birch Rd
7	Sophie Wilson	sophie.wilson@example.com	hello123	444-333-2222	777 Cedar St
8	Michael Brown	michael.brown@example.com	brown123	999-888-7777	888 Pine Ave
9	Olivia Taylor	olivia.taylor@example.com	taylorpass	222-111-0000	555 Oak Rd
10	Daniel White	daniel.white@example.com	whiteword	666-444-3333	333 Maple St

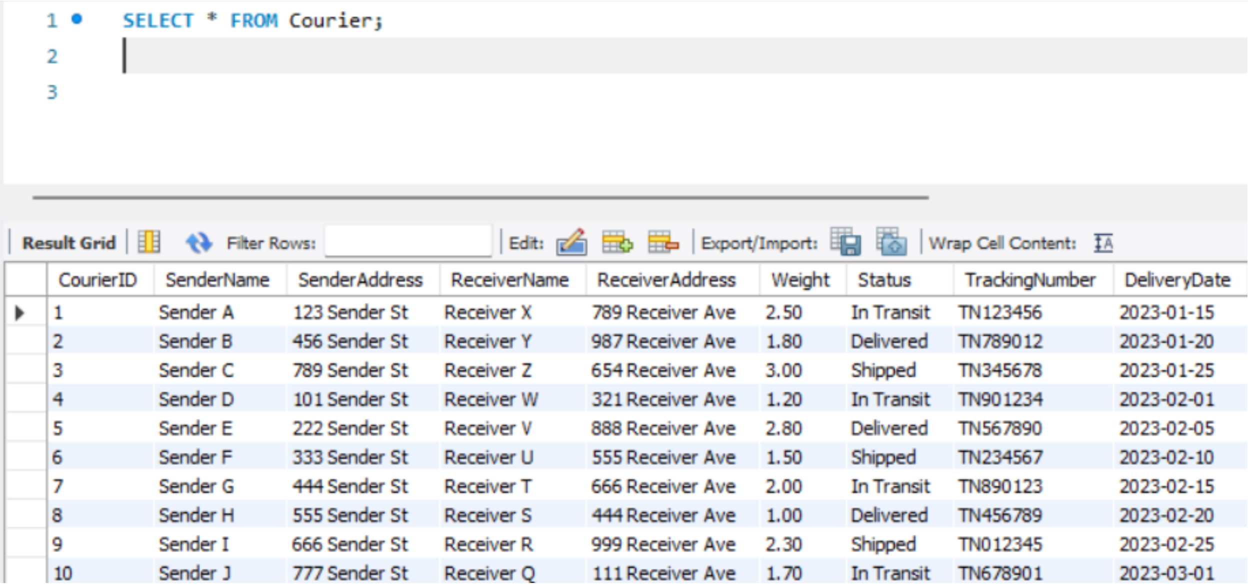
2. List all orders for a specific customer:



```
SELECT * FROM Courier WHERE SenderName = "Sender A" OR ReceiverName = "Sender A";
```

CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
1	Sender A	123 Sender St	Receiver X	789 Receiver Ave	2.50	In Transit	TN123456	2023-01-15

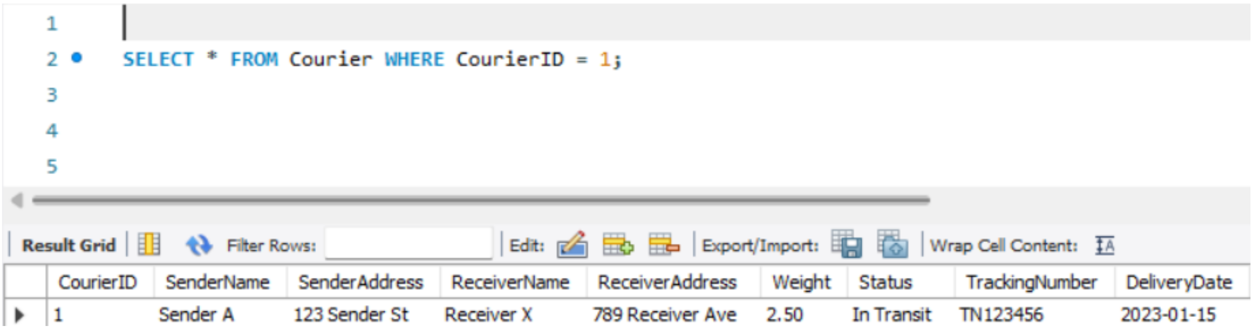
3. List all couriers.



```
SELECT * FROM Courier;
```

CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
1	Sender A	123 Sender St	Receiver X	789 Receiver Ave	2.50	In Transit	TN123456	2023-01-15
2	Sender B	456 Sender St	Receiver Y	987 Receiver Ave	1.80	Delivered	TN789012	2023-01-20
3	Sender C	789 Sender St	Receiver Z	654 Receiver Ave	3.00	Shipped	TN345678	2023-01-25
4	Sender D	101 Sender St	Receiver W	321 Receiver Ave	1.20	In Transit	TN901234	2023-02-01
5	Sender E	222 Sender St	Receiver V	888 Receiver Ave	2.80	Delivered	TN567890	2023-02-05
6	Sender F	333 Sender St	Receiver U	555 Receiver Ave	1.50	Shipped	TN234567	2023-02-10
7	Sender G	444 Sender St	Receiver T	666 Receiver Ave	2.00	In Transit	TN890123	2023-02-15
8	Sender H	555 Sender St	Receiver S	444 Receiver Ave	1.00	Delivered	TN456789	2023-02-20
9	Sender I	666 Sender St	Receiver R	999 Receiver Ave	2.30	Shipped	TN012345	2023-02-25
10	Sender J	777 Sender St	Receiver Q	111 Receiver Ave	1.70	In Transit	TN678901	2023-03-01

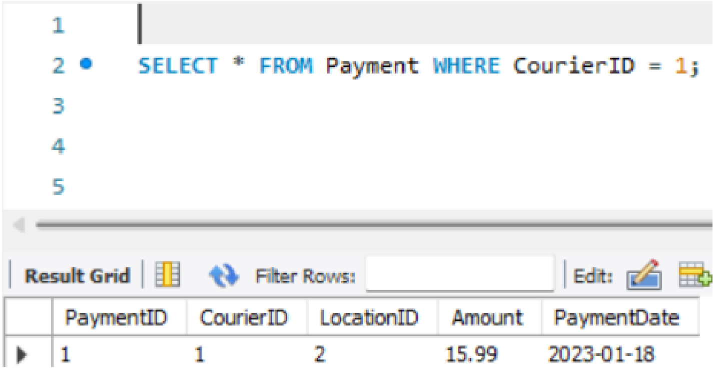
4. List all packages for a specific order:



```
SELECT * FROM Courier WHERE CourierID = 1;
```

CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
1	Sender A	123 Sender St	Receiver X	789 Receiver Ave	2.50	In Transit	TN123456	2023-01-15

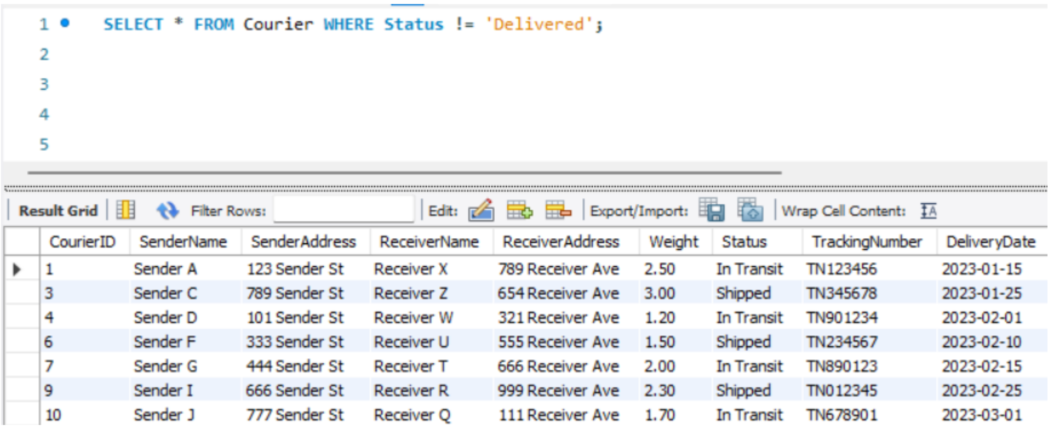
5. List all deliveries for a specific courier:



```sql
SELECT * FROM Payment WHERE CourierID = 1;
```

PaymentID	CourierID	LocationID	Amount	PaymentDate
1	1	2	15.99	2023-01-18

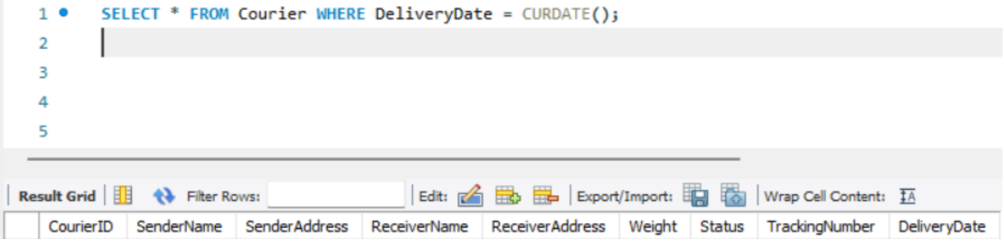
6. List all undelivered packages.



```sql
SELECT * FROM Courier WHERE Status != 'Delivered';
```

CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
1	Sender A	123 Sender St	Receiver X	789 Receiver Ave	2.50	In Transit	TN123456	2023-01-15
3	Sender C	789 Sender St	Receiver Z	654 Receiver Ave	3.00	Shipped	TN345678	2023-01-25
4	Sender D	101 Sender St	Receiver W	321 Receiver Ave	1.20	In Transit	TN901234	2023-02-01
6	Sender F	333 Sender St	Receiver U	555 Receiver Ave	1.50	Shipped	TN234567	2023-02-10
7	Sender G	444 Sender St	Receiver T	666 Receiver Ave	2.00	In Transit	TN890123	2023-02-15
9	Sender I	666 Sender St	Receiver R	999 Receiver Ave	2.30	Shipped	TN012345	2023-02-25
10	Sender J	777 Sender St	Receiver Q	111 Receiver Ave	1.70	In Transit	TN678901	2023-03-01

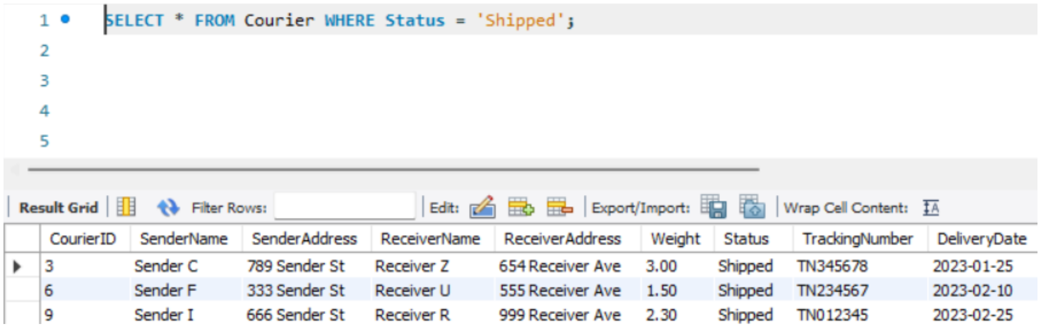
7. List all packages that are scheduled for delivery today.



```sql
SELECT * FROM Courier WHERE DeliveryDate = CURDATE();
```

CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate

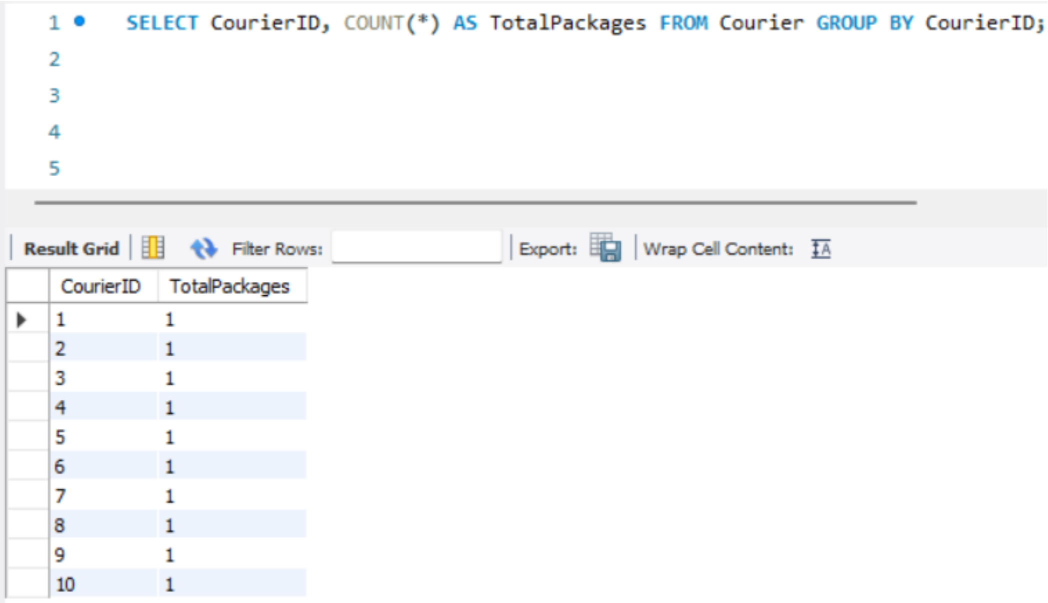
8. List all packages with a specific status.



```sql
SELECT * FROM Courier WHERE Status = 'Shipped';
```

CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
3	Sender C	789 Sender St	Receiver Z	654 Receiver Ave	3.00	Shipped	TN345678	2023-01-25
6	Sender F	333 Sender St	Receiver U	555 Receiver Ave	1.50	Shipped	TN234567	2023-02-10
9	Sender I	666 Sender St	Receiver R	999 Receiver Ave	2.30	Shipped	TN012345	2023-02-25

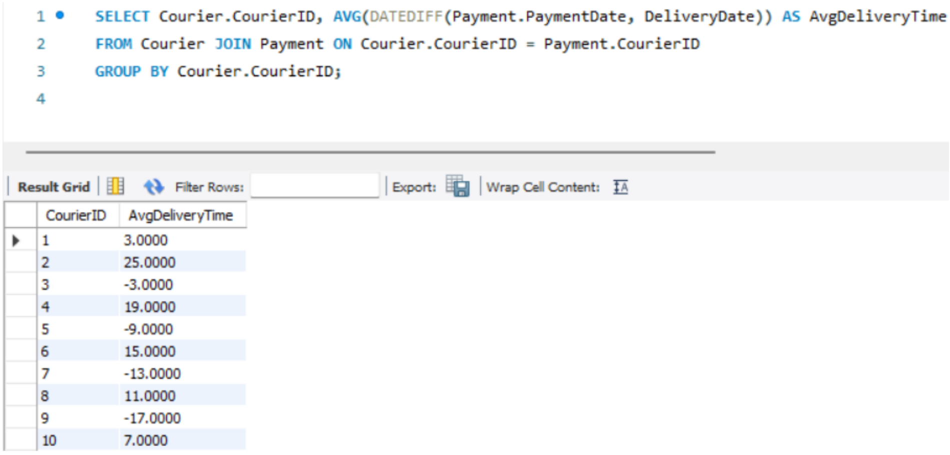
9. Calculate the total number of packages for each courier.



```sql
SELECT CourierID, COUNT(*) AS TotalPackages FROM Courier GROUP BY CourierID;
```

CourierID	TotalPackages
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1

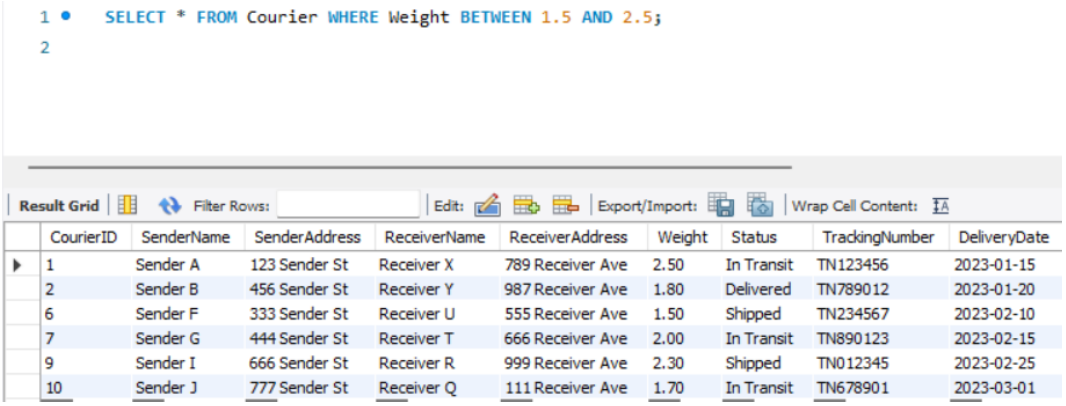
10. Find the average delivery time for each courier.



```sql
SELECT Courier.CourierID, AVG(DATEDIFF(Payment.PaymentDate, DeliveryDate)) AS AvgDeliveryTime
FROM Courier JOIN Payment ON Courier.CourierID = Payment.CourierID
GROUP BY Courier.CourierID;
```

CourierID	AvgDeliveryTime
1	3.0000
2	25.0000
3	-3.0000
4	19.0000
5	-9.0000
6	15.0000
7	-13.0000
8	11.0000
9	-17.0000
10	7.0000

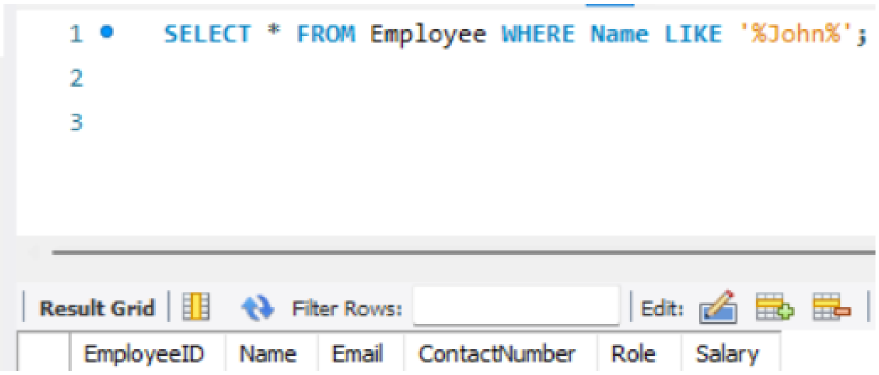
11. List all packages with a specific weight range:



```sql
SELECT * FROM Courier WHERE Weight BETWEEN 1.5 AND 2.5;
```

CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
1	Sender A	123 Sender St	Receiver X	789 Receiver Ave	2.50	In Transit	TN123456	2023-01-15
2	Sender B	456 Sender St	Receiver Y	987 Receiver Ave	1.80	Delivered	TN789012	2023-01-20
6	Sender F	333 Sender St	Receiver U	555 Receiver Ave	1.50	Shipped	TN234567	2023-02-10
7	Sender G	444 Sender St	Receiver T	666 Receiver Ave	2.00	In Transit	TN890123	2023-02-15
9	Sender I	666 Sender St	Receiver R	999 Receiver Ave	2.30	Shipped	TN012345	2023-02-25
10	Sender J	777 Sender St	Receiver Q	111 Receiver Ave	1.70	In Transit	TN678901	2023-03-01

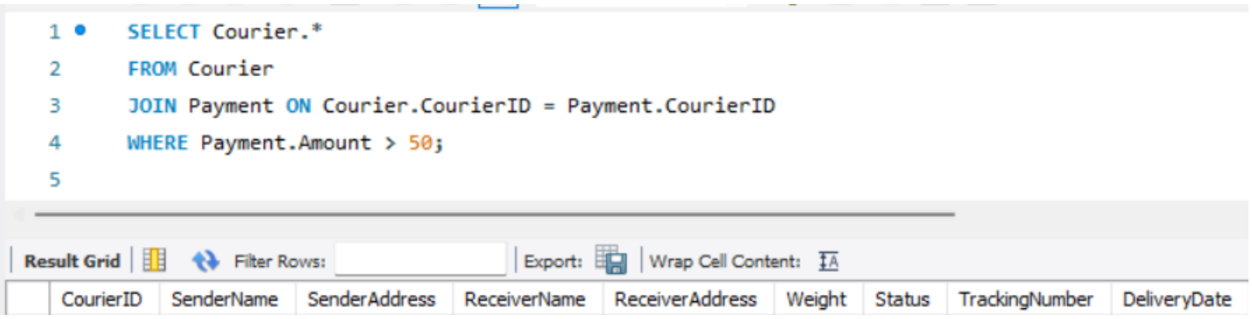
12. Retrieve employees whose names contain 'John'.



```sql
SELECT * FROM Employee WHERE Name LIKE '%John%';


```

	EmployeeID	Name	Email	ContactNumber	Role	Salary

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

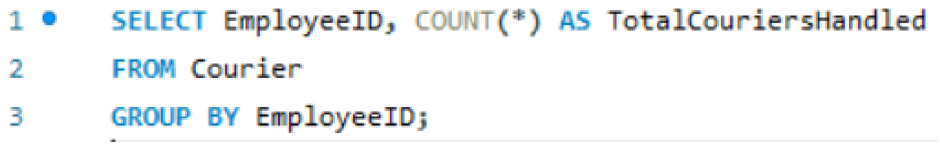


```sql
SELECT Courier.*
FROM Courier
JOIN Payment ON Courier.CourierID = Payment.CourierID
WHERE Payment.Amount > 50;

```

	CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate

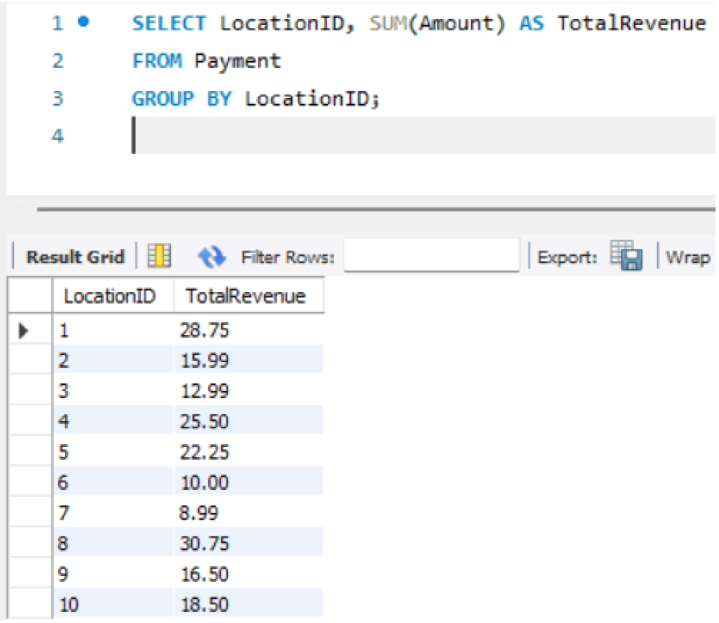
Task 3: GroupBy, Aggregate Functions, Having, Order By, where

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



```sql
SELECT EmployeeID, COUNT(*) AS TotalCouriersHandled
FROM Courier
GROUP BY EmployeeID;
```

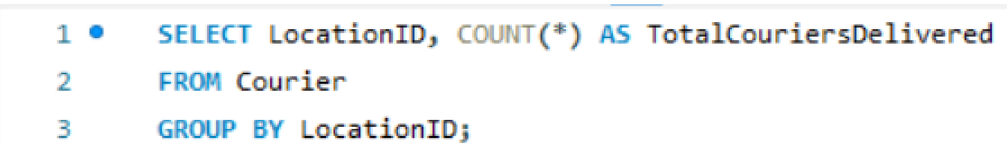
2. Calculate the total revenue generated by each location.



```sql
SELECT LocationID, SUM(Amount) AS TotalRevenue
FROM Payment
GROUP BY LocationID;
```

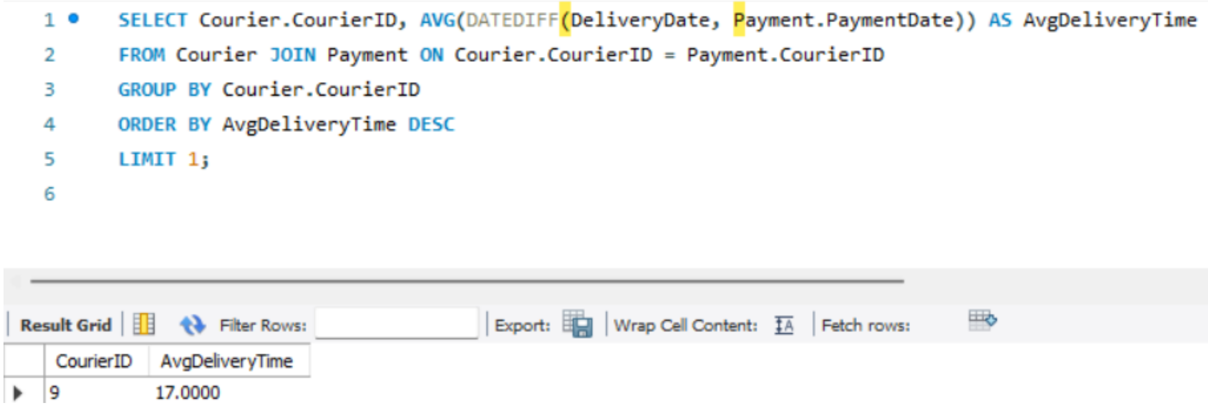
LocationID	TotalRevenue
1	28.75
2	15.99
3	12.99
4	25.50
5	22.25
6	10.00
7	8.99
8	30.75
9	16.50
10	18.50

