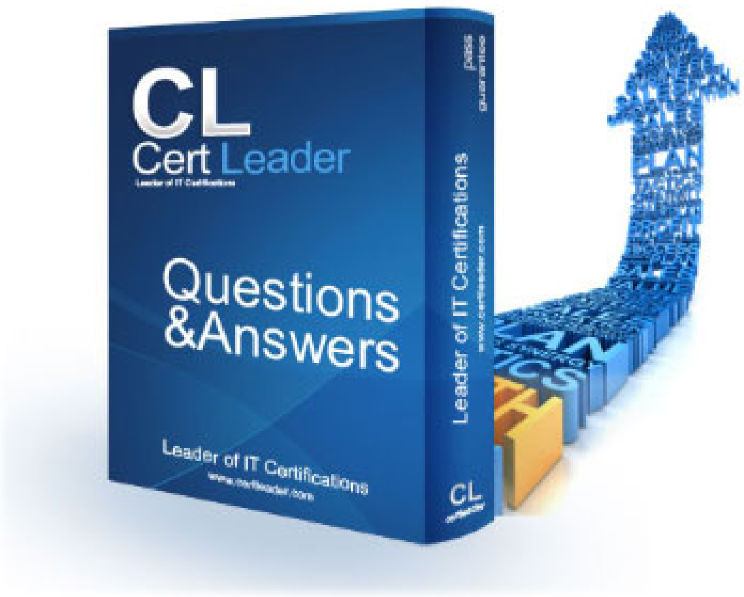
1Z0-051 Dumps

Oracle Database: SQL Fundamentals I

https://www.certleader.com/1Z0-051-dumps.html



NEW QUESTION 1
Which two statements are true about sequences created in a single instance database? (Choose two.)

A. CURRVAL is used to refer to the last sequence number that has been generated
B. DELETE <sequencename> would remove a sequence from the database
C. The numbers generated by a sequence can be used only for one table
D. When the MAXVALUE limit for a sequence is reached, you can increase the MAXVALUE limit by using the ALTER SEQUENCE statement
E. When a database instance shuts down abnormally, the sequence numbers that have been cached but not used would be available once again when the database instance is restarted

Answer: AD

Explanation:
Gaps in the Sequence
Although sequence generators issue sequential numbers without gaps, this action occurs independent of a commit or rollback. Therefore, if you roll back a statement containing a sequence, the number is lost.
Another event that can cause gaps in the sequence is a system crash. If the sequence caches values in memory, those values are lost if the system crashes.
Because sequences are not tied directly to tables, the same sequence can be used for multiple tables.
However, if you do so, each table can contain gaps in the sequential numbers.
Modifying a Sequence
If you reach the MAXVALUE limit for your sequence, no additional values from the sequence are allocated and you will receive an error indicating that the sequence exceeds the MAXVALUE. To continue to use the sequence, you can modify it by using the ALTER SEQUENCE statement To remove a sequence, use the DROP statement:
DROP SEQUENCE dept_deptid_seq;

NEW QUESTION 2
When does a transaction complete? (Choose all that apply.)

A. When a PL/SQL anonymous block is executed
B. When a DELETE statement is executed
C. When a data definition language statement is executed
D. When a TRUNCATE statement is executed after the pending transaction
E. When a ROLLBACK command is executed

Answer: CDE

NEW QUESTION 3
View the Exhibits and examine the structures of the PROMOTIONS and SALES tables.

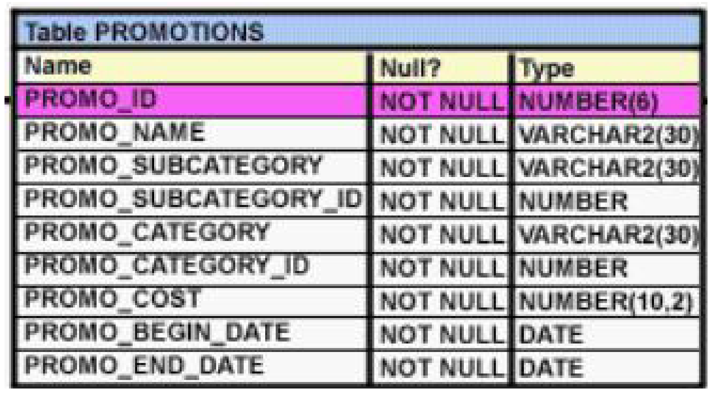


Table PROMOTIONS

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

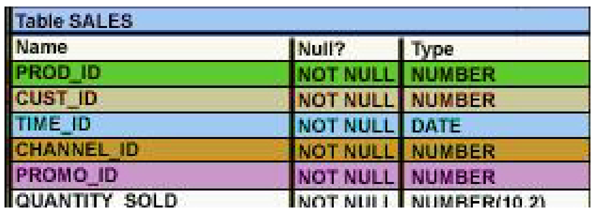


Table SALES

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)

Evaluate the following SQL statements:
Which statement is true regarding the output of the above query?

A. It gives details of product IDs that have been sold irrespective of whether they had a promo or not
B. It gives the details of promos for which there have been no sales
C. It gives the details of promos for which there have been sales
D. It gives details of all promos irrespective of whether they have resulted in a sale or not

Answer: D

NEW QUESTION 4
Which two statements are true regarding the USING and ON clauses in table joins? (Choose two.)

A. The ON clause can be used to join tables on columns that have different names but compatible data types
B. A maximum of one pair of columns can be joined between two tables using the ON clause
C. Both USING and ON clause can be used for equijoins and nonequijoins
D. The WHERE clause can be used to apply additional conditions in SELECT statement containing the ON or the USING clause

Answer: AD

Explanation:
Creating Joins with the USING Clause If several columns have the same names but the data types do not match, use the USING clause to specify the columns for the equijoin. Use the USING clause to match only one column when more than one column matches. The NATURAL JOIN and USING clauses are mutually exclusive Using Table Aliases with the USING clause When joining with the USING clause, you cannot qualify a column that is used in the USING clause itself. Furthermore, if that column is used anywhere in the SQL statement, you cannot alias it. For example, in the query mentioned in the slide, you should not alias the location_id column in the WHERE clause because the column is used in the USING clause. The columns that are referenced in the USING clause should not have a qualifier (table name oralias) anywhere in the SQL statement. Creating Joins with the ON Clause The join condition for the natural join is basically an equijoin of all columns with the same name. Use the ON clause to specify arbitrary conditions or specify columns to join. – ANSWER C The join condition is separated from other search conditions. ANSWER D
The ON clause makes code easy to understand.

NEW QUESTION 5
Which is the valid CREATE [TABLE statement?

A. CREATE TABLE emp9$# (emp_no NUMBER(4));
B. CREATE TABLE 9emp$# (emp_no NUMBER(4));
C. CREATE TABLE emp*123 (emp_no NUMBER(4));
D. CREATE TABLE emp9$# (emp_no NUMBER(4). date DATE);

Answer: A

Explanation:
Schema Object Naming Rules
Every database object has a name. In a SQL statement, you represent the name of an object with a quoted identifier or a nonquoted identifier.
A quoted identifier begins and ends with double quotation marks ("). If you name a schema object using a quoted identifier, then you must use the double quotation marks whenever you refer to that object.
A nonquoted identifier is not surrounded by any punctuation.
The following list of rules applies to both quoted and nonquoted identifiers unless otherwise indicated:
Names must be from 1 to 30 bytes long with these exceptions:
Names of databases are limited to 8 bytes.
Names of database links can be as long as 128 bytes.
If an identifier includes multiple parts separated by periods, then each attribute can be up to 30 bytes long.
Each period separator, as well as any surrounding double quotation marks, counts as one byte. For example, suppose you identify a column like this:
"schema"."table"."column"
Nonquoted identifiers cannot be Oracle Database reserved words (ANSWER D). Quoted identifiers can be reserved words, although this is not recommended. Depending on the Oracle product you plan to use to access a database object, names might be further restricted by other product-specific reserved words. The Oracle SQL language contains other words that have special meanings. These words include datatypes, schema names, function names, the dummy system table DUAL, and keywords (the uppercase words in SQL statements, such as DIMENSION, SEGMENT, ALLOCATE, DISABLE, and so forth). These words are not reserved. However, Oracle uses them internally in specific ways. Therefore, if you use these words as names for objects and object parts, then your SQL statements may be more difficult to read and may lead to unpredictable results. In particular, do not use words beginning with SYS_ as schema object names, and do not use the names of SQL built-in functions for the names of schema objects or user-defined functions. You should use ASCII characters in database names, global database names, and database link names, because ASCII characters provide optimal compatibility across different platforms and operating systems. Nonquoted identifiers must begin with an alphabetic character (ANSWER B - begins with 9) from your database character set. Quoted identifiers can begin with any character. Nonquoted identifiers can contain only alphanumeric characters from your database character set and the underscore (_), dollar sign ($), and pound sign (#). Database links can also contain periods (.) and "at" signs (@). Oracle strongly discourages you from using $ and # in nonquoted identifiers. Quoted identifiers can contain any characters and punctuations marks as well as spaces. However, neither quoted nor nonquoted identifiers can contain double quotation marks or the null character (\0). Within a namespace, no two objects can have the same name. Nonquoted identifiers are not case sensitive. Oracle interprets them as uppercase. Quoted identifiers are case sensitive. By enclosing names in double quotation marks, you can give the following names to different objects in the same namespace: employees "employees" "Employees" "EMPLOYEES"
Note that Oracle interprets the following names the same, so they cannot be used for different objects in the same namespace: employees EMPLOYEES "EMPLOYEES" Columns in the same table or view cannot have the same name. However, columns in different tables or views can have the same name. Procedures or functions contained in the same package can have the same name, if their arguments are not of the same number and datatypes. Creating multiple procedures or functions with the same name in the same package with different arguments is called overloading the procedure or function.

NEW QUESTION 6
See the Exhibit and Examine the structure of the CUSTOMERS table:

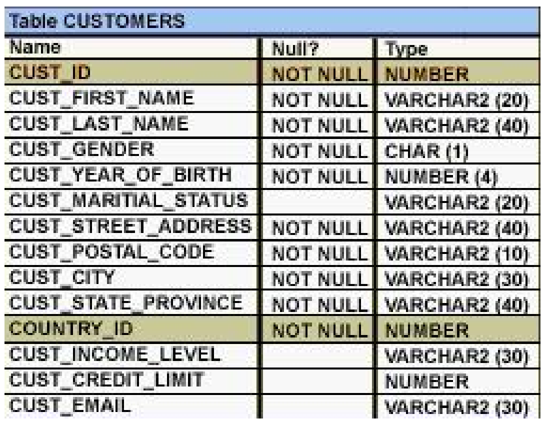


Table CUSTOMERS

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_MARITIAL_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)

Using the CUSTOMERS table, you need to generate a report that shows an increase in the credit limit by 15% for all customers. Customers whose credit limit has not been entered should have the message "Not Available" displayed.
Which SQL statement would produce the required result?

A. SELECT NVL(cust_credit_limit,'Not Available')*.15 "NEW CREDIT" FROM customers;
B. SELECT NVL(cust_credit_limit*.15,'Not Available') "NEW CREDIT" FROM customers;
C. SELECT TO_CHAR(NVL(cust_credit_limit*.15,'Not Available')) "NEW CREDIT" FROM customers;
D. SELECT NVL(TO_CHAR(cust_credit_limit*.15),'Not Available') "NEW CREDIT" FROM customers;

Answer: D

Explanation:
NVL Function
Converts a null value to an actual value:
Data types that can be used are date, character, and number.
Data types must match:
–
NVL(commission_pct,0)
–
NVL(hire_date,'01-JAN-97')
–
NVL(job_id,'No Job Yet')

NEW QUESTION 7
View the Exhibit and examine the description for the CUSTOMERS table.

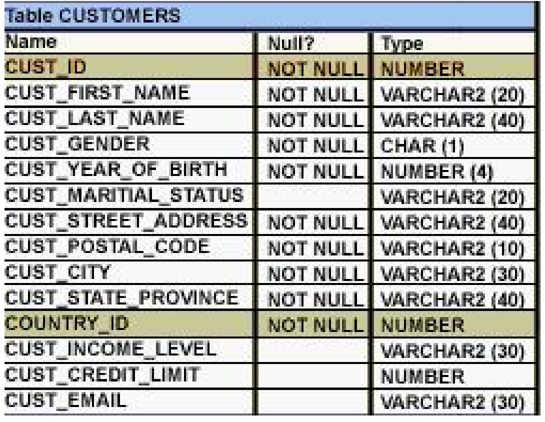


Table CUSTOMERS

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_MARITIAL_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)

You want to update the CUST_CREDIT_LIMIT column to NULL for all the customers, where CUST_INCOME_LEVEL has NULL in the CUSTOMERS table. Which SQL statement will accomplish the task?

A. UPDATE customers SET cust_credit_limit = NULL WHERE CUST_INCOME_LEVEL = NULL;
B. UPDATE customers SET cust_credit_limit = NULL WHERE cust_income_level IS NULL;
C. UPDATE customers SET cust_credit_limit = TO_NUMBER(NULL) WHERE cust_income_level = TO_NUMBER(NULL);
D. UPDATE customers SET cust_credit_limit = TO_NUMBER(' ',9999) WHERE cust_income_level IS NULL;

Answer: B

NEW QUESTION 8
View the Exhibit and examine the structure of the PRODUCTS table. You need to generate a report in the following format: CATEGORIES 5MP Digital Photo Camera's category is Photo Y Box's category is Electronics Envoy Ambassador's category is Hardware Which two queries would give the required output? (Choose two.)

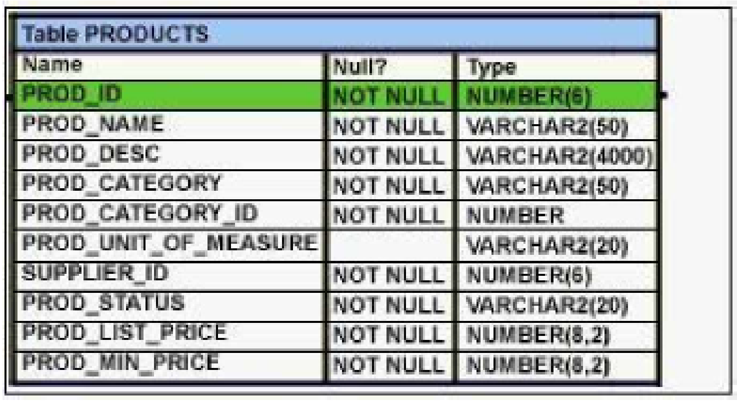


Table PRODUCTS

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)

A. SELECT prod_name || q'''s category is ' || prod_category CATEGORIES FROM products;
B. SELECT prod_name || q'['s]'category is ' || prod_category CATEGORIES FROM products;
C. SELECT prod_name || q'\'s\' || ' category is ' || prod_category CATEGORIES FROM products;
D. SELECT prod_name || q'<'s >' || 'category is ' || prod_category CATEGORIES FROM products;

Answer: CD

Explanation:
So, how are words that contain single quotation marks dealt with? There are essentially two mechanisms available. The most popular of these is to add an additional single quotation mark next to each naturally occurring single quotation mark in the character string Oracle offers a neat way to deal with this type of character literal in the form of the alternative quote (q) operator. Notice that the problem is that Oracle chose the single quote characters as the special pair of symbols that enclose or wrap any other character literal. These character-enclosing symbols could have been anything other than single quotation marks. Bearing this in mind, consider the alternative quote (q) operator. The q operator enables you to choose from a set of possible pairs of wrapping symbols for character literals as alternatives to the single quote symbols. The options are any single-byte or multibyte character or the four brackets: (round brackets), {curly braces}, [squarebrackets], or <angle brackets>. Using the q operator, the character delimiter can effectively be changed from a single quotation mark to any other character The syntax of the alternative quote operator is as follows: q'delimiter'character literal which may include the single quotes delimiter' where delimiter can be any character or bracket.
Alternative Quote (q) Operator
Specify your own quotation mark delimiter.

Select any delimiter.
Increase readability and usability.
SELECT department_name || q'[Department's Manager Id:]'
|| manager_id
AS "Department and Manager"
FROM departments;
Alternative Quote (q) Operator
Many SQL statements use character literals in expressions or conditions. If the literal itself contains a single quotation mark, you can use the quote (q) operator and select your own quotation mark delimiter.
You can choose any convenient delimiter, single-byte or multibyte, or any of the following character pairs: [ ], { }, ( ), or < >. In the example shown, the string contains a single quotation mark, which is normally interpreted as a delimiter of a character string. By using the q operator, however, brackets [] are used as the quotation mark delimiters. The string between the brackets delimiters is interpreted as a literal character string.

NEW QUESTION 9
See the Exhibit and examine the structure of the PROMOSTIONS table: Exhibit:

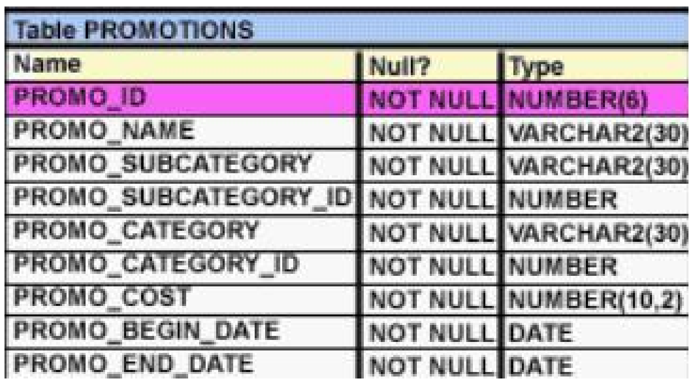


Table PROMOTIONS

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

Which SQL statements are valid? (Choose all that apply.)

A. SELECT promo_id, DECODE(NVL(promo_cost,0), promo_cost, promo_cost * 0.25, 100) "Discount" FROM promotions;
B. SELECT promo_id, DECODE(promo_cost, 10000, DECODE(promo_category, 'G1', promo_cost *.25, NULL), NULL) "Catcost" FROM promotions;
C. SELECT promo_id, DECODE(NULLIF(promo_cost, 10000), NULL, promo_cost*.25, 'N/A') "Catcost" FROM promotions;
D. SELECT promo_id, DECODE(promo_cost, >10000, 'High', <10000, 'Low') "Range" FROM promotions;

Answer: AB

Explanation:
The DECODE Function Although its name sounds mysterious, this function is straightforward. The DECODE function implements ifthen-else conditional logic by testing its first two terms for equality and returns the third if they are equal and optionally returns another term if they are not. The DECODE function takes at least three mandatory parameters, but can take many more. The syntax of the function is DECODE(expr1,comp1, iftrue1, [comp2,iftrue2...[compN,iftrueN]], [iffalse]).

NEW QUESTION 10
The user Alice wants to grant all users query privileges on her DEPT table. Which SQL statement accomplishes this?

A. GRANT select ON dept TO ALL_USERS;
B. GRANT select ON dept TO ALL;
C. GRANT QUERY ON dept TO ALL_USERS
D. GRANT select ON dept TO PUBLIC;

Answer: D

Explanation: view the columns associated with the constraint names in the USER_CONS_COLUMNS view.
Incorrect Answer: Atable to view all constraints definition and names Bshow all object name belong to user Cdoes not display column associated Eno such view
Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 10-25

NEW QUESTION 10
Which two statements about sub queries are true? (Choose two.)

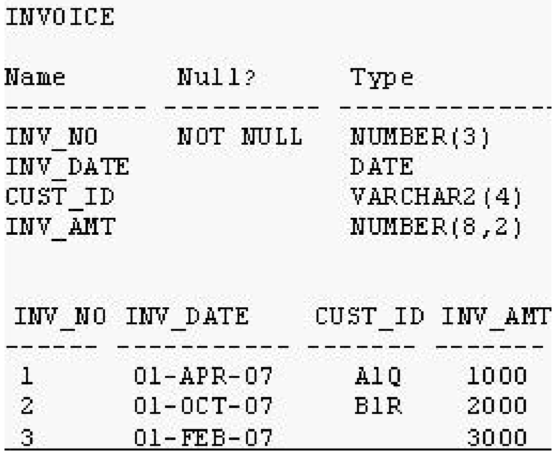
A. A sub query should retrieve only one ro
B. A sub query can retrieve zero or more row
C. A sub query can be used only in SQL query statement
D. Sub queries CANNOT be nested by more than two level
E. A sub query CANNOT be used in an SQL query statement that uses group function
F. When a sub query is used with an inequality comparison operator in the outer SQL statement, the column list in the SELECT clause of the sub query should contain only one colum

Answer: BF

Explanation: Explanation: sub query can retrieve zero or more rows, sub query is used with an inequality comparison operator in the outer SQL statement, and the column list in the SELECT clause of the sub query should contain only one column.
Incorrect Answer: Asub query can retrieve zero or more rows Csub query is not SQL query statement Dsub query can be nested Egroup function can be use with sub query

NEW QUESTION 15

See the Exhibit and examine the structure and data in the INVOICE table: Exhibit:



```
INVOICE

Name       Null?      Type
--------- ---------- --------------
INV_NO     NOT NULL   NUMBER(3)
INV_DATE              DATE
CUST_ID               VARCHAR2(4)
INV_AMT               NUMBER(8,2)


 INV_NO INV_DATE    CUST_ID INV_AMT
------ ----------- ------- -------
 1      01-APR-07    A1Q     1000
 2      01-OCT-07    B1R     2000
 3      01-FEB-07            3000
```

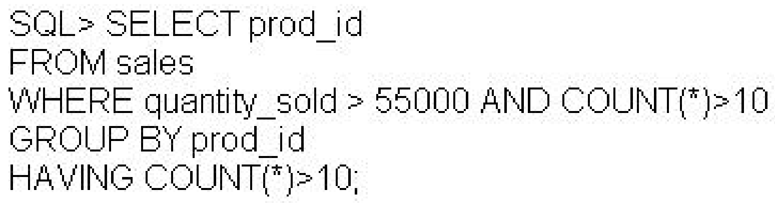
Which two SQL statements would executes successfully? (Choose two.)

A. SELECT MAX(inv_date),MIN(cust_id) FROM invoice;
B. SELECT MAX(AVG(SYSDATE - inv_date)) FROM invoice;
C. SELECT (AVG(inv_date) FROM invoice;
D. SELECT AVG(inv_date - SYSDATE),AVG(inv_amt) FROM invoice;

Answer: AD

NEW QUESTION 20

View the Exhibit and examine the structure of the SALES table.
The following query is written to retrieve all those product IDs from the SALES table that have more than 55000 sold and have been ordered more than 10 times.



