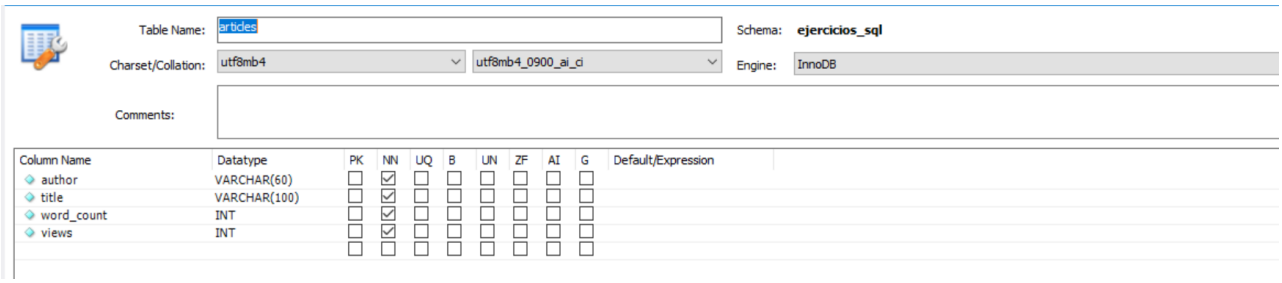
1. Normalize the following blog database and write the DDL scripts to create the database tables:

author	title	word count	views
Maria Charlotte	Best Paint Colors	814	14
Juan Perez	Small Space Decorating Tips	1146	221
Maria Charlotte	Hot Accessories	986	105
Maria Charlotte	Mixing Textures	765	22
Juan Perez	Kitchen Refresh	1242	307
Maria Charlotte	Homemade Art Hacks	1002	193
Gemma Alcocer	Refinishing Wood Floors	1571	7542

Creo una tabla inicial



Inserto los registros en la tabla

```
INSERT INTO articles (author, title, word_count, views)
VALUES
('Maria Charlotte', 'Best Paint Colors', 814, 14),
('Juan Perez', 'Small Space Decorating Tips', 1146, 221),
('Maria Charlotte', 'Hot Accessories', 986, 105),
('Maria Charlotte', 'Mixing Textures', 765, 22),
('Juan Perez', 'Kitchen Refresh', 1242, 307),
('Maria Charlotte', 'Homemade Art Hacks', 1002, 193),
('Gemma Alcocer', 'Refinishing Wood Floors', 1571, 7542);
```

A continuación, muestro todos los registros de la tabla, siguiendo las formas normales, se cumple la 1NF ya que no hay filas duplicadas, es decir, cada fila debe ser única.

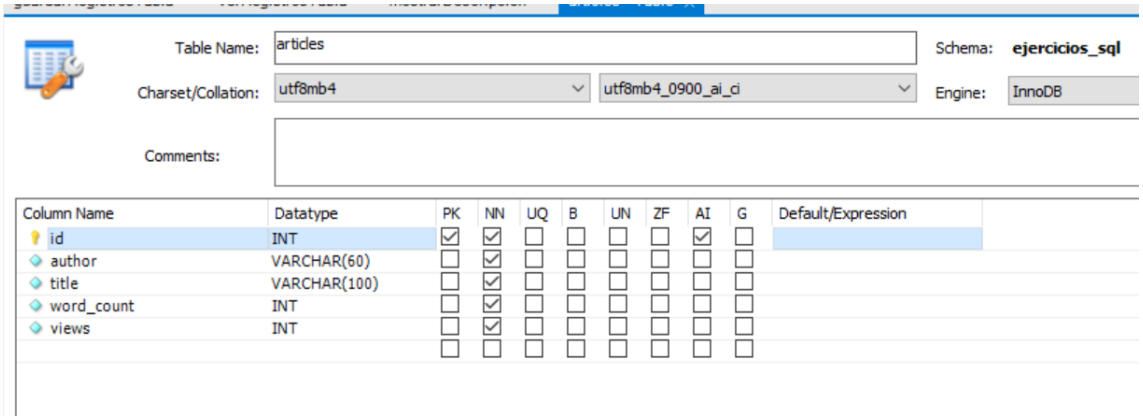
author	title	word_count	views
Maria Charlotte	Best Paint Colors	814	14
Juan Perez	Small Space Decorating Tips	1146	221
Maria Charlotte	Hot Accessories	986	105
Maria Charlotte	Mixing Textures	765	22
Juan Perez	Kitchen Refresh	1242	307
Maria Charlotte	Homemade Art Hacks	1002	193
Gemma Alcocer	Refinishing Wood Floors	1571	7542

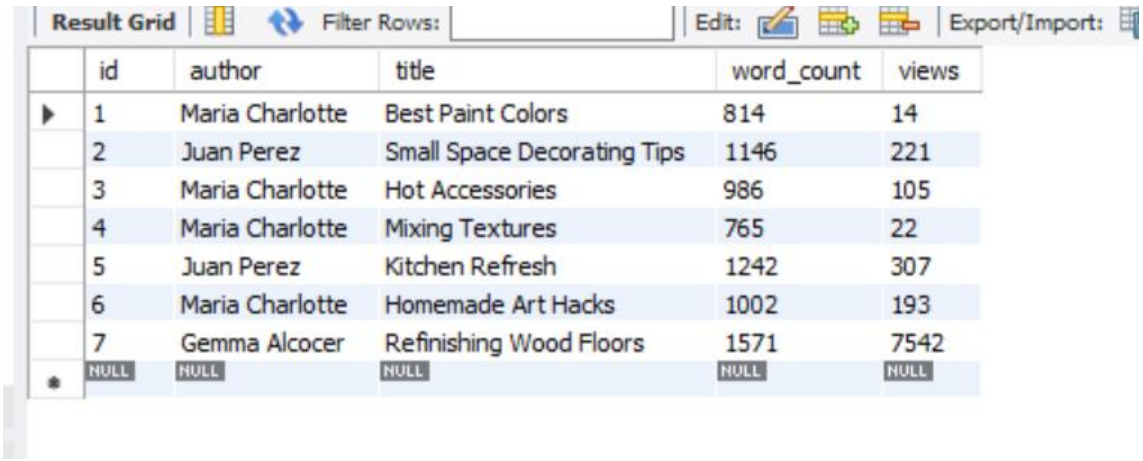
Dado que se cumple la 1NF, modifico la tabla para agregar la clave primaria, ya que si vemos las columnas, no hay clave primaria.

