Using Views

1. What is a view in SQL?
    - A) A stored procedure that returns data
    - B) A virtual table based on the result of a SELECT query
    - C) A table that holds temporary data
    - D) A function that updates data
2. Which SQL statement is used to create a view?
    - A) CREATE VIEW
    - B) CREATE TABLE
    - C) CREATE FUNCTION
    - D) CREATE PROCEDURE
3. Can views be used to restrict access to certain columns in a table?
    - A) Yes
    - B) No
4. Which statement is used to delete a view?
    - A) DROP VIEW
    - B) DELETE VIEW
    - C) REMOVE VIEW
    - D) ALTER VIEW
5. Can a view be updated directly?
    - A) Always
    - B) Never
    - C) It depends on the complexity of the view
    - D) Only if it is a materialized view

Using Inline TVFs

6. What does TVF stand for in SQL?
    - A) Table-Valued Function
    - B) Table-View Function
    - C) Temporary-Value Function
    - D) Temporary-View Function
7. What is an Inline TVF?
    - A) A function that returns a scalar value
    - B) A function that returns a table
    - C) A view that includes subqueries
    - D) A stored procedure with parameters
8. How is an Inline TVF different from a stored procedure?
    - A) TVFs cannot return tables
    - B) TVFs can be used in the FROM clause of a query
    - C) TVFs cannot have parameters
    - D) TVFs are not reusable
9. Which keyword is used to define an Inline TVF?
    - A) FUNCTION
    - B) VIEW

- C) PROCEDURE
- D) TABLE

10. Can Inline TVFs accept parameters?
    - A) Yes
    - B) No

Using Derived Tables

11. What is a derived table in SQL?
    - A) A temporary table that is the result of a subquery in the FROM clause
    - B) A permanent table stored in the database
    - C) A table that is created and updated dynamically
    - D) A view that includes a subquery
12. How do you create a derived table in a query?
    - A) By using a subquery in the FROM clause
    - B) By using the CREATE TABLE statement
    - C) By using the ALTER TABLE statement
    - D) By using the CREATE VIEW statement
13. Can derived tables be used with JOIN operations?
    - A) Yes
    - B) No
14. Do derived tables persist after the query execution?
    - A) Yes
    - B) No

Using CTEs

15. What does CTE stand for?
    - A) Common Table Expression
    - B) Complex Table Expression
    - C) Common Temporary Expression
    - D) Complex Temporary Expression
16. How do you define a CTE in SQL?
    - A) By using the WITH keyword
    - B) By using the CREATE CTE statement
    - C) By using the DECLARE CTE statement
    - D) By using the SELECT INTO statement
17. Can CTEs be recursive?
    - A) Yes
    - B) No
18. Which of the following is a correct usage of a CTE?
    - A) WITH CTEName AS (SELECT * FROM Employees) SELECT * FROM CTEName;
    - B) CREATE CTE CTEName AS (SELECT * FROM Employees) SELECT * FROM CTEName;

- C) DECLARE CTEName AS (SELECT * FROM Employees) SELECT * FROM CTEName;
- D) SELECT * FROM CTEName AS (SELECT * FROM Employees);

19. Can you use multiple CTEs in a single query?
    - A) Yes
    - B) No
20. What is the main advantage of using CTEs?
    - A) They improve query performance
    - B) They provide better readability and organization for complex queries
    - C) They persist beyond the query execution
    - D) They are faster than derived tables