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

Which statement is true regarding this SQL statement?

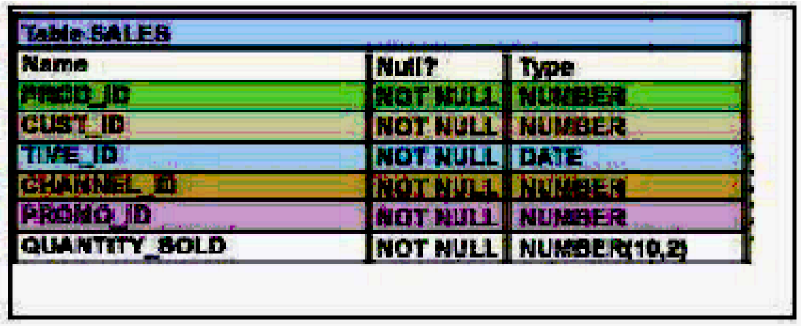


Table SALES		
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER
CUST_ID	NOT NULL	NUMBER
TIME_ID	NOT NULL	DATE
CHANNEL_ID	NOT NULL	NUMBER
PROMO_ID	NOT NULL	NUMBER
QUANTITY_SOLD	NOT NULL	NUMBER(10,2)

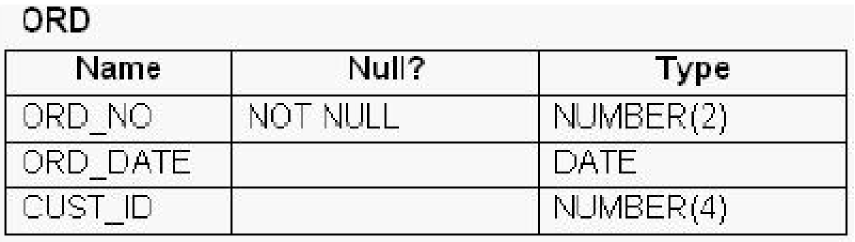
A. It executes successfully and generates the required resul
B. It produces an error because COUNT(*) should be specified in the SELECT clause als
C. It produces an error because COUNT(*) should be only in the HAVING clause and not in the WHERE claus
D. It executes successfully but produces no result because COUNT(prod_id) should be used instead of COUNT(*).

Answer: C

Explanation:
Restricting Group Results with the HAVING Clause
You use the HAVING clause to specify the groups that are to be displayed, thus further restricting the groups on the basis of aggregate information.
In the syntax, group_condition restricts the groups of rows returned to those groups for which the specified condition is true.
The Oracle server performs the following steps when you use the HAVING clause:
1.
Rows are grouped.
2.
The group function is applied to the group.
3.
The groups that match the criteria in the HAVING clause are displayed.
The HAVING clause can precede the GROUP BY clause, but it is recommended that you place the GROUP BY clause first because it is more logical. Groups are formed and group functions are calculated before the HAVING clause is applied to the groups in the SELECT list.
Note: The WHERE clause restricts rows, whereas the HAVING clause restricts groups.

NEW QUESTION 24

See the Exhibit and examine the structure of ORD table: Exhibit:



ORD

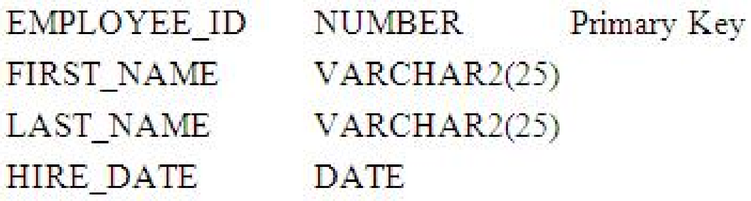
Name	Null?	Type
ORD_NO	NOT NULL	NUMBER(2)
ORD_DATE		DATE
CUST_ID		NUMBER(4)

Evaluate the following SQL statements that are executed in a user session in the specified order:
CREATE SEQUENCE ord_seq;
SELECT ord_seq.nextval
FROM dual;
INSERT INTO ord
VALUES (ord_seq.CURRVAL, '25-jan-2007,101);
UPDATE ord
SET ord_no= ord_seq.NEXTVAL
WHERE cust_id =101;
What would be the outcome of the above statements?

A. All the statements would execute successfully and the ORD_NO column would contain the value 2 for the CUST_ID 101.
B. The CREATE SEQUENCE command would not execute because the minimum value and maximum value for the sequence have not been specifie
C. The CREATE SEQUENCE command would not execute because the starting value of the sequence and the increment value have not been specifie
D. All the statements would execute successfully and the ORD_NO column would have the value 20 for the CUST_ID 101 because the default CACHE value is 20.

Answer: A

NEW QUESTION 25
Examine the structure of the EMPLOYEES table:



EMPLOYEE_ID	NUMBER	Primary Key
FIRST_NAME	VARCHAR2(25)	
LAST_NAME	VARCHAR2(25)	
HIRE_DATE	DATE	

Which INSERT statement is valid?

A. INSERT INTO employees (employee_id, first_name, last_name, hire_date) VALUES (1000, 'John', 'Smith', '01/01/01');
B. INSERT INTO employees(employee_id, first_name, last_name, hire_date) VALUES (1000, 'John', 'Smith', '01 January 01');
C. INSERT INTO employees(employee_id, first_name, last_name, Hire_date) VALUES (1000, 'John', 'Smith', To_date('01/01/01'));
D. INSERT INTO employees(employee_id, first_name, last_name, hire_date) VALUES (1000, 'John', 'Smith', 01-Jan-01);

Answer: D

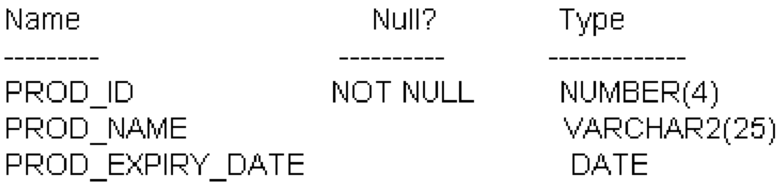
Explanation: It is the only statement that has a valid date; all other will result in an error. Answer A is incorrect, syntax error, invalid date format

NEW QUESTION 27
Evaluate this SQL statement:
SELECT e.emp_name, d.dept_name
FROM employees e
JOIN departments d
USING (department_id)
WHERE d.department_id NOT IN (10,40)
ORDER BY dept_name;
The statement fails when executed. Which change fixes the error?

A. remove the ORDER BY clause
B. remove the table alias prefix from the WHERE clause
C. remove the table alias from the SELECT clause
D. prefix the column in the USING clause with the table alias
E. prefix the column in the ORDER BY clause with the table alias
F. replace the condition "d.department_id NOT IN (10,40)" in the WHERE clause with "d.department_id <> 10 AND d.department_id <> 40"

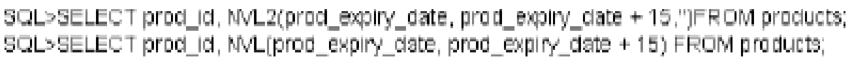
Answer: B

NEW QUESTION 31
The PRODUCTS table has the following structure:



```
Name                     Null?        Type
---------                ----------   -------------
PROD_ID                  NOT NULL     NUMBER(4)
PROD_NAME                             VARCHAR2(25)
PROD_EXPIRY_DATE                      DATE
```

Evaluate the following two SQL statements:



```
SQL>SELECT prod_id, NVL2(prod_expiry_date, prod_expiry_date + 15,'')FROM products;
SQL>SELECT prod_id, NVL(prod_expiry_date, prod_expiry_date + 15) FROM products;
```

Which statement is true regarding the outcome?

A. Both the statements execute and give the same result
B. Both the statements execute and give different results
C. Only the second SQL statement executes successfully
D. Only the first SQL statement executes successfully

Answer: B

Explanation:

Using the NVL2 Function The NVL2 function examines the first expression. If the first expression is not null, the NVL2 function returns the second expression. If the first expression is null, the third expression is returned.
Syntax NVL2(expr1, expr2, expr3) In the syntax: expr1 is the source value or expression that may contain a null expr2 is the value that is returned if expr1 is not null expr3 is the value that is returned if expr1 is null

NEW QUESTION 36
See the structure of the PROGRAMS table:

```
Name              Null?        Type
----------        -------------  ----------------
PROG_ID           NOT NULL     NUMBER(3)
PROG_COST                      NUMBER(8,2)
START_DATE        NOT NULL     DATE
END_DATE                       DATE
```

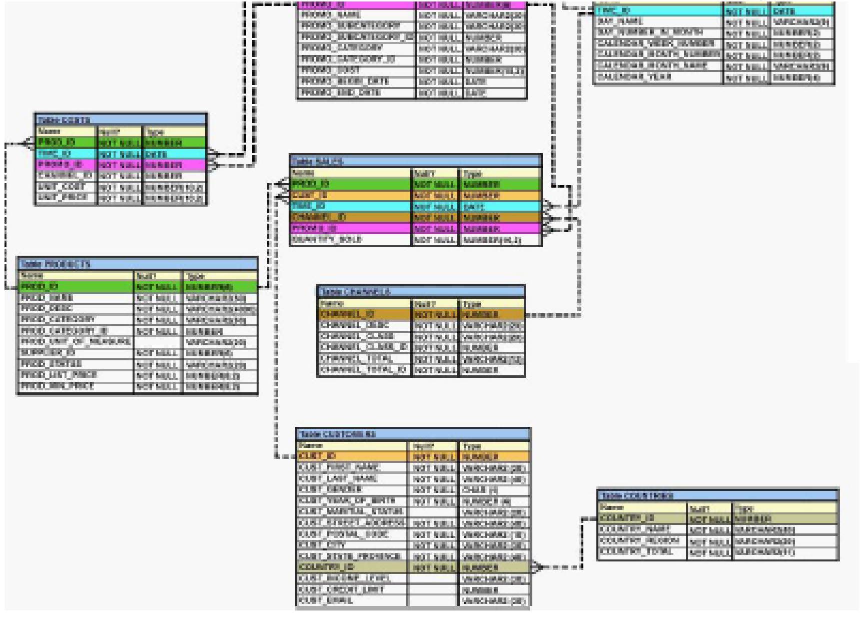
Which two SQL statements would execute successfully? (Choose two.)

A. SELECT NVL(ADD_MONTHS(END_DATE,1),SYSDATE) FROM programs;
B. SELECT TO_DATE(NVL(SYSDATE-END_DATE,SYSDATE)) FROM programs;
C. SELECT NVL(MONTHS_BETWEEN(start_date,end_date),'Ongoing') FROM programs;
D. SELECT NVL(TO_CHAR(MONTHS_BETWEEN(start_date,end_date)),'Ongoing') FROM programs;

Answer: AD

Explanation:
NVL Function
Converts a null value to an actual value:
Data types that can be used are date, character, and number.
Data types must match:
–
NVL(commission_pct,0)
–
NVL(hire_date,'01-JAN-97')
–
NVL(job_id,'No Job Yet')
MONTHS_BETWEEN(date1, date2): Finds the number of months between date1 and date2 The result can be positive or negative. If date1 is later than date2, the result is positive; if date1 is earlier than date2, the result is negative. The noninteger part of the result represents a portion of the month. MONTHS_BETWEEN returns a numeric value. - answer C NVL has different datatypes -numeric and strings, which is not possible!
The data types of the original and if null parameters must always be compatible. They must either be of the same type, or it must be possible to implicitly convert if null to the type of the original parameter. The NVL function returns a value with the same data type as the original parameter.

NEW QUESTION 37
You work as a database administrator at ABC.com. You study the exhibit carefully and examine the structure of CUSTOMRS AND SALES tables.



Evaluate the following SQL statement: Exhibit:

```
UPDATE (SELECT prod_id, cust_id, quantity_sold, time_id
           FROM sales)
SET time_id = '22-MAR-2007'
WHERE cust_id = (SELECT cust_id
                    FROM customers
                    WHERE cust_last_name = 'Roberts' AND
                    credit_limit = 600);
```

Which statement is true regarding the execution of the above UPDATE statement?

A. It would execute and restrict modifications to only the column specified in the SELECT statement
B. It would not execute because two tables cannot be used in a single UPDATE statement
C. It would not execute because a sub query cannot be used in the WHERE clause of an UPDATE statement
D. It would not execute because the SELECT statement cannot be used in place of the table name

Answer: A

NEW QUESTION 38
Which one is a system privilege?

A. SELECT
B. DELETE
C. EXECUTE
D. ALTER TABLE
E. CREATE TABLE

Answer: E

NEW QUESTION 39
Which statement is true regarding the UNION operator?

A. The number of columns selected in all SELECT statements need to be the same
B. Names of all columns must be identical across all SELECT statements
C. By default, the output is not sorted
D. NULL values are not ignored during duplicate checking

Answer: A

Explanation:
The SQL UNION query allows you to combine the result sets of two or more SQL SELECT statements. It removes duplicate rows between the various SELECT statements. Each SQL SELECT statement within the UNION query must have the same number of fields in the result sets with similar data types.

NEW QUESTION 43
Which statement is true regarding the default behavior of the ORDER BY clause?

A. In a character sort, the values are case-sensitive
B. NULL values are not considered at all by the sort operation
C. Only those columns that are specified in the SELECT list can be used in the ORDER BY clause
D. Numeric values are displayed from the maximum to the minimum value if they have decimal positions

Answer: A

Explanation:
Character Strings and Dates
Character strings and date values are enclosed with single quotation marks.
Character values are case-sensitive and date values are format-sensitive.
The default date display format is DD-MON-RR.

NEW QUESTION 45
You work as a database administrator at ABC.com. You study the exhibit carefully.

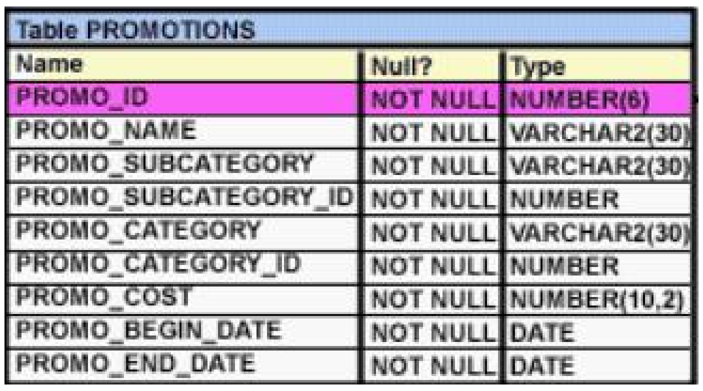


Table PROMOTIONS

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

Exhibit:
Which two SQL statements would execute successfully? (Choose two.)

A. UPDATE promotions SET promo_cost = promo_cost+ 100 WHERE TO_CHAR(promo_end_date, 'yyyy') > '2000';
B. SELECT promo_begin_date FROM promotions WHERE TO_CHAR(promo_begin_date,'mon dd yy')='jul 01 98';
C. UPDATE promotions SET promo_cost = promo_cost+ 100 WHERE promo_end_date > TO_DATE(SUBSTR('01-JAN-2000',8));
D. SELECT TO_CHAR(promo_begin_date,'dd/month') FROM promotions WHERE promo_begin_date IN (TO_DATE('JUN 01 98'), TO_DATE('JUL 01 98'));

Answer: AB

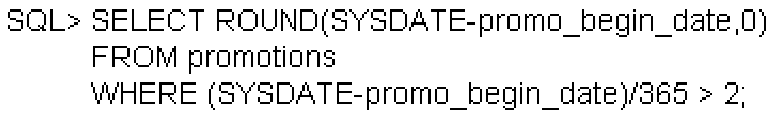
NEW QUESTION 48
You need to generate a list of all customer last names with their credit limits from the CUSTOMERS table. Those customers who do not have a credit limit should appear last in the list. Winch two queries would achieve the required result? (Choose two.)

A. SELECT cust_last_nam
B. cust_credit_limit FROM customers ORDER BY cust_credit_limit DESC:
C. SELECT cust_last_nam
D. cust_credit_limit FROM customers ORDER BY cust_credit_limit:
E. SELECT cust_last_nam
F. cust_credit_limit FROM customers ORDER BY cust_credit_limit NULLS LAST:
G. SELECT cust_last_nam
H. cust_credit_limit FROM customers ORDER BY cust_last_nam
I. cust_credit_limit NULLS LAST:

Answer: BC

Explanation:
If the ORDER BY clause is not used, the sort order is undefined, and the Oracle server may not fetch rows in the same order for the same query twice. Use the ORDER BY clause to display the rows in a specific order. Note: Use the keywords NULLS FIRST or NULLS LAST to specify whether returned rows containing null values should appear first or last in the ordering sequence. ANSWER C Sorting The default sort order is ascending:
.
Numeric values are displayed with the lowest values first (for example, 1 to 999).
.
Date values are displayed with the earliest value first (for example, 01-JAN-92 before 01-JAN-95).
.
Character values are displayed in the alphabetical order (for example, “A” first and “Z” last).
.
Null values are displayed last for ascending sequences and first for descending sequences.
-ANSWER B
. You can also sort by a column that is not in the SELECT list.

NEW QUESTION 50
You are currently located in Singapore and have connected to a remote database in
Chicago.
You issue the following command:
Exhibit:



```
SQL> SELECT ROUND(SYSDATE-promo_begin_date,0)
        FROM promotions
        WHERE (SYSDATE-promo_begin_date)/365 > 2;
```

PROMOTIONS is the public synonym for the public database link for the PROMOTIONS table.
What is the outcome?

A. Number of days since the promo started based on the current Singapore data and tim
B. An error because the ROUND function specified is invalid
C. An error because the WHERE condition specified is invalid
D. Number of days since the promo started based on the current Chicago data and time

Answer: D

NEW QUESTION 54
Which three statements are true about multiple-row sub queries? (Choose three.)

A. They can contain a subquery within a sub quer
B. They can return multiple columns as well as row
C. They cannot contain a sub query within a sub quer
D. They can return only one column but multiple row
E. They can contain group functions and GROUP BY and HAVING clause
F. They can contain group functions and the GROUP BY clause, but not the HAVING claus

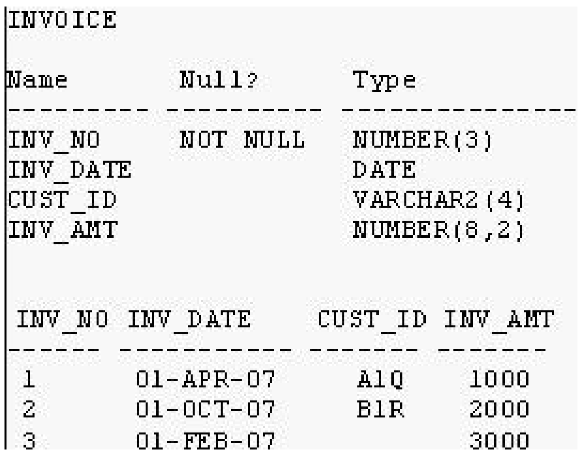
Answer: ABE

NEW QUESTION 59
Which three statements are true regarding sub queries? (Choose three.)

A. Multiple columns or expressions can be compared between the main query and sub query
B. Main query and sub query can get data from different tables
C. Sub queries can contain GROUP BY and ORDER BY clauses
D. Main query and sub query must get data from the same tables
E. Sub queries can contain ORDER BY but not the GROUP BY clause
F. Only one column or expression can be compared between the main query and subqeury

Answer: ABC

NEW QUESTION 63
See the Exhibit and examine the structure and data in the INVOICE table: Exhibit:



```
INVOICE

Name       Null?      Type
--------- ---------- ---------------
INV_NO     NOT NULL   NUMBER(3)
INV_DATE              DATE
CUST_ID               VARCHAR2(4)
INV_AMT               NUMBER(8,2)


 INV_NO INV_DATE    CUST_ID INV_AMT
------ ----------- ------- -------
 1      01-APR-07    A10     1000
 2      01-OCT-07    B1R     2000
 3      01-FEB-07            3000
```

Which two SQL statements would execute successfully? (Choose two.)

A. SELECT MAX(inv_date),MIN(cust_id) FROM invoice;

B. SELECT AVG(inv_date-SYSDATE),AVG(inv_amt) FROM invoice;
C. SELECT MAX(AVG(SYSDATE-inv_date)) FROM invoice;
D. SELECT AVG(inv_date) FROM invoice;

Answer: AB

NEW QUESTION 66
You want to create an ORD_DETAIL table to store details for an order placed having the following business requirement:
1) The order ID will be unique and cannot have null values.
2) The order date cannot have null values and the default should be the current date.
3) The order amount should not be less than 50.
4) The order status will have values either shipped or not shipped.
5) The order payment mode should be cheque, credit card, or cash on delivery (COD).
Which is the valid DDL statement for creating the ORD_DETAIL table?

A. CREATE TABLE ord_details (ord_id NUMBER(2) CONSTRAINT ord_id_nn NOT NULL, ord_date DATE DEFAULT SYSDATE NOT NULL, ord_amount NUMBER(5, 2) CONSTRAINT ord_amount_min CHECK (ord_amount > 50), ord_status VARCHAR2(15) CONSTRAINT ord_status_chk CHECK (ord_status IN ('Shipped', 'Not Shipped')), ord_pay_mode VARCHAR2(15) CONSTRAINT ord_pay_chk CHECK (ord_pay_mode IN ('Cheque', 'Credit Card', 'Cash On Delivery')));
B. CREATE TABLE ord_details (ord_id NUMBER(2) CONSTRAINT ord_id_uk UNIQUE NOT NULL, ord_date DATE DEFAULT SYSDATE NOT NULL, ord_amount NUMBER(5, 2) CONSTRAINT ord_amount_min CHECK (ord_amount > 50), ord_status VARCHAR2(15) CONSTRAINT ord_status_chk CHECK (ord_status IN ('Shipped', 'Not Shipped')), ord_pay_mode VARCHAR2(15) CONSTRAINT ord_pay_chk CHECK (ord_pay_mode IN ('Cheque', 'Credit Card', 'Cash On Delivery')));
C. CREATE TABLE ord_details (ord_id NUMBER(2) CONSTRAINT ord_id_pk PRIMARY KEY, ord_date DATE DEFAULT SYSDATE NOT NULL, ord_amount NUMBER(5, 2) CONSTRAINT ord_amount_min CHECK (ord_amount >= 50), ord_status VARCHAR2(15) CONSTRAINT ord_status_chk CHECK (ord_status IN ('Shipped', 'Not Shipped')), ord_pay_mode VARCHAR2(15) CONSTRAINT ord_pay_chk CHECK (ord_pay_mode IN ('Cheque', 'Credit Card', 'Cash On Delivery')));
D. CREATE TABLE ord_details (ord_id NUMBER(2), ord_date DATE NOT NULL DEFAULT SYSDATE, ord_amount NUMBER(5, 2) CONSTRAINT ord_amount_min CHECK (ord_amount >= 50), ord_status VARCHAR2(15) CONSTRAINT ord_status_chk CHECK (ord_status IN ('Shipped', 'Not Shipped')), ord_pay_mode VARCHAR2(15) CONSTRAINT ord_pay_chk CHECK (ord_pay_mode IN ('Cheque', 'Credit Card', 'Cash On Delivery')));

Answer: C

NEW QUESTION 70
Examine the statement:
Create synonym emp for hr.employees;
What happens when you issue the statement?