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



```sql
SELECT LocationID, COUNT(*) AS TotalCouriersDelivered
FROM Courier
GROUP BY LocationID;
```

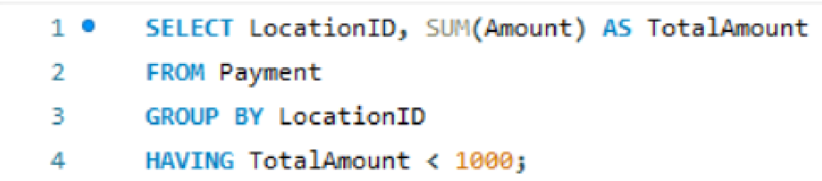
4. Find the courier with the highest average delivery time.



```sql
SELECT Courier.CourierID, AVG(DATEDIFF(DeliveryDate, Payment.PaymentDate)) AS AvgDeliveryTime
FROM Courier JOIN Payment ON Courier.CourierID = Payment.CourierID
GROUP BY Courier.CourierID
ORDER BY AvgDeliveryTime DESC
LIMIT 1;
```

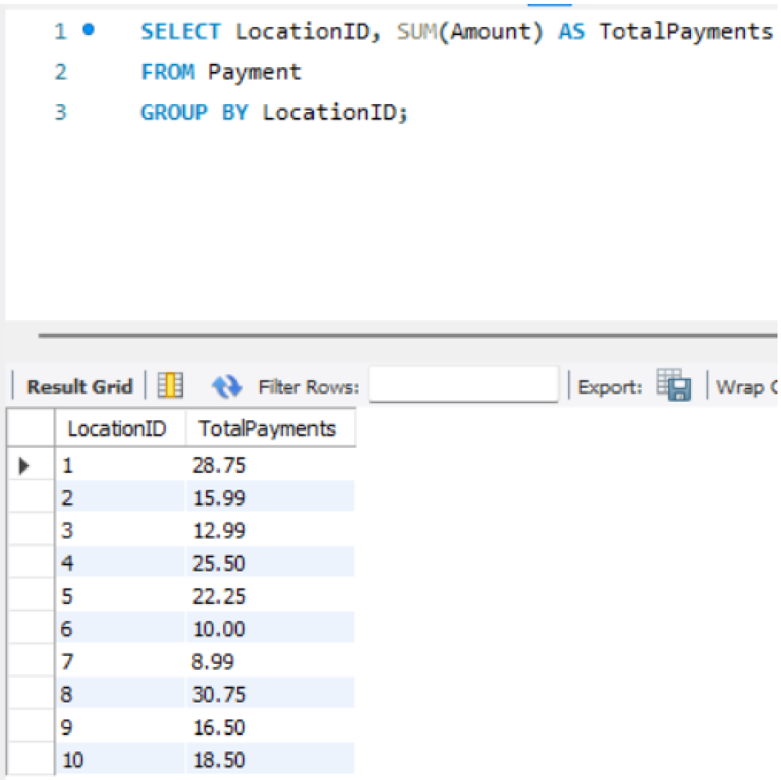
CourierID	AvgDeliveryTime
9	17.0000

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



```sql
SELECT LocationID, SUM(Amount) AS TotalAmount
FROM Payment
GROUP BY LocationID
HAVING TotalAmount < 1000;
```

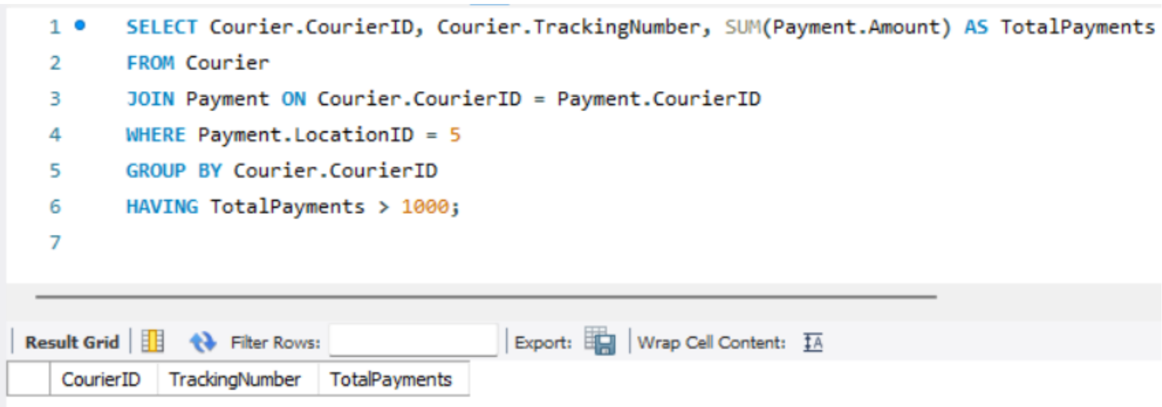
6. Calculate Total Payments per Location.



```sql
SELECT LocationID, SUM(Amount) AS TotalPayments
FROM Payment
GROUP BY LocationID;
```

LocationID	TotalPayments
1	28.75
2	15.99
3	12.99
4	25.50
5	22.25
6	10.00
7	8.99
8	30.75
9	16.50
10	18.50

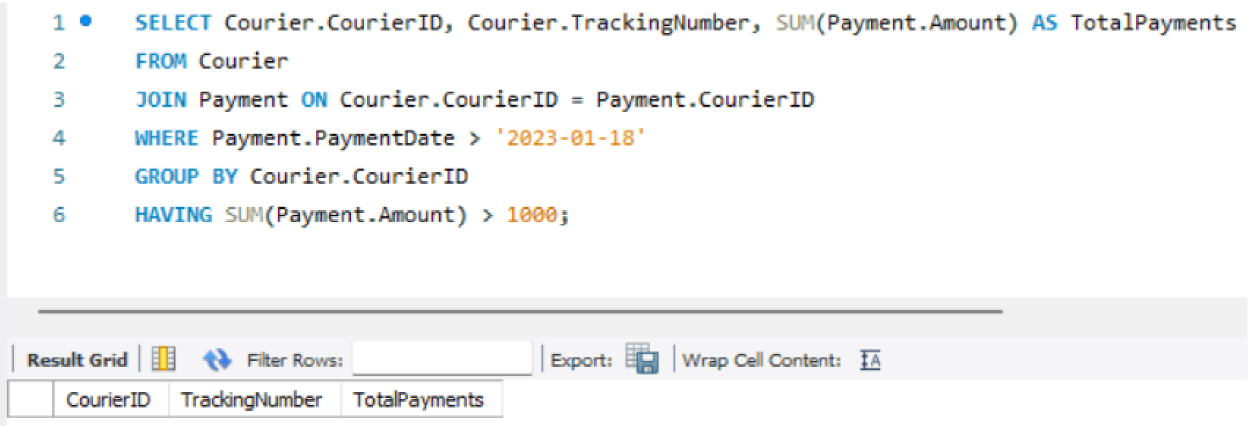
7. Retrieve couriers who have received payments totaling more than $1000 in a specific location (LocationID = X).



```sql
SELECT Courier.CourierID, Courier.TrackingNumber, SUM(Payment.Amount) AS TotalPayments
FROM Courier
JOIN Payment ON Courier.CourierID = Payment.CourierID
WHERE Payment.LocationID = 5
GROUP BY Courier.CourierID
HAVING TotalPayments > 1000;
```

CourierID	TrackingNumber	TotalPayments

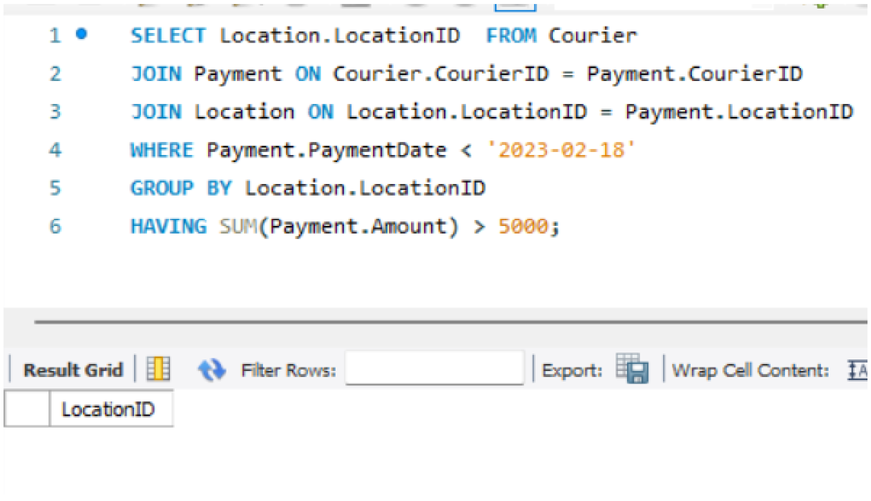
8. Retrieve couriers who have received payments totaling more than $1000 after a certain date (PaymentDate > 'YYYY-MM-DD'):



```sql
SELECT Courier.CourierID, Courier.TrackingNumber, SUM(Payment.Amount) AS TotalPayments
FROM Courier
JOIN Payment ON Courier.CourierID = Payment.CourierID
WHERE Payment.PaymentDate > '2023-01-18'
GROUP BY Courier.CourierID
HAVING SUM(Payment.Amount) > 1000;
```

CourierID	TrackingNumber	TotalPayments

9. Retrieve locations where the total amount received is more than $5000 before a certain date (PaymentDate > 'YYYY-MM-DD').

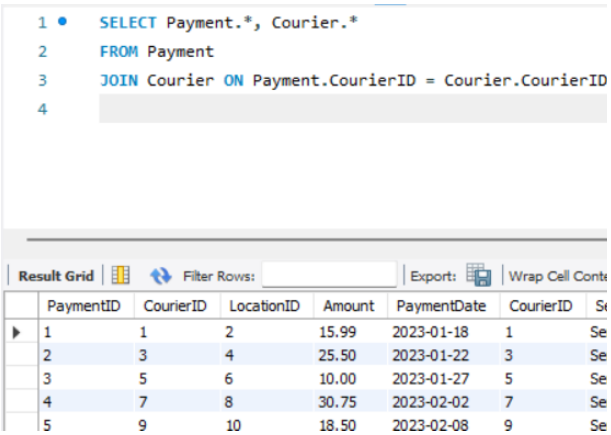


```sql
SELECT Location.LocationID  FROM Courier
JOIN Payment ON Courier.CourierID = Payment.CourierID
JOIN Location ON Location.LocationID = Payment.LocationID
WHERE Payment.PaymentDate < '2023-02-18'
GROUP BY Location.LocationID
HAVING SUM(Payment.Amount) > 5000;
```

LocationID

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

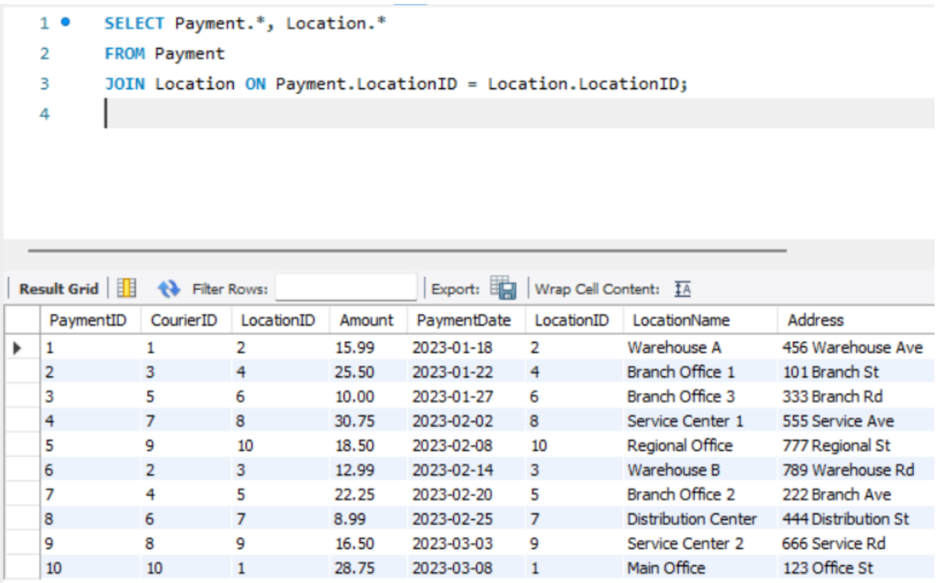
1. Retrieve Payments with Courier Information.



```sql
SELECT Payment.*, Courier.*
FROM Payment
JOIN Courier ON Payment.CourierID = Courier.CourierID
```

