IT Skill Test GIC Myanmar

Duration: 30 Minutes *Total Questions: 8*

---

1. You are working on a report that displays employee salaries. Some employees have NULL values for their commission. You want to display 'No Commission' instead of NULL. Which SQL query would you use?

A. SELECT employee_id, salary, COALESCE(commission, 'No Commission') FROM employees;

B. SELECT employee_id, salary, NVL(commission, 'No Commission') FROM employees;

C. SELECT employee_id, salary, NULLIF(commission, 'No Commission') FROM employees;

D. SELECT employee_id, salary, CASE WHEN commission IS NULL THEN 'No Commission' ELSE commission END FROM employees;

2. In a database tracking product inventory, you need to compare the current stock level with the minimum required stock. If they are the same, you want to return NULL, otherwise return the current stock. Which function would you use?

A. NVL(current_stock, min_stock)

B. COALESCE(current_stock, min_stock)

C. NULLIF(current_stock, min_stock)

D. CASE WHEN current_stock = min_stock THEN NULL ELSE current_stock END

3. You're working with a table that contains both numeric and string data types. You need to concatenate a string column with a numeric column. Which of the following approaches is most appropriate?

A. Use the || operator directly without any conversion

B. Use TO_CHAR to convert the numeric column before concatenation

C. Use TO_NUMBER to convert the string column before concatenation

D. Use CAST to convert both columns to NVARCHAR2 before concatenation

4. In a financial application, you're working with a column that sometimes contains the string 'N/A' instead of NULL for missing values. You want to treat 'N/A' the same as NULL in your calculations. Which function would be most appropriate to use?

A. NVL(column, NULL)

B. NULLIF(column, 'N/A')

C. COALESCE(column, NULL)

D. CASE WHEN column = 'N/A' THEN NULL ELSE column END

5. You have a table with a DATE column and a VARCHAR2 column containing dates in the format 'YYYY-MM-DD'. You need to compare these two columns. What's the most efficient way to do this?

A. TO_DATE(varchar2_column, 'YYYY-MM-DD') = date_column

B. TO_CHAR(date_column, 'YYYY-MM-DD') = varchar2_column

C. CAST(varchar2_column AS DATE) = date_column

D. date_column = varchar2_column

6. In a database tracking product sales, you have a column for discount percentage that can be NULL (no discount), 0 (0% discount), or a positive number. You want to display 'Full Price' for NULL, 'No Discount' for 0, and the actual percentage for positive numbers. Which SQL construct would you use?

A. NVL2(discount, CASE WHEN discount = 0 THEN 'No Discount' ELSE TO_CHAR(discount) || '%' END, 'Full Price')

B. COALESCE(CASE WHEN discount = 0 THEN 'No Discount' ELSE TO_CHAR(discount) || '%' END, 'Full Price')

C. NULLIF(CASE WHEN discount = 0 THEN 'No Discount' ELSE TO_CHAR(discount) || '%' END, 'Full Price')

D. NVL(CASE WHEN discount = 0 THEN 'No Discount' ELSE TO_CHAR(discount) || '%' END, 'Full Price')

7. You're working on a query that involves mathematical operations on columns that might contain NULL values. You want to ensure that any NULL value is treated as 0 in these calculations, but without modifying the original data. Which approach would be most efficient?

A. Use NVL on each column in every calculation

B. Use COALESCE on each column in every calculation

C. Create a view that replaces NULL with 0 and use this view in calculations

D. Use NVL2 to handle both NULL and non-NULL cases in each calculation

8. You have a table with a numeric column that sometimes contains negative values. For reporting purposes, you want to display negative values as positive, zero values as 'Zero', and NULL values as 'N/A'. Which SQL construct would you use?

A. CASE WHEN column < 0 THEN ABS(column) WHEN column = 0 THEN 'Zero' ELSE NVL(TO_CHAR(column), 'N/A') END

B. NVL2(column, CASE WHEN column < 0 THEN ABS(column) WHEN column = 0 THEN 'Zero' ELSE TO_CHAR(column) END, 'N/A')

C. COALESCE(CASE WHEN column < 0 THEN ABS(column) WHEN column = 0 THEN 'Zero' ELSE TO_CHAR(column) END, 'N/A')

D. NULLIF(CASE WHEN column < 0 THEN ABS(column) WHEN column = 0 THEN 'Zero' ELSE TO_CHAR(column) END, 'N/A')