NAMES: GATESI UWASE KEVINE

ID : 27997

GROUP : E

CORRECTION OF DBMS:

Q1. Concatenate first and last name as full_name.

ANSWER: SELECT EMP_FNAME || ' ' || EMP_LNAME AS full_name FROM employees1;

Q2. Convert all employee names to lowercase.

ANSWER: SELECT LOWER(EMP_FNAME), LOWER(EMP_LNAME) FROM employees1;

Q3. Extract first 3 letters of employee's first name

ANSWER: SELECT SUBSTR(EMP_FNAME, 1, 3) AS first_3_letters

FROM EMPLOYEES1;

Q4: Replace '@company.com' in email with '@org.com'

ANSWER: SELECT REPLACE(EMAIL, '@company.com', '@org.com') AS updated_email

FROM EMPLOYEES1;

Q5. Trim spaces from a padded string

ANSWER: SELECT TRIM(EMP_FNAME) AS trimmed_fname

FROM EMPLOYEES1;

Q6. Count characters in employee's full name

ANSWER: SELECT LENGTH(EMP_FNAME || EMP_LNAME) AS name_length

FROM EMPLOYEES1;

Q7. **Find position of '@' in email using INSTR**

ANSWER: SELECT INSTR(EMAIL, '@') AS at_position

FROM EMPLOYEES1;

Q8. Add 'Mr.' or 'Ms.' before names based on gender

ANSWER: SELECT

CASE WHEN GENDER = 'M' THEN 'Mr. ' ELSE 'Ms. ' END || EMP_FNAME || ' ' || EMP_LNAME AS titled_name

FROM EMPLOYEES1;

Q9: Format project names to uppercase

ANSWER: SELECT UPPER(PROJECT_NAME) AS project_upper

FROM EMPLOYEES1;

Q10. Remove any dashes from project names

ANSWER: SELECT REPLACE(PROJECT_NAME, '-', '') AS project_no_dash

FROM EMPLOYEES1;

Q11. Create a label like “Emp: John Doe (HR)”

ANSWER: SELECT 'Emp: ' || EMP_FNAME || ' ' || EMP_LNAME || ' (' || DEPARTMENT || ')' AS label

FROM EMPLOYEES1;

Q12. **Check email length for each employee**

```
ANSWER: SELECT EMAIL, LENGTH(EMAIL) AS email_length
FROM EMPLOYEES1;
```

Q13. Extract last name only from email (before @)

```
ANSWER: SELECT SUBSTR(EMAIL, 1, INSTR(EMAIL, '@') - 1) AS
last_name_part
FROM EMPLOYEES1;
```

Q14. **Format: “LASTNAME, Firstname” using UPPER and CONCAT**

ANSWER: SELECT UPPER(EMP_LNAME) || ', ' || EMP_FNAME AS formatted_name

FROM EMPLOYEES1;

Q15. Add “(Active)” next to employee names who have current projects

ANSWER: SELECT EMP_FNAME || ' ' || EMP_LNAME ||

CASE WHEN PROJECT_END_DATE IS NULL OR PROJECT_END_DATE > SYSDATE THEN ' (Active)' ELSE '' END AS emp_status

FROM EMPLOYEES1;

Numeric Function Exercises (16–25)

Q16. Round salary to nearest whole number

ANSWER: SELECT ROUND(BASIC_SALARY) AS rounded_salary

FROM EMPLOYEES1;

Q17. Show only even salaries using MOD

ANSWER: SELECT BASIC_SALARY

FROM EMPLOYEES1

WHERE MOD(BASIC_SALARY, 2) = 0;

Q18. Show difference between two project end/start dates using DATEDIFF

ANSWER: SELECT PROJECT_END_DATE - PROJECT_START_DATE AS days_difference

FROM EMPLOYEES1;

Q19. Show absolute difference in salaries between two employees

ANSWER: SELECT ABS(

(SELECT BASIC_SALARY FROM EMPLOYEES1 WHERE EMP_ID = 101) -

(SELECT BASIC_SALARY FROM EMPLOYEES1 WHERE EMP_ID = 102)

) AS salary_diff

FROM dual;

