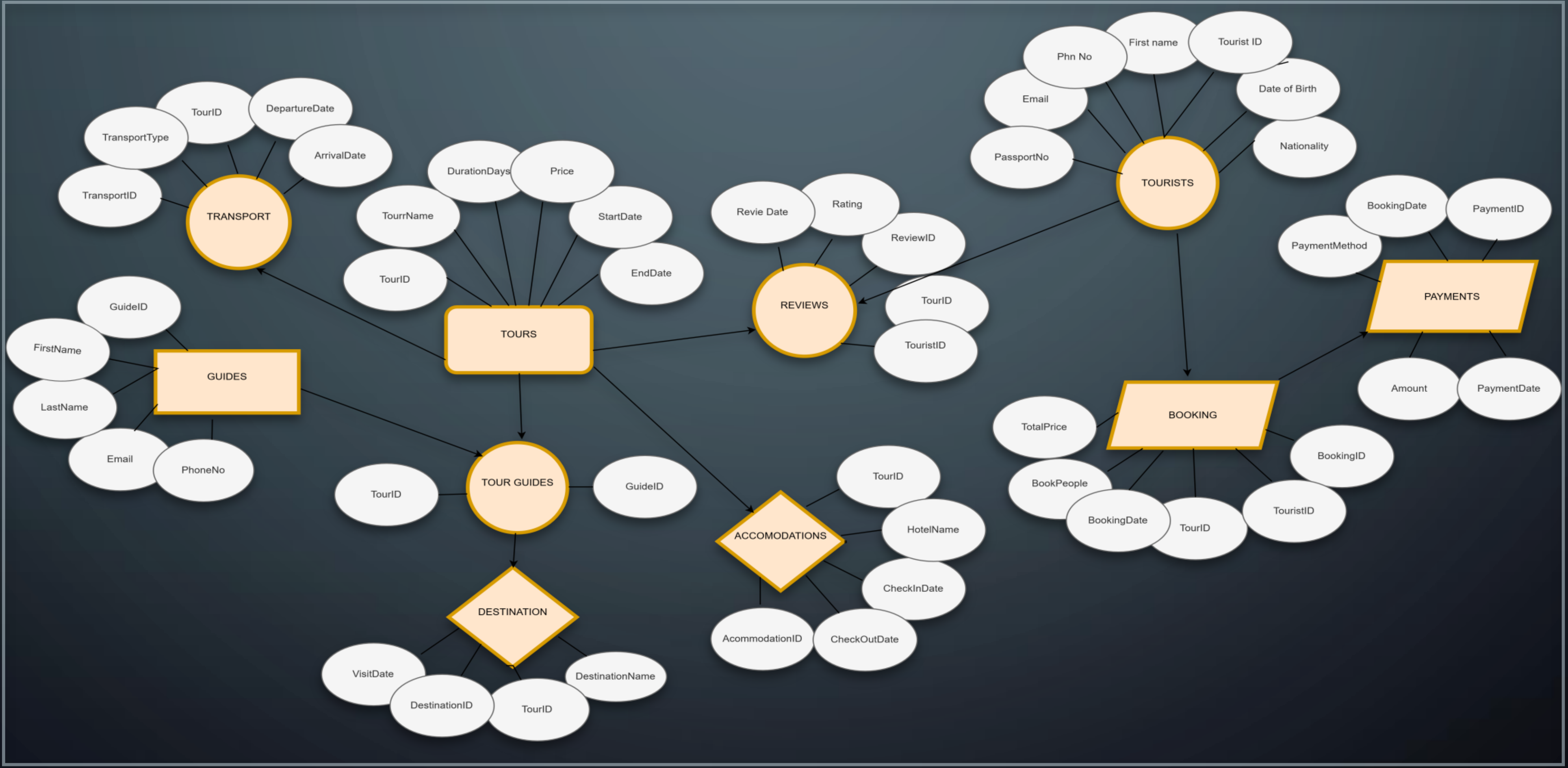
SINDH MADRESSATUL ISLAM UNIVERSITY

TOPIC NAME: TOURISM MANAGEMENT
DEPARTMENT OF SOFTWARE ENGINEERING

GROUP MEMBERS:
AQSA HANIF (BSE-23F-138)
HADIYA MUNIR (BSE-23F-147)
MUTAIBA ALI (BSE-23F-158)

ER MODEL OF TOURISM MANAGEMENT:



WHAT IS TOURISM MANAGEMENT AND HOW IT WORKS:

TOURISM MANAGEMENT:

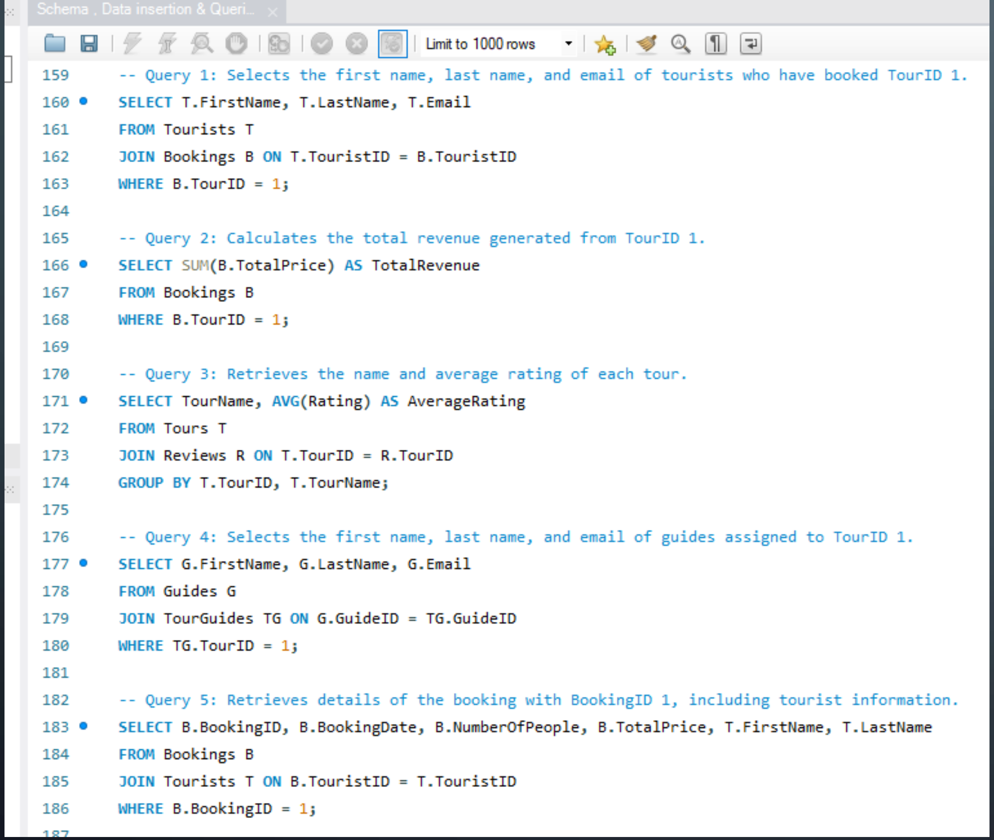
Tourism management involves overseeing and organizing various aspects of the tourism industry, such as planning, marketing, operations, and customer service. It focuses on managing destinations, attractions, accommodations, transportation, and activities to ensure a positive experience for travelers.

HOW IT WORKS:

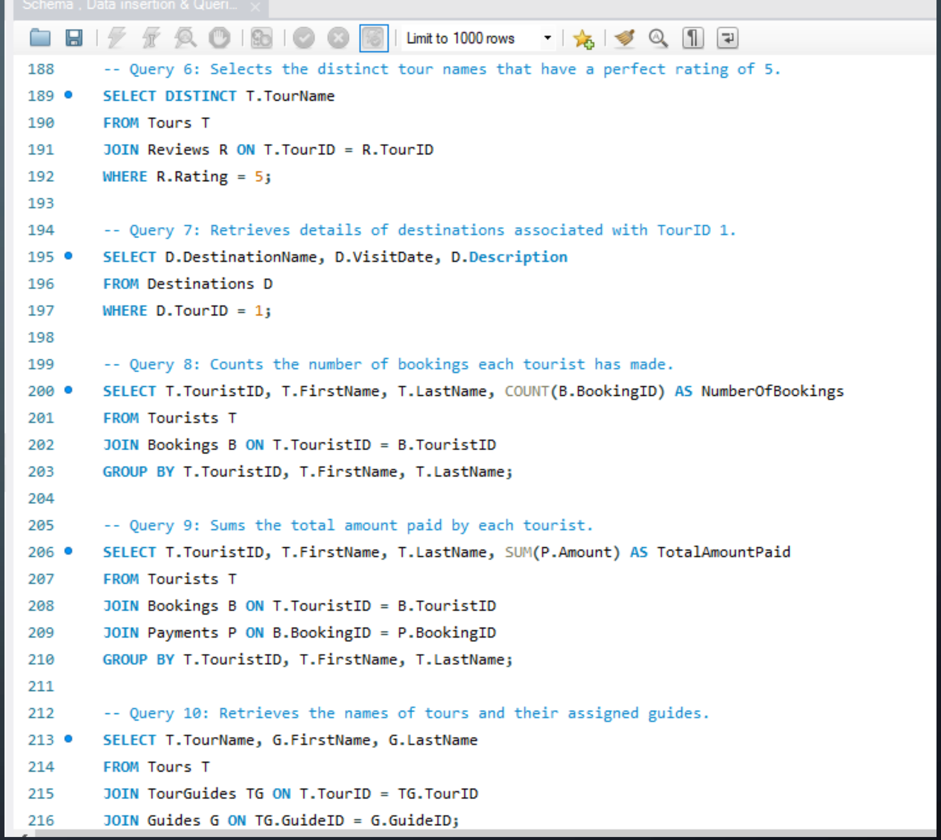
In tourism management, businesses use databases to keep track of all the important information like customer details, bookings, accommodations, and attractions. These databases help in organizing data efficiently and making it easier to access when needed. Additionally, tourism management databases allow businesses to run queries to extract specific data such as customer preferences, popular destinations, revenue reports, and booking patterns.

