1. Which query finds the nth highest salary in SQL (portable way)?

A) SELECT salary FROM Employee ORDER BY salary DESC LIMIT n-1,1;

B) SELECT salary FROM (SELECT salary, DENSE_RANK() OVER (ORDER BY salary DESC) r FROM Employee) t WHERE r=n;

C) SELECT TOP n salary FROM Employee ORDER BY salary DESC;

D) SELECT salary FROM Employee;

Answer: B

2. Which SQL function ranks rows without gaps in ranking sequence?

A) RANK()

B) DENSE_RANK()

C) ROW_NUMBER()

D) NTILE()

Answer: B

3. Which SQL function ranks rows with possible gaps?

A) RANK()

B) DENSE_RANK()

C) ROW_NUMBER()

D) NTILE()

Answer: A

4. Which SQL function assigns unique number to each row in order?

A) RANK()

B) DENSE_RANK()

C) ROW_NUMBER()

D) NTILE()

Answer: C

5. Which SQL function divides rows into n buckets?

A) RANK()

B) DENSE_RANK()

C) ROW_NUMBER()

D) NTILE()

Answer: D

6. Which query finds employees with salary greater than average of their department?

A) SELECT * FROM Employee WHERE salary > AVG(salary) GROUP BY dept_id;

B) SELECT * FROM Employee e WHERE salary > (SELECT AVG(salary) FROM Employee WHERE dept_id=e.dept_id);

C) SELECT * FROM Employee WHERE salary > ALL(AVG(salary));

D) SELECT * FROM Employee WHERE salary > (AVG(salary));

Answer: B

7. Which SQL feature allows recursive queries?

A) WITH RECURSIVE

B) CONNECT BY

C) CTE

D) All of these depending on DBMS

Answer: D

8. Which query finds all managers in a hierarchy recursively?

A) WITH RECURSIVE cte AS (SELECT id, manager_id FROM Employee UNION ALL SELECT e.id, e.manager_id FROM Employee e JOIN cte c ON e.id=c.manager_id) SELECT * FROM cte;

B) SELECT manager FROM Employee;

C) SELECT * FROM Employee WHERE manager_id IS NOT NULL;

D) None

Answer: A

9. Which query finds employees earning more than their department's average salary using window functions?

A) SELECT * FROM (SELECT e.*, AVG(salary) OVER (PARTITION BY dept_id) avg_sal FROM Employee e) t WHERE salary > avg_sal;

B) SELECT * FROM Employee WHERE salary > AVG(salary);

C) SELECT * FROM Employee GROUP BY dept_id HAVING salary > AVG(salary);

D) SELECT * FROM Employee;

Answer: A

10. Which SQL feature allows pivoting rows to columns?

A) PIVOT

B) CASE

C) GROUP BY

D) UNPIVOT

Answer: A

11. Which SQL feature allows converting columns to rows?

A) PIVOT

B) UNPIVOT

C) TRANSPOSE

D) CASE

Answer: B

12. Which query finds duplicate rows in a table?

A) SELECT col, COUNT(*) FROM Table GROUP BY col HAVING COUNT(*)>1;

B) SELECT DISTINCT * FROM Table;

C) SELECT * FROM Table WHERE col IS DUPLICATE;

D) SELECT * FROM Table;

Answer: A

13. Which query deletes duplicate rows keeping only one?

A) DELETE FROM Employee WHERE id NOT IN (SELECT MIN(id) FROM Employee GROUP BY salary);

B) DELETE DUPLICATE FROM Employee;

C) TRUNCATE Employee;

D) None

Answer: A

14. Which query finds employees who earn more than at least one employee in another department?

A) SELECT * FROM Employee e1 WHERE salary > ANY(SELECT salary FROM Employee e2 WHERE e1.dept_id<>e2.dept_id);

B) SELECT * FROM Employee WHERE salary > MIN(salary);

C) SELECT * FROM Employee WHERE salary > SOME(salary);