Q20. Raise salary by 10% using POWER

ANSWER: SELECT BASIC_SALARY * POWER(1.1, 1) AS increased_salary

FROM EMPLOYEES1;

Q21. **Generate a random number for testing IDs**

```
SELECT ROUND(DBMS_RANDOM.VALUE(1, 10000)) AS random_id
FROM dual;
```

Q22. Use CEIL and FLOOR on a floating salary

ANSWER: SELECT CEIL(BASIC_SALARY) AS ceiling_salary,

FLOOR(BASIC_SALARY) AS floor_salary

FROM EMPLOYEES1;

Q23. Use LENGTH() on phone numbers

ANSWER: SELECT PHONE_NUMBER, LENGTH(PHONE_NUMBER) AS phone_length

FROM EMPLOYEES1;

Q24. Count digits in salary amount

ANSWER: SELECT BASIC_SALARY,

LENGTH(TO_CHAR(TRUNC(BASIC_SALARY))) AS digit_count

FROM EMPLOYEES1;

Q25. **Categorize salary: High/Medium/Low using CASE**

ANSWER: SELECT BASIC_SALARY,

CASE

WHEN BASIC_SALARY > 7000 THEN 'High'

WHEN BASIC_SALARY BETWEEN 4000 AND 7000 THEN 'Medium'

ELSE 'Low'

END AS salary_category

FROM EMPLOYEES1;

Date/Time Function Exercises (26–35)

Q26. Show today’s date

ANSWER: SELECT SYSDATE AS today_date FROM dual;

Q27. Calculate how many days an employee has worked

ANSWER: SELECT EMP_ID, SYSDATE - HIRE_DATE AS days_worked

FROM EMPLOYEES1;

Q28. Show employees hired in the current year

ANSWER: SELECT *

FROM EMPLOYEES1

WHERE EXTRACT(YEAR FROM HIRE_DATE) = EXTRACT(YEAR FROM SYSDATE);

Q29. Display current date and time

ANSWER: SELECT SYSTIMESTAMP AS current_datetime FROM dual;

Q30. Extract year, month, day from hire_date

ANSWER: SELECT EXTRACT(YEAR FROM HIRE_DATE) AS hire_year,

EXTRACT(MONTH FROM HIRE_DATE) AS hire_month,

EXTRACT(DAY FROM HIRE_DATE) AS hire_day

FROM EMPLOYEES1;

Q31. Show employees hired before 2020

ANSWER: SELECT *

FROM EMPLOYEES1

WHERE HIRE_DATE < TO_DATE('01-JAN-2020', 'DD-MON-YYYY');

Q32. **List projects that ended in last 30 days**

ANSWER: SELECT *

FROM EMPLOYEES1

WHERE PROJECT_END_DATE BETWEEN SYSDATE - 30 AND SYSDATE;

Q33. Calculate total days between project start and end

ANSWER: SELECT PROJECT_START_DATE, PROJECT_END_DATE,

PROJECT_END_DATE - PROJECT_START_DATE AS total_days

FROM EMPLOYEES1;

Q34. Format date ‘2025-07-23’ to ‘July 23, 2025’

ANSWER: SELECT TO_CHAR(TO_DATE('2025-07-23', 'YYYY-MM-DD'), 'Month DD, YYYY') AS formatted_date

FROM dual;

Q35. Add CASE: if project active (end_date IS NULL), show ‘Ongoing’

ANSWER: SELECT PROJECT_NAME,

CASE WHEN PROJECT_END_DATE IS NULL THEN 'Ongoing'

ELSE TO_CHAR(PROJECT_END_DATE, 'DD-MON-YYYY')

END AS project_status

FROM EMPLOYEES1;

Conditional Function Exercises (36–50)

Q36. Use CASE to label salaries

ANSWER: SELECT BASIC_SALARY,

CASE

WHEN BASIC_SALARY > 7000 THEN 'High Salary'

WHEN BASIC_SALARY BETWEEN 4000 AND 7000 THEN 'Medium Salary'

ELSE 'Low Salary'

END AS salary_label

FROM EMPLOYEES1;

Q37. Use COALESCE to show ‘No Email’ if email is NULL

ANSWER: SELECT COALESCE(EMAIL, 'No Email') AS email_status

FROM EMPLOYEES1;

