

Oracle.1z0-071.v2026-07-01.q224

試験コード : 1z0-071
試験名称 : Oracle Database SQL
認証ベンダー : Oracle
無料問題の数 : 224
バージョン : v2026-07-01
ページの閲覧量 : 104
問題集の閲覧量 : 2475

<https://www.jpnsiken.com/shiken/Oracle.1z0-071.v2026-07-01.q224.html>

質問: 1

round、truncate、mod関数について正しいのはどれですか？

- A. ROUND(MOD(25,3),-1) は無効です
- B. ROUND(MOD(25,3),-1) と TRUNC(MOD(25,3),-1) はどちらも有効で、同じ結果になります。
- C. ROUND(MOD(25,3),-1) と TRUNC(MOD(25,3),-1) はどちらも有効で、異なる結果になります。
- D. TRUNC(MOD(25,3),-1) は無効です。

正解: [\(正解を表示します\)](#)

ROUND関数とTRUNC関数はどちらも数値に適用でき、MOD関数は除算の余りを返す関数です。ROUND関数は数値を指定された小数点以下の桁数に丸めます。小数点以下の桁数は正、ゼロ、負のいずれでも構いません。TRUNC関数は数値を指定された小数点以下の桁数に切り捨てます。

ROUND(MOD(25,3),-1) は、MOD(25,3) の結果である 1 を十の位に丸めて 0 にします。

TRUNC(MOD(25,3),-1) は、MOD(25,3) の結果である 1 を十の位で切り捨て、結果も 0 になります。

どちらも有効ですが、この特定の場合、余り 1 を十の位 (1) に四捨五入または切り捨てると 0 になるため、同じ結果になります。

質問: 2

複数テーブル挿入ステートメントについて、正しい記述は次のうちどれですか？ (2つ選択)

- A. 彼らは常にサブクエリを使用します。
- B. 無条件の INSERT ALL ステートメントでは、ソース テーブルとターゲット テーブルの両方で列数が同じである必要があります。
- C. 条件付き INSERT ALL ステートメントは、ソース行を集約して単一のテーブルに行を挿入します。
- D. ソーステーブルの1行をターゲットテーブルの複数の行に変換できます。
- E. 条件付き挿入ファーストステートメントは常に単一のテーブルに行を挿入します。

正解: [A,D \(コメントを發表する\)](#)

質問: 3

PRODUCT_DETAILS テーブルの説明を確認してください。

Name	Null?	Type
PRODUCT_ID	NOT NULL	NUMBER (2)
PRODUCT_NAME	NOT NULL	VARCHAR2 (25)
PRODUCT_PRICE		NUMBER (8, 2)
EXPIRY_DATE		DATE

正しい記述は次のうちどれですか？ (2つ選択)

- A. PRODUCT_PRICE には値が格納されていなくても、算術式で使用できます。
- B. PRODUCT_ID には PRIMARY KEY 制約を割り当てることができます。
- C. EXPIRY_DATE は算術式では使用できません。
- D. EXPIRY_DATE には、日付が割り当てられていない場合、デフォルトで SYSDATE が含まれます。
- E. PRODUCT_PRICE には、値が割り当てられていない場合、デフォルトでゼロの値が格納されます。
- F. PRODUCT_NAME には重複する値を含めることはできません。

正解: **B,C** ([コメントを发表する](#))

PRODUCT_DETAILS テーブルの属性に関して :

- A. この記述は誤りです。PRODUCT_PRICE に値がない場合 (つまり、NULL の場合)、NULL を含む演算の結果が NULL になるため、算術式で直接使用することはできません。
- B. この記述は正しい。PRODUCT_ID は NOT NULL 制約を持ち、型は NUMBER であるため、主キーとして適格である。主キーでは、すべての値が一意であり、NULL であってはならない。
- C. この記述は正しいです。EXPIRY_DATE は DATE 型であり、日付を扱う日付関数を使用しない限り、算術式で直接使用することはできません。
- D. この記述は誤りです。デフォルトでは、DATE 型の列には、列定義で DEFAULT キーワードを使用して明示的に割り当てない限り、デフォルト値は設定されません。
- E. この記述は誤りです。デフォルトでは、数値列には、DEFAULT キーワードで指定しない限り、デフォルト値は設定されません。
- F. この記述は誤りです。PRODUCT_NAME に重複値が含まれていないという記述は、それが UNIQUE キーまたは PRIMARY KEY であれば正しいですが、提供されたテーブルの説明にはそのような制約は示されていません。

参照 :

Oracle のデータ型に関するドキュメント : https://docs.oracle.com/database/121/SQLRF/sql_elements001.htm#SQLRF00213 Oracle の制約に関するドキュメント : <https://docs.oracle.com/database/121/SQLRF/clauses002.htm#SQLRF52271>

質問: 4

INVOICE テーブルには、データ型が NUMBER の QTY_SOLD 列と、データ型が DATE の INVOICE_DATE 列があり、NLS_DATE_FORMAT は DD-MON-RR に設定されています。クエリ式におけるこれらの列を含むデータ型変換について、正しいのはどれですか? (2つ選択)

- A. invoice_date > '01-02-2019': 暗黙的な型変換を使用しています
- B. qty_sold = '05549821 ': 明示的な変換が必要です
- C. CONCAT(qty_sold, invoice_date): 明示的な変換が必要です
- D. qty_sold が '101' から '110' の間である場合: 暗黙的な型変換を使用します
- E. invoice_date = '15-march-2019': 暗黙的な型変換を使用しています

正解: ([正解を表示します](#))

データ型の変換と SQL 式におけるリテラルの扱いに関する記述は、Oracle SQL における暗黙的および明示的なデータ変換の理解を伴います。

ステートメント A は、invoice_date > '01-02-2019' が、NLS_DATE_FORMAT 設定に基づいて文字列リテラルを日付型に暗黙的に変換することを含み、フォーマットが一致することを前提としているため、正しいです。

ステートメント E は正しいです。なぜなら、A と同様に、invoice_date = '15-march-2019' は暗黙的な変換を伴い、文字列は Oracle NLS_DATE_FORMAT または想定されるデフォルトの日付形式に従って日付型に自動的に変換されるからです。

記述 B、C、D には、不正確または誤解を招く情報が含まれています。

B (qty_sold = '05549821') は誤解を招く可能性があり、数値コンテキストにおける先頭のゼロは通常明示的な変換を必要としませんが、スペースの存在は数値変換ではなくトリミングが必要であることを示唆している可能性があります。

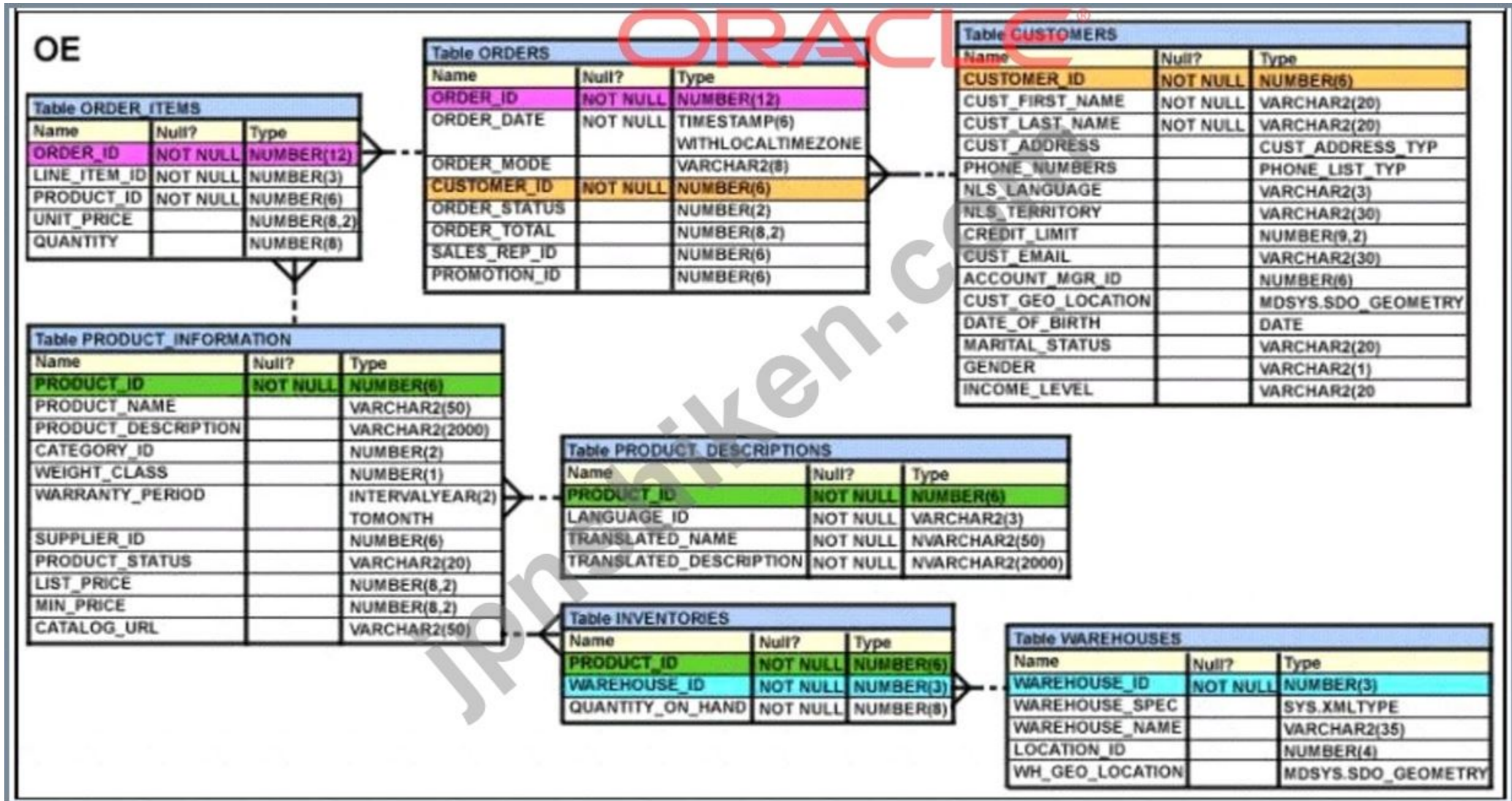
C (CONCAT(qty_sold, invoice_date)) は、CONCAT が文字列型を期待するため、明示的な変換が必要になります。したがって、数値と日付の値は連結する前に明示的に文字列に変換する必要があります。

D (qty_sold BETWEEN '101' AND '110') は暗黙的な型変換を使用しており、qty_sold が数値型の場合、文字列リテラル '101' と '110' は暗黙的に数値に変換されます。

質問: 5

展示資料を参照し、ORDERSテーブルとORDER_ITEMSテーブルの構造を確認してください。

ORDER_IDはORDERSテーブルの主キーです。また、ORDER_ITEMSテーブルの外部キーでもあり、ON DELETE CASCADEオプションを使用して作成されています。どのDELETE文が正常に実行されるでしょうか？



- A. o.order_id = i.order_id の注文 o、注文アイテムを削除します。
- B. 注文合計が 1000 未満の注文を削除します。
- C. DELETE FROM orders WHERE (SELECT order_id FROM order_items);
- D. DELETE order_id FROM orders WHERE order_total < 1000;

正解: (正解を表示します)

質問: 6

次の記述を評価してください。

```
INSERT ALL
  WHEN order_total < 10000 THEN
    INTO small_orders
  WHEN order_total > 10000 AND order_total < 20000 THEN
    INTO medium_orders
  WHEN order_total > 20000 AND order_total < 20000 THEN
    INTO large_orders
SELECT order_id, order_total, customer_id
FROM orders;
```

INSERT文のサブクエリによって返される行の評価に関して、正しい記述はどれですか？

- A. 他のWHEN句の評価結果に関係なく、3つのWHEN句すべてによって評価されます。
- B. これらは最初のWHEN句によって評価されます。条件が真の場合、その行は後続のWHEN句によって評価されます。
- C. 最初のWHEN句によって評価されます。条件が偽の場合、その行は後続のWHEN句によって評価されます。
- D. INSERT文は、WHEN句のいずれも真でない場合をサポートするELSE句が存在しないため、エラーになります。

正解: [A \(コメントを发表する\)](#)

説明

参考文献 :

<http://psoug.org/definition/WHEN.htm>

質問: 7

リレーショナルデータベースにおけるテーブル間の関係を定義することについて、正しい記述を3つ挙げてください。

- A. 外部キー列はNULL値を許容します。
- B. 一意キー列はNULL値を許容します
- C. 主キー列はNULL値を許容します。
- D. すべての主キーまたは一意キーの値は、対応する外部キーの値を参照する必要があります。
- E. すべての外部キー値は、対応する主キーまたは一意キー値を参照する必要があります。

正解: [\(正解を表示します\)](#)

A) 正解。外部キー制約は null を許容できます。これにより、行が別のテーブルへのリンクを持たない可能性が許容されます。B. 不正解。一意キー制約は、制約が複数の列に定義されていない限り、複数の行が null 値を持つことを許可しません。C. 不正解。主キー列は、テーブル全体で NOT NULL かつ一意である必要があります。D. 不正解。主キーまたは一意キーの値は外部キーを参照しません。外部キーが主キーまたは一意キーを参照します。E. 正解。外部キー制約は、各外部キーの値が関連テーブルの主キーまたは一意キーの値と一致することを要求することにより、参照整合性を強制します。

これらはリレーショナルデータベースにおける標準的なルールであり、Oracle Database Concepts Guideで確認できます。

質問: 8

展示資料をご覧になり、顧客表の構造を調べてください。

CUSTOMERS

Name	Null?	Type
CUSTOMER_ID	NOT NULL	NUMBER (6)
CUST_NAME		VARCHAR2 (20)
CUST_EMAIL		VARCHAR2 (30)
INCOME_LEVEL		VARCHAR2 (20)

CUSTOMER_VUは、CUSTOMERS_BR1テーブルを基にしたビューであり、CUSTOMERSテーブルと同じ構造を持っています。顧客情報は、顧客に関する最新情報を反映するように更新する必要があります。以下のMERGEステートメントのエラーは何ですか？

```

MERGE INTO customers c
  USING customer_vu cv
  ON (c.customer_id = cv.customer_id)
  WHEN MATCHED THEN
    UPDATE SET
      c.customer_id = cv.customer_id,
      c.cust_name = cv.cust_name,
      c.cust_email = cv.cust_email,
      c.income_level = cv.income_level
  WHEN NOT MATCHED THEN
    INSERT VALUES (cv.customer_id, cv.cust_name, cv.cust_email, cv.income_level)
    WHERE cv.income_level >100000;
  
```

- A. WHERE句はINSERT文では使用できません。
- B. INTO句がコマンド内で誤った位置にあります。
- C. CUSTOMER_VU はデータソースとして使用できません。
- D. CUSTOMER_ID列は更新できません。

正解: [\(正解を表示します\)](#)

質問: 9

非等結合に関して正しい記述は次のうちどれですか？ (2つ選択)

- A. ON句を使用できます。
- B. USING句を使用できます。
- C. SQL:1999 に準拠した ANSI 結合構文を使用する必要があります。
- D. テーブルエイリアスを使用する必要があります。
- E. Oracleの結合構文を使用できます。

正解: [A,E \(コメントを公表する\)](#)

非等価結合とは、結合条件が等価性以外の基準に基づいている結合のことです。Oracle SQLでは、さまざまな句を使用して非等価結合を実行できます。

- A. ON句を使用できます：確かに、ON句は非等結合を含む、結合のあらゆる条件を指定できます。等結合に限定されません。
- E. Oracleの結合構文を使用できます: WHERE句を使用して結合条件を指定する従来のOracleの結合構文は、非等価結合を含むすべてのタイプの結合に使用できます。

参照：

Oracle Database SQL言語リファレンス12c、特にON句やOracle独自の結合構文を含む結合句に関するセクションを参照してください。

質問: 10

ORACLE_DATAPUMPアクセスドライバを使用して実行できる操作を3つ選択してください。

- A. 外部テーブルのディレクトリオブジェクトを作成する
- B. データベース内のテーブルからデータを読み込み、外部テーブルに挿入する
- C. 外部テーブルに対してDMLステートメントを実行する
- D. 外部テーブルからデータを読み込み、データベース内のテーブルにロードする
- E. 外部テーブルからデータをクエリする
- F. フラットファイル用のディレクトリオブジェクトを作成する

正解: ([正解を表示します](#))

質問: 11

正常に実行される以下のSQL文を調べてください。

```
CREATE TABLE emp
(emp_no NUMBER(2) CONSTRAINT emp_emp_no_pk PRIMARY KEY
ename VARCHAR2(15),
salary NUMBER(8,2),
mgr_no NUMBER(2));
```

```
ALTER TABLE emp ADD CONSTRAINT emp_mgr_fk
FOREIGN KEY (mgr_no)
REFERENCES emp(emp_no)
ON DELETE SET NULL;
```

```
ALTER TABLE emp
DISABLE CONSTRAINT emp_emp_no_pk
CASCADE;
```

```
ALTER TABLE emp
ENABLE CONSTRAINT emp_emp_no_pk;
```

実行後、正しい記述は次のうちどれか？（2選択）

- A. 外部キー制約が有効になり、即時適用されます。
- B. 主キー制約が有効になり、即時適用されます。
- C. 主キー制約が有効になり、延期されます。
- D. 外部キー制約が無効になります。
- E. 外部キー制約が有効になり、延期されます。

正解: ([正解を表示します](#))

質問: 12

正常に実行される以下のSQL文を調べてください。

```
CREATE TABLE emp
(emp_no NUMBER(2) CONSTRAINT emp_emp_no_pk PRIMARY KEY
ename VARCHAR2(15),
salary NUMBER(8,2),
mgr_no NUMBER(2));
```

```
ALTER TABLE emp ADD CONSTRAINT emp_mgr_fk
FOREIGN KEY (mgr_no)
REFERENCES emp(emp_no)
ON DELETE SET NULL;
```

```
ALTER TABLE emp
DISABLE CONSTRAINT emp_emp_no_pk
CASCADE;
```

```
ALTER TABLE emp
ENABLE CONSTRAINT emp_emp_no_pk;
```

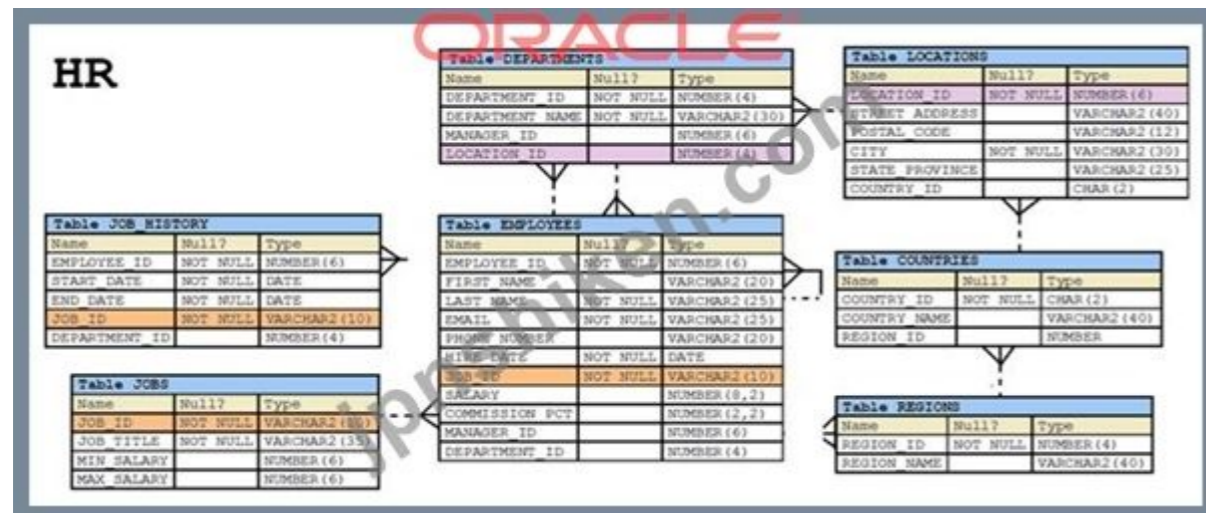
実行後、正しい記述は次のうちどれか？（2選択）

- A. 外部キー制約が無効になります。
- B. 主キー制約が有効になり、即時適用されます。
- C. 主キー制約が有効になり、延期されます。
- D. 外部キー制約が有効になり、即時適用されます。
- E. 外部キー制約が有効になり、延期されます。

正解: (正解を表示します)

質問: 13

展示資料をご覧になり、「従業員」表の説明をご確認ください。



次のSQL文を評価してください。

```
SELECT first_name, employee_id, NEXT_DAY(ADD_MONTHS(hire_date, 6), 1) "Review" FROM employees;
```

このクエリは、従業員のFIRST_NAME、EMPLOYEE_ID、およびレビュー日を取得するために作成されました。レビュー日は、採用後6か月経過後の最初の月曜日です。セッションでは、NLS_TERRITORYパラメータがAMERICAに設定されています。

この質問に関して、正しい記述はどれですか？

- A. クエリが実行され、目的の出力が得られます。
- B. NEXT_DAY 関数は引数として文字列を受け取るため、クエリは実行されません。
- C. 日付関数はネストできないため、クエリは実行されません。

D. クエリは実行されますが、出力には日曜日のレビュー日が表示されます。

正解: [\(正解を表示します\)](#)

質問: 14

SALESテーブルには、データ型がNUMBERの列PROD_IDとQUANTITY_SOLDがあります。

正常に実行されるクエリはどれですか？ 2つ選択してください。)

A. カウント(*) > 10;

SELECT COUNT(prod_id) FROM sales GROUP BY prod_id WHERE quantity_sold > 55000;

B. SELECT prod_id FROM sales WHERE quantity_sold > 55000 GROUP BY prod_id HAVING

C. prod_id;

SELECT prod_id FROM sales WHERE quantity_sold > 55000 AND COUNT(*) > 10 GROUP

D. BY prod_id HAVING COUNT(*) > 10;

SELECT COUNT (prod_id) FROM sales WHERE quantity_sold > 55000 GROUP BY

E. カウント(*) > 10;

SELECT prod_id FROM sales WHERE quantity_sold > 55000 AND COUNT(*) > 10 GROUP

正解: [\(正解を表示します\)](#)

質問: 15

この SELECT ステートメントを調べて、その出力を確認するために展示を参照してください。 2つ選択してください。) SELECT constraints_name, constraints_type, search_condition, r_constraints_name, delete_rule, status, FROM user_constraints WHERE table_name = 'ORDERS'; 出力について正しい記述はどれですか?

A. DELETE_RULE列は、親テーブルから対応する行が削除されたときに、子テーブルの関連行が持つべき状態を示します。

B. 2列目の 'r' はチェック制約を示します。

C. STATUS列は、テーブルが現在使用されているかどうかを示します。

D. R_CONSTRAINT_NAME 列には、制約の別名が含まれています。

正解: [\(正解を表示します\)](#)

質問: 16

以下の2つのクエリとその出力結果を確認してください。

SELECT deptno, dname FROM dept;

Deptno	Dname
10	account ing
20	research
30	sales
40	operations

SELECT ename, job, deptno FROM emp ORDER BY deptno;

Ename	job	deptno
CLARK	MANAGER	10
KING	PRESIDENT	10
MILLER	CLERK	10
JONES	MANAGER	20

それでは、このクエリを調べてみましょう。

```
SELECT ename, dname
FROM emp CROSS JOIN dept WHERE job = 'MANAGER&
AND dept.deptno IN (10, 20);
```

- A. 64
- B. 6
- C. 3
- D. 12

正解: ([正解を表示します](#))

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%ディスカウント、特別な割引コード: **JPNshiken**」

質問: 17

展示物をご覧になり、プロモーション表の構造を調べてください。

Table PROMOTIONS		
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(6)
PROMO_NAME	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY_ID	NOT NULL	NUMBER
PROMO_CATEGORY	NOT NULL	VARCHAR2(30)
PROMO_CATEGORY_ID	NOT NULL	NUMBER
PROMO_COST	NOT NULL	NUMBER(10,2)
PROMO_BEGIN_DATE	NOT NULL	DATE
PROMO_END_DATE	NOT NULL	DATE

「インターネット」カテゴリの最後のプロモーション以降に開始されたすべてのプロモーションについて、プロモーション名と開始日を表示するレポートを作成する必要があります。

どのクエリを実行すれば、必要な出力が得られますか？

- A. `SELECT promo_name, promo_begin_date FROM promotions WHERE promo_begin_date > ALL (SELECT promo_begin_date FROM promotions WHERE promo_category = 'INTERNET');`
- B. `SELECT promo_name, promo_begin_date FROM promotions WHERE promo_begin_date > ANY (SELECT promo_begin_date FROM promotions WHERE promo_category = 'INTERNET');`
- C. `SELECT promo_name, promo_begin_date FROM promotions WHERE promo_begin_date IN (SELECT promo_begin_date FROM promotions WHERE promo_category = 'INTERNET');`
- D. `SELECT promo_name, promo_begin_date FROM promotions WHERE promo_begin_date > ALL (SELECT MAX (promo_begin_date) FROM promotions) AND promo_category = 'INTERNET';`

正解: **A** ([コメントを發表する](#))

質問: 18

複雑な相関サブクエリにおけるWITH句の使用に関して、正しい記述は次のうちどれですか？（3選択）

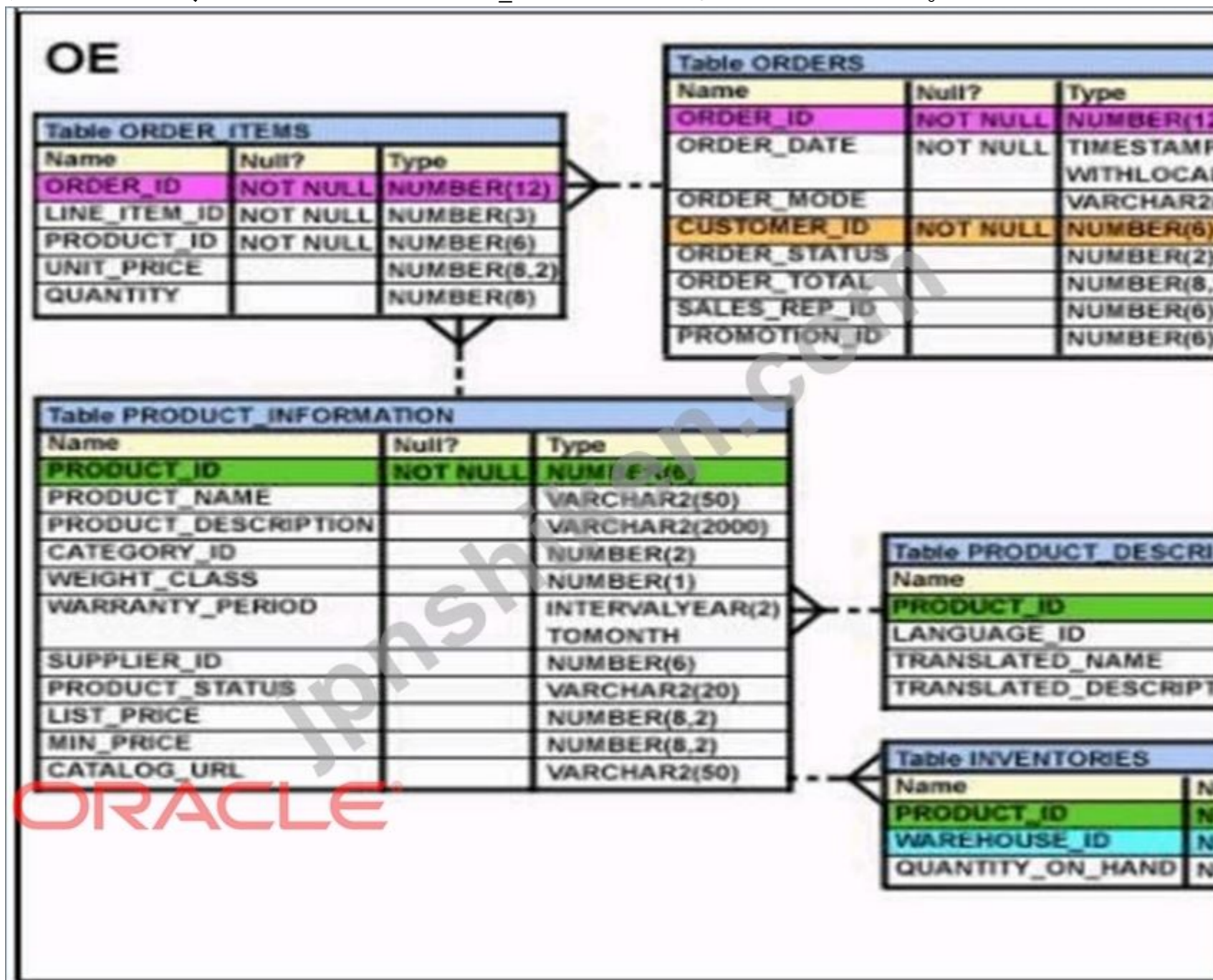
（3選択してください。）

- A. WITH句には複数のクエリを含めることができます。
- B. クエリブロック名とテーブル名が同じ場合、テーブル名が優先されます。
- C. SELECT句でのみ使用できます。
- D. WITH句内のクエリ名は、WITH句内の他のクエリブロックだけでなく、メインのクエリブロックからも参照可能です。

正解: [\(正解を表示します\)](#)

質問: 19

展示内容を確認し、ORDERSテーブルとORDER_ITEMSテーブルの構造を調べてください。



各注文について、注文ID、注文日、および商品総数を表示するビューを作成する必要があります。

どのCREATEVIEWステートメントを使用すれば、ビューを正常に作成できますか？

A. CREATE OR REPLACE VIEW ord_vu (order_id, order_date)

AS SELECT o.order_id, o.order_date, COUNT (i.line_item_id)

「アイテム数」

FROM orders o JOIN order_items i

ON (o.order_id = i.order_id)

o.order_id、o.order_dateでグループ化します。

B. ビュー ord_vu の作成または置換

AS SELECT o.order_id, o.order_date, COUNT (i.line_item_id)

「アイテム数」

FROM orders o JOIN order_items i

ON (o.order_id = i.order_id)

o.order_id、o.order_dateでグループ化します。

C. ビュー ord_vu の作成または置換

AS SELECT o.order_id, o.order_date, COUNT (i.line_item_id) ||

「アイテム数」

FROM orders o JOIN order_items i

ON (o.order_id = i.order_id)

o.order_id、o.order_dateでグループ化

チェックオプション付き。

D. ビュー ord_vu の作成または置換

AS SELECT o.order_id, o.order_date, COUNT (i.line_item_id)

FROM orders o JOIN order_items i

ON (o.order_id = i.order_id)

o.order_id、o.order_dateでグループ化します。

正解: **B** ([コメントを發表する](#))

質問: 20

ORDERSテーブルとORDER_ITEMSテーブルの所有者であるユーザーSCOTTは、以下のGRANTコマンドを発行します。

すべて許可する

注文、注文アイテム

一般向け

上記の記述にはどのような修正が必要ですか？

A. PUBLICは特定のユーザー名に置き換える必要があります。

B. ALLは具体的な権限のリストに置き換える必要があります。

C. WITH GRANT OPTIONs をステートメントに追加する必要があります。

D. ORDERSテーブルとORDER_ITEMSテーブルには、それぞれ別のGRANTステートメントが必要です。

正解: **D** ([コメントを發表する](#))

説明／参考資料 :

参考文献 :

<http://docs.oracle.com/javadb/10.8.3.0/ref/rrefsqjgrant.html>

質問: 21

以下のCREATE TABLEコマンドを評価してください。

```
CREATE TABLE order_item
(order_id NUMBER (3),
item-id NUMBER (2),
qty NUMBER (4),
CONSTRAINT ord_itm_id_pk
PRIMARY KEY (order_id, item_id)
USING INDEX
(CREATE INDEX ord_itm_idx
ON order_item (order_id, item_id)));
```

上記のSQL文に関して、正しい記述はどれですか？

- A. 正常に実行され、ORD_ITM_IDX インデックスのみが作成されます。
- B. CREATE TABLE コマンドでは USING INDEX 句が許可されていないため、エラーが発生します。
- C. 正常に実行され、ORD_ITM_IDX と ORD_ITM_ID_PK という 2 つのインデックスが作成されます。
- D. 複合プライマリーでは USING INDEX 句を使用できないため、エラーが発生します。

正解: ([正解を表示します](#))

質問: 22

stores テーブルには、行が挿入された日付を格納する、データ型が date の start_date という列があります。

開始日が過去 25 か月以内である行の詳細のみを表示したい場合、どの WHERE 句を使用できますか？

- A. WHERE MONTHS_BETWEEN (SYSDATE, start_date) <= 25
- B. TO_NUMBER (start_date - SYSDATE) <= 25 の条件を満たすもの
- C. WHERE ADD_MONTHS (start_date, 25) <= SYSDATE
- D. WHERE MONTHS_BETWEEN (start_date, SYSDATE) <= 25

正解: ([正解を表示します](#))

質問: 23

DEPARTMENT_DETAILS を作成するために使用されたコマンドを調べ、

コース詳細 :

```
SQL>CREATE TABLE DEPARTMENT_DETAILS
```

(部門ID番号主キー、

部門名VARCHAR2(50)、

HODVARCHAR2(50));

```
SQL>CREATE TABLE COURSE_DETAILS
```

(COURSE_IDNUMBER PRIMARY KEY、
コース名VARCHAR2(50)、
DEPARTMENT_IDVARCHAR2(50));

すべての部署IDと、それらに割り当てられている可能性のあるコースIDのリストを作成したいのですね。
どのSQL文を使用すべきですか？

- A. SELECT d.department_id, c.course_id FROM department_details d LEFT OUTER JOIN course_details c ON (d.department_id=c. department_id);
- B. SELECT d.department_id, c.course_id FROM course_details c LEFT OUTER JOIN department_details d ON (c.department_id=d.department_id);
- C. SELECT d.department_id, c.course_id FROM department_details d RIGHT OUTER JOIN course_details c ON (d.department_id=c.department_id);
- D. SELECT d.department_id, c.course_id FROM department_details d RIGHT OUTER JOIN course_details c ON (c.department_id=d.department_id);

正解: ([正解を表示します](#))

質問: 24

展示物をご覧になり、コスト表とプロモーション表の構造を調べてください。

次のSQL文を評価してください。

```
SQL> SELECT prod_id FROM costs  
WHERE promo_id IN (SELECT promo_id FROM promotions  
WHERE promo_cost < ALL  
(SELECT MAX(promo_cost) FROM promotions  
GROUP BY (promo_end_date-  
プロモーション開始日)));
```

上記のSQL文を実行すると、どのような結果になるでしょうか？

- A. 同じ期間内で最も価格の低いプロモーションの製品IDを表示します。
- B. 同じ期間内で最もコストの高いプロモーションの製品IDを表示します。
- C. 同じ期間内で最もコストが低いプロモーションの製品IDを表示します。
- D. 最も低価格なプロモーションの製品IDを表示します。

正解: ([正解を表示します](#))

質問: 25

サブクエリに関して正しい記述を3つ選びなさい。

- A. サブクエリは、SELECT ステートメントの FROM 句で使用できます。
- B. サブクエリの WHERE 句では、AND、OR、NOT などの論理演算子を使用できません。
- C. ORDER BY句はサブクエリで使用できます。
- D. サブクエリがNULLを返した場合でも、メインクエリは行を返す可能性があります。
- E. サブクエリは、WHERE 句、GROUP BY 句、または HAVING 句に配置できます。

正解: ([正解を表示します](#))

