Write Queries to Use SET Operators

1. Find the train numbers for which reservation have not yet been made.

SQL> select train\_number from ticket where pnr\_no in 2 (select pnr\_no from ticket minus

* 1. select pnr\_no from passenger\_details
  2. where reservation\_status ='Confirmed');

TRAIN\_NUMBER

15430

12755

12300

1. Find the train names that do not have a first AC class coach. SQL> select name from train where train\_number in

2 (select train\_number from train minus 3 select train\_no from train\_ticket\_fare 4 where class in ('1A','2A','3A','EA'));

NAME

Charminar Express Kaveri Express

1. Print all the PNR no’s available in the database.

SQL> select pnr\_no from ticket union select pnr\_no from passenger\_details;

PNR\_NO

1098452132

1174520980

1182300376

1198326700

5234109753

1. Find passenger names who have booked to 'Pune'.

SQL> select name from passenger\_details where pnr\_no in(select pnr\_no from ticket where to\_station='Pune');

NAME

Ruhan

Use Nested Query(in Operators)

1. Find the train names that stop in 'Katpadi'.

SQL> select name from train where train\_number in(select train\_number from train where destination='Katpadi');

NAME

Katpadi Express

1. Find the train names that are superfast and the service tax is zero.

SQL> select name from train where train\_number in(select train\_no from train\_ticket\_fare 2 where superfast\_charge is NOT NULL and (service\_tax is NULL or service\_tax=0));

NAME

Rajdhani Express

1. Find the Passenger name who have booked for the train that starts from 'Chennai'. SQL> select name from passenger\_details

2 where pnr\_no in (select pnr\_no from ticket where train\_number in 3 (select train\_number from train where source='Chennai'));

NAME

Ruhan

1. Find the trains names that have all the AC coaches and the base fare is less than 3000 for each case.

SQL> select name from train where train\_number in

2 (select train\_no from train\_ticket\_fare where class in ('1A','2A','3A','EA') 3 intersect select train\_no from train\_ticket\_fare where base\_fare < 3000);

NAME

Shatabdi Express Doronto Express Krishna Express Rajdhani Express

Use Join Query

1. Find the train names that stop in 'Katpadi'.

SQL> select name

1. from train natural join ticket
2. where train.source='Katpadi' or ticket.to\_station='Katpadi';

NAME

Katpadi Express

1. Find the train names that are superfast and the service tax is zero.

SQL> select name from train t,train\_ticket\_fare tf

2 where t.train\_number=tf.train\_no and superfast\_charge is NOT NULL and 3 service\_tax=0;

NAME

Rajdhani Express

1. Find the Passenger name (and train name) who have booked for the train that starts from 'Chennai'.

SQL> select name from passenger\_details where pnr\_no in 2 (select pnr\_no from ticket where train\_number in

3 (select train\_number from train where source='Chennai'));

NAME

Ruhan

1. Display the train names, each type of class and the total fare for each type of class. 8 rows selected.

SQL> select name,class,total\_ticket\_fare from train natural join ticket 2 order by class;

NAME CLASS TOTAL\_TICKET\_FARE

|  |  |  |
| --- | --- | --- |
| Rajdhani Express | 1A | 1200 |
| Shatabdi Express | 1A | 600 |
| Katpadi Express | 1A | 250 |

7 rows selected.

1. Display all the train details and the ticket details (if booked any).

SQL> select \* from train natural join ticket;

TRAIN\_NUMBER CLASS NAME SOURCE DESTINATION START\_TIME

REACH\_TIME TRAVELTIME DISTANCE DAYS TYPE PNR\_NO TRANSACTIONID FROM\_STATION TO\_STATION DATE\_OF\_J

-

-

DATE\_OF\_B TOTAL\_TICKET\_FARE

15291 1A Katpadi Express Chennai Katpadi 15-OCT-22 04.00.00.000000 AM

15-OCT-22 08.30.00.000000 AM 4.5 140 1

Short-Distance 2348761908 7.6543E+14 Chennai Katpadi 15-OCT-22 01-OCT-22 250

11625 1A Shatabdi Express Amritsar Delhi 20-SEP-23 02.00.00.000000 AM

20-SEP-23 09.00.00.000000 AM 7 450 1

Long-Distance 5234109753 9.8070E+14 Amritsar Delhi 20-SEP- 23

17-JAN-23 600

1. Create a sequence to provide values for the PNR no.

SQL> create sequence pnr\_no\_sequence start with 1 increment by 2 minvalue 1 maxvalue 10 nocycle;

Sequence created.

SQL> select \* from user\_sequences;

SEQUENCE\_NAME MIN\_VALUE MAX\_VALUE INCREMENT\_BY C O CACHE\_SIZE LAST\_NUMBER

- - PNR\_NO\_SEQUENCE 1 10 2 N N 20 1

1. Write a query for full outer join using any of the tables above.

SQL> select \* from train train full outer join train\_route on train.train\_number=train\_route.train\_no;

TRAIN\_NUMBER NAME SOURCE DESTINATION START\_TIME REACH\_TIME

TRAVELTIME DISTANCE CLASS DAYS TYPE TRAIN\_NO ROUTE\_NO STA NAME ARRIVAL\_T DEPART\_TI DISTANCE

-

DAY HALT\_TIME

11625 Shatabdi Express Amritsar Delhi 20-SEP-23 02.00.00.000000 AM 20-SEP-23 09.00.00.000000 AM 7 450 1A

1 Long-Distance 1625 4 ATR RamDas Station 20-SEP-23 20-SEP-23 450

1 0

14981 Doronto Express Hyderabad Bangalore 15-AUG-22 05.27.30.000000 AM 17-AUG-22 08.11.00.000000 AM

6 400 EA 1 Short-Distance 4981 67 HYD Memorial Station 17-AUG-22 15-AUG-22 400

1 -2

15430 Krishna Express Vellore Tirupati 04-DEC-19 06.00.00.000000 AM 04-DEC-19 10.00.00.000000 AM 4.2 200 2A

1 Named 5430 11 VEL Katpadi Station 04-DEC-19 04-DEC-19

200

1 0

9 rows selected.