A. An error is generate
B. You will have two identical tables in the HR schema with different name
C. You create a table called employees in the HR schema based on you EMP tabl
D. You create an alternative name for the employees table in the HR schema in your own schem

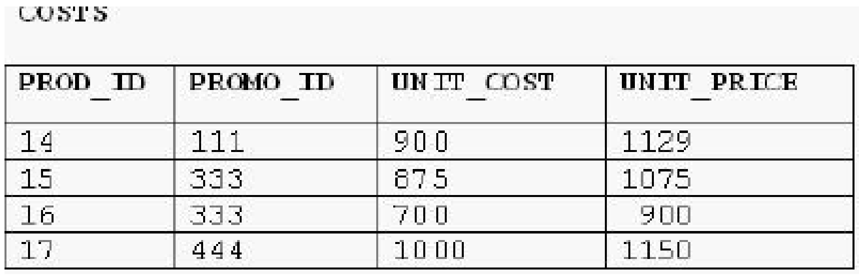
Answer: D

NEW QUESTION 75
Examine the structure and data of the CUSTJTRANS table:
CUSTJRANS
Name Null? Type
CUSTNO NOT NULL CHAR(2) TRANSDATE DATE TRANSAMT NUMBER(6.2) CUSTNO TRANSDATE TRANSAMT
11 01-JAN-07 1000
22 01-FEB-07 2000
33 01-MAR-07 3000
Dates are stored in the default date format dd-mon-rr in the CUSTJTRANS table. Which three SQL statements would execute successfully? (Choose three.)

A. SELECT transdate + '10' FROM custjrans;
B. SELECT * FROM custjrans WHERE transdate = '01-01-07':
C. SELECT transamt FROM custjrans WHERE custno > '11':
D. SELECT * FROM custjrans WHERE transdate='01-JANUARY-07':
E. SELECT custno - 'A' FROM custjrans WHERE transamt > 2000:

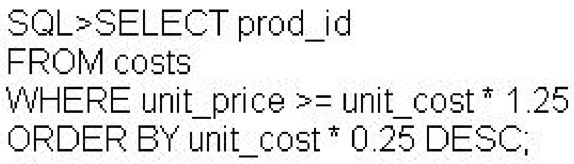
Answer: ACD

NEW QUESTION 79
View the Exhibit and examine the data in the COSTS table.


COSTS

PROD_ID	PROMO_ID	UNIT_COST	UNIT_PRICE
14	111	900	1129
15	333	875	1075
16	333	700	900
17	444	1000	1150

You need to generate a report that displays the IDs of all products in the COSTS table whose unit price is at least 25% more than the unit cost. The details should be displayed in the descending order of 25% of the unit cost. You issue the following query:



```
SQL>SELECT prod_id
FROM costs
WHERE unit_price >= unit_cost * 1.25
ORDER BY unit_cost * 0.25 DESC;
```

Which statement is true regarding the above query?

A. It executes and produces the required resul
B. It produces an error because an expression cannot be used in the ORDER BY claus
C. It produces an error because the DESC option cannot be used with an expression in the ORDER BY claus
D. It produces an error because the expression in the ORDER BY clause should also be specified in the SELECT claus

Answer: A

NEW QUESTION 81
What is true about sequences?

A. The start value of the sequence is always 1.
B. A sequence always increments by 1.
C. The minimum value of an ascending sequence defaults to 1.
D. The maximum value of descending sequence defaults to 1.

Answer: C

NEW QUESTION 84
View the Exhibit and examine the structure of the PROMOTIONS table.
You need to generate a report of all promos from the PROMOTIONS table based on the following conditions:
1.
The promo name should not begin with 'T' or 'N'.
2.
The promo should cost more than $20000.
3.
The promo should have ended after 1st January 2001.
Which WHERE clause would give the required result?

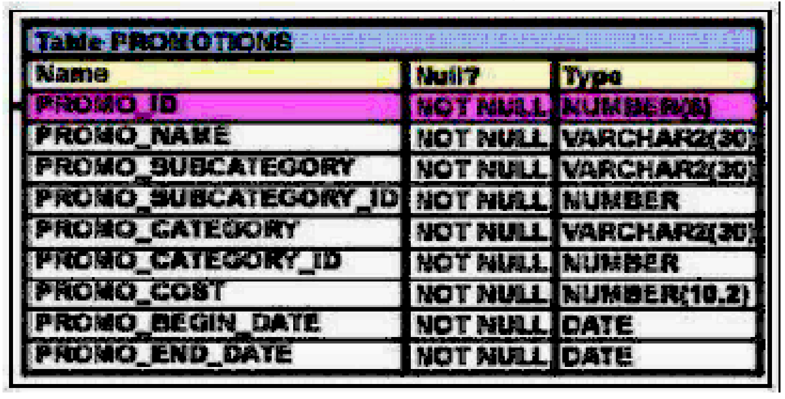


Table PROMOTIONS

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

A. WHERE promo_name NOT LIKE 'T%' OR promo_name NOT LIKE 'N%' AND promo_cost > 20000 AND promo_end_date > '1-JAN-01'
B. WHERE (promo_name NOT LIKE 'T%' AND promo_name NOT LIKE 'N%')OR promo_cost > 20000 OR promo_end_date > '1-JAN-01'
C. WHERE promo_name NOT LIKE 'T%' AND promo_name NOT LIKE 'N%' AND promo_cost > 20000 AND promo_end_date > '1-JAN-01'
D. WHERE (promo_name NOT LIKE '%T%' OR promo_name NOT LIKE '%N%') AND(promo_cost > 20000 AND promo_end_date > '1-JAN-01')

Answer: C

NEW QUESTION 85
Which three statements are true regarding sub queries? (Choose three.)

A. Multiple columns or expressions can be compared between the main query and sub query
B. Sub queries can contain GROUP BY and ORDER BY clauses
C. Only one column or expression can be compared between the main query and subqeury
D. Main query and sub query can get data from different tables
E. Main query and sub query must get data from the same tables
F. Sub queries can contain ORDER BY but not the GROUP BY clause

Answer: ABD

NEW QUESTION 90
You need to calculate the number of days from 1st January 2007 till date . Dates are stored in the default format of dd-mon-rr. Which two SQL statements would give the required output? (Choose two.)

A. SELECT SYSDATE - '01-JAN-2007' FROM DUAL:
B. SELECT SYSDATE - TOJDATE(X)1/JANUARY/2007") FROM DUAL:
C. SELECT SYSDATE - TOJDATE('01-JANUARY-2007') FROM DUAL:
D. SELECT TO_CHAR(SYSDAT
E. 'DD-MON-YYYY') - '01-JAN-2007' FROM DUAL:
F. SELECT TO_DATE(SYSDAT
G. *DD/MONTH/YYYY') - '01/JANUARY/2007' FROM DUAL:

Answer: BC

NEW QUESTION 95
See the Exhibit and examine the structure of the PROMOTIONS table: Exhibit:

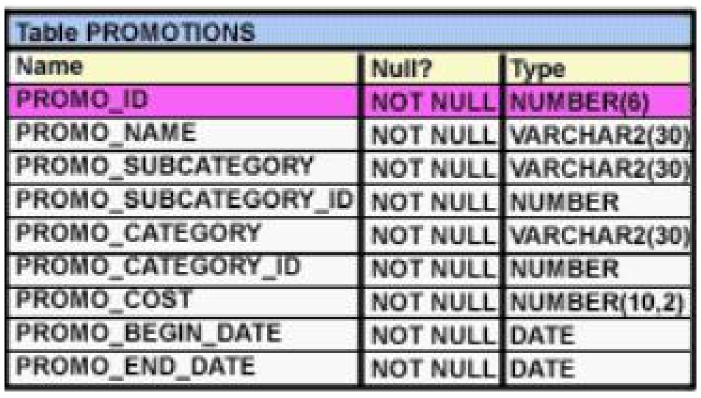
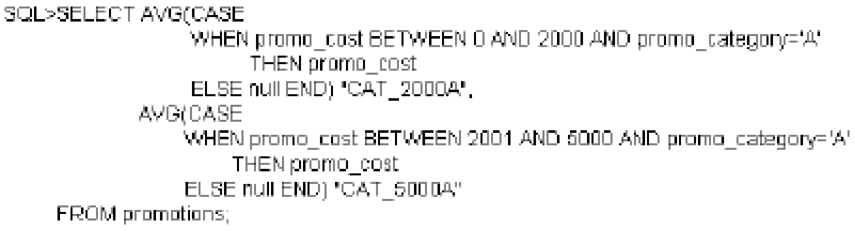


Table PROMOTIONS		
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(6)
PROMO_NAME	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY_ID	NOT NULL	NUMBER
PROMO_CATEGORY	NOT NULL	VARCHAR2(30)
PROMO_CATEGORY_ID	NOT NULL	NUMBER
PROMO_COST	NOT NULL	NUMBER(10,2)
PROMO_BEGIN_DATE	NOT NULL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

Using the PROMOTIONS table, you need to find out the average cost for all promos in the range $0-2000 and $2000-5000 in category A.
You issue the following SQL statements:
Exhibit:



```
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;
```

What would be the outcome?

A. It generates an error because multiple conditions cannot be specified for the WHEN clause
B. It executes successfully and gives the required result
C. It generates an error because CASE cannot be used with group functions
D. It generates an error because NULL cannot be specified as a return value

Answer: B

Explanation:
CASE Expression Facilitates conditional inquiries by doing the work of an IF-THEN-ELSE statement: CASE expr WHEN comparison_expr1 THEN return_expr1 [WHEN comparison_expr2 THEN return_expr2 WHEN comparison_exprn THEN return_exprn ELSE else_expr] END

NEW QUESTION 99
View the Exhibit and examine the structure of the PROMOTIONS table.
Using the PROMOTIONS table, you need to find out the names and cost of all the promos done on 'TV' and 'internet' that ended in the time interval 15th March '00 to 15th October '00.
Which two queries would give the required result? (Choose two.)

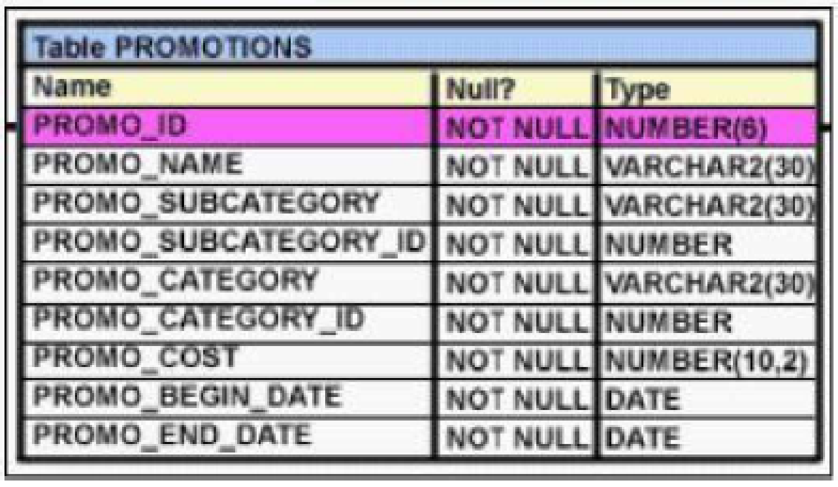


Table PROMOTIONS		
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(6)
PROMO_NAME	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY_ID	NOT NULL	NUMBER
PROMO_CATEGORY	NOT NULL	VARCHAR2(30)
PROMO_CATEGORY_ID	NOT NULL	NUMBER
PROMO_COST	NOT NULL	NUMBER(10,2)
PROMO_BEGIN_DATE	NOT NULL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

A. SELECT promo_name, promo_cost FROM promotions WHERE promo_category IN ('TV', 'internet') AND promo_end_date BETWEEN '15-MAR-00' AND '15-OCT-00';
B. SELECT promo_name, promo_cost FROM promotions WHERE promo_category = 'TV' OR promo_category ='internet' AND promo_end_date >='15-MAR-00' OR promo_end_date <='15-OCT-00';
C. SELECT promo_name, promo_cost FROM promotions WHERE (promo_category BETWEEN 'TV' AND 'internet') AND (promo_end_date IN ('15-MAR-00','15-OCT-00'));
D. SELECT promo_name, promo_cost FROM promotions WHERE (promo_category = 'TV' OR promo_category ='internet') AND (promo_end_date >='15-MAR-00' AND promo_end_date <='15-OCT-00');

Answer: AD

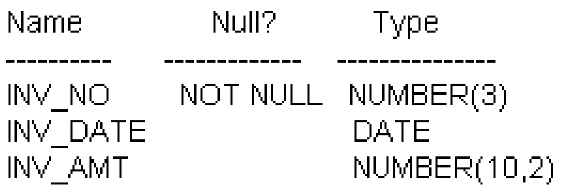
NEW QUESTION 103
Using the CUSTOMERS table, you need to generate a report that shows 50% of each credit amount in each income level. The report should NOT show any repeated credit amounts in each income level. Which query would give the required result?

A. SELECT cust_income_level, DISTINCT cust_credit_limit * 0.50 AS "50% Credit Limit" FROM customers;
B. SELECT DISTINCT cust_income_level, DISTINCT cust_credit_limit * 0.50 AS "50% Credit Limit" FROM customers;
C. SELECT DISTINCT cust_income_level ' ' cust_credit_limit * 0.50 AS "50% Credit Limit" FROM customers;
D. SELECT cust_income_level ' ' cust_credit_limit * 0.50 AS "50% Credit Limit" FROM customers;

Answer: C

Explanation: Duplicate Rows Unless you indicate otherwise, SQL displays the results of a query without eliminating the duplicate rows. To eliminate duplicate rows in the result, include the DISTINCT keyword in the SELECT clause immediately after the SELECT keyword. You can specify multiple columns after the DISTINCT qualifier. The DISTINCT qualifier affects all the selected columns, and the result is every distinct combination of the columns.

NEW QUESTION 105
Examine the structure of the INVOICE table: Exhibit:



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

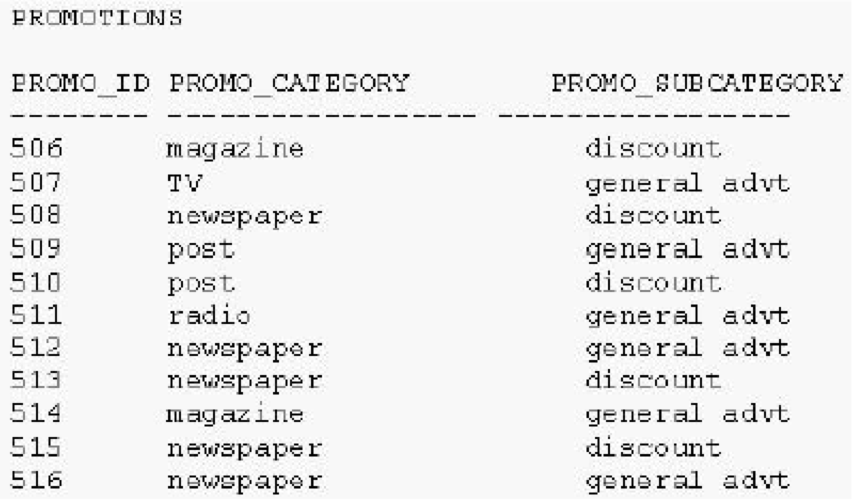
Which two SQL statements would execute successfully? (Choose two.)

A. SELECT inv_no,NVL2(inv_date,'Pending','Incomplete') FROM invoice;
B. SELECT inv_no,NVL2(inv_amt,inv_date,'Not Available') FROM invoice;
C. SELECT inv_no,NVL2(inv_date,sysdate-inv_date,sysdate) FROM invoice;
D. SELECT inv_no,NVL2(inv_amt,inv_amt*.25,'Not Available') FROM invoice;

Answer: AC

Explanation:
The NVL2 Function
The NVL2 function provides an enhancement to NVL but serves a very similar purpose. It evaluates whether a column or expression of any data type is null or not. 5-6 The NVL function\ If the first term is not null, the second parameter is returned, else the third parameter is returned. Recall that the NVL function is different since it returns the original term if it is not null. The NVL2 function takes three mandatory parameters. Its syntax is NVL2(original, ifnotnull, ifnull), where original represents the term being tested. Ifnotnull is returned if original is not null, and ifnull is returned if original is null. The data types of the ifnotnull and ifnull parameters must be compatible, and they cannot be of type LONG. They must either be of the same type, or it must be possible to convert ifnull to the type of the ifnotnull parameter. The data type returned by the NVL2 function is the same as that of the ifnotnull parameter.

NEW QUESTION 109
View the Exhibit and examine the data in the PROMOTIONS table.



```
PROMOTIONS

PROMO_ID PROMO_CATEGORY        PROMO_SUBCATEGORY
-------- ------------------ -----------------
506      magazine                discount
507      TV                      general advt
508      newspaper               discount
509      post                    general advt
510      post                    discount
511      radio                   general advt
512      newspaper               general advt
513      newspaper               discount
514      magazine                general advt
515      newspaper               discount
516      newspaper               general advt
```

You need to display all promo categories that do not have 'discount' in their subcategory.
Which two SQL statements give the required result? (Choose two.)

A. SELECT promo_category FROM promotions MINUS SELECT promo_category FROM promotions WHERE promo_subcategory = 'discount';
B. SELECT promo_category FROM promotions INTERSECT SELECT promo_category FROM promotions WHERE promo_subcategory = 'discount';
C. SELECT promo_category FROM promotions MINUS SELECT promo_category FROM promotions WHERE promo_subcategory <> 'discount';
D. SELECT promo_category FROM promotions INTERSECT SELECT promo_category FROM promotions WHERE promo_subcategory <> 'discount';

Answer: AD

NEW QUESTION 111
You need to display the first names of all customers from the CUSTOMERS table that contain the character 'e' and have the character 'a' in the second last position.
Which query would give the required output?

A. SELECT cust_first_name FROM customers WHERE INSTR(cust_first_name, 'e')<>0 AND SUBSTR(cust_first_name, -2, 1)='a';
B. SELECT cust_first_name FROM customers WHERE INSTR(cust_first_name, 'e')<>'' AND SUBSTR(cust_first_name, -2, 1)='a';
C. SELECT cust_first_name FROM customers WHERE INSTR(cust_first_name, 'e')IS NOT NULL AND SUBSTR(cust_first_name, 1,-2)='a';
D. SELECT cust_first_name FROM customers WHERE INSTR(cust_first_name, 'e')<>0 AND SUBSTR(cust_first_name, LENGTH(cust_first_name),-2)='a';

Answer: A

Explanation:
The SUBSTR(string, start position, number of characters) function accepts three parameters and returns a string consisting of the number of characters extracted from the source string, beginning at the specified start position:
substr('http://www.domain.com',12,6) = domain
The position at which the first character of the returned string begins.
When position is 0 (zero), then it is treated as 1.
When position is positive, then the function counts from the beginning of string to find the first character.
When position is negative, then the function counts backward from the end of string.

substring_length
The length of the returned string. SUBSTR calculates lengths using characters as defined by the input character set. SUBSTRB uses bytes instead of characters. SUBSTRC uses Unicode complete characters.
SUBSTR2 uses UCS2 code points. SUBSTR4 uses UCS4 code points.
When you do not specify a value for this argument, then the function
The INSTR(source string, search item, [start position],[nth occurrence of search item]) function returns a number that represents the position in the source string, beginning from the given start position, where the nth occurrence of the search item begins:
instr('http://www.domain.com','.',1,2) = 18

NEW QUESTION 114
Which two statements are true regarding sub queries? (Choose two.)

A. A sub query can retrieve zero or more row
B. Only two sub queries can be placed at one leve
C. A sub query can be used only in SQL query statement
D. A sub query can appeal* on either side of a comparison operato
E. There is no limit on the number of sub query levels in the WHERE clause of a SELECT statemen

Answer: AD

NEW QUESTION 118
View the Exhibits and examine the structures of the PRODUCTS SALES and CUSTOMERS tables.