質問: 26

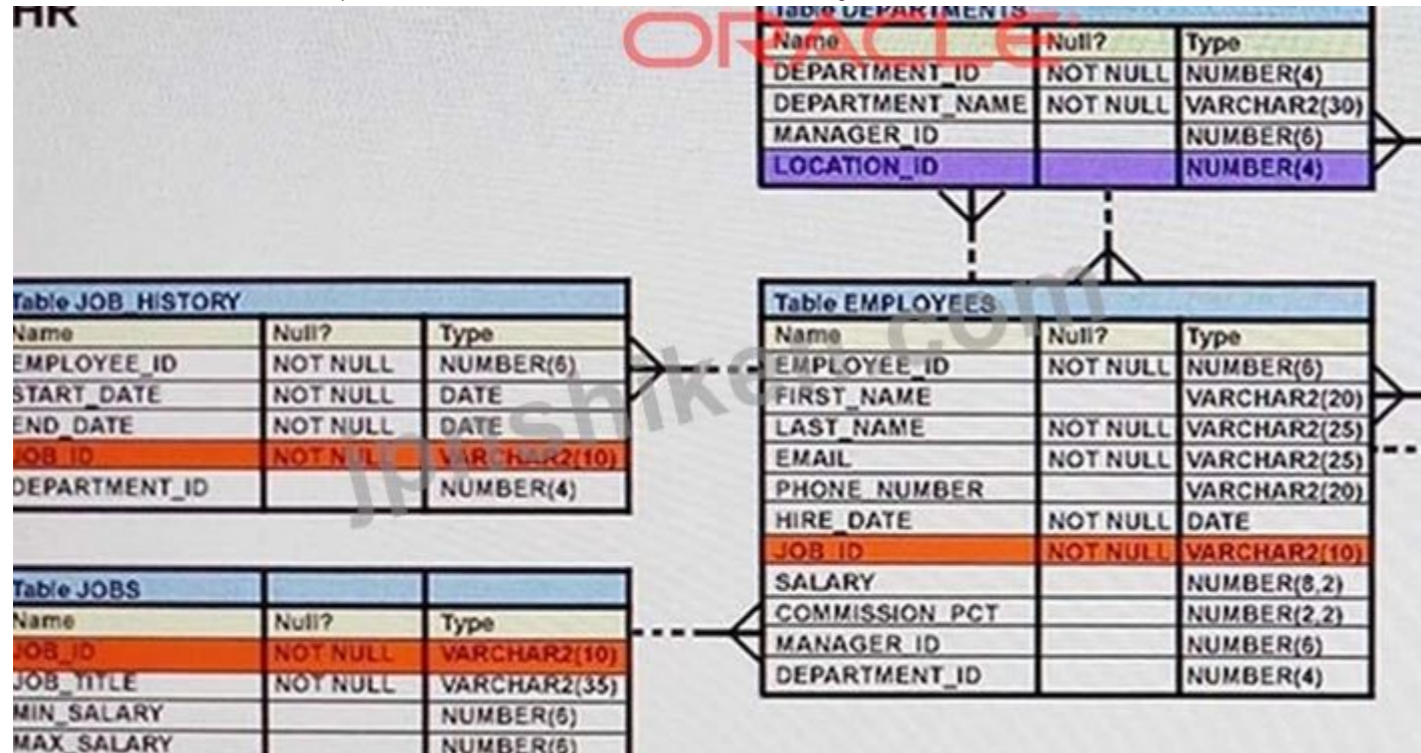
各顧客の信用限度額が1,000ドルずつ増額されたレポートを作成する必要があります。出力では、顧客の姓の欄に「名前」という見出しを付け、増額後の信用限度額の欄に「新しい信用限度額」というラベルを付けてください。列見出しには、各単語の最初の文字のみを大文字で表示してください。

この要件を満たす記述はどれですか？

- A. SELECT INITCAP (cust_last_name) "名前", cust_credit_limit + 1000INITCAP ("新しいクレジット限度額")FROM customers;
 - B. SELECT cust_last_name AS Name, cust_credit_limit + 1000"New Credit Limit"FROM customers;
 - C. SELECT cust_last_name AS "名前", cust_credit_limit + 1000AS "新しいクレジット限度額"FROM customers;
 - D. SELECT cust_last_name AS Name, cust_credit_limit + 1000AS New Credit LimitFROM customers;
- 正解: C (コメントを發表する)

質問: 27

展示資料をご覧になり、従業員表の構造を調べてください。



次のSQL文を評価してください。

```
SELECT employee_id, department_id
```

従業員より

department_id=50 の条件で、department_id で並べ替える。

連合

```
SELECT employee_id, department_id
```

従業員より

where department_id=90

連合

```
SELECT employee_id, department_id
```

従業員より

WHERE department_id=10;

上記のSQL文を実行すると、どのような結果になるでしょうか？

- A. ORDER BY句はSQL文の最後、つまり最後のSELECT文にのみ出現する必要があるため、この文は実行されません。
- B. このステートメントは正常に実行されますが、ORDER BY句は無視され、行はランダムな順序で表示されます。
- C. ORDER BY句では列名ではなく位置表記を使用する必要があるため、このステートメントは実行されません。
- D. このステートメントは正常に実行され、すべての行が DEPARTMENT_ID の昇順で表示されます。

正解: [A \(コメントを發表する\)](#)

質問: 28

外部テーブルについて正しい記述は次のうちどれですか？ (2つ選択)

- A. これらのデータは、SQLまたはPL/SQLのみを使用して取得できます。
- B. それらにインデックスを作成できます。
- C. CREATE TABLE AS SELECT コマンドを使用して、データベース内の既存のデータからそれらにデータを入力できます。
- D. DML ステートメントはそれらには使用できません。
- E. メタデータと実際のデータは両方ともデータベースの外で石化されています。

正解: [C,D \(コメントを發表する\)](#)

質問: 29

BRICKSテーブルの説明を確認してください。

Name	Null?	Type
BRICK_ID		NUMBER(38)
SHAPE		VARCHAR2(30)
COLOR		VARCHAR2(30)
WEIGHT		NUMBER

BRICKS_STAGEテーブルの説明を確認してください。

Name	Null?	Type
WEIGHT		NUMBER
SHAPE		VARCHAR2(30)
COLOR		VARCHAR2(30)

どの2つのクエリが正常に実行されましたか？

- A. ブロックから形状と色を選択します

マイナス

```
SELECT WEIGHT,color FROM bricks_stage;
```

- B. SELECT brick_id,shape FROM bricks

マイナス

bricks_stage から WEIGHT、COLOR を選択します。

- C. レンガの形状と色を選択します

マイナス

```
SELECT color, shape FROM bricks_stage;
```

- D. レンガの形状、色、重量を選択します

マイナス

```
SELECT * FROM bricks_stage;
```

- E. レンガから*を選択

マイナス

```
SELECT * FROM bricks_stage;
```

正解: ([正解を表示します](#))

質問: 30

組み込みデータ型について、正しい記述を3つ挙げてください。

- A. VARCHAR2 は、格納されるデータが非数値であり、特殊文字が含まれていない場合にのみ、列の値を空白で埋めます。
- B. CHAR型の列のデフォルトの長さは常に1文字です。
- C. VARCHAR2 列の定義では、長さを指定する必要はありません。
- D. BLOBは、データベース内に非構造化バイナリデータを格納します。
- E. CHAR型の列定義では、長さを指定する必要はありません。
- F. BFILEは、オペレーティングシステムファイルに非構造化バイナリデータを格納します。

正解: ([正解を表示します](#))

D: 正解です。BLOB (バイナリラージオブジェクト)は、Oracleデータベース内で非構造化バイナリデータを格納するために使用されます。

可変量のデータを格納できます。

F: 正解です。BFILEはOracle SQLのデータ型で、Oracleデータベース外のオペレーティングシステムファイルに格納されているバイナリデータを指すロケータ (ポインタ)を格納するために使用されます。

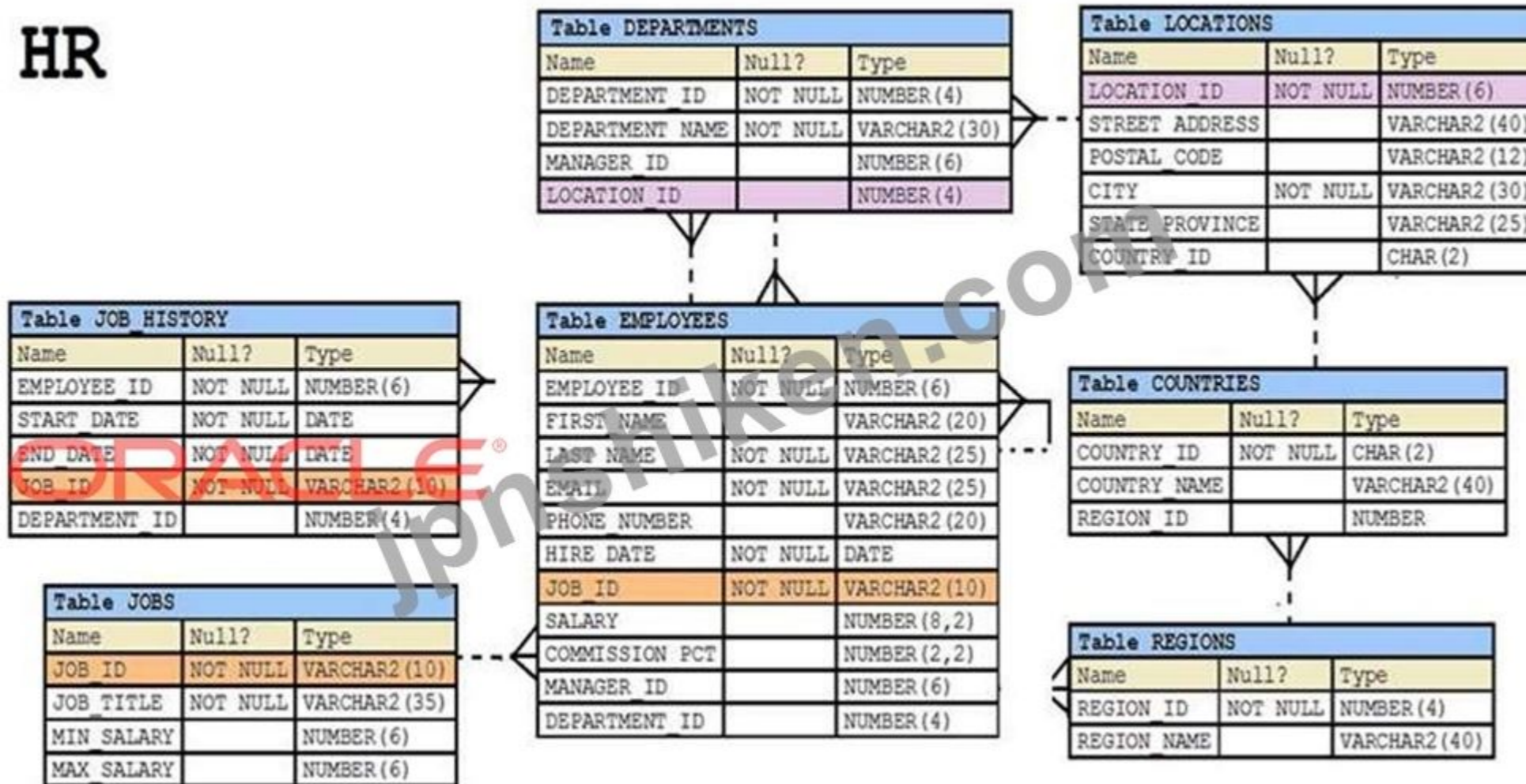
BLOBとBFILEはどちらも大きなバイナリデータに使用されますが、データの実際の格納場所が異なります。BLOBはOracleデータベース内部にデータを格納するのに対し、BFILEはデータベース外部のファイルシステムにデータを保存します。

参考資料 :Oracle Database SQL言語リファレンスガイドでは、BLOBおよびBFILEデータ型の特性と使用方法などを詳しく説明し、それらのストレージ特性とデータ型定義について解説しています。

質問: 31

展示資料を閲覧し、部門別表の構造を調べてください。(2つ選択してください。)

HR



このSQL文を調べてください。

```
SELECT department_id "DEPT_ID", department_name, 'b' FROM
```

部門

```
WHERE departments_id=90
```

連合

```
SELECT department_id, department_name DEPT_NAME, 'a' FROM
```

部門

```
where department_id=10
```

出力結果を並べ替えるために使用できる ORDER BY 句はどれですか？ 2つ選択してください)

- A. 部門IDで並べ替え;
- B. 'b'で並べ替え;
- C. 部門名で並べ替え;
- D. 3個単位で注文してください。

正解: (正解を表示します)

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%ディスカウント、特別な割引コード: **JPNshiken**」

質問: 32

OracleデータベースとSQLについて、正しい記述は次のうちどれですか？ (2つ選択)

- A. クエリは同じスキーマ内のテーブルにのみアクセスできます。
- B. データベースは、ユーザーが作成したテーブルに対して、選択レベルでの読み取りの一貫性を保証します。
- C. UPDATE文を実行すると、データベースインスタンスは更新された各行をロックします。
- D. ユーザーは、同じデータベース内で複数のスキーマの所有者になることができます。
- E. データベースユーザーが行った更新は、ROLLBACK コマンドを使用することで、別のユーザーによってロールバックできます。

正解: ([正解を表示します](#))

質問: 33

SYSDATE関数は、現在のOracleサーバーの日付を次のように表示します。

2019年5月21日

日付を次のように表示したい場合：

2019年5月21日 (月)

どちらの文がこれを実現しますか？

SELECT TO_DATE(SYSDATE, 'FMDAY, DD MONTH, YYYY') FROM DUAL;

- A. SELECT TO_CHAR(SYSDATE, 'FMDD, DAY MONTH, YYYY') FROM DUAL;
- B. SELECT TO_CHAR(SYSDATE, 'FMDAY, DDTH MONTH, YYYY') FROM DUAL;
- C. SELECT TO_CHAR(SYSDATE, 'FMDAY, DD MONTH, YYYY') FROM DUAL;
- D.

正解: **D** ([コメントを發表する](#))

SELECT ID, TO_CHAR(Start_Date,'fmDay Month fmDD, YYYY') AS "Start Date" FROM Employee; 参考: <http://www.java2s.com/Code/Oracle/Data-Type/TOCHARDatefmDayMonthfmDDYYYYEmbeddedspacescanberemovedbyplacingthefmprefix.htm>

質問: 34

顧客テーブルのCUST_CITY列には、CUST_FIRST_NAMEが「Abigail」の場合に「Paris」という値が格納されています。

次のクエリを評価してください。

結果はどうなるだろうか？

- A. エラーメッセージ
- B. アビゲイルは
- C. アビゲイル PA
- D. アビゲイル ッパ

正解: ([正解を表示します](#))

質問: 35

CUSTOMERSテーブルの説明を確認してください。

Name	Null?	Type
CUSTOMER ID	NOT NULL	NUMBER (38)
CUSTOMER_NAME	NOT NULL	VARCHAR2 (100)
INSERT DATE	NOT NULL	DATE

暗黙的な型変換を行う3つの文はどれですか？

- A. SELECT * FROM customers WHERE TO_CHAR(customer_id) = '0001';
- B. SELECT * FROM customers WHERE customer id = '0001';
- C. SELECT * FROM customers WHERE customer_id = 0001;
- D. SELECT. FROM customers WHERE insert_date = DATE *2019-01-01';
- E. SELECT FROM customers WHERE insert date = '01-JAN-19';
- F. SELECT. FRON customers WE TO DATE (Insert _ date) = DATE '2019-01-01';

正解: [\(正解を表示します\)](#)

質問: 36

展示資料をご覧になり、顧客表の構造を調べてください。

CUSTOMERS		
Name	Null?	Type
CUSTOMER_ID	NOT NULL	NUMBER (6)
CUST_NAME		VARCHAR2 (20)
CUST_EMAIL		VARCHAR2 (30)
INCOME_LEVEL		VARCHAR2 (20)

CUSTOMER_VUは、CUSTOMERS_BR1テーブルを基にしたビューであり、CUSTOMERSテーブルと同じ構造を持っています。

顧客情報は、顧客に関する最新情報を反映するように更新する必要があります。

以下のMERGEステートメントのエラーは何ですか？

```
MERGE INTO customers c
  USING customer_vu cv
  ON (c.customer_id = cv.customer_id)
WHEN MATCHED THEN
  UPDATE SET
  c.customer_id = cv.customer_id,
  c.cust_name = cv.cust_name,
  c.cust_email = cv.cust_email,
  c.income_level = cv.income_level
WHEN NOT MATCHED THEN
  INSERT VALUES (cv.customer_id, cv.cust_name, cv.cust_email, cv.income_level)
  WHERE cv.income_level >100000;
```

- A. INTO句がコマンド内で誤った位置にあります。
- B. WHERE句はINSERT文では使用できません。

- C. CUSTOMER_ID列は更新できません。
 - D. CUSTOMER_VU はデータソースとして使用できません。
- 正解: **C** ([コメントを公表する](#))

質問: 37

WHERE句とHAVING句で関数を使用することに関して、正しい記述はどれですか？

- A. HAVING句で単一行関数を使用するにはサブクエリが必要です
- B. WHERE句で単一行関数を使用するにはサブクエリが必要です
- C. WHERE句で集計関数を使用するにはサブクエリが必要です
- D. HAVING句で集計関数を使用するにはサブクエリが必要です

正解: ([正解を表示します](#))

質問: 38

正しい記述は次のうちどれですか？ (2つ選択)

- A. DECODEを使用して評価されたすべての条件は、CASEを使用しても評価できます。
- B. CASEを使用して評価されたすべての条件は、DECODEを使用しても評価できます。
- C. CASEは関数ですが、DECODEは関数ではありません。
- D. DECODEは関数ですが、CASEは関数ではありません。
- E. CASEもDECODEも関数ではありません。
- F. CASEとDECODEはどちらも関数です。

正解: **A,D** ([コメントを公表する](#))

- A. 正しい。DECODE関数は条件を評価できますが、等価性チェックに限定されます。一方、CASE関数は等価性チェックに加え、さまざまな比較演算子を使用してより広範な条件を評価できます。
- D. 正しい。DECODEはOracle SQLの関数で、条件付きクエリの変換を簡略化できます。CASEは関数ではなく、SQLクエリで条件付きロジックを扱う際に、より柔軟性と可読性を高めるためのステートメントです。DECODEは、等価性以外の論理演算を実行できないため、CASEに比べて制限が多い。

質問: 39

以下の関係の種類と例を検討してください。

- 1 One-to-one a) teacher to Student
- 2 One-to-many b) Employees to Manager
- 3 Many-to-one c) Person to SSN
- 4 Many-to-many d) Customers to Products

正しく対応している関係性を示す選択肢はどれですか？ (最適なものを選んでください。)

- A. 1-c、2-d、3-a、および4-b
- B. 1-c、2-a、3-b、および4-d
- C. 1-d、2-b、3-a、4-c
- D. 1-a、2-b、3-c、および4-d

正解: **D** ([コメントを公表する](#))

質問: 40

EMPLOYERS表のこのデータを調べてください。

ID	LAST_NAME	SALARY	DEPT_ID
1	Smith	1000	10
2	Jones	2000	10
3	Marhkham	1500	20
4	Black	1300	20

どのステートメントが正常に実行されますか？

- A. SELECT dept_id, MAX (Last_name), SUM (salary) FROM employees GROUP BY dept_id
- B. SELECT dept_id, LENGTH (last_name), SUM (salary) FROM employees GROUP BY dept_id
- C. SELECT dept_id, STDDEV (last_name), SUM (salary) FROM employees GROUP BY dept_id
- D. SELECT dept_id, INSTR (last_name,'A'), SUM (salary) FROM employees GROUP BY dept_id

正解: [\(正解を表示します\)](#)

SQLでは、GROUP BY句は集計関数と組み合わせて使用され、結果セットを1つまたは複数の列でグループ化します。

- A) このステートメントは正常に実行されます。MAX() は、選択した列の最大値を返すために使用できる集計関数であり、SUM() は値を合計するために使用される集計関数です。どちらも、結果を dept_id でグループ化する GROUP BY 句を含む SELECT ステートメントで有効です。
- B) このステートメントは、GROUP BY句で使用すると正常に実行されません。なぜなら、LENGTH(last_name)は集計関数ではなく、GROUP BY句にも含まれていないからです。集計関数に該当しない選択された列はすべて、GROUP BY句に含める必要があります。
- C) このステートメントは正常に実行されません。STDDEV() は一連の数値の標準偏差を計算する集計関数であり、数値データ型にのみ使用できます。last_name は数値列ではありません。
- D) このステートメントは、B と同じ理由で正常に実行されません。INSTR(last_name, 'A') は集計関数ではないため、SELECT 句にある場合は GROUP BY 句に記述する必要があります。

質問: 41

Oracle SQLのDELETE文とTRUNCATE文について、正しい記述はどれですか？

- A. DELETE文は使用できますが、TRUNCATE文は使用できません。テーブルから行を選択的に削除するために使用できます。
- B. DELTE および TRUNCATE ステートメントでは、テーブルにデータを復元するためにロールバックを実行できます。
- C. DELETE文は、テーブルの特定の列や行からデータを削除するために使用できますが、TRUNCATE文は使用できません。
- D. DELETE および TRUNCATE ステートメントは、実行対象のテーブルのすべてのインデックスを削除します。

正解: [\(正解を表示します\)](#)

質問: 42

展示資料を参照し、PRODUCTSテーブルとORDER_ITEMSテーブルを調べてください。

PRODUCTS	
PRODUCT ID	PRODUCT NAME
1	Inkjet C/8/HQ
2	CPU D300
3	HD 8GB /I
4	HD 12GB /R

ORDER ITEMS			
ORDER ID	PRODUCT ID	QTY	UNIT PRICE
11	1	10	100
22	2	15	120
33	3	10	50
44	1	5	10
66	2	20	125

PRODUCT_NAMEとその商品が注文された回数を表示するために、以下のクエリを実行しました。

```
SQL>SELECT p.product_name, i.item_cnt
```

```
FROM (SELECT product_id, COUNT (*) item_cnt
```

```
注文アイテムから
```

```
GROUP BY product_id) i RIGHT OUTER JOIN products p
```

```
ON i.product_id = p.product_id;
```

上記のステートメントが実行されると、何が起こるでしょうか？

- A. インラインビューと外部結合は一緒に使用できないため、このステートメントは実行されません。
- B. GROUP BY句はインラインでは使用できないため、ステートメントは実行されません。
- C. ITEM_CNT エイリアスは外部クエリに表示できないため、ステートメントは実行されません。
- D. このステートメントは正常に実行され、必要な出力が生成されます。

正解: [\(正解を表示します\)](#)

質問: 43

サブクエリに関して正しい記述を3つ選びなさい。

- A. メインクエリとサブクエリの間で、複数の列または式を比較できます。
- B. サブクエリには ORDER BY 句を含めることができますが、GROUP BY 句を含めることはできません。
- C. メインクエリとサブクエリは、異なるテーブルからデータを取得できます。
- D. サブクエリには、GROUP BY 句と ORDER BY 句を含めることができます。
- E. メインクエリとサブクエリは同じテーブルからデータを取得する必要があります。
- F. メインクエリとサブクエリの間で比較できる列または式は 1 つだけです。

正解: [A,C,D \(コメントを公表する\)](#)

説明

参考文献 :

<http://docs.oracle.com/javadb/10.6.2.1/ref/rrefsqlj13658.html>

質問: 44

クエリのWHERE句で指定された内部結合について、正しい記述はどれですか？

- A. 結合するすべてのテーブルで列名が同じである必要があります。
- B. 結合列に主キーと外部キーの制約が定義されている必要があります。
- C. 等結合条件と非等結合条件の両方に適用されます。
- D. これは等結合条件にのみ適用されます。

正解: [\(正解を表示します\)](#)

質問: 45

展示資料をご覧になり、顧客表の構造を調べてください。

Table customers		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2(20)
CUST_LAST_NAME	NOT NULL	VARCHAR2(40)
CUST_GENDER	NOT NULL	CHAR(1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER(4)
CUST_MARITAL_STATUS		VARCHAR2(20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2(40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2(10)
CUST_CITY	NOT NULL	VARCHAR2(30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2(40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2(30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2(30)

姓がA、B、またはCで始まり、かつ信用限度額が10,000未満のすべての顧客の姓と信用限度額を示すレポートを作成したい。

以下の2つのクエリを評価してください。

```
SQL> SELECT cust_last_name, cust_credit_limit FROM customers  
WHERE (UPPER(cust_last_name) LIKE 'A%' OR  
(cust_last_name) LIKE 'B%' OR UPPER (cust_last_name) LIKE 'C%')
```

かつ、顧客信用限度額が10000未満であること。

```
SQL>SELECT cust_last_name, cust_credit_limit FROM customers  
WHERE UPPER (cust_last_name) BETWEEN 'A' AND 'C'
```

かつ、顧客信用限度額が10000未満であること。

上記のクエリの実行に関して、正しい記述はどれですか？

- A. 最初のクエリのみが正しい結果を返します
- B. 両方とも正常に実行されるが、必要な結果が得られない
- C. 2番目のクエリのみが正しい結果を返します
- D. 両方とも正常に実行され、同じ結果になる

正解: [A \(コメントを發表する\)](#)

質問: 46

テーブル HR.EMPLOYEES には、EMPLOYEE_ID が 109 である行が含まれています。

ユーザーALICEにはHR.EMPLOYEESにアクセスする権限がありません。

ユーザーALICEがセッションを開始します。

ユーザーHRがセッションを開始し、以下のステートメントを正常に実行しました。

従業員に対する削除権限をアリスに付与します。

```
UPDATE employees SET salary = 24000 WHERE employee_id = 109;
```

ALICEは、既存のセッションで以下を実行します。

```
hr.employees テーブルから employee_id = 109 のレコードを削除します。
```

結果はどうなりますか？

- A. DELETEコマンドは行を即座に削除します。
- B. DELETEコマンドはHRのトランザクションが終了するまで待機し、その後エラーを返します。
- C. DELETEコマンドはHRのトランザクションが終了するまで待機してから行を削除します。
- D. DELETEコマンドはすぐにエラーを返します。

正解: ([正解を表示します](#))

有効的な**1z0-071**問題集はJPNTTest.com提供され、**1z0-071**試験に合格することに役に立ちます！JPNTTest.comは今最新**1z0-071**試験問題集を提供します。JPNTTest.com 1z0-071 試験問題集はもう更新されました。ここで**1z0-071**問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> **825**問、**30%** **ディスカウント**、特別な割引コード: **JPNshiken**」

質問: 47

CUSTOMERSテーブルには、データ型がVARCHAR2のCUST_LAST_NAME列があります。

この表には、COST_LAST_MANEの値がAndersonとAussonである行が2つあります。

どのクエリを使えば、CUST_LAST_SAME に対して、最初の行に Oder、2 番目の行に Aus を含む出力が得られますか？

- A. SELECT REPLACE (REPLACE(cust_last_name,'son',''),'An','O') FROM customers;
- B. SELECT REPLACE (TRIM(TRALING'son' FROM cust_last_name),'An','O') FROM customers;
- C. SELECT INITCAP (REPLACE(TRIM('son' FROM cust_last_name),'An','O')) FROM customers;
- D. SELECT REPLACE (SUBSTR(cust_last_name,-3),'An','O') FROM customers;

正解: ([正解を表示します](#))

Oracle SQL の REPLACE 関数は、指定された部分文字列の出現箇所を別の部分文字列に置き換えるために使用されます。このクエリでは、内部の REPLACE 関数呼び出し REPLACE(cust_last_name, 'son', '') が cust_last_name から部分文字列 'son' を削除します。次に、外部の REPLACE 関数呼び出しが部分文字列 'An' を置き換えます。

O」。与えられたデータの場合、Anderson」はまず内部のREPLACEによって Ander」に変換され、次に

Ander」は外側のREPLACEによって Oder」に変換されます。同様に、Ausson」は最初に次のように変更されます。

内側の REPLACE による Aus」は、外側の REPLACE の影響を受けません。

文字列関数 (REPLACEを含む)の機能については、Oracle Database SQL言語リファレンスのドキュメントを参照してください。

質問: 48

COUNT関数について正しい記述は次のうちどれですか？ (2つ選択)

- A. 数値データ型にのみ使用できます。
- B. COUNT (DISTINCT inv_amt) は、INV_AMT 列に重複行と NULL が含まれる行を除外した行数を返します。
- C. COUNT(*) は、重複行や任意の列に NULL が含まれる行を含めたテーブル内の行数を返します。
- D. DISTINCTキーワードでCOUNT関数を使用するSELECT文にはWHERE句を含めることはできません。
- E. COUNT(inv_amt) は、INV_AMT 列に NULL が含まれる行を含むテーブル内の行数を返します。

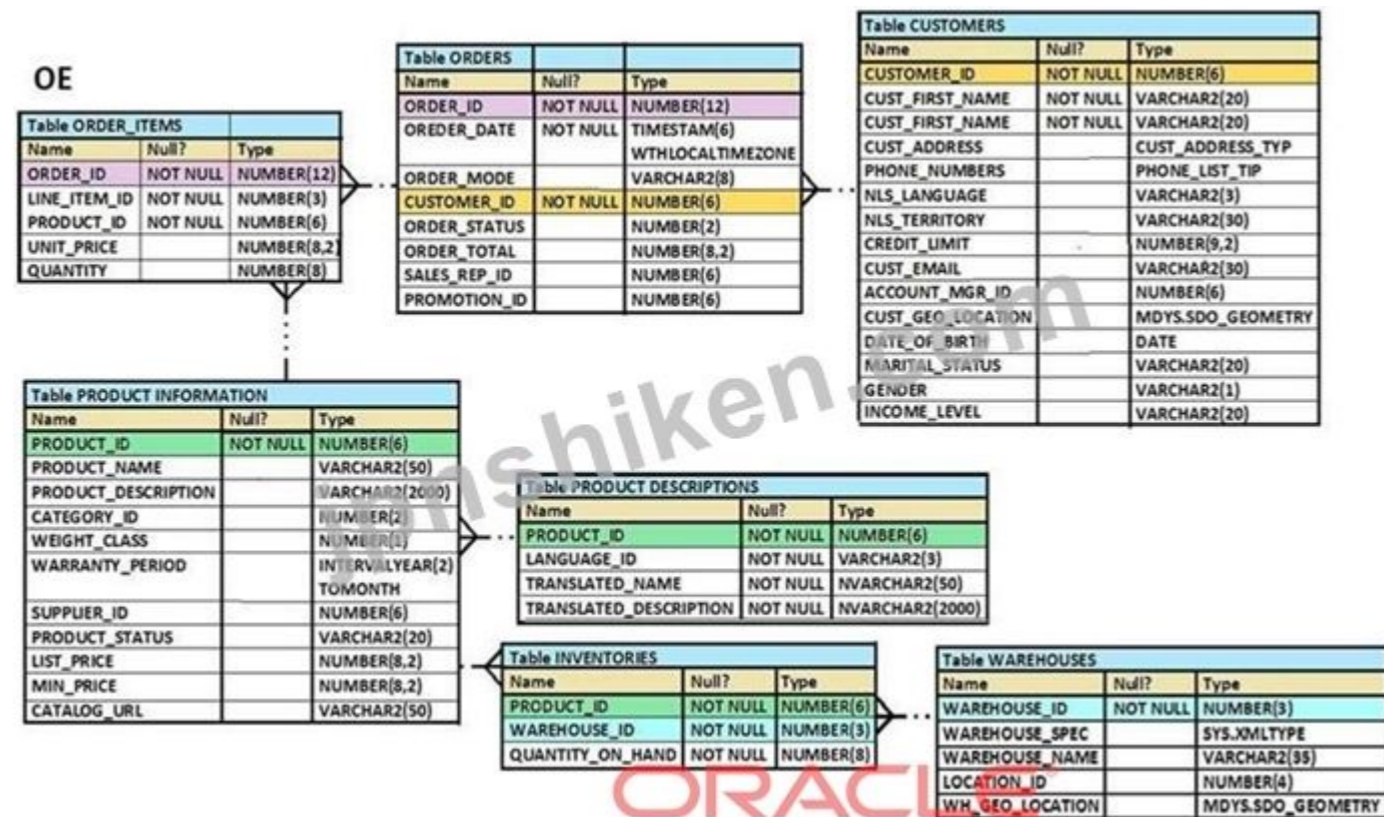
正解: (正解を表示します)

COUNT関数は、SQLで最もよく使用される集計関数の1つで、グループ内の項目の数を決定するために使用されます。正解がBとCである理由は以下のとおりです。

- * A: 間違いです。COUNT関数はNUMBER型だけでなく、あらゆるデータ型で使用できます。データ型に関係なく行数をカウントします。
- * B: 正解です。COUNT(DISTINCT inv_amt) は、inv_amt 列内の一意の非 NULL 値の数をカウントします。重複は除外され、NULL 値は無視されます。
- * C: 正解です。COUNT(*) は、重複行や任意の列に NULL 値がある行も含め、テーブル内のすべての行をカウントします。内容に関係なく、行の総数を示します。
- * D: 間違いです。COUNT関数にはWHERE句を使用できます。WHERE句はカウント前行をフィルタリングするもので、これはSQLの標準的な動作です。
- * E: 誤りです。COUNT(inv_amt) は、inv_amt が NULL でない行をカウントします。inv_amt 列に NULL が含まれている行はカウントされません。

質問: 49

展示資料を参照し、PRODUCT_INFORMATION テーブルの構造を調べてください。(最適な回答を選択してください。)



PRODUCT_ID列は主キーです。

インデックスを作成するには、次のコマンドを使用します。

```
SQL > CREATE INDEX upper_name_idx
```

```
ON product_information(UPPER(product_name));
```

PRODUCT_INFORMATIONテーブルには、他にインデックスは存在しません。

UPPER_NAME_IDX インデックスを使用するクエリはどれですか？

A. SELECT product_id, UPPER(product_name)

製品情報から

WHERE UPPER(product_name) = 'LASERPRO' OR list_price > 1000;

B. SELECT UPPER(product_name)

製品情報から;

C. SELECT UPPER(product_name)

製品情報から

WHERE product_id = 2254;

D. SELECT product_id

製品情報から

WHERE UPPER(product_name) IN ('LASERPRO', 'CABLE');

正解: ([正解を表示します](#))

説明

質問: 50

以下のCREATE TABLEコマンドを評価してください。

```
CREATE TABLE order_item
(order_id NUMBER (3),
item-id NUMBER (2),
qty NUMBER (4),
CONSTRAINT ord_itm_id_pk
PRIMARY KEY (order_id, item_id)
USING INDEX
(CREATE INDEX ord_itm_idx
ON order_item (order_id, item_id)));
```

上記のSQL文に関して、正しい記述はどれですか？

- A. 複合プライマリーでは USING INDEX 句を使用できないため、エラーが発生します。
- B. CREATE TABLE コマンドでは USING INDEX 句が許可されていないため、エラーが発生します。
- C. 正常に実行され、ORD_ITM_IDX インデックスのみが作成されます。
- D. 正常に実行され、ORD_ITM_IDX と ORD_ITM_ID_PK という 2 つのインデックスが作成されます。

正解: ([正解を表示します](#))

質問: 51

集計関数について、正しい記述はどれですか？

- A. AVG関数はNULLを暗黙的にゼロに変換します。
- B. 集計関数は任意のレベルまでネストできます。
- C. MAX関数とMIN関数は、文字データ型の列で使用できます。
- D. 集計関数は、SELECT文のどの句でも使用できます。

正解: ([正解を表示します](#))

説明／参考資料 <https://docs.oracle.com/database/121/SQLRF/functions003.htm>

質問: 52

従業員テーブルの構造を調べます。

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER (6)
FIRST_NAME		VARCHAR2 (20)
LAST_NAME	NOT NULL	VARCHAR2 (25)
EMAIL	NOT NULL	VARCHAR2 (25)
PHONE_NUMBER		VARCHAR2 (20)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2 (10)
SALARY		NUMBER (8, 2)
COMMISSION_PCT		NUMBER (2, 2)
MANAGER_ID		NUMBER (6)
DEPARTMENT_ID		NUMBER (4)

EMPLOYEE_IDとMANAGER_IDの間には親子関係があります。

EMPLOYEE_IDが123である従業員と同じマネージャーの下で働く従業員の姓とマネージャーIDを表示したい。

どのクエリが正しい出力結果をもたらしますか？

A. SELECT e.last_name, m.manager_id

FROM employees e RIGHT OUTER JOIN employees m
(e.manager_id = m.employee_id)

かつ、e.employee_id = 123;

B. SELECT m.last_name, e.manager_id

FROM employees e LEFT OUTER JOIN employees m
(e.manager_id = m.manager_id)

e.employee_id = 123 の場合;

C. SELECT e.last_name, m.manager_id

FROM employees e LEFT OUTER JOIN employees m
(e.employee_id = m.manager_id)

e.employee_id = 123 の場合;

D. SELECT e.last_name, e.manager_id

FROM employees e RIGHT OUTER JOIN employees m
(e.employee_id = m.employee_id)

e.employee_id = 123 の場合;

正解: (正解を表示します)

質問: 53

以下の関係の種類と例を検討してください。

1. 教師と生徒の個別指導
2. 一対多 b) 従業員からマネージャーへ
- 3 多対一 c) 個人と社会保障番号
- 4 多対多 d) 顧客と製品

どの選択肢が、正しく一致した関係性を示していますか？

- A. 1-c、2-a、3-b、4-d
- B. 1-a、2-b、3-c、4-d
- C. 1-c、2-d、3-a、および4-b
- D. 1-d、2-b、3-a、および4-c

正解: **B** ([コメントを发表する](#))

質問: 54

スカラーサブクエリ式について正しい記述はどれですか？ (2つ選択してください。)

- A. 最大で1行まで戻ることができます。
- B. それらを列のデフォルト値として使用できます。
- C. 親ステートメント内のテーブルと関連付けることはできません。
- D. 2つの列を返すことができます。
- E. 括弧で囲む必要があります。

正解: **A,E** ([コメントを发表する](#))

質問: 55

INVOICEテーブルの構造を調べてください。

Name	Null?	Type
INV_NO	NOT NULL	NUMBER(3)
INV_DATE		DATE
INV_AMT		NUMBER(10,2)

