Part1

(a)

```
SELECT DISTINCT highway,area
FROM hw2.caltrans
WHERE condition LIKE '%FOR THE WINTER%' OR condition LIKE '%DUE TO SNOW%'
ORDER BY highway DESC, area DESC
LIMIT 20;
```

Result:

```
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
 SR58    | IN THE CENTRAL CALIFORNIA AREA
 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
```

(b)

```
--method 1 with join
SELECT highway,area, CAST(COUNT(dateNum)AS DECIMAL)/328*100  AS percentage
FROM(
        SELECT DISTINCT L.highway AS highway,L.area AS area, date_trunc('day',reported) AS
dateNum
        FROM
                (SELECT DISTINCT
                        highway,area
                FROM hw2.caltrans
                WHERE condition LIKE '%FOR THE WINTER%' OR condition LIKE '%DUE TO
SNOW%'
                ORDER BY highway DESC, area DESC
        )L
        JOIN hw2.caltrans R
        ON L.highway=R.highway AND L.area=R.area
```

```
        WHERE condition LIKE '%IS CLOSED%') sq
GROUP BY highway,area
ORDER BY percentage DESC
LIMIT 5;
--method without join
SELECT highway,area, CAST(COUNT(dateNum)AS DECIMAL)/328*100  AS percentage
FROM(
        SELECT DISTINCT highway, area, date_trunc('day',reported) AS dateNum
        FROM hw2.caltrans
        WHERE condition LIKE '%IS CLOSED%' AND(highway,area) IN
                (SELECT DISTINCT
                        highway,area
                FROM hw2.caltrans
                WHERE condition LIKE '%FOR THE WINTER%' OR condition LIKE '%DUE TO
SNOW%'
                ORDER BY highway DESC, area DESC
        )
                )sq

GROUP BY highway,area
ORDER BY percentage DESC
LIMIT 5;

--count days
SELECT highway,area,COUNT(dateNum)
FROM(
SELECT DISTINCT highway,area, date_trunc('day',reported) AS dateNum
FROM hw2.caltrans
GROUP BY highway,area,date_trunc('day',reported)) tt
GROUP BY highway,area;
```

Result:

```
highway |                      area                       |
percentage
---------+-------------------------------------------------+----------
--------------
 SR89    | IN THE NORTHERN CALIFORNIA AREA & SIERRA NEVADA |
74.08536585365853658500
 SR120   | IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA  |
74.08536585365853658500
 SR4     | IN THE CENTRAL CALIFORNIA AREA                  |
70.73170731707317073200
 SR203   | IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA  |
68.29268292682926829300
```

```
 SR108   | IN THE CENTRAL CALIFORNIA AREA & SIERRA NEVADA  |
61.89024390243902439000
```

Part 2

The relationship between cross-join and inner-join is not correct. Cross-join is Cartesian product while inner join requires both tables match on the join key. Thus, cross-join cannot be a subset of inner join as the given Venn Diagram shows. Cross-join is another kind of join which is different from both inner join and outer join.

Part 3

(a)

```
SELECT
    L.trip_id AS trip_id, L.user_id AS user_id,
    --calculate_time(L.time,R.time) AS trip_length
    COALESCE((EXTRACT(HOUR from R.time)-EXTRACT(HOUR from L.time))*60+
        EXTRACT(MINUTE from R.time)-EXTRACT(MINUTE from L.time)+
        (EXTRACT(SECOND from R.time)-EXTRACT(SECOND from
L.time))/60,24*60 )
    AS trip_length
FROM hw2.trip_start L
LEFT JOIN hw2.trip_end R
on L.trip_id=R.trip_id AND L.user_id=R.user_id
LIMIT 5;trip_id | user_id |   trip_length
---------+---------+-----------------
       0 |   20685 |               1.2
       2 |   34808 | 2.98333333333333
       3 |   25463 |             1440
       4 |   26965 | 1.56666666666667
       5 |     836 |             0.85
```