Q38. CASE: If hire_date < 2015, mark as ‘Veteran’

ANSWER: SELECT EMP_ID,

CASE WHEN EXTRACT(YEAR FROM HIRE_DATE) < 2015 THEN 'Veteran' ELSE 'Newbie' END AS status

FROM EMPLOYEES1;

Q39. **If salary is NULL, default to 3000 using COALESCE**

```
ANSWER: SELECT COALESCE(BASIC_SALARY, 3000) AS
salary_with_default
FROM EMPLOYEES1;
Q40. CASE: Categorize departments (IT, HR, Other)
ANSWER: SELECT DEPARTMENT,
  CASE
    WHEN DEPARTMENT = 'IT' THEN 'IT'
    WHEN DEPARTMENT = 'HR' THEN 'HR'
    ELSE 'Other'
  END AS dept_category
FROM EMPLOYEES1;
```

Q41. **CASE: If employee has no project, mark as ‘Unassigned’**

ANSWER: SELECT EMP_FNAME, EMP_LNAME,

CASE WHEN PROJECT_NAME IS NULL THEN 'Unassigned' ELSE PROJECT_NAME END AS project_status

FROM EMPLOYEES1;

Q42. Show tax band based on salary

ANSWER: SELECT BASIC_SALARY,

CASE

WHEN BASIC_SALARY > 8000 THEN 'High Tax'

WHEN BASIC_SALARY BETWEEN 5000 AND 8000 THEN 'Medium Tax'

ELSE 'Low Tax'

END AS tax_band

FROM EMPLOYEES1;

Q43. Nested CASE to label project duration

ANSWER: SELECT PROJECT_NAME,

CASE

WHEN PROJECT_END_DATE IS NULL THEN 'Ongoing'

ELSE

CASE

WHEN PROJECT_END_DATE - PROJECT_START_DATE > 365 THEN 'Long Term'

ELSE 'Short Term'

END

END AS project_duration

FROM EMPLOYEES1;

Q44. CASE with MOD to show even/odd salary IDs

ANSWER: SELECT EMP_ID,

CASE WHEN MOD(EMP_ID, 2) = 0 THEN 'Even ID' ELSE 'Odd ID' END AS id_type

FROM EMPLOYEES1;

Q45. Combine COALESCE + CONCAT for fallback names

ANSWER: SELECT COALESCE(EMP_FNAME, 'Unknown') || ' ' || COALESCE(EMP_LNAME, 'Unknown') AS full_name

FROM EMPLOYEES1;

Q46. CASE with LENGTH(): if name length > 10, label “Long Name”

ANSWER: SELECT EMP_FNAME || ' ' || EMP_LNAME AS full_name,

CASE WHEN LENGTH(EMP_FNAME || EMP_LNAME) > 10 THEN 'Long Name' ELSE 'Short Name' END AS name_length_label

FROM EMPLOYEES1;

Q47. CASE + UPPER(): if email has ‘TEST’, mark as dummy account

ANSWER: SELECT EMAIL,

CASE WHEN UPPER(EMAIL) LIKE '%TEST%' THEN 'Dummy Account' ELSE 'Valid Account' END AS account_type

FROM EMPLOYEES1;

Q48. Show seniority based on hire year

ANSWER: SELECT EMP_ID,

CASE WHEN EXTRACT(YEAR FROM HIRE_DATE) < 2015 THEN 'Senior' ELSE 'Junior' END AS seniority

FROM EMPLOYEES1;

Q49. **CASE to determine salary increment range**

ANSWER: SELECT BASIC_SALARY,

CASE

WHEN BASIC_SALARY < 4000 THEN 'Low Increment'

WHEN BASIC_SALARY BETWEEN 4000 AND 7000 THEN 'Medium Increment'

ELSE 'High Increment'

END AS increment_range

FROM EMPLOYEES1;

Q50. CASE with CURDATE() to determine anniversary month

ANSWER: SELECT EMP_ID,

CASE

WHEN EXTRACT(MONTH FROM HIRE_DATE) = EXTRACT(MONTH FROM SYSDATE) THEN 'Anniversary Month'

ELSE 'Not Anniversary Month'

```
  END AS anniversary_status
FROM EMPLOYEES1;


END
```