

CS 186 Discussion Section

Week 10

Peter Alvaro and Kuang Chen

March 29, 2009

1 SQL Review

1.1 basics

SELECT [DISTINCT], FROM, WHERE, ORDER BY ASC | DESC

1.2 set manipulation

UNION, INTERSECT, EXCEPT, IN, EXISTS, *operator* ANY | ALL

1.3 aggregates

COUNT, SUM, AVG, MAX, MIN

1.4 grouping

GROUP BY, HAVING

1.5 joins

INNER, LEFT | RIGHT | FULL OUTER, NATURAL

2 Writing SQL

Consider the following schema on table 1:

```
Flights(flight_num, source_city, destination_city)
Departures(flight_num, date, plane_type)
Passengers(passenger_id, passenger_name, passenger_address)
Bookings(passenger_id, flight_num, date, seat_number)
```

Table 1: Schema for the supply chain database.

The key fields are underlined. Express the following queries in SQL (feel free to abbreviate relation and attribute names and to use INTERSECT and EXCEPT if you need to):

1. Find the `passenger_id` of all passengers who have a seat booked on a plane of type “747” from San Francisco to Washington. *Do not return any duplicate values.*

```

SELECT DISTINCT B.passenger_id
FROM Flights F, Departures D, Bookings B
WHERE B.flight_num = D.flight_num
      AND B.date = D.date
      AND F.flight_num = D.flight_num
      AND F.source_city = "San Francisco"
      AND F.destination_city = "Washington"
      AND D.plane_type = "747";

```

2. Find the cities that have direct (non-stop) flights to both Honolulu and Newark.

```

SELECT DISTINCT source_city
FROM Flights F
WHERE F.dest_city = "Honolulu"
      AND F.source_city IN
      (SELECT source_city
       FROM Flights F2
       WHERE dest_city = "Newark");

```

Alternatively, we could use the Intersect operator or self-join the Flights relation. Here is how the query should look:

```

SELECT DISTINCT source_city
FROM Flights F1, Flights F2
WHERE F1.dest_city = "Honolulu"
      AND F2.dest_city = "Newark"
      AND F1.source_city = F2.source_city;

```

or

```

(SELECT DISTINCT source_city
 FROM Flights F
 WHERE F.dest city = "Honolulu")
INTERSECT
(SELECT DISTINCT source_city
 FROM Flights F
 WHERE F.dest city = "Newark")

```

3. Find the flight_num and date of all flights for which there are no reservations.

```

SELECT flight_num, date
FROM Departures D
WHERE NOT EXISTS
      (SELECT *
       FROM Bookings B
       WHERE B.flight_num = D.flight_num
             AND B.date = D.date);

```

4. Find the passenger_name of all passengers who have a seat booked on at least one plane of *every* type.

```

SELECT DISTINCT passenger_name
FROM Passengers P
WHERE
    (SELECT COUNT(DISTINCT D.plane_type)
     FROM Departures D, Bookings B
     WHERE D.flight_num = B.flight_num
          AND D.date = B.date
          AND B.passenger_id = P.passenger_id) =
        (SELECT COUNT(DISTINCT D.plane_type)
         FROM Departures D);

```

or

```

SELECT DISTINCT passenger_name
FROM Passengers P
WHERE NOT EXISTS
    (SELECT D.plane_type
     FROM Departures D
     WHERE NOT EXISTS
        (SELECT *
         FROM Departures D2, Bookings B
         WHERE D2.flight_num = B.flight_num
              AND D2.date = B.date
              AND B.passenger_id = P.passenger_id
              AND D.plane_type = D2.plane_type));

```

5. Print an ordered list of all source cities and the number of distinct destination cities that they have direct (non-stop) flights to, with airplanes of type "747". The list should be ordered in decreasing number of destinations and should contain only those source cities that have flights to 25 or more distinct destinations.

For example, the output should look like:

source_city	NumDestinations
Chicago	228
Atlanta	100
...	
San Francisco	25

```

SELECT source_city, COUNT(DISTINCT destination_city) AS NumDestinations
FROM Flights F, Departures D
WHERE F.flight_num = D.flight_num
     AND D.plane_type = "747"
GROUP BY source_city
HAVING NumDestinations >= 25
ORDER BY NumDestinations DESC;

```