Want to test your intermediate SQL knowledge? Try out these 7 example conceptual questions I have used in past interviews:  
  
1. When will ROW\_NUMBER and RANK give different results? Give an example.  
  
2. Is it possible for LEFT JOIN and INNER JOIN to produce the same results? If yes, when? If no, why?  
  
3. Why would I use DENSE\_RANK instead of RANK? What about RANK instead of DENSE\_RANK?  
  
4. What happens if I GROUP BY a column that is not in the SELECT statement? Why does this happen?  
  
5. LAG and LEAD are especially useful in what type of scenarios?  
  
6. For dealing with NULL values, why would I choose to use IFNULL vs. CASE WHEN?  
  
7. Do temp tables make your code cleaner and faster, one of the two, or none? Why?  
  
Any favorites that you ask?

For point 1-> Row\_Number() is useful when finding out duplicates because Row\_Number Assigns unique row\_number to duplicates as well. Whereas Rank() would give the same Rank to duplicate records.

JOIN - combines tables columnwise Union - combines table rowwise

Even if am right, i still need some enlgihtenment.

1.) When we use PARTITION BY any categorical field/ when the column on which windowing function is applied is having repeated values

2.) When joining tables has the same number of rows AND same values in the matching column

3.) DENSE\_Rank() - gives a consecutive number to the rows ordered within a partition

RANK() - also does the same but the difference is when there is a tie on a value of that particular field within the partition.

4.) Yes, that's possible.

Ex: SELECT MAX(salary) FROM employee GROUP BY dept\_id;

5.) Lag, Lead is most useful when calculating the increase/decrease in sales/temparature/.... Mostly used to find the performance.

6.) CASE WHEN - possible when replacement is done for a limited number. If the number of NULL's are many then we need to impute the NULL with a value (could be Mean/median/.....) using IFNULL.

7.) Yes, Local to that session and used to perform and data validations/ query checks. Temp tables are useful when we are trying to load/insert very huge data into the Data warehouse. they act as a staging table