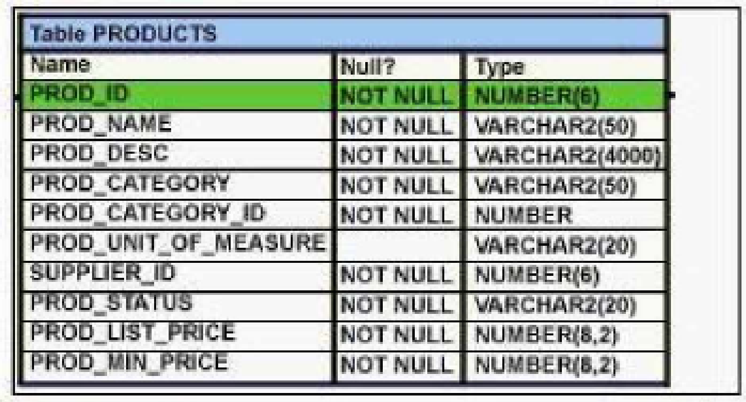


Table PRODUCTS

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)

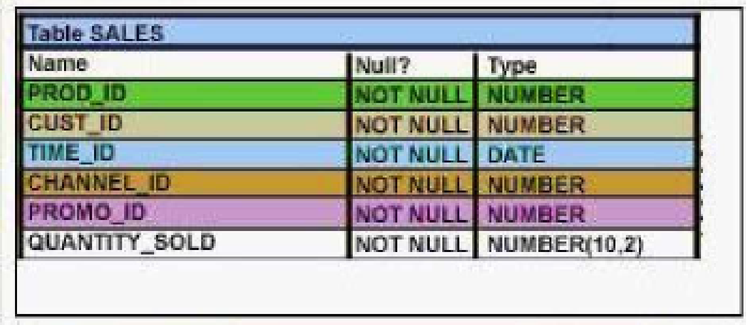


Table SALES

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)

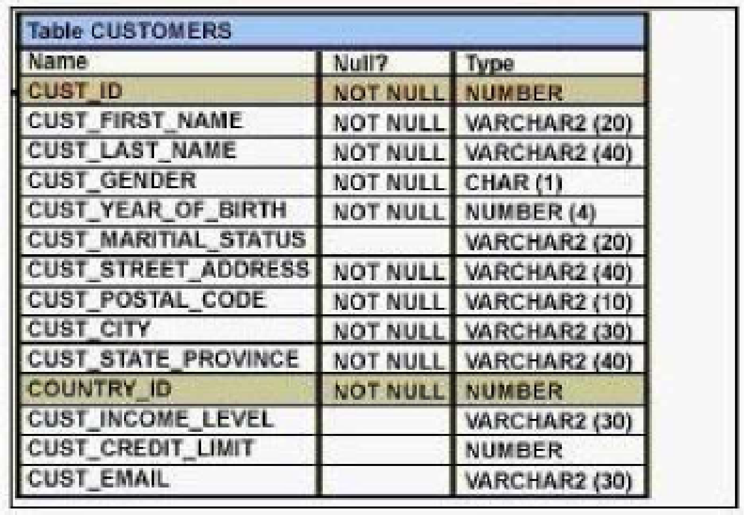


Table CUSTOMERS

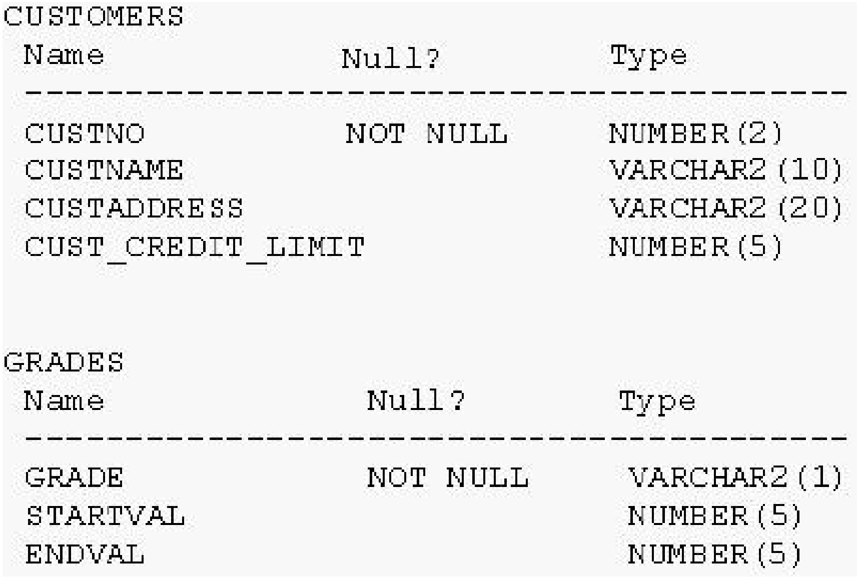
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2 (20)
CUST_LAST_NAME	NOT NULL	VARCHAR2 (40)
CUST_GENDER	NOT NULL	CHAR (1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER (4)
CUST_MARITIAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
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)

You need to generate a report that gives details of the customer's last name, name of the product, and the quantity sold for all customers in Tokyo'. Which two queries give the required result? (Choose two.)

A. SELECT c.cust_last_name,p.prod_name, s.quantity_sold FROM sales s JOIN products p USING(prod_id) JOIN customers c USING(cust_id) WHERE c.cust_city='Tokyo';
B. SELECT c.cust_last_name, p.prod_name, s.quantity_sold FROM products p JOIN sales s JOIN customers c ON(p.prod_id=s.prod_id) ON(s.cust_id=c.cust_id) WHERE c.cust_city='Tokyo';
C. SELECT c.cust_last_name, p.prod_name, s.quantity_sold FROM products p JOIN sales s ON(p.prod_id=s.prod_id) JOIN customers c ON(s.cust_id=c.cust_id) AND c.cust_city='Tokyo';
D. SELECT c.cust_id,c.cust_last_name,p.prod_id, p.prod_name, s.quantity_sold FROM products p JOIN sales s USING(prod_id) JOIN customers c USING(cust_id) WHERE c.cust_city='Tokyo';

Answer: AC

NEW QUESTION 123
See the exhibit and examine the structure of the CUSTOMERS and GRADES tables:



```
CUSTOMERS
 Name                Null?          Type
 -----------------------------------------
 CUSTNO              NOT NULL       NUMBER(2)
 CUSTNAME                           VARCHAR2(10)
 CUSTADDRESS                        VARCHAR2(20)
 CUST_CREDIT_LIMIT                  NUMBER(5)


GRADES
 Name                 Null?         Type
 -----------------------------------------
 GRADE                NOT NULL       VARCHAR2(1)
 STARTVAL                            NUMBER(5)
 ENDVAL                              NUMBER(5)
```

You need to display names and grades of customers who have the highest credit limit.
Which two SQL statements would accomplish the task? (Choose two.)

A. SELECT custname, grade FROM customers, grades WHERE (SELECT MAX(cust_credit_limit) FROM customers) BETWEEN startval and endval;
B. SELECT custname, grade FROM customers, grades WHERE (SELECT MAX(cust_credit_limit) FROM customers) BETWEEN startval and endval AND cust_credit_limit BETWEEN startval AND endval;
C. SELECT custname, grade FROM customers, grades WHERE cust_credit_limit = (SELECT MAX(cust_credit_limit) FROM customers) AND cust_credit_limit BETWEEN startval AND endval;
D. SELECT custname, grade FROM customers , grades WHERE cust_credit_limit IN (SELECT MAX(cust_credit_limit) FROM customers) AND MAX(cust_credit_limit) BETWEEN startval AND endval;

Answer: BC

NEW QUESTION 128
View the Exhibit and examine the structure of the PROMOTIONS table. Which SQL statements are valid? (Choose all that apply.)

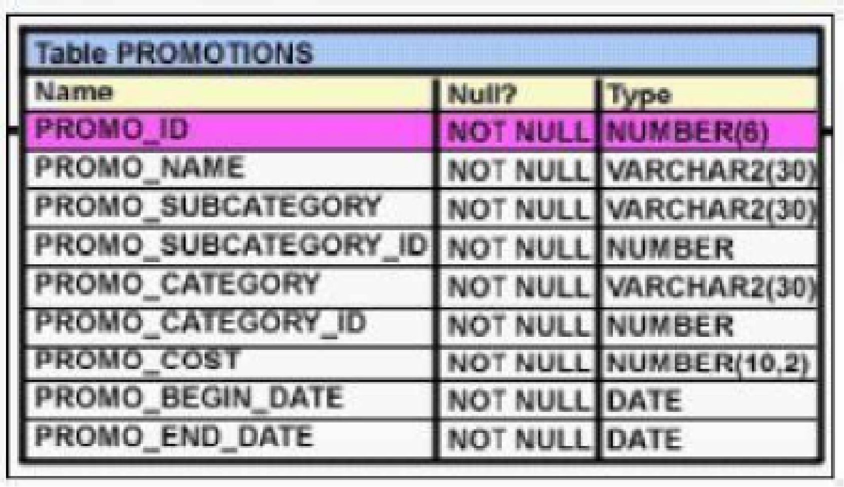


Table PROMOTIONS

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

A. SELECT promo_i
B. DECODE(NVL(promo_cost.O).promo_cost * 0.25. 100) "Discount" FROM promotions;
C. SELECT promo i
D. DECODE(promo_cos
E. 10000. DECODE(promo_categor
F. 'Gl\ promo_cost * 25. NULL). NULL) "Catcost" FROM promotions;
G. SELECT promo_i
H. DECODE(NULLIF(promo_cos
I. 10000). NUL
J. promo_cost*.25, *N/A') "Catcost" FROM promotions;
K. SELECT promo_i
L. DECODE(promo_cos
M. >10000. 'High'. <10000. 'Low') "Range"FROM promotions;

Answer: AB

Explanation:
Note: there are some syntax issues in this question.

NEW QUESTION 133
You need to display the date 11-Oct-2007 in words as ‘Eleventh of October, Two Thousand Seven’. Which SQL statement would give the required result?

A. SELECT TO_CHAR('11-oct-2007', 'fmDdspth "of" Month, Year') FROM DUAL;
B. SELECT TO_CHAR(TO_DATE('11-oct-2007'), 'fmDdspth of month, year') FROM DUAL;
C. SELECT TO_CHAR(TO_DATE('11-oct-2007'), 'fmDdthsp "of" Month, Year') FROM DUAL;
D. SELECT TO_DATE(TO_CHAR('11-oct-2007','fmDdspth "of" Month, Year')) FROM DUAL;

Answer: C

Explanation:
Using the TO_CHAR Function with Dates TO_CHAR converts a datetime data type to a value of VARCHAR2 data type in the format specified by the

format_model. A format model is a character literal that describes the format of datetime stored in a character string. For example, the datetime format model for the string '11-Nov-1999' is 'DD-Mon-YYYY'. You can use the TO_CHAR function to convert a date from its default format to the one that you specify. Guidelines
.
The format model must be enclosed with single quotation marks and is case-sensitive.
.
The format model can include any valid date format element. But be sure to separate the date value from the format model with a comma.
.
The names of days and months in the output are automatically padded with blanks.
.
To remove padded blanks or to suppress leading zeros, use the fill mode fm element.
Elements of the Date Format Model
DY Three-letter abbreviation of the day of the week
DAY Full name of the day of the week
DD Numeric day of the month
MM Two-digit value for the month
MON Three-letter abbreviation of the month
MONTH Full name of the month
YYYY Full year in numbers
YEAR Year spelled out (in English)

NEW QUESTION 135
Which statement is true regarding synonyms?

A. Synonyms can be created only for a table
B. Synonyms are used to reference only those tables that are owned by another user
C. The DROP SYNONYM statement removes the synonym and the table on which the synonym has been created becomes invalid
D. A public synonym and a private synonym can exist with the same name for the same table

Answer: D

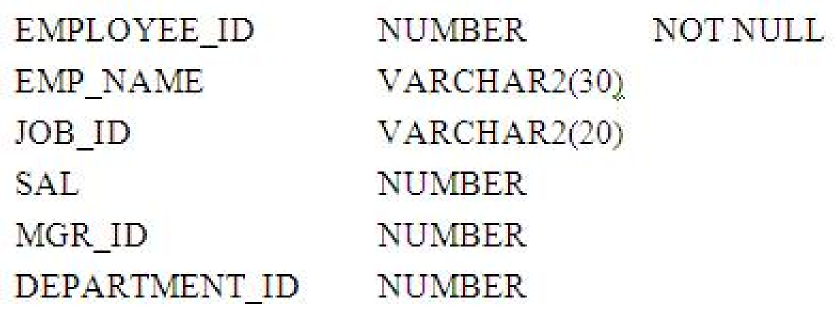
NEW QUESTION 140
A SELECT statement can be used to perform these three functions:
Choose rows from a table.
Choose columns from a table
Bring together data that is stored in different tables by creating a link between
them.
Which set of keywords describes these capabilities?

A. difference, projection, join
B. selection, projection, join
C. selection, intersection, join
D. intersection, projection, join
E. difference, projection, product

Answer: B

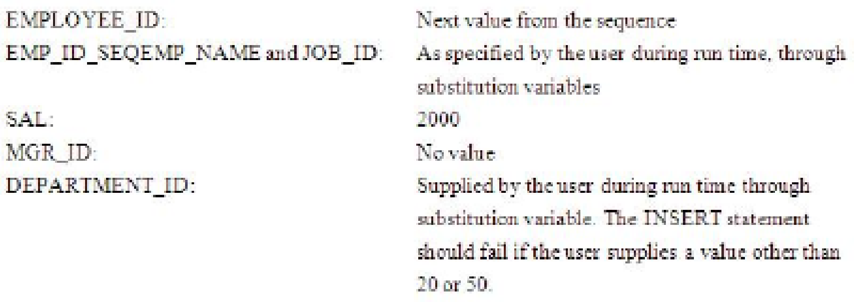
Explanation: Explanation: choose rows from a table is SELECTION,
Choose column from a table is PROJECTION
Bring together data in different table by creating a link between them is JOIN.
Incorrect Answer:
Aanswer should have SELECTION, PROJECTION and JOIN.
Canswer should have SELECTION, PROJECTION and JOIN.
Danswer should have SELECTION, PROJECTION and JOIN.
Eanswer should have SELECTION, PROJECTION and JOIN.
Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 1-6

NEW QUESTION 141
Examine the structure of the EMPLOYEES table:



EMPLOYEE_ID	NUMBER	NOT NULL
EMP_NAME	VARCHAR2(30)	
JOB_ID	VARCHAR2(20)	
SAL	NUMBER	
MGR_ID	NUMBER	
DEPARTMENT_ID	NUMBER	

You want to create a SQL script file that contains an INSERT statement. When the script is run, the INSERT statement should insert a row with the specified values into the EMPLOYEES table. The INSERT statement should pass values to the table columns as specified below:



EMPLOYEE_ID:	Next value from the sequence
EMP_ID_SEQEMP_NAME and JOB_ID:	As specified by the user during run time, through substitution variables
SAL:	2000
MGR_ID:	No value
DEPARTMENT_ID:	Supplied by the user during run time through substitution variable. The INSERT statement should fail if the user supplies a value other than 20 or 50.

Which INSERT statement meets the above requirements?

A. INSERT INTO employees VALUES (emp_id_seq.NEXTVAL, '&ename', '&jobid', 2000, NULL, &did);
B. INSERT INTO employees VALUES (emp_id_seq.NEXTVAL, '&ename', '&jobid', 2000, NULL, &did IN (20,50));
C. INSERT INTO (SELECT * FROM employees WHERE department_id IN (20,50)) VALUES (emp_id_seq.NEXTVAL, '&ename', '&jobid', 2000, NULL, &did);
D. INSERT INTO (SELECT * FROM employees WHERE department_id IN (20,50) WITH CHECK OPTION) VALUES (emp_id_seq.NEXTVAL, '&ename', '&jobid', 2000, NULL, &did);
E. INSERT INTO (SELECT * FROM employees WHERE (department_id = 20 AND department_id = 50) WITH CHECK OPTION) VALUES (emp_id_seq.NEXTVAL, '&ename', '&jobid', 2000, NULL, &did);

Answer: D

NEW QUESTION 144
View the Exhibit and examine the structure of the CUSTOMERS table .Which statement would display the highest credit limit available in each income level in each city in the CUSTOMERS table?

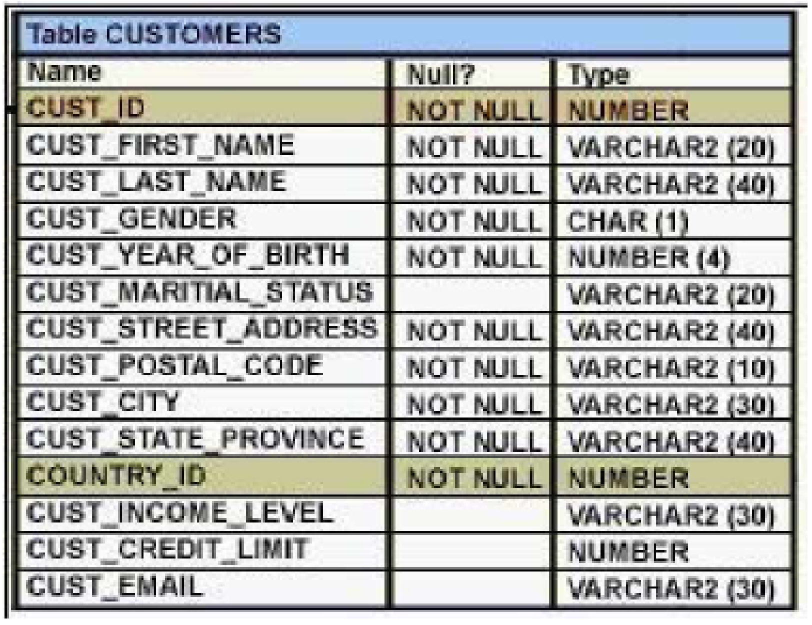


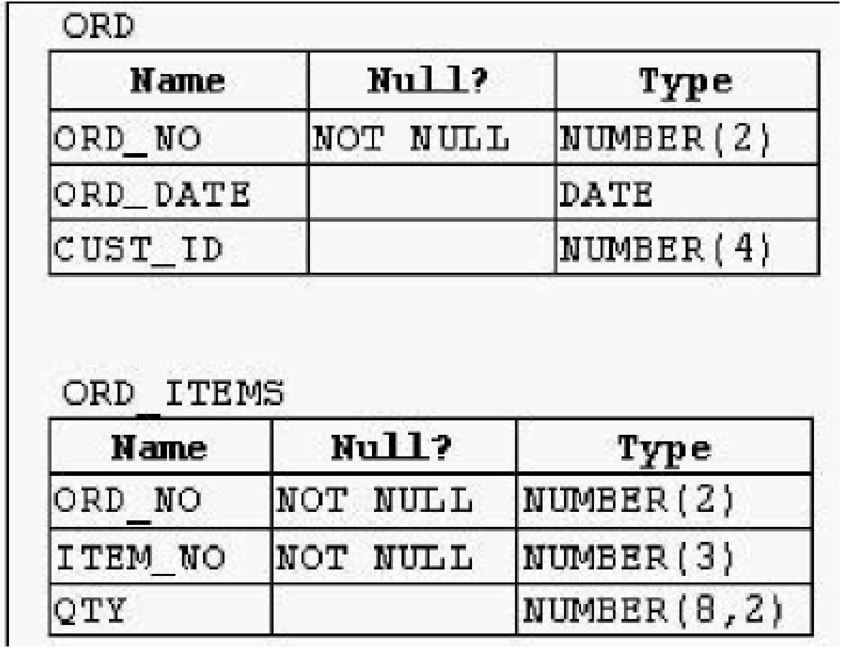
Table CUSTOMERS

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_MARITIAL_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 cust_city, cust_income_level, MAX(cust_credit_limit) FROM customers GROUP BY cust_city, cust_income_level, cust_credit_limit;
B. SELECT cust_city, cust_income_level, MAX(cust_credit_limit) FROM customers GROUP BY cust_city, cust_income_level;
C. SELECT cust_city, cust_income_level, MAX(cust_credit_limit) FROM customers GROUP BY cust_credit_limit, cust_income_level, cust_city ;
D. SELECT cust_city, cust_income_level, MAX(cust_credit_limit) FROM customers GROUP BY cust_city, cust_income_level, MAX(cust_credit_limit);

Answer: B

NEW QUESTION 146
View the Exhibit and examine the structure of ORD and ORD_ITEMS tables.
The ORD_NO column is PRIMARY KEY in the ORD table and the ORD_NO and ITEM_NO columns are composite PRIMARY KEY in the ORD_ITEMS table.
Which two CREATE INDEX statements are valid? (Choose two.)



ORD

Name	Null?	Type
ORD_NO	NOT NULL	NUMBER(2)
ORD_DATE		DATE
CUST_ID		NUMBER(4)

ORD_ITEMS

Name	Null?	Type
ORD_NO	NOT NULL	NUMBER(2)
ITEM_NO	NOT NULL	NUMBER(3)
QTY		NUMBER(8,2)

A. CREATE INDEX ord_idx1 ON ord(ord_no);
B. CREATE INDEX ord_idx2 ON ord_items(ord_no);
C. CREATE INDEX ord_idx3 ON ord_items(item_no);
D. CREATE INDEX ord_idx4 ON ord,ord_items(ord_no, ord_date,qty);

Answer: BC

Explanation: How Are Indexes Created?
You can create two types of indexes.
Unique index: The Oracle server automatically creates this index when you define a column in a table to have a PRIMARY KEY or a UNIQUE constraint. The name of the index is the name that is given to the constraint.
Nonunique index: This is an index that a user can create. For example, you can create the FOREIGN KEY column index for a join in a query to improve the speed of retrieval.
Note: You can manually create a unique index, but it is recommended that you create a unique constraint, which implicitly creates a unique index.

NEW QUESTION 150
View the Exhibit and examine the description for the CUSTOMERS table.

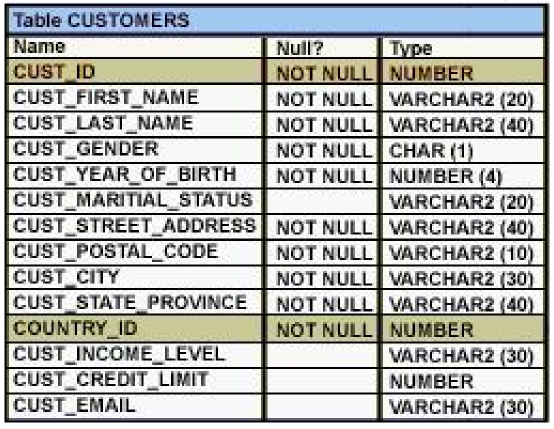


Table CUSTOMERS

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_MARITIAL_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)

You want to update the CUST_INCOME_LEVEL and CUST_CREDIT_LIMIT columns for the customer with the CUST_ID 2360. You want the value for the CUST_INCOME_LEVEL to have the same value as that of the customer with the CUST_ID 2560 and the CUST_CREDIT_LIMIT to have the same value as that of the customer with CUST_ID 2566.
Which UPDATE statement will accomplish the task?