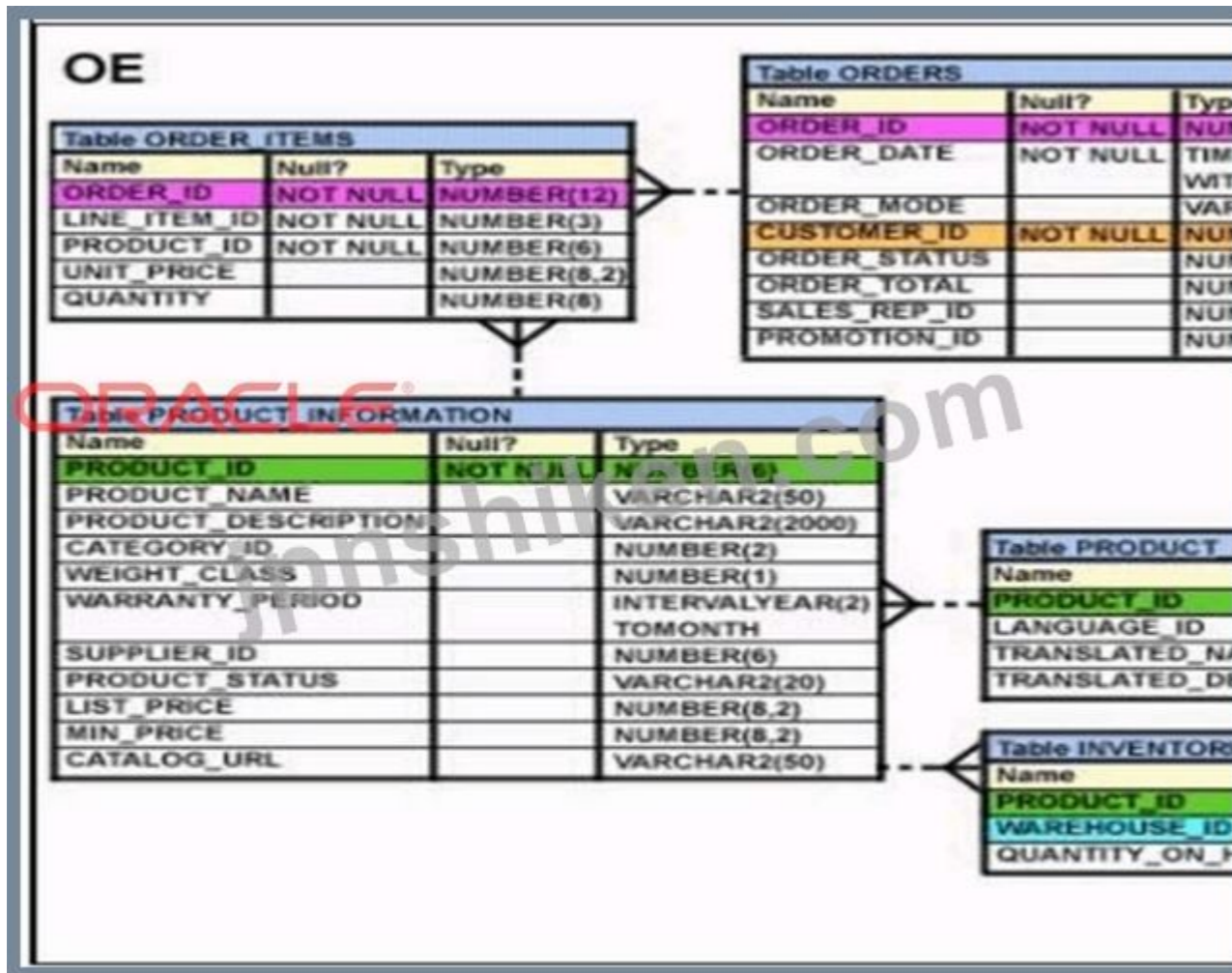
どの2つのSQL文が正常に実行されるでしょうか？ (2つ選択してください。)

- A. SELECT inv_no,NVL2(inv_date,'Pending','Incomplete')FROM invoice;
- B. SELECT inv_no,NVL2(inv_amt,inv_amt*.25,'利用不可')FROM invoice;
- C. SELECT inv_no,NVL2(inv_amt,inv_date,'利用不可')FROM invoice;
- D. SELECT inv_no,NVL2(inv_date,sysdate-inv_date,sysdate)FROM invoice;

正解: **A,D** ([コメントを发表する](#))

質問: 56

展示資料を参照し、PRODUCT_INFORMATION テーブルの構造を調べてください。(最適な回答を選択してください。)



PRODUCT_ID列は主キーです。

インデックスを作成するには、次のコマンドを使用します。

```
SQL > CREATE INDEX upper_name_idx
```

```
ON product_information(UPPER(product_name));
```

PRODUCT_INFORMATIONテーブルには、他にインデックスは存在しません。

UPPER_NAME_IDX インデックスを使用するクエリはどれですか？

A. SELECT UPPER(product_name)

製品情報から;

B. SELECT UPPER(product_name)

製品情報から

```
WHERE product_id = 2254;
```

C. SELECT product_id

製品情報から

```
WHERE UPPER(product_name) IN ('LASERPRO', 'CABLE');
```

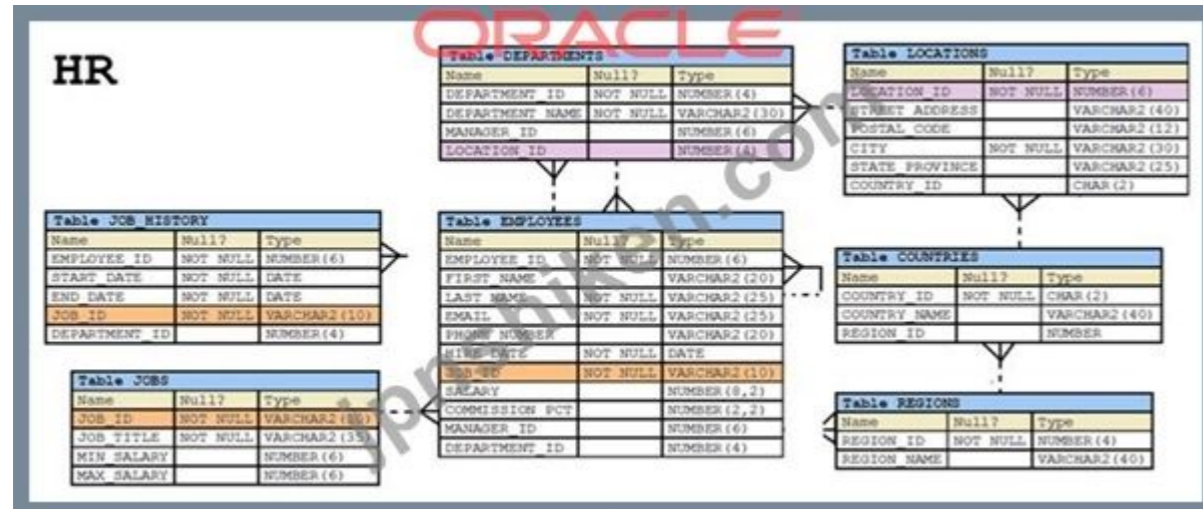
D. SELECT product_id, UPPER(product_name)

製品情報から

```
WHERE UPPER(product_name) = 'LASERPRO' OR list_price > 1000;
```

正解: C (コメントを发表する)

展示物をご覧になり、各部署および従業員に関する表の説明をご確認ください。



あなたは、全従業員の従業員ID、名、部署名を取得するために、以下のSQL文を作成しました。

```
SELECT employee_id, first_name, department_name
```

従業員より

部門間の自然な結合。

上記のSQL文を実行しても、期待する出力が得られません。その原因は何でしょうか？

A. FROM句では、DEPARTMENTSテーブルはEMPLOYEESテーブルより前に使用されていません。

B. EMPLOYEES テーブルと DEPARTMENTS テーブルには、同じ列名とデータ型を持つ列が複数存在します。

自然結合では、各テーブルで同じ列が1つあれば十分です。EMPLOYEESテーブルとDEPARTMENTSテーブルには、同じ列が2つあります (Department_IDとManager_ID)。

C. NATURAL JOIN句にUSING句がありません。

D. SELECT句の列名にテーブル接頭辞がありません。

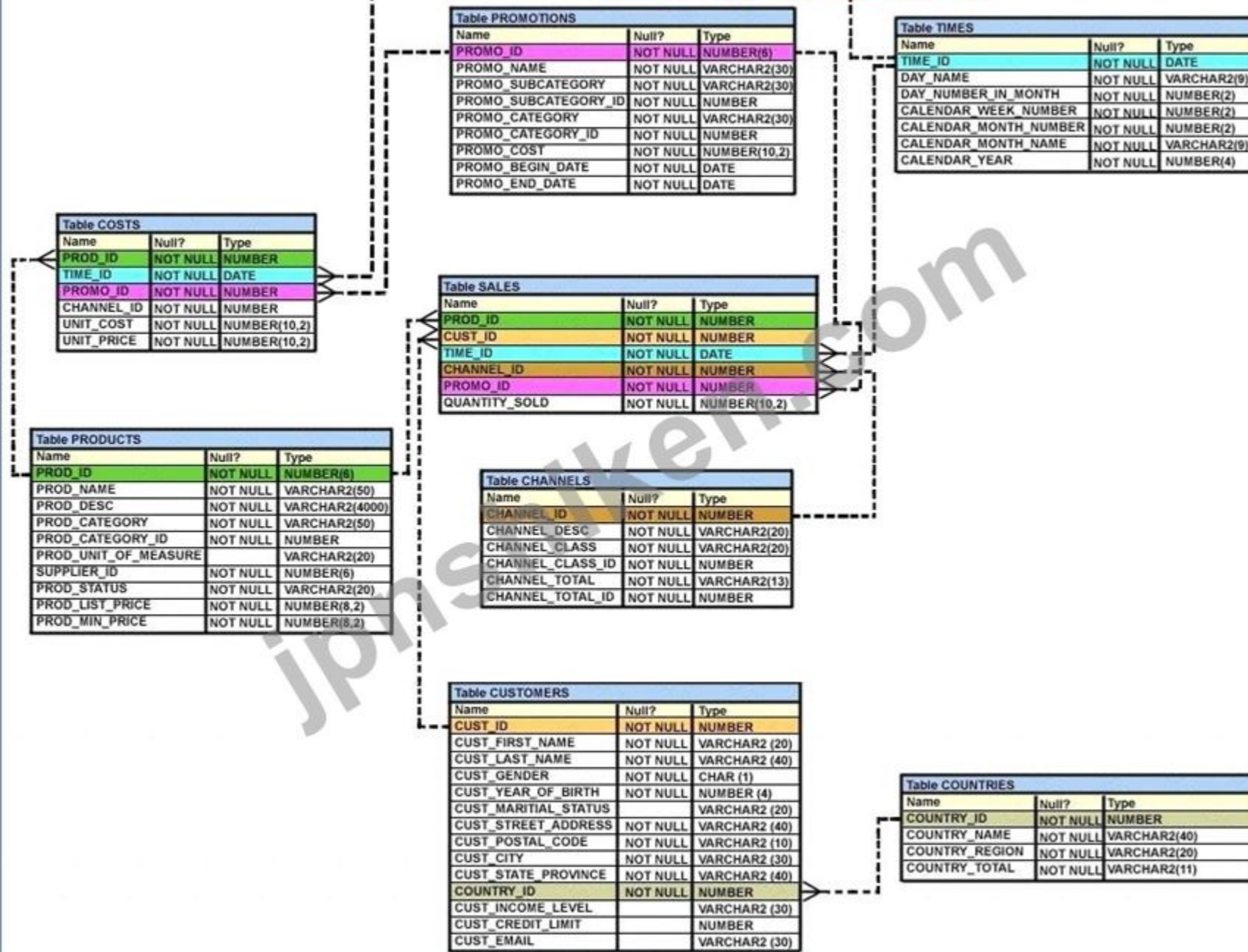
正解: (正解を表示します)

質問: 58

展示物をご覧になり、売上、顧客、製品、および時間の表の構造を調べてください。

SH

ORACLE



PROD_ID列はSALESテーブルの外部キーであり、PRODUCTSテーブルを参照しています。

同様に、SALESテーブルのCUST_ID列とTIME_ID列も、それぞれCUSTOMERSテーブルとTIMESテーブルを参照する外部キーです。

次のCREATE TABLEコマンドを評価してください。

```
CREATE TABLE new_sales (prod_id, cust_id, order_date DEFAULT SYSDATE)
```

として

```
SELECT prod_id, cust_id, time_id
```

売上から;

上記のコマンドに関して、正しい記述はどれですか？

- A. 列定義でデフォルト値を指定できないため、NEW_SALES テーブルは作成されません。
- B. CREATE TABLE コマンドと SELECT 句の列名が一致しないため、NEW_SALES テーブルは作成されません。
- C. NEW_SALES テーブルが作成され、指定された列に定義されているすべての NOT NULL 制約が新しいテーブルに渡されます。

D. NEW_SALES テーブルが作成され、指定された列に定義されているすべての FOREIGN KEY 制約が新しいテーブルに渡されます。

正解: ([正解を表示します](#))

質問: 59

PRODUCT_INFORMATION テーブルの説明を確認してください。

Name	NULL?	Type
PROD_ID	NOT NULL	NUMBER(2)
PROD_NAME		VARCHAR2 (10)
LIST_PRICE		NUMBER(6,2)

A. SELECT COUNT(DISTINCT list_price) FROM product_information WHERE list_price is null.

B. SELECT count(nvl(list_price,0)) FROM product_information WHERE list_price is null;

C. list_priceがNULLであるproduct_informationからcount(list_price)を選択します。

D. SELECT (COUNT(list_price) FROM Product_intormation WHERE list_price=NULL;

正解: ([正解を表示します](#))

質問: 60

外部テーブルに関して、正しい記述はどれですか？

A. CREATE TABLE AS SELECT ステートメントを使用すると、外部テーブルからデータベース内の通常のテーブルにデータをアンロードできます。

B. ORACLE_LOADERとORACLE_DATAPUMPは、外部テーブルで使用する場合、まったく同じ機能を持ちます。

C. 外部テーブルのデフォルトの拒否制限は無制限です。

D. 外部テーブルのデータとメタデータは、データベースの外部に保存されます。

正解: ([正解を表示します](#))

質問: 61

PROMOTIONS テーブルの PROMO_BEGIN_DATE 列はデータ型であり、デフォルトの日付形式は DD-MON-RR です。クエリで PROMO_BEGIN_DATE を使用する式について、正しい記述は次の 2 つのうちどれですか。

A. TONUMBER (PROMO BEGIN_DATE) - 5 は数値を返します

B. PROMO_BEGIN_DATE - 5 は日付を返します

C. PROMO_BEGIN_DATE - SYSDATE は数値を返します

D. PROMO_BEGIN_DATE - SYSDATE はエラーを返します

E. TODATE(PROMO BEGIN_DATE *5) は日付を返します

正解: ([正解を表示します](#))

Oracle SQLでは、日付から数値を引くと、結果は日付になります。ある日付から別の日付を引くと、結果は2つの日付間の日数になります。

* B: PROMO_BEGIN_DATE - 5 は、PROMO_BEGIN_DATE から 5 日を差し引くことで、PROMO_BEGIN_DATE より 5 日前の新しい日付になります。

* C: PROMO_BEGIN_DATE - SYSDATE は、PROMO_BEGIN_DATE と現在の日付 (SYSDATE) の間の日数を返します。

誤った選択肢は以下のとおりです。

* A: TONUMBER(PROMO_BEGIN_DATE) - 5 は数値を返しません。

PROMO_BEGIN_DATEは日付であり、TONUMBERはOracleで日付を数値に変換するための有効な関数ではありません。

* D: PROMO_BEGIN_DATE - SYSDATE はエラーを返さず、上記の説明どおり、2 つの日付間の日数を返します。

- * E: TODATE(PROMO_BEGIN_DATE * 5) は、PROMO_BEGIN_DATE が日付であるため、日付を返しません。
 - * 5 は Oracle SQL では有効な演算ではありません。日付と数値を掛け合わせることはできませんし、TODATE は有効な関数ではありません。正しい関数名は TO_DATE です。
- 参考文献：
- * Oracleの日付演算に関するドキュメント :データベースSQL言語リファレンス - 日時関数

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%ディスカウント、特別な割引コード: **JPNshiken**」

質問: 62

展示資料をご覧になり、売上表の構造を調べてください。

Table SALES		
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER
CUST_ID	NOT NULL	NUMBER
TIME_ID	NOT NULL	DATE
CHANNEL_ID	NOT NULL	NUMBER
PROMO_ID	NOT NULL	NUMBER
QUANTITY_SOLD	NOT NULL	NUMBER(10,2)

以下のクエリは、販売数が55,000を超え、かつ注文数が10個を超えるすべての製品IDをSALESテーブルから取得するために作成されています。

```
SQL> SELECT prod_id
FROM sales
WHERE quantity_sold > 55000 AND COUNT(*) > 10
GROUP BY prod_id
HAVING COUNT(*) > 10;
```

このSQL文に関して、正しい記述はどれですか？

- A. 実行は成功しますが、COUNT(*) の代わりに COUNT(prod_id) を使用する必要があるため、結果は生成されません。
- B. SELECT句にもCOUNT(*)を指定する必要があるため、エラーが発生します。
- C. 正常に実行され、必要な結果が生成されます。
- D. COUNT(*)はHAVING句にのみ存在し、WHERE句には存在してはならないため、エラーが発生します。

正解: [\(正解を表示します\)](#)

質問: 63

顧客テーブルのCUST_CITY列には、CUST_FIRST_NAMEが「Abigail」の場合に「Paris」という値が格納されています。

次のクエリを評価してください。

```
SQL> SELECT INITCAP(cust_first_name || ' ' ||
                UPPER(SUBSTR(cust_city, -LENGTH(cust_city), 2)))
FROM customers
WHERE cust_first_name = 'Abigail';
```

結果はなるだろうか？

- A. アビゲイル PA
- B. エラーメッセージ
- C. アビゲイルは
- D. アビゲイル パ

正解: [\(正解を表示します\)](#)

質問: 64

展示資料を参照し、パーティション化されておらず、索引構成テーブルでもないEMPテーブルの構造を調べてください。(2つ選択してください。)

EMP		
Name	Null?	Type
EMPNO	NOT NULL	NUMBER (4)
FIRST_NAME		VARCHAR2 (20)
LAST_NAME		VARCHAR2
SALARY		NUMBER (10, 2)
DEPTNO		NUMBER (2)

このSQL文を評価してください。

```
ALTER TABLE emp
```

first_name 列を削除します。

正しい記述は次のうちどれですか？ (2つ選択)

- A. SQL ステートメントに SET UNUSED オプションを追加すれば、FIRST_NAME 列の削除をロールバックできます。
- B. FIRST_NAME列にはデータが含まれていない場合、その列は削除されます。
- C. テーブルに少なくとも1つの列が残っている場合、FIRST_NAME列は削除されます。
- D. FIRST_NAME列は、SQLステートメントにCASCADEオプションを追加すれば、複合PRIMARY KEYの一部であっても削除できます。

正解: [\(正解を表示します\)](#)

質問: 65

役割に関する以下の記述のうち、正しいものはどれですか？ (3選)

- A. ロールはALTER ROLEステートメントを使用してロールに割り当てられます。
- B. ロールとは、関連する権限の名前付きグループであり、ユーザーにのみ割り当てることができます。
- C. ALTER USER ステートメントを使用して、ユーザーにロールを割り当てます。
- D. 1つの役割を複数のユーザーに割り当てることができます。
- E. 1人のユーザーに複数の役割を割り当てることができます

F. 権限は、ALTER ROLE ステートメントを使用してロールに割り当てられます。

G. GRANT ステートメントを使用して、ロールに権限が割り当てられます。

正解: [\(正解を表示します\)](#)

ロールとは、Oracleデータベースにおける権限の集合体を表す名称です。

A) 誤り。ALTER ROLE ステートメントを使用して、ロールを他のロールに割り当てることはできません。

B) 誤り。役割はユーザーと他の役割の両方に割り当てることができます。

C) 正しい。ロールはALTER USERステートメントを使用してユーザーに割り当てられますが、これは唯一の方法ではありません。

D) 正しい。1つの役割を複数のユーザーに割り当てることができ、ユーザー権限の管理が簡素化されます。

E) 正しい。1人のユーザーに複数の役割を割り当てることができます。

F) 誤り。ALTER ROLE ステートメントを使用してロールに権限を割り当てることはできません。

G). 正しい。GRANT ステートメントを使用して、ロールに権限が割り当てられます。

質問: 66

SET VERIFY ON コマンドについて、正しい記述は次のうちどれですか？ (2つ選択)

A. DEFINEコマンドで作成された変数の値を表示します。

B. SQL DeveloperとSQL*Plusで使用できます。

C. SQL*plusでのみ使用できます。

D. && で始まる変数の値を表示します。

E. クエリのWHERE句でのみ使用される変数の値を表示します。

正解: [\(正解を表示します\)](#)

SET VERIFY ON コマンドは、SQL*Plus および SQL Developer が置換変数に関する情報を表示する方法に関連しています。

A. DEFINE コマンドで作成された変数の値を表示します。VERIFY が ON に設定されている場合、SQL*Plus および SQL Developer は、DEFINE コマンドを使用して置換変数が再定義されたとき、またはセッション中に新しい値が指定されたときに、置換変数の古い値と新しい値を表示します。

B. SQL DeveloperとSQL*Plusで使用できます。従来はSQL*Plusに関連付けられていましたが、SET VERIFYコマンドはSQL Developerでもサポートされており、両方の環境で置換変数の値の表示を制御できます。

参照 :

Oracle SQL*Plusユーザーガイドおよびリファレンス、特にSETコマンドと置換変数に関するセクションを参照してください。

質問: 67

展示資料をご覧になり、PRODUCTS表のデータをご確認ください。

PRODUCTS				
PROD_ID	PROD_NAME	PROD_CATEGORY	PROD_MIN_PRICE	PROD_UNIT_OF_MEASURE
101	Envoy 256MB - 40GB	Hardware	6000	Nos.
102	Y Box	Electronics	9000	
103	DVD-R Disc, 4.7 GB	Software/Other	2000	Nos.
104	Documentation Set - Spanish	Software/Other	4000	

PRODUCTS テーブルから、「ソフトウェア/その他」カテゴリに属し、最低価格が \$2000 または \$4000 で、単位が指定されていない製品名を表示する必要があります。
あなたは次のクエリを発行します。

```
SQL > SELECT prod_name, prod_category, prod_min_price
FROM products
WHERE prod_category LIKE '%Other%' AND (prod_min_price = 2000
OR
prod_min_price = 4000) AND prod_unit_of_measure <> '';
```

次の記述のうち、正しいものはどれですか？

- A. 正常に実行されますが、結果は返されません。
- B. PROD_CATEGORY列に指定された条件が無効であるため、エラーが発生します。
- C. PROD_UNIT_OF_MEASURE に指定された条件が無効であるため、エラーが発生します。
- D. 正常に実行され、必要な結果が返されます。

正解: [\(正解を表示します\)](#)

質問: 68

グローバル一時テーブルについて、正しい記述を3つ挙げてください。

- A. セッションによって挿入されたグローバル一時テーブルの行は、そのテーブルに対する選択権限が付与されている他のセッションでも利用可能です。
- B. セッションで発行された TRUNCATE コマンドは、発行セッションのグローバル一時テーブル内のすべての行を削除します。
- C. グローバル一時テーブルの領域割り当てはセッション開始時に発生します。
- D. グローバル一時テーブルに対する DELETE コマンドはロールバックできません。
- E. グローバル一時テーブルの定義は、複数のセッションで使用できます。
- F. セッション終了時に存在するグローバル一時テーブルの行はすべて削除されます。

正解: [\(正解を表示します\)](#)

質問: 69

展示資料をご覧になり、CUSTOMERStableの構造を調べてください。

CUSTOMERStableを使用して、クレジット限度額の増加を示すレポートを生成する必要があります。

すべてのお客様に15%の手数料がかかります。クレジット限度額が入力されていないお客様には、「利用不可」というメッセージが表示されます。

どのSQL文が要求される結果を生成しますか？

Table CUSTOMERS		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2 (20)
CUST_LAST_NAME	NOT NULL	VARCHAR2 (40)
CUST_GENDER	NOT NULL	CHAR (1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER (4)
CUST_MARITAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

A. SELECT TO_CHAR (NVL(cust_credit_limit * .15), '利用不可') "新規クレジット" FROM customers;

B. SELECT NVL(cust_credit_limit), '利用不可') "新規クレジット"

お客様からの声;

C. SELECT NVL (TO_CHAR(cust_credit_limit * .15), '利用不可') "新規クレジット" FROM customers;

D. SELECT NVL(cust_credit_limit * .15), '利用不可') "新規クレジット"

お客様からの声;

正解: ([正解を表示します](#))

質問: 70

展示資料を参照し、プロモーション表の構造を調べてください。

Table PROMOTIONS		
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(6)
PROMO_NAME	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY_ID	NOT NULL	NUMBER
PROMO_CATEGORY	NOT NULL	VARCHAR2(30)
PROMO_CATEGORY_ID	NOT NULL	NUMBER
PROMO_COST	NOT NULL	NUMBER(10,2)
PROMO_BEGIN_DATE	NOT NULL	DATE
PROMO_END_DATE	NOT NULL	DATE

PROMOTIONSテーブルを使用して、

カテゴリーAにおいて、0～2000ドルと2000～5000ドルの範囲にあるすべてのプロモーションの平均コストを調べる必要があります。
以下のSQL文を発行します。

```
SQL>SELECT AVG(CASE
      WHEN promo_cost BETWEEN 0 AND 2000 AND promo_category='A'
      THEN promo_cost
      ELSE null END) "CAT_2000A",
AVG(CASE
      WHEN promo_cost BETWEEN 2001 AND 5000 AND promo_category='A'
      THEN promo_cost
      ELSE null END) "CAT_5000A"
FROM promotions;
```

結果はどのようなだろうか？

- A. WHEN句には複数の条件を指定できないため、エラーが発生します。
- B. 正常に実行され、必要な結果が得られます。
- C. CASE はグループ関数では使用できないため、エラーが発生します。
- D. NULLを戻り値として指定できないため、エラーが発生します。

正解: [\(正解を表示します\)](#)

説明

CASE式

IF-THEN-ELSE文の役割を果たすことで、条件付き照会を容易にします。

CASE expr WHEN comparison_expr1 THEN return_expr1

[WHEN comparison_expr2 THEN return_expr2

WHEN comparison_exprn THEN return_exprn

ELSE else_expr]

終わり

質問: 71

データ辞書について、正しい記述は次のうちどれですか？ (2つ選択)

- A. すべてのユーザー操作はデータ辞書に記録されます。
- B. データディクショナリは、データベースオブジェクト、権限、およびデータの変更を反映するために常に更新されます。
- C. SYSユーザーは、データディクショナリ内のすべての基本テーブルとユーザーがアクセスできるビューを所有します。
- D. データディクショナリ内の基本テーブルには、DBA_ という接頭辞が付きます。
- E. すべてのユーザーは、デフォルトでデータ辞書内のすべての情報にアクセスする権限を持っています。

正解: [\(正解を表示します\)](#)

質問: 72

このコマンドの一部を調べてみましょう。

```
CREATE TABLE cust (
  cust_id NUMBER(2),
  credit_limit NUMBER(10)
)
ORGANIZATION EXTERNAL
```

このコマンドを正常に実行するために必要な2つの句はどれですか？

- A. アクセスドライバのTYPE句
- B. 拒否制限条項
- C. アクセスパラメータ条項
- D. デフォルトディレクトリ条項
- E. LOCATION条項

正解: A,E ([コメントを發表する](#))

質問: 73

展示資料をご覧になり、PRODUCTS表のデータをご確認ください。

PRODUCTS	
PRODUCT ID	PRODUCT NAME
3054	Plasma Monitor
1782	Compact 400/DQ
1791	Industrial 700/HD
2302	Inkjet B/6
2459	LaserPro 1200/8/BW

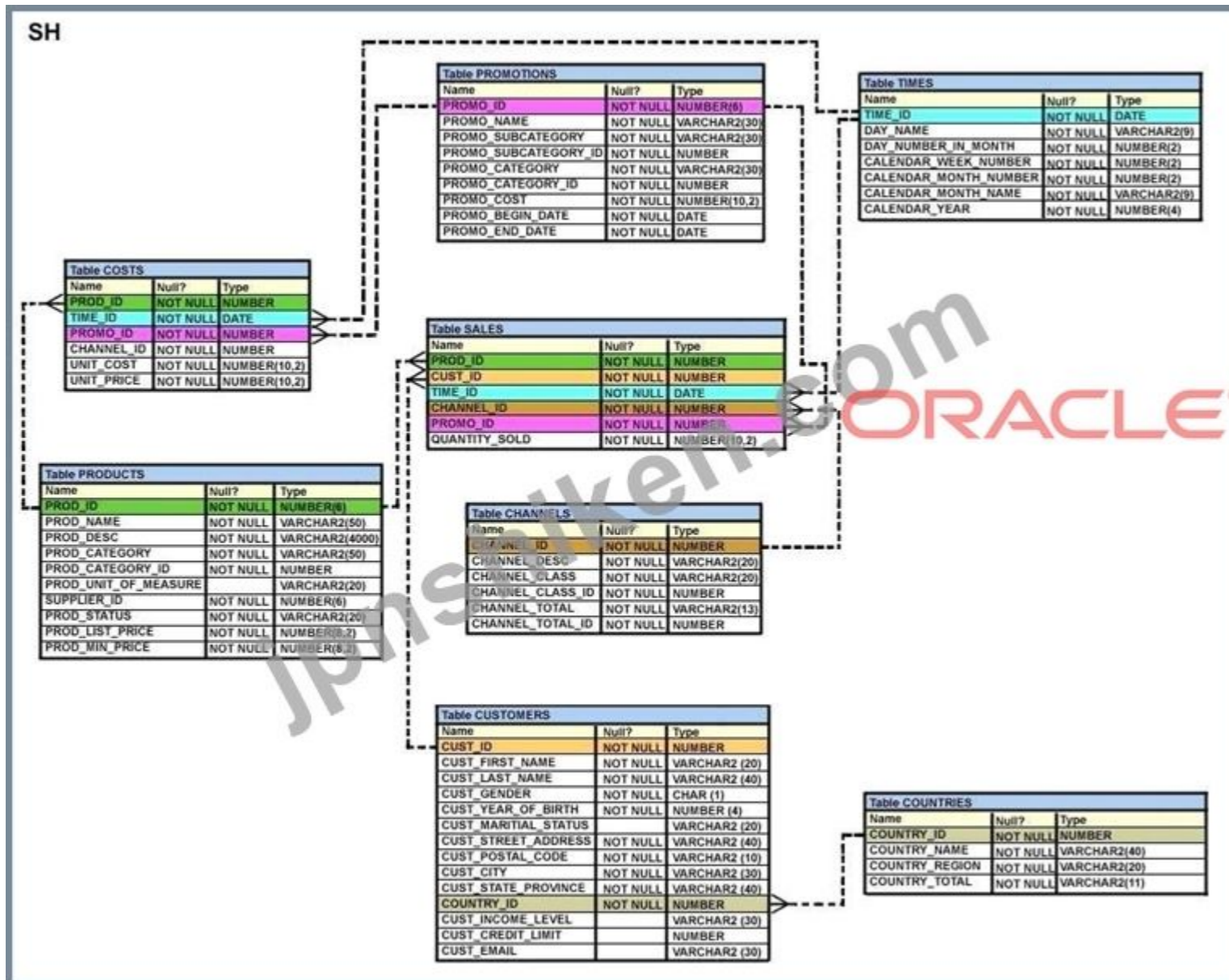
NULL値を格納できない「PRICE」という名前の列を追加するステートメントはどれですか？

- A. ALTER TABLE productsADD price NUMBER(8,2) DEFAULT CONSTRAINT p_nn NOT NULL.
- B. ALTER TABLE productsADD price NUMBER(8,2) NOT NULL;
- C. ALTER TABLE productsADD price NUMBER(8,2) DEFAULT 0 NOT NULL;
- D. ALTER TABLE productsADD price NUMBER(8,2) DEFAULT NOT NULL;

正解: C ([コメントを發表する](#))

質問: 74

展示資料をご覧になり、売上表と販売チャネル表の説明をご確認ください。



あなたは以下のSQL文を発行しました。

```
INSERT INTO SALES VALUES (23, 2300, SYSDATE,
    (SELECT CHANNEL_ID
     FROM CHANNELS
     WHERE CHANNEL_DESC='DIRECT SALES'),
    12, 4, 500);
```

結果に関して正しい記述はどれですか？（最も適切なものを選んでください。）

- A. サブクエリではVALUES句は不要なので、ステートメントは失敗します。
- B. VALUES句内のサブクエリが単一引用符で囲まれていないため、ステートメントは失敗します。
- C. VALUES句ではサブクエリを使用できないため、このステートメントは失敗します。
- D. このステートメントが実行され、SALES テーブルに新しい行が挿入されます。

正解: (正解を表示します)

質問: 75

展示資料をご覧になり、費用表とプロモーション表の構造をご確認ください。

プロモーション期間内で、プロモーション費用が最も高いPROD_IDよりも低いプロモーション費用を持つprod_idを表示したい。
このSQL文を調べてください。

```
SELECT prod_id
FROM costs
WHERE promo_id IN
  (SELECT promo_id
   FROM promotions
   WHERE promo_cost < ALL
     (SELECT MAX(promo_cost)
      FROM promotions
      GROUP BY (promo_end_date - promo_begin_date)));
```

結果はどのようなだろうか？

- A. 正常に実行され、必要な結果が得られます。
- B. GROUP BY句が無効であるため、エラーが発生します。
- C. all キーワードが無効であるため、エラーが発生します。
- D. 正常に実行されますが、必要な結果が得られません。

正解: [\(正解を表示します\)](#)

質問: 76

展示資料をご覧になり、プロモーション表の構造をご確認ください。

Table PROMOTIONS		
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(6)
PROMO_NAME	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY_ID	NOT NULL	NUMBER
PROMO_CATEGORY	NOT NULL	VARCHAR2(30)
PROMO_CATEGORY_ID	NOT NULL	NUMBER
PROMO_COST	NOT NULL	NUMBER(10,2)
PROMO_BEGIN_DATE	NOT NULL	DATE
PROMO_END_DATE	NOT NULL	DATE

次のSQL文を評価してください。

```
SQL>SELECT promo_name, CASE
      WHEN promo_cost >=(SELECT AVG(promo_cost)
        FROM promotions
        WHERE promo_category='TV')
      THEN 'HIGH'
      ELSE 'LOW'
      END COST_REMARK
FROM promotions;
```

上記のクエリの結果に関して、正しい記述はどれですか？

- A. 表内のすべてのプロモーションの COST_REMARK を表示します。
- B. サブクエリはCASE式で使用できないため、エラーが発生します。
- C. 「TV」というプロモーションカテゴリのすべてのプロモーションの COST_REMARK を表示します。
- D. サブクエリでエラーが発生するため、エラーが発生します。

正解: ([正解を表示します](#))

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%
ディスカウント、特別な割引コード: **JPNshiken**」

質問: 77

EMPLOYEES テーブルには、EMPLOYEE_ID、JOB_ID、SALARY などの列が含まれています。

EMPLOYEE_ID列のみがインデックス化されます。

従業員100番と200番に対応する行が存在します。

この記述を検討してください。

```
UPDATE employees
SET (job_id, salary) =
(SELECT job_id, salary
FROM employees
WHERE employee_id = 200)
WHERE employee_id = 100;
```

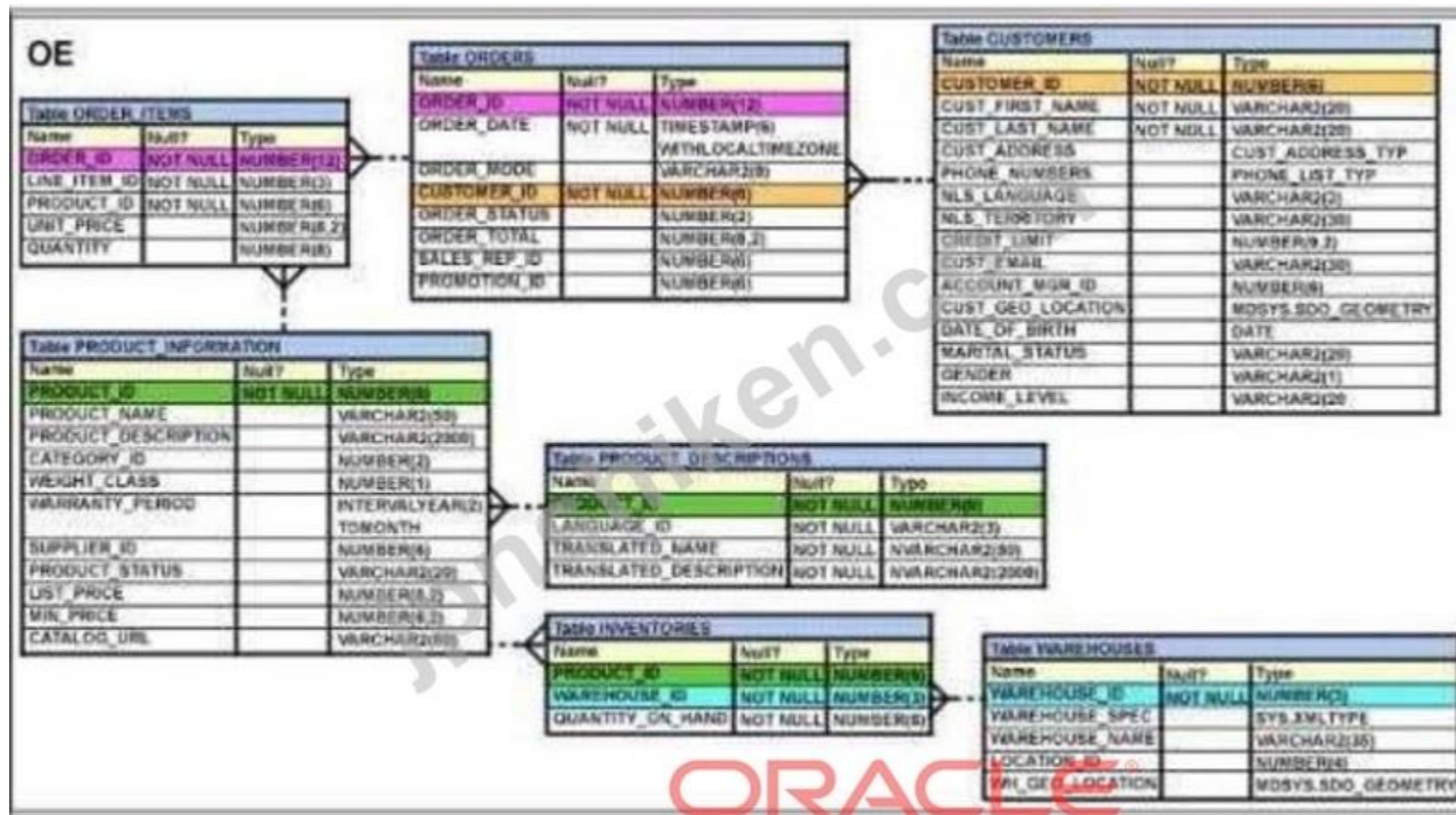
正しい記述はどれですか？ 2つ選択してください。）

- A. 従業員100の給与は、従業員200の給与と同じ値に設定されます。
- B. 従業員100のJOB_IDは、従業員200のJOB_IDと同じ値に設定されます。
- C. 従業員100番と200番の給与は、更新コマンド実行前と同じになります。
- D. 従業員100と200は、更新コマンド実行前と同じJOB_IDを持つことになります。
- E. 従業員200のJOB_IDは、従業員100のJOB_IDと同じ値に設定されます。
- F. 従業員200の給与は、従業員100の給与と同じ値に設定されます。