CONTRIBUTION OF EACH MEMBER:

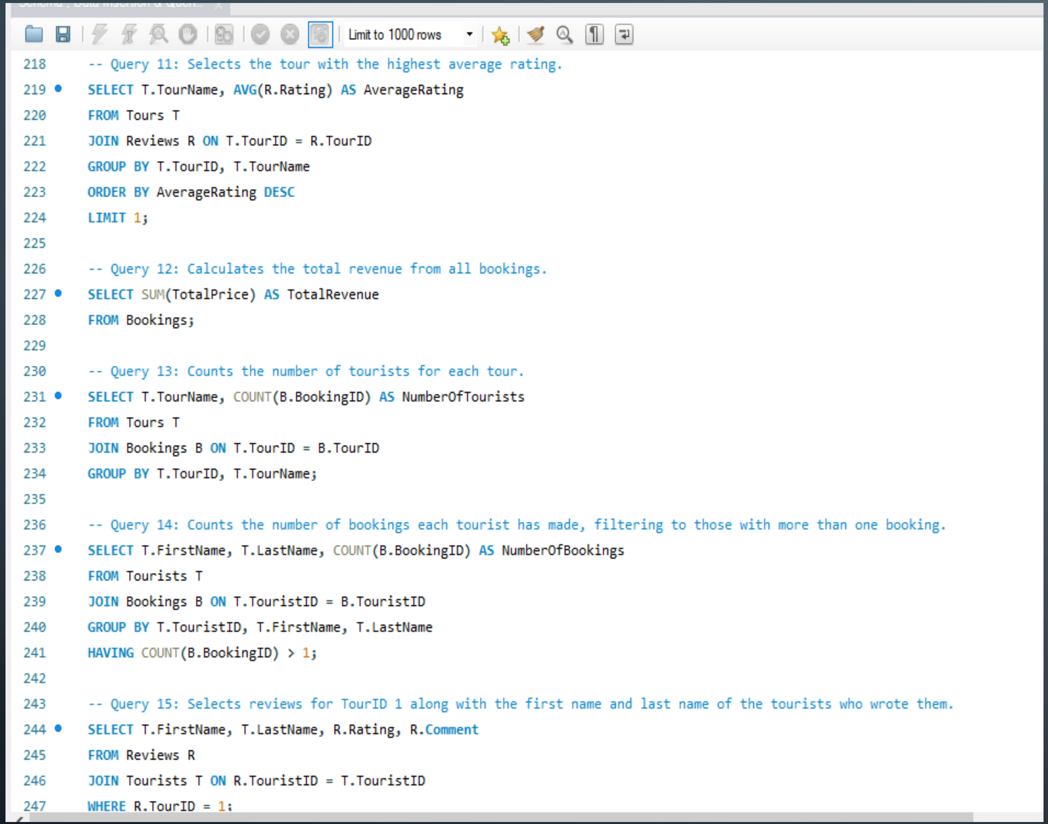
MUTAIBA ALI:



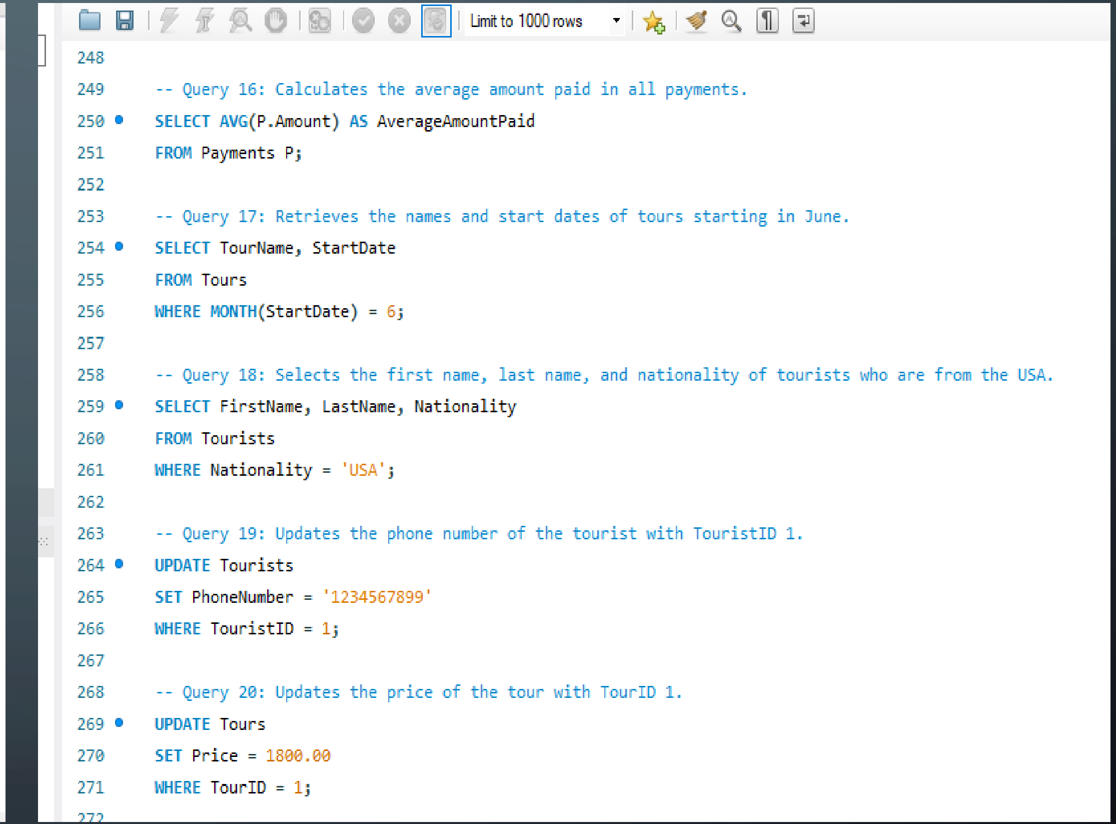
```sql
-- Query 1: Selects the first name, last name, and email of tourists who have booked TourID 1.
SELECT T.FirstName, T.LastName, T.Email
FROM Tourists T
JOIN Bookings B ON T.TouristID = B.TouristID
WHERE B.TourID = 1;

-- Query 2: Calculates the total revenue generated from TourID 1.
SELECT SUM(B.TotalPrice) AS TotalRevenue
FROM Bookings B
WHERE B.TourID = 1;

-- Query 3: Retrieves the name and average rating of each tour.
SELECT TourName, AVG(Rating) AS AverageRating
FROM Tours T
JOIN Reviews R ON T.TourID = R.TourID
GROUP BY T.TourID, T.TourName;

-- Query 4: Selects the first name, last name, and email of guides assigned to TourID 1.
SELECT G.FirstName, G.LastName, G.Email
FROM Guides G
JOIN TourGuides TG ON G.GuideID = TG.GuideID
WHERE TG.TourID = 1;

-- Query 5: Retrieves details of the booking with BookingID 1, including tourist information.
SELECT B.BookingID, B.BookingDate, B.NumberOfPeople, B.TotalPrice, T.FirstName, T.LastName
FROM Bookings B
JOIN Tourists T ON B.TouristID = T.TouristID
WHERE B.BookingID = 1;
```



```sql
-- Query 6: Selects the distinct tour names that have a perfect rating of 5.
SELECT DISTINCT T.TourName
FROM Tours T
JOIN Reviews R ON T.TourID = R.TourID
WHERE R.Rating = 5;

-- Query 7: Retrieves details of destinations associated with TourID 1.
SELECT D.DestinationName, D.VisitDate, D.Description
FROM Destinations D
WHERE D.TourID = 1;

-- Query 8: Counts the number of bookings each tourist has made.
SELECT T.TouristID, T.FirstName, T.LastName, COUNT(B.BookingID) AS NumberOfBookings
FROM Tourists T
JOIN Bookings B ON T.TouristID = B.TouristID
GROUP BY T.TouristID, T.FirstName, T.LastName;

-- Query 9: Sums the total amount paid by each tourist.
SELECT T.TouristID, T.FirstName, T.LastName, SUM(P.Amount) AS TotalAmountPaid
FROM Tourists T
JOIN Bookings B ON T.TouristID = B.TouristID
JOIN Payments P ON B.BookingID = P.BookingID
GROUP BY T.TouristID, T.FirstName, T.LastName;

-- Query 10: Retrieves the names of tours and their assigned guides.
SELECT T.TourName, G.FirstName, G.LastName
FROM Tours T
JOIN TourGuides TG ON T.TourID = TG.TourID
JOIN Guides G ON TG.GuideID = G.GuideID;
```

