

ADVANCED SQL REPORTS

1. BOOKINGS,
2. CATERINGS,
3. PAYMENTS,
4. FEEDBACK

1. --CHECKING THE INFO OF THE CUSTOMER WHO RECENTLY BOOKED A FLIGHT AND THE FIRST CUSTOMER IN THE DATABASE

```
SELECT *
FROM bookings
WHERE BOOKINGDATE = (SELECT MAX(BOOKINGDATE) FROM BOOKINGS) OR
BOOKINGDATE = (SELECT MIN(BOOKINGDATE) FROM BOOKINGS);
```

--CHECKING THE INFO OF THE CUSTOMER WHO RECENTLY BOOKED A FLIGHT AND THE FIRST CUSTOMER IN THE DATABASE

```
SELECT *
FROM bookings
WHERE BOOKINGDATE = (SELECT MAX(BOOKINGDATE) FROM BOOKINGS) OR BOOKINGDATE = (SELECT MIN(BOOKINGDATE) FROM BOOKINGS);
```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0.336 seconds

	BOOKINGID	CUSTOMERID	FLIGHTID	BOOKINGDATE	BOOKINGTIME	SEATNUMBER	BOOKINGSTATUS
1	901	401	701	24-01-01	1000	1	Confirmed
2	920	420	720	24-01-20	600	20	Confirmed

2. --CHECKING THE AVERAGE BOOKING SIZE OF THE WEBSITE ,LOWEST AND THE HIGHEST

```
SELECT AVG(PAYMENTAMOUNT) AS AVGBOOKING, MAX(PAYMENTAMOUNT) AS
HIGHESTBOOKING, MIN(PAYMENTAMOUNT) AS LOWESTBOOKING
FROM PAYMENTS;
```

--CHECKING THE AVERAGE BOOKING SIZE OF THE WEBSITE ,LOWEST AND THE HIGHEST

```
SELECT AVG(PAYMENTAMOUNT) AS AVGBOOKING, MAX(PAYMENTAMOUNT) AS HIGHESTBOOKING, MIN(PAYMENTAMOUNT) AS LOWESTBOOKING
FROM PAYMENTS;
```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0.108 seconds

	AVGBOOKING	HIGHESTBOOKING	LOWESTBOOKING
1	234	400	100

3. SHOWING TOTAL NUMBER OF CATERINGS PER CATERINGTYPE

```
SELECT cateringtype ,COUNT(CATERINGID) NOOFCATERING
FROM catering
GROUP BY cateringtype;
```

--SHOWING TOTAL NUMBER OF CATERINGS PER CATERINGTYPE

```
SELECT cateringtype ,COUNT(CATERINGID) NOOFCATERING
FROM catering
GROUP BY cateringtype;
```

Query Result x Query Result 1 x

SQL | All Rows Fetched: 3 in 0.117 seconds

CATERINGTYPE	NOOFCATERING
1 Meal	7
2 Snack	6
3 Beverage	7

4. --SHOW ALL THE INFO OF COMPANIES WHICH IS OWNED BY DRINKMASTER

```
SELECT *
FROM CATERING
WHERE UPPER(parentcompany) = 'DRINK MASTER'
ORDER BY cateringid;
```

--SHOW ALL THE INFO OF COMPANIES WHICH IS OWNED BY DRINKMASTER

```
SELECT *
FROM CATERING
WHERE UPPER(parentcompany) = 'DRINK MASTER'
ORDER BY cateringid;
```

Script Output x Query Result x

SQL | All Rows Fetched: 7 in 0.111 seconds

CATERINGID	FLIGHTID	CATERINGTYPE	YEARSOFEXPERINCE	CATERINGDETAILS	PARENTCOMPANY
1	802	702 Beverage		3 Standard beverage service for Economy class passengers	Drink Master
2	805	705 Beverage		2 Premium beverage service for Business class passengers	Drink Master
3	808	708 Beverage		5 Customized beverage service for First class passengers	Drink Master
4	811	711 Beverage		4 Standard beverage service for Economy class passengers	Drink Master
5	814	714 Beverage		5 Premium beverage service for First class passengers	Drink Master
6	817	717 Beverage		3 Standard beverage service for all passengers	Drink Master
7	820	720 Beverage		5 Premium beverage service for First class passengers	Drink Master

5. --SELECTING ALL FLIGHT SERVICE WHICH HAS A NEGATIVE FEEDBACK FROM CUSTOMERS

```
SELECT *
FROM feedback
WHERE UPPER(feedbacktype) = 'BAD'
```

ORDER BY FLIGHTID;

```
--SELECTING ALL FLIGHT SERVICE WHICH HAS A NEGATIVE FEEDBACK FROM CUSTOMERS
SELECT *
FROM feedback
WHERE UPPER(feedbacktype) = 'BAD'
ORDER BY FLIGHTID;
```

Query Result x

SQL | All Rows Fetched: 6 in 0.407 seconds

	FEEDBACKID	CUSTOMERID	FLIGHTID	CATERINGID	FEEDBACKTYPE	FEEDBACK_DETAILS	FEEDBACK_DATE
1	1112	402	702	802	BAD	Unfortunately, the flight was delayed, causing inconvenience.	24-04-06
2	1116	406	706	806	BAD	Flight delay made me miss my connecting flight.	24-04-06
3	1117	407	707	807	BAD	Unfortunately, my flight got cancelled last minute.	24-04-06
4	1121	411	711	811	BAD	Flight cancellation without prior notice.	24-04-06
5	1125	415	715	815	BAD	Flight cancellation caused inconvenience.	24-04-06
6	1128	418	718	818	BAD	Last-minute cancellation was disappointing.	24-04-06

6. --LIST OF CUSTOMERS WHO HAVE PAID MORE THAN AVERAGE FOR A BOOKING
 SELECT CUSTOMERID
 FROM payments
 WHERE paymentamount > (SELECT AVG(PAYMENTAMOUNT) FROM PAYMENTS);

```
--LIST OF CUSTOMERS WHO HAVE PAID MORE THAN AVERAGE FOR A BOOKING
SELECT CUSTOMERID
FROM payments
WHERE paymentamount > (SELECT AVG(PAYMENTAMOUNT) FROM PAYMENTS );
```

Query Result x

SQL | All Rows Fetched: 9 in 0.11 seconds

	CUSTOMERID
1	404
2	408
3	410
4	412
5	414
6	416
7	418
8	419
9	420

7. SELECT FEEDBACKID
 FROM feedback
 WHERE feedbackid > (SELECT FEEDBACKID FROM FEEDBACK WHERE FLIGHTID
 =706)
 GROUP BY FEEDBACKID;