正解: **A,B** ([コメントを發表する](#))

質問: 78

展示資料を参照し、PRODUCT_INFORMATION テーブルの構造を調べてください。（最適な回答を選択してください。）



PRODUCT_ID列は主キーです。

インデックスを作成するには、次のコマンドを使用します。

```
SQL > CREATE INDEX upper_name_idx
```

```
ON product_information(UPPER(product_name));
```

PRODUCT_INFORMATIONテーブルには、他にインデックスは存在しません。

UPPER_NAME_IDX インデックスを使用するクエリはどれですか？

A. SELECT product_id

製品情報から

```
WHERE UPPER(product_name) IN ('LASERPRO', 'CABLE');
```

B. SELECT UPPER(product_name)

製品情報から;

C. SELECT product_id, UPPER(product_name)

製品情報から

```
WHERE UPPER(product_name) = 'LASERPRO' OR list_price > 1000;
```

D. SELECT UPPER(product_name)

製品情報から

```
WHERE product_id = 2254;
```

正解: **A** ([コメントを发表する](#))

質問: 79

次のCREATE TABLEコマンドを評価してください。

```

CREATE TABLE order_item
(order_id NUMBER (3),
item_id NUMBER (2),
qty NUMBER (4),
CONSTRAINT ord_itm_id_pk
PRIMARY KEY (order_id, item_id)
USING INDEX
(CREATE INDEX ord_itm_idx
ON order_item (order_id, item_id)));

```

上記のSQL文に関して、正しい記述はどれですか？

- A. 正常に実行され、ORD_ITM_IDX インデックスのみが作成されます。
- B. 複合プライマリーでは USING INDEX 句を使用できないため、エラーが発生します。
- C. 正常に実行され、ORD_ITM_IDX と ORD_ITM_ID PK という 2 つのインデックスが作成されます。
- D. CREATE TABLE コマンドでは USING INDEX が許可されていないため、エラーが発生します。

正解: [A \(コメントを發表する\)](#)

質問: 80

PRODUCTS テーブルのデータを調べてください。

PROD ID	PROD NAME	PROD LIST	CATEGORY ID
101	Plate	10	1
102	Cup	20	1
103	Saucer	20	1
104	Knife	30	1
105	Fork	30	1

以下のクエリを調べてください。

1. 製品名、製品リストを選択します

FROM 製品

WHERE prod_list NOT IN(10,20) AND category_id=1;

2. SELECT prod name, | prod_list

FROM 製品

WHERE prod_list <> ANY (10,20) AND category_id= 1;

SELECT 製品名、製品リスト

FROM 製品

WHERE prod_list <> ALL (10, 20) AND category_id=1;

どのクエリが同じ出力を生成しますか？

- A. 1と3
- B. 1、2、3
- C. 1と2
- D. 2と3

正解: [\(正解を表示します\)](#)

質問: 81

次のSQL文を評価してください。

```
SELECT product_name || '注文できません'
```

製品情報から

```
WHERE product_status = 'obsolete';
```

上記のクエリを実行中に、以下のエラーが発生しました。

エラー

ORA-01756: 引用符付き文字列が正しく終了していません

クエリを正常に実行するには、どのような手順を踏みますか？

- A. 引用符 (q) 演算子と区切り文字を使用して、リテラル文字列で単一引用符を使用できるようにします。
- B. SELECT句内のリテラル文字列を二重引用符で囲みます。
- C. SELECT句内の文字リテラル文字列を単一引用符で囲まないでください。
- D. SELECT句内のリテラル文字列内の単一引用符を否定するためにエスケープ文字を使用します。

正解: ([正解を表示します](#))

参考文献 :

http://docs.oracle.com/cd/B19306_01/server.102/b14200/sql_elements003.htm

質問: 82

指定された順序で実行される以下のSQL文を調べてください。

```
CREATE TABLE emp  
emp_no NUMBER(2) CONSTRAINT emp_emp_no_pk PRIMARY KEY  
ename VARCHAR2(15),  
salary NUMBER(8, 2),  
mgr_no NUMBER(2) CONSTRAINT emp_mgr_fk REFERENCES emp  
(emp_no));
```

```
ALTER TABLE emp  
DISABLE CONSTRAINT emp_emp_no_pk CASCADE;
```

```
ALTER TABLE emp  
ENABLE CONSTRAINT emp_emp_no_pk;
```

外部キーEMP_MGR_FKの状態はどうなりますか？

- A. 無効のままとなり、外部キー制約を削除して再作成することによってのみ有効にできます。
- B. 有効化され、即座に有効になります。
- C. 有効化され、延期されます。
- D. 無効のままとなり、手動で再度有効にすることができます。

正解: ([正解を表示します](#))

質問: 83

ORDERSテーブルとORDER_ITEMSテーブルの所有者であるユーザーSCOTTは、以下のGRANTコマンドを発行します。

すべて許可する

注文、注文アイテム

一般向け

その記述を修正するにはどうすればよいでしょうか？

- A. PUBLICは特定のユーザー名に置き換える必要があります。
- B. ALL は具体的な権限のリストに置き換える必要があります。
- C. WITH GRANT OPTION をステートメントに追加する必要があります。
- D. ORDERSテーブルとORDER_ITEMSテーブルには、それぞれ別のGRANTステートメントが必要です。

正解: ([正解を表示します](#))

説明

参考文献 :

<http://docs.oracle.com/javadb/10.8.3.0/ref/rrefsqjgrant.html>

質問: 84

このコマンドを実行します。

```
ALTER TABLE employees SET UNUSED (department_id);
```

正しいのは次のうちどれですか？

- A. クエリは、DEPARTMENT_ID 列のデータを表示できます。
- B. DEPARTMENT_ID 列が占有するストレージ領域は、COMMIT が発行された後にのみ解放されます。
- C. テーブル内のすべての牽引車に対して、DEPARTMENT_ID 列が null に設定されています
- D. EMPLOYEES テーブルに DEPARTMENT_ID という名前の新しい列を追加できます。
- E. DEPARTMENT_ID列のデータを更新することはできません。
- F. DEPARTMENT_ID列はごみ箱から復元できます

正解: ([正解を表示します](#))

D. 確かに、DEPARTMENT_ID 列を UNUSED に設定した後、EMPLOYEES テーブルに DEPARTMENT_ID という名前の新しい列を追加できます。UNUSED 句は列を削除するのではなく、使用されなくなったことを示すだけです。

E. 確かに、列が「未使用」とマークされると、その列を更新することはできません。DML操作ではアクセスできなくなります。

A、B、C、F は正しくありません。理由は次のとおりです。A. 列が UNUSED に設定されると、クエリに使用できなくなります。B. COMMIT を発行してもストレージ領域はすぐに解放されません。実際の削除と領域の解放は、その後 DROP UNUSED COLUMNS 操作を発行したときに行われます。C. DEPARTMENT_ID 列は null に設定されるのではなく、UNUSED としてマークされるため、使用できなくなります。F. UNUSED 列はごみ箱に入れられるのではなく、削除対象としてマークされるだけで、DROP UNUSED COLUMNS コマンドで領域を解放できます。

参照 :

OracleのALTER TABLEに関するドキュメント :Oracle Database SQL言語リファレンス ALTER TABLE SET UNUSEDの理解 :Oracle Database管理者ガイド

質問: 85

相関サブクエリで使用されるEXISTS演算子に関して、正しい記述は次のうちどれですか？ (2つ選択してください。)

- A. 外部クエリは、結果セット内のすべての値が処理されるまで、内部クエリの結果セットの評価を続けます。
- B. 内部クエリによって取得された値が外部クエリの結果に存在するかどうかをテストするために使用されます。
- C. 外部クエリによって取得された値が内部クエリの結果セットに存在するかどうかをテストするために使用されます。
- D. 外側のクエリは、最初の値が見つかった時点で内側のクエリの結果セットの評価を停止します。

正解: ([正解を表示します](#))

質問: 86

顧客テーブルは以下の構造になっています。

Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2(20)
CUST_LAST_NAME	NOT NULL	VARCHAR2(30)
CUST_INCOME_LEVEL		VARCHAR2(30)
CUST_CREDIT_LIMIT		NUMBER

以下のタスクを実行するクエリを作成する必要があります。

- 顧客の氏名（名と税額を表示します。税額は顧客の信用限度額の5%です。
- 収入レベルに価値のある顧客のみを対象とする。
- 税額がゼロの顧客は考慮対象としない。

どの記述が、要求されたすべてのタスクを達成しますか？

```
SELECT cust_first_name, cust_credit_limit * .05 AS TAX_AMOUNT
```

A. お客様から

```
WHERE (cust_income_level, tax_amount) IS NOT NULL;
```

B. お客様から

```
WHERE cust_income_level IS NOT NULL AND
```

cust_credit_limit は NULL ではありません。

```
SELECT cust_first_name, cust_credit_limit * .05 AS TAX_AMOUNT
```

C. お客様から

```
WHERE cust_income_level IS NOT NULL AND
```

tax_amount は NULL ではありません。

```
SELECT cust_first_name, cust_credit_limit * .05 AS TAX_AMOUNT
```

D. 顧客から

```
WHERE cust_income_level <> NULL AND
```

```
tax_amount <> NULL;
```

```
SELECT cust_first_name, cust_credit_limit * .05 AS TAX_AMOUNT
```

正解: [\(正解を表示します\)](#)

質問: 87

ユーザーANDREWに以下の権限を付与する必要があります。

- CUSTOMERSテーブルのTITLE列とADDRESS列を変更します。
- 他のユーザーにその権限を付与する。

どちらの文がこれを実現しますか？

A. GRANT UPDATE (title, address) ON customers TO andrew WITH ADMIN OPTION;

B. GRANT UPDATE ON customers. title, customers.address TO andrew WITH GRANT OPTION;

C. GRANT UPDATE ON customers.title, customers.address TO andrew WITH ADMIN OPTION;

D. GRANT UPDATE (title, address) ON customers TO andrew;

E. GRANT UPDATE ON customers. title, customers.address TO andrew;

F. GRANT UPDATE (title, address) ON customers TO andrew WITH GRANT OPTION:

正解: (正解を表示します)

A: その通りです。WITH ADMIN OPTION を指定した GRANT ステートメントを使用すると、権限付与を受けたユーザーは指定された列を変更できるだけでなく、その権限を他のユーザーに付与することもできます。

B :C、D、E、Fは誤りです。理由は以下のとおりです。

* BとCは構文が間違っているため誤りです。GRANTステートメントで複数の列を列挙する場合、正しい構文では各列の前にテーブル名を含めません。

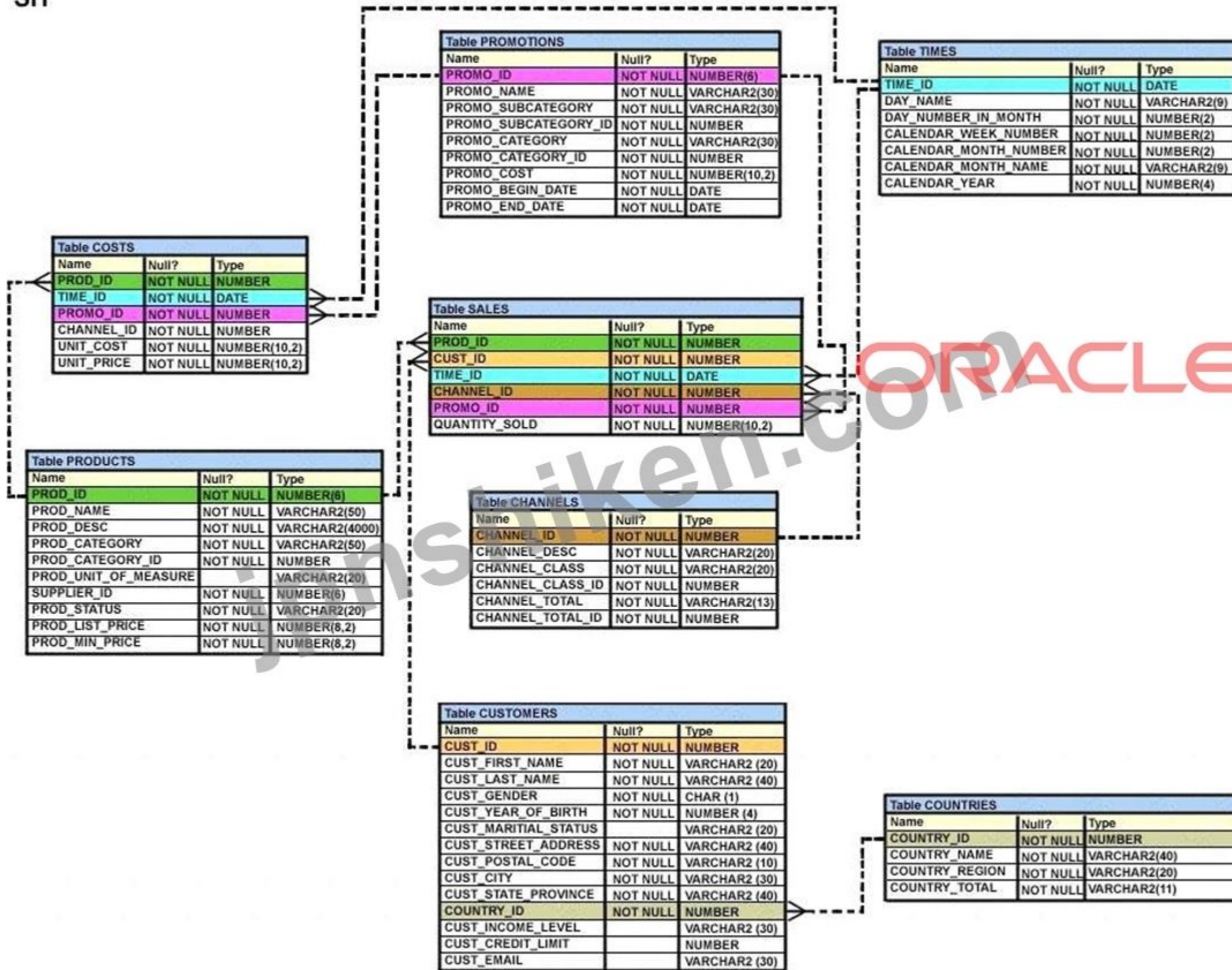
* DとEには、受給者が他者に許可を与えるために必要なWITH GRANT OPTIONが含まれていません。

* F は誤りです。キーワード WITH ADMIN OPTION は、権限付与を受けたユーザーがさらに他のユーザーに権限を付与できるようにするために使用されるべきであり、オブジェクト権限のコンテキストでは WITH GRANT OPTION が正しいです。

質問: 88

展示物をご覧になり、売上、顧客、製品、および時間の表の構造を調べてください。

SH



SALESテーブルのPROD_ID列は、PRODUCTSテーブルを参照する外部キーです。

SALESテーブルのCUST_ID列とTIME_ID列は、それぞれCUSTOMERSテーブルとTIMESテーブルを参照する外部キーでもあります。

このコマンドを調べてください。

```
CREATE TABLE new_sales (prod_id, cust_id, order_date DEFAULT SYSDATE)
```

として

```
SELECT prod_id, cust_id, time_id
```

売上から;

次の記述のうち、正しいものはどれですか？

A. CREATE TABLE コマンドと SELECT 句の列名が一致しないため、NEW_SALES テーブルは作成されません。

B. NEW_SALES テーブルが作成され、SALES テーブルから選択された列に定義されているすべての FOREIGN KEY 制約が、NEW_SALES テーブルの対応する列に作成されます。

C. 列定義でデフォルト値を指定できないため、NEW_SALES テーブルは作成されません。

D. NEW_SALES テーブルが作成され、SALES テーブルから選択された列に定義されているすべての NOT NULL 制約が、NEW_SALES テーブルの対応する列に作成されます。

正解: [\(正解を表示します\)](#)

質問: 89

展示物をご覧になり、売上およびプロモーションに関する表の説明をご確認ください。

SALESテーブルから、PROMOTIONSテーブルのPROMO_NAME列の値がblowout saleまたはeveryday low priceのいずれかである行を削除したい。

有効なDELETE文はどれですか？ 3つ選択してください。)

A. DELETE FROM sales WHERE promo_id = (SELECT promo_id FROM promotions WHERE promo_name = 'blowout sale') OR promo_name = 'everyday low price');

B. DELETE FROM sales WHERE promo_id = (SELECT promo_id FROM promotions WHERE promo_name = 'blowout sale') OR promo_id = (SELECT promo_id FROM promotions WHERE promo_name = 'everyday low price');

C. DELETE FROM sales WHERE promo_id = (SELECT promo_id FROM promo_name = 'blowout sale') AND promo_id = (SELECT promo_id FROM promotions WHERE promo_name = 'everyday low price') FROM promotions WHERE promo_name = 'everyday low price');

D. DELETE FROM sales WHERE promo_id IN (SELECT promo_id FROM promotions WHERE promo_name IN = 'blowout sale','everyday low price');

正解: **A,B,D** ([コメントを発表する](#))

質問: 90

展示内容を確認し、製品情報と在庫表の構造を調べてください。

資材部門から、在庫数が5未満のすべての製品について、製品ID、サプライヤーID、在庫数を含むリストを提出するよう求められています。

このタスクを実行できるSQL文はどれですか？ 2つ選択してください)

A. 製品ID、在庫数量、仕入先IDを選択してください

製品情報より

在庫と在庫数量が 5 未満であるものを自然結合します。

B. SELECT i. product_id, i. quantity_on hand, pi. supplier_id

FROM product information pi JOIN inventorys i USING (product id) AND quantity .on hand < 5;

C. SELECT i.製品ID、i.在庫数量、pi.サプライヤーID

FROM product_information pi JOIN inventorys i

ON (pi.製品ID=i.製品ID) AND 在庫数量 < 5;

D. SELECT i.製品ID、i.在庫数量、pi.サプライヤーID

FROM 製品情報 pi と在庫 i を結合

ON (pi.product id=i.product id)WHERE 在庫数量 < 5;

正解: **C,D** ([コメントを発表する](#))

質問: 91

*_TABLES ビューについて、正しい記述は次のうちどれですか？ (2つ選択)

- A. DBA TABLES でテーブルを表示するには、任意の TABLE システム権限を持っているか、テーブルに対するオブジェクト権限が付与されている必要があります。
- B. USER TABLESは、現在のユーザーが所有するすべてのテーブルを表示します。
- C. USER_TABLES でテーブルを表示するには、任意の TABLE システム権限を持っているか、テーブルに対するオブジェクト権限が付与されている必要があります。
- D. ALL TABLESは、現在のユーザーが所有するすべてのテーブルを表示します。
- E. ALL_TABLES でテーブルを表示するには、ANY TABLE システム権限を持っているか、テーブルに対するオブジェクト権限が付与されている必要があります。
- F. すべてのユーザーが DBA_TABLES を正常に照会できます。

正解: B,D (コメントを发表する)

Oracleでは、*_TABLESビューはテーブルに関する情報を提供します。

B. USER_TABLES は、現在のユーザーが所有するすべてのテーブルを表示するため、この記述は正しいです。自分のテーブルを表示するために追加の権限は必要ありません。

D. ALL_TABLES は、現在のユーザーが直接所有または権限を通じてアクセスできるすべてのテーブルを表示するため、このステートメントは真となります。

A、C、E、Fは誤りです。具体的には以下のとおりです。

AとEは誤りです。DBA_TABLESまたはALL_TABLES内のテーブルを表示するには、ANY TABLEシステム権限は必要ありません。SELECT_CATALOG_ROLEまたは同等の権限が必要です。

Cは誤りです。なぜなら、ユーザーとしてUSER_TABLES内の自分のテーブルを表示するために追加の権限は必要ないからです。

Fは誤りです。なぜなら、すべてのユーザーがDBA_TABLESを照会できるわけではなく、特定の権限または役割が必要だからです。

参照：

Oracle Databaseリファレンス、12cリリース1 (12.1)「静的データディクショナリビュー」

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%ディスカウント、特別な割引コード: **JPNshiken**」

質問: 92

どの2つのクエリが正常に実行されましたか？

A. SELECT prod_id, prod_name FROM products

交差する

SELECT 100, prod_name FROM newproducts;

B. SELECT prod_id FROM products

ユニオンオール

SELECT prod_id, prod_name FROM new_products;

C. SELECT prod_id, exp_date FROM products

ユニオンオール

SELECT prod_id, NULL FROM new_products;

D. 製品からk個を選択してください

マイナス

SELECT prod_id FROM new_products;

E. 製品から選択*

連合

```
SELECT * FROM new_products;
```

正解: C,E ([コメントを发表する](#))

質問: 93

制約条件の使用に関して、正しいのはどれですか？ (2つ選択)

- A. 子テーブルの外部キー列と、親テーブルの参照される主キー列は同じ名前であればなりません。
- B. テーブルには複数の主キー制約と複数の外部キー制約を設定できます。
- C. テーブルには、主キーと外部キーの制約をそれぞれ1つずつしか設定できません。
- D. PRIMARY KEYとFOREIGN KEY制約は、列レベルとテーブルレベルの両方で指定できます。
- E. テーブルには1つのPRIMARY KEYしか設定できませんが、FOREIGN KEY制約は複数設定できます。
- F. NOT NULLは列レベルとテーブルレベルで指定できます。

正解: ([正解を表示します](#))

Oracle Database 12cでは、制約の動作と特性を理解することが重要です。

- A. この記述は誤りです。子テーブルの外部キー列は、親テーブルの参照される主キー列と同じ名前である必要はありません。必要なのは、データ型が同じであることと、外部キー列の値が親テーブルの主キー列の値に対応していることです。
- B. この記述は誤りです。テーブルには複数のPRIMARY KEY制約を設定することはできません。定義上、PRIMARY KEYはテーブル内の行の一意の識別子であり、そのような識別子は1つしか存在できません。
- C. この記述はBと同じ理由で誤りです。テーブルには主キーは1つしか設定できません。ただし、他のテーブルの主キーを参照する外部キー制約は複数設定できます。
- D. これは正しいです。PRIMARY KEYとFOREIGN KEYの制約は、列定義を使用して列レベルで指定することも、ALTER TABLEステートメントを使用してテーブルレベルで指定することもできます。
- E. これは正しいです。テーブルには一意の行識別子を定義するため、PRIMARY KEY制約は1つしか設定できません。しかし、他のテーブルのキーを参照するFOREIGN KEY制約は複数設定できるため、複雑なリレーショナル構造を構築できます。
- F. この記述は誤りです。NOT NULL制約は個々の列に適用されるため、常に列レベルで定義されます。テーブルレベルで指定することはできません。

参照：

Oracleの制約に関するドキュメント: <https://docs.oracle.com/database/121/SQLRF/clauses002.htm#SQLRF52271> OracleのNOT NULL制約に関するドキュメント: <https://docs.oracle.com/database/121/SQLRF/clauses002.htm#SQLRF52162>

質問: 94

展示資料をご覧になり、ORDERSテーブルの構造をご確認ください。ORDER_ID列は、ORDERSテーブルの主キーです。

OE

Table ORDER_ITEMS		
Name	Null?	Type
ORDER_ID	NOT NULL	NUMBER(12)
LINE_ITEM_ID	NOT NULL	NUMBER(3)
PRODUCT_ID	NOT NULL	NUMBER(6)
UNIT_PRICE		NUMBER(8,2)
QUANTITY		NUMBER(8)

Table ORDERS		
Name	Null?	Type
ORDER_ID	NOT NULL	NUMBER(12)
ORDER_DATE	NOT NULL	TIMESTAMP(6) WITH LOCAL TIMEZONE
ORDER_MODE		VARCHAR2(8)
CUSTOMER_ID	NOT NULL	NUMBER(6)
ORDER_STATUS		NUMBER(2)
ORDER_TOTAL		NUMBER(8,2)
SALES_REP_ID		NUMBER(6)
PROMOTION_ID		NUMBER(6)

Table CUSTOMERS		
Name	Null?	Type
CUSTOMER_ID	NOT NULL	NUMBER(6)
CUST_FIRST_NAME	NOT NULL	VARCHAR2(20)
CUST_LAST_NAME	NOT NULL	VARCHAR2(20)
CUST_ADDRESS		CUST_ADDRESS_TYP
PHONE_NUMBERS		PHONE_LIST_TIP
NLS_LANGUAGE		VARCHAR2(3)
NLS_TERRITORY		VARCHAR2(30)
CREDIT_LIMIT		NUMBER(9,2)
CUST_EMAIL		VARCHAR2(30)
ACCOUNT_MGR_ID		NUMBER(6)
CUST_GEO_LOCATION		MDYS.SDO_GEOMETRY
DATE_OF_BIRTH		DATE
MARITAL_STATUS		VARCHAR2(20)
GENDER		VARCHAR2(1)
INCOME_LEVEL		VARCHAR2(20)

Table PRODUCT INFORMATION		
Name	Null?	Type
PRODUCT_ID	NOT NULL	NUMBER(6)
PRODUCT_NAME		VARCHAR2(50)
PRODUCT_DESCRIPTION		VARCHAR2(2000)
CATEGORY_ID		NUMBER(2)
WEIGHT_CLASS		NUMBER(1)
WARRANTY_PERIOD		INTERVAL YEAR(2) TO MONTH
SUPPLIER_ID		NUMBER(6)
PRODUCT_STATUS		VARCHAR2(20)
LIST_PRICE		NUMBER(8,2)
MIN_PRICE		NUMBER(8,2)
CATALOG_URL		VARCHAR2(50)

Table PRODUCT DESCRIPTIONS		
Name	Null?	Type
PRODUCT_ID	NOT NULL	NUMBER(6)
LANGUAGE_ID	NOT NULL	VARCHAR2(3)
TRANSLATED_NAME	NOT NULL	NVARCHAR2(50)
TRANSLATED_DESCRIPTION	NOT NULL	NVARCHAR2(2000)

Table INVENTORIES		
Name	Null?	Type
PRODUCT_ID	NOT NULL	NUMBER(6)
WAREHOUSE_ID	NOT NULL	NUMBER(3)
QUANTITY_ON_HAND	NOT NULL	NUMBER(8)

Table WAREHOUSES		
Name	Null?	Type
WAREHOUSE_ID	NOT NULL	NUMBER(3)
WAREHOUSE_SPEC		SYS.XMLTYPE
WAREHOUSE_NAME		VARCHAR2(35)
LOCATION_ID		NUMBER(4)
WH_GEO_LOCATION		MDYS.SDO_GEOMETRY

次のCREATE TABLEコマンドを評価してください。

```
CREATE TABLE new_orders(ord_id, ord_date DEFAULT SYSDATE, cus_id)
```

として

```
SELECT order_id,order_date,customer_id
```

注文から;

上記のコマンドに関して、正しい記述はどれですか？

- A. CREATE TABLE コマンドと SELECT 句の列名が一致しないため、NEW_ODRDERS テーブルは作成されません。
- B. NEW_ODRDERS テーブルが作成され、ORDERS テーブルの指定された列に定義されているすべての制約が新しいテーブルに渡されます。
- C. 列定義でデフォルト値を指定できないため、NEW_ODRDERS テーブルは作成されません。
- D. NEW_ODRDERS テーブルが作成され、指定された列に定義された NOT NULL 制約のみが新しいテーブルに渡されます。

正解: [\(正解を表示します\)](#)

展示資料をご覧になり、製品および売上表をご確認ください。

展示資料 1

Table PRODUCTS		
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER (6)
PROD_NAME	NOT NULL	VARCHAR2 (50)
PROD_DESC	NOT NULL	VARCHAR2 (4000)
PROD_CATEGORY	NOT NULL	VARCHAR2 (50)
PROD_CATEGORY_ID	NOT NULL	NUMBER
PROD_UNIT_OF_MEASURE		VARCHAR2 (20)
SUPPLIER_ID	NOT NULL	NUMBER (6)
PROD_STATUS	NOT NULL	VARCHAR2 (20)
PROD_LIST_PRICE	NOT NULL	NUMBER (8, 2)
PROD_MIN_PRICE	NOT NULL	NUMBER (8, 2)

展示資料2

Table SALES		
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER
CUST_ID	NOT NULL	NUMBER
TIME_ID	NOT NULL	DATE
CHANNEL_ID	NOT NULL	NUMBER
PROMO_ID	NOT NULL	NUMBER
QUANTITY_SOLD	NOT NULL	NUMBER (10, 2)

商品名とその商品の販売回数を表示するには、以下のクエリを実行します。

```
SQL>SELECT p.prod_name, i.item_cnt
FROM (SELECT prod_id, COUNT(*) item_cnt
FROM sales
GROUP BY prod_id) I RIGHT OUTER JOIN products p
ON i.prod_id = p.prod_id;
```

上記のステートメントが実行されると、何が起こりますか？

- A. ステートメントは正常に実行され、必要な出力が生成されます。
- B. FROM句のサブクエリではGROUP BY句を使用できないため、このステートメントはエラーになります。
- C. ITEM_CNT は外部クエリで表示できないため、ステートメントはエラーになります。
- D. FROM句内のサブクエリと外部結合は同時に使用できないため、このステートメントはエラーになります。

正解: (正解を表示します)

質問: 96

展示内容を確認し、PROJ_TASK_DETAILSテーブルのデータを調べてください。(最適な回答を選択してください。)

PROJ_TASK_DETAILS

TASK_ID	BASED_ON	TASK_IN_CHARGE	TASK_START_DATE	TASK_END_DATE
P01		KING	10-SEPT-07	12-SEPT-07
P02	P01	KOCHAR	13-SEPT-07	14-SEPT-07
P03		GREEN	14-SEPT-07	18-SEPT-07
P04	P03	SCOTT	19-SEPT-07	20-SEPT-07

PROJ_TASK_DETAILSテーブルには、プロジェクトのタスクとそれらの間の関係に関する情報が格納されます。

BASED_ON列は、タスク間の依存関係を示します。

一部のタスクは、他のタスクの完了に依存しない。

すべてのタスクID、そのタスクが依存するタスクのタスクID、およびそのタスクの担当者名を記載したレポートを作成する必要があります。

どのクエリを実行すれば、必要な結果が得られますか？

A. SELECT p.task_id, p.based_on, d.task_in_charge

proj_task_details p から proj_task_details d へのフル外部結合

ON (p.based_on = d.task_id);

B. SELECT p.task_id, p.based_on, d.task_in_charge

proj_task_details p から proj_task_details d を結合

ON (p.task_id = d.task_id);

C. SELECT p.task_id, p.based_on, d.task_in_charge

proj_task_details p から proj_task_details d を結合

ON (p.based_on = d.task_id);

D. SELECT p.task_id, p.based_on, d.task_in_charge

proj_task_details p から proj_task_details d への左外部結合

ON (p.based_on = d.task_id);

正解: [\(正解を表示します\)](#)

質問: 97

PROMOTIONSテーブルのPROMO_BEGTN_DATE列はデータ型がDATEで、デフォルトの日付形式はDD-MON-RRです。

クエリに含まれる PROMO_BEGIN_DATE を使用する式について、正しい記述は次のうちどれですか？ (2つ選択)

A. PROMO_BEGIN_DATE-SYSDATE はエラーを返します。

B. TO_NUMBER(PROMO_BEGIN_DATE)-5 は数値を返します

C. PROMO_BEGIN_DATE-5 は日付を返します。

D. TO_DATE(PROMO_BEGIN_DATE * 5) は日付を返します

E. PROMO_BEGIN_DATE-SYSDATE は数値を返します。

正解: [\(正解を表示します\)](#)

質問: 98

DEPARTMENT_DETAILS を作成するために使用されたコマンドを調べ、

コース詳細テーブル :

```
SQL>CREATE TABLE DEPARTMENT_DETAILS
(DEPARTMENT_ID NUMBER PRIMARY KEY,
DEPARTMENT_NAME VARCHAR2(50),
IOD VARCHAR2(50));

SQL>CREATE TABLE COURSE_DETAILS
(COURSE_ID NUMBER PRIMARY KEY,
COURSE_NAME VARCHAR2(50),
DEPARTMENT_ID NUMBER REFERENCES DEPARTMENT_DETAILS (DEPARTMENT_ID));
```

COURSE_DETAILSテーブルに存在しないすべての部署IDのリストを生成したい。

SQL文を実行します。

```
SQL> SELECT d.department_id FROM course_details c INNER JOIN
department_details d ON c.department_id<>d.department_id;
```

結果はどうなりますか？

- A. 正常に実行され、必要なリストが表示されます。
- B. ON句の条件が有効ではないため、失敗します。
- C. 正常に実行されますが、誤ったリストが表示されます。
- D. 使用されている結合タイプが間違っているため失敗します。

正解: [\(正解を表示します\)](#)

質問: 99

正しいのは次のうちどれですか？

- A. LAST_DAY は前月の最終日の日付のみを返します。
- B. CEILは、指定された数値以下の最大の整数を返します。
- C. LAST_DAY は、現在の月の最終日の日付のみを返します。
- D. ADD_MONTHS は、暗黙的に DATE データ型に変換できる文字列を扱います。
- E. ADD_MONTHS は、日付に暦月数を加算します。
- F. CEILでは、数値データ型の引数が必要です。
- G. LAST_DAY は、関数に渡された date 引数の月の最終日の日付を返します。

正解: [\(正解を表示します\)](#)

質問: 100

ビジネスルールを検証する :

各学生は複数のプロジェクトに取り組むことができ、各プロジェクトには複数の学生が参加できる。

最適なデータ保存を実現し、以下の形式でレポートを生成できるようにするために、エンティティ関係モデル (ERD) を設計する必要があります。

学生ID 名 姓 プロジェクトID プロジェクト名 プロジェクトタスク

このシナリオにおいて、正しい記述はどれですか？ (2つ選択)

- A. ER図では、STUDENTSエンティティとPROJECTSエンティティの間に1:Mの関係が必要です。
- B. ER図には、STUDENTSエンティティとPROJECTSエンティティ間の多対多の関係が必要であり、これは1対多の関係に解決されなければなりません。
- C. STUDENT_IDはSTUDENTSentityの主キーであり、PROJECTSentityの外部キーである必要があります。
- D. PROJECT_IDはPROJECTSentityの主キーであり、STUDENTSentityの外部キーである必要があります。
- E. STUDENT_IDとPROJECT_IDの複合キーを持つ関連テーブルを作成する必要があります。これは、STUDENTSエンティティとPROJECTSエンティティにリンクされた外部キーです。

正解: ([正解を表示します](#))

説明/参考資料:

参考文献:

<http://www.oracle.com/technetwork/issue-archive/2011/11-nov/o61sql-512018.html>

質問: 101

正常に実行される以下のステートメントを調べてください。

```
SELECT d.* --, e.avg_sal
FROM departments d
ORDER BY (SELECT AVG(e.salary) AS avg_sal
FROM employees e
WHERE e.department_id = d.department_id);

SELECT d.* --, e.avg_sal
FROM departments d
JOIN (SELECT department_id, AVG(salary) AS avg_sal
FROM employees
GROUP BY department_id) e
ON e.department_id = d.department_id
ORDER BY e.avg_sal;
```

どちらの表も、部門を平均給与順に並べています。

正しいのは次のうちどれですか？

- A. 従業員がいない部署は、2番目の文のみに表示されます。
- B. 最初のステートメントのみ、従業員がいない部署が表示されます。
- C. 選択リストに e.avg_sal を追加すると、両方のステートメントが正常に実行されます。
- D. どちらの文も従業員がいない部署を表示します。
- E. 選択リストに E.AVG_SAK を追加すると、最初のステートメントのみが正常に実行されます。
- F. 選択リストに E.AVG_SAL を追加すると、2番目のステートメントのみが正常に実行されます。

正解: **A,C** ([コメントを發表する](#))

* A. 2番目のステートメントのみが、従業員がいない部門を表示します。これは、2番目のステートメントが LEFT JOIN を使用して、従業員テーブルに一致するエントリがない部門も含め、部門テーブルからすべての部門を含めるためです。部門に従業員がいない場合、AVG(salary) は NULL になり、部門は引き続き表示されます。

* C. 選択リストに e.avg_sal を追加すれば、どちらのステートメントも正常に実行されます。これは正しいです。どちらのステートメントも、サブクエリまたは結合操作によって、e.avg_sal を平均給与として計算します。

選択リストに e.avg_sal を追加すると、部門ごとの平均給与が表示されます。ただし、最初のステートメントでは、従業員がいない部門は表示されないことに注意してください。これは、従業員テーブル2からNULL値を受け入れるための結合を使用していないためです。

参考文献:

* SQL JOIN の理解 - Stack Overflow1.

