Base de datos 1 : Blog

1- Tabla autores y blogs.

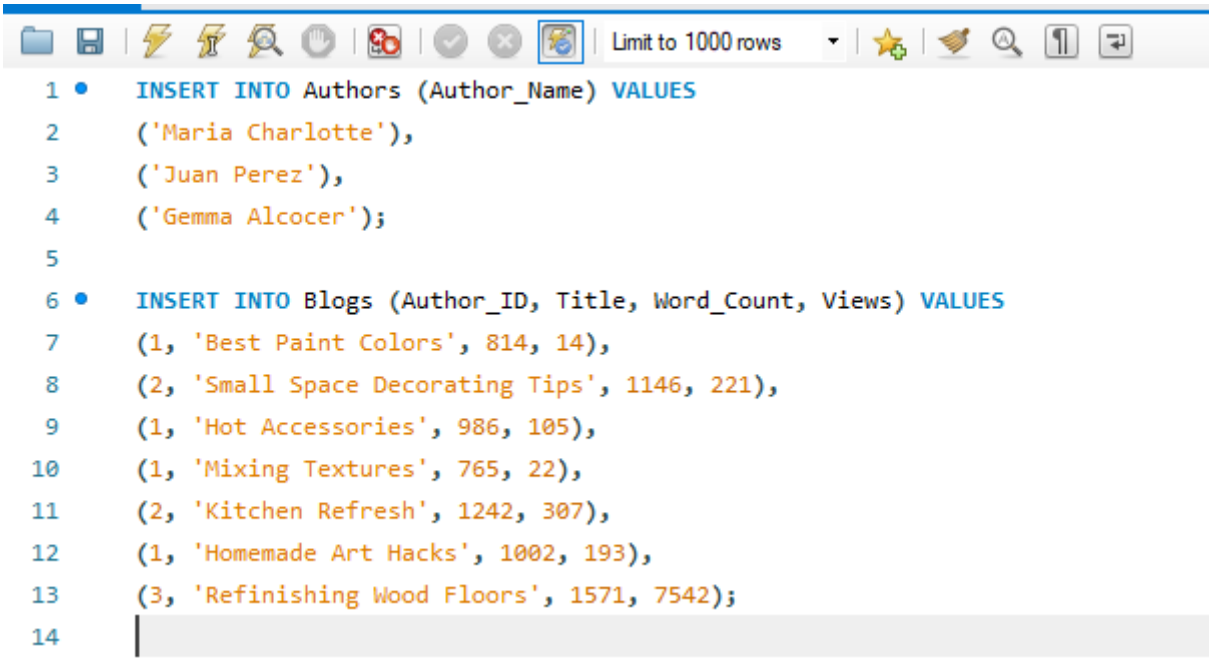
```sql

CREATE TABLE Authors (
    Author_ID INT PRIMARY KEY AUTO_INCREMENT,
    Author_Name VARCHAR(100) NOT NULL
);

CREATE TABLE Blogs (
    Blog_ID INT PRIMARY KEY AUTO_INCREMENT,
    Author_ID INT NOT NULL,
    Title VARCHAR(255) NOT NULL,
    Word_Count INT NOT NULL,
    Views INT NOT NULL,
    FOREIGN KEY (Author_ID) REFERENCES Authors(Author_ID)
);

```

2- Introducir datos en la tabla.