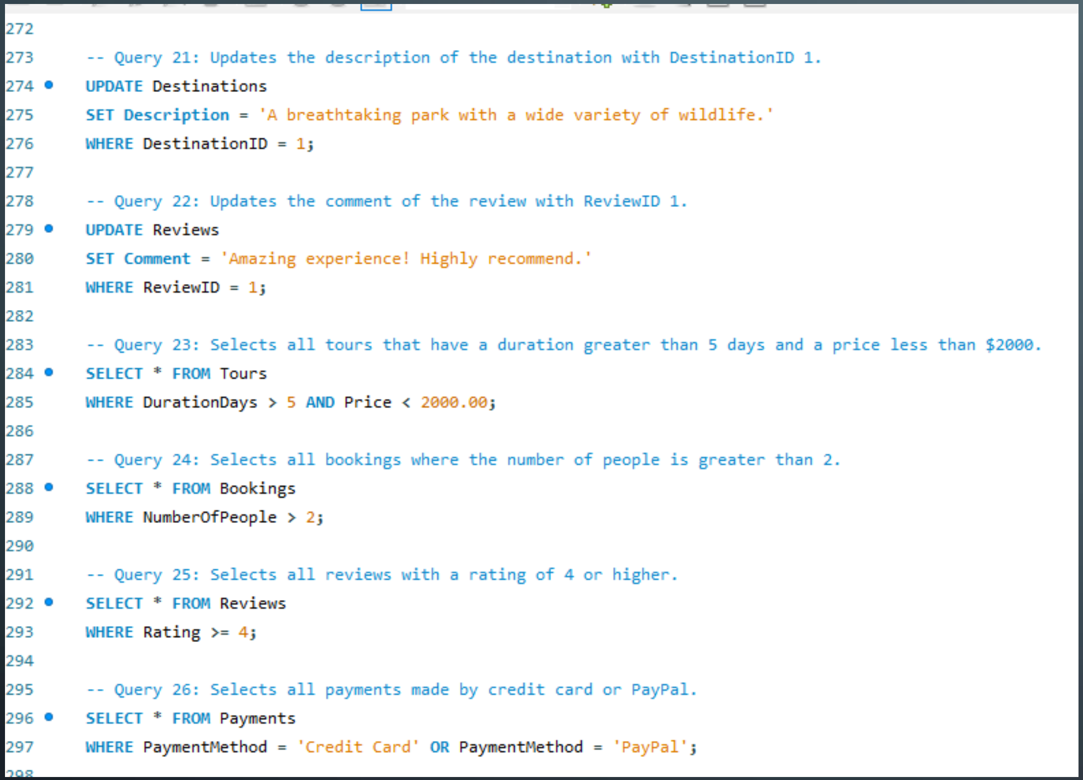


```sql
-- Query 11: Selects the tour with the highest average rating.
SELECT T.TourName, AVG(R.Rating) AS AverageRating
FROM Tours T
JOIN Reviews R ON T.TourID = R.TourID
GROUP BY T.TourID, T.TourName
ORDER BY AverageRating DESC
LIMIT 1;

-- Query 12: Calculates the total revenue from all bookings.
SELECT SUM(TotalPrice) AS TotalRevenue
FROM Bookings;

-- Query 13: Counts the number of tourists for each tour.
SELECT T.TourName, COUNT(B.BookingID) AS NumberOfTourists
FROM Tours T
JOIN Bookings B ON T.TourID = B.TourID
GROUP BY T.TourID, T.TourName;

-- Query 14: Counts the number of bookings each tourist has made, filtering to those with more than one booking.
SELECT T.FirstName, T.LastName, COUNT(B.BookingID) AS NumberOfBookings
FROM Tourists T
JOIN Bookings B ON T.TouristID = B.TouristID
GROUP BY T.TouristID, T.FirstName, T.LastName
HAVING COUNT(B.BookingID) > 1;

-- Query 15: Selects reviews for TourID 1 along with the first name and last name of the tourists who wrote them.
SELECT T.FirstName, T.LastName, R.Rating, R.Comment
FROM Reviews R
JOIN Tourists T ON R.TouristID = T.TouristID
WHERE R.TourID = 1;
```

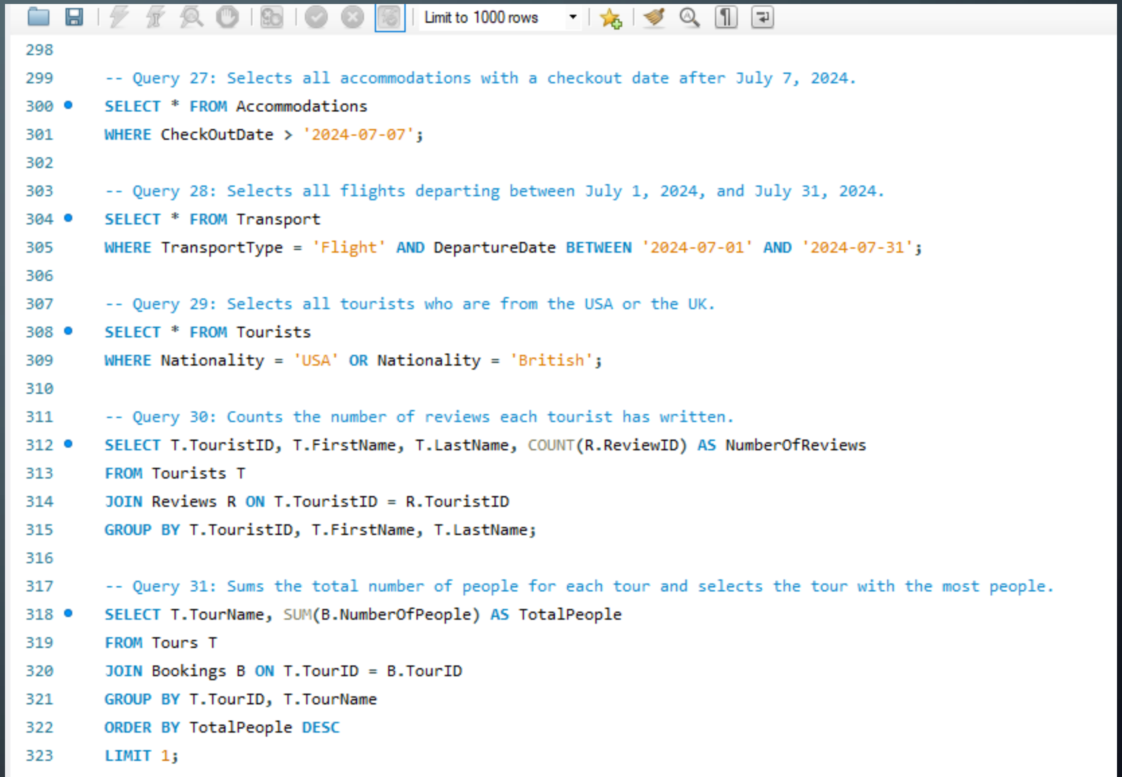


```sql

-- Query 16: Calculates the average amount paid in all payments.
SELECT AVG(P.Amount) AS AverageAmountPaid
FROM Payments P;

-- Query 17: Retrieves the names and start dates of tours starting in June.
SELECT TourName, StartDate
FROM Tours
WHERE MONTH(StartDate) = 6;

-- Query 18: Selects the first name, last name, and nationality of tourists who are from the USA.
SELECT FirstName, LastName, Nationality
FROM Tourists
WHERE Nationality = 'USA';

-- Query 19: Updates the phone number of the tourist with TouristID 1.
UPDATE Tourists
SET PhoneNumber = '1234567899'
WHERE TouristID = 1;

-- Query 20: Updates the price of the tour with TourID 1.
UPDATE Tours
SET Price = 1800.00
WHERE TourID = 1;
```