Write Queries to. (Middle Level) Assignment 2 Use Correlated (and nested) Query

1. Find the train names for which ten tickets have been reserved.

SQL> select name from train where train\_number in(select train\_number from train intersect

2 select train\_number from ticket group by train\_number having count(\*)=10); no rows selected

1. Find the trains that have more than ten substations.

SQL> select train\_no from train\_route where station\_code in (select station\_code from 2 train\_route group by station\_code having count(\*)>10);

no rows selected

1. Find the passengers who do not pass through 'Mettupalam'.

SQL> select passenger\_details.name ,passenger\_details.age from train\_route,ticket,passenger\_details where passenger\_details.pnr\_no=ticket.pnr\_no

2 and ticket.train\_number=train\_route.train\_no and 3 train\_route.station\_code !='Mettupalam';

NAME AGE

|  |  |
| --- | --- |
| Jayalata | 78 |
| Suresh | 67 |
| Mahira | 22 |
| 6 rows selected. |  |

1. Find passengers who have booked for superfast trains.

SQL> select \* from passenger\_details where pnr\_no in (select pnr\_no from passenger\_details intersect select pnr\_no from ticket where train\_number in

2 (select train\_number from ticket intersect select train\_no from train\_ticket\_fare 3 where superfast\_charge!=0));

PNR\_NO SERIAL\_N NAME AGE

RESERVATION\_STATUS

5234109753 3 Jayalata 78

Confirmed

1098452132 4 Suresh 67

Waitlisted

1182300376 10 Mahira 22

Waitlisted

Complex queries(use groupby/groupby having/join/nested)

1. Take the start station code and end station code and display the train details. SQL> select train.train\_number,name,source,destination,start\_time,reach\_time, 2 traveltime,distance,train.class,days,train.type from train,ticket where

3 train.train\_number=ticket.train\_number and ticket.from\_station='&from\_station' 4 and ticket.to\_station='&to\_station';

Enter value for from\_station: Chennai

old 3: train.train\_number=ticket.train\_number and ticket.from\_station='&from\_station' new 3: train.train\_number=ticket.train\_number and ticket.from\_station='Chennai'

Enter value for to\_station: Pune

old 4: and ticket.to\_station='&to\_station' new 4: and ticket.to\_station='Pune'

TRAIN\_NUMBER NAME SOURCE DESTINATION START\_TIME REACH\_TIME TRAVELTIME DISTANCE CLASS DAYS TYPE

-

13213 Rajdhani Express Chennai Pune 20-OCT-22 03.00.00.780000 AM 23-OCT-22 05.00.00.000000 AM

78.3 1523 EV 3 Long-Distance

1. List the train names and the number of sub stations it has.

SQL> select train\_no,count(station\_code) from train\_route group by train\_no;

TRAIN\_NO COUNT(STATION\_CODE)

14981 1

12300 1

11625 1

15430 1

1. List the stations where all types of trains stop.

SQL> select station\_code from train\_route where train\_no in 2 (select train\_number from train);

STA

--- ATR HYD VEL RJK

6 rows selected.

1. List the train names that have at least four bookings. SQL> select name from train where train\_number in
2. (select train\_number from train intersect select train\_number from ticket 3 group by train\_number having count(\*)>=4);

no rows selected

1. Create a table cancellation history ( Insert values from ticket and passenger table).

SQL> create table cancellation\_list as select train\_number,passenger\_details.name, 2 ticket.pnr\_no,ticket.date\_of\_journey from ticket,passenger\_details where

1. ticket.pnr\_no=passenger\_details.pnr\_no; Table created.

SQL> select \* from cancellation\_list;

TRAIN\_NUMBER NAME PNR\_NO DATE\_OF\_J

11625 Jayalata 5234109753 20-SEP-23

12300 Suresh 1098452132 19-NOV-22

15430 Mahira 1182300376 04-DEC-19

15430 Lauryn 1174520980 04-DEC-19

6 rows selected.

1. Create a table for all the train numbers and class available in train\_ticket\_fare with total seats.

SQL> create table Total\_seat\_count as

1. (select ttf.train\_no,ttf.class,count(t.train\_number) Total\_seats from train\_ticket\_fare ttf,ticket t where ttf.train\_no = t.train\_number
2. and ttf.class=t.class
3. group by ttf.train\_no,ttf.class); Table created.

SQL> select \* from total\_seat\_count;

TRAIN\_NO CLASS TOTAL\_SEATS

12755 1A 1

12300 EV 1

1. Find the station name that has highest number of trains stopping at.

SQL> select station\_code,count(station\_code) from train\_route group by 2 station\_code having count(station\_code)=(select max(mycount) from 3 (select station\_code,count(station\_code) mycount from train\_route 4 group by station\_code));

STA COUNT(STATION\_CODE)

|  |  |
| --- | --- |
| HYD | 1 |
| BMB | 1 |
| RJK | 1 |
| MSB | 1 |

6 rows selected.