PaymentID	CourierID	LocationID	Amount	PaymentDate	CourierID	Se
1	1	2	15.99	2023-01-18	1	Se
2	3	4	25.50	2023-01-22	3	Se
3	5	6	10.00	2023-01-27	5	Se
4	7	8	30.75	2023-02-02	7	Se
5	9	10	18.50	2023-02-08	9	Se

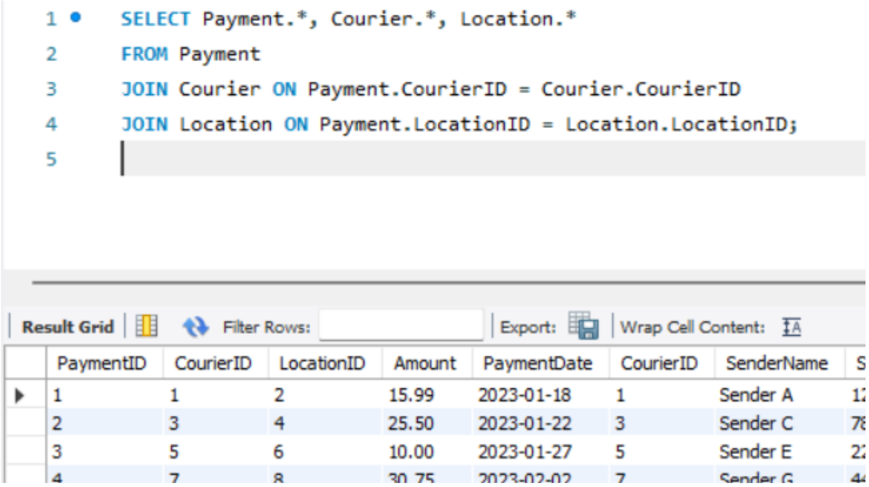
2. Retrieve Payments with Location Information.



```sql
SELECT Payment.*, Location.*
FROM Payment
JOIN Location ON Payment.LocationID = Location.LocationID;
```

PaymentID	CourierID	LocationID	Amount	PaymentDate	LocationID	LocationName	Address
1	1	2	15.99	2023-01-18	2	Warehouse A	456 Warehouse Ave
2	3	4	25.50	2023-01-22	4	Branch Office 1	101 Branch St
3	5	6	10.00	2023-01-27	6	Branch Office 3	333 Branch Rd
4	7	8	30.75	2023-02-02	8	Service Center 1	555 Service Ave
5	9	10	18.50	2023-02-08	10	Regional Office	777 Regional St
6	2	3	12.99	2023-02-14	3	Warehouse B	789 Warehouse Rd
7	4	5	22.25	2023-02-20	5	Branch Office 2	222 Branch Ave
8	6	7	8.99	2023-02-25	7	Distribution Center	444 Distribution St
9	8	9	16.50	2023-03-03	9	Service Center 2	666 Service Rd
10	10	1	28.75	2023-03-08	1	Main Office	123 Office St

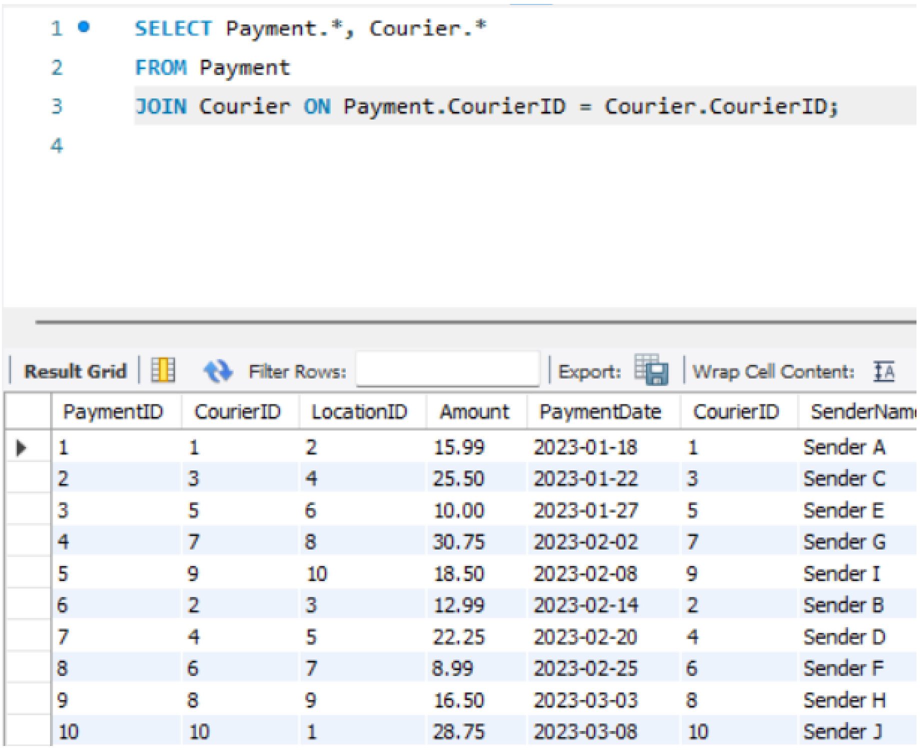
3. Retrieve Payments with Courier and Location Information



```sql
SELECT Payment.*, Courier.*, Location.*
FROM Payment
JOIN Courier ON Payment.CourierID = Courier.CourierID
JOIN Location ON Payment.LocationID = Location.LocationID;
```

PaymentID	CourierID	LocationID	Amount	PaymentDate	CourierID	SenderName	S
1	1	2	15.99	2023-01-18	1	Sender A	12
2	3	4	25.50	2023-01-22	3	Sender C	78
3	5	6	10.00	2023-01-27	5	Sender E	22
4	7	8	30.75	2023-02-02	7	Sender G	44

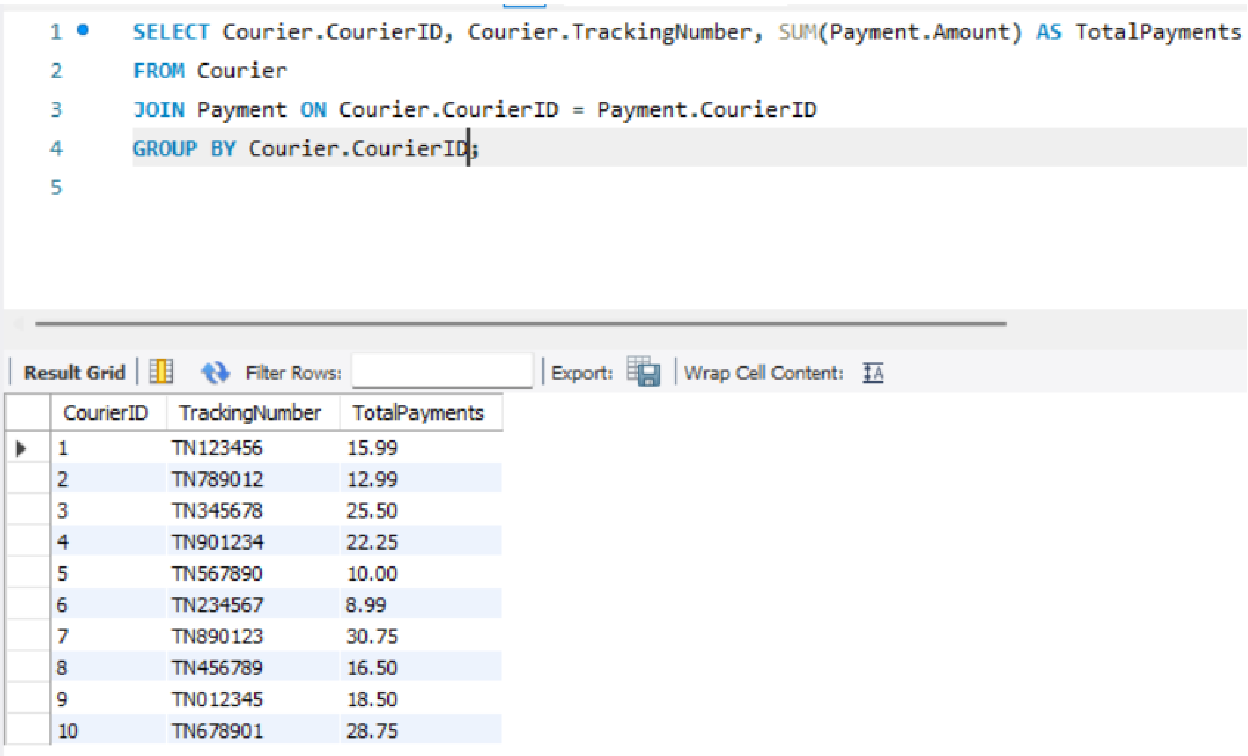
4. List all payments with courier details



```sql
SELECT Payment.*, Courier.*
FROM Payment
JOIN Courier ON Payment.CourierID = Courier.CourierID;
```

PaymentID	CourierID	LocationID	Amount	PaymentDate	CourierID	SenderNam
1	1	2	15.99	2023-01-18	1	Sender A
2	3	4	25.50	2023-01-22	3	Sender C
3	5	6	10.00	2023-01-27	5	Sender E
4	7	8	30.75	2023-02-02	7	Sender G
5	9	10	18.50	2023-02-08	9	Sender I
6	2	3	12.99	2023-02-14	2	Sender B
7	4	5	22.25	2023-02-20	4	Sender D
8	6	7	8.99	2023-02-25	6	Sender F
9	8	9	16.50	2023-03-03	8	Sender H
10	10	1	28.75	2023-03-08	10	Sender J

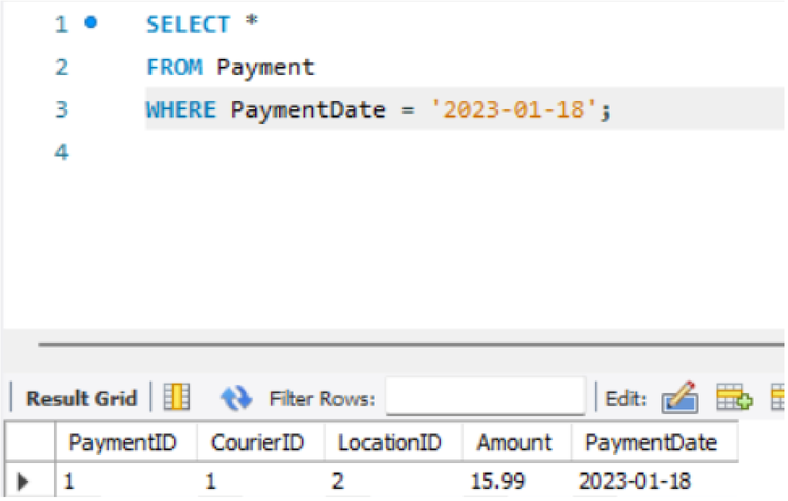
5. Total payments received for each courier



```sql
SELECT Courier.CourierID, Courier.TrackingNumber, SUM(Payment.Amount) AS TotalPayments
FROM Courier
JOIN Payment ON Courier.CourierID = Payment.CourierID
GROUP BY Courier.CourierID;
```

CourierID	TrackingNumber	TotalPayments
1	TN123456	15.99
2	TN789012	12.99
3	TN345678	25.50
4	TN901234	22.25
5	TN567890	10.00
6	TN234567	8.99
7	TN890123	30.75
8	TN456789	16.50
9	TN012345	18.50
10	TN678901	28.75

