**Q1.** What will be the result of the following query?

SELECT 10 / 3;

A. 3  
B. 3.3333  
C. Error  
D. 3.0

**Q2.** What does the following SQL query return?

SELECT UPPER(SUBSTR('sqlselect', 1, 3));

A. sql  
B. SQL  
C. SEL  
D. Sql

**Q3.** Which of the following queries finds the 3rd highest salary from a table employees(salary)?

1. SELECT MAX(salary)

FROM employees

WHERE salary < (SELECT MAX(salary) FROM employees);

1. SELECT salary

FROM employees

ORDER BY salary DESC

LIMIT 1 OFFSET 2;

c. SELECT TOP 3 salary FROM employees ORDER BY salary DESC;

d. SELECT salary FROM employees ORDER BY salary ASC LIMIT 3;

**Q4.** Which clause is used to filter groups created by GROUP BY?

A. WHERE  
B. FILTER  
C. HAVING  
D. ORDER BY

**Q5.** Consider:

SELECT name, COUNT(\*)

FROM students

GROUP BY name

HAVING COUNT(\*) > 1;

What does this query return?

A. Names of all students  
B. Names of students appearing more than once  
C. Count of all students  
D. Syntax error

**Q6.** What is the default sorting order of ORDER BY clause?

A. ASC  
B. DESC  
C. RANDOM  
D. Depends on DBMS

**Q7.** Given:

SELECT name FROM students WHERE marks BETWEEN 80 AND 90;

This query is equivalent to:

A. marks >= 80 AND marks <= 90  
B. marks > 80 AND marks < 90  
C. marks = 80 OR marks = 90  
D. marks != 80 AND marks != 90

**Q8.** Which of these is a correct use of alias in SQL?

A. SELECT name AS "Student Name"  
B. SELECT name = "Student Name"  
C. SELECT name -> Student Name  
D. SELECT Student Name FROM students

**Q9.** Identify the issue:

SELECT name, AVG(score)

FROM students

WHERE score > 50

GROUP BY name

ORDER BY AVG(score);

A. GROUP BY not allowed after WHERE  
B. ORDER BY AVG(score) is not valid  
C. No issue  
D. Cannot use WHERE with GROUP BY

**Q10.** What does the following query return?

SELECT name FROM students WHERE name LIKE '\_a%';

A. Names starting with ‘a’  
B. Names ending with ‘a’  
C. Names where second character is ‘a’  
D. Names with only one character

**Q11.** Which query gives the number of unique departments?

A. SELECT COUNT(department) FROM employees;  
B. SELECT COUNT(\*) FROM employees;  
C. SELECT COUNT(DISTINCT department) FROM employees;  
D. SELECT DISTINCT COUNT(department) FROM employees;

**Q12.** Choose the correct query to get total salary by department:

1. SELECT department, SUM(salary) FROM employees;
2. SELECT department, SUM(salary) FROM employees GROUP BY department;
3. SELECT department, salary FROM employees GROUP BY department;
4. SELECT SUM(salary), department FROM employees;

**Q13.** Which of the following allows filtering data after aggregation?

A. WHERE  
B. HAVING  
C. GROUP BY  
D. JOIN

**Q14.** Which SQL statement finds employees who earn more than the average salary?

1. SELECT name FROM employees

WHERE salary > (SELECT AVG(salary) FROM employees);

1. SELECT name FROM employees

WHERE salary IN (SELECT AVG(salary) FROM employees);

1. SELECT name FROM employees

GROUP BY salary > AVG(salary);

1. SELECT name FROM employees

WHERE salary > AVG(salary);

**Q15.** Which statement returns duplicate entries?

A. SELECT DISTINCT name FROM students;  
B. SELECT name FROM students;  
C. SELECT UNIQUE name FROM students;  
D. SELECT ALL name FROM students;

**Q16.** What will be the output of:

SELECT COUNT(NULL) FROM students;

A. Total students  
B. 0  
C. Error  
D. NULL

**Q17.** Which is **not valid** SQL syntax?

A. SELECT \* FROM students ORDER BY 1;  
B. SELECT name FROM students ORDER BY 1;  
C. SELECT name, age FROM students ORDER BY age;  
D. SELECT name, age FROM students ORDER age;

**Q18.** What does SELECT NULLIF(5, 5); return?

A. 5  
B. 0  
C. NULL  
D. Error

**Q19.** Choose the best query to rename column “salary” to “monthly\_salary”:

A. SELECT salary monthly\_salary FROM employees;  
B. SELECT salary AS monthly\_salary FROM employees;  
C. SELECT salary -> monthly\_salary FROM employees;  
D. SELECT salary = monthly\_salary FROM employees;

**Q20.** The COALESCE(a, b, c) function returns:

A. First argument  
B. First non-null argument  
C. Last argument  
D. Count of nulls