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

Field	Type	Null	Key	Default	Extra
author	varchar(60)	NO		NULL	
title	varchar(100)	NO		NULL	
word_count	int	NO		NULL	
views	int	NO		NULL	

Agrego a la tabla la clave primaria





Lo siguiente que he de verificar el la Forma Normal 3NF, ya que se cumplen tanto la 1NF como la 2NF. Es decir, no puede haber dependencia transitiva, por lo tanto, hay que ver que columnas están relacionadas, y partir de ello, si da el caso de la dependencia transitiva, habrá que crear tablas adicionales.

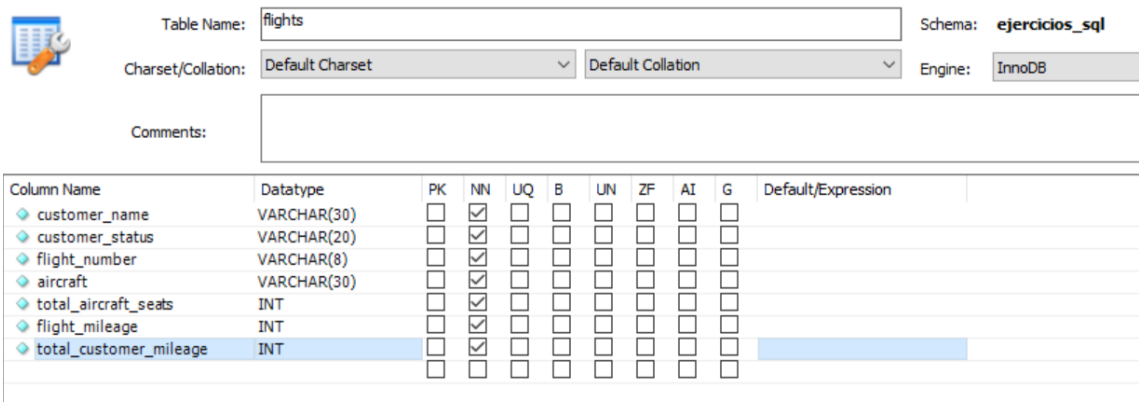
En este caso, no hay dependencias transitivas, por lo tanto, no es necesario crear tablas adicionales.

2. Normalize the following airline database and write the DDL scripts to create the database tables:

Customer Name	Customer Status	Flight Number	Aircraft	Total Aircraft Seats	Flight Mileage	Total Customer Mileage
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Agustine Riviera	Silver	DL122	Airbus A330	236	4370	115235
Alaina Sepulvida	None	DL122	Airbus A330	236	4370	6008
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Tom Jones	Gold	DL122	Airbus A330	236	4370	205767
Tom Jones	Gold	DL53	Boeing 777	264	2078	205767
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Sam Rio	None	DL143	Boeing 747	400	135	2653
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Tom Jones	Gold	DL222	Boeing 777	264	1765	205767
Jessica James	Silver	DL143	Boeing 747	400	135	127656

Customer Name	Customer Status	Flight Number	Aircraft	Total Aircraft Seats	Flight Mileage	Total Customer Mileage
Sam Rio	None	DL143	Boeing 747	400	135	2653
Ana Janco	Silver	DL222	Boeing 777	264	1765	136773
Jennifer Cortez	Gold	DL222	Boeing 777	264	1765	300582
Jessica James	Silver	DL122	Airbus A330	236	4370	127656
Sam Rio	None	DL37	Boeing 747	400	531	2653
Christian Janco	Silver	DL222	Boeing 777	264	1765	14642

Creo la tabla



Guardo los registros en la tabla

```
INSERT INTO flights (customer_name, customer_status, flight_number, aircraft, total_aircraft_seats, flight_mileage, total_customer_mileage)
VALUES
('Agustine Riviera', 'Silver', 'DL143', 'Boeing 747', 400, 135, 115235),
('Agustine Riviera', 'Silver', 'DL122', 'Airbus A330', 236, 4370, 115235),
('Alaina Sepulvida', 'None', 'DL122', 'Airbus A330', 236, 4370, 6008),
('Agustine Riviera', 'Silver', 'DL143', 'Boeing 747', 400, 135, 115235),
('Tom Jones', 'Gold', 'DL122', 'Airbus A330', 236, 4370, 205767),
('Tom Jones', 'Gold', 'DL53', 'Boeing 777', 264, 2078, 205767),
('Agustine Riviera', 'Silver', 'DL143', 'Boeing 747', 400, 135, 115235),
('Sam Rio', 'None', 'DL143', 'Boeing 747', 400, 135, 2653),
('Agustine Riviera', 'Silver', 'DL143', 'Boeing 747', 400, 135, 115235),
('Tom Jones', 'Gold', 'DL222', 'Boeing 777', 264, 1765, 205767),
('Jessica James', 'Silver', 'DL143', 'Boeing 747', 400, 135, 127656),
('Sam Rio', 'None', 'DL143', 'Boeing 747', 400, 135, 2653),
('Ana Janco', 'Silver', 'DL222', 'Boeing 777', 264, 1765, 136773),
('Jennifer Cortez', 'Gold', 'DL222', 'Boeing 777', 264, 1765, 300582),
('Jessica James', 'Silver', 'DL122', 'Airbus A330', 236, 4370, 127656),
('Sam Rio', 'None', 'DL37', 'Boeing 747', 400, 531, 2653),
('Christian Janco', 'Silver', 'DL222', 'Boeing 777', 264, 1765, 14642);
```