--DISPLAY ALL THE FEEDBACKID WHO ARE GREATER THAN THAT OF FLIGHT 706

```

SELECT FEEDBACKID
FROM feedback
WHERE feedbackid > (SELECT FEEDBACKID FROM FEEDBACK WHERE FLIGHTID =706)
GROUP BY FEEDBACKID;

```

Query Result x

SQL | All Rows Fetched: 14 in 0.119 seconds

	FEEDBACKID
1	1117
2	1118
3	1119
4	1120
5	1121
6	1122
7	1123
8	1124
9	1125
10	1126
11	1127
12	1128
13	1129
14	1130

8. --DISPLAYING ALL THE CUSTOMER NO IN BOOKING AND PAYMENTS WITHOUT DUPLICATE .

```

SELECT CUSTOMERID
FROM BOOKINGS
UNION
SELECT CUSTOMERID
FROM PAYMENTS;

```

```
--DISPLAYING ALL THE CUSTOMER NO IN BOOKING AND PAYMENTS WITHOUT DUPLICATE  
SELECT CUSTOMERID  
FROM BOOKINGS  
UNION  
SELECT CUSTOMERID  
FROM PAYMENTS;
```

Query Result x

SQL | All Rows Fetched: 20 in 0.115 seconds

	CUSTOMERID
1	401
2	402
3	403
4	404
5	405
6	406
7	407
8	408
9	409
10	410
11	411
12	412
13	413
14	414
15	415
16	416
17	417
18	418
19	419
20	420

1. FLIGHTS

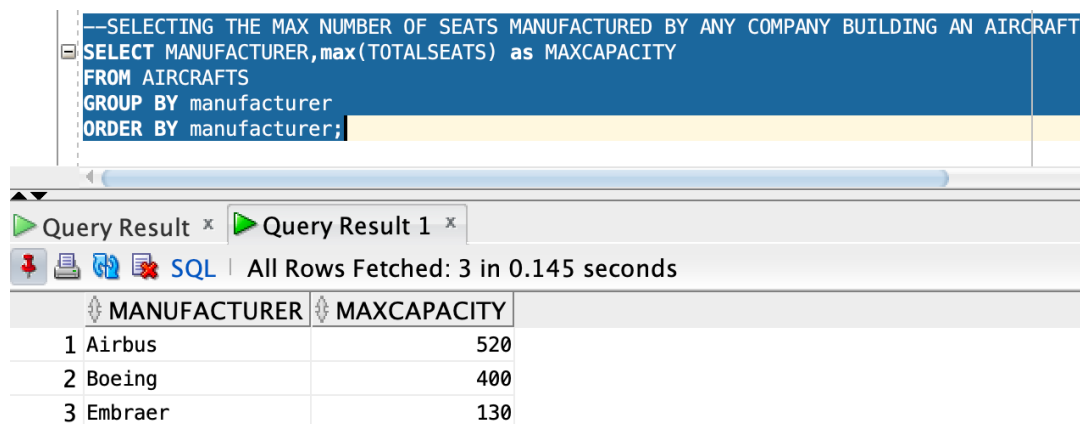
2. AIRCRAFTS

3. AIRPORT

4. FREQFLYER

1. SELECTING THE MAX NUMBER OF SEATS MANUFACTURED BY ANY COMPANY BUILDING AN AIRCRAFT

```
SELECT MANUFACTURER, max(TOTALSEATS) as MAXCAPACITY
FROM AIRCRAFTS
GROUP BY manufacturer
ORDER BY manufacturer;
```



The screenshot shows a SQL query execution window. The query is:


```
--SELECTING THE MAX NUMBER OF SEATS MANUFACTURED BY ANY COMPANY BUILDING AN AIRCRAFT
SELECT MANUFACTURER, max(TOTALSEATS) as MAXCAPACITY
FROM AIRCRAFTS
GROUP BY manufacturer
ORDER BY manufacturer;
```

 The results are displayed in a table with two columns: MANUFACTURER and MAXCAPACITY. The results are ordered by manufacturer.

	MANUFACTURER	MAXCAPACITY
1	Airbus	520
2	Boeing	400
3	Embraer	130

2. DISPLAY ALL THE DETAILS FOR MANUFACTURED AIRCRAFT WHO ARE MANUFACTURED BEFORE Airbus A350 AND SHOWN THEM IN DECENDING ORDER OF MANUFACTURE

```
SELECT *
FROM aircrafts
WHERE YEAROFMANUFACTURE < (SELECT YEAROFMANUFACTURE FROM
AIRCRAFTS WHERE aircrafttype ='Airbus A350')
ORDER BY aircrafts.yearofmanufacture DESC;
```

--DISPLAY ALL THE DETAILS FOR MANUFACTURED AIRCRAFT WHO ARE MANUFACTURED BEFORE Airbus A350 AND SHOWN THEM IN DECENDING ORDER OF MANUFACTURE

```

SELECT *
FROM aircrafts
WHERE YEAROFMANUFACTURE < (SELECT YEAROFMANUFACTURE FROM AIRCRAFTS WHERE aircrafttype ='Airbus A350')
ORDER BY aircrafts.yearofmanufacture DESC;

```

Query Result x Query Result 1 x Query Result 2 x

SQL | All Rows Fetched: 13 in 0.118 seconds

	AIRCRAFTID	COMPANYID	AIRCRAFTTYPE	MANUFACTURER	MODELNO	TOTALSEATS	YEAROFMANUFACTURE	LASTMAINTAINCE
1	603	303	Boeing 777	Boeing	777-300ER	350	2018 23-03-10	
2	606	306	Embraer E190	Embraer	E190-E2	110	2017 23-08-12	
3	605	305	Boeing 787	Boeing	787-9	250	2016 23-06-30	
4	602	302	Airbus A320	Airbus	A320neo	150	2015 23-01-20	
5	607	307	Boeing 747	Boeing	747-8	400	2014 23-09-05	
6	608	308	Airbus A380	Airbus	A380-800	500	2013 23-11-18	
7	612	312	Airbus A321	Airbus	A321neo	200	2012 23-03-15	
8	614	314	Boeing 767	Boeing	767-300ER	290	2011 23-06-10	
9	619	319	Boeing 737	Boeing	737-700	150	2010 23-12-05	
10	601	301	Boeing 737	Boeing	737-800	180	2010 23-02-15	
11	611	311	Boeing 757	Boeing	757-200	220	2008 23-02-28	
12	616	316	Boeing 777	Boeing	777-200LR	360	2007 23-09-15	
13	615	315	Airbus A340	Airbus	A340-600	320	2006 23-08-30	

