NAMES:Uwabera Giramata sheilaa

ID:28741 GROUP:E

CORRECTION OF DBMS:

1. SELECT CONCAT(first\_name, ' ', last\_name) AS full\_name FROM employees;

2. SELECT LOWER(CONCAT(first\_name, ' ', last\_name)) AS full\_name\_lowercase FROM employees;

3. SELECT SUBSTRING(first\_name, 1, 3) AS name\_prefix FROM employees;

OR SELECT LEFT(FIRST\_NAME,3) AS name\_prefix FROM employees;

4. SELECT REPLACE(email, '@company.com', '@org.com') AS new\_email FROM employees;

5. SELECT TRIM(' padded string ') AS trimmed\_string;

6. SELECT LENGTH(CONCAT(first\_name, ' ', last\_name)) AS name\_length FROM employees;

7. SELECT INSTR(email, '@') AS at\_position FROM employees;

8. SELECT CASE

WHEN gender = 'M' THEN CONCAT('Mr. ', first\_name, ' ', last\_name)

WHEN gender = 'F' THEN CONCAT('Ms. ', first\_name, ' ', last\_name)

ELSE CONCAT(first\_name, ' ', last\_name)

END AS titled\_name FROM employees;

9. SELECT UPPER(project\_name) AS project\_name\_upper from projects;

10. SELECT REPLACE(project\_name, '-', '') AS cleaned\_project\_name from projects;

11. SELECT CONCAT('Emp: ', e.first\_name, ' ', e.last\_name, ' (', d.department\_name, ')') AS employee\_label from employees e. JOIN departments d ON e.department\_id = d.department\_id;

12. SELECT email, LENGTH(email) AS email\_length from employees;

13. SELECT SUBSTRING(email, 1, INSTR(email, '@') - 1) AS email\_last\_name from employees;

14. SELECT CONCAT(UPPER(last\_name), ', ', first\_name) AS formatted\_name from employees;

15. SELECT CONCAT(e.first\_name, ' ', e.last\_name CASE

WHEN p.end\_date IS NULL OR p.end\_date > CURDATE() THEN ' (Active)' ELSE ''

END) AS employee\_status

FROM employees e

LEFT JOIN employee\_projects ep ON e.employee\_id = ep.employee\_id

LEFT JOIN projects p ON ep.project\_id = p.project\_id;

16. SELECT ROUND(salary, 0) AS rounded\_salary from employees;

17. SELECT salary from employees WHERE MOD(salary, 2) = 0;

18. SELECT project\_name, DATEDIFF(end\_date, start\_date) AS duration\_days

from projects WHERE end\_date IS NOT NULL;

19. SELECT ABS(e1.salary - e2.salary) AS salary\_difference

from employees e1

CROSS JOIN employees e2

WHERE e1.employee\_id = 101 AND e2.employee\_id = 102;

20. SELECT salary \* POWER(1.1, 1) AS increased\_salary from employees;

21. SELECT FLOOR(RAND() \* 1000) AS test\_id;

22. SELECT salary, CEIL(salary) AS ceil\_salary, FLOOR(salary) AS floor\_salary from employees;

23.

24. SELECT salary

CASE

WHEN salary >= 5000 THEN 'High'

WHEN salary >= 3000 THEN 'Medium'

ELSE 'Low'

END AS salary\_category from employees;

25. SELECT salary, LENGTH(CAST(FLOOR(salary) AS CHAR)) AS salary\_digits From employees;

26. SELECT CURRENT\_DATE AS today\_date;

27. SELECT first\_name, last\_name, DATEDIFF(CURDATE(), hire\_date) AS days\_worked

from employees;

28. SELECT first\_name, last\_name fromemployees

WHERE YEAR(hire\_date) = YEAR(CURDATE());

29. SELECT NOW() AS current\_datetime;

30. SELECT hire\_date,YEAR(hire\_date) AS hire\_year,MONTH(hire\_date) AS hire\_month,

DAY(hire\_date) AS hire\_day from employees;