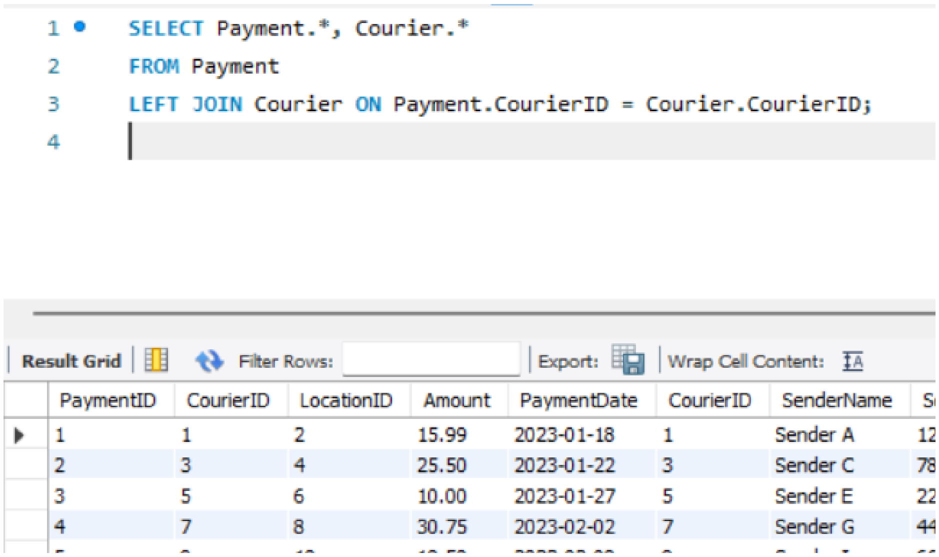
6. List payments made on a specific date



```sql
SELECT *
FROM Payment
WHERE PaymentDate = '2023-01-18';
```

PaymentID	CourierID	LocationID	Amount	PaymentDate
1	1	2	15.99	2023-01-18

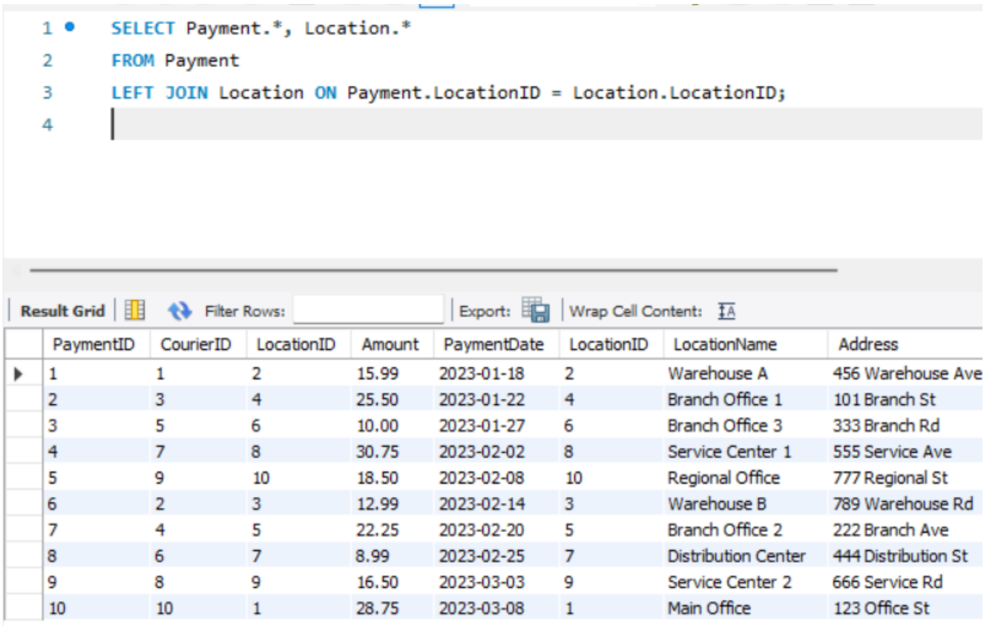
7. Get Courier Information for Each Payment



```sql
SELECT Payment.*, Courier.*
FROM Payment
LEFT JOIN Courier ON Payment.CourierID = Courier.CourierID;
```

PaymentID	CourierID	LocationID	Amount	PaymentDate	CourierID	SenderName	S
1	1	2	15.99	2023-01-18	1	Sender A	12
2	3	4	25.50	2023-01-22	3	Sender C	78
3	5	6	10.00	2023-01-27	5	Sender E	22
4	7	8	30.75	2023-02-02	7	Sender G	44

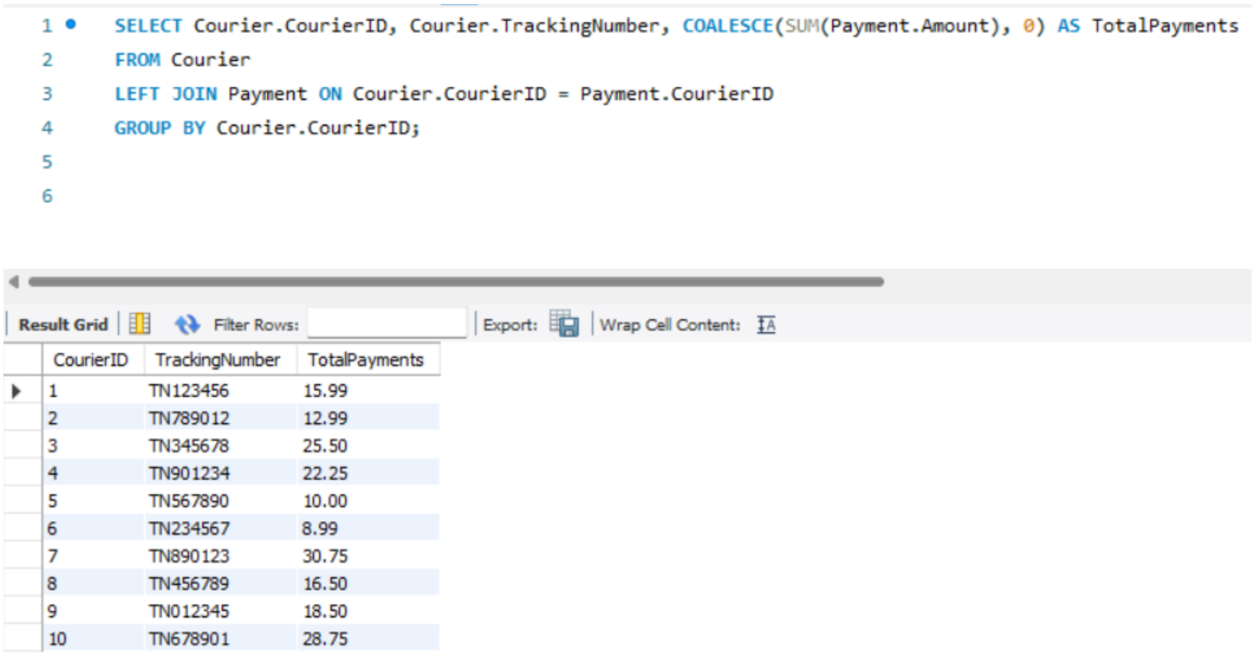
8. Get Payment Details with Location



```sql
SELECT Payment.*, Location.*
FROM Payment
LEFT JOIN Location ON Payment.LocationID = Location.LocationID;
```

PaymentID	CourierID	LocationID	Amount	PaymentDate	LocationID	LocationName	Address
1	1	2	15.99	2023-01-18	2	Warehouse A	456 Warehouse Ave
2	3	4	25.50	2023-01-22	4	Branch Office 1	101 Branch St
3	5	6	10.00	2023-01-27	6	Branch Office 3	333 Branch Rd
4	7	8	30.75	2023-02-02	8	Service Center 1	555 Service Ave
5	9	10	18.50	2023-02-08	10	Regional Office	777 Regional St
6	2	3	12.99	2023-02-14	3	Warehouse B	789 Warehouse Rd
7	4	5	22.25	2023-02-20	5	Branch Office 2	222 Branch Ave
8	6	7	8.99	2023-02-25	7	Distribution Center	444 Distribution St
9	8	9	16.50	2023-03-03	9	Service Center 2	666 Service Rd
10	10	1	28.75	2023-03-08	1	Main Office	123 Office St

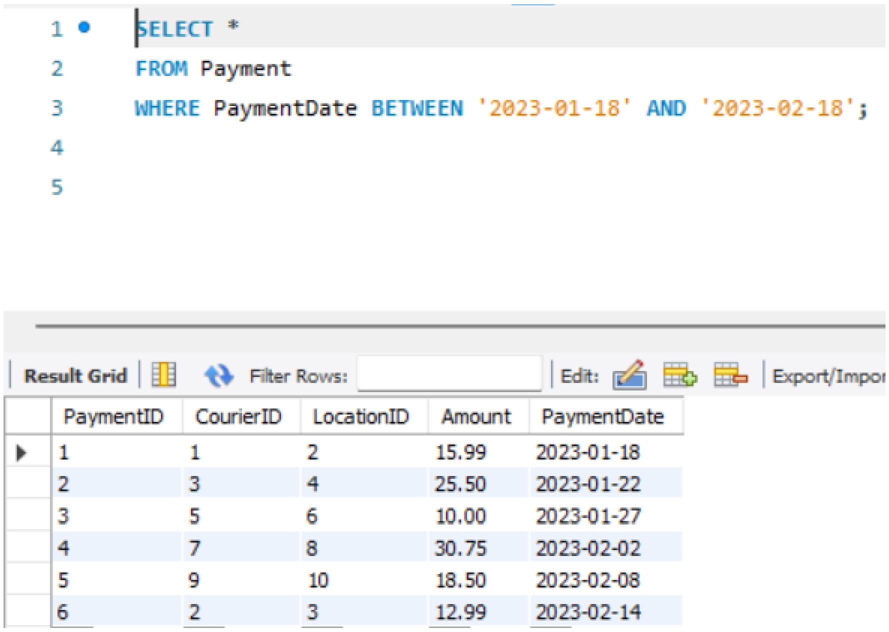
9. Calculating Total Payments for Each Courier



```sql
SELECT Courier.CourierID, Courier.TrackingNumber, COALESCE(SUM(Payment.Amount), 0) AS TotalPayments
FROM Courier
LEFT JOIN Payment ON Courier.CourierID = Payment.CourierID
GROUP BY Courier.CourierID;
```

CourierID	TrackingNumber	TotalPayments
1	TN123456	15.99
2	TN789012	12.99
3	TN345678	25.50
4	TN901234	22.25
5	TN567890	10.00
6	TN234567	8.99
7	TN890123	30.75
8	TN456789	16.50
9	TN012345	18.50
10	TN678901	28.75

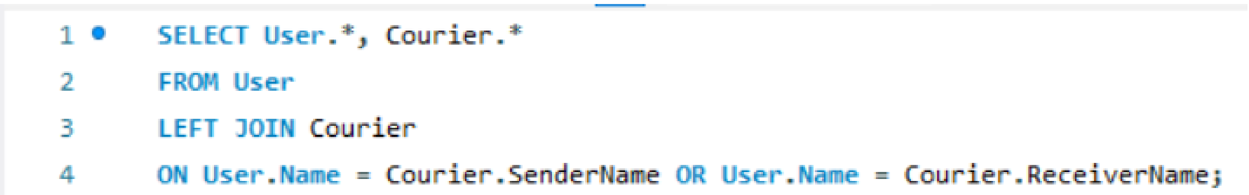
10. List Payments Within a Date Range



```sql
SELECT *
FROM Payment
WHERE PaymentDate BETWEEN '2023-01-18' AND '2023-02-18';
```

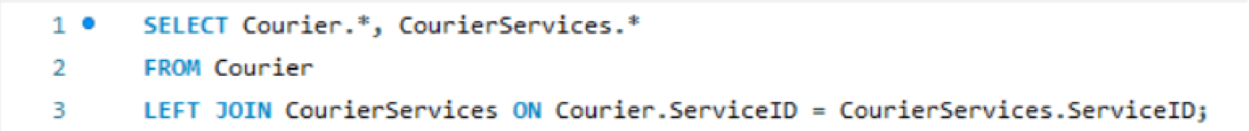
PaymentID	CourierID	LocationID	Amount	PaymentDate
1	1	2	15.99	2023-01-18
2	3	4	25.50	2023-01-22
3	5	6	10.00	2023-01-27
4	7	8	30.75	2023-02-02
5	9	10	18.50	2023-02-08
6	2	3	12.99	2023-02-14

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



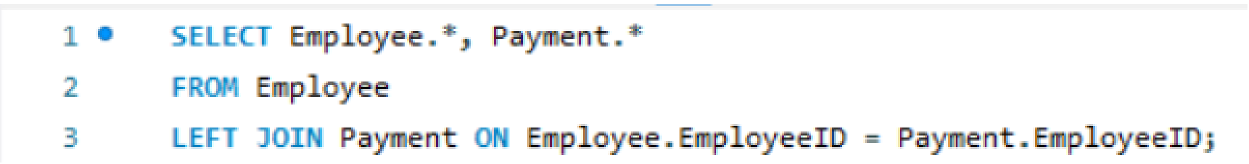
```sql
SELECT User.*, Courier.*
FROM User
LEFT JOIN Courier
ON User.Name = Courier.SenderName OR User.Name = Courier.ReceiverName;
```

