Q) Create an aggregation query where PostgreSQL uses sort-based aggregation

Solution:
Here's an example of an aggregation query where PostgreSQL would use sort-based aggregation:

```
    SELECT dept_name, COUNT(*)
FROM instructor
GROUP BY dept_name
ORDER BY COUNT(*) DESC;
```

This query is counting the number of instructors in each department, and then ordering the results by the count in descending order. PostgreSQL will use sort-based aggregation to perform this query, which involves sorting the data by department and then aggregating the counts.

```
explain SELECT dept_name, COUNT(*)
FROM instructor
GROUP BY dept_name
ORDER BY COUNT(*) DESC;
                                  QUERY PLAN
------------------------------------------------------------------------------
-
 Sort  (cost=26.24..26.74 rows=200 width=66)
   Sort Key: (count(*)) DESC
   ->  HashAggregate  (cost=16.60..18.60 rows=200 width=66)
         Group Key: dept_name
         ->  Seq Scan on instructor  (cost=0.00..14.40 rows=440 width=58)
(5 rows)


explain analyze SELECT dept_name, COUNT(*)
FROM instructor
GROUP BY dept_name
ORDER BY COUNT(*) DESC;
                                                          QUERY PLAN
------------------------------------------------------------------------------
-----------------------------------------------
 Sort  (cost=26.24..26.74 rows=200 width=66) (actual time=0.169..0.173
rows=17 loops=1)
   Sort Key: (count(*)) DESC
   Sort Method: quicksort  Memory: 26kB
   ->  HashAggregate  (cost=16.60..18.60 rows=200 width=66) (actual
time=0.067..0.074 rows=17 loops=1)
         Group Key: dept_name
         Batches: 1  Memory Usage: 40kB
         ->  Seq Scan on instructor  (cost=0.00..14.40 rows=440 width=58)
(actual time=0.011..0.018 rows=50 loops=1)
 Planning Time: 0.173 ms
 Execution Time: 0.234 ms
(9 rows)
```