3. DISPLAY ALL THE FLIGHTIDS WHOSE ARRIVAL TERMINAL IS GREATER THAN OR EQUAL TO THAT OF FLIGHT 703

```

SELECT FLIGHTID
FROM FLIGHTS
WHERE FLIGHTID >= (SELECT FLIGHTID FROM FLIGHTS WHERE FLIGHTID=703)
ORDER BY FLIGHTID;

```

--DISPLAY ALL THE FLIGHTIDS WHOSE ARRIVAL TERMINAL IS GREATER THAN OR EQUAL TO THAT OF FLIGHT 703

```

SELECT FLIGHTID
FROM FLIGHTS
WHERE FLIGHTID >= (SELECT FLIGHTID FROM FLIGHTS WHERE FLIGHTID=703)
ORDER BY FLIGHTID;

```

Query Result x Query Result 1 x Query Result 2 x

SQL | All Rows Fetched: 18 in 0.116 seconds

	FLIGHTID
1	703
2	704
3	705
4	706
5	707
6	708
7	709
8	710
9	711
10	712
11	713
12	714
13	715
14	716
15	717
16	718
17	719
18	720

4. DISPLAY ALL THE FLIGHTS WHO HAVE ARRIVED BEFORE FLIGHT 709 IN A ASCENDING ORDER
- ```
SELECT *
FROM FLIGHTS
WHERE arrivaltime < (SELECT ARRIVALTIME FROM FLIGHTS WHERE FLIGHTID = 709)
ORDER BY ARRIVALTIME;
```

--DISPLAY ALL THE FLIGHTS WHO HAVE ARRIVED BEFORE FLIGHT 709 IN A ASCENDING ORDER

```
SELECT *
FROM FLIGHTS
WHERE arrivaltime < (SELECT ARRIVALTIME FROM FLIGHTS WHERE FLIGHTID = 709)
ORDER BY ARRIVALTIME;
```

Query Result x Query Result 1 x Query Result 2 x Query Result 3 x

SQL All Rows Fetched: 14 in 0.241 seconds

|    | FLIGHTID | AIRCRAFTID | PILOTID | DEPARTUREAIRPORTID | ARRIVALAIRPORTID | PORTID | CREWID | COMPANYID | ARRIVALTERMINAL | DEPARTURETERMINAL | GATE | ARRIVAL... | DEPA... | FLIGHTSTA |
|----|----------|------------|---------|--------------------|------------------|--------|--------|-----------|-----------------|-------------------|------|------------|---------|-----------|
| 1  | 711      | 611        | 511     | 211                | 212              | 111    | 811    | 311       | 1               | 2                 | 13   | 1          | 20      | On Time   |
| 2  | 704      | 604        | 504     | 204                | 205              | 104    | 804    | 304       | 4               | 5                 | 6    | 2          | 20      | Cancelled |
| 3  | 719      | 619        | 519     | 219                | 220              | 119    | 819    | 319       | 4               | 5                 | 21   | 3          | 22      | On Time   |
| 4  | 712      | 612        | 512     | 212                | 213              | 112    | 812    | 312       | 2               | 3                 | 14   | 4          | 23      | Delayed   |
| 5  | 705      | 605        | 505     | 205                | 206              | 105    | 805    | 305       | 5               | 6                 | 7    | 5          | 1       | On Time   |
| 6  | 720      | 620        | 520     | 220                | 201              | 120    | 820    | 320       | 5               | 6                 | 22   | 6          | 1       | On Time   |
| 7  | 713      | 613        | 513     | 213                | 214              | 113    | 813    | 313       | 3               | 4                 | 15   | 7          | 3       | On Time   |
| 8  | 706      | 606        | 506     | 206                | 207              | 106    | 806    | 306       | 1               | 2                 | 8    | 9          | 5       | Delayed   |
| 9  | 714      | 614        | 514     | 214                | 215              | 114    | 814    | 314       | 4               | 5                 | 16   | 10         | 6       | On Time   |
| 10 | 707      | 607        | 507     | 207                | 208              | 107    | 807    | 307       | 2               | 3                 | 9    | 12         | 7       | On Time   |
| 11 | 715      | 615        | 515     | 215                | 216              | 115    | 815    | 315       | 5               | 6                 | 17   | 14         | 9       | Delayed   |
| 12 | 701      | 601        | 501     | 201                | 202              | 101    | 801    | 301       | 1               | 2                 | 3    | 15         | 10      | On Time   |
| 13 | 708      | 608        | 508     | 208                | 209              | 108    | 808    | 308       | 3               | 4                 | 10   | 15         | 10      | On Time   |
| 14 | 716      | 616        | 516     | 216                | 217              | 116    | 816    | 316       | 1               | 2                 | 18   | 17         | 12      | On Time   |

5. SELECTING COUNTRY HAVING MORE THAN ONE AIRPORT
- ```
SELECT COUNTRY,COUNT(airportid) AS TOTALNUMBEROFAIRPORTS
FROM AIRPORT
GROUP BY country
HAVING COUNT(AIRPORTID) > 1;
```

--SELECTING COUNTRY HAVING MORE THAN ONE AIRPORT

```
SELECT COUNTRY,COUNT(airportid) AS TOTALNUMBEROFAIRPORTS
FROM AIRPORT
GROUP BY country
HAVING COUNT(AIRPORTID) > 1;
```

Query Result x Query Result 1 x Query Result 2 x Query Result 3 x

SQL All Rows Fetched: 1 in 0.105 seconds

	COUNTRY	TOTALNUMBEROFAIRPORTS
1	United States	6

6. SELECTING AVERAGE POINTS IN EACH CATEGORY IN FREQUENCY FLYER
- ```
SELECT STATUS,ROUND(AVG(POINTS),2) AS AVERAGEPOINTS
FROM freqflyer
GROUP BY STATUS;
```



```
--SELECTING AVERAGE POINTS IN EACH CATEGORY IN FREQUENCY FLYER
SELECT STATUS,ROUND(AVG(POINTS),2) AS AVERAGEPOINTS
FROM freqflyer
GROUP BY STATUS;
```