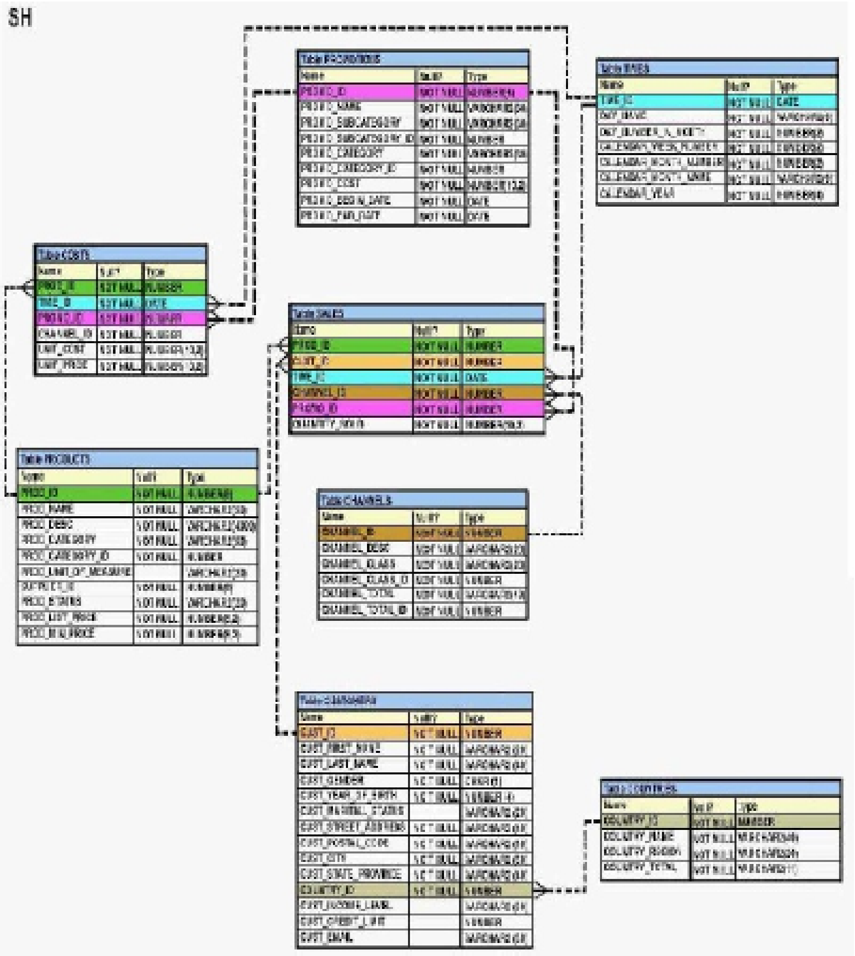
A. UPDATE customers SET cust_income_level = (SELECT cust_income_level FROM customers WHERE cust_id = 2560), cust_credit_limit = (SELECT cust_credit_limit FROM customers WHERE cust_id = 2566) WHERE cust_id=2360;
B. UPDATE customers SET (cust_income_level,cust_credit_limit) = (SELECT cust_income_level, cust_credit_limit FROM customers WHERE cust_id=2560 OR cust_id=2566) WHERE cust_id=2360;
C. UPDATE customers SET (cust_income_level,cust_credit_limit) = (SELECT cust_income_level, cust_credit_limit FROM customers WHERE cust_id IN(2560, 2566) WHERE cust_id=2360;
D. UPDATE customers SET (cust_income_level,cust_credit_limit) = (SELECT cust_income_level, cust_credit_limit FROM customers WHERE cust_id=2560 AND cust_id=2566) WHERE cust_id=2360;

Answer: A

Explanation:
Updating Two Columns with a Subquery
You can update multiple columns in the SET clause of an UPDATE statement by writing multiple subqueries. The syntax is as follows:

```
UPDATE table
SET column =
(SELECT column
FROM table
WHERE condition)
[ ,
column =
(SELECT column
FROM table
WHERE condition)]
[WHERE condition ] ;
```

NEW QUESTION 151
View the Exhibit and examine the description for the PRODUCTS and SALES table.



PROD_ID is a primary key in the PRODUCTS table and foreign key in the SALES table. You want to remove all the rows from the PRODUCTS table for which no sale was done for the last three years. Which is the valid DELETE statement?

A. DELETE FROM products WHERE prod_id = (SELECT prod_id FROM sales WHERE time_id - 3*365 = SYSDATE);
B. DELETE FROM products WHERE prod_id = (SELECT prod_id FROM sales WHERE SYSDATE >= time_id - 3*365);
C. DELETE FROM products WHERE prod_id IN (SELECT prod_id FROM sales WHERE SYSDATE - 3*365 >= time_id);
D. DELETE FROM products WHERE prod_id IN (SELECT prod_id FROM sales WHERE time_id >= SYSDATE - 3*365);

Answer: C

NEW QUESTION 154
View the Exhibit and examine the structure of the PROMOTIONS table.

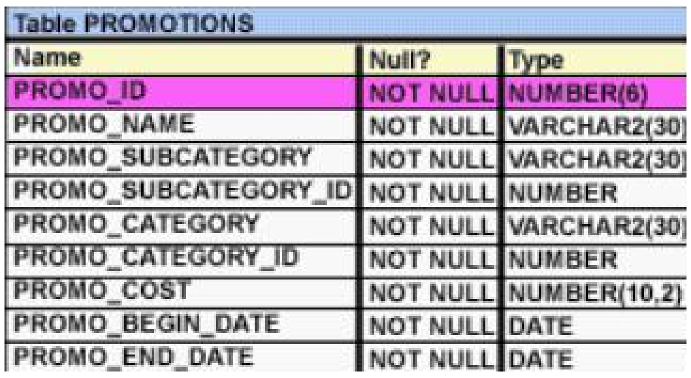


Table PROMOTIONS

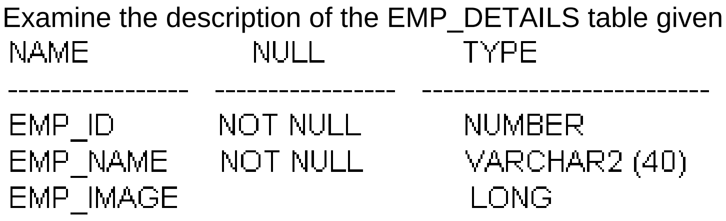
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(6)
PROMO_NAME	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY_ID	NOT NULL	NUMBER
PROMO_CATEGORY	NOT NULL	VARCHAR2(30)
PROMO_CATEGORY_ID	NOT NULL	NUMBER
PROMO_COST	NOT NULL	NUMBER(10,2)
PROMO_BEGIN_DATE	NOT NULL	DATE
PROMO_END_DATE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

You have to generate a report that displays the promo name and start date for all promos that started after the last promo in the 'INTERNET' category.
Which query would give you the required output?

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

Answer: C

NEW QUESTION 158
Examine the description of the EMP_DETAILS table given below: Exhibit:



```
NAME              NULL             TYPE
-----------------  -----------------  ---------------------------
EMP_ID           NOT NULL          NUMBER
EMP_NAME         NOT NULL          VARCHAR2 (40)
EMP_IMAGE                           LONG
```

Which two statements are true regarding SQL statements that can be executed on the EMP_DETAIL table? (Choose two.)

A. An EMP_IMAGE column can be included in the GROUP BY clause
B. You cannot add a new column to the table with LONG as the data type
C. An EMP_IMAGE column cannot be included in the ORDER BY clause
D. You can alter the table to include the NOT NULL constraint on the EMP_IMAGE column

Answer: BC

Explanation:
LONG Character data in the database character set, up to 2GB. All the functionality of LONG (and more) is provided by CLOB; LONGs should not be used in a modern database, and if your database has any columns of this type they should be converted to CLOB.
There can only be one LONG column in a table.
Guidelines
A LONG column is not copied when a table is created using a subquery.
A LONG column cannot be included in a GROUP BY or an ORDER BY clause.
Only one LONG column can be used per table.
No constraints can be defined on a LONG column.
You might want to use a CLOB column rather than a LONG column.

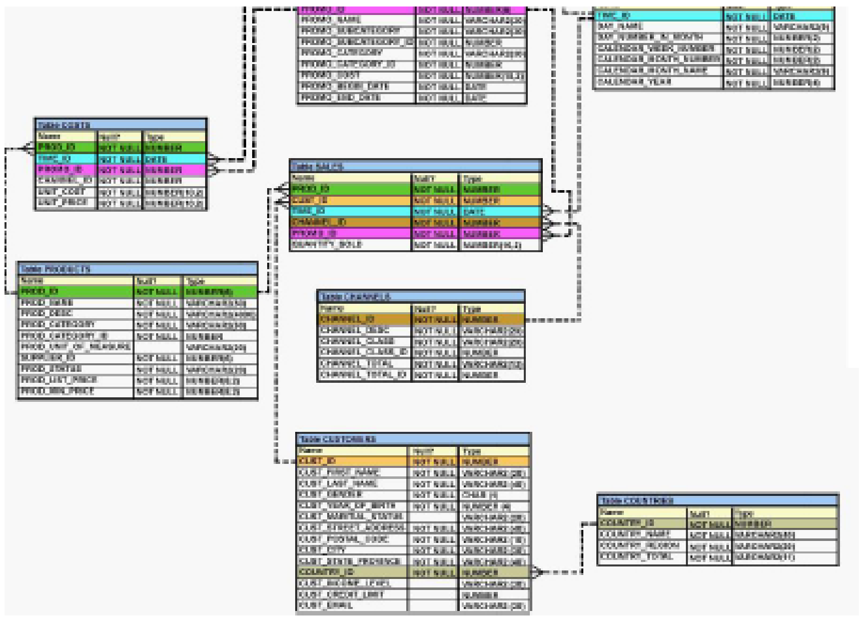
NEW QUESTION 163
Which two statements are true regarding the USING clause in table joins?(Choose two.)

A. It can be used to join a maximum of three table
B. It can be used to restrict the number of columns used in a NATURAL joi
C. It can be used to access data from tables through equijoins as well as nonequijoin
D. It can be used to join tables that have columns with the same name and compatible data type

Answer: BD

Explanation:
NATURAL JOIN operation A NATURAL JOIN is a JOIN operation that creates an implicit join clause for you based on the common columns in the two tables being joined. Common columns are columns that have the same name in both tables. If the SELECT statement in which the NATURAL JOIN operation appears has an asterisk (*) in the select list, the asterisk will be expanded to the following list of columns (in this order):
All the common columns
Every column in the first (left) table that is not a common column
Every column in the second (right) table that is not a common column
An asterisk qualified by a table name (for example, COUNTRIES.*) will be expanded to
every column of that table that is not a common column.
If a common column is referenced without being qualified by a table name, the column
reference points to the column in the first (left) table if the join is an INNER JOIN or a LEFT
OUTER JOIN. If it is a RIGHT OUTER JOIN, unqualified references to a common column
point to the column in the second (right) table.
Syntax
TableExpression NATURAL [{ LEFT | RIGHT } [OUTER] | INNER] JOIN {
TableViewOrFunctionExpression |
(TableExpression) }
Examples
If the tables COUNTRIES and CITIES have two common columns named COUNTRY and
COUNTRY_ISO_CODE, the following two SELECT statements are equivalent:
SELECT * FROM COUNTRIES NATURAL JOIN CITIES
SELECT * FROM COUNTRIES JOIN CITIES
USING (COUNTRY, COUNTRY_ISO_CODE)

NEW QUESTION 166
See the Exhibit and Examine the structure of SALES and PROMOTIONS tables: Exhibit:



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

TIME_ID	NOT NULL	DATE
DAY_NAME	NOT NULL	VARCHAR2(9)
DAY_NUMBER_IN_MONTH	NOT NULL	NUMBER(2)
CALENDAR_WEEK_NUMBER	NOT NULL	NUMBER(2)
CALENDAR_MONTH_NUMBER	NOT NULL	NUMBER(2)
CALENDAR_MONTH_NAME	NOT NULL	VARCHAR2(9)
CALENDAR_YEAR	NOT NULL	NUMBER(4)

Table COSTS

Name	Null?	Type
PROD_ID	NOT NULL	NUMBER
TIME_ID	NOT NULL	DATE
PROMO_ID	NOT NULL	NUMBER
CHANNEL_ID	NOT NULL	NUMBER
UNIT_COST	NOT NULL	NUMBER(10,2)
UNIT_PRICE	NOT NULL	NUMBER(10,2)

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)

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)

Table CHANNELS

Name	Null?	Type
CHANNEL_ID	NOT NULL	NUMBER
CHANNEL_DESC	NOT NULL	VARCHAR2(20)
CHANNEL_CLASS	NOT NULL	VARCHAR2(20)
CHANNEL_CLASS_ID	NOT NULL	NUMBER
CHANNEL_TOTAL	NOT NULL	VARCHAR2(13)
CHANNEL_TOTAL_ID	NOT NULL	NUMBER

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)

Table COUNTRIES

Name	Null?	Type
COUNTRY_ID	NOT NULL	NUMBER
COUNTRY_NAME	NOT NULL	VARCHAR2(40)
COUNTRY_REGION	NOT NULL	VARCHAR2(20)
COUNTRY_TOTAL	NOT NULL	VARCHAR2(11)

You want to delete rows from the SALES table, where the PROMO_NAME column in the PROMOTIONS table has either blowout sale or everyday low price as values.
Which DELETE statements are valid? (Choose all that apply.)

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

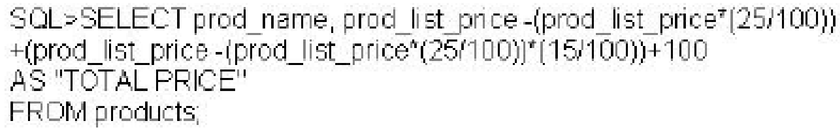
Answer: BCD

NEW QUESTION 170
Which three tasks can be performed using SQL functions built into Oracle Database? (Choose three.)

A. Combining more than two columns or expressions into a single column in the output
B. Displaying a date in a nondefault format
C. Substituting a character string in a text expression with a specified string
D. Finding the number of characters in an expression

Answer: BCD

NEW QUESTION 174
View the Exhibit and examine the structure of the PRODUCTS table.
All products have a list price.
You issue the following command to display the total price of each product after a discount of 25% and a tax of 15% are applied on it. Freight charges of S100 have to be applied to all the products.



```
SQL>SELECT prod_name, prod_list_price -(prod_list_price*(25/100))
+(prod_list_price -(prod_list_price*(25/100))*(15/100))+100
AS "TOTAL PRICE"
FROM products;
```

What would be the outcome if all the parentheses are removed from the above statement?

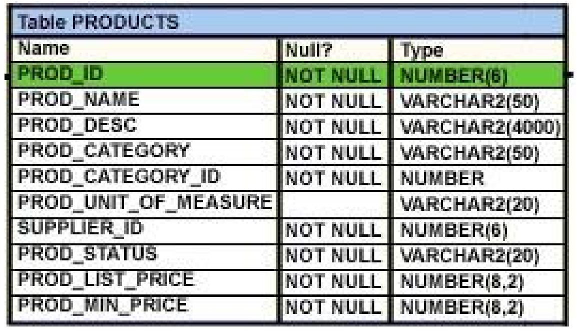
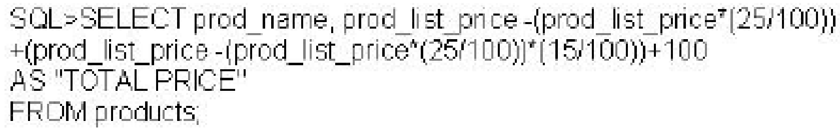


Table PRODUCTS		
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER(6)
PROD_NAME	NOT NULL	VARCHAR2(50)
PROD_DESC	NOT NULL	VARCHAR2(4000)
PROD_CATEGORY	NOT NULL	VARCHAR2(50)
PROD_CATEGORY_ID	NOT NULL	NUMBER
PROD_UNIT_OF_MEASURE		VARCHAR2(20)
SUPPLIER_ID	NOT NULL	NUMBER(6)
PROD_STATUS	NOT NULL	VARCHAR2(20)
PROD_LIST_PRICE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

A. It produces a syntax erro
B. The result remains unchange
C. The total price value would be lower than the correct valu
D. The total price value would be higher than the correct valu

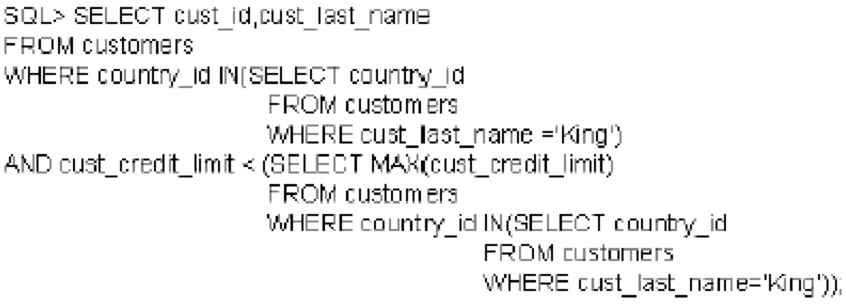
Answer: B

NEW QUESTION 178
View the Exhibit and examine the structure of the CUSTOMERS table. Exhibit:



```
SQL>SELECT prod_name, prod_list_price -(prod_list_price*(25/100))
+(prod_list_price -(prod_list_price*(25/100))*(15/100))+100
AS "TOTAL PRICE"
FROM products;
```

you issue the following SQL statement on the CUSTOMERS table to display the customers who are in the same country as customers with the last name 'king' and whose credit limit is less than the maximum credit limit in countries that have customers with the last name 'king'.



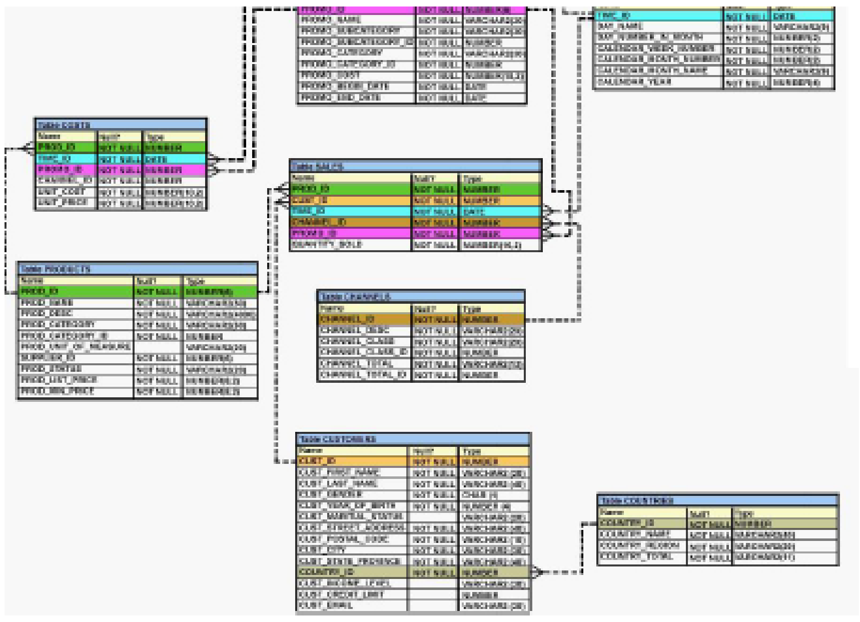
```
SQL> SELECT cust_id,cust_last_name
FROM customers
WHERE country_id IN(SELECT country_id
                    FROM customers
                    WHERE cust_last_name ='King')
AND cust_credit_limit < (SELECT MAX(cust_credit_limit)
                    FROM customers
                    WHERE country_id IN(SELECT country_id
                                FROM customers
                                WHERE cust_last_name='King'));
```

Which statement is true regarding the outcome of the above query?

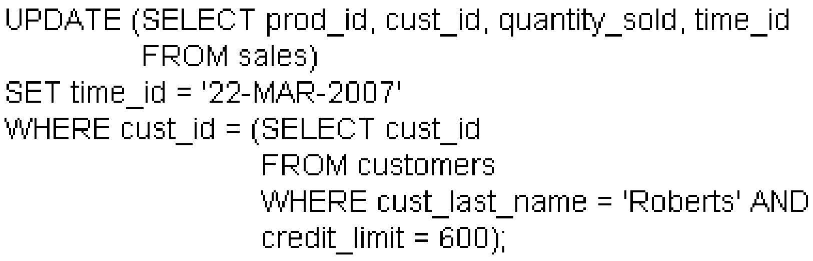
A. It produces an error and the < operator should be replaced by < ANY to get the required output
B. It produces an error and the IN operator should be replaced by = in the WHERE clause of the main query to get the required output
C. It executes and shows the required result
D. It produces an error and the < operator should be replaced by < ALL to get the required output

Answer: C

NEW QUESTION 181
You work as a database administrator at ABC.com. You study the exhibit carefully.
Exhibit:



and examine the structure of CUSTOMRS AND SALES tables:
Evaluate the following SQL statement:
Exhibit:



```
UPDATE (SELECT prod_id, cust_id, quantity_sold, time_id
          FROM sales)
SET time_id = '22-MAR-2007'
WHERE cust_id = (SELECT cust_id
                  FROM customers
                  WHERE cust_last_name = 'Roberts' AND
                  credit_limit = 600);
```

Which statement is true regarding the execution of the above UPDATE statement?

A. It would not execute because the SELECT statement cannot be used in place of the table name
B. It would execute and restrict modifications to only the column specified in the SELECT statement
C. It would not execute because a sub query cannot be used in the WHERE clause of an UPDATE statement
D. It would not execute because two tables cannot be used in a single UPDATE statement

Answer: B

NEW QUESTION 183
Which statement is true regarding the COALESCE function?

A. It can have a maximum of five expressions in a lis
B. It returns the highest NOT NULL value in the list for all row
C. It requires that all expressions in the list must be of the same data typ
D. It requires that at least one of the expressions in the list must have a NOT NULL valu

Answer: C

Explanation:
The COALESCE Function The COALESCE function returns the first nonnull value from its parameter list. If all its parameters are null, then null is returned. The COALESCE function takes two mandatory parameters and any number of optional parameters. The syntax is COALESCE(expr1, expr2,…,exprn), where expr1 is returned if it is not null, else expr2 if it is not null, and so on. COALESCE is a general form of the NVL function, as the following two equations illustrate: COALESCE(expr1,expr2) = NVL(expr1,expr2) COALESCE(expr1,expr2,expr3) = NVL(expr1,NVL(expr2,expr3)) The data type COALESCE returns if a not null value is found is the same as that of the first not null parameter. To avoid an “ORA-00932: inconsistent data types” error, all not null parameters must have data types compatible with the first not null parameter.

NEW QUESTION 186
Examine these statements:
CREATE ROLE registrar;
GRANT UPDATE ON student_grades TO registrar;
GRANT registrar to user1, user2, user3;
What does this set of SQL statements do?