* OracleのJOIN操作に関するドキュメント2。

注: 他の選択肢は誤りです。理由は以下のとおりです。

* B. 最初のステートメントでは、従業員レコードが一致しない部門を含む結合を使用していないため、従業員がいない部門は表示されません。

* D. 前述のとおり、最初のステートメントでは従業員がいない部署は表示されません。

* E. オプションにタイプミスがあります。正しくは E.AVG_SAL です。修正したとしても、最初のステートメントだけでは avg_sal 値を取り込むための結合が含まれていないため、正常に実行されません。

* F. 2番目の文では従業員がいない部署が表示されますが、1番目の文では表示されません。したがって、この選択肢は誤りです。

質問: 102

Oracleデータベースについて正しい記述を2つ選びなさい。

- A. データのないNUMBER列はゼロ値になります。
- B. テーブルには複数の主キーを設定できます。
- C. テーブルには複数の外部キーを設定できます。
- D. 列定義では、複数のデータ型を指定できます。
- E. データのないVARCHAR2列にはNULL値があります。

正解: ([正解を表示します](#))

質問: 103

どの2つのステートメントが正常に実行されますか？

- A. SELECT TO_DATE(TO_CHAR('2019-DEC-25 03:30', 'YYYY-MON-DD HH12:MI')) FROM DUAL;
- B. SELECT TO_DATE('2019-DEC-25 15:30', 'YYYY-MON-DD HH24:MI', 'NLS_DATE_LANGUAGE = AMERICAN') DUALから;
- C. SELECT TO_CHAR('2019-DEC-25 15:30', 'YY-MON-D HH24:M2', 'NLS_DATE_LANGUAGE = AMERICAN') FROM DUAL;
- D. SELECT TO_CHAR('2019-DEC-25 15:30', 'YYYY-MON-DD HH24:MI') FROM DUAL
- E. SELECT TO_CHAR(TO_DATE('2019-DEC-25 03:30', 'YYYY-MON-DD HH12:MI')) FROM DUAL

正解: ([正解を表示します](#))

質問: 104

どの2つのクエリが文字列 Hello! we're ready を返しますか？

- A. SELECT q'[こんにちは！準備完了です]'FROM DUAL;
- B. SELECT "こんにちは！準備ができました"FROM DUAL;
- C. SELECT 'Hello! We're ready' FROM DUAL;
- D. SELECT 'Hello! we\ re ready' ESCAPE'N'FROM DUAL;
- E. SELECT q! こんにちは! 準備完了! 'FROM DUAL;

正解: ([正解を表示します](#))

質問: 105

展示内容を確認し、PROJ_TASK_DETAILSテーブルのデータを調べてください。(最適な回答を選択してください。)

PROJ_TASK_DETAILS

TASK_ID	BASED_ON	TASK_IN_CHARGE	TASK_START_DATE	TASK_END_DATE
P01		KING	10-SEPT-07	12-SEPT-07
P02	P01	KOCHAR	13-SEPT-07	14-SEPT-07
P03		GREEN	14-SEPT-07	18-SEPT-07
P04	P03	SCOTT	19-SEPT-07	20-SEPT-07

PROJ_TASK_DETAILSテーブルには、プロジェクトのタスクとそれらの間の関係に関する情報が格納されます。

BASED_ON列は、タスク間の依存関係を示します。

一部のタスクは、他のタスクの完了に依存しない。

すべてのタスクID、そのタスクが依存するタスクのタスクID、およびそのタスクの担当者名を記載したレポートを作成する必要があります。

どのクエリを実行すれば、必要な結果が得られますか？

- A. SELECT p.task_id, p.based_on, d.task_in_charge FROM proj_task_details p LEFT OUTER JOIN proj_task_details d ON (p.based_on = d.task_id);
- B. SELECT p.task_id, p.based_on, d.task_in_charge FROM proj_task_details p FULL OUTER JOIN proj_task_details d ON (p.based_on = d.task_id);
- C. SELECT p.task_id, p.based_on, d.task_in_charge FROM proj_task_details p JOIN proj_task_details d ON (p.based_on = d.task_id);
- D. SELECT p.task_id, p.based_on, d.task_in_charge FROM proj_task_details p JOIN proj_task_details d ON (p.task_id = d.task_id);

正解: [A \(コメントを發表する\)](#)

質問: 106

展示資料を参照し、ORDER_ITEMS テーブルの構造を確認してください。

ORDER_ID	LINE_ITEM_ID	PRODUCT_ID	UNIT_PRICE	QUANTITY
2355	4	2322	19	188
2355	5	2323	17	190
2355	9	2359	226.6	204
2355	1	2289	46	200
2356	5	2308	58	47
2356	6	2311	95	51
2356	1	2264	199.1	38
2356	2	2274	148.5	34
2356	3	2293	98	40
2356	4	2299	72	44
2357	2	2245	462	26
2357	3	2252	788.7	26
2357	4	2257	371.8	29
2357	5	2262	95	29

ORDER_ITEMS テーブル内のすべての注文の中で、合計金額が最も高い注文の ORDER_ID を選択する必要があります。

どのクエリを実行すれば、目的の結果が得られますか？

- A. SELECT order_id FROM order_items GROUP BY order_id HAVING SUM(unit_price*quantity) = (SELECT MAX(SUM(unit_price*quantity)) FROM order_items GROUP BY order_id);
- B. SELECT order_id FROM order_items WHERE (unit_price * quantity) = (SELECT MAX(unit_price * quantity) FROM order_items GROUP BY order_id)
- C. SELECT order_id FROM order_items WHERE (unit_price * quantity) = (SELECT MAX(unit_price*quantity) FROM order_items) GROUP BY order_id;
- D. SELECT order_id FROM order_items WHERE (unit_price * quantity) MAX(unit_price*quantity) GROUP BY order_id;

正解: [\(正解を表示します\)](#)

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%ディスカウント、特別な割引コード: **JPNshiken**」

質問: 107

データディクショナリに変更を加える2つのステートメントはどれですか？

- A. scott.emp から削除;
- B. スコットの従業員に対する助成金の更新を財務マネージャーに指示する。
- C. TRUNCATE TABLE emp;
- D. ALTER SESSION は NLs を設定します。_DATE フォーマット = 'DD/MM/YYYY';
- E. SELECT * FROM user_tab_privs;

正解: B,C ([コメントを發表する](#))

質問: 108

EMPLOYEES テーブルの説明を確認してください。

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER (38)
DEPARTMENT_ID	NOT NULL	NUMBER (38)
MANAGER_ID		NUMBER (38)

どの2つのクエリが、上司が別の部署に所属する従業員の行を返しますか？

- A. 従業員を選択してください。*

従業員から

WHERE manager_id NOT IN (

SELECT mgr.employee_id

従業員マネージャーより

WHERE emp.department_id <> mgr.department_id

);

- B. 従業員を選択*

従業員から

存在しない場合 (

NULLを選択します

従業員マネージャーより

WHERE emp.manager_id = mgr.employee_id

かつ emp.department_id <> mgr.department_id

);

- C. SELECT emp.*

従業員から

左折して従業員マネージャーに加わる

```
ON emp.manager_id = mgr.employee_id
```

```
AND 従業員部門ID <> 管理者部門ID;
```

D. 従業員を選択してください。*

従業員から

従業員マネージャーに右から参加する

```
ON emp.manager_id = マネージャー従業員ID
```

```
従業員部門ID <> 管理者部門ID
```

```
WHERE emp.employee_id IS NOT NULL;
```

E. 従業員を選択します。*

従業員から

従業員マネージャーに加わる

```
ON emp.manager_id = mgr.employee_id
```

```
従業員の部門IDと管理者の部門IDが一致しない。
```

正解: ([正解を表示します](#))

上司が別の部署に所属している従業員を検索するには、従業員の DEPARTMENT_ID と上司の DEPARTMENT_ID を比較するサブクエリまたは結合を使用できます。

A) このクエリは、NOT IN サブクエリが EMPLOYEE_ID と MANAGER_ID を誤って比較しようとしていること、およびサブクエリ内の関連条件が誤っているため、正しくありません。

B) このクエリは正しいです。NOT EXISTS 句は、MANAGER_ID が別の従業員 (mgr) の EMPLOYEE_ID と一致し、かつ DEPARTMENT_ID がそのマネージャーの DEPARTMENT_ID と異なる従業員を正しく識別します。

C) このクエリは、LEFT JOIN がすべての従業員を返すため、誤りです。また、マネージャーが同じ部署にいる従業員を除外するための WHERE 句がありません。

D) このクエリは誤りです。RIGHT JOIN では、結果の行がマネージャーが別の部署に所属する従業員のものであることを保証できません。また、すべてのマネージャーが返されるため、要件を満たしていません。

E) このクエリは正しいです。JOIN により、返される行は MANAGER_ID がマネージャー (mgr) の EMPLOYEE_ID と一致する従業員 (emp) の行であることが保証され、DEPARTMENT_ID がマネージャーの DEPARTMENT_ID と異なる従業員のみが含まれるように正しくフィルタリングされます。

質問: 109

このクエリを調べてください。

```
SELECT SUBSTR (SYSDATE,1,5) 'Result' FROM DUAL
```

次の記述のうち、正しいものはどれですか？

A. 式を TO_CHAR(SUNBSTR(SYSDATE,1,5)) に変更しない限り失敗します。

B. 式を SUBSTR (TO_CHAR(SYSDATE),1,5) に変更しない限り失敗します。

C. 式を SUBSTR (TO_CHAR(TRUNC(SYSDATE)),1,5) に変更しない限り失敗します。

D. 暗黙的なデータ型変換で正常に実行されます

正解: ([正解を表示します](#))

Oracle Database 12c では、SYSDATE 関数は現在のセッションの日付形式で日付値を返します。SUBSTR 関数は文字列から部分文字列を抽出するために使用されます。しかし、SYSDATE は DATE データ型であり、文字列ではないため、SUBSTR 関数を直接使用することはできません。したがって、SUBSTR 関数を適用する前に、まず TO_CHAR 関数を使用して SYSDATE を文字データ型に変換する必要があります。

したがって、クエリの正しい形式は次のようになります。

```
SELECT SUBSTR(TO_CHAR(SYSDATE), 1, 5) AS "Result" FROM DUAL;
```

これは、現在の日時をデフォルトの日付形式を使用して文字列に変換し、そこから最初の5文字を抽出します。

質問: 110

PROMOTIONSテーブルの構造を調べてください。

NAME	NULL?	TYPE
PROMO_ID	NOT NULL	NUMBER(6)
PROMO_NAME	NOT NULL	VARCHAR2(30)
PROMO_CATEGORY	NOT NULL	VARCHAR2(30)
PROMO_COST	NOT NULL	NUMBER(10,2)

経営陣は、各プロモーションカテゴリーにおける個別のプロモーション費用に関する報告書を必要としている。

この要件を満たすクエリはどれでしょうか？

- A. SELECT promo_category, DISTINCT promo_cost FROM promotions;
- B. SELECT DISTINCT promo_category, promo_cost FROM promotions ORDER BY 1;
- C. SELECT DISTINCT promo_cost, promo_category FROM promotions;
- D. SELECT DISTINCT promo_cost, DISTINCT promo_category FROM promotions;

正解: [B \(コメントを发表する\)](#)

質問: 111

CUSTOMERSテーブルの説明を確認してください。

Name	Null?	Type
CUSTNO	NOT NULL	NUMBER(3)
CUSTNAME	NOT NULL	VARCHAR2 (25)
CUSTADDRESS		VARCHAR2 (35)
CUST_CREDIT_LIMIT		NUMBER(5)

CUSTNOは主キーです。

重複している顧客名をすべてリストアップすることで、異なる顧客番号を使用して顧客情報が複数回入力されていないかどうかを確認する必要があります。

必要な結果を得るために使用できる2つの方法はどれですか？

- A. 左外側結合（自己結合）
- B. セルフジョイントによる外側引き抜きジョイント
- C. サブクエリ
- D. 右外側結合（自己結合）
- E. 自己参加

正解: [\(正解を表示します\)](#)

異なるCUSTNO値で入力された重複する顧客情報を検索するには、通常、同じテーブル内のレコードを比較する手法を使用します。

* C. サブクエリ: サブクエリを使用して、CUSTOMERS テーブルの顧客の詳細を比較し、重複を特定できます。このサブクエリは、異なる CUSTNO を持つ顧客の詳細の一致を検索できます。

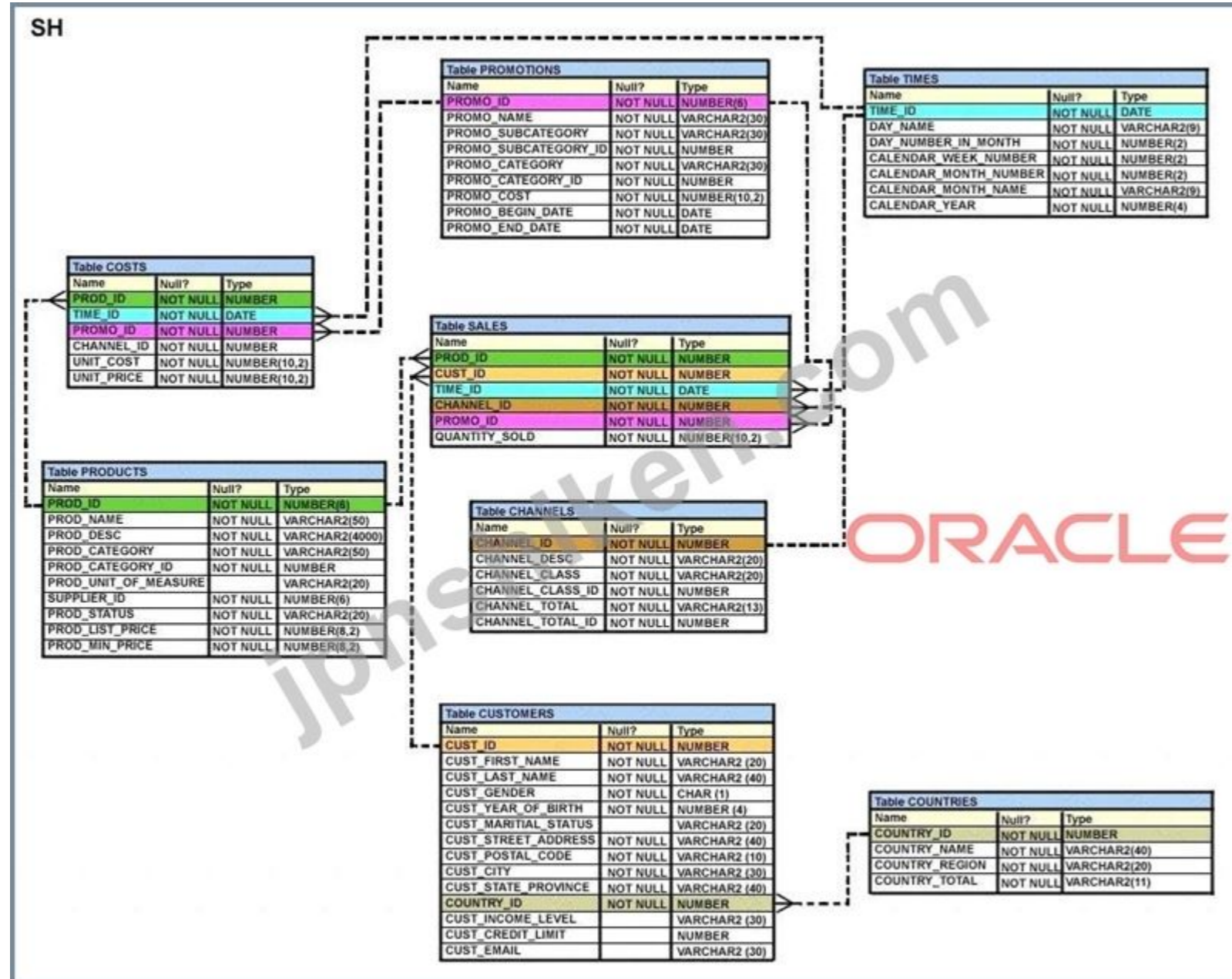
* E. 自己結合: この目的には自己結合も使用できます。これは、顧客の詳細を含む列で CUSTOMERS テーブルをそれ自身に結合して、同じ情報を持つが CUSTNO が異なる行を識別するものです。

参考文献:

* Oracle Database SQL Language Reference 12c、特に自己結合とサブクエリに関するセクションを参照してください。これらは、指定された条件に基づいて重複行を検索するために利用できます。

質問: 112

展示物をご覧になり、売上およびプロモーションに関する表の説明をご確認ください。



SALESテーブルから、PROMOTIONSテーブルのPROMO_NAME列の値がblowout saleまたはeveryday low priceのいずれかである行を削除したい。

有効なDELETE文はどれですか？ (3つ選択してください。)

- A. DELETE FROM sales WHERE promo_id = (SELECT promo_id FROM promotions WHERE promo_name = 'blowout sale') OR promo_name = 'everyday low price');
- B. DELETEFROM salesWHERE promo_id IN (SELECT promo_idFROM promotionsWHERE promo_name IN = 'blowout sale','everyday low price');
- C. DELETE FROM sales WHERE promo_id = (SELECT promo_id FROM promotions WHERE promo_name = 'blowout sale') OR promo_id = (SELECT promo_id FROM promotions WHERE promo_name = 'everyday low price')
- D. DELETE FROM sales WHERE promo_id = (SELECT promo_id FROM promo_name = 'blowout sale') AND promo_id = (SELECT promo_id FROM promotions WHERE promo_name = 'everyday low price') FROM promotions WHERE promo_name = 'everyday low price';

正解: (正解を表示します)

質問: 113

内部結合と外部結合について、正しい記述を3つ選びなさい。

- A. フル外部結合では、一致する行と一致しない行の両方が返されます。
- B. フル外部結合はOracle構文を使用する必要があります。
- C. 2つのテーブルに複数の結合条件がある場合、外部結合を使用できます。
- D. 外部結合は、クエリごとに2つのテーブル間でのみ使用できます。
- E. 内部結合は一致する行を返します。
- F. 左外部結合または右外部結合は、一致しない行のみを返します。

正解: ([正解を表示します](#))

A: その通りです。フル外部結合では、結合に関与する両方のテーブルから、一致した行と一致しなかった行の両方が返されます。これは、左外部結合と右外部結合の両方の結果を結合したものです。

E: 正しい。内部結合は、定義上、両方のテーブルで値が一致する行を返します。両方のテーブルで値が一致しない行は、内部結合の結果セットには含まれません。

内部結合は、結合条件に基づいて結合されたテーブルの行を照合しますが、外部結合は、一方のテーブルに一致する行が存在するかどうかに関係なく、一方または両方のテーブルのすべての行を含めます。

参照:

Oracle SQLのドキュメントでは、内部結合、左外部結合、右外部結合、完全外部結合など、さまざまな種類の結合について説明し、それらが生成する結果セットの違いを明確にしています。

質問: 114

どの2つのクエリが文字列 Hello! we're ready を返しますか？

- A. SELECT q! こんにちは! 準備完了! 'FROM DUAL;
- B. SELECT "こんにちは! 準備ができました" FROM |DUAL;
- C. SELECT q[こんにちは! 準備ができました]'FROM DUAL;
- D. SELECT 'Hello! we\ re ready' ESCAPE'N'FROMDUAL:
- E. SELECT 'Hello! We're ready' FROM DUAL;

正解: **A,C** ([コメントを發表する](#))

Oracle SQLでは、qクォート演算子を使用すると、シングルクォートやその他の特殊文字を含む文字列リテラルをエスケープすることなく定義できます。オプションAやCのようにqクォートメカニズムを使用するクエリは、文字列内のシングルクォートを含め、文字列をそのまま返します。

* A: その通りです。qクォート演算子と感嘆符!を区切り文字として使用することで、文字列にシングルクォートを含めることができます。

* B: 誤りです。Oracle SQLでは、二重引用符"は列名などの識別子に使用され、文字列リテラルには使用されません。

* C: その通りです。これもq引用演算子を使用し、角括弧[]を区切り文字として使用します。

* D: 誤りです。バックslash\はOracle SQLの文字列リテラルではエスケープ文字として使用されず、ESCAPEキーワードもここでは誤って使用されています。

* E: 誤りです。これは文字列内のシングルクォートを考慮しておらず、文字列リテラルが途中で終了してしまいます。また、qクォート演算子や適切なエスケープメカニズムがありません。

質問: 115

外部テーブルについて正しい記述は次のうちどれですか？ (2つ選択)

- A. それらにインデックスを作成できます。
- B. CREATE TABLE AS SELECT コマンドを使用して、データベース内の既存のデータからそれらにデータを入力できます。

- C. これらに対してはDMLステートメントを使用できません。
- D. データはSQLまたはPL/SQLのみを使用して取得できます。
- E. メタデータと実際のデータは両方ともデータベースの外に保存されます。

正解: ([正解を表示します](#))

Oracleの外部テーブルの場合 :

- * C. DML ステートメントは使用できません: Oracle では外部テーブルは読み取り専用です。外部テーブルを使用すると、データベース内のテーブルにあるかのように外部ソースのデータにアクセスできますが、DML 操作 (INSERT、UPDATE、DELETE) はサポートされません。
- * D. データは SQL または PL/SQL のみを使用して取得できます。通常のデータベース テーブルと同様に、SQL または PL/SQL を使用して外部テーブルのデータをクエリできます。誤った選択肢は以下のとおりです。
- * A: 外部テーブルに直接インデックスを作成することはできません。
- * B: 外部テーブルは、主にデータベース外に保存されているデータにアクセスするためのものであるため、CREATE TABLE AS SELECT コマンドを使用してデータを挿入することはできません。
- * E: 外部テーブルのメタデータのみがデータベースに保存され、実際のデータは外部ファイルに残ります。

質問: 116

展示資料を参照し、PRODUCT_INFORMATIONテーブルとINVENTORIESテーブルの構造を確認してください。

OE

Table ORDER_ITEMS		
Name	Null?	Type
ORDER_ID	NOT NULL	NUMBER(12)
LINE_ITEM_ID	NOT NULL	NUMBER(3)
PRODUCT_ID	NOT NULL	NUMBER(6)
UNIT_PRICE		NUMBER(8,2)
QUANTITY		NUMBER(8)

Table ORDERS		
Name	Null?	Type
ORDER_ID	NOT NULL	NUMBER(12)
ORDER_DATE	NOT NULL	TIMESTAMP(6) WITHLOCALTIMEZONE
ORDER_MODE		VARCHAR2(8)
CUSTOMER_ID	NOT NULL	NUMBER(6)
ORDER_STATUS		NUMBER(2)
ORDER_TOTAL		NUMBER(8,2)
SALES_REP_ID		NUMBER(6)
PROMOTION_ID		NUMBER(6)

Table CUSTOMERS		
Name	Null?	Type
CUSTOMER_ID	NOT NULL	NUMBER(6)
CUST_FIRST_NAME	NOT NULL	VARCHAR2(20)
CUST_LAST_NAME	NOT NULL	VARCHAR2(20)
CUST_ADDRESS		CUST_ADDRESS_TYP
PHONE_NUMBERS		PHONE_LIST_TYP
NLS_LANGUAGE		VARCHAR2(3)
NLS_TERRITORY		VARCHAR2(30)
CREDIT_LIMIT		NUMBER(9,2)
CUST_EMAIL		VARCHAR2(30)
ACCOUNT_MGR_ID		NUMBER(6)
CUST_GEO_LOCATION		MDYS.SDO_GEOMETRY
DATE_OF_BIRTH		DATE
MARITAL_STATUS		VARCHAR2(20)
GENDER		VARCHAR2(1)
INCOME_LEVEL		VARCHAR2(20)

Table PRODUCT INFORMATION		
Name	Null?	Type
PRODUCT_ID	NOT NULL	NUMBER(6)
PRODUCT_NAME		VARCHAR2(50)
PRODUCT_DESCRIPTION		VARCHAR2(2000)
CATEGORY_ID		NUMBER(2)
WEIGHT_CLASS		NUMBER(1)
WARRANTY_PERIOD		INTERVALYEAR(2) TOMONTH
SUPPLIER_ID		NUMBER(6)
PRODUCT_STATUS		VARCHAR2(20)
LIST_PRICE		NUMBER(8,2)
MIN_PRICE		NUMBER(8,2)
CATALOG_URL		VARCHAR2(50)

Table PRODUCT DESCRIPTIONS		
Name	Null?	Type
PRODUCT_ID	NOT NULL	NUMBER(6)
LANGUAGE_ID	NOT NULL	VARCHAR2(3)
TRANSLATED_NAME	NOT NULL	NVARCHAR2(50)
TRANSLATED_DESCRIPTION	NOT NULL	NVARCHAR2(2000)

Table INVENTORIES		
Name	Null?	Type
PRODUCT_ID	NOT NULL	NUMBER(6)
WAREHOUSE_ID	NOT NULL	NUMBER(3)
QUANTITY_ON_HAND	NOT NULL	NUMBER(8)

Table WAREHOUSES		
Name	Null?	Type
WAREHOUSE_ID	NOT NULL	NUMBER(3)
WAREHOUSE_SPEC		SYS.XMLTYPE
WAREHOUSE_NAME		VARCHAR2(35)
LOCATION_ID		NUMBER(4)
WH_GEO_LOCATION		MDYS.SDO_GEOMETRY

資材部門から、在庫数量が5未満のすべての製品について、PRODUCT_ID、SUPPLIER_ID、およびQUANTITY_ON_HANDを含むリストを提出するよう求められています。
このタスクを実行できるSQL文はどれですか？ 2つ選択してください。

- A. SELECT i.product_id, i.quantity_on_hand, pi.supplier_id FROM product_information pi JOIN inventories i ON (pi.product_id=i.product_id) AND quantity_on_hand < 5;
- B. SELECT i.product_id, i.quantity_on_hand, pi.supplier_id FROM product_information pi JOIN inventories i ON (pi.product_id=i.product_id) WHERE quantity_on_hand < 5;
- C. SELECT product_id, quantity_on_hand, supplier_id FROM product_information NATURAL JOIN inventories AND quantity_on_hand < 5;
- D. SELECT i.product_id, i.quantity_on_hand, pi.supplier_id FROM product_information pi JOIN inventories i USING (product_id) AND quantity_on_hand < 5;

正解: (正解を表示します)

質問: 117

給与が最も低い従業員3名の名前を返す2つの文はどれですか？

- A. SELECT 姓、給与

従業員より
最初の3行のみ取得する
給与順に並べ替える。

B. 姓、給与を選択

従業員より
給与順に並べ替え
最初の3つのRONのみFETCE;

C. SELECT last_name, salary

FBOMの従業員

取る

選択項目で並べ替え

ROINUM <= 3

給与は

従業員);

D. SELECT last_name, salary

から

(SELECT " FROM employees ORDER BY salary"

WHERE ROWNUM <=3

E. SELECT last_name, salary

従業員より

WHERE ROWNUM <=3

給与順に並べ替え

正解: [\(正解を表示します\)](#)

A :この記述は正しいです。従業員を給与順に並べ替え、最初の3行を取得します。

B: このステートメントには「FETCE」と「RONS」というタイプミスがあり、正しくは「FETCH」と「ROWS」です。そのため、正常に実行されません。

C: このステートメントは、構文エラーと「ORDER BY」句の誤った使用により、正常に実行されません。

D: この記述は正しいです。サブクエリを使用して従業員を給与順に並べ替え、ROWNUM擬似列を使用して結果を最初の3件に限定しています。

E: このステートメントは正常に実行されません。ROWNUMは「ORDER BY」句より前に評価されるため、最低給与ではなく、テーブルの自然順序に基づいて最初の3行が返されません。

ROWNUMの動作と「FETCH FIRST」構文については、『Oracle Database SQL言語リファレンス12c』で説明されています。

質問: 118

PRODUCT_STATUSテーブルの説明を確認してください。

Name	Null?	Type
PROD_ID	NOT NULL	NUMBER(2)
STATUS	NOT NULL	VARCHAR2(15)

「ステータス」列には、各行について「在庫あり」または「在庫切れ」の値が表示されます。

どの2つのクエリが正常に実行されますか？

ORACLE®
SELECT prod_id "CURRENT AVAILABILITY" || q'('s not available)' FROM
product_status WHERE status = 'OUT OF STOCK';

SELECT prod_id || q'('s not available)' "CURRENT AVAILABILITY" FROM
product_status WHERE status = 'OUT OF STOCK';

SELECT prod_id || q'('s not available)' FROM product_status WHERE status = 'OUT
OF STOCK';

SELECT prod_id || q"'s not available" FROM product_status WHERE status = 'OUT OF
STOCK';

SELECT prod_id || q'('s not available)' 'CURRENT AVAILABILITY' FROM
product_status WHERE status = 'OUT OF STOCK';

SELECT prod_id q's not available" FROM product_status WHERE status = 'OUT OF
STOCK';

- A. 選択肢D
- B. オプションF
- C. オプションA
- D. オプションC
- E. オプションB
- F. オプションE

正解: ([正解を表示します](#))

質問: 119

MERGE文について正しい記述は次のうちどれですか？ (2つ選択)

- A. WHEN MATCHED句を使用すると、実行する更新を指定できます。
- B. WHEN NOT MATCHED 句を使用すると、実行する更新を指定できます。
- C. WHEN NOT MATCHED 句を使用すると、実行する削除を指定できます。
- D. WHEN MATCHED句を使用すると、実行する挿入を指定できます。
- E. WHEN NOT MATCHED 句を使用すると、実行する挿入を指定できます。

正解: ([正解を表示します](#))

質問: 120

データ操作言語 (DML) について、正しい記述はどれですか？

- A. DML は、外部キー制約に対して ON DELETE CASCADE オプションが設定されている場合にのみ、親テーブルの主キー値を削除するときに外部キー制約を無効にします。
- B. トランザクションは、1つ以上のDMLステートメントで構成されます。
- C. 各DMLステートメントはデフォルトでトランザクションを形成します。
- D. DML は、親テーブルの主キー値を変更する際に、外部キー制約を自動的に無効にします。

正解: [\(正解を表示します\)](#)

質問: 121

展示資料をご覧になり、顧客表の構造を調べてください。

Table CUSTOMERS		
Name	Null?	Type
CUST_ID	NOT_NULL	NUMBER
CUST_FIRST_NAME	NOT_NULL	VARCHAR2(20)
CUST_LAST_NAME	NOT_NULL	VARCHAR2(20)
CUST_GENDER	NOT_NULL	CHAR(1)
CUST_YEAR_OF_BIRTH	NOT_NULL	NUMBER(4)
CUST_MARITAL_STATUS		VARCHAR2(20)
CUST_STREET_ADDRESS	NOT_NULL	VARCHAR2(40)
CUST_POSTAL_CODE	NOT_NULL	VARCHAR2(10)
CUST_CITY	NOT_NULL	VARCHAR2(30)
CUST_STATE_PROVINCE	NOT_NULL	VARCHAR2(40)
COUNTRY_ID	NOT_NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2(30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2(30)

次のSQL文を評価してください。

```
SQL> SELECT cust_city, COUNT(cust_last_name)
FROM customers
WHERE cust_credit_limit > 1000
GROUP BY cust_city
HAVING AVG(cust_credit_limit) BETWEEN 5000 AND 6000;
```

上記のクエリの結果に関して、正しい記述はどれですか？

- A. BETWEEN演算子はHAVING句では使用できないため、エラーが返されます。
- B. 正常に実行されました。
- C. WHERE句とHAVING句は同じSELECT文では使用できないため、エラーが返されます。
- D. WHERE句とHAVING句は同じ列に条件を適用するために使用できないため、エラーが返されます。

正解: [\(正解を表示します\)](#)

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%ディスカウント、特別な割引コード: **JPNshiken**」

質問: 122

複合クエリで使用されるINTERSECT演算子について、正しい記述はどれですか？

- A. INTERSECT は UNION または UNION ALL よりも優先順位が低い。
- B. NULL値を無視します。
- C. 選択された列のNULL値を処理します。
- D. 同じ SQL ステートメント内で複数の INTERSECT 演算子を使用することはできません。

正解: [\(正解を表示します\)](#)

質問: 123

BRICKSテーブルを調べてください。

COLOUR	SHAPE	WEIGHT
Red	cube	5
Red	cylinder	10
Blue	cube	15
Blue	cylinder	20

次のクエリを記述します。

選択

レングb1からレングb2を交差結合する

ただし、b1. 体重 < b2. 体重:

このクエリはいくつの行を返しますか？

- A. 0
- B. 16
- C. 10
- D. 4
- E. 1
- F. 6

正解: [F \(コメントを发表する\)](#)

質問: 124

役割に関する以下の記述のうち、正しいものはどれですか？ (3選)

- A. ロールはALTER ROLEステートメントを使用してロールに割り当てられます。
- B. ロールとは、関連する権限の名前付きグループであり、ユーザーにのみ割り当てることができます。
- C. ALTER USER ステートメントを使用して、ユーザーにロールを割り当てます。
- D. 1つの役割を複数のユーザーに割り当てることができます。

- E. 1人のユーザーに複数の役割を割り当てることができます
- F. 権限は、ALTER ROLE ステートメントを使用してロールに割り当てられます。
- G. GRANT ステートメントを使用して、ロールに権限が割り当てられます。

正解: D,E,G (コメントを发表する)

ロールとは、Oracleデータベースにおける権限の集合体を表す名称です。

- A. 誤り。ALTER ROLE ステートメントを使用して、ロールを他のロールに割り当てることはできません。
- B. 誤り。役割はユーザーと他の役割の両方に割り当てることができます。
- C. 正しい。ロールはALTER USERステートメントを使用してユーザーに割り当てられますが、これは唯一の方法ではありません。
- D. 正しい。1つの役割を複数のユーザーに割り当てることができ、ユーザー権限の管理が簡素化されます。
- E. 正しい。1人のユーザーに複数の役割を割り当てることができます。
- F. 誤り。ALTER ROLE ステートメントを使用してロールに権限を割り当てることはできません。
- G. 正しい。権限はGRANTステートメントを使用してロールに割り当てられます。

参照：

Oracleのユーザー権限とロールに関するドキュメント <https://docs.oracle.com/database/121/DBSEG/authorization.htm#DBSEG99914>

質問: 125

展示資料をご覧になり、PRODUCTS表の構造を調べてください。

Table PRODUCTS		
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER(6)
PROD_NAME	NOT NULL	VARCHAR2(50)
PROD_DESC	NOT NULL	VARCHAR2(4000)
PROD_CATEGORY	NOT NULL	VARCHAR2(50)
PROD_CATEGORY_ID	NOT NULL	NUMBER
PROD_UNIT_OF_MEASURE		VARCHAR2(20)
SUPPLIER_ID	NOT NULL	NUMBER(6)
PROD_STATUS	NOT NULL	VARCHAR2(20)
PROD_LIST_PRICE	NOT NULL	NUMBER(8,2)
PROD_MIN_PRICE	NOT NULL	NUMBER(8,2)

最も多くのアイテムを含むカテゴリを表示する必要があります。

あなたは次のクエリを発行します。

```
SQL > SELECT COUNT(*), prod_category_id
FROM products
GROUP BY prod_category_id
HAVING COUNT(*) =
(SELECT MAX(COUNT(*))
FROM products);
```

結果はどうなりますか？

- A. 正常に実行され、正しい出力が得られます。
- B. 正常に実行されますが、正しい出力が得られません。
- C. サブクエリにGROUP BY句がないため、エラーが発生します。
- D. = は無効であり、IN 演算子に置き換える必要があるため、エラーが発生します。

正解: C ([コメントを发表する](#))

質問: 126

展示資料をご覧になり、「プロモーション」表のデータをご確認ください。

PROMO_NAME	PROMO_CATEGORY	PROMO_COST	PROMO_BEGIN_DATE
NO PROMOTION #	NO PROMOTION	0	01-JAN-99
newspaper promotion #16-108	newspaper	200	23-DEC-00
post promotion #20-232	post	300	25-SEP-98
newspaper promotion #16-349	newspaper	400	10-JUL-98
internet promotion #14-471	internet	600	26-FEB-00
TV promotion #13-448	TV	1100	06-AUG-00
internet promotion #25-86	internet	1400	20-SEP-98
TV promotion #12-49	TV	1500	10-AUG-00
post promotion #21-166	post	2000	25-SEP-98
newspaper promotion #19-210	newspaper	2100	19-MAR-99
post promotion #20-282	post	2300	06-DEC-00
newspaper promotion #16-327	newspaper	2800	09-APR-99
internet promotion #29-289	internet	3000	01-NOV-98
TV promotion #12-252	TV	3100	20-JUN-98
magazine promotion #26-258	magazine	3200	04-MAY-00