Query Result x | Query Result 1 x | Query Result 2 x | Query R

SQL | All Rows Fetched: 3 in 0.105 seconds

|   | STATUS  | AVERAGEPOINTS |
|---|---------|---------------|
| 1 | Silver  | 2790          |
| 2 | Regular | 1500          |
| 3 | Gold    | 4514.29       |

7. SELECTING THE CUSTOMER WHO HAVE POINTS MORE THAN AVG OF THE WHOLE  
 SELECT CUSTOMERID , POINTS,STATUS  
 FROM freqflyer  
 WHERE points > (SELECT AVG(POINTS) FROM FREQFLYER )  
 ORDER BY CUSTOMERID;

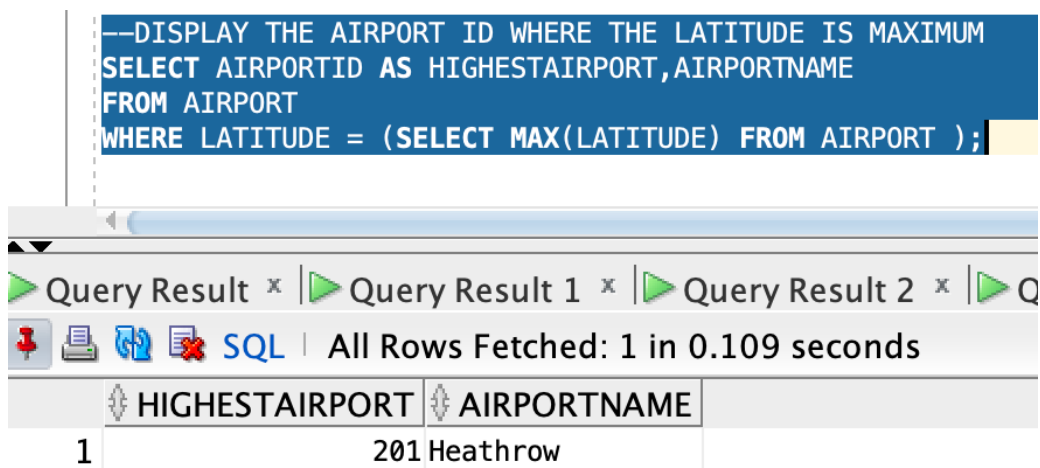
```
--SELECTING THE CUSTOMER WHO HAVE POINTS MORE THAN AVG OF THE WHOLE
SELECT CUSTOMERID , POINTS,STATUS
FROM freqflyer
WHERE points > (SELECT AVG(POINTS) FROM FREQFLYER)
ORDER BY CUSTOMERID;
```

Query Result x | Query Result 1 x | Query Result 2 x | Query Resu

SQL | All Rows Fetched: 9 in 0.11 seconds

|   | CUSTOMERID | POINTS | STATUS |
|---|------------|--------|--------|
| 1 | 401        | 5000   | Gold   |
| 2 | 404        | 4500   | Gold   |
| 3 | 405        | 3500   | Silver |
| 4 | 407        | 4000   | Gold   |
| 5 | 410        | 5000   | Gold   |
| 6 | 413        | 4100   | Gold   |
| 7 | 416        | 4800   | Gold   |
| 8 | 417        | 3800   | Silver |
| 9 | 419        | 4200   | Gold   |

8. DISPLAY THE AIRPORT ID WHERE THE LATITUDE IS MAXIMUM  
SELECT AIRPORTID AS HIGHESTAIRPORT,AIRPORTNAME  
FROM AIRPORT  
WHERE LATITUDE = (SELECT MAX(LATITUDE) FROM AIRPORT );



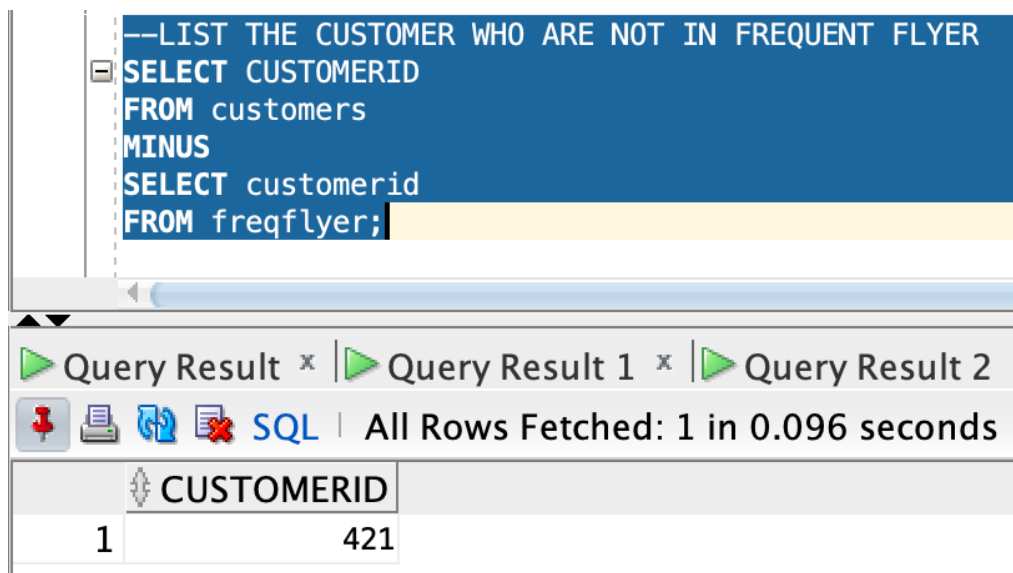
```
--DISPLAY THE AIRPORT ID WHERE THE LATITUDE IS MAXIMUM
SELECT AIRPORTID AS HIGHESTAIRPORT,AIRPORTNAME
FROM AIRPORT
WHERE LATITUDE = (SELECT MAX(LATITUDE) FROM AIRPORT);
```

Query Result x | Query Result 1 x | Query Result 2 x | Q

SQL | All Rows Fetched: 1 in 0.109 seconds

|   | HIGHESTAIRPORT | AIRPORTNAME |
|---|----------------|-------------|
| 1 | 201            | Heathrow    |

9. LIST THE CUSTOMER WHO ARE NOT IN FREQUENT FLYER  
SELECT CUSTOMERID  
FROM customers  
MINUS  
SELECT customerid  
FROM freqflyer;



```
--LIST THE CUSTOMER WHO ARE NOT IN FREQUENT FLYER
SELECT CUSTOMERID
FROM customers
MINUS
SELECT customerid
FROM freqflyer;
```