La tabla resultante inicial es está

customer_name	customer_status	flight_number	aircraft	total_aircraft_seats	flight_mileage	total_customer_mileage
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Agustine Riviera	Silver	DL122	Airbus A330	236	4370	115235
Alaina Sepulvida	None	DL122	Airbus A330	236	4370	6008
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Tom Jones	Gold	DL122	Airbus A330	236	4370	205767
Tom Jones	Gold	DL53	Boeing 777	264	2078	205767
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Sam Rio	None	DL143	Boeing 747	400	135	2653
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Tom Jones	Gold	DL222	Boeing 777	264	1765	205767
Jessica James	Silver	DL143	Boeing 747	400	135	127656
Sam Rio	None	DL143	Boeing 747	400	135	2653
Ana Janco	Silver	DL222	Boeing 777	264	1765	136773
Jennifer Cortez	Gold	DL222	Boeing 777	264	1765	300582
Jessica James	Silver	DL122	Airbus A330	236	4370	127656
Sam Rio	None	DL37	Boeing 747	400	531	2653
Christian Janco	Silver	DL222	Boeing 777	264	1765	14642

Para comprobar que se cumpla 1NF, al ser una tabla con más registros y hacerlo de forma optima, en vez de verlo a ojo, lo voy a hacer con queries donde filtró resultados que vea que pueden estar duplicados.

Veo que hay varios Agustine Riviera.

```
1 • select * from flights
2   where customer_name = 'Agustine Riviera';
3
4
5
6
7
```

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

customer_name	customer_status	flight_number	aircraft	total_aircraft_seats	flight_mileage	total_customer_mileage
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Agustine Riviera	Silver	DL122	Airbus A330	236	4370	115235
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235

En el numero de vuelo, parece que uno distinto y el resto son iguales, filtro también por número de vuelo, ya que parece que Boeing 747 está repetido

```
1 • select * from flights
2   where customer_name = 'Agustine Riviera' and aircraft = 'Boeing 747';
3
4
5
6
7
```

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

customer_name	customer_status	flight_number	aircraft	total_aircraft_seats	flight_mileage	total_customer_mileage
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235

Ahora se aprecia de que están duplicados, por lo tanto, tengo que eliminar esos 3, para quede 1 con esos valores. Elimino de 1 en 1, ya que al poner el limit 3 me ha borrado todas las coincidencias, así que, ejecuto la queri hasta que me quede en 1.

```
delete from flights
where customer_name = 'Agustine Riviera' and aircraft = 'Boeing 747'
limit 1;
```

Con esto he conseguido hacer que una fila sea única.

```
select * from flights
where customer_name = 'Agustine Riviera' and aircraft = 'Boeing 747';
```

Filter Rows: | Export: | Wrap Cell Content:

customer_name	customer_status	flight_number	aircraft	total_aircraft_seats	flight_mileage	total_customer_mileage
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235

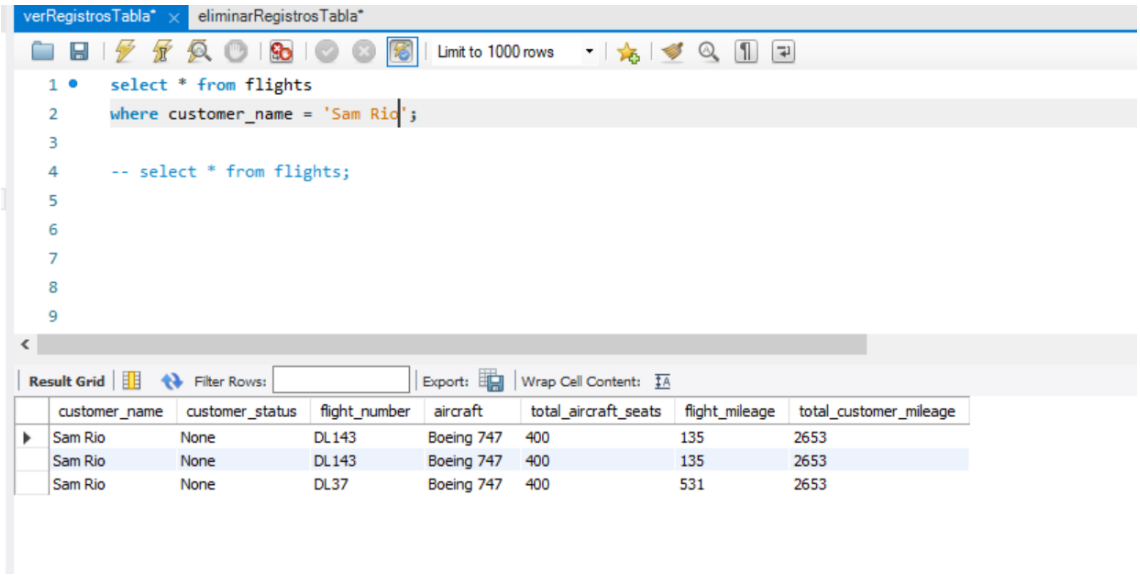
También he apreciado en la tabla que Tom Jones aparece repetido, pero en el resto de columnas, tiene valores distintos, así que no hace falta eliminar.

```
1  select * from flights
2  where customer_name = 'Tom Jones';
3
4  -- select * from flights;
5
6
7
8
9
```

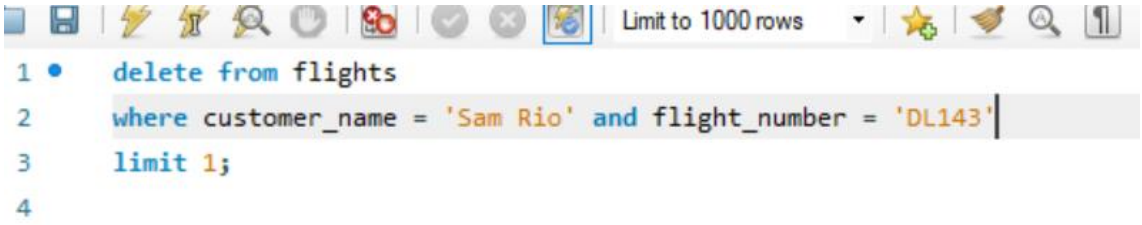
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customer_name	customer_status	flight_number	aircraft	total_aircraft_seats	flight_mileage	total_customer_mileage
Tom Jones	Gold	DL122	Airbus A330	236	4370	205767
Tom Jones	Gold	DL53	Boeing 777	264	2078	205767
Tom Jones	Gold	DL222	Boeing 777	264	1765	205767

