Q1. Which SQL command is used to retrieve all columns from a table named `Students`?
A) GET * FROM Students;
B) SELECT ALL FROM Students;
C) SELECT * FROM Students;
D) RETRIEVE ALL FROM Students;

Q2. Which clause is used to filter rows in a SQL query?
A) ORDER BY
B) GROUP BY
C) WHERE
D) HAVING

Q3. Which keyword ensures unique values in a query result?
A) UNIQUE
B) DISTINCT
C) DIFFERENT
D) ONLY

Q4. Which clause sorts the query results?
A) WHERE
B) ORDER BY
C) GROUP BY
D) HAVING

Q5. Which aggregate function counts the number of rows?
A) SUM()
B) COUNT()
C) AVG()
D) MAX()

Q6. What will the following query return?

```
SELECT name FROM Students WHERE age > 21;
```

A) Names of students aged exactly 21
B) Names of students older than 21
C) Names of all students
D) Total number of students

Q7. Which of the following is a valid SQL query to retrieve student names in ascending order?
A) SELECT name FROM Students SORT BY name ASC;
B) SELECT name FROM Students ORDER BY name ASC;
C) SELECT name FROM Students ORDER ASC;
D) SELECT name SORT BY Students ASC;

Q8. Which of the following SQL commands retrieves the average value of a column `age`?
A) SELECT TOTAL(age) FROM Students;
B) SELECT AVG(age) FROM Students;
C) SELECT SUM(age)/COUNT(age) FROM Students;
D) SELECT MEAN(age) FROM Students;

Q9. What is the purpose of a subquery in SQL?
A) To define a new table
B) To perform a query inside another query
C) To modify data
D) To sort data

Q10. Which clause is used to filter results after applying GROUP BY?
A) WHERE
B) HAVING
C) ORDER BY
D) DISTINCT

MEDIUM QUESTIONS (Q11–Q25)

Q11. Which of the following is true about `GROUP BY`?
A) It filters individual rows
B) It groups rows with the same values in specified columns
C) It sorts the result set
D) It creates a temporary table

Q12. Which of the following SQL statements counts the total number of students per course?
A) `SELECT course_id, SUM(*) FROM Students GROUP BY course_id;`
B) `SELECT course_id, COUNT(*) FROM Students GROUP BY course_id;`
C) `SELECT course_id, COUNT(*) FROM Students WHERE course_id;`
D) `SELECT course_id, COUNT(*) FROM Students ORDER BY course_id;`

Q13. Which operator allows filtering rows based on a list of values?
A) IN
B) LIKE
C) BETWEEN
D) EXISTS

Q14. Which SQL query calculates the total fees per course?
A) `SELECT course_id, SUM(fee) AS TotalFee FROM Students GROUP BY course_id;`
B) `SELECT course_id, COUNT(fee) AS TotalFee FROM Students GROUP BY course_id;`
C) `SELECT course_id, AVG(fee) AS TotalFee FROM Students;`
D) `SELECT course_id, SUM(fee) FROM Students;`

Q15. Which SQL statement is correct to retrieve students whose name starts with 'A'?
A) `SELECT name FROM Students WHERE name LIKE 'A%';`
B) `SELECT name FROM Students WHERE name START 'A';`
C) `SELECT name FROM Students LIKE 'A%';`
D) `SELECT name FROM Students WHERE name BEGIN 'A';`

Q16. Which JOIN type returns only matching rows from both tables?
A) INNER JOIN
B) LEFT JOIN
C) RIGHT JOIN
D) FULL OUTER JOIN

Q17. Which clause would you use to retrieve the highest fee from the Students table?
A) MAX(fee)
B) TOP(fee)
C) HIGH(fee)
D) SUM(fee)

Q18. Which SQL command calculates the average age of students?
A) SELECT SUM(age)/COUNT(age) FROM Students;
B) SELECT AVG(age) FROM Students;
C) SELECT MEAN(age) FROM Students;
D) SELECT TOTAL(age)/COUNT(age) FROM Students;

Q19. Which clause filters records after aggregation?
A) WHERE
B) HAVING
C) ORDER BY
D) DISTINCT

Q20. Which SQL query retrieves students in course CS using a subquery?
A) `SELECT name FROM Students WHERE course_id = (SELECT course_id FROM Courses WHERE course_name='CS');`
B) `SELECT name FROM Students INNER JOIN Courses ON course_name='CS';`
C) `SELECT name FROM Students WHERE course_id IN 'CS';`
D) `SELECT name FROM Students USING course='CS';`

Q21. Which SQL query sorts students by age descending and fee ascending?
A) `ORDER BY age DESC, fee ASC`
B) `ORDER BY fee DESC, age ASC`
C) `SORT BY age DESC, fee ASC`
D) `ORDER BY age ASC, fee DESC`

Q22. Which aggregate function calculates total sum of a column?
A) SUM()
B) COUNT()
C) AVG()
D) MAX()

Q23. Which subquery depends on the outer query for its value?
A) Correlated subquery
B) Single-row subquery
C) Multi-row subquery
D) Non-correlated subquery