Query Result x | Query Result 1 x | Query Result 2 x

SQL | All Rows Fetched: 1 in 0.096 seconds

|   | CUSTOMERID |
|---|------------|
| 1 | 421        |

10. DISPLAY ALL THE CUSTOMERS AND THEIR POINTS COLLECTED IF THE CUSTOMER DOESN'T HAVE ANY POINT DISPLAY 0  
 SELECT CUSTOMERID ,POINTS AS POINTS  
 FROM freqflyer  
 UNION  
 SELECT CUSTOMERID , 0 AS POINTS  
 FROM CUSTOMERS  
 ORDER BY CUSTOMERID;

--DISPLAY ALL THE CUSTOMERS AND THEIR POINTS COLLECTED IF THE CUSTOMER DOESN'T HAVE ANY POINT DISPLAY 0

```

SELECT CUSTOMERID ,POINTS AS POINTS
FROM freqflyer
UNION
SELECT CUSTOMERID , 0 AS POINTS
FROM CUSTOMERS
ORDER BY CUSTOMERID;

```

Query Result x | Query Result 1 x | Query Result 2 x | Query Result 3 x | Query Result 4 x | ... x |

SQL All Rows Fetched: 41 in 0.109 seconds

|    | CUSTOMERID | POINTS |
|----|------------|--------|
| 21 | 411        | 0      |
| 22 | 411        | 3200   |
| 23 | 412        | 0      |
| 24 | 412        | 1800   |
| 25 | 413        | 0      |
| 26 | 413        | 4100   |
| 27 | 414        | 0      |
| 28 | 414        | 2500   |
| 29 | 415        | 0      |
| 30 | 415        | 1900   |
| 31 | 416        | 0      |
| 32 | 416        | 4800   |
| 33 | 417        | 0      |
| 34 | 417        | 3800   |
| 35 | 418        | 0      |
| 36 | 418        | 1200   |
| 37 | 419        | 0      |
| 38 | 419        | 4200   |
| 39 | 420        | 0      |
| 40 | 420        | 3000   |
| 41 | 421        | 0      |

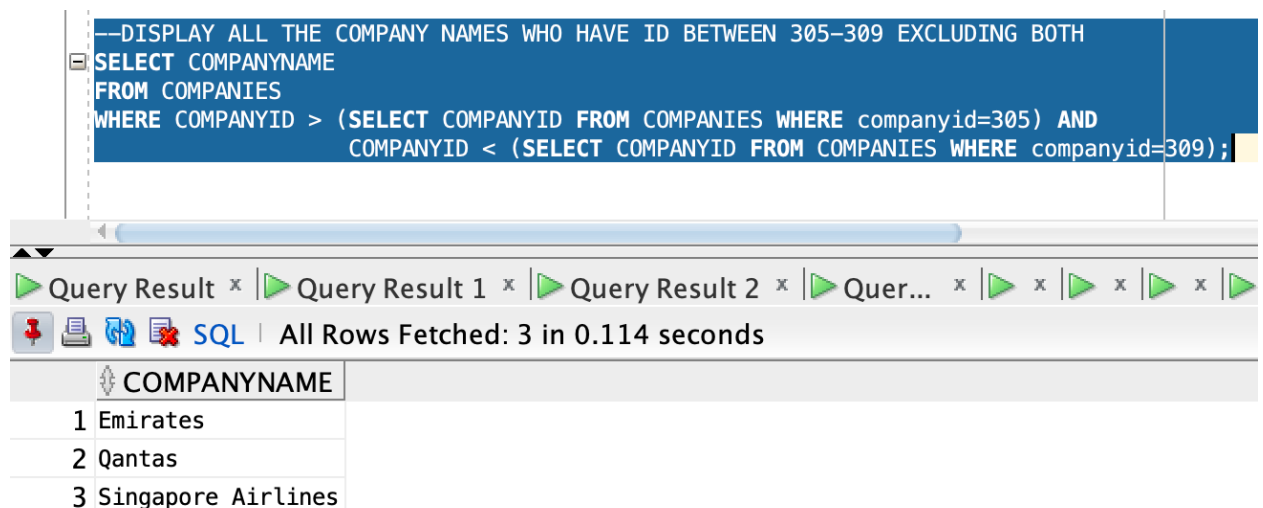
1. COMPANY
2. PILOTS
3. PORTS
4. CUSTOMER
5. CREW

1. DISPLAY ALL THE COMPANY NAMES WHO HAVE ID BETWEEN 305-309 EXCLUDING BOTH

```
SELECT COMPANYNAME
FROM COMPANIES
```

```
WHERE COMPANYID > (SELECT COMPANYID FROM COMPANIES WHERE
companyid=305) AND
```

```
COMPANYID < (SELECT COMPANYID FROM COMPANIES WHERE
companyid=309);
```



The screenshot shows a SQL query execution window. The query is:   
 --DISPLAY ALL THE COMPANY NAMES WHO HAVE ID BETWEEN 305-309 EXCLUDING BOTH  
 SELECT COMPANYNAME  
 FROM COMPANIES  
 WHERE COMPANYID > (SELECT COMPANYID FROM COMPANIES WHERE companyid=305) AND  
 COMPANYID < (SELECT COMPANYID FROM COMPANIES WHERE companyid=309);

The results are displayed in a table with the following data:

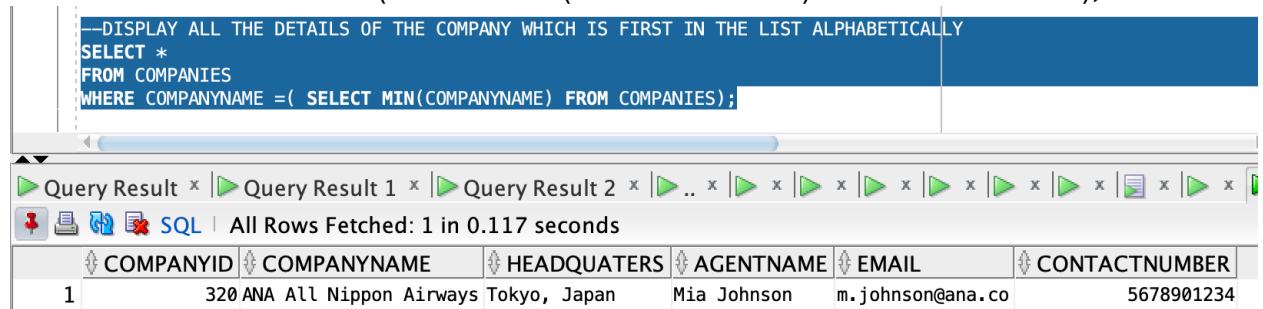
|   | COMPANYNAME        |
|---|--------------------|
| 1 | Emirates           |
| 2 | Qantas             |
| 3 | Singapore Airlines |