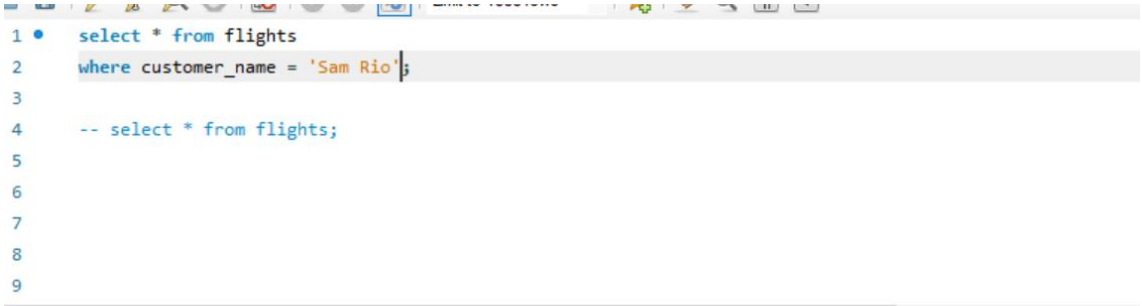
Otro nombre de cliente repetido es Sam Rio y se aprecia que hay 2 filas idénticas, así que habrá que eliminar una de ellas



customer_name	customer_status	flight_number	aircraft	total_aircraft_seats	flight_mileage	total_customer_mileage
Sam Rio	None	DL143	Boeing 747	400	135	2653
Sam Rio	None	DL143	Boeing 747	400	135	2653
Sam Rio	None	DL37	Boeing 747	400	531	2653

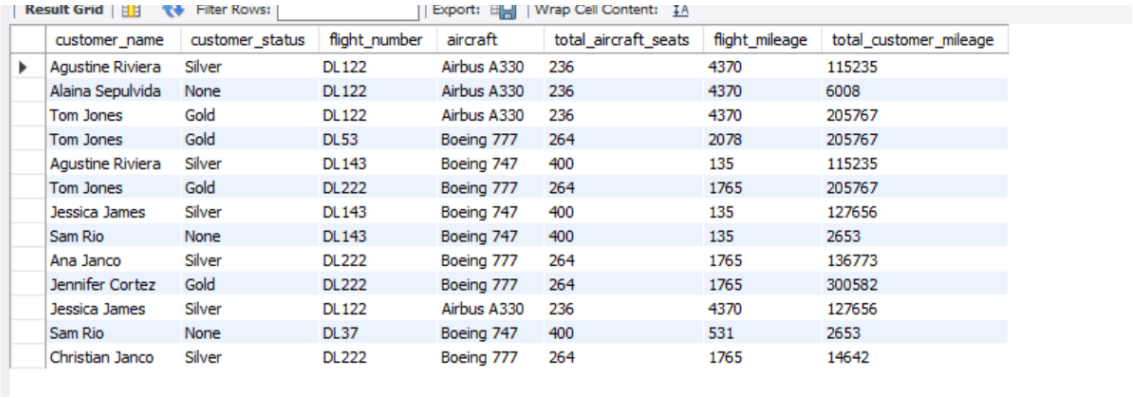


Ahora no hay filas repetidas



customer_name	customer_status	flight_number	aircraft	total_aircraft_seats	flight_mileage	total_customer_mileage
Sam Rio	None	DL143	Boeing 747	400	135	2653
Sam Rio	None	DL37	Boeing 747	400	531	2653

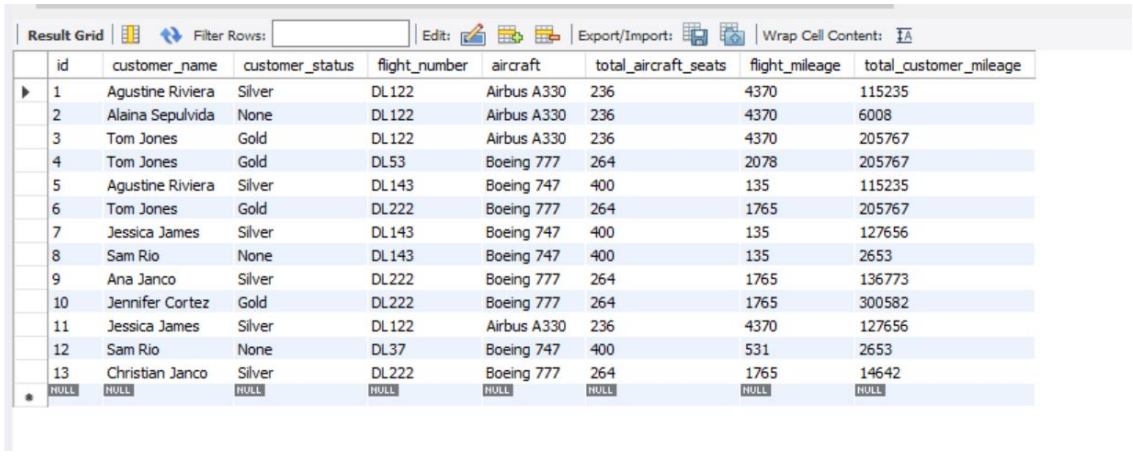
También aparece repetida Jessica James, pero sus filas tienen distintos valores, por lo tanto, no es necesario eliminar ninguna fila. Ya he conseguido que se cumpla 1NF.



