PART I: Teradata

1. Show a breakdown, by state, of the number of stores

```
SELECT
     state
     COUNT(state) AS "Number of Stores"
FROM
     store_dimension
GROUP BY
     state
ORDER BY
     state;
```

State	Number of Stores
AK	2
AR	1
AZ	2
CA	6
CO	4
FL	16
GA	5
HI	1
IL	9
IN	7
KS	1
KY	3
LA	4
MD	3
ME	2
MI	3
MN	4
MO	6
MS	2
MT	1
NC	3
ND	1
NH	2
NJ	2
NM	1
NY	8
OH	5
OK	2
PA	4
RI	1
SC	3
TN	6
TX	17
UT	2
VA	4
WA	2
WI	6
WV	2

2. List the top 5 shoppers who spent the most on MLK JR Day in 2000

```
SELECT TOP 5
     date_description AS "Date",
     member_key AS MEMBER,
     total_scan_amount AS "TOTAL SPENT"
FROM
      item_scan_fact a
INNER JOIN
      (SELECT
         date_key,
         date_description
      FROM
         date_dimension
      WHERE
         date_description = '2000-01-17') AS b
ON
     a.transaction_date_key = b.date_key
GROUP BY
     transaction_date_key,
     total_scan_amount,
     member_key,
     date_description
ORDER BY
     a.total_scan_amount DESC;
```

Date	MEMBER	TOTAL SPENT
2000-01-17	22196205	6146.20
2000-01-17	27125029	1614.98
2000-01-17	19896225	1314.98
2000-01-17	27104369	1114.98
2000-01-17	24229963	1114.98

3. Count the number of members who visited 2 or more stores in 2000

```
SELECT
  COUNT(*) AS "Members Who Visited 2+ Stores in 2000"
FROM(
  SELECT
     a.member_key
  FROM
     item_scan_fact a
  INNER JOIN(
     SELECT
        date_key
     FROM
        date_dimension
     WHERE
        year_number = '2000') AS b
  ON
     a.transaction_date_key = b.date_key
  GROUP BY
     a.member_key
  HAVING
     COUNT(DISTINCT a.store_key) > 1) a;
```

Members Who Visited 2+ Stores in 2000
3604

4. Count the number of items that have never sold

```
SELECT
   COUNT(a.item_key)
FROM
   item_dimension a
WHERE
    a.item_key
NOT IN(
    SELECT
       item_key
    FROM
       item_scan_fact);
```

Items Never Sold
422215

5. By store, list the maximum number of store visits made by a customer

```
SELECT *
FROM
      (SELECT
         store,
         COUNT(c.visit_number) AS totals,
         customer
      FROM
            (SELECT
               a.store_key AS store,
               a.visit_number,
               a.member_key AS customer
            FROM
               item_scan_fact AS a,
               store_dimension AS b

            WHERE
               a.store_key = b.store_key
            GROUP BY
               a.store_key,
               a.visit_number,
               a.member_key
            ) c

      GROUP BY
            store,
            customer
      ) d
QUALIFY row_number() OVER(
   partition by store order by totals desc) = 1;
```

store	totals	customer
3	720	22988103
4	612	25529657
5	1508	28118711
15	2676	30455657
16	1606	25400711
17	1556	28105199
18	3829	27102566
19	2364	28315725
20	1539	18988528
21	183	28556772
22	2575	25015150
23	1355	18692568
24	1683	17593244
25	1122	26631703
26	56	26039953
27	1706	30945232
28	3092	28051061
29	1741	25821363
151	988	22988103
152	715	25529657
153	1977	28118711