D) SELECT * FROM Employee;

Answer: A

15. Which query finds employees who earn more than every employee in another department?

A) SELECT * FROM Employee e1 WHERE salary > ALL(SELECT salary FROM Employee e2 WHERE e1.dept_id<>e2.dept_id);

B) SELECT * FROM Employee WHERE salary > MAX(salary);

C) SELECT * FROM Employee WHERE salary > ANY(salary);

D) None

Answer: A

16. Which query finds the employee(s) with maximum salary using correlated subquery?

A) SELECT * FROM Employee e WHERE salary=(SELECT MAX(salary) FROM Employee);

B) SELECT * FROM Employee e WHERE NOT EXISTS(SELECT 1 FROM Employee e2 WHERE e2.salary>e.salary);

C) SELECT TOP 1 * FROM Employee ORDER BY salary DESC;

D) All of these

Answer: D

17. Which SQL clause improves query readability for nested subqueries?

A) WITH (CTE)

B) GROUP BY

C) HAVING

D) WINDOW

Answer: A

18. Which query lists departments with more than average number of employees?

A) SELECT dept_id FROM Employee GROUP BY dept_id HAVING COUNT(*)>(SELECT AVG(cnt) FROM (SELECT COUNT(*) cnt FROM Employee GROUP BY dept_id) t);

B) SELECT dept_id FROM Employee WHERE COUNT(*)>AVG(COUNT(*));

C) SELECT dept_id FROM Employee;

D) None

Answer: A

19. Which query returns the top 10% of salaries?

A) SELECT salary FROM Employee ORDER BY salary DESC FETCH FIRST 10 PERCENT ROWS ONLY;

B) SELECT TOP 10 PERCENT salary FROM Employee ORDER BY salary DESC;

C) SELECT * FROM (SELECT salary, NTILE(10) OVER (ORDER BY salary DESC) t FROM Employee) WHERE t=1;

D) All of these depending on DBMS

Answer: D

20. Which query uses analytic function to calculate cumulative salary?

A) SELECT name, SUM(salary) OVER (ORDER BY salary) cum_sal FROM Employee;

B) SELECT name, CUME_SUM(salary) FROM Employee;

C) SELECT SUM(salary) FROM Employee GROUP BY salary;

D) SELECT * FROM Employee;

Answer: A

21. Which SQL function computes running percentage?

A) PERCENT_RANK()

B) NTILE()

C) CUME_DIST()

D) Both A and C

Answer: D

22. Which query finds the median salary?

A) SELECT PERCENTILE_CONT(0.5) WITHIN GROUP (ORDER BY salary) FROM Employee;

B) SELECT MEDIAN(salary) FROM Employee;

C) SELECT salary FROM Employee WHERE ROWNUM=50;

D) None

Answer: A

23. Which query finds employees who earn above department median salary?

A) SELECT * FROM (SELECT e.*, PERCENTILE_CONT(0.5) WITHIN GROUP (ORDER BY salary) OVER (PARTITION BY dept_id) med FROM Employee e) t WHERE salary>med;

B) SELECT * FROM Employee WHERE salary>MEDIAN(salary);

C) SELECT * FROM Employee;

D) None

Answer: A

24. Which SQL feature helps detect performance issues?

A) EXPLAIN PLAN

B) ANALYZE

C) EXPLAIN QUERY

D) DEBUG

Answer: A

25. Which join may generate same output as CROSS JOIN in wrong condition?

A) INNER JOIN without ON

B) LEFT JOIN

C) RIGHT JOIN

D) FULL JOIN

Answer: A

26. Which query finds departments having employees with all possible job roles?

A) SELECT dept_id FROM Employee GROUP BY dept_id HAVING COUNT(DISTINCT job)=(SELECT COUNT(DISTINCT job) FROM Employee);

B) SELECT dept_id FROM Employee WHERE job=ALL(job);

C) SELECT * FROM Employee;