A. The set of statements contains an error and does not wor
B. It creates a role called REGISTRAR, adds the MODIFY privilege on the STUDENT_GRADES object to the role, and gives the REGISTRAR role to three user
C. It creates a role called REGISTRAR, adds the UPDATE privilege on the STUDENT_GRADES object to the role, and gives the REGISTRAR role to three user
D. It creates a role called REGISTRAR, adds the UPDATE privilege on the STUDENT_GRADES object to the role, and creates three users with the rol
E. It creates a role called REGISTRAR, adds the UPDATE privilege on three users, and gives the REGISTRAR role to the STUDENT_GRADES objec
F. It creates a role called STUDENT_GRADES, adds the UPDATE privilege on three users, and gives the UPDATE role to the registra

Answer: C

Explanation: the statement will create a role call REGISTRAR, grant UPDATE on student_grades to registrar, grant the role to user1,user2 and user3.
Incorrect Answer: Athe statement does not contain error Bthere is no MODIFY privilege Dstatement does not create 3 users with the role Eprivilege is grant to role then grant to user Fprivilege is grant to role then grant to user

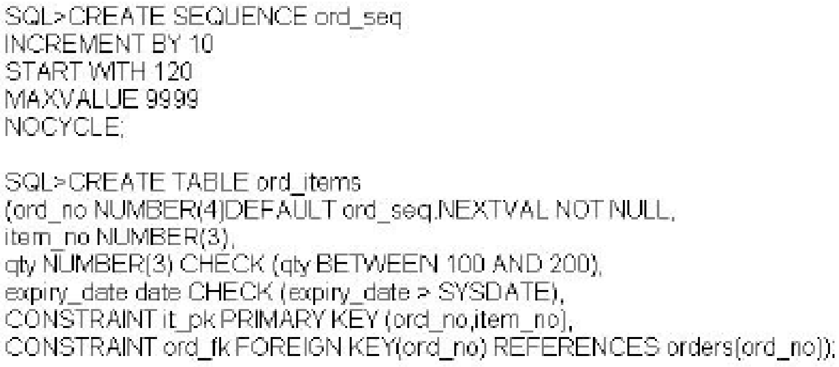
NEW QUESTION 188
You issue the following command to drop the PRODUCTS table:

SQL>DROP TABLE products;
What is the implication of this command? (Choose all that apply.)

A. All data in the table are deleted but the table structure will remain
B. All data along with the table structure is deleted
C. All views and synonyms will remain but they are invalidated
D. The pending transaction in the session is committed
E. All indexes on the table will remain but they are invalidated

Answer: BCD

NEW QUESTION 189
Evaluate the following SQL commands:



```
SQL>CREATE SEQUENCE ord_seq
INCREMENT BY 10
START WITH 120
MAXVALUE 9999
NOCYCLE;

SQL>CREATE TABLE ord_items
(ord_no NUMBER(4)DEFAULT ord_seq.NEXTVAL NOT NULL,
item_no NUMBER(3),
qty NUMBER(3) CHECK (qty BETWEEN 100 AND 200),
expiry_date date CHECK (expiry_date > SYSDATE),
CONSTRAINT it_pk PRIMARY KEY (ord_no,item_no),
CONSTRAINT ord_fk FOREIGN KEY(ord_no) REFERENCES orders(ord_no));
```

The command to create a table fails. Identify the reason for the SQL statement failure?
(Choose all that apply.)

A. You cannot use SYSDATE in the condition of a CHECK constrain
B. You cannot use the BETWEEN clause in the condition of a CHECK constrain
C. You cannot use the NEXTVAL sequence value as a DEFAULT value for a colum
D. You cannot use ORD_NO and ITEM_NO columns as a composite primary key because ORD NO is also the FOREIGN KE

Answer: AC

Explanation:
CHECK Constraint The CHECK constraint defines a condition that each row must satisfy. The condition can use the same constructs as the query conditions, with the following exceptions: References to the CURRVAL, NEXTVAL, LEVEL, and ROWNUM pseudocolumns Calls to SYSDATE, UID, USER, and USERENV functions Queries that refer to other values in other rows A single column can have multiple CHECK constraints that refer to the column in its
definition.
There is no limit to the number of CHECK constraints that you can define on a column.
CHECK constraints can be defined at the column level or table level.
CREATE TABLE employees
(...
salary NUMBER(8,2) CONSTRAINT emp_salary_min
CHECK (salary > 0),

NEW QUESTION 191
You need to create a table for a banking application. One of the columns in the table has the following requirements:
You want a column in the table to store the duration of the credit period
The data in the column should be stored in a format such that it can be easily added and subtracted with DATE data type without using conversion
The maximum period of the credit provision in the application is 30 days
the interest has to be calculated for the number of days an individual has taken a credit for
Which data type would you use for such a column in the table?

A. INTERVAL YEAR TO MONTH
B. NUMBER
C. TIMESTAMP
D. DATE
E. INTERVAL DAY TO SECOND

Answer: E

NEW QUESTION 196
View the Exhibit to examine the description for the SALES table. Which views can have all DML operations performed on it? (Choose all that apply.)

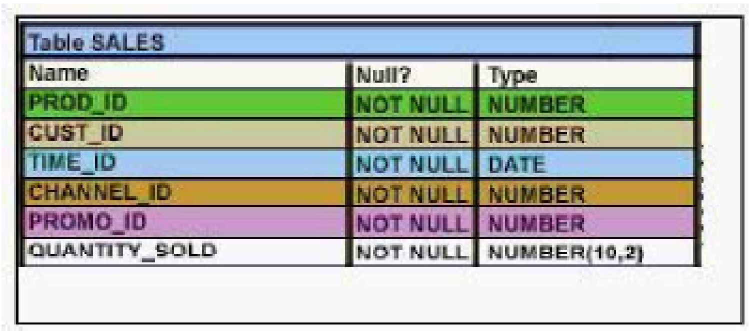


Table SALES

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)

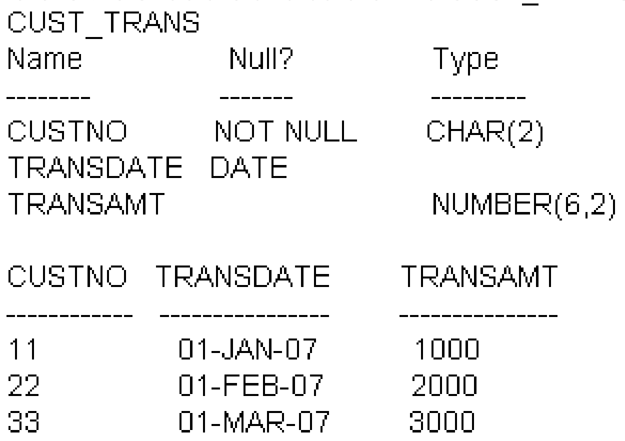
A. CREATE VIEW v3 AS SELECT * FROM SALES WHERE cust_id = 2034 WITH CHECK OPTION;
B. CREATE VIEW v1 AS SELECT * FROM SALES WHERE time_id <= SYSDATE - 2*365 WITH CHECK OPTION;
C. CREATE VIEW v2 AS SELECT prod_id, cust_id, time_id FROM SALES WHERE time_id <= SYSDATE - 2*365 WITH CHECK OPTION;

D. CREATE VIEW v4 AS SELECT prod_id, cust_id, SUM(quantity_sold) FROM SALES WHERE time_id <= SYSDATE - 2*365 GROUP BY prod_id, cust_id WITH CHECK OPTION;

Answer: AB

Explanation:
Creating a View You can create a view by embedding a subquery in the CREATE VIEW statement. In the syntax: CREATE [OR REPLACE] [FORCE|NOFORCE] VIEW view [(alias[, alias]...)] AS subquery [WITH CHECK OPTION [CONSTRAINT constraint]] [WITH READ ONLY [CONSTRAINT constraint]]; OR REPLACE Re-creates the view if it already exists FORCE Creates the view regardless of whether or not the base tables exist NOFORCE Creates the view only if the base tables exist (This is the default.) View Is the name of the view alias Specifies names for the expressions selected by the view's query (The number of aliases must match the number of expressions selected by the view.) subquery Is a complete SELECT statement (You can use aliases for the columns in the SELECT list.) WITH CHECK OPTION Specifies that only those rows that are accessible to the view can be inserted or updated ANSWER D constraint Is the name assigned to the CHECK OPTION constraint WITH READ ONLY Ensures that no DML operations can be performed on this view Rules for Performing DML Operations on a View You cannot add data through a view if the view includes: Group functions A GROUP BY clause The DISTINCT keyword The pseudocolumn ROWNUM keyword Columns defined by expressions NOT NULL columns in the base tables that are not selected by the view – ANSWER C

NEW QUESTION 200
Here is the structure and data of the CUST_TRANS table: Exhibit:



```
CUST_TRANS
Name            Null?          Type
--------       -------         ---------
CUSTNO         NOT NULL       CHAR(2)
TRANSDATE   DATE
TRANSAMT                       NUMBER(6,2)

CUSTNO   TRANSDATE      TRANSAMT
------------ ----------------    ---------------
11            01-JAN-07        1000
22            01-FEB-07        2000
33            01-MAR-07        3000
```

Dates are stored in the default date format dd-mm-rr in the CUST_TRANS table.
Which three SQL statements would execute successfully? (Choose three.)

A. SELECT transdate + '10' FROM cust_trans;
B. SELECT * FROM cust_trans WHERE transdate = '01-01-07';
C. SELECT transamt FROM cust_trans WHERE custno > '11';
D. SELECT * FROM cust_trans WHERE transdate='01-JANUARY-07';
E. SELECT custno + 'A' FROM cust_trans WHERE transamt > 2000;

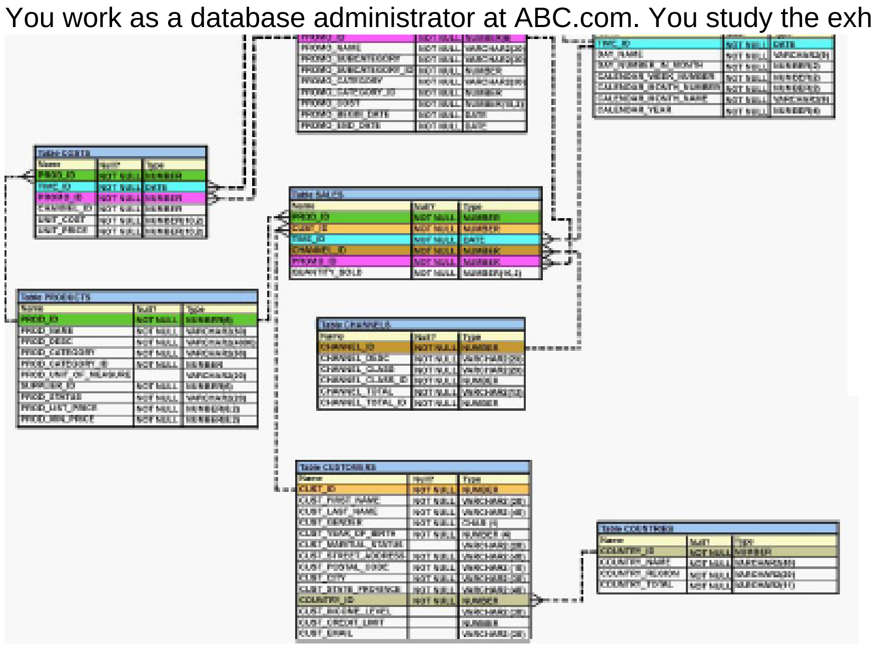
Answer: ACD

NEW QUESTION 204
You need to calculate the number of days from 1st Jan 2007 till date:
Dates are stored in the default format of dd-mm-rr.
Which two SQL statements would give the required output? (Choose two.)

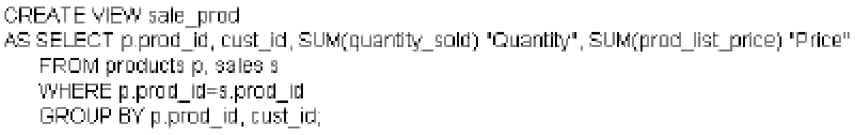
A. SELECT SYSDATE - TO_DATE('01/JANUARY/2007') FROM DUAL;
B. SELECT TO_DATE(SYSDATE,'DD/MONTH/YYYY')-'01/JANUARY/2007' FROM DUAL;
C. SELECT SYSDATE - TO_DATE('01-JANUARY-2007') FROM DUAL
D. SELECT SYSDATE - '01-JAN-2007' FROM DUAL
E. SELECT TO_CHAR(SYSDATE,'DD-MON-YYYY')-'01-JAN-2007' FROM DUAL;

Answer: AC

NEW QUESTION 206
You work as a database administrator at ABC.com. You study the exhibit carefully. Exhibit:



You want to create a SALE_PROD view by executing the following SQL statements:



```
CREATE VIEW sale_prod
AS SELECT p.prod_id, cust_id, SUM(quantity_sold) "Quantity", SUM(prod_list_price) "Price"
    FROM products p, sales s
    WHERE p.prod_id=s.prod_id
    GROUP BY p.prod_id, cust_id;
```

Which statement is true regarding the execution of the above statement?

A. The view will be created and you can perform DLM operations on the view
B. The view will not be created because the join statements are not allowed for creating a view
C. The view will not be created because the GROUP BY clause is not allowed for creating a view
D. The view will be created but no DML operations will be allowed on the view

Answer: D

Explanation:
Rules for Performing DML Operations on a View You cannot add data through a view if the view includes: Group functions A GROUP BY clause The DISTINCT keyword The pseudocolumn ROWNUM keyword Columns defined by expressions NOT NULL columns in the base tables that are not selected by the view

NEW QUESTION 208
Which three statements/commands would cause a transaction to end? (Choose three.)

A. COMMIT
B. SELECT
C. CREATE
D. ROLLBACK
E. SAVEPOINT

Answer: ACD

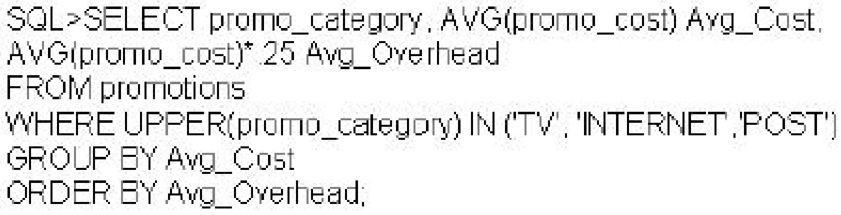
NEW QUESTION 210
Which four are types of functions available in SQL? (Choose 4)

A. string
B. character
C. integer
D. calendar
E. numeric
F. translation
G. date
H. conversion

Answer: BEGH

Explanation: Explanation: SQL have character, numeric, date, conversion function.
Incorrect Answer:
ASQL have character, numeric, date, conversion function.
CSQL have character, numeric, date, conversion function.
DSQL have character, numeric, date, conversion function.
FSQL have character, numeric, date, conversion function.
Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 3-3

NEW QUESTION 211
View the Exhibit and examine the structure of the PROMOTIONS table. Evaluate the following SQL statement:



```
SQL>SELECT promo_category, AVG(promo_cost) Avg_Cost,
AVG(promo_cost)*.25 Avg_Overhead
FROM promotions
WHERE UPPER(promo_category) IN ('TV', 'INTERNET','POST')
GROUP BY Avg_Cost
ORDER BY Avg_Overhead;
```

The above query generates an error on execution.
Which clause in the above SQL statement causes the error?

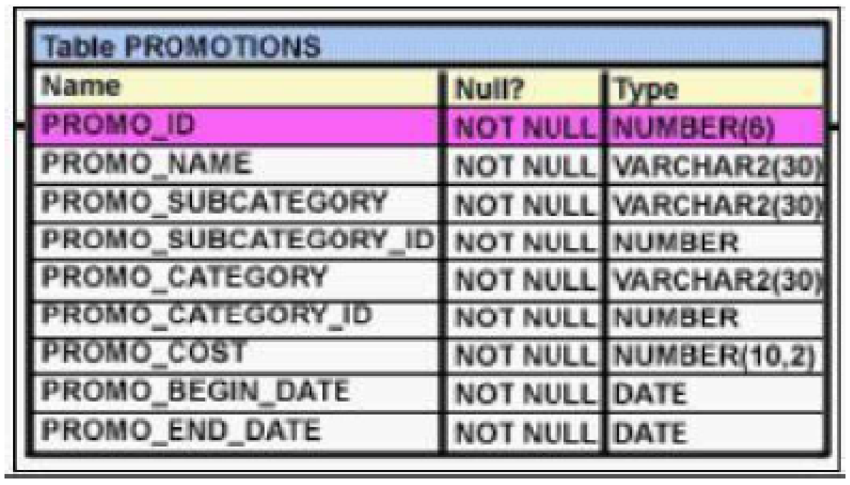
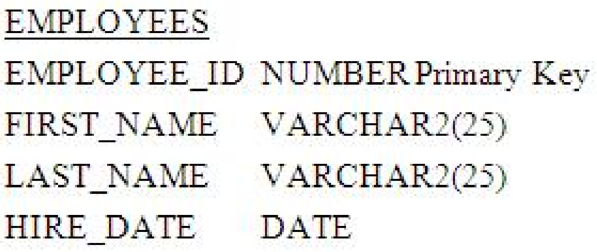


Table PROMOTIONS		
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(6)
PROMO_NAME	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY	NOT NULL	VARCHAR2(30)
PROMO_SUBCATEGORY_ID	NOT NULL	NUMBER
PROMO_CATEGORY	NOT NULL	VARCHAR2(30)
PROMO_CATEGORY_ID	NOT NULL	NUMBER
PROMO_COST	NOT NULL	NUMBER(10,2)
PROMO_BEGIN_DATE	NOT NULL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

A. WHERE
B. SELECT
C. GROUP BY
D. ORDER BY

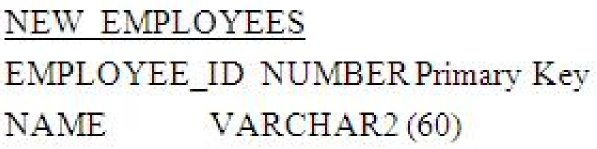
Answer: C

NEW QUESTION 216
Examine the structure of the EMPLOYEES and NEW_EMPLOYEES tables:



EMPLOYEES

EMPLOYEE_ID	NUMBER Primary Key
FIRST_NAME	VARCHAR2(25)
LAST_NAME	VARCHAR2(25)
HIRE_DATE	DATE



NEW_EMPLOYEES

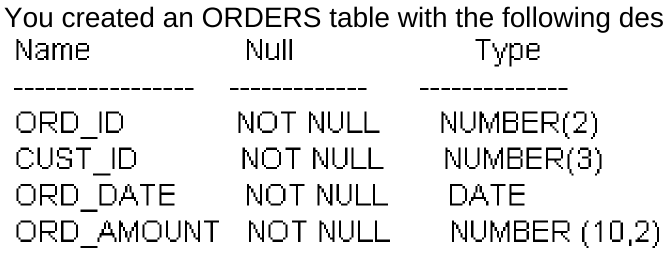
EMPLOYEE_ID	NUMBER Primary Key
NAME	VARCHAR2 (60)

Which DELETE statement is valid?

A. DELETE FROM employeesWHERE employee_id = (SELECT employee_id FROM employees);
B. DELETE * FROM employeesWHERE employee_id=(SELECT employee_id FROM new_employees);
C. DELETE FROM employeesWHERE employee_id IN (SELECT employee_id FROM new_employees WHERE name = ‘Carrey’);
D. DELETE * FROM employeesWHERE employee_id IN (SELECT employee_id FROM new_employees WHERE name = ‘Carrey’);

Answer: C

NEW QUESTION 219
You created an ORDERS table with the following description: Exhibit:



```
Name              Null           Type
-----------------  -------------  --------------
ORD_ID            NOT NULL       NUMBER(2)
CUST_ID           NOT NULL       NUMBER(3)
ORD_DATE          NOT NULL       DATE
ORD_AMOUNT        NOT NULL       NUMBER (10,2)
```

You inserted some rows in the table. After some time, you want to alter the table by creating the PRIMARY KEY constraint on the ORD_ID column.
Which statement is true in this scenario?

A. You cannot add a primary key constraint if data exists in the column
B. You can add the primary key constraint even if data exists, provided that there are no duplicate values
C. The primary key constraint can be created only a the time of table creation
D. You cannot have two constraints on one column

Answer: B

NEW QUESTION 220
View the Exhibit and examine the structure of the CUSTOMERS table.

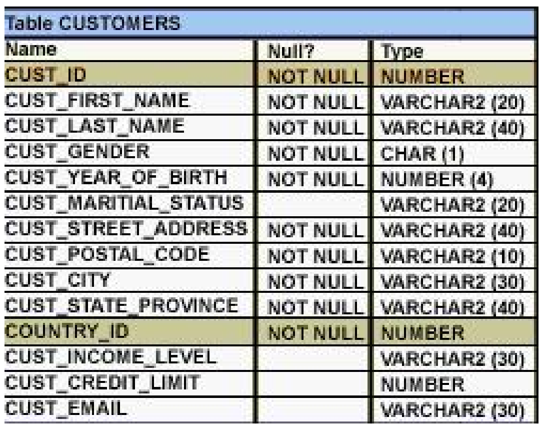
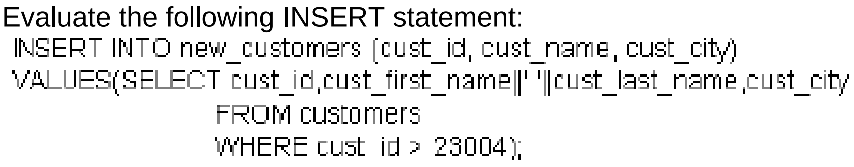


Table CUSTOMERS

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_MARITIAL_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)

NEW_CUSTOMERS is a new table with the columns CUST_ID, CUST_NAME and CUST_CITY that have the same data types and size as the corresponding columns in the CUSTOMERS table.
Evaluate the following INSERT statement:



```
INSERT INTO new_customers (cust_id, cust_name, cust_city)
VALUES(SELECT cust_id,cust_first_name||' '||cust_last_name,cust_city
                    FROM customers
                    WHERE cust_id > 23004);
```

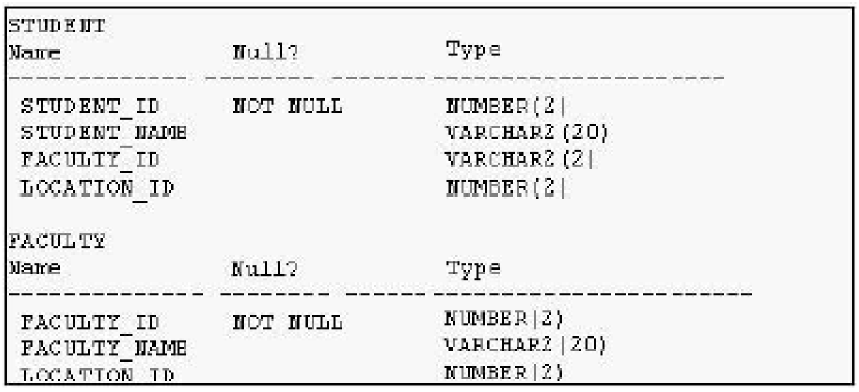
The INSERT statement fails when executed. What could be the reason?