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

customer_name	customer_status	flight_number	aircraft	total_aircraft_seats	flight_mileage	total_customer_mileage
Agustine Riviera	Silver	DL122	Airbus A330	236	4370	115235
Alaina Sepulvida	None	DL122	Airbus A330	236	4370	6008
Tom Jones	Gold	DL122	Airbus A330	236	4370	205767
Tom Jones	Gold	DL53	Boeing 777	264	2078	205767
Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
Tom Jones	Gold	DL222	Boeing 777	264	1765	205767
Jessica James	Silver	DL143	Boeing 747	400	135	127656
Sam Rio	None	DL143	Boeing 747	400	135	2653
Ana Janco	Silver	DL222	Boeing 777	264	1765	136773
Jennifer Cortez	Gold	DL222	Boeing 777	264	1765	300582
Jessica James	Silver	DL122	Airbus A330	236	4370	127656
Sam Rio	None	DL37	Boeing 747	400	531	2653
Christian Janco	Silver	DL222	Boeing 777	264	1765	14642

Como se cumple 1NF, lo siguiente que hay que hacer es añadir la columna que hace referencia a la clave primaria, que sería el id, de tipo int y con auto increment para que se inserte automáticamente.



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	customer_name	customer_status	flight_number	aircraft	total_aircraft_seats	flight_mileage	total_customer_mileage
1	Agustine Riviera	Silver	DL122	Airbus A330	236	4370	115235
2	Alaina Sepulvida	None	DL122	Airbus A330	236	4370	6008
3	Tom Jones	Gold	DL122	Airbus A330	236	4370	205767
4	Tom Jones	Gold	DL53	Boeing 777	264	2078	205767
5	Agustine Riviera	Silver	DL143	Boeing 747	400	135	115235
6	Tom Jones	Gold	DL222	Boeing 777	264	1765	205767
7	Jessica James	Silver	DL143	Boeing 747	400	135	127656
8	Sam Rio	None	DL143	Boeing 747	400	135	2653
9	Ana Janco	Silver	DL222	Boeing 777	264	1765	136773
10	Jennifer Cortez	Gold	DL222	Boeing 777	264	1765	300582
11	Jessica James	Silver	DL122	Airbus A330	236	4370	127656
12	Sam Rio	None	DL37	Boeing 747	400	531	2653
13	Christian Janco	Silver	DL222	Boeing 777	264	1765	14642
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Por último, hemos de ver si hay que crear tablas adicionales para cumplir 3NF o no es necesario, para eso, hay que ver las relaciones entre columnas. Veamos las relaciones entre numero de vuelo, modelo de avión y asientos.

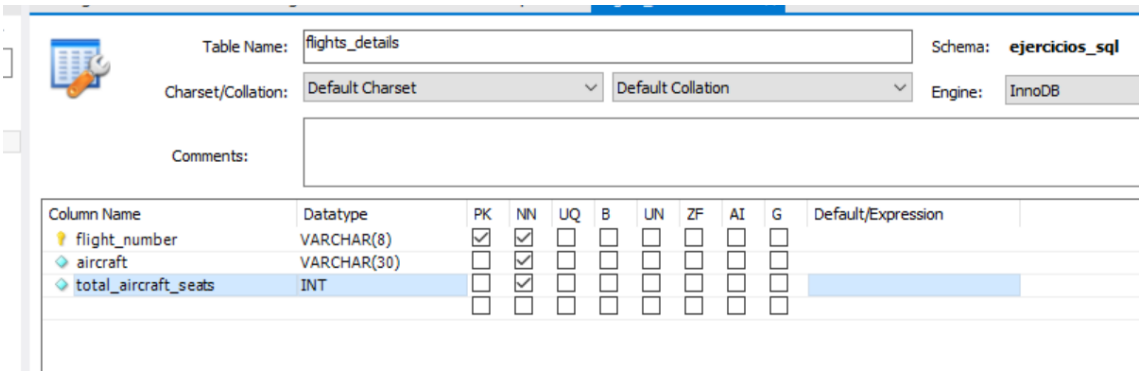
```
2
3 •   select flight_number, aircraft, total_aircraft_seats
4     from flights;
5
6
7
8
9
```

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

flight_number	aircraft	total_aircraft_seats
DL122	Airbus A330	236
DL122	Airbus A330	236
DL122	Airbus A330	236
DL53	Boeing 777	264
DL143	Boeing 747	400
DL222	Boeing 777	264
DL143	Boeing 747	400
DL143	Boeing 747	400
DL222	Boeing 777	264
DL222	Boeing 777	264
DL122	Airbus A330	236
DL37	Boeing 747	400
DL222	Boeing 777	264

Se aprecia que las 3 columnas están relacionadas, eso quiere decir que hay dependencia transitiva, por lo tanto, hay que crear una tabla adicional que contenga el id (que va a ser el flight number, y a su vez, clave primaria), aircraft y total_aircraft_seats.

Por otro lado, en la tabla inicial, eliminar las columnas aircraft y total_aurcraft_seats y mantener flight_number, ya que está va a ser la clave foránea. He de crear una tabla adicional y guardar los registros.



```
INSERT INTO flights_details (flight_number, aircraft, total_aircraft_seats)
VALUES
('DL143', 'Boeing 747', 400),
('DL122', 'Airbus A330', 236),
('DL53', 'Boeing 777', 264),
('DL222', 'Boeing 777', 264),
('DL37', 'Boeing 747', 400);
```

```
5
6  select * from flights_details;
7
8
9
```

Result Grid | Filter Rows: | Edit: | Export/Import:

flight_number	aircraft	total_aircraft_seats
DL122	Airbus A330	236
DL143	Boeing 747	400
DL222	Boeing 777	264
DL37	Boeing 747	400
DL53	Boeing 777	264
NULL	NULL	NULL