2. DISPLAY ALL THE DETAILS OF THE COMPANY WHICH IS FIRST IN THE LIST ALPHABETICALLY

```
SELECT *
```

```
FROM COMPANIES
```

```
WHERE COMPANYNAME =(SELECT MIN(COMPANYNAME) FROM COMPANIES);
```

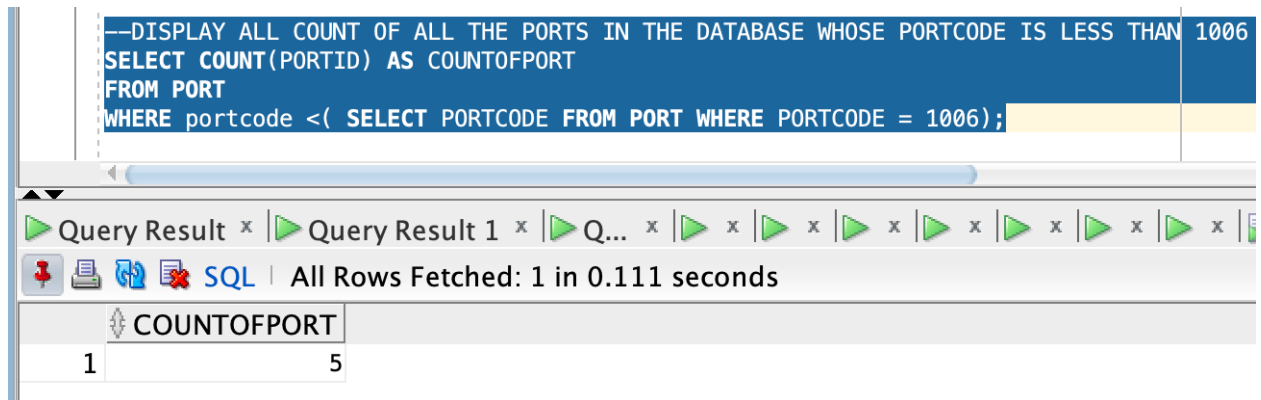


The screenshot shows a SQL query execution window. The query is:   
 --DISPLAY ALL THE DETAILS OF THE COMPANY WHICH IS FIRST IN THE LIST ALPHABETICALLY  
 SELECT \*  
 FROM COMPANIES  
 WHERE COMPANYNAME =( SELECT MIN(COMPANYNAME) FROM COMPANIES);

The results are displayed in a table with the following data:

|   | COMPANYID | COMPANYNAME            | HEADQUATERS  | AGENTNAME   | EMAIL            | CONTACTNUMBER |
|---|-----------|------------------------|--------------|-------------|------------------|---------------|
| 1 | 320       | ANA All Nippon Airways | Tokyo, Japan | Mia Johnson | m.johnson@ana.co | 5678901234    |

3. DISPLAY ALL COUNT OF ALL THE PORTS IN THE DATABASE WHOSE PORTCODE IS LESS THAN 1006  
 SELECT COUNT(PORTID) AS COUNTOFPORT  
 FROM PORT  
 WHERE portcode <( SELECT PORTCODE FROM PORT WHERE PORTCODE = 1006);



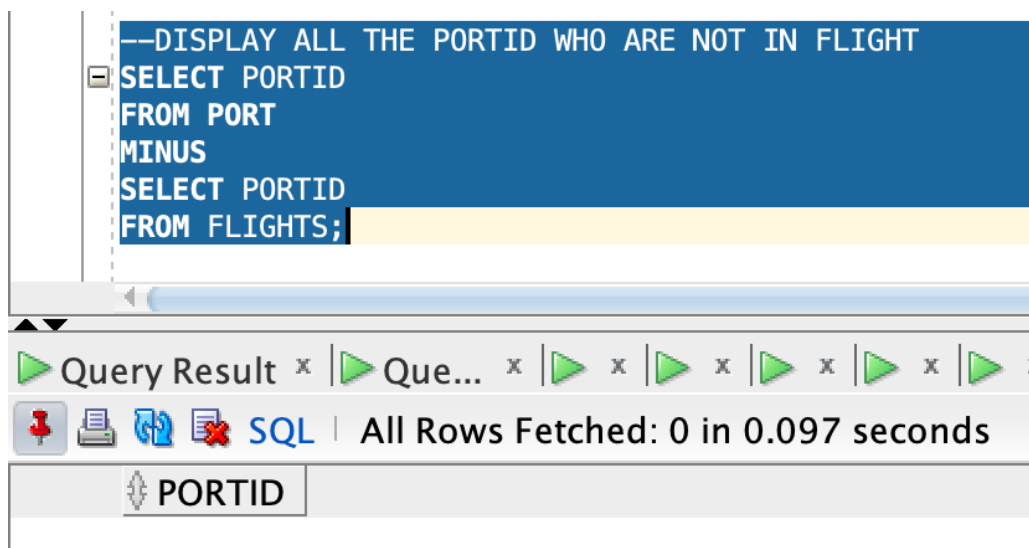
```
--DISPLAY ALL COUNT OF ALL THE PORTS IN THE DATABASE WHOSE PORTCODE IS LESS THAN 1006
SELECT COUNT(PORTID) AS COUNTOFPORT
FROM PORT
WHERE portcode <(SELECT PORTCODE FROM PORT WHERE PORTCODE = 1006);
```

Query Result x Query Result 1 x Q... x x x x x x x x x x x x x

SQL | All Rows Fetched: 1 in 0.111 seconds

| COUNTOFPORT |
|-------------|
| 5           |

4. DISPLAY ALL THE PORTID WHO ARE NOT IN FLIGHT  
 SELECT PORTID  
 FROM PORT  
 MINUS  
 SELECT PORTID  
 FROM FLIGHTS;



```
--DISPLAY ALL THE PORTID WHO ARE NOT IN FLIGHT
SELECT PORTID
FROM PORT
MINUS
SELECT PORTID
FROM FLIGHTS;
```

Query Result x Que... x x x x x x x x x

SQL | All Rows Fetched: 0 in 0.097 seconds

| PORTID |
|--------|
|--------|

\*\*there are no ports which are not in the flights table, all the ports are used, so no inactive port.

