Computer Science 143, Homework 2

Michael Wu
UID: 404751542

May 2nd, 2018

Problem 1

a) I ran the following query

```
SELECT highway, area
FROM caltrans
WHERE text LIKE '%closed%'
  AND (text LIKE '%for the winter%'
    OR text LIKE '%due to snow%')
GROUP BY highway, area
ORDER BY highway DESC, area DESC
LIMIT 20;
```

and got the result set shown below.

highway	area
US395	IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA
SR89	IN THE NORTHERN CALIFORNIA AREA & SIERRA NEVADA
SR89	IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA
SR88	IN THE CENTRAL CALIFORNIA & SIERRA NEVADA
SR4	IN THE CENTRAL CALIFORNIA AREA
SR38	IN THE SOUTHERN CALIFORNIA AREA
SR330	IN THE SOUTHERN CALIFORNIA AREA
SR33	IN THE SOUTHERN CALIFORNIA AREA
SR3	IN THE NORTHERN CALIFORNIA AREA
SR270	IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA
SR267	IN THE NORTHERN CALIFORNIA AREA
SR203	IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA
SR20	IN THE NORTHERN CALIFORNIA AREA
SR2	IN THE SOUTHERN CALIFORNIA AREA
SR18	IN THE SOUTHERN CALIFORNIA AREA
SR172	IN THE NORTHERN CALIFORNIA AREA
SR168	IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA
SR158	IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA
SR138	IN THE SOUTHERN CALIFORNIA AREA
SR130	IN THE CENTRAL CALIFORNIA AREA

b) I ran the following query

```
SELECT a.highway, a.area, closedPct
FROM
  (SELECT highway, area
   FROM caltrans
   WHERE text LIKE '%closed%'
     AND (text LIKE '%for the winter%'
       OR text LIKE '%due to snow%')
   GROUP BY highway, area
   ORDER BY highway DESC, area DESC)
  AS a
JOIN
  (SELECT highway, area, count(*)*100/365 AS closedPct
   FROM
     (SELECT DATE(reported), highway, area
```

```
      FROM caltrans
      WHERE text LIKE '%closed%'
      GROUP BY highway, area, DATE(reported))
     AS closedDays
   GROUP BY highway, area)
  AS closure
ON a.highway = closure.highway
  AND a.area = closure.area
ORDER BY closedPct DESC
LIMIT 5;
```

and got the following result set.

highway	area	closedPct
SR120	IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA	66.5753
SR89	IN THE NORTHERN CALIFORNIA AREA & SIERRA NEVADA	66.5753
SR4	IN THE CENTRAL CALIFORNIA AREA	63.5616
SR203	IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA	61.3699
SR108	IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA	55.6164

Problem 2

a) A natural join will result in a cross join when there are no common attributes. Thus the venn diagram should have their areas overlap. In fact, any equi join with no join key will result in a cross join, so cross join should be a subset of equi join. The same can be said about non equi join, since that can result in a cross join as well. Overall the relationship between different types of joins is fairly complex and cannot be expressed in a nice form using a venn diagram, as there are too many types of joins. Venn diagrams are usually only good if there are two sets being compared.

Problem 3

a) I ran the following query

```
SELECT trip_starts.trip_id, trip_starts.user_id,
  IF(
    ISNULL(trip_ends.time),
```

```
    '24:00:00',
    SEC_TO_TIME(
      TIMESTAMPDIFF(SECOND, trip_starts.time, trip_ends.time)
    )
  ) AS trip_length
FROM trip_starts
LEFT JOIN trip_ends
ON trip_starts.trip_id=trip_ends.trip_id
LIMIT 5;
```

and got the following result.

trip_id	user_id	trip_length
0	20685	00:01:12
2	34808	00:02:59
3	25463	24:00:00
4	26965	00:01:34
5	836	00:00:51

b) I ran the following query

```
SELECT trip_starts.trip_id, trip_starts.user_id,
  IF(
    ISNULL(trip_ends.time),
    217.00,
    1+0.15*CEILING(
      TIMESTAMPDIFF(SECOND, trip_starts.time, trip_ends.time)
      /60
    )
  ) AS trip_charge
FROM trip_starts
LEFT JOIN trip_ends
ON trip_starts.trip_id=trip_ends.trip_id
LIMIT 5;
```

and got the following results

trip_id	user_id	trip_charge
0	20685	1.30
2	34808	1.45
3	25463	217.00
4	26965	1.30
5	836	1.15

c) I ran the following query

```
SELECT trip_starts.user_id,
  SUM(IF(
    ISNULL(trip_ends.time),
    217.00,
    1+0.15*CEILING(
      TIMESTAMPDIFF(SECOND, trip_starts.time, trip_ends.time)
      /60
    )
  )) AS monthly_total
FROM trip_starts
LEFT JOIN trip_ends
ON trip_starts.trip_id=trip_ends.trip_id
WHERE MONTH(trip_starts.time) = 3
GROUP BY trip_starts.user_id
LIMIT 5;
```

and got the following result.

user_id	monthly_total
0	222.50
1	4.05
2	665.05
3	11.90
4	444.55

The user with `user_id` = 2 owes \$665.05 for the month of March.

d) You would have to use a self left join on the trip id, and then use a where clause to only choose rows where the left side is a start entry and the right side is an end entry.