Base de datos 2: Airlane

1- Crear las tablas y añadir datos.

```sql
CREATE TABLE Customers (
    Customer_ID INT PRIMARY KEY AUTO_INCREMENT,
    Customer_Name VARCHAR(100) NOT NULL,
    Customer_Status VARCHAR(20),
    Total_Customer_Mileage INT
);

CREATE TABLE Aircraft (
    Aircraft_ID INT PRIMARY KEY AUTO_INCREMENT,
    Aircraft_Name VARCHAR(100) NOT NULL,
    Total_Seats INT
);

CREATE TABLE Flights (
    Flight_ID INT PRIMARY KEY AUTO_INCREMENT,
    Flight_Number VARCHAR(20) NOT NULL,
    Aircraft_ID INT,
    Flight_Mileage INT,
    FOREIGN KEY (Aircraft_ID) REFERENCES Aircraft(Aircraft_ID)
);
```

```sql
CREATE TABLE Bookings (
    Booking_ID INT PRIMARY KEY AUTO_INCREMENT,
    Customer_ID INT,
    Flight_ID INT,
    FOREIGN KEY (Customer_ID) REFERENCES Customers(Customer_ID),
    FOREIGN KEY (Flight_ID) REFERENCES Flights(Flight_ID)
);
```

```sql
INSERT INTO Flights (Flight_Number, Aircraft_ID, Flight_Mileage) VALUES
('DL143', 1, 135),
('DL122', 2, 4370),
('DL53', 3, 2078),
('DL222', 3, 1765),
('DL37', 1, 531);
```

```sql
INSERT INTO Customers (Customer_Name, Customer_Status, Total_Customer_Mileage) VALUES
('Agustine Riviera', 'Silver', 115235),
('Alaina Sepulvida', 'None', 6008),
('Tom Jones', 'Gold', 205767),
('Sam Rio', 'None', 2653),
('Jessica James', 'Silver', 127656),
('Ana Janco', 'Silver', 136773),
('Jennifer Cortez', 'Gold', 300582),
('Christian Janco', 'Silver', 14642);
```

```sql
INSERT INTO Aircraft (Aircraft_Name, Total_Seats) VALUES
('Boeing 747', 400),
('Airbus A330', 236),
('Boeing 777', 264);
```

```sql
INSERT INTO Bookings (Customer_ID, Flight_ID) VALUES
(1, 1), (1, 2), (1, 1), (1, 1), (1, 1),    -- Agustine Riviera
(2, 2),                                    -- Alaina Sepulvida
(3, 2), (3, 3), (3, 4),                    -- Tom Jones
(4, 1), (4, 1), (4, 5),                    -- Sam Rio
(5, 1), (5, 2),                            -- Jessica James
(6, 4),                                    -- Ana Janco
(7, 4),                                    -- Jennifer Cortez
(8, 4);                                    -- Christian Janco
```

In the Airline database write the SQL script to get the total number of flights in the database.

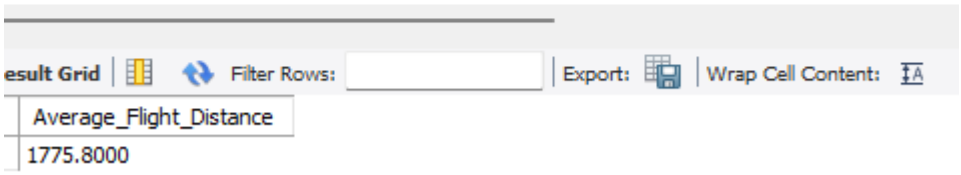
```sql
SELECT COUNT(DISTINCT Flight_Number) AS Total_Flights
FROM Flights;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Total_Flights
5

In the Airline database write the SQL script to get the average flight distance.

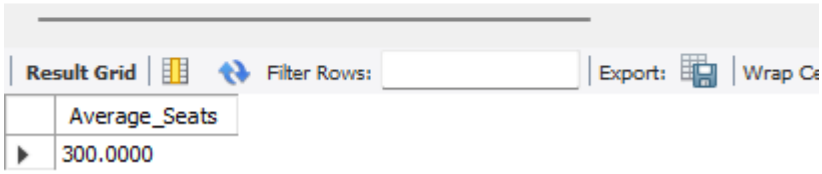
```
SELECT AVG(Flight_Mileage) AS Average_Flight_Distance
FROM Flights;
```


esult Grid | Filter Rows: | Export: | Wrap Cell Content:

Average_Flight_Distance
1775.8000

In the Airline database write the SQL script to get the average number of seats.

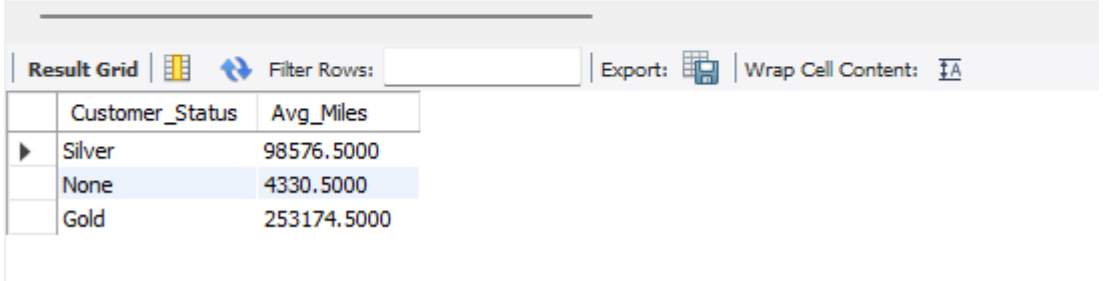
```
SELECT AVG(Total_Seats) AS Average_Seats
FROM Aircraft;
```


Result Grid | Filter Rows: | Export: | Wrap Ce

Average_Seats
300.0000

In the Airline database write the SQL script to get the average number of miles flown by customers grouped by status.