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



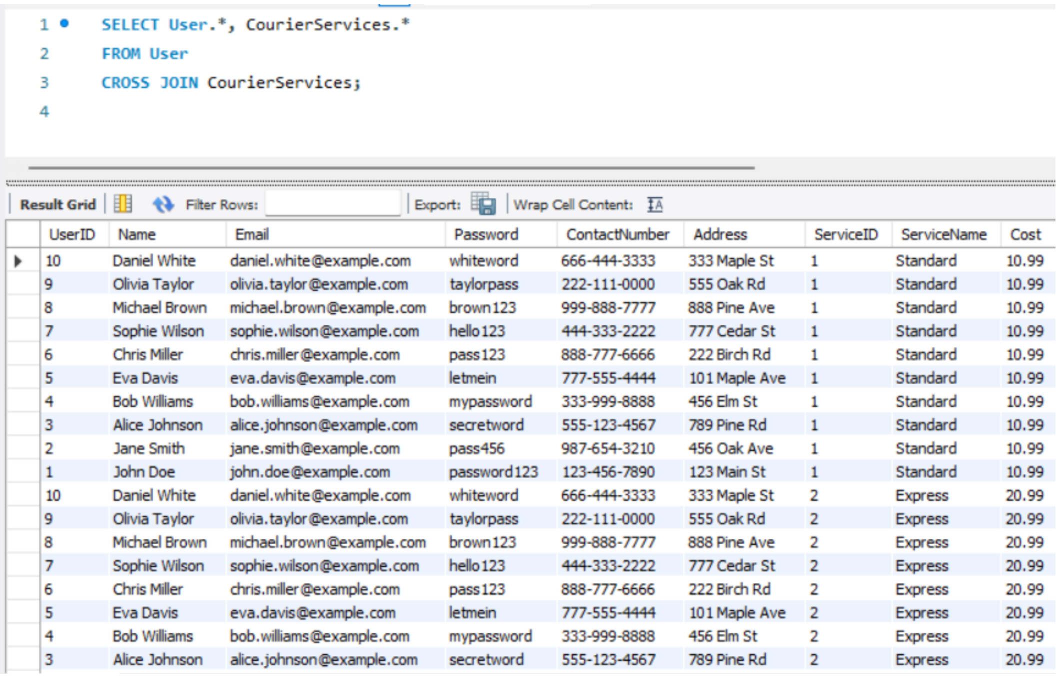
```sql
SELECT Courier.*, CourierServices.*
FROM Courier
LEFT JOIN CourierServices ON Courier.ServiceID = CourierServices.ServiceID;
```

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



```sql
SELECT Employee.*, Payment.*
FROM Employee
LEFT JOIN Payment ON Employee.EmployeeID = Payment.EmployeeID;
```

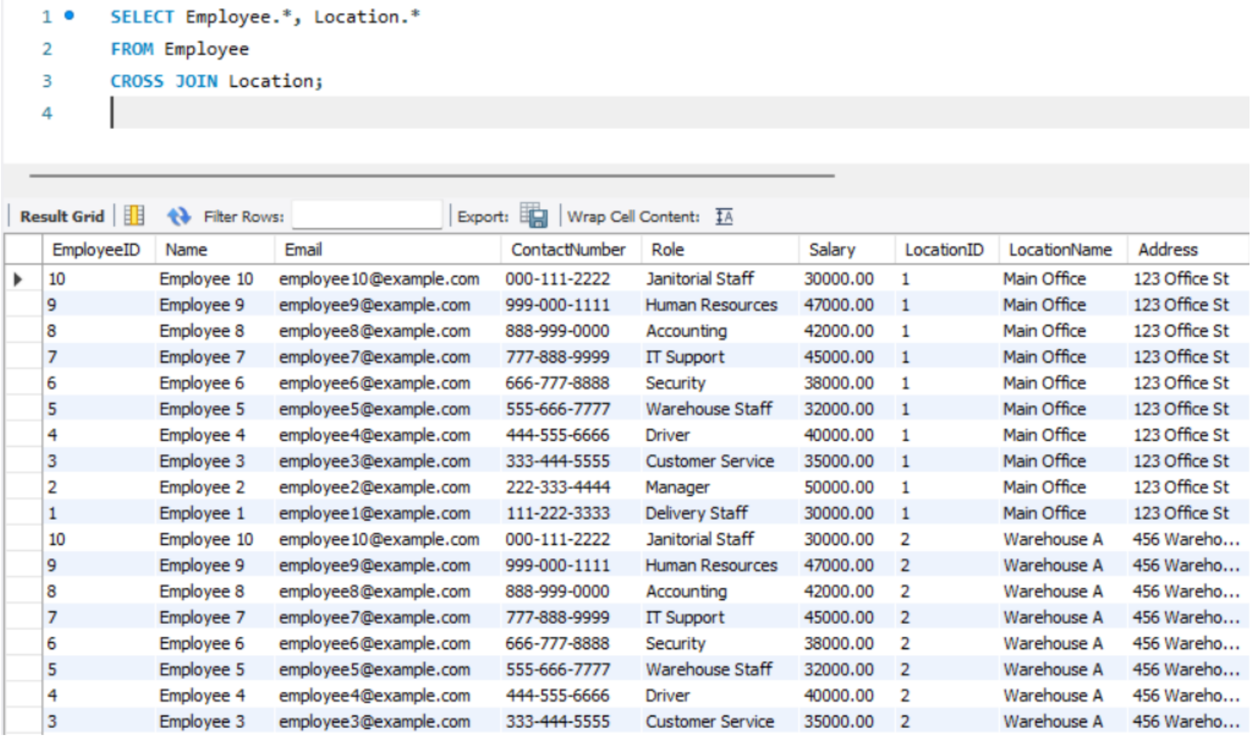
14. List all users and all courier services, showing all possible combinations.



```sql
SELECT User.*, CourierServices.*
FROM User
CROSS JOIN CourierServices;
```

UserID	Name	Email	Password	ContactNumber	Address	ServiceID	ServiceName	Cost
10	Daniel White	daniel.white@example.com	whiteword	666-444-3333	333 Maple St	1	Standard	10.99
9	Olivia Taylor	olivia.taylor@example.com	taylorpass	222-111-0000	555 Oak Rd	1	Standard	10.99
8	Michael Brown	michael.brown@example.com	brown123	999-888-7777	888 Pine Ave	1	Standard	10.99
7	Sophie Wilson	sophie.wilson@example.com	hello123	444-333-2222	777 Cedar St	1	Standard	10.99
6	Chris Miller	chris.miller@example.com	pass123	888-777-6666	222 Birch Rd	1	Standard	10.99
5	Eva Davis	eva.davis@example.com	letmein	777-555-4444	101 Maple Ave	1	Standard	10.99
4	Bob Williams	bob.williams@example.com	mypassword	333-999-8888	456 Elm St	1	Standard	10.99
3	Alice Johnson	alice.johnson@example.com	secretword	555-123-4567	789 Pine Rd	1	Standard	10.99
2	Jane Smith	jane.smith@example.com	pass456	987-654-3210	456 Oak Ave	1	Standard	10.99
1	John Doe	john.doe@example.com	password123	123-456-7890	123 Main St	1	Standard	10.99
10	Daniel White	daniel.white@example.com	whiteword	666-444-3333	333 Maple St	2	Express	20.99
9	Olivia Taylor	olivia.taylor@example.com	taylorpass	222-111-0000	555 Oak Rd	2	Express	20.99
8	Michael Brown	michael.brown@example.com	brown123	999-888-7777	888 Pine Ave	2	Express	20.99
7	Sophie Wilson	sophie.wilson@example.com	hello123	444-333-2222	777 Cedar St	2	Express	20.99
6	Chris Miller	chris.miller@example.com	pass123	888-777-6666	222 Birch Rd	2	Express	20.99
5	Eva Davis	eva.davis@example.com	letmein	777-555-4444	101 Maple Ave	2	Express	20.99
4	Bob Williams	bob.williams@example.com	mypassword	333-999-8888	456 Elm St	2	Express	20.99
3	Alice Johnson	alice.johnson@example.com	secretword	555-123-4567	789 Pine Rd	2	Express	20.99

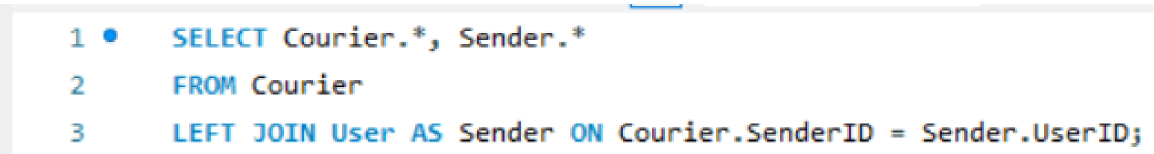
15. List all employees and all locations, showing all possible combinations:



```sql
SELECT Employee.*, Location.*
FROM Employee
CROSS JOIN Location;
```

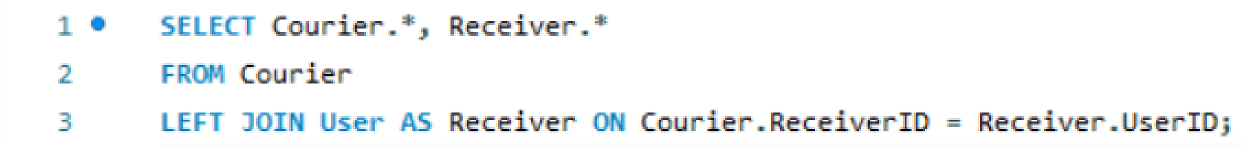
EmployeeID	Name	Email	ContactNumber	Role	Salary	LocationID	LocationName	Address
10	Employee 10	employee10@example.com	000-111-2222	Janitorial Staff	30000.00	1	Main Office	123 Office St
9	Employee 9	employee9@example.com	999-000-1111	Human Resources	47000.00	1	Main Office	123 Office St
8	Employee 8	employee8@example.com	888-999-0000	Accounting	42000.00	1	Main Office	123 Office St
7	Employee 7	employee7@example.com	777-888-9999	IT Support	45000.00	1	Main Office	123 Office St
6	Employee 6	employee6@example.com	666-777-8888	Security	38000.00	1	Main Office	123 Office St
5	Employee 5	employee5@example.com	555-666-7777	Warehouse Staff	32000.00	1	Main Office	123 Office St
4	Employee 4	employee4@example.com	444-555-6666	Driver	40000.00	1	Main Office	123 Office St
3	Employee 3	employee3@example.com	333-444-5555	Customer Service	35000.00	1	Main Office	123 Office St
2	Employee 2	employee2@example.com	222-333-4444	Manager	50000.00	1	Main Office	123 Office St
1	Employee 1	employee1@example.com	111-222-3333	Delivery Staff	30000.00	1	Main Office	123 Office St
10	Employee 10	employee10@example.com	000-111-2222	Janitorial Staff	30000.00	2	Warehouse A	456 Wareho...
9	Employee 9	employee9@example.com	999-000-1111	Human Resources	47000.00	2	Warehouse A	456 Wareho...
8	Employee 8	employee8@example.com	888-999-0000	Accounting	42000.00	2	Warehouse A	456 Wareho...
7	Employee 7	employee7@example.com	777-888-9999	IT Support	45000.00	2	Warehouse A	456 Wareho...
6	Employee 6	employee6@example.com	666-777-8888	Security	38000.00	2	Warehouse A	456 Wareho...
5	Employee 5	employee5@example.com	555-666-7777	Warehouse Staff	32000.00	2	Warehouse A	456 Wareho...
4	Employee 4	employee4@example.com	444-555-6666	Driver	40000.00	2	Warehouse A	456 Wareho...
3	Employee 3	employee3@example.com	333-444-5555	Customer Service	35000.00	2	Warehouse A	456 Wareho...