A. The VALUES clause cannot be used in an INSERT with a subquery
B. The total number of columns in the NEW_CUSTOMERS table does not match the total number of columns in the CUSTOMERS table
C. The WHERE clause cannot be used in a sub query embedded in an INSERT statement
D. Column names in the NEW_CUSTOMERS and CUSTOMERS tables do not match

Answer: A

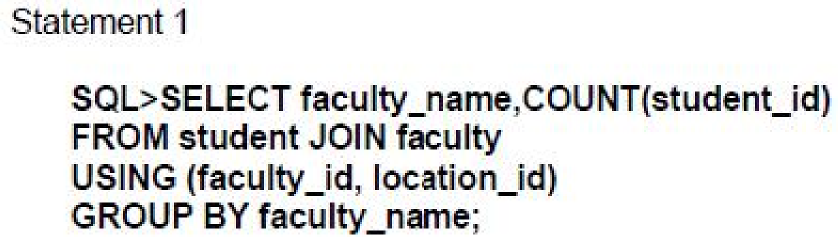
Explanation:
Copying Rows from Another Table
Write your INSERT statement with a subquery:
Do not use the VALUES clause.
Match the number of columns in the INSERT clause to those in the subquery.
Inserts all the rows returned by the subquery in the table, sales_reps.

NEW QUESTION 223
View the Exhibit for the structure of the STUDENT and FACULTY tables.



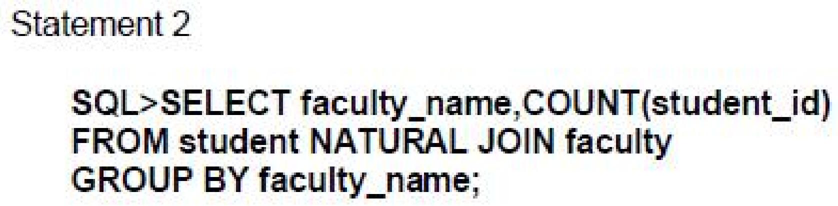
```
STUDENT
Name                Null?           Type
------------- --------- ------- -------------------
 STUDENT_ID         NOT NULL        NUMBER(2)
 STUDENT_NAME                       VARCHAR2(20)
 FACULTY_ID                         VARCHAR2(2)
 LOCATION_ID                        NUMBER(2)

FACULTY
Name                Null?           Type
-------------- -------- ------ ---------------------
 FACULTY_ID         NOT NULL        NUMBER(2)
 FACULTY_NAME                       VARCHAR2(20)
 LOCATION_ID                        NUMBER(2)
```

You need to display the faculty name followed by the number of students handled by the faculty at the base location. Examine the following two SQL statements:



Statement 1

```
SQL>SELECT faculty_name,COUNT(student_id)
FROM student JOIN faculty
USING (faculty_id, location_id)
GROUP BY faculty_name;
```



Statement 2

```
SQL>SELECT faculty_name,COUNT(student_id)
FROM student NATURAL JOIN faculty
GROUP BY faculty_name;
```

Which statement is true regarding the outcome?

A. Only statement 1 executes successfully and gives the required resul
B. Only statement 2 executes successfully and gives the required resul
C. Both statements 1 and 2 execute successfully and give different result
D. Both statements 1 and 2 execute successfully and give the same required resul

Answer: D

NEW QUESTION 226
View the Exhibit and examine the structure of the CUSTOMERS table.

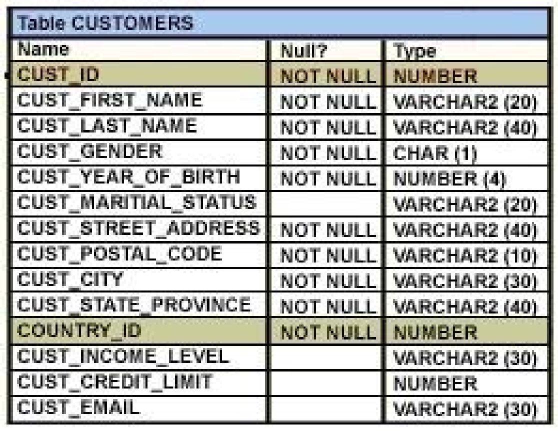


Table CUSTOMERS

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_MARITIAL_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)

Which two tasks would require subqueries or joins to be executed in a single statement? (Choose two.)

A. listing of customers who do not have a credit limit and were born before 1980
B. finding the number of customers, in each city, whose marital status is 'married'
C. finding the average credit limit of male customers residing in 'Tokyo' or 'Sydney'
D. listing of those customers whose credit limit is the same as the credit limit of customers residing in the city 'Tokyo'
E. finding the number of customers, in each city, whose credit limit is more than the average credit limit of all the customers

Answer: DE

Explanation:
Describe the Types of Problems That the Subqueries Can Solve There are many situations where you will need the result of one query as the input for another. Use of a Subquery Result Set for Comparison Purposes Which employees have a salary that is less than the average salary? This could be answered by two statements, or by a single statement with a subquery. The following example uses two statements: select avg(salary) from employees; select last_name from employees where salary < result_of_previous_query ;
Alternatively, this example uses one statement with a subquery:
select last_name from employees where salary < (select avg(salary)from employees);
In this example, the subquery is used to substitute a value into the WHERE clause of the parent query: it is returning a single value, used for comparison with the rows retrieved by the parent query.
The subquery could return a set of rows. For example, you could use the following to find

all departments that do actually have one or more employees assigned to them:
select department_name from departments where department_id in
(select distinct(department_id) from employees);

NEW QUESTION 228
Examine the data in the CUST_NAME column of the CUSTOMERS table. CUST_NAME
Renske Ladwig Jason Mallin Samuel McCain Allan MCEwen Irene Mikkilineni Julia Nayer
You need to display customers' second names where the second name starts with "Mc" or "MC."
Which query gives the required output?

A. SELECT SUBSTR(cust_name, INSTR(cust_name,' ')+1) FROM customers WHERE INITCAP(SUBSTR(cust_name, INSTR(cust_name,' ')+1))='Mc';
B. SELECT SUBSTR(cust_name, INSTR(cust_name,' ')+1) FROM customers WHERE INITCAP(SUBSTR(cust_name, INSTR(cust_name,' ')+1)) LIKE 'Mc%';
C. SELECT SUBSTR(cust_name, INSTR(cust_name,' ')+1) FROM customers WHERE SUBSTR(cust_name, INSTR(cust_name,' ')+1) LIKE INITCAP('MC%');
D. SELECT SUBSTR(cust_name, INSTR(cust_name,' ')+1) FROM customers WHERE INITCAP(SUBSTR(cust_name, INSTR(cust_name,' ')+1)) = INITCAP('MC%');

Answer: B

NEW QUESTION 230
You need to write a SQL statement that returns employee name, salary, department ID, and maximum salary earned in the department of the employee for all employees who earn less than the maximum salary in their department.
Which statement accomplishes this task?

A. SELECT a.emp_name, a.sal, b.dept_id, MAX(sal) FROM employees a, departments b WHERE a.dept_id = b.dept_id AND a.sal < MAX(sal) GROUP BY b.dept_id;
B. SELECT a.emp_name, a.sal, a.dept_id, b.maxsal FROM employees a, (SELECT dept_id, MAX(sal) maxsal FROM employees GROUP BY dept_id) b WHERE a.dept_id = b.dept_id AND a.sal < b.maxsal;
C. SELECT a.emp_name, a.sal, a.dept_id, b.maxsal FROM employees a WHERE a.sal < (SELECT MAX(sal) maxsal FROM employees b GROUP BY dept_id);
D. SELECT emp_name, sal, dept_id, maxsal FROM employees, (SELECT dept_id, MAX(sal) maxsal FROM employees GROUP BY dept_id) WHERE a.sal < maxsal;

Answer: B

Explanation: function MAX(column_name)
Incorrect Answer:
Ainvalid statement
Cinner query return more than one line
Dcolumn maxsal does not exists.
Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 5-7

NEW QUESTION 232
Examine the structure of the TRANSACTIONS table:
Name Null Type
TRANS_ID NOT NULL NUMBER(3)
CUST_NAME VARCHAR2(30)
TRANS_DATE TIMESTAMP
TRANS_AMT NUMBER(10,2)
You want to display the date, time, and transaction amount of transactions that where done before 12 noon.
The value zero should be displayed for transactions where the transaction amount has not been entered.
Which query gives the required result?

A. SELECT TO_CHAR(trans_date,'dd-mon-yyyy hh24:mi:ss'), TO_CHAR(trans_amt,'$99999999D99') FROM transactions WHERE TO_NUMBER(TO_DATE(trans_date,'hh24')) < 12 AND COALESCE(trans_amt,NULL)<>NULL;
B. SELECT TO_CHAR(trans_date,'dd-mon-yyyy hh24:mi:ss'), NVL(TO_CHAR(trans_amt,'$99999999D99'),0) FROM transactions WHERE TO_CHAR(trans_date,'hh24') < 12;
C. SELECT TO_CHAR(trans_date,'dd-mon-yyyy hh24:mi:ss'), COALESCE(TO_NUMBER(trans_amt,'$99999999.99'),0) FROM transactions WHERE TO_DATE(trans_date,'hh24') < 12;
D. SELECT TO_DATE (trans_date,'dd-mon-yyyy hh24:mi:ss'), NVL2(trans_amt,TO_NUMBER(trans_amt,'$99999999.99'), 0) FROM transactions WHERE TO_DATE(trans_date,'hh24') < 12;

Answer: B

NEW QUESTION 233
The STUDENT_GRADES table has these columns:
STUDENT_ID NUMBER(12)
SEMESTER_END DATE
GPA NUMBER(4,3)
Which statement finds the highest grade point average (GPA) per semester?

A. SELECT MAX(gpa) FROM student_grades WHERE gpa IS NOT NULL;
B. SELECT (gpa) FROM student_grades GROUP BY semester_end WHERE gpa IS NOT NULL;
C. SELECT MAX(gpa) FROM student_grades WHERE gpa IS NOT NULL GROUP BY semester_end;
D. SELECT MAX(gpa) GROUP BY semester_end WHERE gpa IS NOT NULL FROM student_grades;
E. SELECT MAX(gpa) FROM student_grades GROUP BY semester_end WHERE gpa IS NOT NULL;

Answer: C

Explanation: Explanation: For highest gpa value MAX function is needed, for result with per semester GROUP BY clause is needed
Incorrect Answer: Aper semester condition is not included Bresult would not display the highest gpa value Dinvalid syntax error Einvalid syntax error Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 5-7

NEW QUESTION 237
View the Exhibit and examine the structure of the PRODUCTS table.
You want to display only those product names with their list prices where the list price is at least double the minimum price. The report should start with the product name having the maximum list price satisfying this
condition.
Evaluate the following SQL statement:
SQL>SELECT prod_name,prod_list_price FROM products WHERE prod_list_price >= 2 * prod_min_price
Which ORDER BY clauses can be added to the above SQL statement to get the correct output?
(Choose all that apply.)

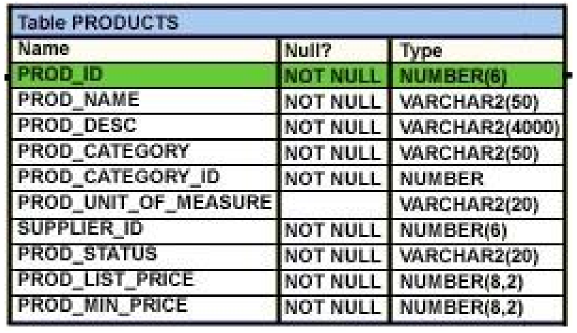


Table PRODUCTS

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)

A. ORDER BY prod_list_price DESC, prod_name;
B. ORDER BY (2*prod_min_price)DESC, prod_name;
C. ORDER BY prod_name, (2*prod_min_price)DESC;
D. ORDER BY prod_name DESC, prod_list_price DESC;
E. ORDER BY prod_list_price DESC, prod_name DESC;

Answer: AE

Explanation:
Using the ORDER BY Clause The order of rows that are returned in a query result is undefined. The ORDER BY clause can be used to sort the rows. However, if you use the ORDER BY clause, it must be the last clause of the SQL statement. Further, you can specify an expression, an alias, or a column position as the sort condition. Syntax SELECT expr FROM table [WHERE condition(s)] [ORDER BY {column, expr, numeric_position} [ASC|DESC]]; In the syntax: ORDER BY specifies the order in which the retrieved rows are displayed ASC orders the rows in ascending order (This is the default order.)
DESC orders the rows in descending order If the ORDER BY clause is not used, the sort order is undefined, and the Oracle server may not fetch rows in the same order for the same query twice. Use the ORDER BY clause to display the rows in a specific order. Note: Use the keywords NULLS FIRST or NULLS LAST to specify whether returned rows containing null values should appear first or last in the ordering sequence.

NEW QUESTION 238
Which statements are true regarding the FOR UPDATE clause in a SELECT statement? (Choose all that apply.)

A. It locks only the columns specified in the SELECT lis
B. It locks the rows that satisfy the condition in the SELECT statemen
C. It can be used only in SELECT statements that are based on a single tabl
D. It can be used in SELECT statements that are based on a single or multiple table
E. After it is enforced by a SELECT statement, no other query can access the same rows until a COMMIT or ROLLBACK is issue

Answer: BD

Explanation:
FOR UPDATE Clause in a SELECT Statement Locks the rows in the EMPLOYEES table where job_id is SA_REP. Lock is released only when you issue a ROLLBACK or a COMMIT. If the SELECT statement attempts to lock a row that is locked by another user, the database waits until the row is available, and then returns the results of the SELECT statement. FOR UPDATE Clause in a SELECT Statement When you issue a SELECT statement against the database to query some records, no locks are placed on the selected rows. In general, this is required because the number of records locked at any given time is (by default) kept to the absolute minimum: only those records that have been changed but not yet committed are locked. Even then, others will be able to read those records as they appeared before the change (the “before image” of the data). There are times, however, when you may want to lock a set of records even before you change them in your program. Oracle offers the FOR UPDATE clause of the SELECT statement to perform this locking. When you issue a SELECT...FOR UPDATE statement, the relational database management system (RDBMS) automatically obtains exclusive row-level locks on all the rows identified by the SELECT statement, thereby holding the records “for your changes only.” No one else will be able to change any of these records until you perform a ROLLBACK or a COMMIT. You can append the optional keyword NOWAIT to the FOR UPDATE clause to tell the Oracle server not to wait if the table has been locked by another user. In this case, control will be returned immediately to your program or to your SQL Developer environment so that you can perform other work, or simply wait for a period of time before trying again. Without the NOWAIT clause, your process will block until the table is available, when the locks are released by the other user through the issue of a COMMIT or a ROLLBACK command.

NEW QUESTION 242
Top N analysis requires _____ and _____. (Choose two.)

A. the use of rowid
B. a GROUP BY clause
C. an ORDER BY clause
D. only an inline view
E. an inline view and an outer query

Answer: CE

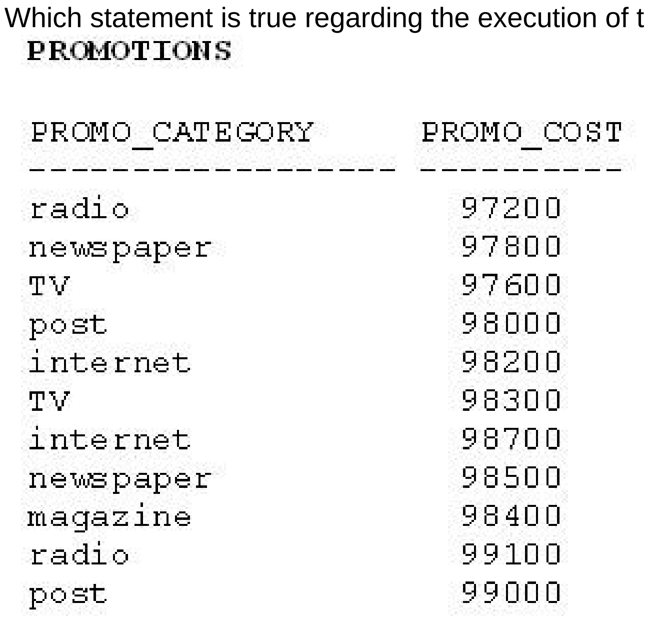
Explanation:
The correct statement for Top-N Analysis SELECT [coloumn_list], ROWNUM FROM (SELECT [coloumn_list]
FROM table
ORDER BY Top-N_coloumn)
WHERE ROWNUM <= N;
Incorrect Answer:
AROWID is not require
BGROUP BY clause is not require
DMust have inline view and outer query.
Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 11-23

NEW QUESTION 246
View the Exhibits and examine the structures of the PRODUCTS and SALES tables. Which two SQL statements would give the same output? (Choose two.)

A. SELECT prod_id FROM products INTERSECT SELECT prod_id FROM sales;
B. SELECT prod_id FROM products MINUS SELECT prod_id FROM sales;
C. SELECT DISTINCT p.prod_id FROM products p JOIN sales s ON p.prod_id=s.prod_id;
D. SELECT DISTINCT p.prod_id FROM products p JOIN sales s ON p.prod_id <> s.prod_id;

Answer: AC

NEW QUESTION 250
View the Exhibit and examine the data in the PROMO_CATEGORY and PROMO_COST columns of the PROMOTIONS table.
Evaluate the following two queries:
SQL>SELECT DISTINCT promo_category to_char(promo_cost)"code"
FROM promotions
ORDER BY code;
SQL>SELECT DISTINCT promo_category promo_cost "code"
FROM promotions
ORDER BY 1;
Which statement is true regarding the execution of the above queries?



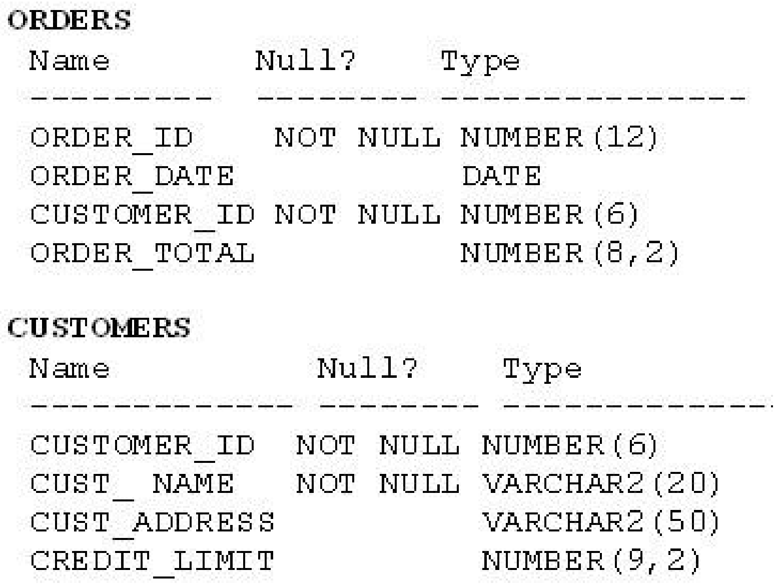
```
PROMOTIONS

PROMO_CATEGORY      PROMO_COST
------------------- ----------
radio                   97200
newspaper               97800
TV                      97600
post                    98000
internet                98200
TV                      98300
internet                98700
newspaper               98500
magazine                98400
radio                   99100
post                    99000
```

A. Only the first query executes successfull
B. Only the second query executes successfull
C. Both queries execute successfully but give different result
D. Both queries execute successfully and give the same resul

Answer: B

Explanation:
Note: You cannot use column alias in the WHERE clause.

NEW QUESTION 254
View the Exhibit and examine the structure of the ORDERS and CUSTOMERS tables.



```
ORDERS
 Name        Null?     Type
 ---------  --------- ---------------
 ORDER_ID    NOT NULL NUMBER(12)
 ORDER_DATE           DATE
 CUSTOMER_ID NOT NULL NUMBER(6)
 ORDER_TOTAL          NUMBER(8,2)

CUSTOMERS
 Name           Null?     Type
 -------------  --------  -------------
 CUSTOMER_ID   NOT NULL NUMBER(6)
 CUST_ NAME    NOT NULL VARCHAR2(20)
 CUST_ADDRESS           VARCHAR2(50)
 CREDIT_LIMIT           NUMBER(9,2)
```

Evaluate the following SQL command:
SQL> SELECT o.order_id, c.cust_name, o.order_total, c.credit_limit FROM orders o JOIN customers c USING (customer_id) WHERE o.order_total > c.credit_limit FOR UPDATE ORDER BY o.order_id;
Which two statements are true regarding the outcome of the above query? (Choose two.)

A. It locks all the rows that satisfy the condition in the statemen
B. It locks only the columns that satisfy the condition in both the table
C. The locks are released only when a COMMIT or ROLLBACK is issue
D. The locks are released after a DML statement is executed on the locked row

Answer: AC

Explanation:
FOR UPDATE Clause in a SELECT Statement
.
Locks the rows in the EMPLOYEES table where job_id is SA_REP.
.
Lock is released only when you issue a ROLLBACK or a COMMIT.
.
If the SELECT statement attempts to lock a row that is locked by another user, the database waits until the row is available, and then returns the results of the SELECTstatement SELECT employee_id, salary, commission_pct, job_id FROM employees WHERE job_id = 'SA_REP' FOR UPDATE ORDER BY employee_id;

NEW QUESTION 256
Which statements are true regarding the WHERE and HAVING clauses in a SELECT statement?
(Choose all that apply.)