(b)

```
SELECT
       L.trip_id AS trip_id, L.user_id AS user_id,
       1+CEIL(COALESCE((EXTRACT(HOUR from R.time)-EXTRACT(HOUR from L.time))*60+
              EXTRACT(MINUTE from R.time)-EXTRACT(MINUTE from L.time)+
              (EXTRACT(SECOND from R.time)-EXTRACT(SECOND from
L.time))/60,24*60 ))*0.15 AS trip_charge
FROM hw2.trip_start L
LEFT JOIN hw2.trip_end R
on L.trip_id=R.trip_id AND L.user_id=R.user_id
LIMIT 5;
trip_id | user_id | trip_charge
---------+---------+-------------
       0 |   20685 |         1.3
       2 |   34808 |        1.45
       3 |   25463 |         217
       4 |   26965 |         1.3
```

```
       5 |      836 |         1.15
(5 rows)
```

(c)

```
SELECT COALESCE(A.user_id,B.user_id) AS
user_id,COALESCE(A.trip_charge,0)+COALESCE(B.trip_charge,0) AS monthly_total
FROM
(SELECT user_id,COUNT(day_total)*100 AS trip_charge
FROM
        (SELECT user_id,dateOfuse, SUM(trip_charge) AS day_total
        FROM  (
                SELECT
                L.trip_id AS trip_id, L.user_id AS user_id, EXTRACT(DAY FROM L.time) As
dateOfuse,
                1+CEIL(COALESCE((EXTRACT(HOUR from R.time)-EXTRACT(HOUR from
L.time))*60+
                        EXTRACT(MINUTE from R.time)-EXTRACT(MINUTE from L.time)+
                        (EXTRACT(SECOND from R.time)-EXTRACT(SECOND from
L.time))/60,24*60 ))*0.15 AS trip_charge
                FROM hw2.trip_start L
                LEFT JOIN hw2.trip_end R
                on L.trip_id=R.trip_id AND L.user_id=R.user_id
                WHERE EXTRACT(MONTH FROM L.time)=3 AND EXTRACT(YEAR FROM
L.time)=2018
        ) sq1
        GROUP BY user_id,dateOfuse)sq11
WHERE day_total >=100
GROUP BY user_id)A
FULL JOIN
(SELECT user_id,SUM(day_total) AS trip_charge
FROM
        (SELECT user_id,dateOfuse, SUM(trip_charge) AS day_total
        FROM  (
                SELECT
                L.trip_id AS trip_id, L.user_id AS user_id,  EXTRACT(DAY FROM L.time) As
dateOfuse,
                1+CEIL(COALESCE((EXTRACT(HOUR from R.time)-EXTRACT(HOUR from
L.time))*60+
                        EXTRACT(MINUTE from R.time)-EXTRACT(MINUTE from L.time)+
                        (EXTRACT(SECOND from R.time)-EXTRACT(SECOND from
L.time))/60,24*60 ))*0.15 AS trip_charge
                FROM hw2.trip_start L
                LEFT JOIN hw2.trip_end R
                on L.trip_id=R.trip_id AND L.user_id=R.user_id
```

```
                WHERE EXTRACT(MONTH FROM L.time)=3 AND EXTRACT(YEAR FROM
L.time)=2018
        ) sq2
        GROUP BY user_id,dateOfuse)sq22
WHERE day_total <100
GROUP BY user_id)B
ON A.user_id=B.user_id
LIMIT 5;
user_id | monthly_total
---------+---------------
       0 |         105.5
       1 |          4.05
       2 |        314.05
       3 |          11.9
       4 |        210.55
(5 rows)
```

User_id =2 owe 314.05 for March, 2018.

(d) If we record the starts and ends of trips in one table, then we need to do a self-join here and the join key would be left.trip_id=right.trip_id AND left.user_id=right.user_id.