D) None

Answer: A

27. Which query finds employees whose salary is greater than their manager?

A) SELECT e.* FROM Employee e JOIN Employee m ON e.manager_id=m.id WHERE e.salary>m.salary;

B) SELECT * FROM Employee WHERE salary>manager.salary;

C) SELECT * FROM Employee WHERE salary>salary;

D) None

Answer: A

28. Which query finds employees who manage more than 5 employees?

A) SELECT manager_id FROM Employee GROUP BY manager_id HAVING COUNT(*)>5;

B) SELECT * FROM Employee WHERE manager_id>5;

C) SELECT manager_id FROM Employee;

D) None

Answer: A

29. Which query finds employee(s) with maximum salary in each department?

A) SELECT * FROM Employee e WHERE salary=(SELECT MAX(salary) FROM Employee e2 WHERE e2.dept_id=e.dept_id);

B) SELECT dept_id, MAX(salary) FROM Employee GROUP BY dept_id;

C) SELECT * FROM Employee;

D) Both A and B

Answer: D

30. Which query finds departments where every employee earns above company average?

A) SELECT dept_id FROM Employee e GROUP BY dept_id HAVING MIN(salary)>(SELECT AVG(salary) FROM Employee);

B) SELECT dept_id FROM Employee WHERE salary>AVG(salary);

C) SELECT * FROM Employee;

D) None

Answer: A

31. Which query finds employees whose salary is same as their department’s maximum?

A) SELECT * FROM Employee e WHERE salary=(SELECT MAX(salary) FROM Employee e2 WHERE e2.dept_id=e.dept_id);

B) SELECT * FROM Employee WHERE salary=MAX(salary);

C) SELECT * FROM Employee WHERE salary=ANY(salary);

D) None

Answer: A

32. Which query finds employees who joined earliest in each department?

A) SELECT * FROM Employee e WHERE join_date=(SELECT MIN(join_date) FROM Employee e2 WHERE e2.dept_id=e.dept_id);

B) SELECT dept_id, MIN(join_date) FROM Employee GROUP BY dept_id;

C) SELECT * FROM Employee;

D) Both A and B

Answer: D

33. Which query finds top 3 salaries in each department?

A) SELECT * FROM (SELECT e.*, DENSE_RANK() OVER (PARTITION BY dept_id ORDER BY salary DESC) r FROM Employee e) t WHERE r<=3;

B) SELECT TOP 3 salary FROM Employee GROUP BY dept_id;

C) SELECT * FROM Employee WHERE salary=3;

D) None

Answer: A

34. Which query finds departments where max salary is greater than average company salary?

A) SELECT dept_id FROM Employee GROUP BY dept_id HAVING MAX(salary)>(SELECT AVG(salary) FROM Employee);

B) SELECT dept_id FROM Employee WHERE MAX(salary)>AVG(salary);

C) SELECT dept_id FROM Employee;

D) None

Answer: A

35. Which query finds employees earning within top 5% salaries?

A) SELECT * FROM (SELECT e.*, CUME_DIST() OVER (ORDER BY salary DESC) c FROM Employee e) t WHERE c<=0.05;

B) SELECT * FROM Employee WHERE salary>ALL(salary);

C) SELECT * FROM Employee;

D) None

Answer: A

36. Which query finds the difference between highest and lowest salary?

A) SELECT MAX(salary)-MIN(salary) FROM Employee;

B) SELECT salary FROM Employee WHERE salary=MAX-MIN;

C) SELECT salary FROM Employee;

D) None

Answer: A

37. Which query finds employees who joined in last 6 months?

A) SELECT * FROM Employee WHERE join_date>=CURRENT_DATE-INTERVAL '6' MONTH;

B) SELECT * FROM Employee WHERE join_date>SYSDATE-6;

C) Both A and B depending on DBMS

D) None

Answer: C

38. Which query finds employees working in more than one department?

A) SELECT id FROM Employee GROUP BY id HAVING COUNT(DISTINCT dept_id)>1;

