Subject Name: Advance Database System **Module**: 2

Quarter: Prelim

Identification: (Sa ubos magtuon)

1. _____- specifies a temporary named result set.
2. _____- Doesn't use repeated procedural loops/recursion.
3. _____- Uses recursion
4. _____– A query (SELECT statement) inside another query.
5. _____– work as part of the row selection process.
6. _____– returns a temporary or virtual table.
7. _____– a subquery that is nested in the list of another SELECT statement
8. _____– allows users to match one item from any of those in the list.
9. _____– returns true if any of the subquery values satisfy the condition.
10. _____– returns true if all the subquery values meet the condition.
11. _____– are used to select data from a table referenced in the outer query.
12. _____– used to check whether a subquery produces any rows of query results.
13. _____– a virtual table that is constructed from other tables or views and saved as an object in the database.
14. _____– used to speed up searches/queries, resulting in high performance.
15. _____– creating an index to a particular column that is frequently searched can give performance benefits.
16. _____– putting an index on a relatively large table that contains a great number of rows can improve performance.
17. _____– a database that is frequently updated should have f ewer indexes as it slows the performance of inserts, updates, and deletes.
18. _____– create an index only if necessary, because indexes take up spaces within the database.
19. _____– based on only one table column.
20. _____– does not allow any duplicate values to be inserted into the table.
21. _____– based on two or more columns of a table.
22. _____– deleting an index can be done using the DROP command.

Subject Name: Advance Database System **Module**: 2

Quarter: Prelim

Types of CTE:

- _____

- _____

Factors to consider creating an index:

- _____

- _____

- _____

- _____

Subject Name: Advance Database System **Module**: 2

Quarter: Prelim

Answer Key:

1. **Common Table Expression (CTE)** - specifies a temporary named result set.
2. **Non-Recursive CTE** - Doesn't use repeated procedural loops/recursion.
3. **Recursive CTE** - Uses recursion
4. **Subqueries** – A query (SELECT statement) inside another query.
5. **WHERE clause** – work as part of the row selection process.
6. **FROM clause** – returns a temporary or virtual table.
7. **SELECT clause** – a subquery that is nested in the list of another SELECT statement
8. **IN operator** – allows users to match one item from any of those in the list.
9. **ANY** – returns true if any of the subquery values satisfy the condition.
10. **ALL** – returns true if all the subquery values meet the condition.
11. **Correlated Subqueries** – are used to select data from a table referenced in the outer query.
12. **EXISTS operator** – used to check whether a subquery produces any rows of query results.
13. **Views** – a virtual table that is constructed from other tables or views and saved as an object in the database.
14. **Index** – used to speed up searches/queries, resulting in high performance.
15. **Frequency of search** – creating an index to a particular column that is frequently searched can give performance benefits.
16. **Size of table** – putting an index on a relatively large table that contains a great number of rows can improve performance.
17. **Number of updates** – a database that is frequently updated should have f ewer indexes as it slows the performance of inserts, updates, and deletes.
18. **Space considerations** – create an index only if necessary, because indexes take up spaces within the database.
19. **Single-Column Indexes** – based on only one table column.
20. **Unique Indexes** – does not allow any duplicate values to be inserted into the table.
21. **Composite Indexes** – based on two or more columns of a table.
22. **Dropping Index** – deleting an index can be done using the DROP command.

Subject Name: Advance Database System **Module**: 2

Quarter: Prelim

Types of CTE:

- Non-Recursive CTE

- Recursive CTE

Factors to consider creating an index:

- Frequency of search

- Size of table

- Number of updates

- Space considerations