5. DISPLAYING THE AVERAGE NUMBER OF FLIGHTS FOR A PILOT  
SELECT AVG(TOTALFLIGHTS) AS AVGFLYINGHOURS  
FROM PILOTS;

```
--DISPLAYING THE AVERAGE NUMBER OF FLIGHTS FOR A PILOT
SELECT AVG(TOTALFLIGHTS) AS AVGFLYINGHOURS
FROM PILOTS;
```

Query Result x | x | x | x | x | x | x | x | x

SQL | All Rows Fetched: 1 in 0.105 seconds

|   | AVGFLYINGHOURS |
|---|----------------|
| 1 | 1450           |

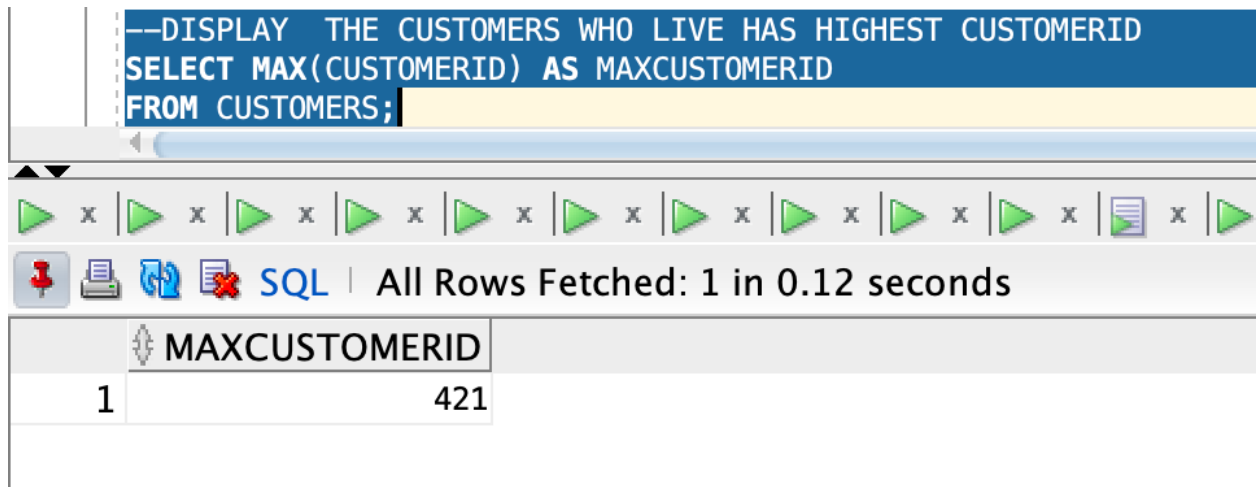
- ```
6. SELECT THE PILOTS WHOSE TOTAL FLIGHT FALL UNDER AVERAGE OF THE WHOLE
SELECT PILOTID
FROM PILOTS
WHERE TOTALFLIGHTS < (SELECT AVG(TOTALFLIGHTS) FROM PILOTS);
```

```
--SELECT THE PILOTS WHOSE TOTAL FLIGHT FALL UNDER AVERAGE OF THE WHOLE
SELECT PILOTID
FROM PILOTS
WHERE TOTALFLIGHTS < (SELECT AVG(TOTALFLIGHTS) FROM PILOTS);
```

SQL | All Rows Fetched: 10 in 0.174 seconds

	PILOTID
1	501
2	502
3	503
4	504
5	505
6	506
7	507
8	508
9	509
10	510

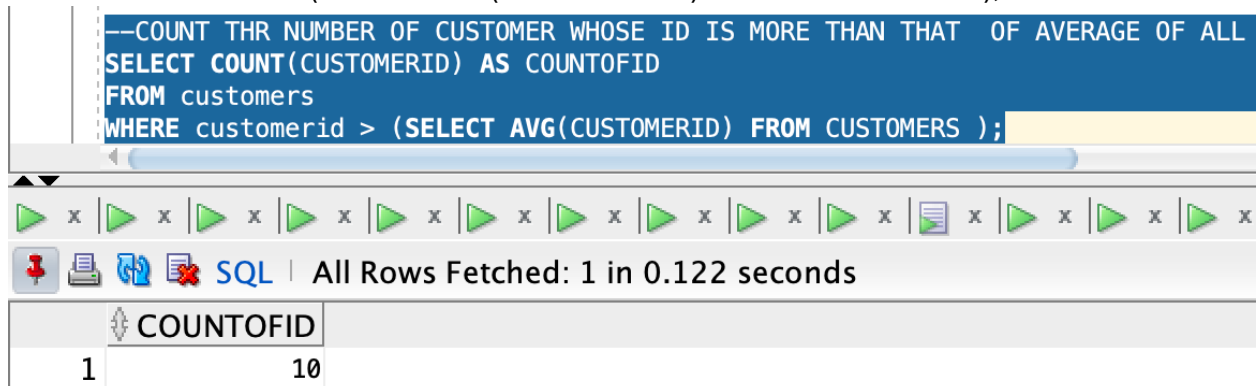
7. DISPLAY THE CUSTOMERS WHO LIVE HAS HIGHEST CUSTOMERID
 SELECT MAX(CUSTOMERID) AS MAXCUSTOMERID
 FROM CUSTOMERS;



The screenshot shows a SQL query editor with the following text: `--DISPLAY THE CUSTOMERS WHO LIVE HAS HIGHEST CUSTOMERID`, `SELECT MAX(CUSTOMERID) AS MAXCUSTOMERID`, and `FROM CUSTOMERS;`. Below the editor is a toolbar with various icons. A status bar indicates "All Rows Fetched: 1 in 0.12 seconds". The result is displayed in a table with one column, `MAXCUSTOMERID`, and one row with the value 421.