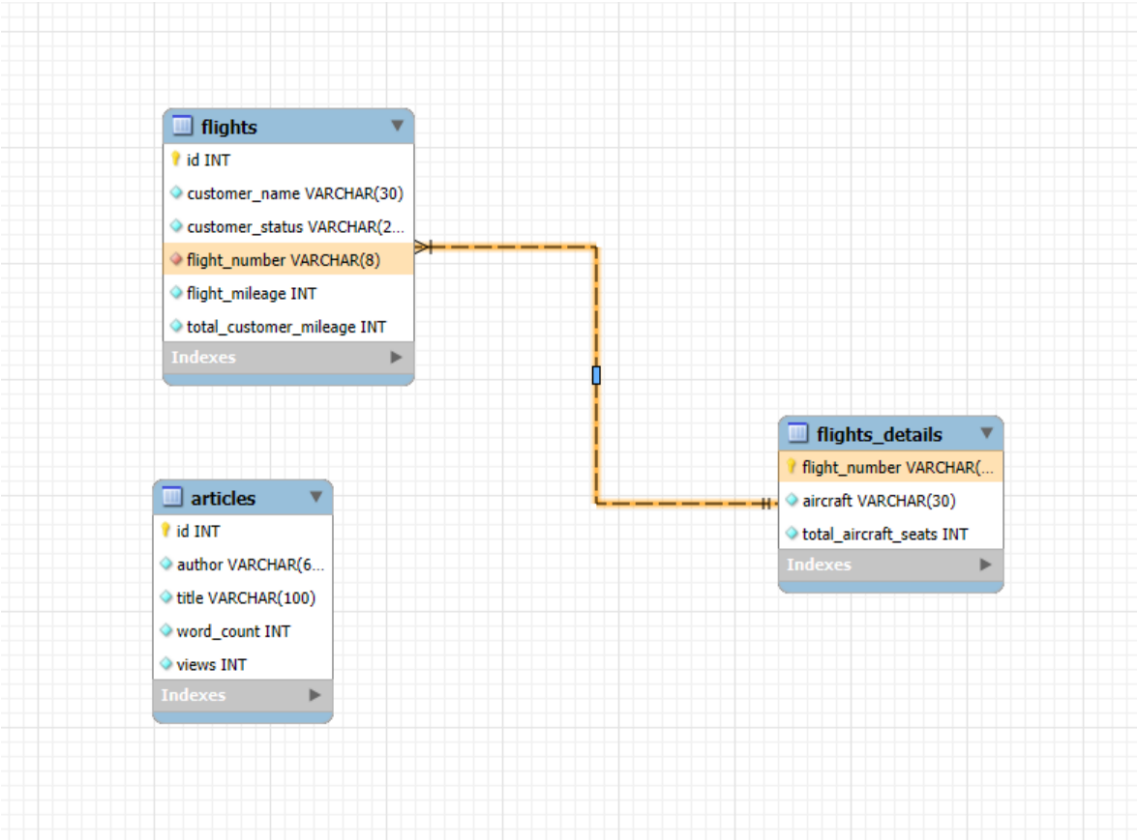
La tabla inicial queda así

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap

id	customer_name	customer_status	flight_number	flight_mileage	total_customer_mileage
1	Agustine Riviera	Silver	DL122	4370	115235
2	Alaina Sepulvida	None	DL122	4370	6008
3	Tom Jones	Gold	DL122	4370	205767
4	Tom Jones	Gold	DL53	2078	205767
5	Agustine Riviera	Silver	DL143	135	115235
6	Tom Jones	Gold	DL222	1765	205767
7	Jessica James	Silver	DL143	135	127656
8	Sam Rio	None	DL143	135	2653
9	Ana Janco	Silver	DL222	1765	136773
10	Jennifer Cortez	Gold	DL222	1765	300582
11	Jessica James	Silver	DL122	4370	127656
12	Sam Rio	None	DL37	531	2653
13	Christian Janco	Silver	DL222	1765	14642
NULL	NULL	NULL	NULL	NULL	NULL

Por último, hay que definir la clave foránea de la tabla flights (tabla inicial) que es el flight_number, que va a hacer referencia a la columna flight_number de flights_details.

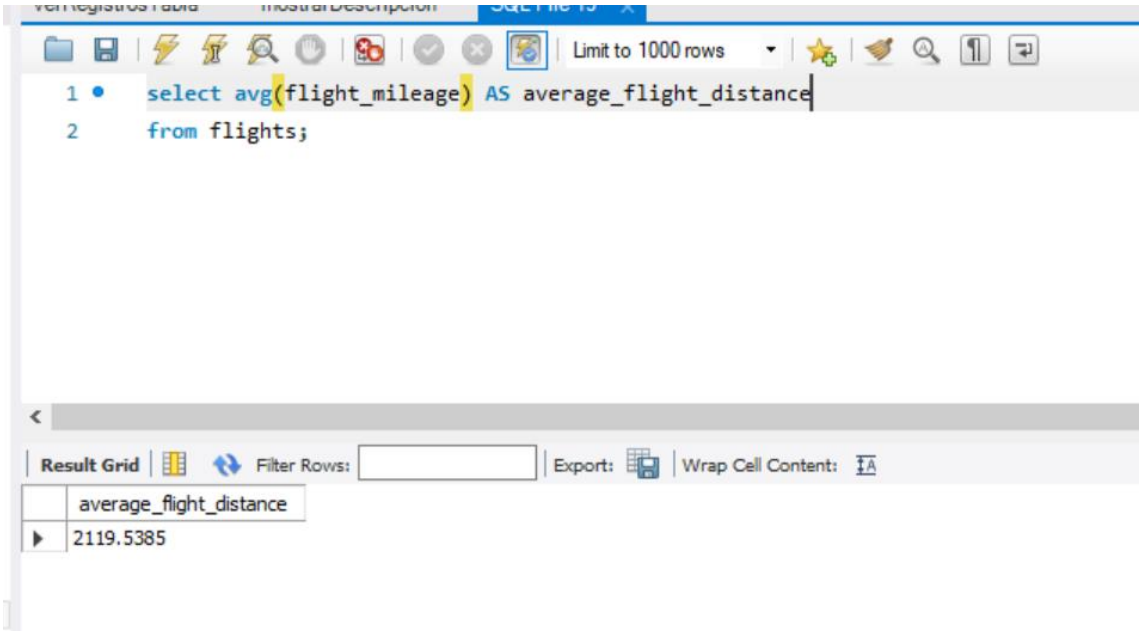
Con esto hemos conseguido establecer esta relación