B) SELECT * FROM Employee WHERE dept_id>1;

C) SELECT * FROM Employee;

D) None

Answer: A

39. Which query finds employees with salary equal to department average?

A) SELECT * FROM Employee e WHERE salary=(SELECT AVG(salary) FROM Employee e2 WHERE e2.dept_id=e.dept_id);

B) SELECT * FROM Employee WHERE salary=AVG(salary);

C) SELECT * FROM Employee;

D) None

Answer: A

40. Which query finds employees who joined before their manager?

A) SELECT e.* FROM Employee e JOIN Employee m ON e.manager_id=m.id WHERE e.join_date<m.join_date;

B) SELECT * FROM Employee WHERE join_date<manager;

C) SELECT * FROM Employee;

D) None

Answer: A

41. Which SQL feature is used to create materialized query results?

A) MATERIALIZED VIEW

B) PERSISTENT QUERY

C) STORED SELECT

D) CACHE

Answer: A

42. Which query finds departments where every employee earns more than 50,000?

A) SELECT dept_id FROM Employee GROUP BY dept_id HAVING MIN(salary)>50000;

B) SELECT dept_id FROM Employee WHERE salary>50000;

C) SELECT * FROM Employee;

D) None

Answer: A

43. Which query finds employees who do not manage anyone?

A) SELECT * FROM Employee e WHERE NOT EXISTS(SELECT 1 FROM Employee m WHERE m.manager_id=e.id);

B) SELECT * FROM Employee WHERE manager=NULL;

C) SELECT * FROM Employee;

D) None

Answer: A

44. Which query finds departments with maximum average salary?

A) SELECT dept_id FROM Employee GROUP BY dept_id ORDER BY AVG(salary) DESC FETCH FIRST 1 ROW ONLY;

B) SELECT dept_id FROM Employee WHERE salary>AVG(salary);

C) SELECT * FROM Employee;

D) None

Answer: A

45. Which query finds the department(s) with highest employee count?

A) SELECT dept_id FROM Employee GROUP BY dept_id HAVING COUNT(*)=(SELECT MAX(c) FROM (SELECT COUNT(*) c FROM Employee GROUP BY dept_id) t);

B) SELECT dept_id FROM Employee WHERE COUNT(*)=MAX;

C) SELECT * FROM Employee;

D) None

Answer: A

46. Which SQL feature is used to rank rows randomly?

A) RANDOM()

B) ORDER BY RAND()

C) NEWID()

D) Both B and C depending on DBMS

Answer: D

47. Which query finds employees earning same as at least one employee in another department?

A) SELECT * FROM Employee e1 WHERE EXISTS(SELECT 1 FROM Employee e2 WHERE e1.salary=e2.salary AND e1.dept_id<>e2.dept_id);

B) SELECT * FROM Employee WHERE salary=ANY(salary);

C) SELECT * FROM Employee;

D) None

Answer: A

48. Which query finds average salary per job role across departments?

A) SELECT job, AVG(salary) FROM Employee GROUP BY job;

B) SELECT job, AVG(salary) FROM Employee GROUP BY job, dept_id;

C) SELECT job, salary FROM Employee;

D) None

Answer: A

49. Which query finds employees earning more than department average and less than department maximum?

A) SELECT * FROM Employee e WHERE salary>(SELECT AVG(salary) FROM Employee e2 WHERE e2.dept_id=e.dept_id) AND salary<(SELECT MAX(salary) FROM Employee e3 WHERE e3.dept_id=e.dept_id);

B) SELECT * FROM Employee WHERE salary=AVG(salary);

C) SELECT * FROM Employee;

D) None

Answer: A

50. Which query finds top earner in each department using window function?

A) SELECT * FROM (SELECT e.*, RANK() OVER (PARTITION BY dept_id ORDER BY salary DESC) r FROM Employee e) t WHERE r=1;

B) SELECT * FROM Employee WHERE salary=MAX(salary);

C) SELECT * FROM Employee;

D) None

Answer: A