	MAXCUSTOMERID
1	421

8. COUNT THR NUMBER OF CUSTOMER WHOSE ID IS MORE THAN THAT OF AVERAGE OF ALL
 SELECT COUNT(CUSTOMERID) AS COUNTOFID
 FROM customers
 WHERE customerid > (SELECT AVG(CUSTOMERID) FROM CUSTOMERS);



The screenshot shows a SQL query editor with the following text: `--COUNT THR NUMBER OF CUSTOMER WHOSE ID IS MORE THAN THAT OF AVERAGE OF ALL`, `SELECT COUNT(CUSTOMERID) AS COUNTOFID`, `FROM customers`, and `WHERE customerid > (SELECT AVG(CUSTOMERID) FROM CUSTOMERS);`. Below the editor is a toolbar with various icons. A status bar indicates "All Rows Fetched: 1 in 0.122 seconds". The result is displayed in a table with one column, `COUNTOFID`, and one row with the value 10.

	COUNTOFID
1	10

9. SELECT FLIGHT DETAILS WHICH WAS OPERATED BY PILOT ID 512,509 AND 515
 SELECT *
 FROM FLIGHTS
 WHERE PILOTID = (SELECT PILOTID FROM PILOTS WHERE PILOTID =512)
 OR PILOTID = (SELECT PILOTID FROM PILOTS WHERE PILOTID =509)
 OR PILOTID = (SELECT PILOTID FROM PILOTS WHERE PILOTID =515);

```

--SELECT FLIGHT DETAILS WHICH WAS OPERATED BY PILOT ID 512,509 AND 515
SELECT *
FROM FLIGHTS
WHERE PILOTID = (SELECT PILOTID FROM PILOTS WHERE PILOTID =512)
OR PILOTID = (SELECT PILOTID FROM PILOTS WHERE PILOTID =509)
OR PILOTID = (SELECT PILOTID FROM PILOTS WHERE PILOTID =515);

```

Query Result x Query Result 1 x Query Result 2 x Query... x | All Rows Fetched: 3 in 0.136 seconds

	FLIGHTID	AIRCRAFTID	PILOTID	DEPARTUR...	ARRIVAL...	PORTID	CREWID	COMPANYID	ARRIVAL...	DEPA...	GATE	ARRIVALTIME	DEPARTURETIME	FLIGHTSTATUS
1	709	609	509	209	210	109	809	309	4	5	11	18		14 DeLayed
2	712	612	512	212	213	112	812	312	2	3	14	4		23 DeLayed
3	715	615	515	215	216	115	815	315	5	6	17	14		9 DeLayed

10. SELECT PILOTS WHO IS MOST EXPERIENCED IN THE DATABASE

```

SELECT PILOTID,FIRSTNAME || LASTNAME AS PILOTFULLNAME,TOTALFLIGHTS
FROM PILOTS
WHERE TOTALFLIGHTS = (SELECT MAX(TOTALFLIGHTS) FROM PILOTS );

```

```

--SELECT PILOTS WHO IS MOST EXPERIENCED IN THE DATABASE
SELECT PILOTID,FIRSTNAME || LASTNAME AS PILOTFULLNAME,TOTALFLIGHTS
FROM PILOTS
WHERE TOTALFLIGHTS = (SELECT MAX(TOTALFLIGHTS) FROM PILOTS );

```

SQL | All Rows Fetched: 1 in 0.124 seconds

	PILOTID	PILOTFULLNAME	TOTALFLIGHTS
1	520	MiaJohnson	2400

11. DISPLAY PILOT INFO OF THE FLIGHTS WHICH ARE DELAYED OR CANCELLED

```

SELECT PILOTID
FROM FLIGHTS
WHERE UPPER(FLIGHTSTATUS) = 'DELAYED' OR UPPER(FLIGHTSTATUS) =
'CANCELLED'
GROUP BY PILOTID
ORDER BY PILOTID;

```



```
--DISPLAY PILOT INFO OF THE FLIGHTS WHICH ARE DELAYED OR CANCELLED
SELECT PILOTID
FROM FLIGHTS
WHERE UPPER(FLIGHTSTATUS) = 'DELAYED' OR UPPER(FLIGHTSTATUS) = 'CANCELLED'
GROUP BY PILOTID
ORDER BY PILOTID;
```

SQL | All Rows Fetched: 7 in 0.136 seconds

	PILOTID
1	502
2	504
3	506
4	509
5	512
6	515
7	518

12. DISPLAY ALL THE CREWMEMBERS IN CREW AS WELL AS FLIGHTS INCLUDING ALL THE DUPLICATES

```
SELECT CREWID,flightattendant1 || ' ' || flightattendant2 || ' ' ||flightattendant3 AS
CREWMEMBERS
FROM CREW
UNION ALL
SELECT CREWID, TO_CHAR(NULL)
FROM flights
GROUP BY CREWID
```

ORDER BY CREWID;

```
--DISPLAY ALL THE CREWMEMBERS IN CREW AS WELL AS FLIGHTS INCLUDING ALL THE DUPLICATES
SELECT CREWID,flightattendant1 || ',' || flightattendant2 || ',' ||flightattendant3 AS CREWMEMBERS
FROM CREW
UNION ALL
SELECT CREWID, TO_CHAR(NULL)
FROM flights
GROUP BY CREWID
ORDER BY CREWID;
```

SQL | All Rows Fetched: 40 in 0.143 seconds

	CREWID	CREWMEMBERS
20	810	(null)
21	811	(null)
22	811	Sophia Anderson,Owen Moore,Isabella Harris
23	812	(null)
24	812	Mason Martinez,Harper Johnson,Jacob Jones
25	813	Thomas Taylor,Liam Brown,Zoe Moore
26	813	(null)
27	814	(null)
28	814	Aiden White,Charlotte Thomas,William Johnson
29	815	Grace Wilson,Michael Garcia,Mia Anderson
30	815	(null)
31	816	Liam Johnson,Sophie White,Mason Brown
32	816	(null)
33	817	(null)
34	817	Charlotte Taylor,Lucas Thomas,Lily Martinez
35	818	Ella Johnson,Ethan Garcia,Olivia White
36	818	(null)
37	819	(null)
38	819	Oliver Anderson,Ava Robinson,James Brown
39	820	Sophie Johnson,Benjamin Moore,Chloe Thomas
40	820	(null)

13. DISPLAY THE CREW ID WHICH HAS HIGHEST PILOTID

SELECT CREWID

FROM CREW

WHERE pilotid = (SELECT MAX(PILOTID) FROM CREW)

```
--DISPLAY THE CREW ID WHICH HAS HIGHEST PILOTID
SELECT CREWID
FROM CREW
WHERE pilotid = (SELECT MAX(PILOTID) FROM CREW)
```

SQL | All Rows Fetched: 1 in 0.164 second

	CREWID
1	820