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

```
1 •  select count(*)
2    from flights;
```

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



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

```
1 •  select avg(total_aircraft_seats) AS average_number_seats
2    from flights_details;
```

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

average_number_seats
312.8000

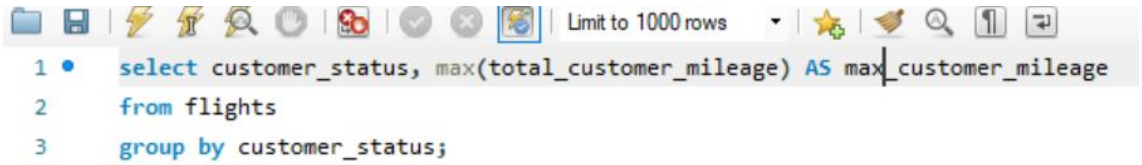
6. In the Airline database write the SQL script to get the average number of miles flown by customers grouped by status.

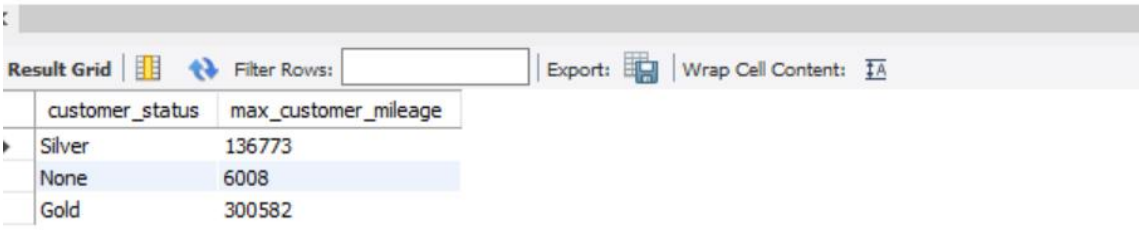
```
1 •  select customer_status, avg(total_customer_mileage) AS average_customer_mileage
2    from flights
3    group by customer_status;
```

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

customer_status	average_customer_mileage
Silver	106199.5000
None	3771.3333
Gold	229470.7500

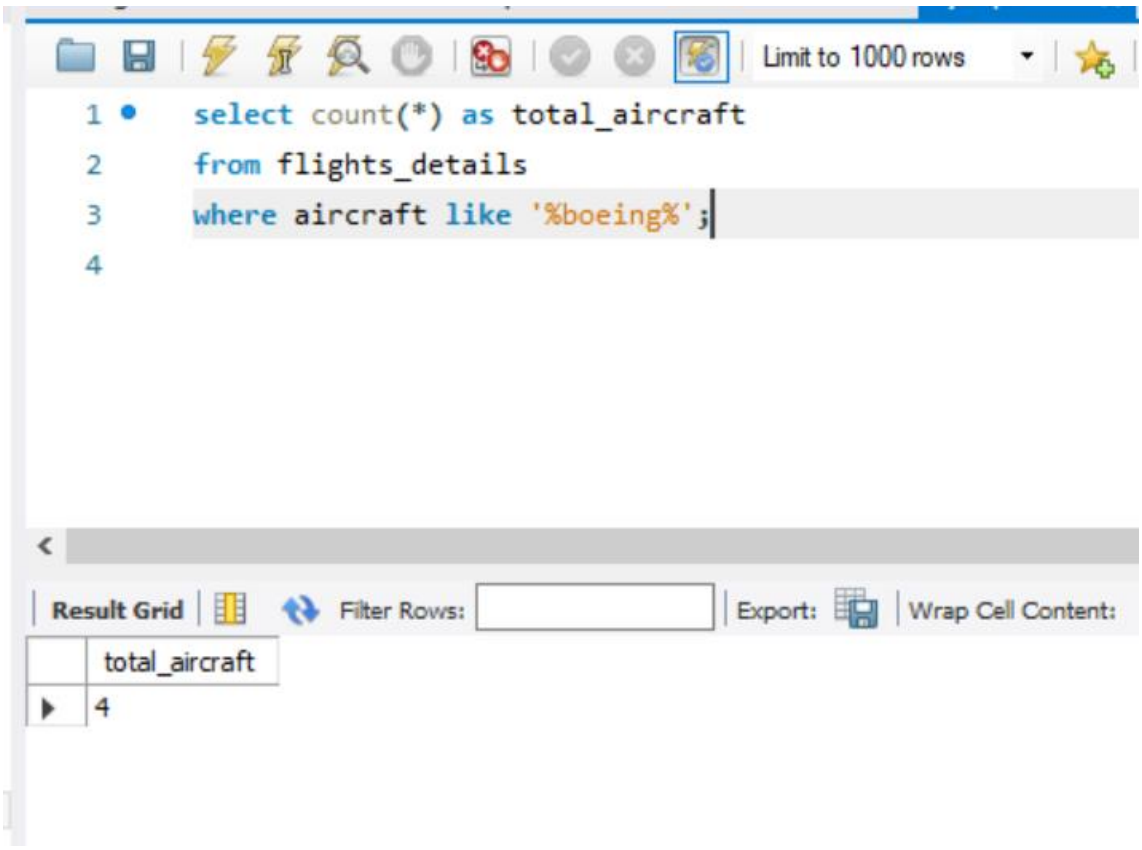
7. In the Airline database write the SQL script to get the maximum number of miles flown by customers grouped by status.