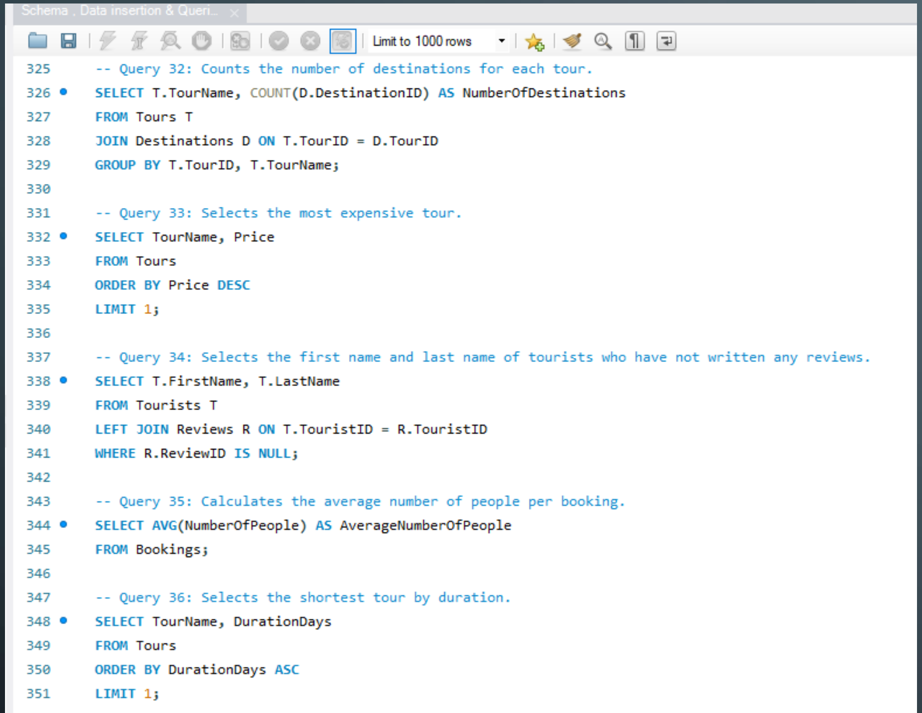
AQSA HANIF:



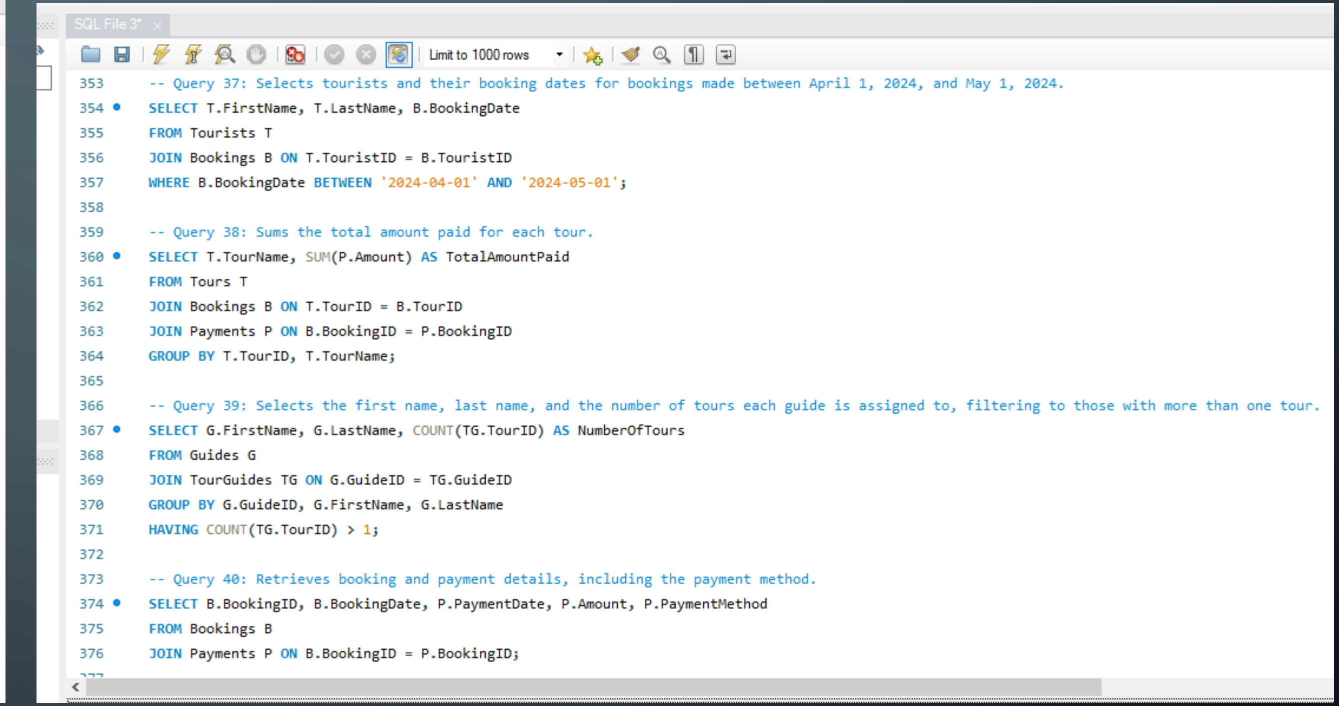
```sql
-- Query 21: Updates the description of the destination with DestinationID 1.
UPDATE Destinations
SET Description = 'A breathtaking park with a wide variety of wildlife.'
WHERE DestinationID = 1;

-- Query 22: Updates the comment of the review with ReviewID 1.
UPDATE Reviews
SET Comment = 'Amazing experience! Highly recommend.'
WHERE ReviewID = 1;

-- Query 23: Selects all tours that have a duration greater than 5 days and a price less than $2000.
SELECT * FROM Tours
WHERE DurationDays > 5 AND Price < 2000.00;

-- Query 24: Selects all bookings where the number of people is greater than 2.
SELECT * FROM Bookings
WHERE NumberOfPeople > 2;

-- Query 25: Selects all reviews with a rating of 4 or higher.
SELECT * FROM Reviews
WHERE Rating >= 4;

-- Query 26: Selects all payments made by credit card or PayPal.
SELECT * FROM Payments
WHERE PaymentMethod = 'Credit Card' OR PaymentMethod = 'PayPal';
```



```sql
-- Query 27: Selects all accommodations with a checkout date after July 7, 2024.
SELECT * FROM Accommodations
WHERE CheckOutDate > '2024-07-07';

-- Query 28: Selects all flights departing between July 1, 2024, and July 31, 2024.
SELECT * FROM Transport
WHERE TransportType = 'Flight' AND DepartureDate BETWEEN '2024-07-01' AND '2024-07-31';

-- Query 29: Selects all tourists who are from the USA or the UK.
SELECT * FROM Tourists
WHERE Nationality = 'USA' OR Nationality = 'British';

-- Query 30: Counts the number of reviews each tourist has written.
SELECT T.TouristID, T.FirstName, T.LastName, COUNT(R.ReviewID) AS NumberOfReviews
FROM Tourists T
JOIN Reviews R ON T.TouristID = R.TouristID
GROUP BY T.TouristID, T.FirstName, T.LastName;

-- Query 31: Sums the total number of people for each tour and selects the tour with the most people.
SELECT T.TourName, SUM(B.NumberOfPeople) AS TotalPeople
FROM Tours T
JOIN Bookings B ON T.TourID = B.TourID
GROUP BY T.TourID, T.TourName
ORDER BY TotalPeople DESC
LIMIT 1;
```

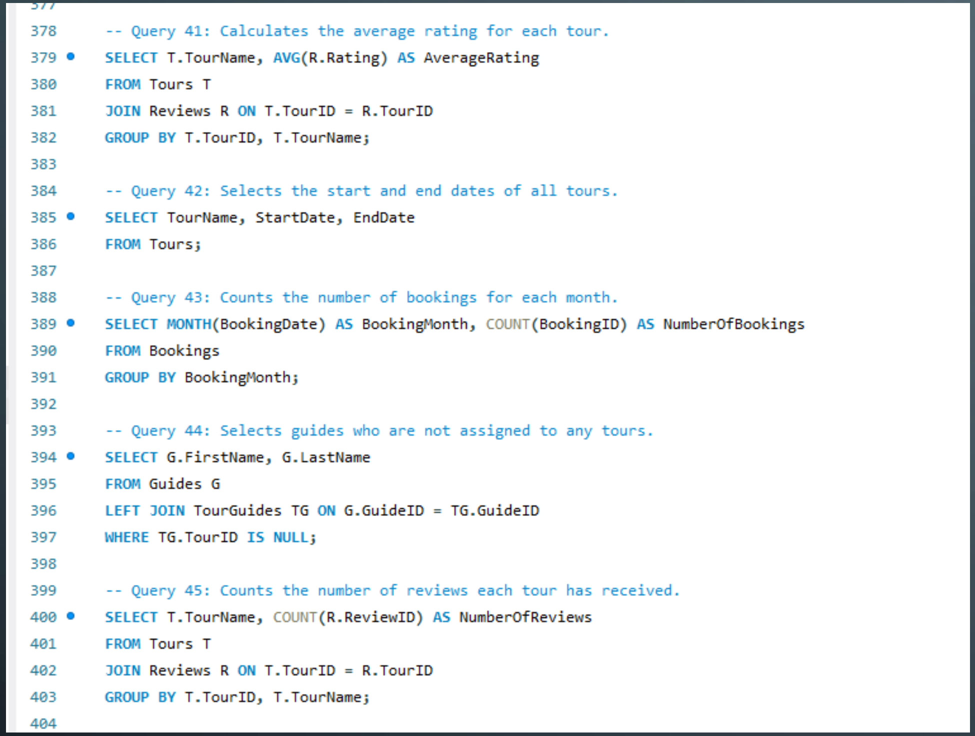


```sql
-- Query 32: Counts the number of destinations for each tour.
SELECT T.TourName, COUNT(D.DestinationID) AS NumberOfDestinations
FROM Tours T
JOIN Destinations D ON T.TourID = D.TourID
GROUP BY T.TourID, T.TourName;

-- Query 33: Selects the most expensive tour.
SELECT TourName, Price
FROM Tours
ORDER BY Price DESC
LIMIT 1;

-- Query 34: Selects the first name and last name of tourists who have not written any reviews.
SELECT T.FirstName, T.LastName
FROM Tourists T
LEFT JOIN Reviews R ON T.TouristID = R.TouristID
WHERE R.ReviewID IS NULL;

-- Query 35: Calculates the average number of people per booking.
SELECT AVG(NumberOfPeople) AS AverageNumberOfPeople
FROM Bookings;

-- Query 36: Selects the shortest tour by duration.
SELECT TourName, DurationDays
FROM Tours
ORDER BY DurationDays ASC
LIMIT 1;
```