PROMO_BEGIN_DATEは、デフォルトの日付形式であるdd-mon-rrで保存されます。

2000年1月1日より前に開始されたPOSTカテゴリのすべてのプロモーションについて、名称、費用、開始日を記載したレポートを作成する必要があります。

どのSQL文を使用しますか？

- A. SELECT promo_name, promo_cost, promo_begin_date FROM promotions WHERE promo_category LIKE 'P%' AND promo_begin_date < '1-JANUARY-00';
- B. SELECT promo_name, promo_cost, promo_begin_date FROM promotions WHERE promo_category
= 'post'かつ promo_begin_date < '01-01-00'未満であること。
- C. SELECT promo_name, promo_cost, promo_begin_date FROM promotions WHERE promo_cost LIKE
'post%' かつ promo_begin_date < '01-01-2000';
- D. SELECT promo_name, promo_cost, promo_begin_date FROM promotions WHERE promo_category LIKE '%post%' AND promo_begin_date < '1-JAN-00';

正解: D ([コメントを发表する](#))

質問: 127

以下の列仕様とデータ型を持つ SALES テーブルを作成する必要があります。(最適な回答を選択してください。) SALESID: 数値

店舗ID: 番号

アイテムID: 番号

数量：数値 値が指定されていない場合は1に設定する必要があります。

SLSDATE: 日付。値が指定されていない場合は、現在の日付に設定する必要があります。PAYMENT: 最大 30 文字の文字。値が指定されていない場合は、CASH に設定する必要があります。どのステートメントでテーブルが作成されますか？

A. テーブル販売の作成

(salesid 番号 (4)、
店舗ID番号 4)、
アイテムID番号 4)、
数量番号デフォルト1、
Slstartdate DATE DEFAULT SYSDATE、
payment VARCHAR2(30) DEFAULT 'CASH');

B. テーブルセールスを作成

(販売ID番号(4)、
店舗ID番号 4)、
商品ID番号 4)、
数量番号のデフォルト値 = 1、
SLSDATE DATE DEFAULT 'SYSDATE'、
支払い VARCHAR2(30) デフォルト現金);

C. テーブル 売上」を作成

(販売ID番号(4)、
店舗
番号 4)
商品ID番号 4)、
数量 数値 デフォルト値 = 1、
SLSDATE 日付 デフォルト SYSDATE、
支払い VARCHAR2(30) デフォルト値 = "現金");

D. CREATE TABLE Sales

(販売ID番号(4)、
店舗ID番号 4)、
商品ID番号 4)、
数量番号のデフォルト値 = 1、
SLSDATE 日付 デフォルト SYSDATE、
支払い変動
CHAR2(30) デフォルト値 = "CASH");

正解: ([正解を表示します](#))

質問: 128

次のクエリを評価してください。

SQL> SELECT TRUNC (ROUND (156.00, -1),-1)

デュアルから;

結果はどうなるだろうか？

- A. 150
- B. 200
- C. 160
- D. 16
- E. 100

正解: [C \(コメントを發表する\)](#)

参考文献 :

https://docs.oracle.com/cd/B19306_01/server.102/b14200/functions135.htm

https://docs.oracle.com/cd/B28359_01/olap.111/b28126/dml_functions_2127.htm

質問: 129

代入変数について正しい記述はどれですか？ (2つ選択してください。)

- A. 置換変数はSELECT文でのみ使用できます。
- B. 列名の入力を促すために使用する置換変数は、単一引用符で囲む必要があります。
- C. && で始まる置換変数は、セッション内で undefined に設定されていない限り、セッション内で一度だけ値の入力を求めます。
- D. 列名の入力を促すために使用する置換変数は、二重引用符で囲む必要があります。
- E. 置換変数は、SELECT文のどの句でも使用できます。
- F. & で始まる置換変数は、セッション中に常に一度だけ値の入力を求めます。

正解: [\(正解を表示します\)](#)

質問: 130

DEPARTMENT_DETAILS を作成するために使用されたコマンドを調べ、

コース詳細 :

対応する学科IDの有無に関わらず、すべてのコースIDを表示するレポートを作成したいが、コースが存在しない学科には学科IDを表示したくない。

どのSQL文を使用すべきですか？

- A. `SELECT c.course_id, d.department_id FROM course_details c FULL OUTER JOIN department_details d ON (c.department_id=d.department_id)`
- B. `SELECT c.course_id, d.department_id FROM course_details c RIGHT OUTER JOIN .department_details d ON (c.department_id=d.department_id)`
- C. `SELECT course_id, department_id, FROM department_details d RIGHT OUTER JOIN course_details c USING (department_id)`
- D. `SELECT c.course_id, d.department_id FROM course_details c FULL OUTER JOIN department_details d ON (c.department_id<>d.department_id)`

正解: [\(正解を表示します\)](#)

質問: 131

取引について正しい記述はどれですか？

- A. セーブポイントで終わる一連のデータ操作言語 (DML) ステートメントが連続して実行されると、単一のトランザクションが構成されます。
- B. 実行される各データ定義言語 (DDL) ステートメントは、単一のトランザクションを形成します。
- C. COMMITで終わる一連のDDLステートメントが連続して実行されると、単一のトランザクションが形成されます。
- D. DDL および DML ステートメントの組み合わせが、COMMIT で終わる一連のシーケンスで実行され、単一のトランザクションを形成します。

正解: [\(正解を表示します\)](#)

説明

参考文献 :

<https://docs.oracle.com/database/121/CNCPT/transact.htm#CNCPT038>

質問: 132

展示資料をご覧になり、ORDERSテーブルとCUSTOMERSテーブルの構造をご確認ください。

ORDERS		
Name	Null?	Type
ORDER_ID	NOT NULL	NUMBER(4)
ORDER_DATE	NOT NULL	DATE
ORDER_MODE		VARCHAR2(8)
CUSTOMER_ID	NOT NULL	NUMBER(6)
ORDER_TOTAL		NUMBER(8,2)

CUSTOMERS		
Name	Null?	Type
CUSTOMER_ID	NOT NULL	NUMBER(6)
CUST_FIRST_NAME	NOT NULL	VARCHAR2(20)
CUST_LAST_NAME	NOT NULL	VARCHAR2(20)
CREDIT_LIMIT		NUMBER(9,2)
CUST_ADDRESS		VARCHAR2(40)

cust_last_name列の値がRobertsである顧客は1人だけです。CUST_LAST_NAMEがRobertsで、CREDIT_LIMITが600である顧客の行をORDERSテーブルに追加するには、どのINSERT文を使用すればよいですか？

- A. INSERT INTO orders VALUES(1, '10-mar-2007', 'direct', (SELECT customer_id FROM customers WHERE cust_last_name='Roberts' AND credit_limit=600), 1000);
- B. INSERT INTO orders (order_id, order_data, order_mode, (SELECT customer_id FROM customers WHERE cust_last_name='Roberts' AND credit_limit=600), order_total) VALUES (1, '10-mar-2007', 'direct', &customer_id, 1000);
- C. INSERT INTO (SELECT o.order_id, o.order_date, o.order_mode, c.customer_id, o.order_total FROM orders o, customers c WHERE o.customer_id = c.customer_id AND c.cust_last_name='Roberts' AND c.credit_limit=600) VALUES (1, '10-mar-2007', 'direct', (SELECT customer_id FROM customers WHERE cust_last_name='Roberts' AND credit_limit=600), 1000);
- D. INSERT INTO orders (order_id, order_data, order_mode, (SELECT customer_id FROM customers WHERE cust_last_name='Roberts' AND credit_limit=600), order_total) VALUES (1, '10-mar-2007', 'direct', &&customer_id, 1000);

正解: [\(正解を表示します\)](#)

質問: 133

ORDERS テーブルには、日付型の DATE 列 ORDER_DATE があります。日付のデフォルトの表示形式は DD-MON-RR です。変換関数の正しい使用法を示す WHERE 条件は次の 2 つです。

- A. WHERE order_date > TO_CHAR(ADD_MONTHS(SYSDATE, 6), 'MON DD YYYY')
- B. WHERE TO_CHAR(order_date, 'MON DD YYYY') = 'JAN 20 2019';
- C. WHERE order_date > TO_DATE('JUL 10 2018', 'MON DD YYYY');
- D. WHERE order_date IN (TO_DATE ('Oct 21 2018', 'MON DD YYYY'), TO_CHAR('Nov 21 2018', 'MON DD YYYY'));

E. WHERE order_date > TO_DATE(ADD_MONTHS(SYSDATE,6),'MON DD YYYY');

正解: [\(正解を表示します\)](#)

SQLでは、日付に対する操作を行う際に、変換関数を正しく使用することが非常に重要です。Oracleでは、文字列を日付に変換するにはTO_DATE関数を、日付や数値を文字列に変換するにはTO_CHAR関数を使用します。

ステートメント C は正しいです。WHERE order_date > TO_DATE('JUL 10 2018','MON DD YYYY'); は、TO_DATE 関数の適切な使用例です。この関数は、文字列 'JUL 10 2018' を 'MON DD YYYY' 形式の日付型に変換し、それを order_date と比較します。

記述A、B、D、Eは誤りであるか、変換関数を誤用しています。

Aは誤りです。TO_CHARは日付や数値を文字列に変換するために使用されるものであり、その逆ではないため、order_dateと比較すべきではありません。

Bは誤りです。order_dateはDATE型であり、DATE型を文字列と比較する際には変換を行うべきではありません。ここではTO_CHARではなくTO_DATEを使用する必要があります。

Dは、TO_DATEとTO_CHARを同じINリストに混在させているため誤りです。INリストには日付型のみを含める必要があります。

Eは誤りです。TO_DATEは引数として文字列を受け取るべきであり、ADD_MONTHSが返す日付を受け取るべきではありません。

質問: 134

DESCRIBEコマンドについて、正しい記述を3つ選びなさい。

A. SQL Developerから使用できます。

B. 既存のビューの構造を表示するために使用できます。

C. SQL*Plusからのみ使用できます。

D. NOT NULL制約を持つ列に対して、その制約を表示します。

E. 各列に定義されているすべての制約を表示します。

F. 制約を持つ列または複数の列の主キー制約を表示します。

正解: [\(正解を表示します\)](#)

A: その通りです。DESCRIBEコマンドは、SQL Developerだけでなく、SQL*Plusなどの他のOracleデータベースツールからも使用できます。このコマンドは、テーブル、ビュー、またはその他のオブジェクトの構造を表示するために使用され、列名、データ型、列がNULL許容かどうかなどの情報が表示されます。

B: その通りです。DESCRIBEコマンドを使用すると、既存のビューの構造を表示でき、テーブルの場合と同様の情報が表示されます。これには、列、データ型、およびその他の関連する詳細情報が含まれます。

D: その通りです。DESCRIBEを使用すると、NOT NULL制約を持つ列の制約が表示されます。これは、テーブルまたはビューの構造に関する基本情報の一部であり、開発者がそこに格納されているデータの要件を理解するのに役立ちます。

質問: 135

MANAGERは、権限や役割を持たない既存の役割です。

EMPは、CREATE TABLE権限を含む既存のロールです。

EMPLOYEESは、HRスキーマに存在するテーブルです。

どの2つのコマンドが正常に実行されましたか？

A. GRANT SELECT, INSERT ON hr.employees TO manager WITH GRANT OPTION;

B. マネージャー、従業員にシーケンスの作成権限を付与します。

C. GRANT CREATE TABLE, emp TO manager;

D. GRANT CREATE TABLE, SELECT ON hr. employees TO manager;

E. マネージャーに任意のセッションの作成、任意のテーブルの作成権限を付与します。

正解: [\(正解を表示します\)](#)

質問: 136

この記述を検討してください。

紛失した注文を作成する

(sareal_no 一意の番号、

order_id 番号 主キー、

order_date DATE NOT NULL、

status VARCHAR2 (10) CHECK (status IN ('CREDIT', 'CASH')),

product_id 番号参照製品 (product_id)、

注文合計番号);

テーブルのどの2つの列に自動的にインデックスが作成されますか？

A. シリアル番号

B. 注文日

C. 製品ID

D. 注文合計

E. 注文ID

F. ステータス

正解: A,E (コメントを发表する)

UNIQUE制約とPRIMARY KEY制約が定義された列には、インデックスが自動的に作成されます。

A) 正しい。serial_no 列は UNIQUE 制約で定義されているため、Oracle はこの列に自動的にインデックスを作成します。

E) 正しい。order_id 列は PRIMARY KEY 制約で定義されており、Oracle は自動的に主キー列に一意インデックスを作成します。

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%
ディスカウント、特別な割引コード: **JPNshiken**」

質問: 137

単一行関数について正しい記述は次のうちどれですか？ (2つ選択)

A. CEIL: 正の数と負の数の両方に使用できます

B. MOD: 除算演算の商を返します

C. TRUNC: 数値と日付の値に使用できます

D. CONCAT: 任意の数の値を結合するために使用できます

E. FLOOR: 指定された数値以上の最小の整数を返します

正解: (正解を表示します)

質問: 138

単一行関数について正しい記述は次のうちどれですか？ (2つ選択)

A. CONCAT: 任意の数の値を結合するために使用できます

- B. MOD: 除算演算の商を返します
- C. CEIL: 正の数と負の数の両方に使用できます
- D. FLOOR: 指定された数値以上の最小の整数を返します
- E. TRUNC: NUMBER 値と DATE 値で使用できます

正解: [\(正解を表示します\)](#)

単一行関数は単一行に対して動作し、行ごとに1つの結果を返します。Oracle 12c SQLのコンテキストで各オプションを見ていきましょう。

- A. CONCAT: この関数は一度に2つの値しか結合できません。3つ以上の値を連結する必要がある場合は、CONCAT関数をネストするか、||演算子を使用する必要があります。
- B. MOD: MOD関数は除算演算の商ではなく、余りを返します。
- C. CEIL: この関数は、指定された数値以上の最小の整数を返します。正の数と負の数の両方に対応しています。
- D. FLOOR: 指定された数値以下で、指定された数値以上ではない最大の整数を返します。
- E. TRUNC: この関数は、NUMBER 値と DATE 値の両方で使用でき、指定した小数点以下の桁数、または日付の特定のコンポーネントに切り捨てることができます。

参照 :

Oracle Database SQL言語リファレンス 12cリリース1 (12.1)、関数

Oracle Database SQL言語リファレンス 12c リリース 1 (12.1)、CONCAT関数 Oracle Database SQL言語リファレンス 12c リリース 1 (12.1)、MOD関数 Oracle Database SQL言語リファレンス 12c リリース 1 (12.1)、CEIL関数とFLOOR関数 Oracle Database SQL言語リファレンス 12c リリース 1 (12.1)、TRUNC関数

質問: 139

演算子の優先順位に関する規則について、正しい記述を2つ選択してください。

- A. NULL値は式内の演算子の優先順位に影響します
- B. 等優先順位の算術演算子は、式内で左から右に評価されます。
- C. 連結演算子 || は、式の中で常に加算と減算の前に評価されます。
- D. 複数の括弧を使用すると、式内の演算子のデフォルトの優先順位を上書きできます。
- E. 二項演算子 #」は、SQL文の式において最も高い優先順位を持ちます。

正解: [D,E \(コメントを发表する\)](#)

質問: 140

EMPLOYEES テーブルには、EMPLOYEE_ID、JOB_ID、SALARY などの列が含まれています。

EMPLOYEES_ID列のみがインデックス化されます。

従業員100番と200番に対応する行が存在します。

この記述を検討してください。

従業員向け最新情報

```
SET (job_id, salary)
```

```
(SELECT job_id, salary
```

```
従業員より
```

```
WHERE employee_id = 200)
```

```
WHERE 従業員ID=100;
```

正しい記述は次のうちどれですか？ 2つ選択)

- A. 従業員100と200の給与は、更新コマンド実行前と同じになります。
- B. 従業員100の給与は、従業員200の給与と同じ値に設定されます。
- C. 従業員100のJOB_IDは、従業員200のJOB_IDと同じ値に設定されます。

- D. 従業員100と200は、更新コマンド実行前と同じJOB IDを持ちます。
- E. 従業員200の給与は、従業員100の給与と同じ値に設定されます。
- F. 従業員200のJOB_IDは、従業員100のJOB_IDと同じ値に設定されます。

正解: ([正解を表示します](#))

SQL UPDATE文は、テーブル内の既存のレコードを変更するために使用されます。この文では、EMPLOYEE_ID 200のJOB_IDとSALARYを選択し、これらの値をEMPLOYEE_ID 100の対応する列に設定するサブクエリが含まれています。

B. 従業員100の給与は、従業員200の給与と同じ値に設定されます。これは、サブクエリが従業員200の給与を選択し、外部の更新ステートメントがこの給与を従業員100に設定するためです。

C. 従業員100のJOB_IDは、従業員200のJOB_IDと同じ値に設定されます。同様に、サブクエリにより、従業員100のJOB_IDは従業員200のJOB_IDと一致するように更新されます。

参照：

Oracle Database SQL言語リファレンス12c、特にUPDATE文とサブクエリに関するセクションを参照してください。

質問: 141

INVOICEテーブルの構造を調べてください。

名前はnullですか？ タイプ

INV_NO NOT NULL NUMBER(3)

請求書日付

INV_AMT NUMBER(10,2)

どの2つのSQL文が正常に実行されるでしょうか？

A. SELECT inv_no, NVL2(inv_date, sysdate-inv_date, sysdate)

請求書から;

B. SELECT inv_no, NVL2(inv_amt, inv_amt*.25, '利用不可')

請求書から;

C. SELECT inv_no, NVL2(inv_amt, inv_date, '利用不可')

請求書から;

D. SELECT inv_no, NVL2(inv_date, '保留中', '未完了')

請求書から;

正解: ([正解を表示します](#))

質問: 142

Oracle 12cデータベースで、以下のコマンドを実行してSQL *Loaderを呼び出し、HRスキーマのEMPLOYEEStableにデータをロードしました。

```
$> sqlldr hr/hr@pdb table=employees
```

このコマンドに関して正しい記述はどれですか？ 2つ選択してください。)

A. HR に属する EMPLOYEEStable が既にデータベースに定義されている場合、デフォルト設定で成功します。

B. SQL *Loaderデータファイルの場所が指定されていないため、失敗します。

C. HR ユーザーに CREATE ANY DIRECTORY 権限がない場合、失敗します。

D. SQL *Loader制御ファイルの場所が指定されていないため失敗します。

正解: ([正解を表示します](#))

説明／参考資料：

質問: 143

視聴回数の減少について、正しい記述を2つ選びなさい。

- A. ビューの定義クエリによって選択されたデータは、ビューが削除されると、その基となるテーブルから削除されます。
- B. 読み取り専用ビューは削除できません。
- C. 削除するビューの作成者は、DROP ANY VIEW権限を持っている必要があります。
- D. CASCADE CONSTRAINTS削除対象のビュー内の主キーまたは一意キーを参照する他のオブジェクトの参照整合性制約を指定する必要があります。
- E. 削除されたビューを参照しているビューは無効になります。

正解: C,D (コメントを發表する)

ビューは独自のスキーマ内に存在する必要があるか、またはDROP ANY VIEWシステム権限を持っている必要があります。

CASCADE CONSTRAINTSを指定すると、削除対象のビュー内の主キーと一意キーを参照するすべての参照整合性制約が削除されます。

参考資料 https://docs.oracle.com/cd/B12037_01/server.101/b10759/statements_9009.htm

質問: 144

Oracleデータベースのビューに対してデータ操作言語 (DML) 操作を実行する際に正しい記述を3つ選択してください。

- A. ビューの定義クエリに DISTINCT キーワードが含まれている場合、ビューを使用して基になるテーブルの行を追加または変更することはできません。
- B. ビューの定義クエリに集計関数が含まれている場合、ビューを使用して基になるテーブルの行を追加または変更することはできません。
- C. WITH CHECK句は、ビューを介して基になるテーブルから行を削除する場合には効果がありません。
- D. テーブルに PRIMARY KEY があり、ビューの定義クエリで PRIMARY KEY 列が参照されていない場合、ビューを使用して基となるテーブルの行をクエリすることはできません。
- E. INSERT文はビューを介してテーブルに対して常に実行できます。
- F. ビューの定義クエリで参照されていないデフォルト値のない NOT NULL 制約を持つ列がテーブルにある場合、ビューを使用して基になるテーブルに行を追加することはできません。

正解: (正解を表示します)

質問: 145

次のSQL文を評価してください。

```
SQL> select cust_id, cust_last_name "姓"
```

お客様から

```
WHERE country_id = 10
```

連合

```
SELECT cust_id CUST_NO, cust_last_name
```

お客様から

```
WHERE country_id = 30
```

クエリを完了できるORDER BY句を3つ選びなさい。(3つ選択してください。)

- A. お客様番号による注文
- B. ORDER BY 2, cust_id
- C. 顧客番号」で注文

セット演算におけるORDER BY句の使用

ORDER BY句は、複合クエリの最後に一度だけ出現させることができます。

コンポーネントクエリには、個別のORDER BY句を含めることはできません。

ORDER BY句は、最初のSELECTクエリの列のみを認識します。

デフォルトでは、最初のSELECTクエリの最初の列が、出力を昇順にソートするために使用されます。

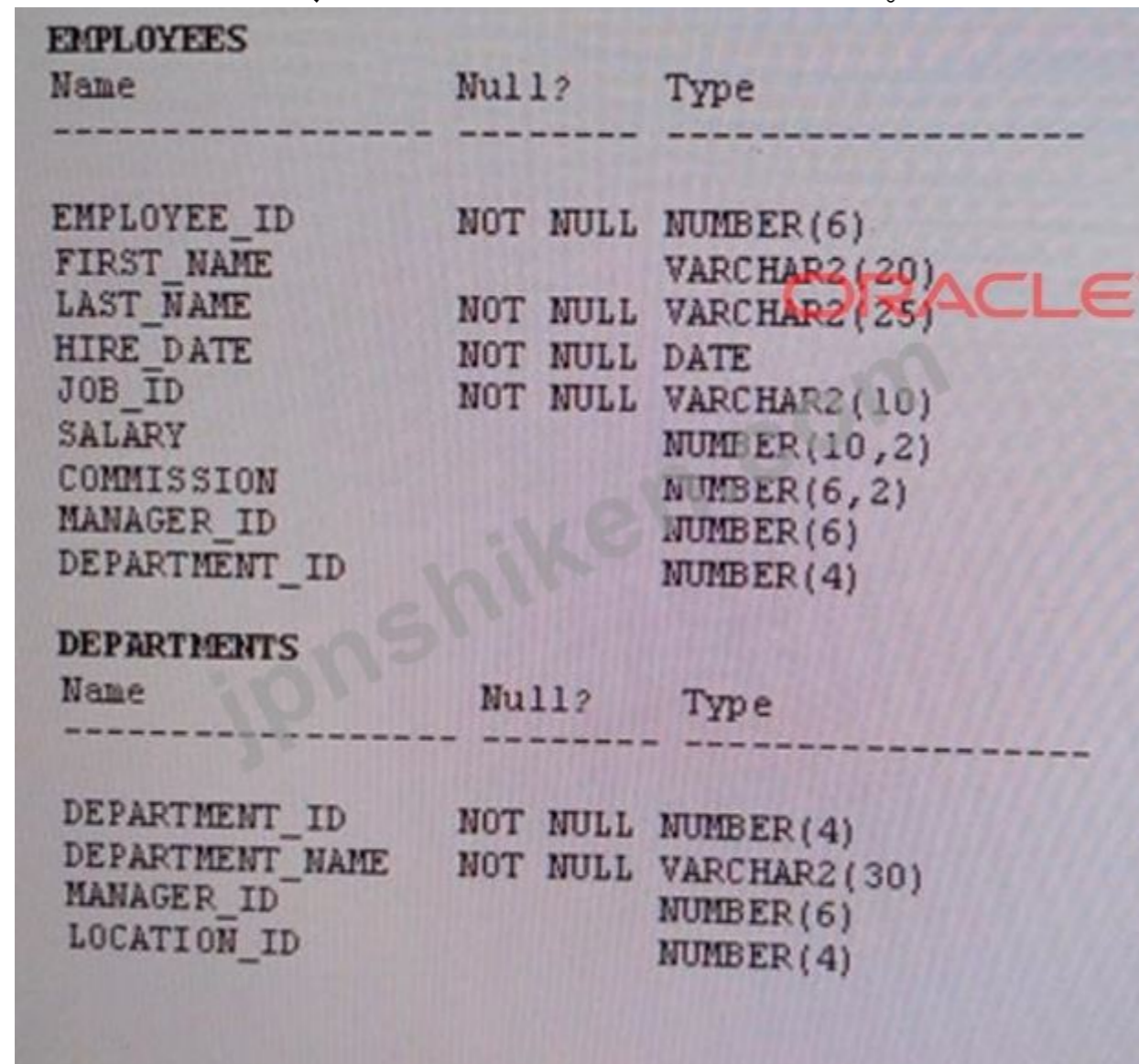
D. 姓で並べ替え

E. 2、1単位で注文

正解: B,D,E (コメントを發表する)

質問: 146

展示資料をご覧になり、従業員と部署の表の構造を調べてください。



```
EMPLOYEES
Name                Null?    Type
-----
EMPLOYEE_ID         NOT NULL NUMBER(6)
FIRST_NAME          VARCHAR2(20)
LAST_NAME           NOT NULL VARCHAR2(25)
HIRE_DATE           NOT NULL DATE
JOB_ID              NOT NULL VARCHAR2(10)
SALARY              NUMBER(10,2)
COMMISSION          NUMBER(6,2)
MANAGER_ID          NUMBER(6)
DEPARTMENT_ID       NUMBER(4)

DEPARTMENTS
Name                Null?    Type
-----
DEPARTMENT_ID       NOT NULL NUMBER(4)
DEPARTMENT_NAME     NOT NULL VARCHAR2(30)
MANAGER_ID          NUMBER(6)
LOCATION_ID           NUMBER(4)
```

以下の要件に従って従業員テーブルを更新する必要があります。

ボストンまたはシアトル (拠点番号900および2700)で勤務する従業員のみを更新してください。

-これらの従業員のdepartment_idを、ロンドン (locationid 2100)に対応する部署IDに設定します。

- location_id 2100 の従業員の給与を、その部署の平均給与の 1.1 倍に設定します。

-location_id 2100の従業員のコミッションを、その部署の平均コミッションの1.5倍に設定します。

このコマンドを実行します。

```

SQL> UPDATE employees
  SET department_id =
    (SELECT department_id
     FROM departments
     WHERE location_id = 2100),
    (salary, commission) =
    (SELECT 1.1*AVG(salary), 1.5*AVG(commission)
     FROM employees, departments
     WHERE departments.location_id IN(2900,2700,2100))
  WHERE department_id IN
    (SELECT department_id
     FROM departments
     WHERE location_id = 2900
     OR location_id = 2700);

```

結果はどうなりますか？

- A. 正常に実行され、目的の更新が生成されます。
- B. サブクエリは更新ステートメントで結合条件を持つことができないため、エラーが発生します。
- C. UPDATE文では複数の列を同時に指定できないため、エラーが発生します。
- D. 正常に実行されますが、目的の更新は生成されません。

正解: [D \(コメントを公表する\)](#)

質問: 147

ORDER_ITEMSテーブルの説明を確認してください。

Name	Null	Type
ORDER_ID		NUMBER(38)
PRODUCT_ID		NUMBER(38)
QUANTITY		NUMBER(38)
UNIT_PRICE		NUMBER(10,2)

この不完全なクエリを調べてください。

SELECT DISTINCT quantity * unit_price total_paid FROM order_items ORDER BY <clause>; クエリが正常に完了するように、<clause> を置き換えることができる2つのオプションはどれですか？

- A. 数量
- B. 数量、単価
- C. 支払総額
- D. product_id
- E. 数量×単価

正解: [\(正解を表示します\)](#)

DISTINCT句を含むSELECT文では、ORDER BY句はSELECTリストに含まれる式のみで並べ替えることができます。

- A) 数量だけでは <句> を置き換えるには不十分です。なぜなら、DISTINCT の後の SELECT リストに含まれていないからです。
- B) このオプションは、SELECT 式で quantity と unit_price の両方が使用されているため、<clause> を正常に置き換えることができます。したがって、それらの個々の値は ORDER BY 句で有効です。

C). total_paid は quantity * unit_price という式の別名ですが、Oracle では DISTINCT クエリの式の別名を ORDER BY 句で使用することは許可されていないため、ORDER BY 句では使用できません。

D). product_id は DISTINCT の後の SELECT リストに含まれていないため、ORDER BY では使用できません。

E) 式 quantity * unit_price はまさに選択されているものなので、<clause> を置き換えることができ、クエリは正常に完了します。

参考文献：

* Oracle Database SQL言語リファレンス、12cリリース1 (12.1) "ORDER BY句"

質問: 148

この部分的な記述を検討してみましょう。

```
SELECT *
FROM employees
WHERE salary = (<subquery>);
```

どちらが正しいですか？

- A. クエリとサブクエリの両方で任意の数の行を選択できます。
- B. クエリでは0行または1行しか選択できませんが、サブクエリでは任意の数の行を選択できます。
- C. クエリとサブクエリの両方で、0行または1行のみを選択できます。
- D. クエリは任意の数の行を選択できますが、サブクエリは0行または1行しか選択できません。

正解: ([正解を表示します](#))

説明／参考資料 <http://dcx.sybase.com/1200/en/dbusage/ug-subquery-s-4318996.html>

質問: 149

SALESテーブルには、データ型がNUMBERの列PROD_IDとQUANTITY_SOLDがあります。

正常に実行されるクエリはどれですか？ 2つ選択してください。)

- A. カウント(*) > 10;
SELECT prod_id FROM sales WHERE quantity_sold > 55000 AND COUNT(*) > 10 GROUP
- B. カウント(*) > 10;
SELECT COUNT(prod_id) FROM sales GROUP BY prod_id WHERE quantity_sold > 55000;
- C. SELECT prod_id FROM sales WHERE quantity_sold > 55000 AND COUNT(*) > 10 GROUP
- D. BY prod_id HAVING COUNT(*) > 10;
SELECT COUNT(prod_id) FROM sales WHERE quantity_sold > 55000 GROUP BY prod_id;
- E. SELECT prod_id FROM sales WHERE quantity_sold > 55000 GROUP BY prod_id HAVING

正解: ([正解を表示します](#))

質問: 150

TRANSACTION テーブルを作成するために使用された SQL ステートメントを調べます。(最適な回答を選択してください。) SQL > CREATE TABLE transaction (trn_id char(2) primary key, Start_date date DEFAULT SYSDATE, End_date date NOT NULL); このテーブルの trn_id には値 'A1' が存在しません。

どのSQL文を使えば、START_DATEにデフォルト値を使用してテーブルに行を正常に挿入できますか？

- A. INSERT INTO transaction VALUES ('A1', DEFAULT, TO_DATE(DEFAULT+10))
- B. INSERT INTO transaction (trn_id, start_date, end_date) VALUES ('A1', , '10-DEC-2014')
- C. INSERT INTO transaction (trn_id, end_date) VALUES ('A1', '10-DEC-2014')

D. INSERT INTO transaction VALUES ('A1', DEFAULT, TO_DATE('SYSDATE+10'))

正解: ([正解を表示します](#))

質問: 151

ユーザー名に 'ch_' という文字列が含まれるすべてのユーザーの詳細を表示する必要があります。(最適な回答を選択してください。) 必要な出力を生成するクエリはどれですか？

A. SELECT * FROM users

ユーザー名が 'ch_' に一致する場合。

B. SELECT * FROM users

ユーザー名が 'ch_%' に一致し、 '_' がエスケープされている箇所。

C. SELECT * FROM users

ユーザー名が 'ch_%'ESCAPE%' に一致する場合。

D. SELECT * FROM users

ユーザー名が 'ch_%' に一致し、 '_' をエスケープする。

正解: ([正解を表示します](#))

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%ディスカウント、特別な割引コード: **JPNshiken**

質問: 152

展示資料を参照してください。

BOX_SIZE	MIN_WEIGHT	MAX_WEIGHT
SMALL		0

どの2つのクエリがCUBEのみを返しますか？

A. SELECT shape FROM bricks JOIN boxes ON weight BETWEEN min_weight AND max_weight;

B. SELECT shape FROM bricks JOIN boxes ON NOT (weight > max_weight);

C. SELECT shape FROM bricks JOIN boxes ON weight < max_weight;

D. SELECT shape FROM bricks JOIN boxes ON weight > min_weight;

E. SELECT shape FROM bricks JOIN boxes ON weight >= min_weight AND weight < max_weight;

正解: ([正解を表示します](#))

質問: 153

ビジネスルールを検証する：

各学生は複数のプロジェクトに取り組むことができ、各プロジェクトには複数の学生が参加できる。

最適なデータ保存を実現し、以下の形式でレポートを生成できるようにするために、エンティティ関係モデル (ERD) を設計する必要があります。

学生ID 名 姓 プロジェクトID プロジェクト名 プロジェクトタスク

この状況において正しい記述はどれですか？ 2つ選択してください。)

- A. ER図では、STUDENTSエンティティとPROJECTSエンティティの間に1:Mの関係が必要です。
- B. ER図には、STUDENTSエンティティとPROJECTSエンティティ間のM:M関係が存在し、それが1:M関係に解決される必要があります。
- C. STUDENT_ID は STUDENTS エンティティの主キーであり、PROJECTS エンティティの外部キーである必要があります。
- D. PROJECT_ID は PROJECTS エンティティの主キーであり、STUDENTS エンティティの外部キーである必要があります。
- E. STUDENT_IDとPROJECT_IDの複合キーを持つ関連テーブルを作成する必要があります。これは、STUDENTSエンティティとPROJECTSエンティティにリンクされた外部キーです。

正解: [B,E \(コメントを发表する\)](#)

説明

<http://www.oracle.com/technetwork/issue-archive/2011/11-nov/o61sql-512018.html>

質問: 154

コマンドを調べてください。

```
SQL> ALTER TABLE books_transactions
      ADD CONSTRAINT fk_book_id FOREIGN KEY (book_id)
      REFERENCES books (book_id) ON DELETE CASCADE;
```

ON DELETE CASCADEとはどういう意味ですか？

- A. BOOKS テーブルが削除されると、BOOK_TRANSACTIONS テーブルのすべての行が削除されますが、テーブル構造は保持されます。
- B. BOOKS テーブルの行が削除されると、BOOKS テーブルの削除された行の BOOK_ID と一致する BOOK_TRANSACTIONS テーブルの行も削除されます。
- C. BOOKS テーブルが削除されると、BOOK_TRANSACTIONS テーブルも削除されます。
- D. BOOKS.BOOK_ID列の値が削除されると、BOOKS_TRANSACTIONS.BOOK_ID列の対応する値が更新されます。

正解: [\(正解を表示します\)](#)

質問: 155

OracleのJOIN構文とANSIのJOIN構文について、正しい記述を3つ選びなさい。

- A. Oracleの結合構文は右外部結合のみをサポートしています。
- B. Oracleの結合構文は、2つのテーブルの直積の作成をサポートしています。
- C. SQL:1999 準拠の ANSI 結合構文は、自然結合をサポートします。
- D. Oracleの結合構文は自然結合をサポートしています。
- E. Oracleの結合構文は、SQL:1999準拠のANSI結合構文よりも優れたパフォーマンスを発揮します。
- F. SQL:1999 準拠の ANSI 結合構文は、2つのテーブルの直積の作成をサポートします。
- G. Oracle の JOIN 構文は、SQL:1999 に準拠した ANSI JOIN 応答よりもパフォーマンスが劣ります。

正解: [\(正解を表示します\)](#)

OracleのJOIN構文とANSIのJOIN構文について :

- B. Oracleの結合構文は、2つのテーブルの直積の作成をサポートしています。これは正しいです。Oracleでは、FROM句に結合条件を指定せずにテーブルを列挙すると、直積が作成されます。
- C. SQL:1999 に準拠した ANSI 結合構文は、自然結合をサポートしています。これは正しいです。ANSI 構文は自然結合をサポートしており、結合するテーブル内で同じ名前の列に基づいてテーブルを結合します。
- F. SQL:1999 に準拠した ANSI 結合構文は、2つのテーブルの直積の作成をサポートしています。これは正しいです。ANSI 標準では、FROM 句に結合条件なしでテーブルがリストされている場合、直積が許可されます。

選択肢A、D、E、Gは誤りです。

Aは誤りです。Oracleの結合構文は、右外部結合を含むすべての種類の結合をサポートしています。

Dは誤りです。Oracle独自の結合構文では「自然結合」という用語は使用されていません。Eは誤りです。Oracleの結合構文とANSIの結合構文には本質的なパフォーマンスの違いはなく、パフォーマンスはクエリの記述方法とデータベースオプティマイザによる処理方法に依存します。

Gが間違っている理由は、Eと同じである。

質問: 156

このコマンドの一部を調べてみましょう。

```
ORACLE
CREATE TABLE cust (
  cust_id NUMBER(2),
  credit_limit NUMBER(10)
)
ORGANIZATION EXTERNAL
```