31. SELECT first\_name, last\_name from employees

WHERE hire\_date < '2020-01-01';

32. SELECT project\_name

from projects

WHERE end\_date IS NOT NULL

AND DATEDIFF(CURDATE(), end\_date) <= 30

AND DATEDIFF(CURDATE(), end\_date) >= 0;

33. SELECT project\_name, DATEDIFF(end\_date, start\_date) AS total\_days

from projects

WHERE end\_date IS NOT NULL;

34. CASE MONTH('2025-07-23') WHEN 7 THEN 'July'ELSE 'Unknown' END,' ',

DAY('2025-07-23'), ', ',YEAR('2025-07-23')) AS formatted\_date;

35. SELECT project\_name, CASE

WHEN end\_date IS NULL THEN 'Ongoing' ELSE 'Completed'END AS project\_status

from projects;

36. SELECT salary, CASE WHEN salary >= 5000 THEN 'High'

WHEN salary >= 3000 THEN 'Medium'

ELSE 'Low'

END AS salary\_label from employees;

37. SELECT COALESCE(email, 'No Email') AS email\_status

From employees;

38. SELECT first\_name, last\_name,CASE WHEN hire\_date < '2015-01-01' THEN 'Veteran'

ELSE 'Regular'

END AS employee\_status

from employees;

39. SELECT COALESCE(salary, 3000) AS adjusted\_salary from employees;

40. SELECT d.department\_name,

CASE

WHEN d.department\_name = 'Information Technology' THEN 'IT'

WHEN d.department\_name = 'Human Resources' THEN 'HR'

ELSE 'Other'

END AS department\_category

From departments d;

41. SELECT e.first\_name, e.last\_name,

CASE

WHEN ep.project\_id IS NULL THEN 'Unassigned'

ELSE 'Assigned'

END AS project\_status from employees e

LEFT JOIN employee\_projects ep ON e.employee\_id = ep.employee\_id;

42. SELECT salary,

CASE

WHEN salary > 6000 THEN 'Band A'

WHEN salary > 4000 THEN 'Band B'

ELSE 'Band C'

END AS tax\_band from employees ; SELECT project\_name,

43.

44. SELECT project\_name,

CASE

WHEN DATEDIFF(end\_date, start\_date) IS NULL THEN 'Ongoing'

WHEN DATEDIFF(end\_date, start\_date) > 365 THEN

CASE

WHEN DATEDIFF(end\_date, start\_date) > 730 THEN 'Very Long'

ELSE 'Long' END ELSE 'Short' END AS duration\_label from projects;

45. SELECT CONCAT(COALESCE(first\_name, 'Unknown'), ' ', COALESCE(last\_name, 'Unknown')) AS full\_name from employees;

46. SELECT CONCAT(first\_name, ' ', last\_name) AS full\_name,

CASE

WHEN LENGTH(CONCAT(first\_name, ' ', last\_name)) > 10 THEN 'Long Name'

ELSE 'Short Name'

END AS name\_length\_label from employees;

47. SELECT email,

CASE

WHEN UPPER(email) LIKE '%TEST%' THEN 'Dummy Account'

ELSE 'Regular Account'

END AS account\_type from employees;

48. SELECT first\_name, last\_name,

CASE

WHEN YEAR(hire\_date) < 2015 THEN 'Senior'

WHEN YEAR(hire\_date) < 2020 THEN 'Mid-level'

ELSE 'Junior'

END AS seniority from employees;

49. SELECT salary,

CASE

WHEN salary >= 5000 THEN '10% Increment'

WHEN salary >= 3000 THEN '7% Increment'

ELSE '5% Increment'

END AS increment\_range from employees;

50. SELECT first\_name, last\_name,

CASE

WHEN MONTH(hire\_date) = MONTH(CURDATE()) THEN 'This Month'

ELSE 'Not This Month'

END AS anniversary\_status from employees;