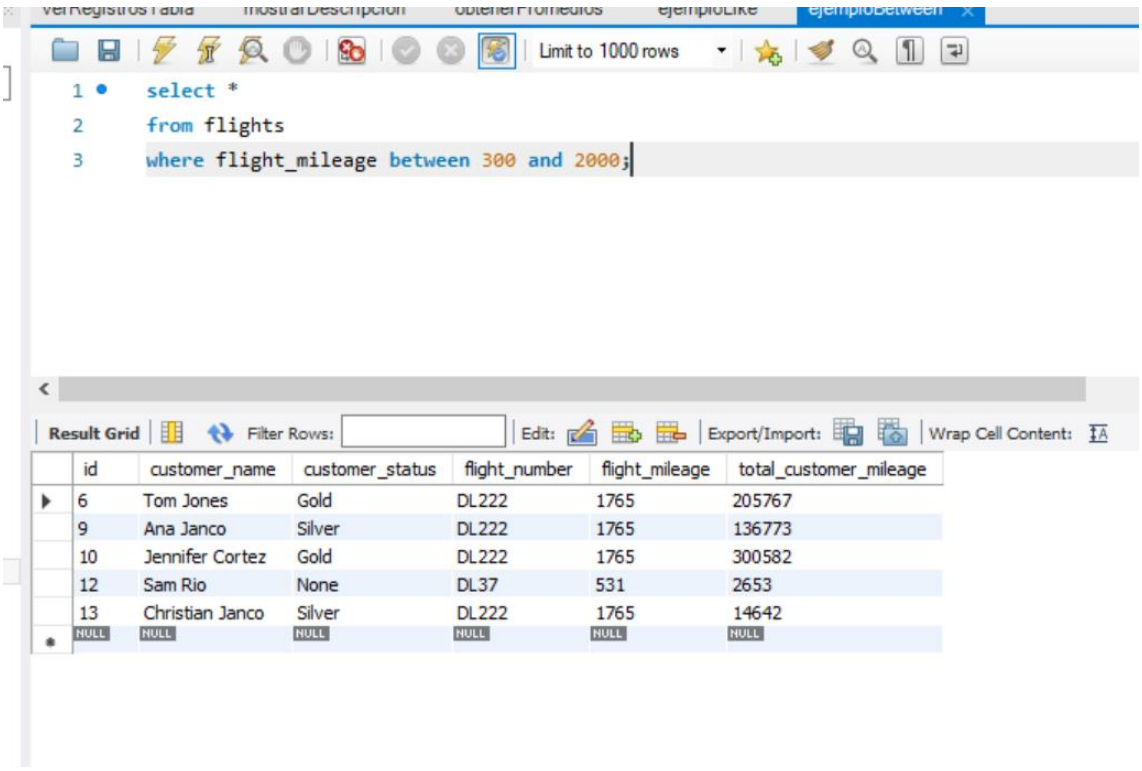
customer_status	max_customer_mileage
Silver	136773
None	6008
Gold	300582

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



total_aircraft
4

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

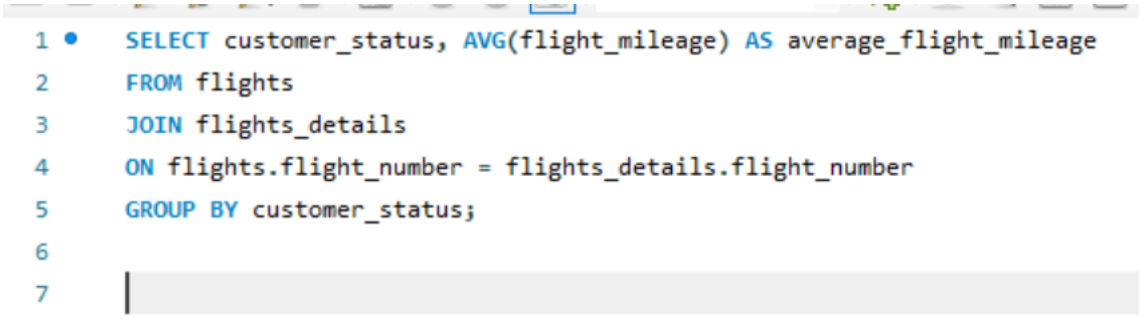


```
1  select *
2  from flights
3  where flight_mileage between 300 and 2000;
```

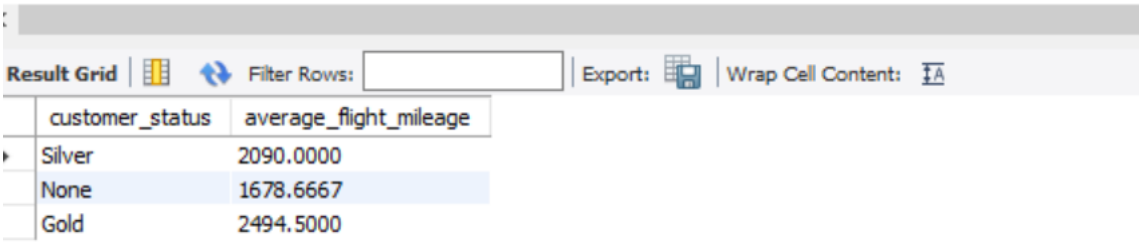
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	customer_name	customer_status	flight_number	flight_mileage	total_customer_mileage
6	Tom Jones	Gold	DL222	1765	205767
9	Ana Janco	Silver	DL222	1765	136773
10	Jennifer Cortez	Gold	DL222	1765	300582
12	Sam Rio	None	DL37	531	2653
13	Christian Janco	Silver	DL222	1765	14642
NULL	NULL	NULL	NULL	NULL	NULL

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



```
1  SELECT customer_status, AVG(flight_mileage) AS average_flight_mileage
2  FROM flights
3  JOIN flights_details
4  ON flights.flight_number = flights_details.flight_number
5  GROUP BY customer_status;
6
7
```



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

customer_status	average_flight_mileage
Silver	2090.0000
None	1678.6667
Gold	2494.5000

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

```
6
7   SELECT aircraft, COUNT(*) AS booking_count
8   FROM flights
9   JOIN flights_details
10  ON flights.flight_number = flights_details.flight_number
11  WHERE customer_status = 'Gold'
12  GROUP BY aircraft
13  ORDER BY booking_count DESC
14  LIMIT 1;
15
16
```

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

aircraft	booking_count
Boeing 777	3