このコマンドを正常に実行するために必要な2つの句はどれですか？

- A. アクセスドライバのTYPE句
- B. デフォルトディレクトリ条項
- C. REJECT LIMIT句
- D. アクセスパラメータ句
- E. LOCATION条項

正解: [\(正解を表示します\)](#)

質問: 157

DEPARTMENT_DETAILSとCOURSE_DETAILSを作成するために使用されたコマンドを調べてください。

```
ORACLE
SQL>CREATE TABLE DEPARTMENT_DETAILS
(DEPARTMENT_ID NUMBER PRIMARY KEY,
DEPARTMENT_NAME VARCHAR2(50),
HOD VARCHAR2(50));
SQL>CREATE TABLE COURSE_DETAILS
(COURSE_ID NUMBER PRIMARY KEY,
COURSE_NAME VARCHAR2(50),
DEPARTMENT_ID NUMBER REFERENCES DEPARTMENT_DETAILS
(DEPARTMENT_ID));
```

対応する学科IDの有無に関わらず、すべてのコースIDを表示するレポートを作成したいが、コースが存在しない学科には学科IDを表示したくない。どのSQL文を使用すべきですか？

- A. SELECT c.course_id, d.department_id FROM course_details c FULL OUTER JOIN department_details d ON (c.department_id=d.department_id)
- B. SELECT c.course_id, d.department_id FROM course_details c RIGHT OUTER JOIN .department_details d ON (c.department_id=d.department_id)
- C. SELECT c.course_id, d.department_id FROM course_details c FULL OUTER JOIN department_details d ON (c.department_id<>d.department_id)
- D. SELECT course_id, department_id, FROM department_details d RIGHT OUTER JOIN course_details c USING (department_id)

正解: [\(正解を表示します\)](#)

質問: 158

データディクショナリに変更を加える2つのステートメントはどれですか？

- A. GRANT UPDATE ON Bcott.emp TO fin_manager;
- B. acott.emp から削除;
- C. ALTER SESSION SET NLS DATE_FORMAT = * DD/MM/ YVYY * ;
- D. TRUNCATE TABLE emp;
- E. SELECT * FROM user_tab_privs;

正解: ([正解を表示します](#))

質問: 159

フルアウター結合について、正しい記述は次のうちどれですか？ 2つ選択)

- A. 結合される両方のテーブルから、一致しない行のみを返します。
- B. 直積によって返される行が含まれます。
- C. 結合される両方のテーブルから、一致した行と一致しなかった行を返します。
- D. 内部結合によって返される行が含まれます。
- E. Oracleの結合演算子 (+)は、WHERE句の結合条件の両側で使用する必要があります。

正解: ([正解を表示します](#))

質問: 160

ORDER_ITEMsテーブルを調べてください。

Name	Null?	Type
ORDER_ID	NOT NULL	NUMBER (38)
PRODUCT_ID	NOT NULL	NUMBER (38)
QUANTITY	NOT NULL	NUMBER (38)

QUANTITYが10の倍数である行を返すクエリはどれですか？ 2つ選択してください)

- A. SELECT * FROM order_items WHERE quantity = TRUNC (quantity, -1);
- B. SELECT * FROM order_items WHERE MOD (quantity, 10) = 0;
- C. SELECT" FROM order_items WHERE FLOOR (quantity / 10) = TRUNC (quantity / 10);
- D. SELECT FROM order_items WHERE quantity / 10 = TRUNC (quantity);
- E. SELECT" FROM order__items WHERE quantity = ROUND (quantity, 1);

正解: ([正解を表示します](#))

* A: このステートメントは機能しません。なぜなら、TRUNC(quantity, -1) は、数値が負の 10 のべき乗である場合に、数量を小数点以下 0 桁に切り捨てますが、数量が 10 の倍数であることを保証するものではないからです。

* B: この記述は正しいです。MOD 関数は除算の余りを返します。MOD(quantity, 10) がゼロに等しいということは、数量が10の倍数であることを意味します。

* C: この記述は正しいです。数量が10の倍数である場合、FLOOR(quantity / 10) は TRUNC(quantity / 10) と等しくなります。これは、FLOOR が数値以下の最大の整数を返し、TRUNC が小数部分を切り捨てるためです。

* D: この記述は誤りです。TRUNC(quantity) は quantity から小数点以下の桁をすべて削除するため、quantity / 10 と比較しても quantity が 10 の倍数であるとは限りません。

* E: この記述は誤りです。ROUND(quantity, 1) は数量を小数点以下第 1 位に丸めるだけで、10 の倍数かどうかをチェックするわけではありません。

質問: 161

Oracleデータベースにおけるタイムゾーン、日付データ型、タイムスタンプデータ型について、正しい記述を3つ挙げてください。

- A. DBTIMEZONE関数は協定世界時 (UTC)からのオフセットを返すことができます
- B. タイムスタンプとローカルタイムゾーンのデータ型列は、行を挿入したセッションのタイムゾーンを使用してデータベースに格納されます。
- C. TIMESTAMPデータ型の列には、年、月、日に関する情報が含まれています。
- D. SESSIONTIMEZONE関数は協定世界時 (UTC)からのオフセットを返すことができます
- E. CURRENT_TIMESTAMP 関数は、タイムゾーン情報を含まないデータを返します。

正解: A,B,D (コメントを发表する)

A: その通りです。DBTIMEZONE関数は、データベースのタイムゾーンをUTC（協定世界時からオフセットした値を返します。Oracle 12cでは、DBTIMEZONE関数はデータベースのタイムゾーンを地域名またはUTCからの数値オフセットとして返すことができます。これは、タイムゾーンの管理に関するOracleのドキュメントに記載されています。

B: 正しいです。TIMESTAMP WITH LOCAL TIME ZONE は、データベースに格納されているデータを、データのクエリまたは挿入を行うセッションのタイムゾーンに調整するデータ型です。データが格納される際、Oracle はセッションのタイムゾーンから UTC に変換し、取得時にはセッションのタイムゾーンに戻します。

これは、同じデータを異なるタイムゾーンで自動的に表示できるように設計された機能であり、グローバルなアプリケーションにおいて非常に役立ちます。

D: 正解です。SESSIONTIMEZONE関数は、現在のセッションのUTCからのタイムゾーンオフセットを返します。これは、異なるタイムゾーンで使用されるアプリケーションにおけるデータ変換の理解と管理に役立ちます。タイムゾーンは、環境設定に応じて、UTCからのオフセットとして、または名前付き地域として表示されます。

質問: 162

展示物をご覧になり、部門と場所の説明（既舎をご確認ください）。

ORACLE

DEPT		
Name	Null?	Type
DEPARTMENT_ID		NUMBER(4)
DEPARTMENT_NAME	NOT NULL	VARCHAR2(30)
MANAGER_ID		NUMBER(6)
LOCATION_ID		NUMBER(4)
CITY		VARCHAR2(30)

LOCATIONS		
Name	Null?	Type
LOCATION_ID	NOT NULL	NUMBER(4)
STREET_ADDRESS		VARCHAR2(40)
POSTAL_CODE		VARCHAR2(12)
CITY	NOT NULL	VARCHAR2(30)
STATE_PROVINCE		VARCHAR2(25)
COUNTRY_ID		CHAR(2)

各部署について、DEPTテーブルのCITY列を、LOCATIONstableのCITY列に対応する値で更新したい。

このタスクを実行するには、どのSQL文を実行すればよいでしょうか？

- A. 部門dを更新

SET city = (SELECT city

場所 l)

WHERE d.location_id = l.location_id;

- B. 部門Dの更新

SET city = ALL (SELECT city
場所から
WHERE d.location_id = l.location_id);

C. 部門dの更新

SET city = ANY (SELECT city
場所 l)

D. 部門dの更新

SET city = (SELECT city
場所から
WHERE d.location_id = l.location_id);

正解: [D \(コメントを公表する\)](#)

質問: 163

MEMBERSテーブルの構造を調べてください。(最適な答えを選択してください。)

SQL文を調べてください。

SQL > SELECT city, last_name LNAME FROM MEMBERS ORDER BY 1, LNAME DESC; を実行すると、どのような結果になりますか?

- A. 都市名を昇順で表示し、さらにその中の姓を降順で並べ替えます。
- B. ORDER BY句では列番号と列エイリアスを同時に使用できないため、失敗します。
- C. ORDER BY句では列エイリアスを使用できないため、失敗します。
- D. すべての都市を降順で表示し、さらにその都市内では姓を降順に並べ替えます。

正解: [\(正解を表示します\)](#)

質問: 164

複数テーブルへのINSERT文について、正しい記述を3つ選びなさい。(3つ選択してください。)

- A. ビューに対して実行できます。
- B. リレーショナルテーブルに対して実行できます。
- C. それらはサブクエリを使用することによってのみ実行できます。
- D. SQL* Loader を使用して外部テーブルに対して実行できます。
- E. 遠隔テーブルでも実行できます。
- F. 計算された各行を複数のテーブルに挿入できます。

正解: [\(正解を表示します\)](#)

質問: 165

展示物をご覧になり、従業員用および部門用厩舎の構造を調べてください。

ORACLE

name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(10,2)
COMMISSION		NUMBER(6,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

DEPARTMENTS

name	Null?	Type
DEPARTMENT_ID	NOT NULL	NUMBER(4)
DEPARTMENT_NAME	NOT NULL	VARCHAR2(30)
MANAGER_ID		NUMBER(6)
LOCATION_ID		NUMBER(4)

EMPLOYEEstableを以下のように更新します。

ボストンまたはシアトル (拠点番号900および2700)で勤務する従業員のみを更新してください。

これらの従業員の department_id をロンドンに対応する department_id に設定してください。

(location_id 2100)

location_id2100の従業員の給与を、所属部署の平均給与の1.1倍に設定する。

location_id2100 の従業員のコミッションを、その従業員の平均コミッションの 1.5 倍に設定します。

部門。

次のコマンドを実行します。

```
UPDATE employees
SET department_id =
  (SELECT department_id
   FROM departments
   WHERE location_id = 2100),
(salary, commission) =
  (SELECT 1.1*AVG(salary), 1.5*AVG(commission)
   FROM employees, departments
   WHERE departments.location_id IN(2900, 2700, 2100))
WHERE department_id IN
  (SELECT department_id
   FROM departments
   WHERE location_id = 2900
   OR location_id = 2700;
```

結果とは何ですか？

- A. 正常に実行され、目的の更新が行われます。
- B. UPDATE文では複数の列 (SALARY、COMMISSION) を同時に指定できないため、エラーが発生します。
- C. UPDATE文ではサブクエリに結合条件を含めることができないため、エラーが発生します。
- D. 正常に実行されますが、期待どおりの更新は行われません。

正解: ([正解を表示します](#))

質問: 166

SQLのWHERE句とHAVING句に関して、正しい記述を3つ選びなさい。(3つ選択してください。)

- A. HAVING句の条件では、列にエイリアスを使用できます。
- B. HAVING句は、データをグループ化した後に、1つ以上の集計結果を除外するために使用されます。
- C. WHERE句は、データをグループ化する前に行を除外するために使用されます。
- D. SQL文では、WHERE句とHAVING句を同時に使用することはできません。
- E. HAVING句の条件には集約機能を持たせることができます。

正解: ([正解を表示します](#))

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%ディスカウント、特別な割引コード: **JPNshiken**」

質問: 167

従業員テーブルの構造を調べます。

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
EMAIL	NOT NULL	VARCHAR2(25)
PHONE_NUMBER		VARCHAR2(20)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

EMPLOYEE_IDとMANAGER_IDの間には親子関係があります。

EMPLOYEE_IDが123である従業員と同じマネージャーの下で働く従業員の姓とマネージャーIDを表示したい。

どのクエリが正しい出力結果をもたらしますか？

SELECT e.last_name, m.manager_id

A. FROM employees e RIGHT OUTER JOIN employees m

(e.employee_id = m.employee_id)

e.employee_id = 123 の場合;

SELECT m.last_name, e.manager_id

B. FROM employees e LEFT OUTER JOIN employees m

(e.employee_id = m.manager_id)

e.employee_id = 123 の場合;

SELECT e.last_name, e.manager_id

C. FROM employees e LEFT OUTER JOIN employees m

(e.manager_id = m.manager_id)

e.employee_id = 123 の場合;

D. FROM employees e RIGHT OUTER JOIN employees m

(e.manager_id = m.employee_id)

かつ、e.employee_id = 123;

SELECT e.last_name, m.manager_id

正解: ([正解を表示します](#))

質問: 168

2つの表の構造を調べてください。

PRODUCTS:

Name	Null?	Type
PROD_ID		CHAR(2)
PROD_NAME		CHAR(4)
EXP_DATE		TIMESTAMP(6)

NEW_PRODUCTS:

Name	Null?	Type
PROD_ID		CHAR(4)
PROD_NAME		VARCHAR2(10)
EXP_DATE		DATE

正常に実行されるクエリはどれですか？ 2つ選択してください。)

- A. `SELECT prod_id FROM products
UNION ALL
SELECT prod_id, prod_name FROM new_products;`
- B. `SELECT * FROM products
MINUS
SELECT prod_id FROM new_products;`
- C. `SELECT * FROM products
UNION
SELECT * FROM new_products;`
- D. `SELECT prod_id, prod_name FROM products
INTERSECT
SELECT 100, prod_name FROM new_products;`

【して】:

```
SELECT prod_id, exp_date FROM products  
UNION ALL  
SELECT prod_id, NULL FROM new_products;
```

正解: ([正解を表示します](#))

質問: 169

展示資料をご覧になり、製品表の構造を調べてください。

ORACLE

Table PRODUCTS		
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER(6)
PROD_NAME	NOT NULL	VARCHAR2(50)
PROD_DESC	NOT NULL	VARCHAR2(4000)
PROD_CATEGORY	NOT NULL	VARCHAR2(50)
PROD_CATEGORY_ID	NOT NULL	NUMBER
PROD_UNIT_OF_MEASURE		VARCHAR2(20)
SUPPLIER_ID	NOT NULL	NUMBER(6)
PROD_STATUS	NOT NULL	VARCHAR2(20)
PROD_LIST_PRICE	NOT NULL	NUMBER(8,2)
PROD_MIN_PRICE	NOT NULL	NUMBER(8,2)

サブクエリが必要となるタスクはどれですか？ (2つ選択してください。)

- A. 各製品ステータスの最小PROD_LIST_PRICEを表示する
- B. PROD_MIN_PRICEが全製品の平均PROD_LIST_PRICEよりも高く、かつステータスが注文可能の製品をすべて表示します。
- C. PROD_LIST_PRICEが平均PROD_LIST_PRICEよりも高い製品の数を表示します。
- D. PROD_LIST_PRICEが1000未満のサプライヤーを表示する
- E. サプライヤー102が供給し、製品ステータスが「廃止」である製品の総数を表示します。

正解: B,C ([コメントを公表する](#))

質問: 170

PROMOTIONSテーブルのPROMO_BEGIN_DATE列はデータ型であり、デフォルトの日付形式はDD-MON-RRです。

クエリでPROMO_BEGIN_DATEを使用する式について、正しい記述は次のうちどれですか？ (2つ選択)

- A. PROMO_BEGIN_DATE - SYSDATE はエラーを返します
- B. TONUMBER (PROMO BEGIN_DATE) - 5 は数値を返します
- C. PROMO_BEGIN_DATE - 5 は日付を返します
- D. PROMO_BEGIN_DATE - SYSDATE は数値を返します
- E. TODATE(PROMO BEGIN_DATE *5) は日付を返します

正解: C,D ([コメントを公表する](#))

質問: 171

このクエリを調べてください。

```
SELECT employee_id, first_name, salary
       FROM employees
      WHERE hire_date > '&1';
```

このクエリの実行時に、入社日の値の入力を求められないようにするには、どの2つの方法を使用すればよいですか？

(2つ選択してください。)

- A. クエリを実行する前に UNDEFINE コマンドを使用してください。
- B. クエリ内の &1」を &&1」に置き換えます。
- C. クエリを実行する前に、SET VERIFY OFF コマンドを実行してください。

- D. クエリをスクリプトに保存し、実行時に置換値をスクリプトに渡します。
- E. クエリを実行する前に SET VERIFY ON コマンドを実行します。
- F. クエリを実行する前に、DEFINEコマンドを使用してください。

正解: ([正解を表示します](#))

質問: 172

セッション内で実行されるたびに、2つの列名とWHERE条件の入力を求めるクエリを作成したいが、テーブル名の入力は初回実行時のみにしたい。クエリで使用されている変数は、セッション内で未定義になることはありませんか？
どのクエリを使用できますか？

```
SELECT '&col1', '&col2'  
FROM &table
```

A. WHERE '&condition' = '&cond';

```
SELECT &col1, &col2  
FROM &table
```

B. WHERE &condition;

【として】:

```
SELECT &col1, &col2  
FROM "&table"  
WHERE &condition;
```

C. WHERE &&condition;

```
SELECT &&col1, &&col2  
FROM &table
```

D. WHERE &&condition = &&cond;

```
SELECT &&col1, &&col2  
FROM &table
```

正解: C ([コメントを發表する](#))

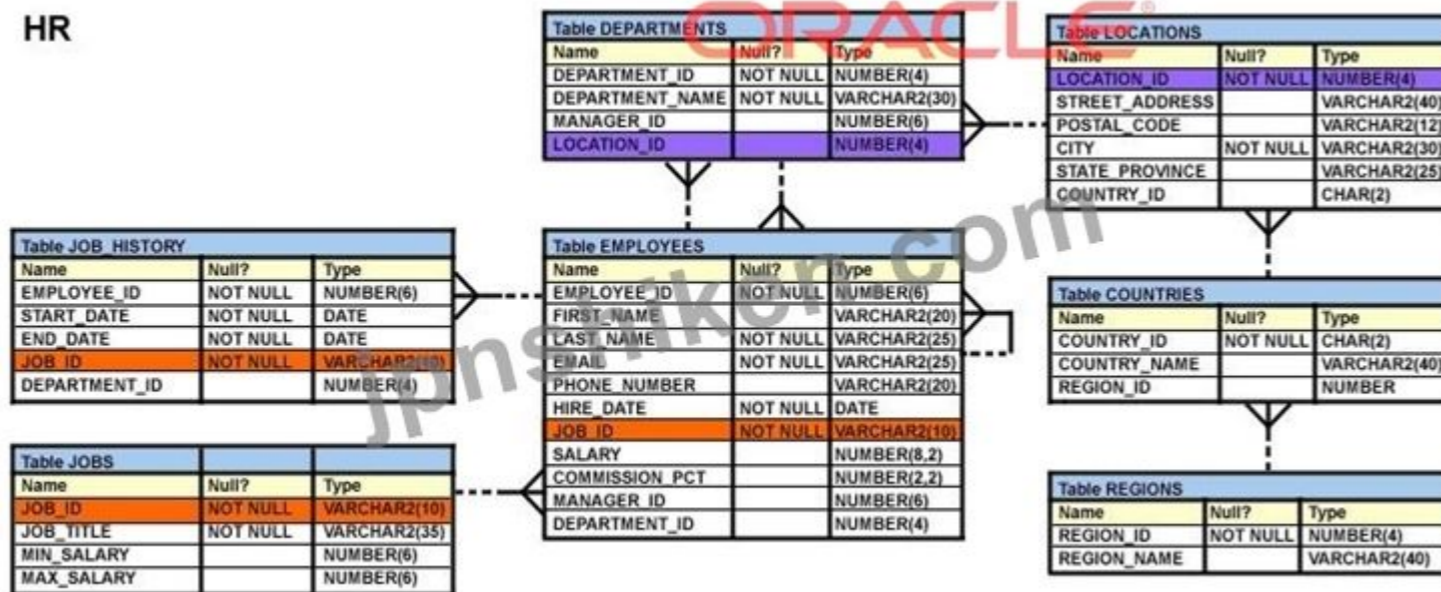
質問: 173

展示資料をご覧になり、従業員表の構造を調べてください。

MANAGER_IDが100であるすべての従業員とそのマネージャーを表示したい。

出力は2列で表示したい。1列目にはマネージャーの姓、2列目には従業員の姓が表示される。

HR



どのSQL文を実行しますか？

- A. SELECT m.last_name "Manager", e.last_name "Employee"
FROM employees m JOIN employees e
m.employee_id = e.manager_id かつ e.manager_id = 100 の場合。
 - B. SELECT m.last_name "Manager", e.last_name "Employee"
FROM employees m JOIN employees e
ON m.employee_id = e.manager_id
 - C. SELECT m.last_name "Manager", e.last_name "Employee"
FROM employees m JOIN employees e
ON e.employee_id = m.manager_id
m.manager_id=100 の場合;
 - D. manager_id=100の場合;
 - E. SELECT m.last_name "Manager", e.last_name "Employee"
FROM employees m JOIN employees e
ON m.employee_id = e.manager_id
m.manager_id=100 の場合;
- 正解: (正解を表示します)

質問: 174

PROMOTIONSテーブルの構造を調べてください。(最適な回答を選択してください。)

NAME	NULL?	TYPE
PROMO_ID	NOT NULL	NUMBER(6)
PROMO_NAME	NOT NULL	VARCHAR2(30)
PROMO_CATEGORY	NOT NULL	VARCHAR2(30)
PROMO_COST	NOT NULL	NUMBER(10,2)

経営陣は、各プロモーションカテゴリーにおける個別のプロモーション費用に関する報告書を必要としている。
この要件を満たすクエリはどれでしょうか？

- A. SELECT DISTINCT promo_category, promo_cost FROM promotions ORDER BY 1

- B. SELECT promo_category, DISTINCT promo_cost FROM promotions
 - C. SELECT DISTINCT promo_cost, promo_category FROM promotions
 - D. SELECT DISTINCT promo_cost, DISTINCT promo_category FROM promotions;
- 正解: ([正解を表示します](#))

質問: 175

BOOKS_TRANSACTIONStableの構造を調べます。

Name	Null?	Type
TRANSACTION_ID	NOT NULL	VARCHAR2 (6)
BORROWED_DATE		DATE
DUE_DATE		DATE
BOOK_ID		VARCHAR2 (8)
MEMBER_ID		VARCHAR2 (6)

MEMBER_IDが入力されていないすべての行について、BOOK_IDを「INVALID」に設定するように、このテーブルを更新したい。
この部分的なSQL文を調べてみましょう。

```
SQL> UPDATE books_transactions
SET    book_id = 'INVALID'
WHERE .....
```

必要な更新を実行するには、WHERE句でどの条件を使用する必要がありますか？

- A. MEMBER_ID = NULL;
- B. MEMBER_ID = "";
- C. MEMBER_ID = ";
- D. MEMBER_ID が NULL です。

正解: ([正解を表示します](#))

質問: 176

BRICKSテーブルの説明を確認してください。

Name	Null?	Type
BRICK_ID		NUMBER(38)
SHAPE		VARCHAR2(30)
COLOR		VARCHAR2(30)
WEIGHT		NUMBER

BRICKS_STAGEテーブルの説明を確認してください。

Name	Null?	Type
WEIGHT		NUMBER
SHAPE		VARCHAR2(30)
COLOR		VARCHAR2(30)

どの2つのクエリが正常に実行されましたか？

A. ブロックから形状と色を選択します

マイナス

SELECT WEIGHT,color FROM bricks_stage;

B. レンガから*を選択

マイナス

SELECT * FROM bricks_stage;

C. SELECT brick_id,shape FROM bricks

マイナス

bricks_stage から WEIGHT、COLOR を選択します。

D. レンガの形状と色を選択します

マイナス

SELECT color, shape FROM bricks_stage;

E. レンガの形状、色、重量を選択してください

マイナス

SELECT * FROM bricks_stage;

正解: ([正解を表示します](#))

質問: 177

取引が完了する状況はどれですか？ 3つ選択してください。)

A. PL/SQL匿名ブロックが実行される時

B. DELETE文が実行された時

C. ROLLBACKコマンドが実行された時

D. データ定義言語 (DDL) ステートメントが実行された時

E. 保留中のトランザクションの後に TRUNCATE ステートメントが実行された場合

正解: ([正解を表示します](#))

参考文献 :

https://docs.oracle.com/cd/B19306_01/server.102/b14220/transact.htm

質問: 178

展示資料をご覧になり、ORDERS テーブルの構造を調べてください。

どのUPDATE文が正しいですか？

A. UPDATE orders SET order_date = '12-mar-2007', order_total = NULL WHERE order_id = 2455;

B. UPDATE orders SET order_date = TO_DATE('12-mar-2007','dd-mon-yyyy'), SET order_total = TO_NUMBER (NULL) WHERE order_id = 2455;

C. UPDATE orders SET order_date = '12-mar-2007', order_total IS NULL WHERE order_id = 2455;

D. UPDATE orders SET order_date = '12-mar-2007', AND order_total TO_NUMBER(NULL) WHERE order_id = 2455;

正解: [\(正解を表示します\)](#)

質問: 179

置換変数について正しい記述は次のうちどれですか？ (2つ選択)

- A. & で始まる置換変数は、セッション内で常に一度だけ値の入力を求めます。
- B. 列名の入力を促すために使用される置換変数は、二重引用符で囲む必要があります。
- C. 列名の入力を促すために使用される置換変数は、単一引用符で囲む必要があります。
- D. & で始まる置換変数は、セッション内で undefined に設定されていない限り、セッション内で値の入力を 1 回だけ促します。
- E. 置換変数は SELECT 文でのみ使用できます。
- F. 置換変数は、SELECT ステートメントのどの句でも使用できます。

正解: D,F [\(コメントを発表する\)](#)

質問: 180

EMPLOYEES テーブルのデータを調べてください。

EMPLOYEE ID	LAST NAME	MONTHLY SATARY	MONTHLY COMMISSION PCT
101	Kochhar	24000	<NULL>
102	Ernst	17000	.5
103	Rajs	21000	.2
104	Lorentz	25000	<NULL>
105	Morris	12000	<NULL >

各従業員の年間総報酬額を計算する式はどれですか？

- A. SELECT last name, (monthly_salary * 12) + (monthly_salary * 12 * NVL (monthly commission pct, 0)) AS annual_comp FROM employees;
- B. SELECT last_name, monthly_salary * 12 + (monthly_salary * 12 * Monthly commission _Pct) AS annual_comp FROM employees;
- C. SELECT last_name (monthly_salary + monthly_commission _ pct) * 12 AS annual_ FROM employees ;

D. 姓を選択してください、
(月給12) + (月間歩合率2) AS

年間報酬
従業員より

;
正解: [\(正解を表示します\)](#)

質問: 181

CUSTOMERS テーブルの説明を確認してください。

Name	Null?	Type
CUSTOMER_ID	NOT NULL	NUMBER (38)
CUSTOMER_NAME	NOT NULL	VARCHAR2 (100)
INSERT_DATE	NOT NULL	DATE

暗黙的な型変換を行うのは、どの2つの文ですか？

- A. SELECT FROM + customers WHERE (customer_id) = '0001';
- B. SELECT FROM + customers WHERE (customer_id) = 0001;
- C. SELECT FROM + customers WHERE TO-CHAR (customer_id) = '0001';
- D. SELECT FROM + customers WHERE insert_date = DATE '2019-01-01';
- E. SELECT FROM + customers WHERE insert_date = '01-JAN-19';

正解: B,E ([コメントを发表する](#))

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%
ディスカウント、特別な割引コード: **JPNshiken**」

質問: 182

MERGE文について正しいのはどれですか？ (3)

- A. テーブルからのみ行を結合できます。
- B. ビューを使用してソース行を生成できます。
- C. 複数のテーブルの行を条件付きで結合して、単一のテーブルに挿入できます。
- D. サブクエリを使用してソース行を生成できます。
- E. 対象テーブルの同じ行を複数回更新できます。
- F. 複数のテーブルで条件付きで行を更新、挿入、または削除できます。

正解: ([正解を表示します](#))

B :その通りです。Oracle DatabaseのMERGE文では、ビューを含むサブクエリを使用して、マージ操作のソースデータを生成できます。これにより、マージするデータの指定において、より高い柔軟性が得られます。

C: 正しい。MERGE は、一致するレコードが存在するかどうかに基づいて、1つ以上のソース テーブル (またはビュー、あるいはサブクエリの結果) から行を条件付きで結合し、単一のターゲット テーブルに行を挿入または更新できます。

D: その通りです。MERGE は確かにサブクエリを使用してソース行を生成できます。マージ操作のソースデータは、テーブル、ビュー、またはサブクエリの結果のいずれかになります。

OracleのMERGE文は、条件付きINSERTまたはUPDATE操作を1つの文で実行できる強力なSQL操作であり、しばしば「upsert」操作と呼ばれます。ターゲットデータセットとソースデータセットのデータが一致するかどうかに基づいて挿入または更新を決定するような複雑なシナリオにも対応できます。

参考資料 :Oracle SQLのMERGEステートメントに関するドキュメントには、サブクエリやビューなどのさまざまなデータソースを使用する方法、およびターゲットテーブルに対して条件付き操作を実行する方法に関する情報が記載されています。

質問: 183

正しいのは次のうちどれですか？

- A. CONCATは2つ以上の文字列を結合します。
- B. FLOORは、指定された数値以下の最大の整数を返します。
- C. CONCATは2つの文字列を結合します。
- D. INSTRは、文字列内のオフセットを1文字のみ検索します。

E. INSTR は、位置 0 から始まる文字列内のオフセットを検索します。

F. FLOORは、指定された数値以下の最大の正の整数を返します。

正解: (正解を表示します)

Oracle SQLのCONCAT関数とFLOOR関数には、それぞれ特有の動作があります。

A. CONCAT関数は2つ以上の文字列を1つの文字列に結合するため、この文は真になります。

B. FLOOR関数は、指定された数値以下の最大の整数を返すため、この記述は真となります。

C. CONCAT は 2 つの文字列を結合できますが、この記述は 2 つ以上の文字列を結合できるため不完全です。

D. INSTR は、文字列内の部分文字列のオフセットを、単一の文字だけでなく見つけることができます。

E. INSTR は Oracle SQL で文字列の検索を位置 0 からではなく位置 1 から開始します。

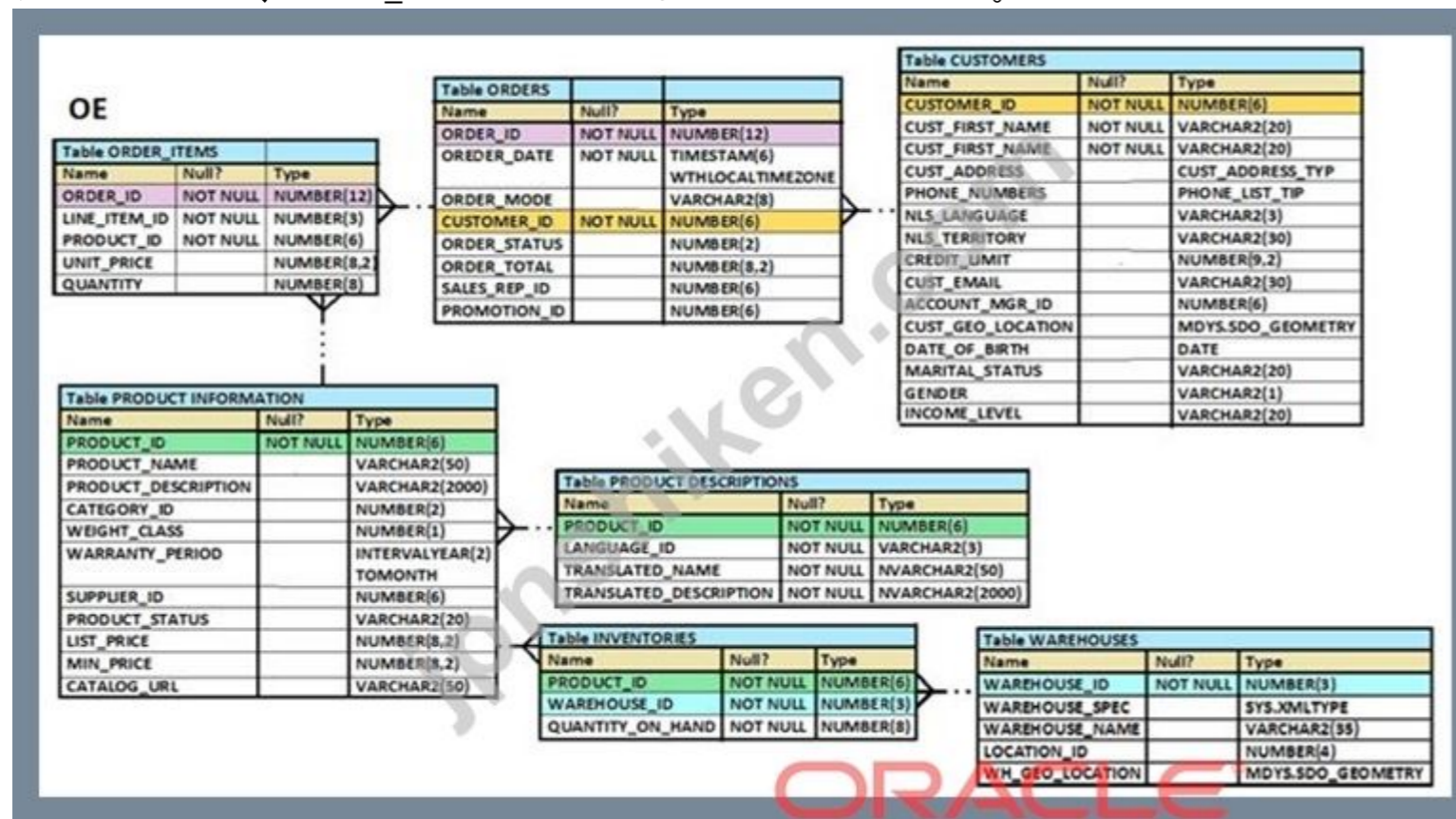
F.FLOORは指定された数値以下の最大の整数を返しますが、正の整数だけでなく、任意の整数を返すことができます。

参照 :

Oracle Database SQL言語リファレンス、12cリリース1 (2.1) 「単一行関数」

質問: 184

展示資料を参照し、ORDER_ITEMS テーブルの構造を確認してください。



次のSQL文を調べてください。

```
SELECT order_id, product_id, unit_price
```

注文アイテムから

ここで、単価

```
(SELECT MAX(unit_price)
```

注文アイテムから

order_idでグループ化する。

ORDER_IDごとに、UNIT_PRICEが最も高い製品のPRODUCT_IDを表示したい場合、上記のSQL文をどのように修正すればよいでしょうか？

- A. = を IN 演算子に置き換えます
- B. = を >ALL 演算子に置き換えます
- C. サブクエリからGROUP BY句を削除し、メインクエリに配置します。
- D. = を >ANY 演算子に置き換えます

正解: [A \(コメントを发表する\)](#)

質問: 185

あなたは、2つの列に以下の仕様を持つ表の構造を設計しています。

COMPONENT_ID - 最大12文字の英数字を含み、行を一意に識別できる必要があります。

EXECUTION_DATETIME - 世紀、年、月、日、時、分、秒を最高精度で含み、コンポーネント間の計算や比較に使用されます。

これらの要件を最も効率的に満たすデータ型を定義する2つの選択肢はどれですか？

- A. EXECUTION_DATETIME は INTERVAL DAY TO SECOND データ型である必要があります。
- B. COMPONENT_IDはROWIDデータ型である必要があります。
- C. EXECUTION_DATETIMEはDATEデータ型である必要があります。
- D. EXECUTION_DATETIMEはTIMESTAMPデータ型である必要があります。
- E. COMPONENT_IDはVARCHAR2データ型である必要があります。
- F. COMPONENT_ID列はCHARデータ型である必要があります。

正解: [C,F \(コメントを发表する\)](#)

質問: 186

演算子と条件の優先順位について、正しいのはどれか (2)

- A. || は +(加算) よりも優先順位が高い
- B. オペレーターは条件前に評価されます。
- C. ANDとORは条件式において同じ優先順位を持つ
- D. 条件式において、NOT は AND および OR よりも優先順位が高い。
- E. + (加算)は* (乗算)よりも優先順位が高い

正解: [\(正解を表示します\)](#)

質問: 187

このクエリを調べてください。

```
SELECT employee_id, first_name, salary
       FROM employees
      WHERE hire_date > '&1';
```

このクエリの実行時に、入社日の値の入力を求められないようにするには、どの2つの方法を使用すればよいですか？

- A. クエリ内の &1」を &&1」に置き換えます。
- B. クエリを実行する前に SET VERIFY ON コマンドを実行してください。
- C. クエリを実行する前に UNDEFINE コマンドを使用してください。
- D. クエリを実行する前に、DEFINEコマンドを使用してください。
- E. クエリをスクリプトに保存し、実行時に置換値をスクリプトに渡します。

F. クエリを実行する前に、SET VERIFY OFF コマンドを実行してください。

正解: (正解を表示します)

質問: 188

DEPARTMENT_DETAILSテーブルとCOURSE_DETAILSテーブルを作成するために使用されたコマンドを調べてください。

```
SQL>CREATE TABLE DEPARTMENT_DETAILS
(DEPARTMENT_ID NUMBER PRIMARY KEY,
DEPARTMENT_NAME VARCHAR2(50),
HOD VARCHAR2(50));
SQL>CREATE TABLE COURSE_DETAILS
(COURSE_ID NUMBER PRIMARY KEY,
COURSE_NAME VARCHAR2(50),
DEPARTMENT_ID NUMBER REFERENCES DEPARTMENT_DETAILS (DEPARTMENT_ID));
```