```
SELECT Customer_Status, AVG(Total_Customer_Mileage) AS Avg_Miles
FROM Customers
GROUP BY Customer_Status;
```


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Customer_Status	Avg_Miles
Silver	98576.5000
None	4330.5000
Gold	253174.5000

In the Airline database write the SQL script to get the maximum number of miles flown by customers grouped by status.

```sql
SELECT Customer_Status, MAX(Total_Customer_Mileage) AS Max_Miles
FROM Customers
GROUP BY Customer_Status;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Customer_Status	Max_Miles
Silver	136773
None	6008
Gold	300582

In the Airline database write the SQL script to get the total number of aircraft with a name containing Boeing.

```sql
SELECT COUNT(*) AS Boeing_Aircraft_Count
FROM Aircraft
WHERE Aircraft_Name LIKE '%Boeing%';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Boeing_Aircraft_Count
2

In the Airline database write the SQL script to find all flights with a distance between 300 and 2000 miles.

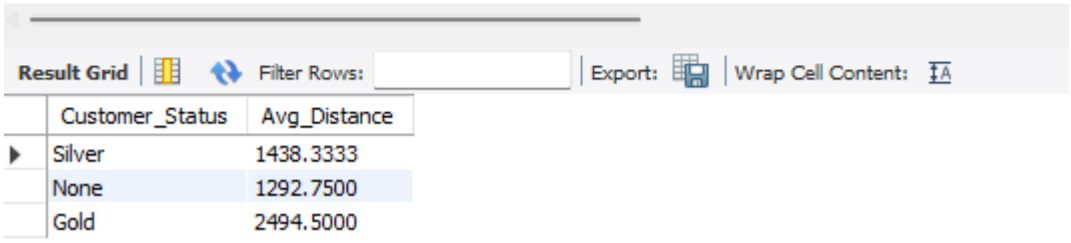
```sql
SELECT Flight_Number, Flight_Mileage
FROM Flights
WHERE Flight_Mileage BETWEEN 300 AND 2000;
```

Result Grid | Filter Rows: | Export: | Wrap Cell

Flight_Number	Flight_Mileage
DL222	1765
DL37	531

In the Airline database write the SQL script to find the average flight distance booked grouped by customer status (this should require a join).

```sql
SELECT c.Customer_Status, AVG(f.Flight_Mileage) AS Avg_Distance
FROM Bookings b
JOIN Customers c ON b.Customer_ID = c.Customer_ID
JOIN Flights f ON b.Flight_ID = f.Flight_ID
GROUP BY c.Customer_Status;
```

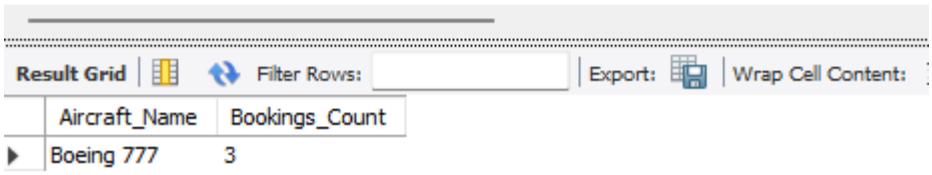


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Customer_Status	Avg_Distance
Silver	1438.3333
None	1292.7500
Gold	2494.5000

In the Airline database write the SQL script to find the most often booked aircraft by gold status members (this should require a join).

```sql
SELECT a.Aircraft_Name, COUNT(*) AS Bookings_Count
FROM Bookings b
JOIN Customers c ON b.Customer_ID = c.Customer_ID
JOIN Flights f ON b.Flight_ID = f.Flight_ID
JOIN Aircraft a ON f.Aircraft_ID = a.Aircraft_ID
WHERE c.Customer_Status = 'Gold'
GROUP BY a.Aircraft_Name
ORDER BY Bookings_Count DESC
LIMIT 1;
```



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Aircraft_Name	Bookings_Count
Boeing 777	3