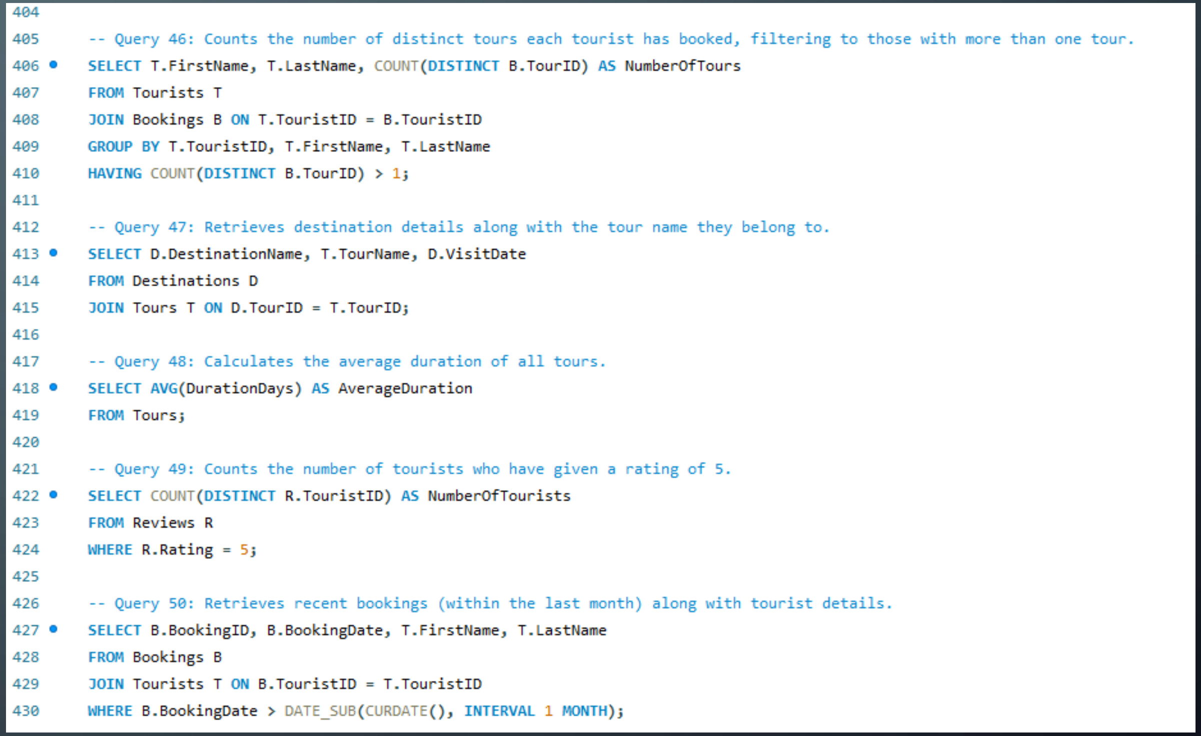


```sql
-- Query 37: Selects tourists and their booking dates for bookings made between April 1, 2024, and May 1, 2024.
SELECT T.FirstName, T.LastName, B.BookingDate
FROM Tourists T
JOIN Bookings B ON T.TouristID = B.TouristID
WHERE B.BookingDate BETWEEN '2024-04-01' AND '2024-05-01';

-- Query 38: Sums the total amount paid for each tour.
SELECT T.TourName, SUM(P.Amount) AS TotalAmountPaid
FROM Tours T
JOIN Bookings B ON T.TourID = B.TourID
JOIN Payments P ON B.BookingID = P.BookingID
GROUP BY T.TourID, T.TourName;

-- Query 39: Selects the first name, last name, and the number of tours each guide is assigned to, filtering to those with more than one tour.
SELECT G.FirstName, G.LastName, COUNT(TG.TourID) AS NumberOfTours
FROM Guides G
JOIN TourGuides TG ON G.GuideID = TG.GuideID
GROUP BY G.GuideID, G.FirstName, G.LastName
HAVING COUNT(TG.TourID) > 1;

-- Query 40: Retrieves booking and payment details, including the payment method.
SELECT B.BookingID, B.BookingDate, P.PaymentDate, P.Amount, P.PaymentMethod
FROM Bookings B
JOIN Payments P ON B.BookingID = P.BookingID;
```

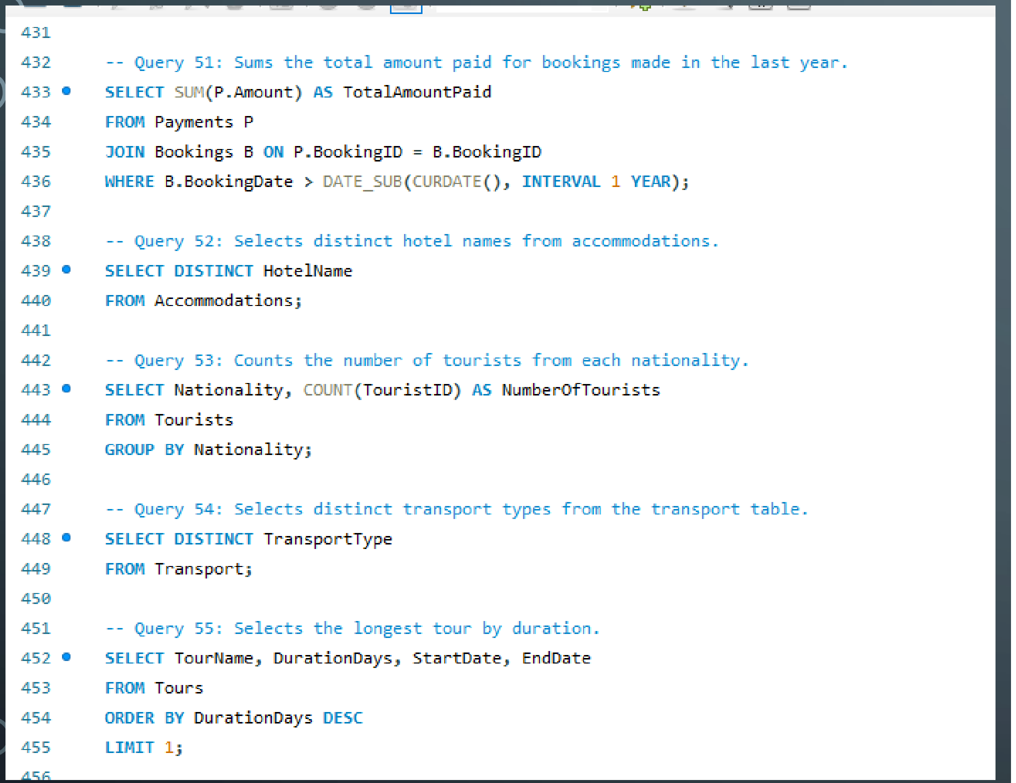
HADIYA MUNIR:



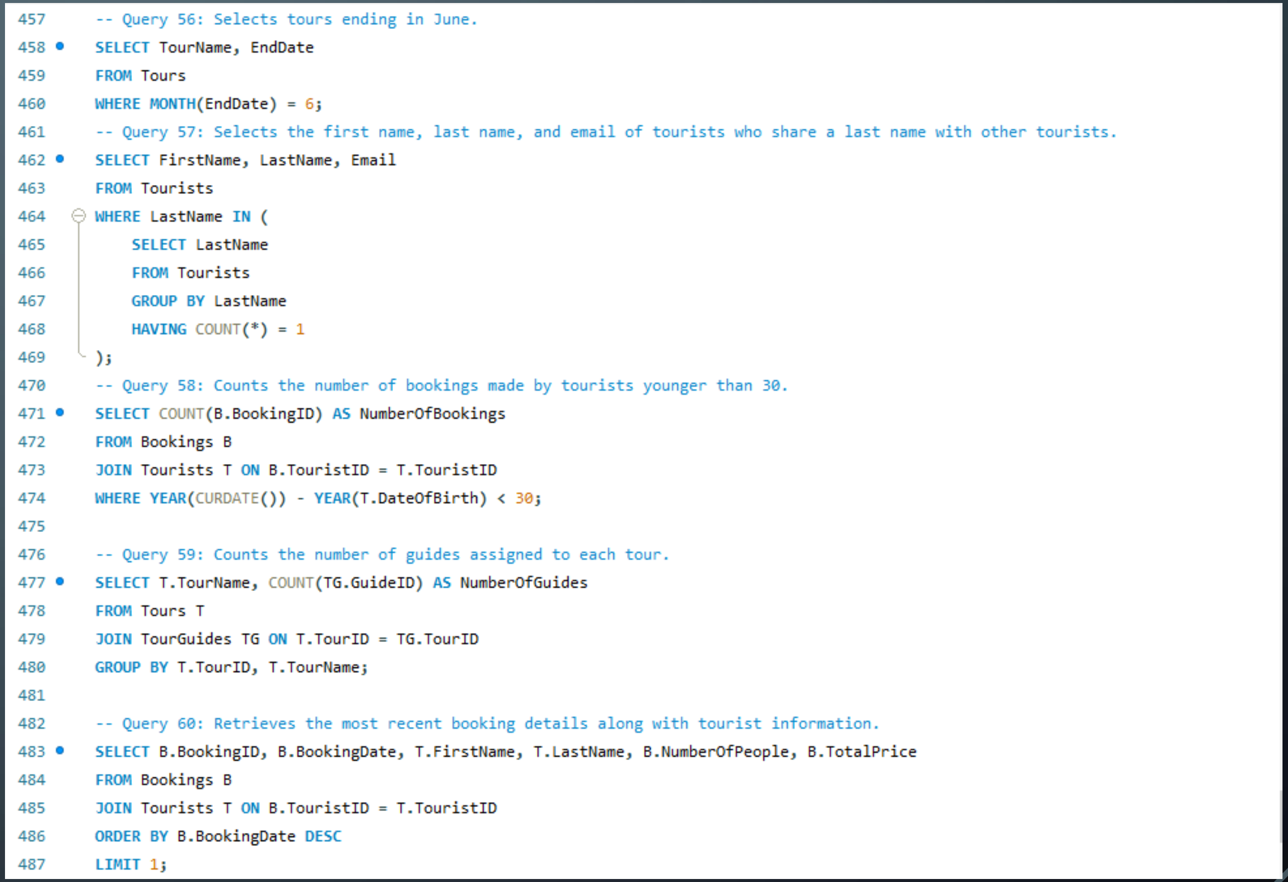
```sql
-- Query 41: Calculates the average rating for each tour.
SELECT T.TourName, AVG(R.Rating) AS AverageRating
FROM Tours T
JOIN Reviews R ON T.TourID = R.TourID
GROUP BY T.TourID, T.TourName;

-- Query 42: Selects the start and end dates of all tours.
SELECT TourName, StartDate, EndDate
FROM Tours;

-- Query 43: Counts the number of bookings for each month.
SELECT MONTH(BookingDate) AS BookingMonth, COUNT(BookingID) AS NumberOfBookings
FROM Bookings
GROUP BY BookingMonth;

-- Query 44: Selects guides who are not assigned to any tours.
SELECT G.FirstName, G.LastName
FROM Guides G
LEFT JOIN TourGuides TG ON G.GuideID = TG.GuideID
WHERE TG.TourID IS NULL;

-- Query 45: Counts the number of reviews each tour has received.
SELECT T.TourName, COUNT(R.ReviewID) AS NumberOfReviews
FROM Tours T
JOIN Reviews R ON T.TourID = R.TourID
GROUP BY T.TourID, T.TourName;
```



```sql
-- Query 46: Counts the number of distinct tours each tourist has booked, filtering to those with more than one tour.
SELECT T.FirstName, T.LastName, COUNT(DISTINCT B.TourID) AS NumberOfTours
FROM Tourists T
JOIN Bookings B ON T.TouristID = B.TouristID
GROUP BY T.TouristID, T.FirstName, T.LastName
HAVING COUNT(DISTINCT B.TourID) > 1;

-- Query 47: Retrieves destination details along with the tour name they belong to.
SELECT D.DestinationName, T.TourName, D.VisitDate
FROM Destinations D
JOIN Tours T ON D.TourID = T.TourID;

-- Query 48: Calculates the average duration of all tours.
SELECT AVG(DurationDays) AS AverageDuration
FROM Tours;

-- Query 49: Counts the number of tourists who have given a rating of 5.
SELECT COUNT(DISTINCT R.TouristID) AS NumberOfTourists
FROM Reviews R
WHERE R.Rating = 5;

-- Query 50: Retrieves recent bookings (within the last month) along with tourist details.
SELECT B.BookingID, B.BookingDate, T.FirstName, T.LastName
FROM Bookings B
JOIN Tourists T ON B.TouristID = T.TouristID
WHERE B.BookingDate > DATE_SUB(CURDATE(), INTERVAL 1 MONTH);
```

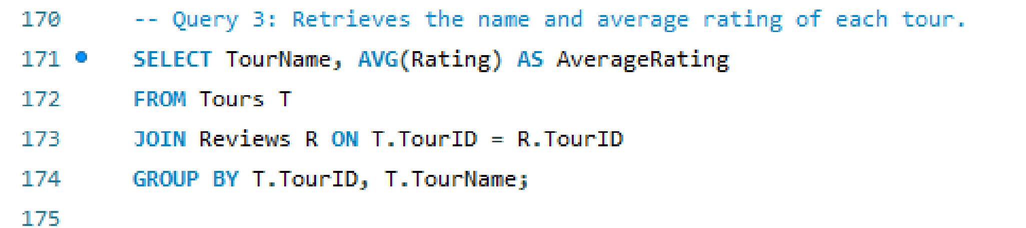


```sql

-- Query 51: Sums the total amount paid for bookings made in the last year.
SELECT SUM(P.Amount) AS TotalAmountPaid
FROM Payments P
JOIN Bookings B ON P.BookingID = B.BookingID
WHERE B.BookingDate > DATE_SUB(CURDATE(), INTERVAL 1 YEAR);

-- Query 52: Selects distinct hotel names from accommodations.
SELECT DISTINCT HotelName
FROM Accommodations;

-- Query 53: Counts the number of tourists from each nationality.
SELECT Nationality, COUNT(TouristID) AS NumberOfTourists
FROM Tourists
GROUP BY Nationality;

-- Query 54: Selects distinct transport types from the transport table.
SELECT DISTINCT TransportType
FROM Transport;

-- Query 55: Selects the longest tour by duration.
SELECT TourName, DurationDays, StartDate, EndDate
FROM Tours
ORDER BY DurationDays DESC
LIMIT 1;
```

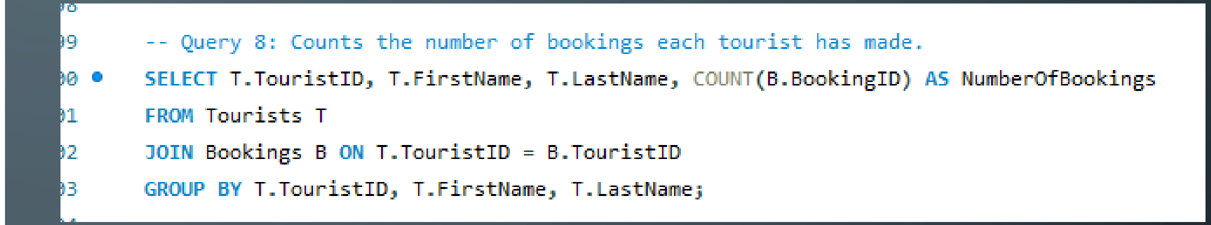


```sql
-- Query 56: Selects tours ending in June.
SELECT TourName, EndDate
FROM Tours
WHERE MONTH(EndDate) = 6;
-- Query 57: Selects the first name, last name, and email of tourists who share a last name with other tourists.
SELECT FirstName, LastName, Email
FROM Tourists
WHERE LastName IN (
    SELECT LastName
    FROM Tourists
    GROUP BY LastName
    HAVING COUNT(*) = 1
);
-- Query 58: Counts the number of bookings made by tourists younger than 30.
SELECT COUNT(B.BookingID) AS NumberOfBookings
FROM Bookings B
JOIN Tourists T ON B.TouristID = T.TouristID
WHERE YEAR(CURDATE()) - YEAR(T.DateOfBirth) < 30;

-- Query 59: Counts the number of guides assigned to each tour.
SELECT T.TourName, COUNT(TG.GuideID) AS NumberOfGuides
FROM Tours T
JOIN TourGuides TG ON T.TourID = TG.TourID
GROUP BY T.TourID, T.TourName;

-- Query 60: Retrieves the most recent booking details along with tourist information.
SELECT B.BookingID, B.BookingDate, T.FirstName, T.LastName, B.NumberOfPeople, B.TotalPrice
FROM Bookings B
JOIN Tourists T ON B.TouristID = T.TouristID
ORDER BY B.BookingDate DESC
LIMIT 1;
```

SCHEMA, DATA INSERTION AND QUERIES:

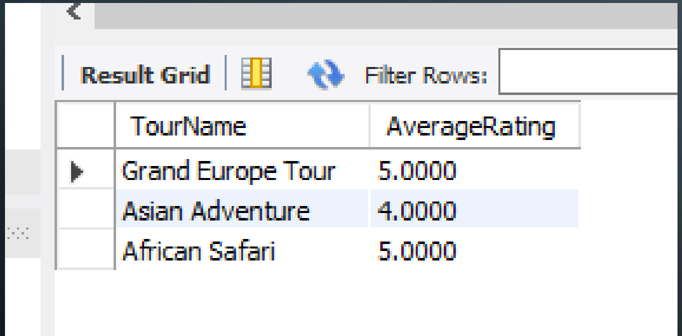


```sql
-- Query 3: Retrieves the name and average rating of each tour.
SELECT TourName, AVG(Rating) AS AverageRating
FROM Tours T
JOIN Reviews R ON T.TourID = R.TourID
GROUP BY T.TourID, T.TourName;
```

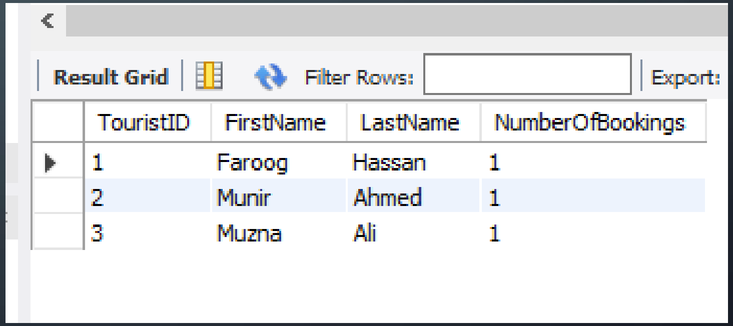


```sql
-- Query 8: Counts the number of bookings each tourist has made.
SELECT T.TouristID, T.FirstName, T.LastName, COUNT(B.BookingID) AS NumberOfBookings
FROM Tourists T
JOIN Bookings B ON T.TouristID = B.TouristID
GROUP BY T.TouristID, T.FirstName, T.LastName;
```

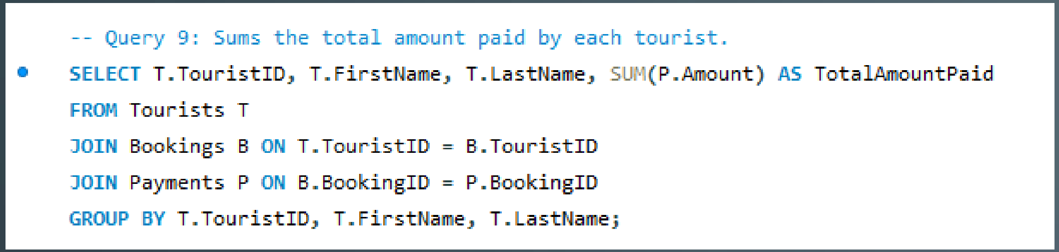
OUTPUT:



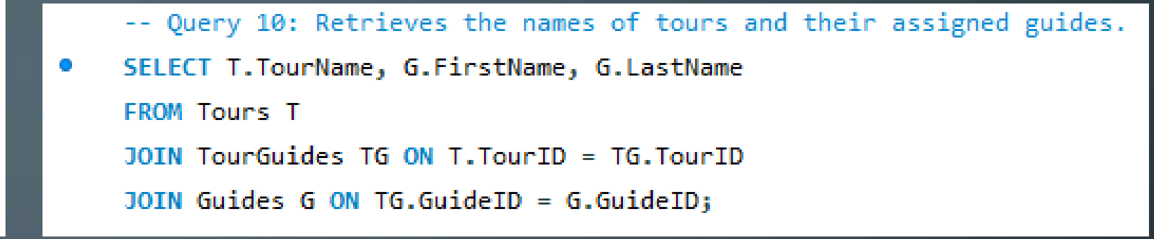
TourName	AverageRating
Grand Europe Tour	5.0000
Asian Adventure	4.0000
African Safari	5.0000



TouristID	FirstName	LastName	NumberOfBookings
1	Farooq	Hassan	1
2	Munir	Ahmed	1
3	Muzna	Ali	1

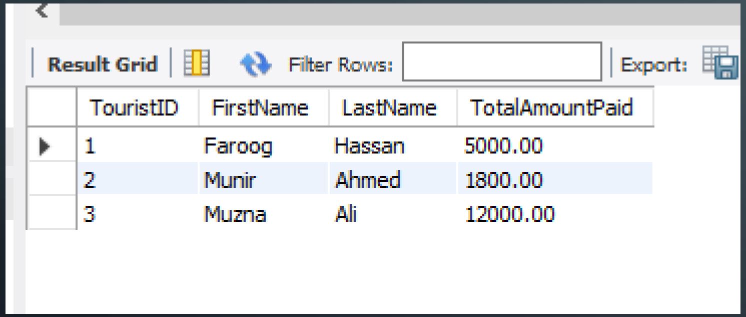


```
-- Query 9: Sums the total amount paid by each tourist.
SELECT T.TouristID, T.FirstName, T.LastName, SUM(P.Amount) AS TotalAmountPaid
FROM Tourists T
JOIN Bookings B ON T.TouristID = B.TouristID
JOIN Payments P ON B.BookingID = P.BookingID
GROUP BY T.TouristID, T.FirstName, T.LastName;
```

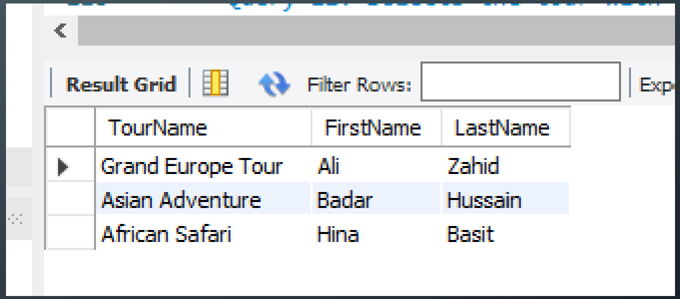


```
-- Query 10: Retrieves the names of tours and their assigned guides.
SELECT T.TourName, G.FirstName, G.LastName
FROM Tours T
JOIN TourGuides TG ON T.TourID = TG.TourID
JOIN Guides G ON TG.GuideID = G.GuideID;
```

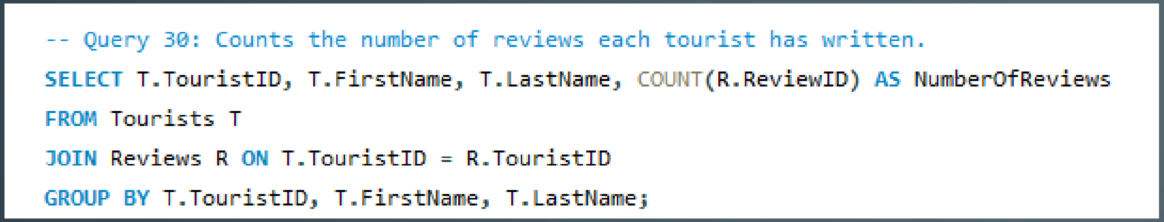
OUTPUT:



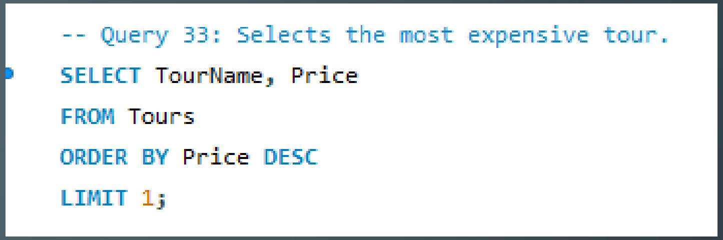
TouristID	FirstName	LastName	TotalAmountPaid
1	Faroog	Hassan	5000.00
2	Munir	Ahmed	1800.00
3	Muzna	Ali	12000.00



TourName	FirstName	LastName
Grand Europe Tour	Ali	Zahid
Asian Adventure	Badar	Hussain
African Safari	Hina	Basit

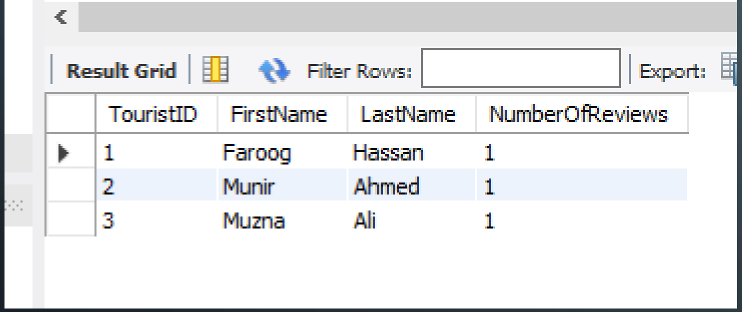


```sql
-- Query 30: Counts the number of reviews each tourist has written.
SELECT T.TouristID, T.FirstName, T.LastName, COUNT(R.ReviewID) AS NumberOfReviews
FROM Tourists T
JOIN Reviews R ON T.TouristID = R.TouristID
GROUP BY T.TouristID, T.FirstName, T.LastName;
```