COURSE_DETAILStableに存在しないすべての部署IDのリストを生成したい。

SQL文を実行します。

```
SQL> SELECT d.department_id FROM course_details c INNER JOIN
department_details d ON c.department_id<>d.department_id;
```

結果はどうなりますか？

- A. 正常に実行され、必要なリストが表示されます。
- B. ON句の条件が無効であるため、失敗します。
- C. 使用されている結合タイプが間違っているため、失敗します。
- D. 正常に実行されますが、誤ったリストが表示されます。

正解: (正解を表示します)

質問: 189

展示資料を参照し、「従業員」表のデータを確認してください。

展示する

ENAME	HIREDATE	SAL	COMM
SMITH	17-DEC-00	800	
ALLEN	20-FEB-99	1600	300
WARD	22-FEB-95	1250	500
JONES	02-APR-98	2975	
MARTIN	28-SEP-99	1250	1400
BLAKE	01-MAY-97	2850	

各従業員にこれまでに支払われた報酬総額を示すレポートを作成したい。

あなたは以下のクエリを発行します。

```
SQL> SELECT ename || 'joined on' || hiredate ||  
' , the total compensation paid is'  
TO_CHAR (ROUND (ROUND (SYSDATE-hiredate) /365 * sal +comm)  
"COMPENSATION UNTIL DATE"  
FROM employees;
```

結果はどうなりますか？

- A. ITは正常に実行され、正しい出力が得られます。
- B. 連結演算子は2つの項目しか結合できないため、エラーが発生します。
- C. 正常に実行されますが、正しい出力が得られません。
- D. 式の中で ROUND 関数が使用されていないため、エラーが発生します。
- E. エイリアスが無効であるため、エラーが発生します。

正解: [C \(コメントを發表する\)](#)

質問: 190

CUSTOMERSテーブルの構造を調べてください。(2つ選択してください。)

CUSTNOは主キーです。

重複している顧客名をすべてリストアップすることで、異なる顧客番号を使用して顧客情報が複数回入力されていないかどうかを確認する必要があります。

必要な結果を得るために使用できる2つの方法はどれですか？

- A. 自己結合による完全外部結合
- B. 自己参加
- C. 自己結合による右外側結合
- D. 左外部結合 (自己結合)
- E. サブクエリ

正解: [\(正解を表示します\)](#)

質問: 191

このクエリを実行します。

SELECT TO CHAR (NEXT_DAY(LAST_DAY(SYSDATE),'MON'),' dd"Monday for" fmMonth rrr') FROM DUAL; 結果はどうなりますか？

- A. エラーが発生します。
- B. 正常に実行されますが、結果は返されません。
- C. 翌月の最初の月曜日の日付を返します。
- D. 当月の最終月曜日の日付を返します。

正解: [\(正解を表示します\)](#)

質問: 192

CUSTOMERSテーブルには、データ型がNUMBERのCUST_CREDIT_LIMIT列があります。

どの2つのクエリが正常に実行されましたか？

- A. SELECT NVL2(cust_credit_limit, TO_CHAR(cust_credit_limit * .15), 'Not
- B. SELECT NVL (cust_credit_limit * .15, '利用不可') FROM customers;
- C. SELECT NVL(TO_CHAR(cust_credit_limit * .15), '利用不可') FROM customers;

D. 顧客から入手可能

SELECT NVL2 (cust_credit_limit * .15, '利用不可') FROM customers;

E. SELECT TO_CHAR(NVL(cust_credit_limit * .15, '利用不可')) FROM customers;

正解: ([正解を表示します](#))

質問: 193

SALESテーブルの構造を調べてください。

NAME	NULL?	TYPE
PRODUCT_ID	NOT NULL	NUMBER(10)
CUSTOMER_ID	NOT NULL	VARCHAR2(10)
TIME_ID	NOT NULL	DATE
CHANNEL_ID	NOT NULL	NUMBER(5)
PROMO_ID	NOT NULL	NUMBER(5)
QUANTITY_SOLD	NOT NULL	NUMBER(10, 2)
PRICE		NUMBER(10, 2)
AMOUNT_SOLD	NOT NULL	NUMBER(10, 2)

この記述を検討してください。

```
SQL > CREATE TABLE sales1 (prod_id, cust_id, quantity_sold, price)
AS
SELECT product_id, customer_id, quantity_sold, price
FROM sales
WHERE 1 = 2;
```

SALES1テーブルについて正しい記述はどれですか？ (2つ選択してください。)

A. 行なしで作成されます。

B. SALES テーブルで PRIMARY KEY と UNIQUE の制約があった選択された列に PRIMARY KEY と UNIQUE の制約があります。

C. 無効なWHERE句のため、作成されません。

D. SELECT句とCREATE TABLE句で指定された列名が一致しないため、作成されません。

E. SALES テーブルでこれらの制約を持っていた選択された列に NOT NULL 制約が設定されます。

正解: ([正解を表示します](#))

質問: 194

BOOKS_TRANSACTIONSテーブルの構造を調べてください。

Name	Null?	Type
TRANSACTION_ID	NOT NULL	VARCHAR2 (6)
TRANSACTION_TYPE		VARCHAR2 (3)
BORROWED_DATE		DATE
DUE_DATE		DATE
BOOK_ID		VARCHAR2 (6)
MEMBER_ID		VARHCAR2 (6)

SQL文を調べてください。

```
SQL> SELECT * FROM books_transactions WHERE borrowed_date<SYSDATE
AND transaction_type= 'RM' OR MEMBER_ID IN ('A101', 'A102');
```

結果について正しい記述はどれですか？

- A. 本日より前にRM TRANSACTION_TYPEで借入を行った会員A101とA102の詳細のみを表示します。
- B. TRANSACTION_TYPEがRM、またはMEMBER_IDがA101とA102のいずれかである、本日より前に借入を行った会員の詳細を表示します。
- C. TRANSACTION_TYPE が RM で、本日より前に借入を行った会員の詳細のみを表示します。
- D. 本日より前にトランザクションタイプがRMで借入を行った会員の詳細と、会員A101またはA102の詳細を表示します。

正解: [\(正解を表示します\)](#)

質問: 195

EMPLOYEES テーブルの説明を確認してください。

Name	Null?	Type
EMPLOYEE_NAME	NOT NULL	VARCHAR2(5)
HIRE_DATE		DATE
SALARY		NUMBER(7,2)

セッションのタイムゾーンはデータベースサーバーと同じです

勤続年数が5年以上の従業員のみを記載する記述は、次のうちどれですか？

- A. SELECT employee_name FROM employees WHERE (CUARENT_DATE - hire_data / 365>5
- B. SELECT employee_name FROM employees WHERE (SYSTIMESTAMP - hire_data) / 365>
- C. SELECT employee_name FROM employees WHERE (SYSNAYW - hire_data / 12> 3
- D. SELECT employee_name FROM employees WHERE (CUNACV_DATE - hire_data / 12> 3
- E. SELECT employee_name FROM employees WHERE (SYSDATE - hire_data) / 365>5
- F. SELECT employee_name FROM employees WHERE (SYSNAYW - hire_data / 12> 3

正解: [\(正解を表示します\)](#)

質問: 196

正常に実行される3つのステートメントを調べてください。

ユーザー finance を pwfin で識別して作成します。

ユーザー fin manager を pwmgr で識別して作成します。

ユーザー fin. Clerk を pwclerk で識別して作成します。

GRANT CREATE SESSON 20 finance, fin clerk;

GRANT SELECT ON scott. Emp を GRANT OPTION 付きで資金調達します。

CONNECT finance/pwfin

GRANT SELECT ON scott.emp To fin__clerk;

正しいのは次のうちどれですか？

- A. ユーザーFINANCEを削除すると、ユーザーFIN_CLERKからSCOTT.EMPに対するSELECT権限が自動的に取り消されます。
- B. ユーザー FINANCE から SCOTT.EMP に対する SELECT を取り消すと、ユーザー FIN_CLERK から権限が取り消されます。
- C. ユーザーFINANCEは、ユーザーFIN MANAGERにCREATE SESSIONを付与できます。
- D. ユーザー FIN CLERK は、ユーザー FIN MANAGER に SCOTT、ENP に対する SELECT 権限を付与できます。
- E. ユーザー FINANCE は FIN MANAGER に SCOTT.ENP に対する ALL 権限を付与できません。

正解: ([正解を表示します](#))

Oracleにおける権限について：

- * オプション A: ユーザー FINANCE を削除すると、ユーザー FIN_CLERK から SCOTT.EMP に対する SELECT が自動的に取り消されます。
- * ユーザーを削除すると、そのユーザーが他のユーザーに付与していたすべての権限が取り消されます。
- * オプション B: ユーザー FINANCE から SCOTT.EMP に対する SELECT 権限を取り消すと、ユーザー FIN_CLERK から権限が取り消されます。
- * FINANCE が FIN_CLERK に SCOTT.EMP に対する SELECT 権限を grant オプション付きで付与しているため、FINANCE から権限を取り消すと、FIN_CLERK から権限が連鎖的に取り消されます。

選択肢C、D、Eは誤りです。理由は以下のとおりです。

- * オプション C: ユーザー FINANCE は自身が所有する権限を付与できますが、CREATE SESSION はシステム権限です。FINANCE が CREATE SESSION に対して WITH ADMIN OPTION」を持っていない限り、この権限を付与することはできません。
- * オプション D: ユーザー FIN_CLERK は、'WITH GRANT OPTION' で権限が付与されている場合にのみ、他のユーザーに権限を付与できますが、この場合は付与されていません。
- * オプション E: ユーザー FINANCE は、SCOTT.EMP に対する権限を他のユーザーに付与できるのは、当該権限が WITH GRANT OPTION」付きで付与されている場合に限りです。権限が GRANT OPTION 付きの SELECT のみの場合、すべての権限を付与することはできません。

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%
ディスカウント、特別な割引コード: **JPNshiken**」

質問: 197

CREATE TABLEコマンドについて、正しい記述を3つ選びなさい。

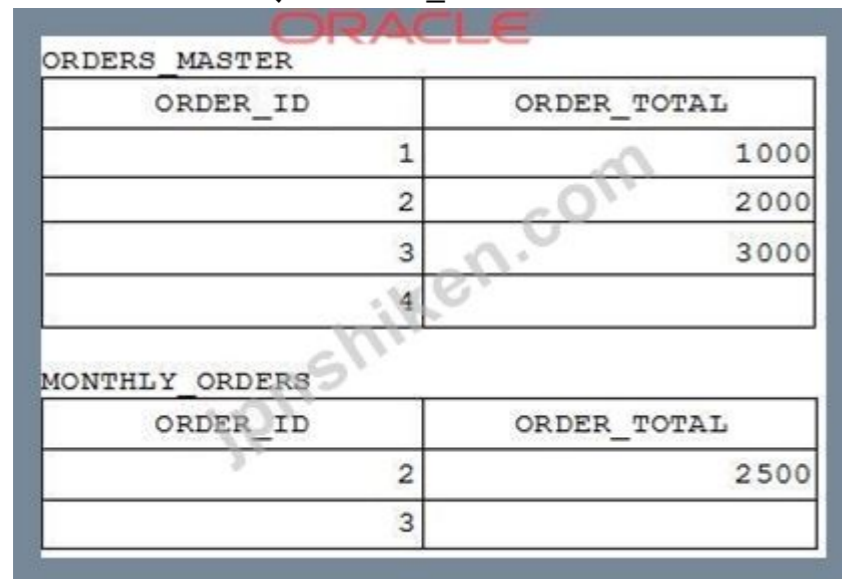
- A. 主キー制約を強制するためのインデックスを作成するCREATE..INDEXステートメントを含めることができます。
- B. 暗黙的にコミットを実行します
- C. テーブルを作成するには、ユーザーはCREATE ANY TABLE権限を持っている必要があります。
- D. 保留中のトランザクションを暗黙的にロールバックします
- E. テーブルの所有者は、テーブルが定義されているテーブルスペースに空き容量を確保している必要があります。
- F. テーブルの所有者は、UNLIMITED TABLESPACEシステム権限を持っている必要があります。

正解: ([正解を表示します](#))

説明／参考資料 https://docs.oracle.com/html/E25494_01/tables003.htm

質問: 198

展示資料を閲覧し、ORDERS_MASTERテーブルとMONTHLY_ORDERSテーブルのデータを確認してください。



ORDERS_MASTER	
ORDER_ID	ORDER_TOTAL
1	1000
2	2000
3	3000
4	

MONTHLY_ORDERS	
ORDER_ID	ORDER_TOTAL
2	2500
3	

次のMERGEステートメントを評価してください。

```
MERGE INTO orders_master o
```

```
USING monthly_orders m
```

```
ON (o.order_id = m.order_id)
```

一致した場合

```
UPDATE SET o.order_total = m.order_total
```

(m.order_total が NULL の場合) を削除します。

一致しない場合は

値を挿入します (m.order_id、m.order_total)。

上記の発言の結果はどうなるでしょうか？

- A. ORDERS_MASTER テーブルには ORDER_ID 1、2、3、4 が含まれます。
- B. ORDERS_MASTER テーブルには ORDER_ID 1、2、4 が含まれます。
- C. ORDERS_MASTER テーブルには ORDER_ID 1、2、3 が含まれます。
- D. ORDERS_MASTER テーブルには ORDER_ID 1 と 2 が含まれます。

正解: [\(正解を表示します\)](#)

説明

https://docs.oracle.com/cd/B28359_01/server.111/b28286/statements_9016.htm

質問: 199

データ型に関して正しい記述を3つ選びなさい。

- A. TIMESTAMPデータ型の列には、秒の小数部分を含む時間値のみが格納されます。
- B. テーブルごとに使用できるLONG列は1つだけです。
- C. BLOBデータ型の列は、オペレーティングシステムファイルにバイナリデータを格納するために使用されます。
- D. CHARdata型の列の値は、定義された最大列幅まで空白で埋められます。
- E. VARCHAR2データ型の列に指定できる最小列幅は1です。

正解: [\(正解を表示します\)](#)

質問: 200

非等結合ステートメントのパフォーマンスについて正しいのはどれですか？

- A. between 条件は、>= および <= 条件を使用するよりも常にパフォーマンスが劣ります。
- B. Oracleの結合構文は、SQL:1999準拠のANSI結合構文よりも優れたパフォーマンスを発揮します。
- C. 使用する結合構文はパフォーマンスに影響しません。
- D. between 条件は、>= および <= 条件を使用するよりも常に優れたパフォーマンスを発揮します。
- E. テーブルエイリアスはパフォーマンスを向上させることができます。

正解: C ([コメントを發表する](#))

Oracle Databaseにおける非等結合SQL文に関連するパフォーマンスへの影響は、しばしば最適化の対象となります。

C. 使用する結合構文はパフォーマンスに影響しません。Oracle Databaseでは、結合を含むクエリのパフォーマンスは、構文 (ANSI構文かOracleの従来構文か) よりも、基となるデータの分布、インデックス、オプティマイザの統計情報、システム構成などの要因に大きく依存します。Oracleのオプティマイザは、さまざまな構文の結合表現を解釈し、それに応じて最適化できるほど高度です。

参照 :

Oracle Database Performance Tuning Guide 12cでは、さまざまな結合構文の影響と、Oracleのオプティマイザがそれらをどのように処理するかについて説明しています。

質問: 201

EMPLOYEESテーブルには1000行のデータがあり、従業員は10年以上会社に勤務しています。

次のSQL文を評価してください。

```
SQL> UPDATE employees
      SET salary = NVL(salary,0) + NVL(comm,0), comm = NVL(comm,0)
      WHERE hire_date < SYSDATE - 600;
```

結果はどうなるだろうか？

- A. 正常に実行され、会社で600日以上勤務している従業員の記録が更新されます。
- B. 式の中で複数のNVL関数が使用されているため、エラーが発生します。
- C. NVL 関数は UPDATE と一緒に使用できないため、エラーが発生します。
- D. 正常に実行されますが、行は更新されません。

正解: ([正解を表示します](#))

質問: 202

見解について正しい記述は次のうちどれですか？ (2つ選択)

- A. CHECK句は、ビューを介して基となるテーブルに特定の行が更新または挿入されるのを防ぎます。
- B. ビューはインデックス化できます。
- C. WITH CHECK句は、ビューをクエリする際に特定の行が表示されないようにします。
- D. ビューは、ビューに対する権限を再付与する必要なく更新できます。
- E. ビューを定義するクエリ内のテーブルは、ビューを作成するために必ず存在していなければなりません。

正解: ([正解を表示します](#))

質問: 203

Oracle Databaseサーバーにおけるトランザクションについて、正しい記述は次のうちどれですか？ (2つ選択)

- A. ユーザーは、別のセッションで同じユーザーが行った未コミットの更新をいつでも確認できます。
- B. データ定義言語 (DDL) ステートメントは、DDLによって引き起こされるデータディクショナリの更新に対してのみ自動的にコミットを実行します。
- C. コミットされていないトランザクションは、ユーザーが SQL*Plus に存在する場合、自動的にコミットされます。
- D. セッションは、自身によって行われた未コミットの更新を常に確認できます。
- E. セッションにコミットされていないトランザクションがある場合、DDL ステートメントは新しいトランザクションを開始する前に COMMIT を発行します。
- F. データ操作言語 (DML) ステートメントは常に新しいトランザクションを開始します。

正解: [\(正解を表示します\)](#)

質問: 204

Oracleデータベースのビューについて、正しい記述を3つ選びなさい。

- A. ビューは、同じスキーマに属している場合にのみテーブルを結合できます。
- B. ビューにはセグメントがありません。
- C. 定義クエリで存在しないテーブルを参照するビューを作成できます。
- D. ビューを使用してテーブルに挿入された行は、ビューが削除されてもテーブルに保持されます。
- E. ビューにはオブジェクト番号がありません。
- F. 定義クエリにWHERE句を含むビューをクエリする場合、SELECT文にWHERE句を含めることはできません。

正解: [\(正解を表示します\)](#)

質問: 205

INSTEAD OFトリガーが定義されていないビューに対してデータ操作言語 (DML) 操作を実行する場合、正しい記述を3つ選択してください。

- A. テーブルに対する削除ステートメントは、ビューを介して常に実行できます。
- B. INSERT文はビューを通してテーブルに対して常に実行できます。
- C. テーブルに PRIMARY KEY があり、ビューの定義クエリで PRIMARY KEY 列が参照されていない場合、ビューを使用して基となるテーブルの行をクエリすることはできません。
- D. ビューの定義クエリで参照されていないデフォルト値のない NOT NULL 制約を持つ列がテーブルにある場合、ビューを使用して基となるテーブルに行を追加することはできません。
- E. ビューの定義クエリに DISTINCT キーワードが含まれている場合、ビューを使用して基となるテーブルの行を追加または変更することはできません。
- F. WITH CHECK 句は、ビューを介して基となるテーブルから行を削除する場合には効果がありません。

正解: [A,B,C \(コメントを發表する\)](#)

質問: 206

123という名前のテーブルを作成します。

どのステートメントが正常に実行されますか？

- A. SELECT * FROM V'123V';
- B. SELECT * FROM '123';
- C. SELECT * FROM "123";
- D. SELECT * FROM TABLE (123);

正解: [\(正解を表示します\)](#)

質問: 207

CUSTOMERSテーブルには、データ型がNUMBERのCUST_CREDIT_LIMIT列があります。
正常に実行されるクエリはどれですか？ 2つ選択してください。)

A. お客様から入手可能です。

SELECT NVL2(cust_credit_limit * .15, '利用不可') FROM customers;

B. SELECT TO_CHAR(NVL(cust_credit_limit * .15, '利用不可')) FROM customers;

C. SLECT NVL(TO_CHAR(cust_credit_limit * .15), '利用不可') FROM customers;

D. SELECT NVL(cust_credit_limit * .15, '利用不可') FROM customers;

E. SELECT NVL2(cust_credit_limit, TO_CHAR(cust_credit_limit * .15), 'Not

正解: ([正解を表示します](#))

質問: 208

制約条件に関して正しい記述はどれですか？ 2つ選択してください。)

A. 主キーと外部キーの制約は、列レベルとテーブルレベルの両方で定義できます。

B. テーブルには主キーは1つしか設定できませんが、外部キーは複数設定できます。

C. 外部キーと親テーブルの主キーは同じ名前であればなりません。

D. テーブルには主キーと外部キーがそれぞれ1つずつしか存在できません。

E. 列レベルとテーブルレベルで定義できるのは主キーのみです。

正解: ([正解を表示します](#))

質問: 209

展示資料を参照し、プロモーション表の構造を調べてください。

Table PROMOTIONS		
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(6)
PROMO_NAME	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY_ID	NOT NULL	NUMBER
PROMO_CATEGORY	NOT NULL	VARCHAR2(30)
PROMO_CATEGORY_ID	NOT NULL	NUMBER
PROMO_COST	NOT NULL	NUMBER(10,2)
PROMO_BEGIN_DATE	NOT NULL	DATE
PROMO_END_DATE	NOT NULL	DATE

PROMOTIONstableを使用して、\$0~\$2000の範囲のすべてのプロモーションの平均コストを調べ、
カテゴリ-Aでは2000ドルから5000ドル。

以下のSQL文を発行します。

```
SQL>SELECT AVG(CASE
      WHEN promo_cost BETWEEN 0 AND 2000 AND promo_category='A'
      THEN promo_cost
      ELSE null END) "CAT_2000A",
      AVG(CASE
      WHEN promo_cost BETWEEN 2001 AND 5000 AND promo_category='A'
      THEN promo_cost
      ELSE null END) "CAT_5000A"
FROM promotions;
```

結果はどのようなだろうか？

- A. WHEN句には複数の条件を指定できないため、エラーが発生します。
- B. 正常に実行され、必要な結果が得られます。
- C. CASE はグループ関数では使用できないため、エラーが発生します。
- D. NULLを戻り値として指定できないため、エラーが発生します。

正解: ([正解を表示します](#))

CASE式

IF-THEN-ELSE文の役割を果たすことで、条件付き照会を容易にします。

```
CASE expr WHEN comparison_expr1 THEN return_expr1
```

```
[WHEN comparison_expr2 THEN return_expr2
```

```
WHEN comparison_exprn THEN return_exprn
```

```
ELSE else_expr]
```

終わり

質問: 210

展示資料をご覧になり、製品表と売上表をご確認ください。

展示資料 1

Table PRODUCTS		
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER (6)
PROD_NAME	NOT NULL	VARCHAR2 (50)
PROD_DESC	NOT NULL	VARCHAR2 (4000)
PROD_CATEGORY	NOT NULL	VARCHAR2 (50)
PROD_CATEGORY_ID	NOT NULL	NUMBER
PROD_UNIT_OF_MEASURE		VARCHAR2 (20)
SUPPLIER_ID	NOT NULL	NUMBER (6)
PROD_STATUS	NOT NULL	VARCHAR2 (20)
PROD_LIST_PRICE	NOT NULL	NUMBER (8, 2)
PROD_MIN_PRICE	NOT NULL	NUMBER (8, 2)

展示資料2

Table SALES		
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER
CUST_ID	NOT NULL	NUMBER
TIME_ID	NOT NULL	DATE
CHANNEL_ID	NOT NULL	NUMBER
PROMO_ID	NOT NULL	NUMBER
QUANTITY_SOLD	NOT NULL	NUMBER (10, 2)

商品名とその商品の販売回数を表示するには、以下のクエリを実行します。

```
SQL>SELECT p.prod_name, i.item_cnt
FROM (SELECT prod_id, COUNT(*) item_cnt
FROM sales
GROUP BY prod_id) I RIGHT OUTER JOIN products p
ON i.prod_id = p.prod_id;
```

上記のステートメントが実行されると、何が起こりますか？

- A. FROM句のサブクエリではGROUP BY句を使用できないため、このステートメントはエラーになります。
- B. ITEM_CNT は外部クエリで表示できないため、ステートメントはエラーになります。
- C. FROM句内のサブクエリと外部結合は併用できないため、このステートメントはエラーになります。
- D. ステートメントは正常に実行され、必要な出力が生成されます。

正解: **D** ([コメントを公表する](#))

質問: 211

このコマンドを実行します。

```
TRUNCATE TABLE depts;
```

正しいのは次のうちどれですか？

- A. ROLLBACK ステートメントを使用して、削除されたデータを復元できます。
- B. FLASHBACK TABLE文を使用して削除されたデータを取得できます。
- C. テーブルに定義されているトリガーをすべて削除します。
- D. 削除された行が使用していたスペースを常に保持します。
- E. テーブルに定義された整合性制約を保持します。
- F. テーブルで定義されたインデックスを保持します。

正解: ([正解を表示します](#))

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> 325問、30%ディスカウント、特別な割引コード: **JPNshiken**」

質問: 212

代入変数について正しい記述はどれですか？ (2つ選択してください。)

- A. & で始まる置換変数は、セッション中に常に一度だけ値の入力を求めます。
- B. 置換変数は、SELECT文のどの句でも使用できます。
- C. 置換変数はSELECT文でのみ使用できます。
- D. && で始まる置換変数は、セッション内で undefined に設定されていない限り、セッション内で一度だけ値の入力を求めます。
- E. 列名の入力を促すために使用される置換変数は、単一引用符で囲む必要があります。
- F. 列名の入力を促すために使用する置換変数は、二重引用符で囲む必要があります。

正解: C,D ([コメントを發表する](#))

質問: 213

このクエリを調べてください。

SELECT TRUNC (ROUND(156.00,-2),-1) FROM DUAL; 結果はどうなりますか？

- A. 16
- B. 160
- C. 150
- D. 200
- E. 100

正解: ([正解を表示します](#))

このクエリでは、ROUND 関数と TRUNC 関数の 2 つの関数を使用します。ROUND 関数は、小数点以下の桁数を指定する -2 によって、数値 156.00 を最も近い百の位に丸めます。結果は 200 になります。次に、TRUNC 関数は、引数 -1 によってこの数値を最も近い 10 位に切り捨てます。この場合、切り捨てによって丸められた値は変更されないため、結果は 200 になります。

- * A. 16 (不正解)
- * B. 160 (誤)
- * C. 150 (誤)
- * D. 200 (誤)

* E. 100 (不正解)

質問: 214

展示資料を参照し、PRODUCT_INFORMATIONテーブルのデータを確認してください。

PRODUCT_INFORMATION				
PDT_ID	SUP_ID	PDT_STATUS	LIST_PRICE	MIN_PRICE
1797	102094	orderable	349	288
2254	102071	obsolete	453	371
2382	102050	under development	850	731
2459	102099	under development	699	568
3127	102087	orderable	498	444
3353	102071	obsolete	489	413
3354	102066	orderable	543	478

サブクエリが必要となるタスクはどれですか？ (2つ選択してください。)

- A. 各製品のステータスごとに最低定価を表示する
- B. 注文可能な商品のうち、最低定価が平均定価を上回るすべての商品を表示します。
- C. 平均定価よりも高い定価の商品数を表示する
- D. 平均定価が500を超えるすべてのサプライヤーIDを表示します
- E. サプライヤー102071から供給され、製品ステータスがOBSOLETEである製品の総数を表示します。

正解: B,C (コメントを发表する)

質問: 215

この記述を検討してください。

姓を選択してください

従業員より

ORDER BY CASE WHEN salary = (SELECT MAX(salary) FROM employees)

そして A]

それ以外の場合は姓

終了、姓 DESC;

正しい記述は次のうちどれですか？ (2つ選択)

- A. 最高給与を得ている従業員の名前が、降順で最初に表示されます。
- B. 最高給与を得ている従業員の名前が昇順で最初に表示されます。
- C. 残りの従業員名はすべて昇順で表示されます。
- D. 残りの従業員名は、順不同で表示されます。
- E. 残りの従業員名はすべて降順で表示されます。
- F. 最高給与を得ている従業員の名前が、順不同で最初に表示されます。

正解: B,E (コメントを发表する)

ORDER BY句にCASE文を使用すると、CASE式の結果に基づいて条件付きソートを行うことができます。

- A. これは正しくありません。THEN 句の A]は、最高給与を得ている従業員を上位に並べ替えますが、彼らの姓が表示される順序を決定するものではありません。
- B. これは正しいです。THEN 句の A]によって、最高給与の受給者全員が上位に並び、また A]は姓のアルファベット順で前に来るため、昇順になります。

C. これは正しくありません。ELSE 句は残りの従業員を last_name でソートしますが、順序を指定していないため、デフォルトで昇順になります。これは、DESC を指定する ORDER BY 句の 2 番目の部分と矛盾します。

D. これは正しくありません。残りの従業員名の順序は ORDER BY 句で指定され、last_name DESC と記述されています。

E. これは正しいです。給与が最も高い従業員がソートされた後、残りの従業員の名前は、ORDER BY 句の last_name DESC により降順で表示されます。

F. この記述は正しくありません。CASE 文で指定された A は、最高給与を得ている従業員の明確な順序を示しています。

参照：

Oracle Database SQL 言語リファレンス、12c リリース 1 (12.1) "ORDER BY 句" Oracle Database SQL 言語リファレンス、12c リリース 1 (12.1) "CASE 式"

質問: 216

ORDER_ITEMS テーブルの説明を確認してください。

Name	Null	Type
ORDER_ID		NUMBER(38)
PRODUCT_ID		NUMBER(38)
QUANTITY		NUMBER(38)
UNIT_PRICE		NUMBER(10,2)

この不完全なクエリを調べてください。

```
SELECT DISTINCT quantity * unit_price total_paid FROM order_items ORDER BY <句>;
```

クエリが正常に完了するように、<clause> を置き換えることができるのは、次のうちどれですか？

- A. product_id
- B. 数量×単価
- C. 支払総額
- D. 数量
- E. 数量、単価

正解: ([正解を表示します](#))

質問: 217

給与が最も低い従業員 3 名の名前を返す 2 つの文はどれですか？

A. SELECT last_name,salary

従業員より

給与順に並べ替え

最初の 3 行のみを取得します。

B. SELECT last_name,salary

従業員より

WHERE ROWNUM<=3

ORDER BY (SELECT salary FROM employees);

C. SELECT last_name, salary

従業員より

WHERE ROWNUM<=3

D. SELECT last_name,salary

FROM (SELECT * FROM employees ORDER BY salary)

E. SELECT last_name,salary

従業員より

最初の3行のみ取得する

給与順に並べ替える。

正解: ([正解を表示します](#))

質問: 218

制約条件に関して正しい記述はどれですか？ 2つ選択してください。)

A. 制約は、テーブルに対する INSERT 操作に対してのみ適用されます。

B. すべての制約は、列レベルとテーブルレベルの両方で定義できます。

C. 外部キーにはNULL値を含めることはできません。

D. UNIQUE制約を持つ列にはNULLを含めることができます。

E. 制約列にデータが含まれていても、制約を無効にすることができます。

正解: ([正解を表示します](#))

質問: 219

展示資料を参照し、PRODUCT_INFORMATIONテーブルの詳細を確認してください。

製品名

カテゴリID サプライヤーID

インクジェット C/8/HQ 12 102094

インクジェット C/4 12 102090

LaserPro 600/6/BW 12 102087

LaserPro 1200/8/BW 12 102099

インクジェット B/6 12 102096

インダストリアル700/ID 12 102086

インダストリアル600/DQ 12 102088

コンパクト400/LQ 12 102087

コンパクト400/DQ 12 102088

HD 12GB /R 13 102090

HD 10GB /I 13 102071

HD 12GB @7200 /SE 13 102057

HD 18.2GB @10000 /E 13 102078

HD 18.2GB @10000 /I 13 102050

HD 18GB /SE 13 102083

HD 6GB /I 13 102072

HD 8.2GB@5400 13 102093

テーブルから、CATEGORY_ID列の値が12または13で、SUPPLIER_ID列の値が102088であるPRODUCT_NAMEを表示するという要件があります。次のSQLステートメントを実行しました。

SELECT product_name

製品情報から

WHERE (category_id = 12 AND category_id = 13) AND supplier_id = 102088; このクエリの実行に関して正しい記述はどれですか?

- A. 実行されますが、出力には行が返されません。
- B. AND論理演算子の両側で同じ列が条件を形成するために使用されているため、実行されません。
- C. 実行され、出力には目的の結果が表示されます。
- D. WHERE句の条件全体が括弧で囲まれていないため、実行されません。

正解: [\(正解を表示します\)](#)

質問: 220

複合クエリで交差演算子を使用した結果について、正しい記述は次のうちどれですか? (2つ選択)

- A. intersect は null を無視します。
- B. 交差するテーブルの順序を逆にすると、出力に影響が出る場合があります。
- C. 複合クエリ内の各SELECT文の列名は異なっても構いません。
- D. intersect は、複合クエリの両側に共通する行を返します。
- E. 複合クエリ内の各選択項目の列数は異なる場合があります。

正解: C,D ([コメントを発表する](#))

C) 正しい。INTERSECT クエリの各 SELECT ステートメントの列名は、データ型と列の順序が一致していれば、一致する必要はありません。D) 正しい。INTERSECT 演算子は、両方の SELECT ステートメントに共通する行のみを返し、両方のクエリの結果の集合の共通部分として機能します。

参考文献:

* INTERSECT演算子に関するOracleドキュメント :Oracle Database SQL言語リファレンス

* INTERSECTの詳細な動作 :Oracle複合クエリ

質問: 221

従業員一覧表の説明を確認してください。

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(4)
LAST_NAME	NOT NULL	VARCHAR2(100)
SALARY	NOT NULL	NUMBER(6,2)
DEPARTMENT_ID	NOT NULL	NUMBER(4)

このクエリを調べてください。

```
1  SELECT e.last_name,  
2         e.salary,  
3         a.avg_sal  
4  FROM employees e  
5  WHERE e.salary > (SELECT AVG(a.salary) AS avg_sal  
6                    FROM employees a  
7                    WHERE a.department_id = e.department_id)  
8 ORDER BY e.last_name;
```

どの行でエラーが発生しますか?

- A. 7行目
- B. 5番線

C. 3行目

D. 8行目

正解: [\(正解を表示します\)](#)

質問: 222

どの2つのクエリが正常に実行されましたか？

A. SELECT INTERVAL '1' DAY - SYSDATE FROM DUAL;

B. SELECT SYSTIMESTAMP + INTERVAL '1' DAY FROM DUAL;

C. SELECT INTERVAL '1' DAY - INTERVAL '1' MINUTE FROM DUAL;

D. DUALからINTERVAL '1' DAY +INTERVAL '1' MONTHを選択します。

E. SELECT SYSDATE "INTERVAL '1' DAY FROM DUAL;

正解: [\(正解を表示します\)](#)

* A: このステートメントは、日付から日数を直接減算することはできないため、正常に実行されません。SYSDATE を最初に TIMESTAMP 型にキャストする必要があります。

* B: この記述は正しいです。現在のタイムスタンプに『1』日分の間隔を追加します。

* C: この記述は正しいです。『1』日から『1』分を減算します。

* D: このステートメントは正常に実行されません。Oracleでは、異なる日付フィールド（日と月）の間隔を直接加算することはできません。

* E: このステートメントには、INTERVAL を囲む引用符の構文エラーとスペルミスがあります。正しくは次のようになります。

'間隔'。

Oracle Database 12c SQLは、SQL言語リファレンスドキュメントに記載されているように、区間演算をサポートしています。

質問: 223

次の記述を評価してください。

```
INSERT ALL
```

```
  WHEN order_total < 10000 THEN
```

```
  INTO small_orders
```

```
  WHEN order_total > 10000 AND order_total < 20000 THEN
```

```
    INTO medium_orders
```

```
  WHEN order_total > 2000000 THEN
```

```
    INTO large_orders
```

```
  SELECT order_id, order_total, customer_id
```

```
  FROM orders;
```

INSERT文のサブクエリによって返される行の評価に関して、正しい記述はどれですか？

A. ELSE句が欠落しているため、INSERT文はエラーを返します。

B. すべての行は、3つのWHEN句すべてによって評価されます。

C. 各行は最初のWHEN句によって評価され、条件が偽の場合は、その行は後続のWHEN句によって評価されます。

D. 各行は最初のWHEN句によって評価され、条件が真であれば、その行は後続のWHEN句によって評価されます。

正解: [\(正解を表示します\)](#)

質問: 224

Oracleデータベースのビューについて、正しい記述を3つ選びなさい。(3つ選択してください。)

- A. ビューを使用してテーブルに挿入された行は、ビューが削除されてもテーブルに保持されます。
- B. 定義クエリに WHERE 句を含むビューをクエリする場合、SELECT ステートメントには WHERE 句を含めることはできません。
- C. 定義クエリで存在しないテーブルを参照するビューを作成できます。
- D. ビューにセグメントがありません。
- E. ビューにはオブジェクト番号がありません。
- F. ビューは、同じスキーマに属している場合にのみテーブルを結合できます。

正解: ([正解を表示します](#))

有効的な1z0-071問題集はJPNTTest.com提供され、1z0-071試験に合格することに役に立ちます！JPNTTest.comは今最新1z0-071試験問題集を提供します。JPNTTest.com 1z0-071試験問題集はもう更新されました。ここで1z0-071問題集のテストエンジンを手に入れます。最新版のアクセス、<https://www.jpntest.com/shiken/1z0-071-mondaishu> **325問、30%ディスカウント**、特別な割引コード: **JPNshiken**」