

## SQL 応用 ～研修受講後テスト～

## ■問題1【SQLのコーディング】

可読性の説明で誤っているものを選んで下さい。

- コメントを残すことでデバッグが容易になる。
- 表名、列名に日本語名は使用できない。
- コーディングの規約を作成し、担当者間で共有すべきである。
- 表や列の役割・意味が判別できるような名前を定義すべきである。

## ■問題2【条件分岐】

CASE 式の記述について、一番相ふさわしいものを選んで下さい。

- CASE WHEN no = 1 THEN 'YES'  
END
- CASE WHEN no THEN 'YES'  
ELSE NULL  
END
- CASE WHEN no = 1 THEN 'YES'  
ELSE NULL  
END
- CASE WHEN no = 1 THEN 'YES'  
ELSE NULL

## ■問題3【条件分岐】

条件分岐関数の説明で、誤っているものを選んで下さい。

- COALESCE 関数は列順に NULL の存在を確認できる。
- COALESCE 関数は NVL 関数と同じように、NULL を他の値に置き換えることができる。
- NULLIF 関数は、NULL を他の値に置き換えることができる関数である。
- NULLIF 関数を使用すると、ゼロ除算を回避できる。

## ■問題4【条件分岐】

MERGE 文の説明で正しいものを全てを選んで下さい。

- ON 句で、比較する双方の表の列を指定する。
- UPDATE 対象がターゲット表、INSERT の対象となるのがソース表である。
- MERGE INTO 句でビューを指定することはできない。
- ターゲット表に対する DML 文では、表名を明示的に指定しない。

## ■問題5【値の集計・分析】

列値が一意かどうか確認する HAVING 句の条件で正しいものを選んで下さい。

- count(\*) > count(列名)
- count(\*) = max(列名)
- min(abs(列名)) = 0
- count(DISTINCT 列名) = count(列名)

**■問題6【値の集計・分析】**

分析関数の説明で誤っているものを選んで下さい。

- 分析関数はももとの行数を保ったままグループに対する分析処理を行う。
- PARTITION BY 句を省略した場合、検索結果全体が1つのグループとして扱われる。
- 集約関数を分析関数として使用する場合、関数名の引数には列名を指定せず()のまま指定する。
- DENSE\_RANK 関数を使用する場合、同意が複数あっても順位は飛ばない。

**■問題7【値の比較】**

相関副問合せの説明で誤っているものを選んで下さい。

- 相関副問合せを実行するには、副問合せ内で主問合せを参照する。
- 相関副問合せはパフォーマンスが問題になるため、通常は使用しない。
- 相関副問合せの処理を分析関数で代替できる場合は、分析関数を使用した方が良い。
- 相関副問合せを実行する際は、副問合せと主問合せは異なる表から検索する。

**■問題8【値の比較】**

EXISTS 述語の説明で誤っているものを選んで下さい。

- EXISTS 述語に続く副問合せの選択リストでは、どのような列を指定してもエラーにはならない。
- EXISTS 述語は値の存在有無のみの確認のため、主問合せ側の WHERE 句には比較対象列は指定しない。
- EXISTS 述語使用時は、内部的に副問合せの結果全てと値の比較を行っている。
- EXISTS 述語と CASE 式を組み合わせることで、値の存在有無で処理を分岐できる。

**■問題9【NULL の取り扱い】**

データベースから NULL を排除する方法について誤っているものを選んで下さい。

- 列に NOT NULL 制約を定義する。
- 列に DEFAULT 句を定義する。
- 列内の NULL を他の値に置き換える。
- COALESCE 関数で NULL を他の値に置き換える。

**■問題10【NULL の取り扱い】**

NULL 取り扱いの説明で正しいものを全て選んで下さい。

- NOT EXISTS 述語に続く副問合せの結果に NULL が含まれていると正しい結果が表示されない。
- ALL 述語に続く副問合せの結果に NULL が含まれていると正しい結果が表示されない。
- ソートを行うと、NULL は一番小さな値として扱われる。
- 外部結合を行うと NULL が発生するため、COALESCE 関数などで NULL を他の値に置き換えることを検討する。