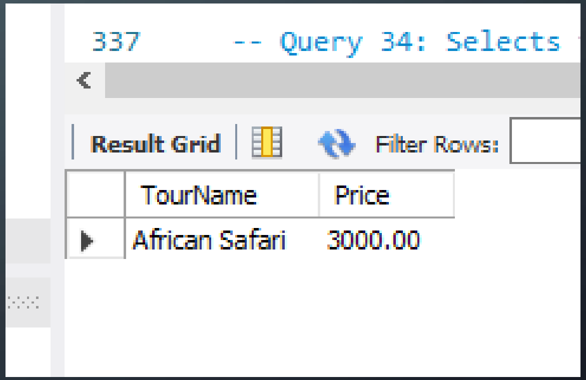


```sql
-- Query 33: Selects the most expensive tour.
SELECT TourName, Price
FROM Tours
ORDER BY Price DESC
LIMIT 1;
```

OUTPUT:

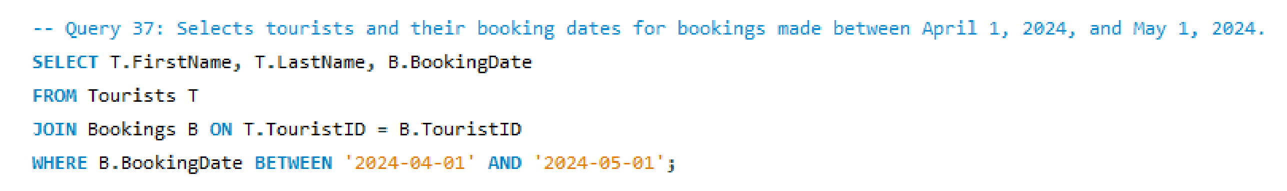


TouristID	FirstName	LastName	NumberOfReviews
1	Faroog	Hassan	1
2	Munir	Ahmed	1
3	Muzna	Ali	1

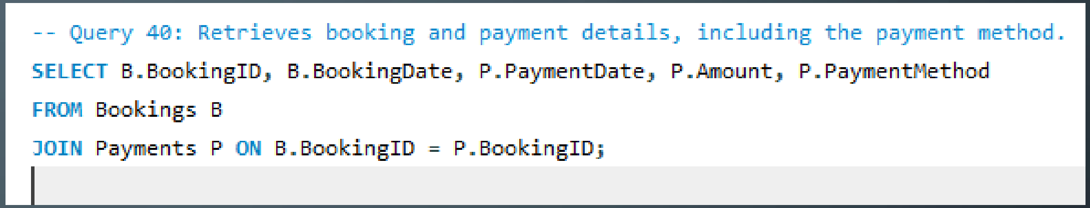


```sql
-- Query 34: Selects
```

TourName	Price
African Safari	3000.00

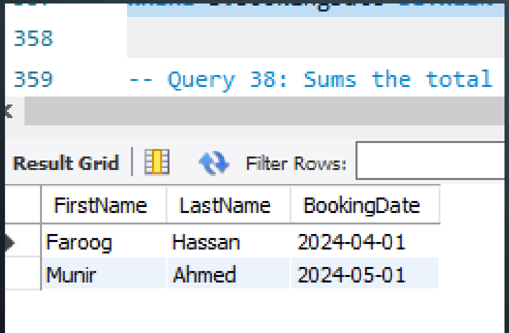


```sql
-- Query 37: Selects tourists and their booking dates for bookings made between April 1, 2024, and May 1, 2024.
SELECT T.FirstName, T.LastName, B.BookingDate
FROM Tourists T
JOIN Bookings B ON T.TouristID = B.TouristID
WHERE B.BookingDate BETWEEN '2024-04-01' AND '2024-05-01';
```



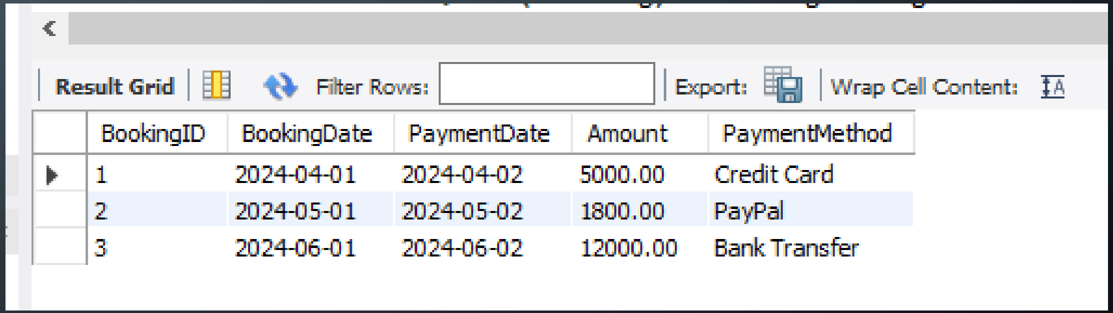
```sql
-- Query 40: Retrieves booking and payment details, including the payment method.
SELECT B.BookingID, B.BookingDate, P.PaymentDate, P.Amount, P.PaymentMethod
FROM Bookings B
JOIN Payments P ON B.BookingID = P.BookingID;
```

OUTPUT:

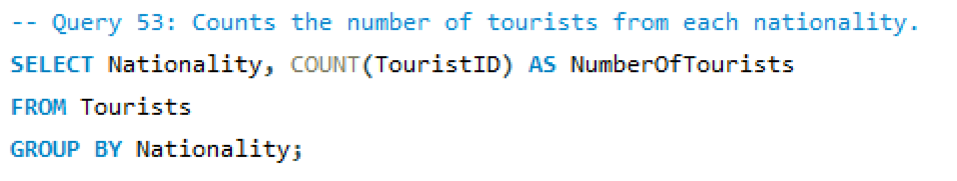


```sql
-- Query 38: Sums the total
```

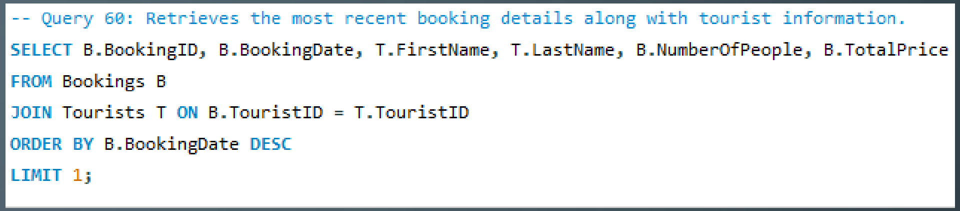
FirstName	LastName	BookingDate
Faroog	Hassan	2024-04-01
Munir	Ahmed	2024-05-01



BookingID	BookingDate	PaymentDate	Amount	PaymentMethod
1	2024-04-01	2024-04-02	5000.00	Credit Card
2	2024-05-01	2024-05-02	1800.00	PayPal
3	2024-06-01	2024-06-02	12000.00	Bank Transfer

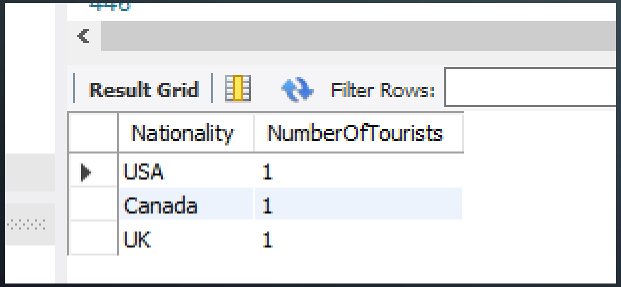


```
-- Query 53: Counts the number of tourists from each nationality.
SELECT Nationality, COUNT(TouristID) AS NumberOfTourists
FROM Tourists
GROUP BY Nationality;
```

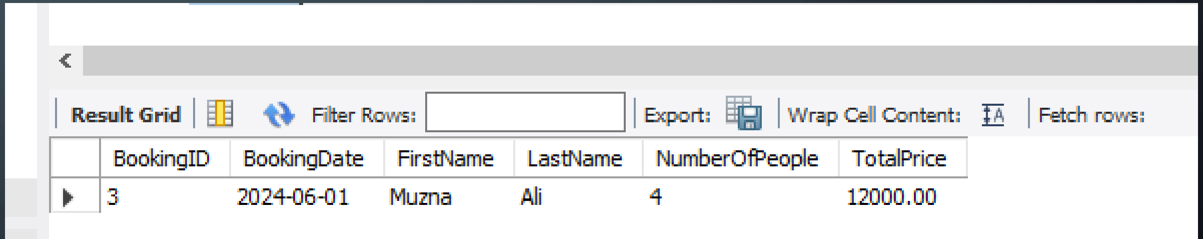


```
-- Query 60: Retrieves the most recent booking details along with tourist information.
SELECT B.BookingID, B.BookingDate, T.FirstName, T.LastName, B.NumberOfPeople, B.TotalPrice
FROM Bookings B
JOIN Tourists T ON B.TouristID = T.TouristID
ORDER BY B.BookingDate DESC
LIMIT 1;
```

OUTPUT:



Nationality	NumberOfTourists
USA	1
Canada	1
UK	1



BookingID	BookingDate	FirstName	LastName	NumberOfPeople	TotalPrice
3	2024-06-01	Muzna	Ali	4	12000.00

THANK YOU!