Q24. Which SQL statement removes duplicate rows in the result?
A) DISTINCT
B) UNIQUE
C) DIFFERENT
D) ONLY

Q25. Which SQL statement retrieves all students older than 20 and sorts them by age?
A) `SELECT * FROM Students WHERE age>20 SORT BY age;`
B) `SELECT * FROM Students FILTER age>20 ORDER BY age;`
C) `SELECT * FROM Students WHERE age>20 ORDER BY age;`
D) `SELECT * FROM Students AGE>20 ORDER BY age;`

HARD QUESTIONS (Q26–Q40)

Q26. Which query retrieves students whose fee is above the average of their course?
A) Correlated subquery using AVG in WHERE clause
B) Subquery in FROM clause
C) Subquery in SELECT clause without correlation
D) JOIN with Courses table only

Q27. Correct sequence of query execution in SQL:
A) SELECT → WHERE → GROUP BY → HAVING → ORDER BY
B) WHERE → SELECT → GROUP BY → HAVING → ORDER BY
C) SELECT → GROUP BY → HAVING → WHERE → ORDER BY
D) GROUP BY → WHERE → SELECT → HAVING → ORDER BY

Q28. Which query retrieves students in CS or IT using a subquery?
A) `SELECT name FROM Students WHERE course_id IN (SELECT course_id FROM Courses WHERE course_name IN ('CS','IT'));`
B) `SELECT name FROM Students WHERE course_id = 'CS' OR 'IT';`
C) `SELECT name FROM Students INNER JOIN Courses ON course_name='CS','IT';`
D) `SELECT name FROM Students USING course IN ('CS','IT');`

Q29. Which query counts students per course and only includes courses with more than one student?
A) `GROUP BY course_id HAVING COUNT(*)>1`
B) `WHERE COUNT(*)>1 GROUP BY course_id`
C) `HAVING COUNT(*)>1` without GROUP BY
D) `GROUP BY course_id WHERE COUNT(*)>1`

Q30. SQL statement to add a column `email` to the Students table:
A) `ALTER TABLE Students ADD COLUMN email VARCHAR(50);`
B) `INSERT COLUMN email INTO Students;`
C) `UPDATE TABLE Students ADD email VARCHAR(50);`
D) `MODIFY TABLE Students ADD email VARCHAR(50);`

Q31. Which JOIN returns all rows from both tables with NULLs if no match?
A) INNER JOIN
B) LEFT JOIN
C) RIGHT JOIN
D) FULL OUTER JOIN

Q32. Which statement retrieves second-highest salary from Employees table?
A) `SELECT MAX(salary) FROM Employees;`
B) `SELECT MAX(salary) FROM Employees WHERE salary<(SELECT MAX(salary) FROM Employees);`
C) `SELECT TOP 2 salary FROM Employees;`
D) `SELECT DISTINCT salary ORDER BY salary DESC LIMIT 1,1;`

Q33. Which query calculates total fees per course only if total fees > 9000?
A) `GROUP BY course_id HAVING SUM(fee)>9000`
B) `WHERE SUM(fee)>9000 GROUP BY course_id`
C) `SELECT SUM(fee)>9000 GROUP BY course_id`
D) `FILTER SUM(fee)>9000`

Q34. Which clause is applied last in SQL query execution?
A) WHERE
B) GROUP BY
C) HAVING
D) ORDER BY

Q35. Which query filters student groups after aggregation by course_id?
A) `GROUP BY course_id HAVING COUNT(*)>1`
B) `WHERE COUNT(*)>1 GROUP BY course_id`
C) `WHERE course_id GROUP BY COUNT(*)>1`
D) `HAVING COUNT(*)>1` without GROUP BY

Q36. Which SQL function returns the maximum value from a column?
A) MAX()
B) SUM()
C) AVG()
D) COUNT()

Q37. Which of the following removes all rows from a table but keeps the table structure?
A) DROP TABLE Students;
B) DELETE FROM Students;
C) TRUNCATE TABLE Students;
D) REMOVE ALL FROM Students;

Q38. How do you rename a column `age` to `student_age`?
A) `ALTER TABLE Students RENAME COLUMN age TO student_age;`
B) `RENAME COLUMN Students.age TO student_age;`
C) `MODIFY COLUMN age AS student_age;`
D) `UPDATE COLUMN age TO student_age;`

Q39. How to revoke all privileges from a MySQL user `pankaj`?
A) `REVOKE ALL PRIVILEGES ON *.* FROM 'pankaj'@'localhost';`
B) `REMOVE ALL PRIVILEGES FOR 'pankaj';`
C) `DENY ALL PRIVILEGES TO 'pankaj';`
D) `DELETE USER PRIVILEGES 'pankaj';`

Q40. Which command ensures new privileges take effect immediately?
A) APPLY PRIVILEGES;
B) REFRESH PRIVILEGES;
C) FLUSH PRIVILEGES;
D) UPDATE PRIVILEGES;