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



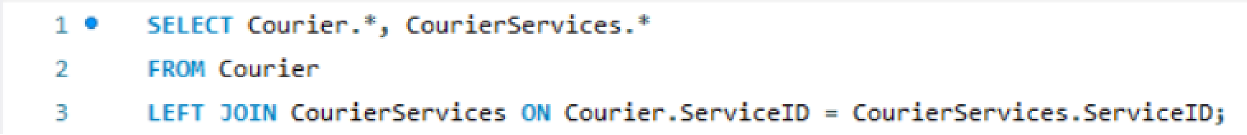
```sql
SELECT Courier.*, Sender.*
FROM Courier
LEFT JOIN User AS Sender ON Courier.SenderID = Sender.UserID;
```

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



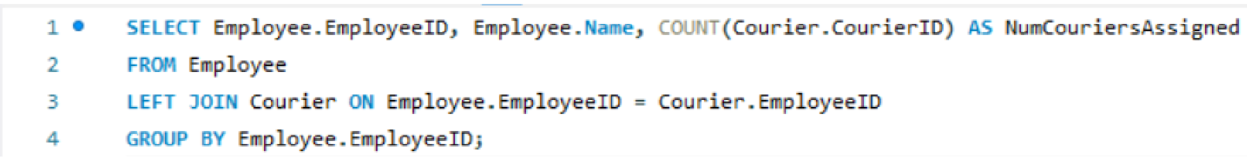
```sql
SELECT Courier.*, Receiver.*
FROM Courier
LEFT JOIN User AS Receiver ON Courier.ReceiverID = Receiver.UserID;
```

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



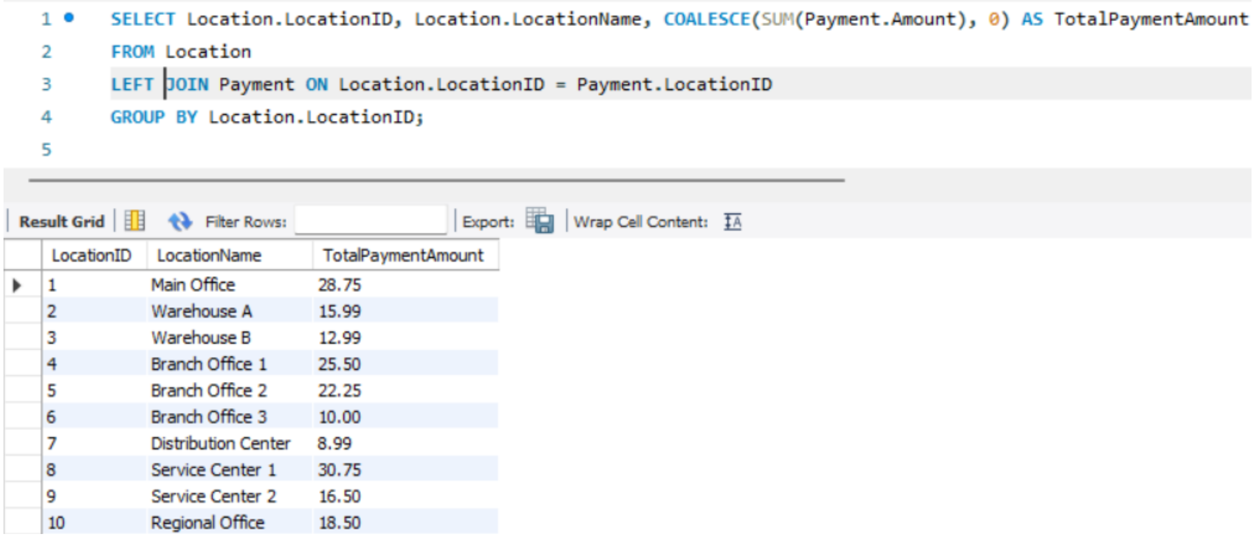
```sql
SELECT Courier.*, CourierServices.*
FROM Courier
LEFT JOIN CourierServices ON Courier.ServiceID = CourierServices.ServiceID;
```

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



```sql
SELECT Employee.EmployeeID, Employee.Name, COUNT(Courier.CourierID) AS NumCouriersAssigned
FROM Employee
LEFT JOIN Courier ON Employee.EmployeeID = Courier.EmployeeID
GROUP BY Employee.EmployeeID;
```

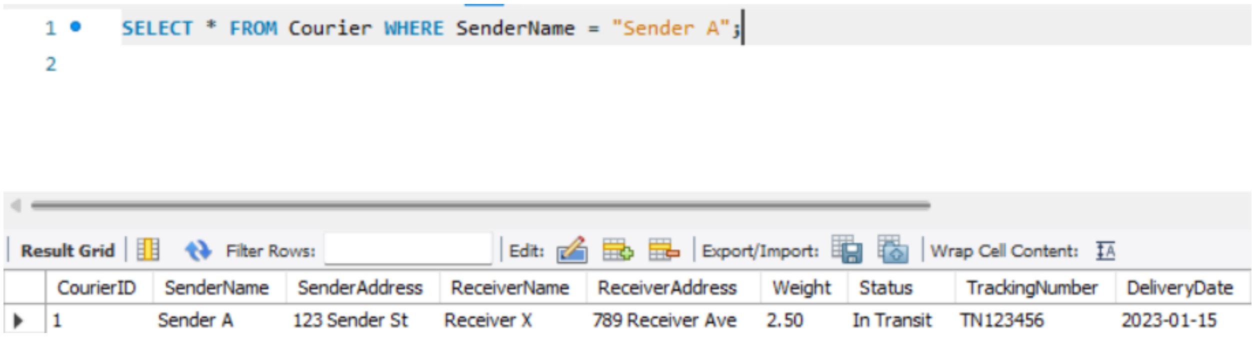
20. Retrieve a list of locations and the total payment amount received at each location.



```sql
SELECT Location.LocationID, Location.LocationName, COALESCE(SUM(Payment.Amount), 0) AS TotalPaymentAmount
FROM Location
LEFT JOIN Payment ON Location.LocationID = Payment.LocationID
GROUP BY Location.LocationID;

```

LocationID	LocationName	TotalPaymentAmount
1	Main Office	28.75
2	Warehouse A	15.99
3	Warehouse B	12.99
4	Branch Office 1	25.50
5	Branch Office 2	22.25
6	Branch Office 3	10.00
7	Distribution Center	8.99
8	Service Center 1	30.75
9	Service Center 2	16.50
10	Regional Office	18.50

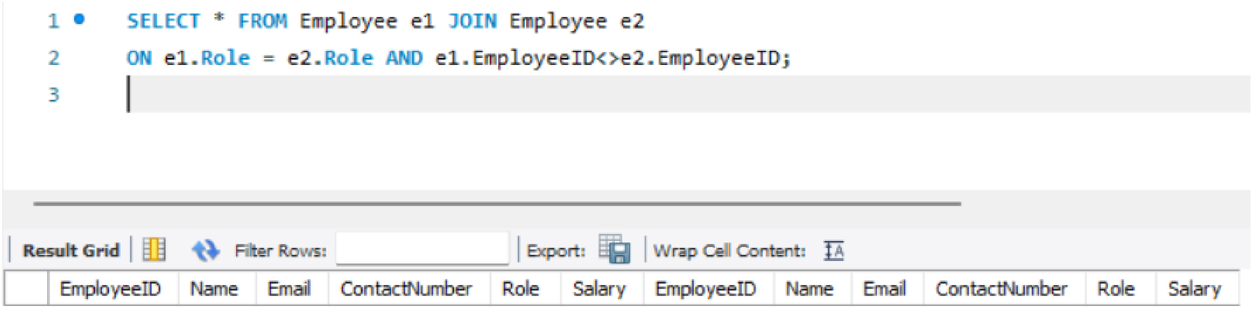
21. Retrieve all couriers sent by the same sender (based on SenderName):



```
SELECT * FROM Courier WHERE SenderName = "Sender A";
```

	CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
▶	1	Sender A	123 Sender St	Receiver X	789 Receiver Ave	2.50	In Transit	TN123456	2023-01-15

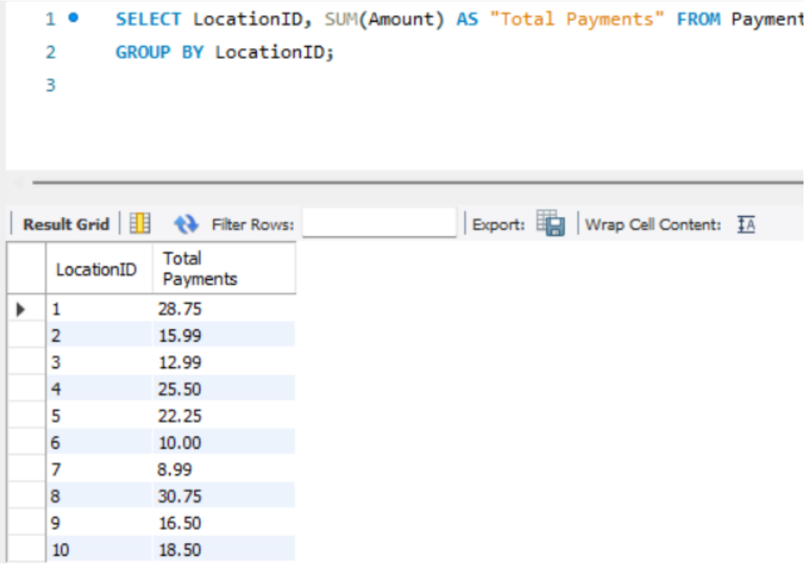
22. List all employees who share the same role.



```
SELECT * FROM Employee e1 JOIN Employee e2
ON e1.Role = e2.Role AND e1.EmployeeID<>e2.EmployeeID;
```

	EmployeeID	Name	Email	ContactNumber	Role	Salary	EmployeeID	Name	Email	ContactNumber	Role	Salary

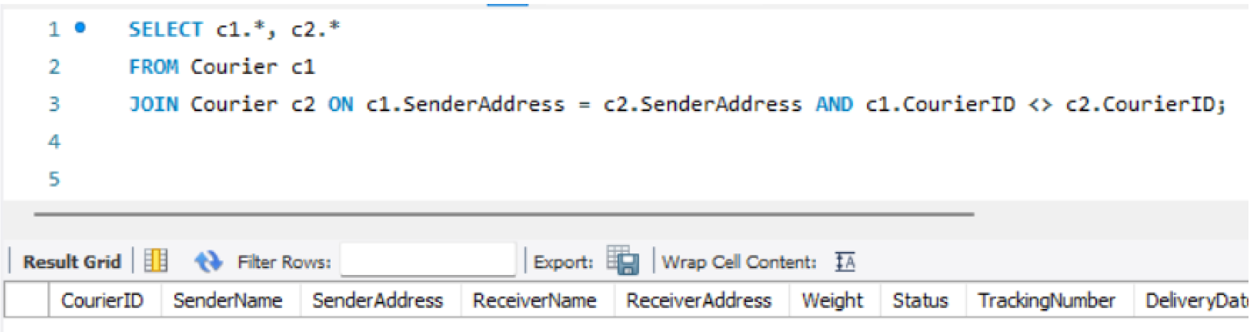
23. Retrieve all payments made for couriers sent from the same location.



```
SELECT LocationID, SUM(Amount) AS "Total Payments" FROM Payment
GROUP BY LocationID;
```

	LocationID	Total Payments
▶	1	28.75
	2	15.99
	3	12.99
	4	25.50
	5	22.25
	6	10.00
	7	8.99
	8	30.75
	9	16.50
	10	18.50

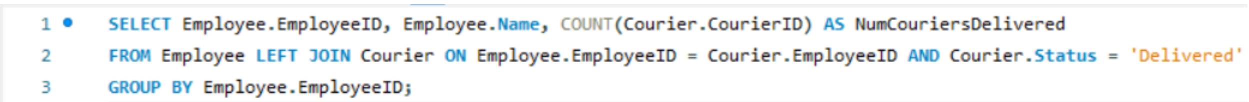
24. Retrieve all couriers sent from the same location (based on SenderAddress).



```sql
SELECT c1.*, c2.*
FROM Courier c1
JOIN Courier c2 ON c1.SenderAddress = c2.SenderAddress AND c1.CourierID <> c2.CourierID;
```

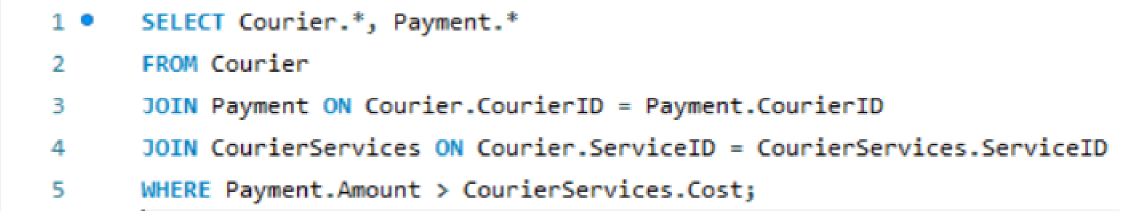
	CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDat

25. List employees and the number of couriers they have delivered:



```sql
SELECT Employee.EmployeeID, Employee.Name, COUNT(Courier.CourierID) AS NumCouriersDelivered
FROM Employee LEFT JOIN Courier ON Employee.EmployeeID = Courier.EmployeeID AND Courier.Status = 'Delivered'
GROUP BY Employee.EmployeeID;
```

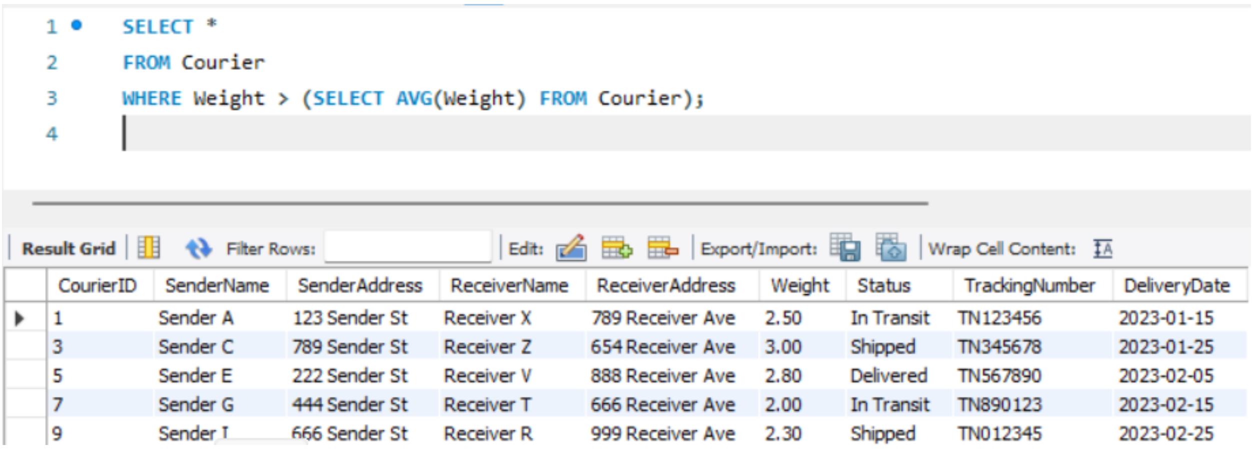
26. Find couriers that were paid an amount greater than the cost of their respective courier services.



```sql
SELECT Courier.*, Payment.*
FROM Courier
JOIN Payment ON Courier.CourierID = Payment.CourierID
JOIN CourierServices ON Courier.ServiceID = CourierServices.ServiceID
WHERE Payment.Amount > CourierServices.Cost;
```

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

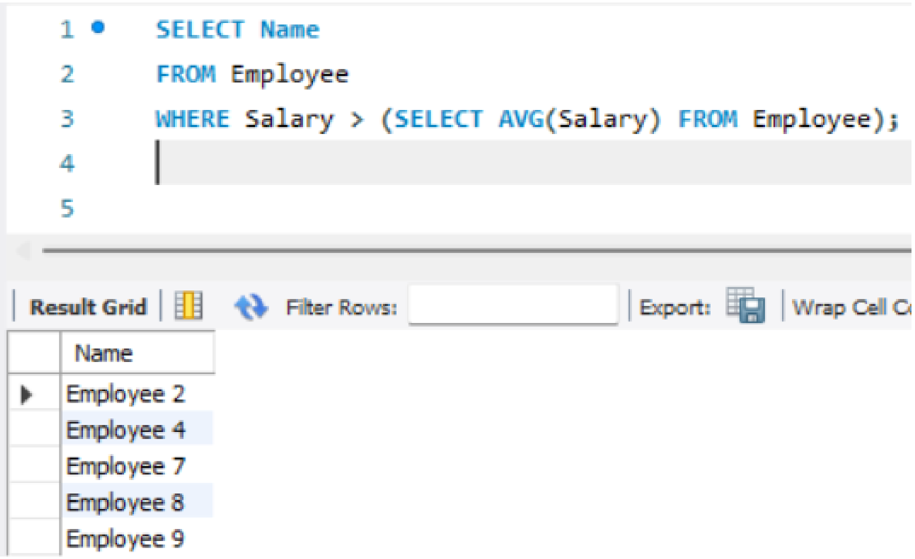
1. Find couriers that have a weight greater than the average weight of all couriers.



```sql
SELECT *
FROM Courier
WHERE Weight > (SELECT AVG(Weight) FROM Courier);
```

	CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
▶	1	Sender A	123 Sender St	Receiver X	789 Receiver Ave	2.50	In Transit	TN123456	2023-01-15
	3	Sender C	789 Sender St	Receiver Z	654 Receiver Ave	3.00	Shipped	TN345678	2023-01-25
	5	Sender E	222 Sender St	Receiver V	888 Receiver Ave	2.80	Delivered	TN567890	2023-02-05
	7	Sender G	444 Sender St	Receiver T	666 Receiver Ave	2.00	In Transit	TN890123	2023-02-15
	9	Sender I	666 Sender St	Receiver R	999 Receiver Ave	2.30	Shipped	TN012345	2023-02-25

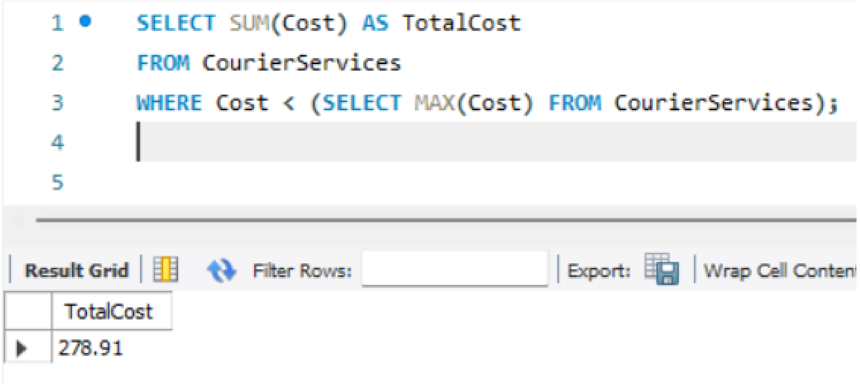
2. Find the names of all employees who have a salary greater than the average salary.



```sql
SELECT Name
FROM Employee
WHERE Salary > (SELECT AVG(Salary) FROM Employee);
```

Name
Employee 2
Employee 4
Employee 7
Employee 8
Employee 9

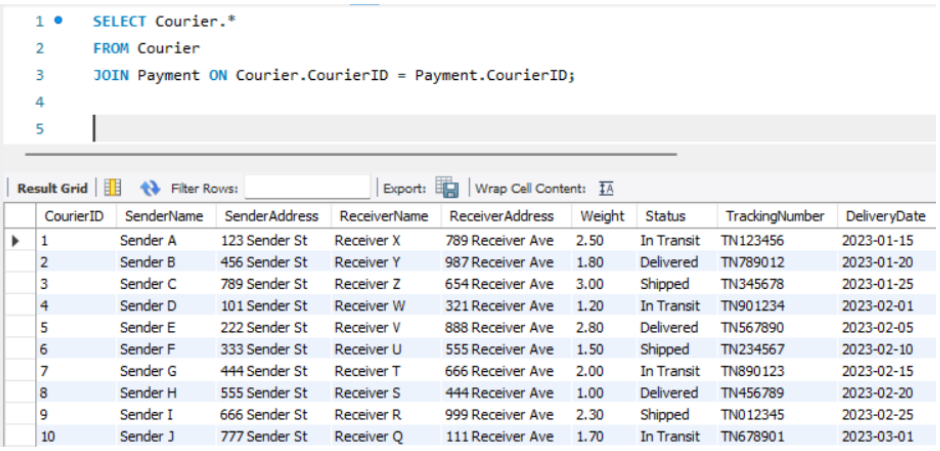
3. Find the total cost of all courier services where the cost is less than the maximum cost.



```sql
SELECT SUM(Cost) AS TotalCost
FROM CourierServices
WHERE Cost < (SELECT MAX(Cost) FROM CourierServices);
```

TotalCost
278.91

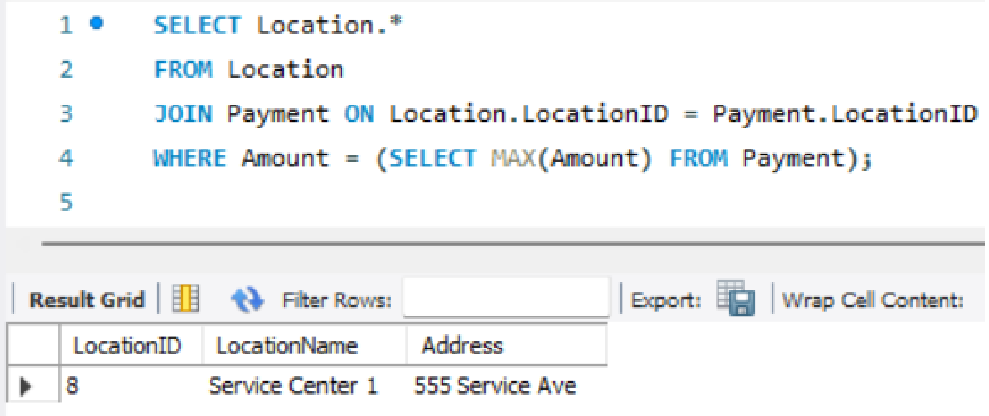
4. Find all couriers that have been paid for



```sql
SELECT Courier.*
FROM Courier
JOIN Payment ON Courier.CourierID = Payment.CourierID;
```

CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
1	Sender A	123 Sender St	Receiver X	789 Receiver Ave	2.50	In Transit	TN123456	2023-01-15
2	Sender B	456 Sender St	Receiver Y	987 Receiver Ave	1.80	Delivered	TN789012	2023-01-20
3	Sender C	789 Sender St	Receiver Z	654 Receiver Ave	3.00	Shipped	TN345678	2023-01-25
4	Sender D	101 Sender St	Receiver W	321 Receiver Ave	1.20	In Transit	TN901234	2023-02-01
5	Sender E	222 Sender St	Receiver V	888 Receiver Ave	2.80	Delivered	TN567890	2023-02-05
6	Sender F	333 Sender St	Receiver U	555 Receiver Ave	1.50	Shipped	TN234567	2023-02-10
7	Sender G	444 Sender St	Receiver T	666 Receiver Ave	2.00	In Transit	TN890123	2023-02-15
8	Sender H	555 Sender St	Receiver S	444 Receiver Ave	1.00	Delivered	TN456789	2023-02-20
9	Sender I	666 Sender St	Receiver R	999 Receiver Ave	2.30	Shipped	TN012345	2023-02-25
10	Sender J	777 Sender St	Receiver Q	111 Receiver Ave	1.70	In Transit	TN678901	2023-03-01

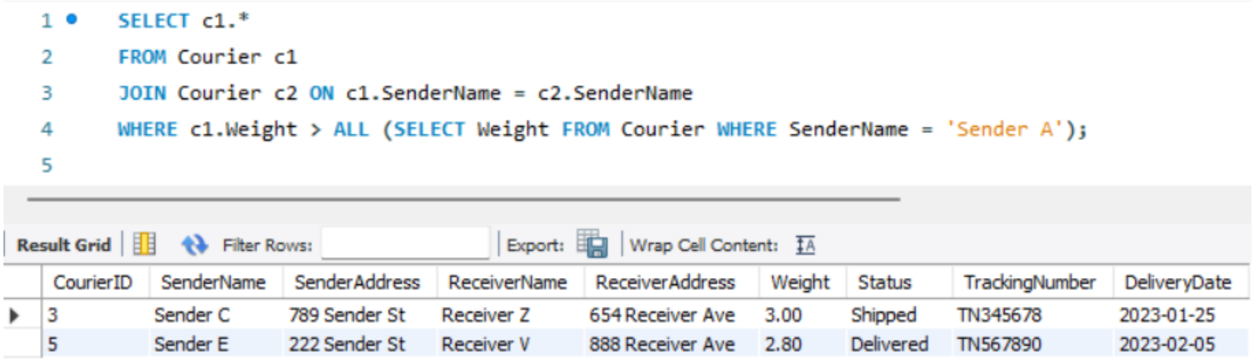
5. Find the locations where the maximum payment amount was made.



```sql
SELECT Location.*
FROM Location
JOIN Payment ON Location.LocationID = Payment.LocationID
WHERE Amount = (SELECT MAX(Amount) FROM Payment);
```

LocationID	LocationName	Address
8	Service Center 1	555 Service Ave

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



```sql
SELECT c1.*
FROM Courier c1
JOIN Courier c2 ON c1.SenderName = c2.SenderName
WHERE c1.Weight > ALL (SELECT Weight FROM Courier WHERE SenderName = 'Sender A');
```

CourierID	SenderName	SenderAddress	ReceiverName	ReceiverAddress	Weight	Status	TrackingNumber	DeliveryDate
3	Sender C	789 Sender St	Receiver Z	654 Receiver Ave	3.00	Shipped	TN345678	2023-01-25
5	Sender E	222 Sender St	Receiver V	888 Receiver Ave	2.80	Delivered	TN567890	2023-02-05