A. The HAVING clause can be used with aggregate functions in subquerie
B. The WHERE clause can be used to exclude rows after dividing them into group
C. The WHERE clause can be used to exclude rows before dividing them into group
D. The aggregate functions and columns used in the HAVING clause must be specified in the SELECT list of the quer
E. The WHERE and HAVING clauses can be used in the same statement only if they are applied to different columns in the tabl

Answer: AC

NEW QUESTION 258
Examine the SQL statement that creates ORDERS table:
CREATE TABLE orders (SER_NO NUMBER UNIQUE, ORDER_ID NUMBER, ORDER_DATE DATE NOT NULL, STATUS VARCHAR2(10) CHECK (status IN ('CREDIT', 'CASH')), PROD_ID NUMBER REFERENCES PRODUCTS(PRODUCT_ID), ORD_TOTAL NUMBER, PRIMARY KEY (order_id, order_date));
For which columns would an index be automatically created when you execute the above SQL statement? (Choose two.)

A. SER_NO
B. ORDER_ID
C. STATUS
D. PROD_ID
E. ORD_TOTAL
F. composite index on ORDER_ID and ORDER_DATE

Answer: AF

Explanation: Index exist for UNIQUE and PRIMARY KEY constraints
Incorrect Answer: BORDER_ID is neither UNIQUE nor PRIMARY KEY CSTATUS is neither UNIQUE nor PRIMARY KEY DPROD_ID is neither UNIQUE nor PRIMARY KEY EORD_TOTAL is neither UNIQUE nor PRIMARY KEY
Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 10-15

NEW QUESTION 260
Which SQL statements would display the value 1890.55 as $1,890.55? (Choose three.)

A. SELECT TO_CHAR(1890.55,'$0G000D00') FROM DUAL;
B. SELECT TO_CHAR(1890.55,'$9,999V99') FROM DUAL;
C. SELECT TO_CHAR(1890.55,'$99,999D99') FROM DUAL;

D. SELECT TO_CHAR(1890.55,'$99G999D00') FROM DUAL;
E. SELECT TO_CHAR(1890.55,'$99G999D99') FROM DUAL;

Answer: ADE

NEW QUESTION 263
Evaluate the SQL statement
DROP TABLE DEPT:
Which four statements are true of the SQL statement? (Choose four)

A. You cannot roll back this statemen
B. All pending transactions are committe
C. All views based on the DEPT table are delete
D. All indexes based on the DEPT table are droppe
E. All data in the table is deleted, and the table structure is also delete
F. All data in the table is deleted, but the structure of the table is retaine
G. All synonyms based on the DEPT table are delete

Answer: ABDE

Explanation:
You cannot roll back DROP TABLE statement. All pending transactions related on this table are committed. If the table is dropped, Oracle automatically drops any index, trigger and constraint associated with the table as well. All data in the table is deleted, and the table structure is also deleted.
Incorrect Answers
C:All views based on the DEPT table become invalid, but they are not deleted.
F:All data in the table is deleted, and the table structure is also deleted. Command TRUNCATE deletes all data in the table, but does not delete the structure of the table.
G:All synonyms based on the DEPT table are not deleted after dropping the table.
OCP Introduction to Oracle 9i: SQL Exam Guide, Jason Couchman, p. 225 Chapter 5: Creating Oracle Database Objects

NEW QUESTION 265
Which four are valid Oracle constraint types? (Choose four.)

A. CASCADE
B. UNIQUE
C. NONUNIQUE
D. CHECK
E. PRIMARY KEY
F. CONSTANT
G. NOT NULL

Answer: BDEG

Explanation:
Oracle constraint type is Not Null, Check, Primary Key, Foreign Key and Unique Incorrect Answer: AIs not Oracle constraint CIs not Oracle constraint FIs not Oracle constraint Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 10-3

NEW QUESTION 269
Which SQL statement returns a numeric value?

A. SELECT ADD_MONTHS(MAX(hire_Date), 6) FROM EMP;
B. SELECT ROUND(hire_date) FROM EMP;
C. SELECT sysdate-hire_date FROM EMP;
D. SELECT TO_NUMBER(hire_date + 7) FROM EMP;

Answer: C

Explanation:
DATE value subtract DATE value will return numeric value.
Incorrect Answer: Adoes not return numeric value Bdoes not return numeric value Ddoes not return numeric value
Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 9-13

NEW QUESTION 272
View the Exhibits and examine PRODUCTS and SALES tables.

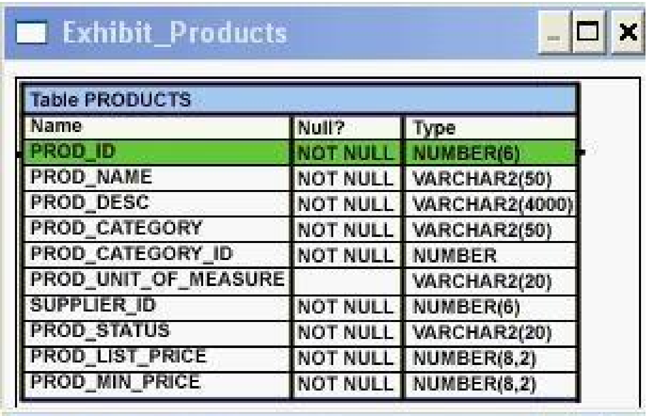


Table PRODUCTS

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)

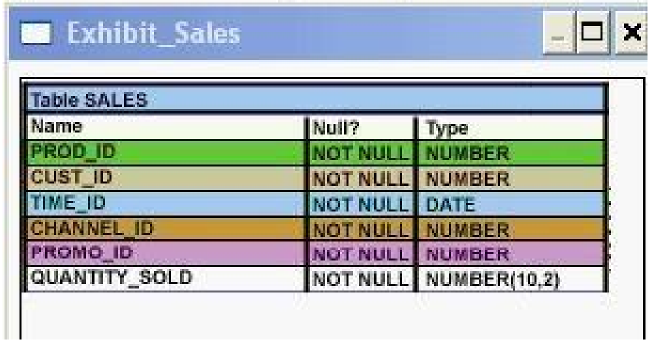


Table SALES

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)

You issue the following query to display product name and the number of times the product has been sold:
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;
What happens when the above statement is executed?

A. The statement executes successfully and produces the required outpu
B. The statement produces an error because ITEM_CNT cannot be displayed in the outer quer
C. The statement produces an error because a subquery in the FROM clause and outer-joins cannot be used togethe
D. The statement produces an error because the GROUP BY clause cannot be used in a subquery in the FROM claus

Answer: A

NEW QUESTION 275
Examine the statement:
GRANT select, insert, update
ON student_grades
TO manager
WITH GRANT OPTION;
Which two are true? (Choose two.)

A. MANAGER must be a rol
B. It allows the MANAGER to pass the specified privileges on to other user
C. It allows the MANAGER to create tables that refer to the STUDENT_GRADES tabl
D. It allows the MANAGER to apply all DML statements on the STUDENT_GRADES tabl
E. It allows the MANAGER the ability to select from, insert into, and update the STUDENT_GRADES tabl
F. It allows the MANAGER the ability to select from, delete from, and update the STUDENT_GRADES tabl

Answer: BE

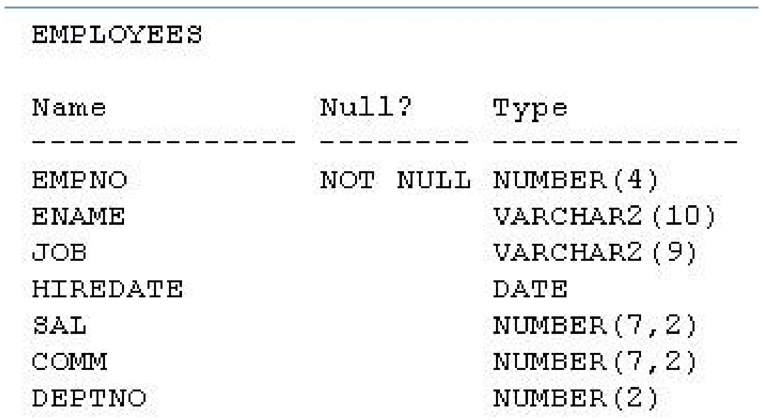
Explanation:
GRANT ROLE to ROLE/USER
Incorrect Answer: ARole can be grant to user CCreate table privilege is not granted DExecute privilege is not granted FDelete privilege is not granted
Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 13-15

NEW QUESTION 276
View the Exhibit and examine the data in the EMPLOYEES table:
You want to display all the employee names and their corresponding manager names.
Evaluate the following query:
SQL> SELECT e.employee_name "EMP NAME", m.employee_name "MGR NAME"
FROM employees e ______________ employees m
ON e.manager_id = m.employee_id;
Which JOIN option can be used in the blank in the above query to get the required output?
Exhibit:

A. only inner JOIN
B. only FULL OUTER JOIN
C. only LEFT OUTER JOIN
D. only RIGHT OUTER JOIN

Answer: C

NEW QUESTION 277
View the Exhibit and examine the structure of the EMPLOYEES table.



```
EMPLOYEES

Name            Null?     Type
--------------  --------  -------------
EMPNO           NOT NULL  NUMBER(4)
ENAME                     VARCHAR2(10)
JOB                       VARCHAR2(9)
HIREDATE                  DATE
SAL                       NUMBER(7,2)
COMM                      NUMBER(7,2)
DEPTNO                    NUMBER(2)
```

Examine the data in the ENAME and HIREDATE columns of the EMPLOYEES table:
ENAME HIREDATE
SMITH 17-DEC-80 ALLEN 20-FEB-81 WARD 22-FEB-81
You want to generate a list of user IDs as follows: USERID
Smi17DEC80 All20FEB81 War22FEB81
You issue the following query:
SQL>SELECT CONCAT(SUBSTR(INITCAP(ename),1,3), REPLACE(hiredate,'-'))
"USERID"
FROM employees;
What is the outcome?

A. It executes successfully and gives the correct outpu
B. It executes successfully but does not give the correct outpu
C. It generates an error because the REPLACE function is not vali
D. It generates an error because the SUBSTR function cannot be nested in the CONCAT functio

Answer: A

Explanation:
REPLACE(text, search_string,replacement_string) Searches a text expression for a character string and, if found, replaces it with a specified replacement string The REPLACE Function The REPLACE function replaces all occurrences of a search item in a source string with a replacement term and returns the modified source string. If the length of the replacement term is different from that of the search item, then the lengths of the returned and source strings will be different. If the search string is not found, the source string is returned unchanged. Numeric and date literals and expressions are evaluated before being implicitly cast as characters when they occur as parameters to the REPLACE function. The REPLACE function takes three parameters, with the first two being mandatory. Its syntax is REPLACE (source string, search item, [replacement term]). If the replacement term parameter is omitted, each occurrence of the search item is removed from the source string. In other words, the search item is replaced by an empty string. . The following queries illustrate the REPLACE function with numeric and date expressions: Query 1: select replace(10000-3,'9','85') from dual Query 2: select replace(sysdate, 'DEC','NOV') from dual

NEW QUESTION 278
View the Exhibit and examine the structures of the EMPLOYEES and DEPARTMENTS tables.
You want to update the EMPLOYEES table as follows:4 ? 4;
-Update only those employees who work in Boston or Seattle (locations 2900 and 2700).
-Set department_id for these employees to the department_id corresponding to London
(location_id 2100).
-Set the employees' salary in location_id 2100 to 1.1 times the average salary of their department.
-Set the employees' commission in location_id 2100 to 1.5 times the average commission of their department.
You issue the following command: 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)
What is the outcome?

A. It executes successfully and gives the correct resul
B. It executes successfully but does not give the correct resul
C. It generates an error because a subquery cannot have a join condition in an UPDATE statemen
D. It generates an error because multiple columns (SALARY, COMMISION) cannot be specified together in an UPDATE statemen

Answer: B

NEW QUESTION 282
View the Exhibit and examine the structure of the CUSTOMERS table.
In the CUSTOMERS table, the CUST_LAST_NAME column contains the values 'Anderson' and 'Ausson'.
You issue the following query:
SQL> SELECT LOWER(REPLACE(TRIM('son' FROM cust_last_name),'An','O'))
FROM CUSTOMERS
WHERE LOWER(cust_last_name) LIKE 'a%n';
What would be the outcome?

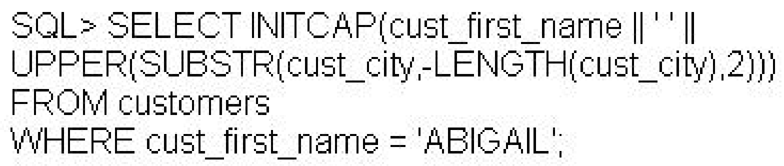
A. 'Oder' and 'Aus'
B. an error because the TRIM function specified is not valid
C. an error because the LOWER function specified is not valid
D. an error because the REPLACE function specified is not valid

Answer: B

Explanation:
Function Purpose ROUND(column|expression, n) Rounds the column, expression, or value to n decimal places or, if n is omitted, no decimal places (If n is negative, numbers to the left of decimal point are rounded.) TRUNC(column|expression, n) Truncates the column, expression, or value to n decimal places or, if n

is omitted, n defaults to zero The TRIM Function The TRIM function removes characters from the beginning or end of character literals, columns or expressions to yield one potentially shorter character item. Numeric and date literals are automatically cast as characters when they occur as parameters to the TRIM function. Numeric or date expressions are evaluated first before being converted to strings ready to be trimmed. The TRIM function takes a parameter made up of an optional and a mandatory component. Its syntax is TRIM ([trailing|leading|both] trimstring from s). The string to be trimmed (s) is mandatory. The following points list the rules governing the use of this function:
TRIM(s) removes spaces from both sides of the input string.
TRIM(trailing trimstring from s) removes all occurrences of trimstring from the end of the string s if it is present. TRIM(leading trimstring from s) removes all occurrences of trimstring from the beginning of the string s if it is present.
TRIM(both trimstring from s) removes all occurrences of trimstring from the beginning and
end of the string s if it is present.
The following queries illustrate the usage of this function:
Query 1: select trim(trailing 'e' from 1+2.14||' is pie') from dual
Query 2: select trim(both '*' from '*******Hidden*******') from dual
Query 3: select trim(1 from sysdate) from dual
ORA-30001: trim set should have only one character
30001. 00000 - "trim set should have only one character"
*Cause: Trim set contains more or less than 1 character. This is not allowed in TRIM
function.
REPLACE(text, search_string, replacement_string)
Searches a text expression for a character string and, if found, replaces it with a specified
replacement string

NEW QUESTION 286
In the CUSTOMERS table, the CUST_CITY column contains the value 'Paris' for the
CUST_FIRST_NAME 'ABIGAIL'.
Evaluate the following query:



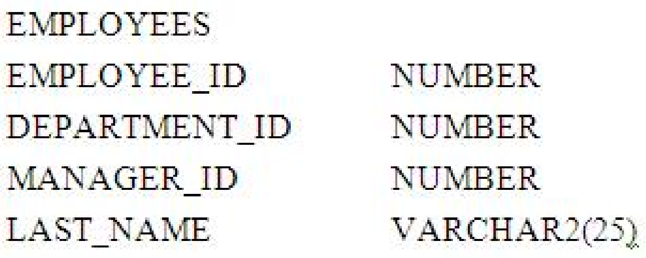
```
SQL> SELECT INITCAP(cust_first_name || ' ' ||
UPPER(SUBSTR(cust_city,-LENGTH(cust_city),2)))
FROM customers
WHERE cust_first_name = 'ABIGAIL';
```

What would be the outcome?

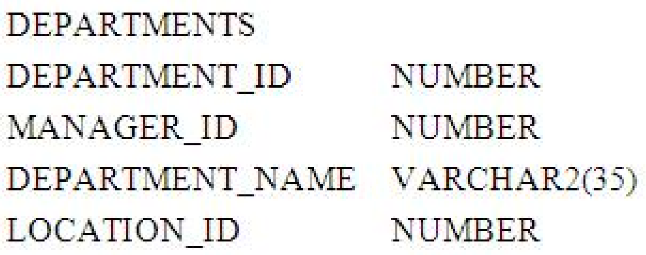
A. Abigail PA
B. Abigail Pa
C. Abigail IS
D. an error message

Answer: B

NEW QUESTION 290
View the Exhibits and examine the structures of the COSTS and PROMOTIONS tables.



EMPLOYEES	
EMPLOYEE_ID	NUMBER
DEPARTMENT_ID	NUMBER
MANAGER_ID	NUMBER
LAST_NAME	VARCHAR2(25)



DEPARTMENTS	
DEPARTMENT_ID	NUMBER
MANAGER_ID	NUMBER
DEPARTMENT_NAME	VARCHAR2(35)
LOCATION_ID	NUMBER

Evaluate the following SQL statement:
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_datepromo_ begin_date)));
What would be the outcome of the above SQL statement?

A. It displays prod IDs in the promo with the lowest cos
B. It displays prod IDs in the promos with the lowest cost in the same time interva
C. It displays prod IDs in the promos with the highest cost in the same time interva
D. It displays prod IDs in the promos with cost less than the highest cost in the same time interva

Answer: D

NEW QUESTION 294
Examine the structure of the EMPLOYEES and DEPARTMENTS tables:
You want to create a report displaying employee last names, department names, and locations. Which query should you use to create an equi-join?

A. SELECT last_name, department_name, location_id FROM employees , departments ;
B. SELECT employees.last_name, departments.department_name, departments.location_id FROM employees e, departments D WHERE e.department_id =d.department_id;

C. SELECT e.last_name, d.department_name, d.location_id FROM employees e, departments D WHERE manager_id =manager_id;
D. SELECT e.last_name, d.department_name, d.location_id FROM employees e, departments D WHERE e.department_id =d.department_id;

Answer: D

Explanation:
Equijoins are also called simple joins or inner joins. Equijoin involve primary key and foreign key.
Incorrect Answer: Athere is no join B invalid syntax Cdoes not involve the join in the primary and foreign key
Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 4-8

NEW QUESTION 296
View the Exhibit and examine the structure of the PRODUCTS table.

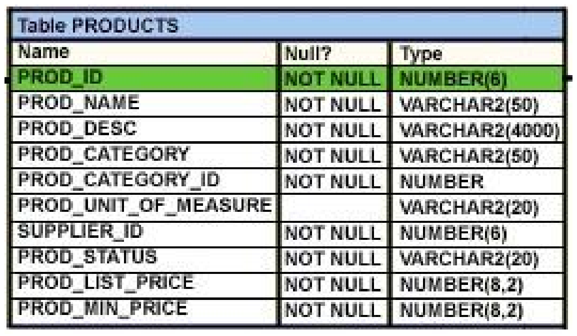


Table PRODUCTS

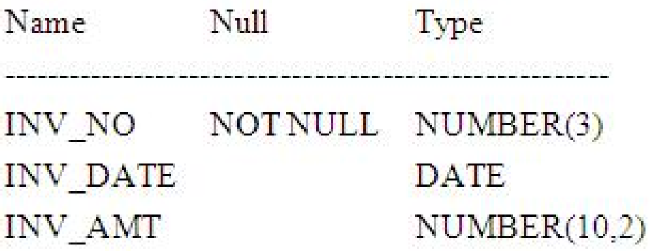
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER(6)
PROD_NAME	NOT NULL	VARCHAR2(50)
PROD_DESC	NOT NULL	VARCHAR2(4000)
PROD_CATEGORY	NOT NULL	VARCHAR2(50)
PROD_CATEGORY_ID	NOT NULL	NUMBER
PROD_UNIT_OF_MEASURE		VARCHAR2(20)
SUPPLIER_ID	NOT NULL	NUMBER(6)
PROD_STATUS	NOT NULL	VARCHAR2(20)
PROD_LIST_PRICE	NOT NULL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

You want to display the category with the maximum number of items. You issue the following query:
SQL>SELECT COUNT(*),prod_category_id FROM products GROUP BY prod_category_id HAVING COUNT(*) = (SELECT MAX(COUNT(*)) FROM products);
What is the outcome?

A. It executes successfully and gives the correct outpu
B. It executes successfully but does not give the correct outpu
C. It generates an error because the subquery does not have a GROUP BY claus
D. It generates an error because = is not valid and should be replaced by the IN operato

Answer: C

NEW QUESTION 297
View the Exhibit and examine the structure and data in the INVOICE table.



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

Which statements are true regarding data type conversion in expressions used in queries? (Choose all that apply.)

A. inv_amt ='0255982' : requires explicit conversion
B. inv_date > '01-02-2008' : uses implicit conversion
C. CONCAT(inv_amt,inv_date) : requires explicit conversion
D. inv_date = '15-february-2008' : uses implicit conversion
E. inv_no BETWEEN '101' AND '110' : uses implicit conversion

Answer: DE

Explanation:
In some cases, the Oracle server receives data of one data type where it expects data of a different data type. When this happens, the Oracle server can automatically convert the data to the expected data type. This data type conversion can be done implicitly by the Oracle server or explicitly by the user. Explicit data type conversions are performed by using the conversion functions. Conversion functions convert a value from one data type to another. Generally, the form of the function names follows the convention data type TO data type. The first data type is the input data type and the second data type is the output. Note: Although implicit data type conversion is available, it is recommended that you do the explicit data type conversion to ensure the reliability of your SQL statements.

NEW QUESTION 299
The EMPLOYEES table has these columns:
LAST NAMEVARCHAR2(35) SALARYNUMBER(8,2) HIRE_DATEDATE
Management wants to add a default value to the SALARY column. You plan to alter the table by using this SQL statement:
ALTER TABLE EMPLOYEES MODIFY (SALARY DEFAULT 5000);
What is true about your ALTER statement?

A. Column definitions cannot be altered to add DEFAULT value
B. A change to the DEFAULT value affects only subsequent insertions to the tabl
C. Column definitions cannot be altered at add DEFAULT values for columns with a NUMBER data typ
D. All the rows that have a NULL value for the SALARY column will be updated with the value 5000.

Answer: B

Explanation:
A change to the DEFAULT value affects only subsequent insertions to the table. Existing
rows will not be affected.
Incorrect Answers

A:Column definitions can be altered to add DEFAULT values.
C:Column definitions can be altered to add DEFAULT values. It works for columns with a NUMBER data type also.
D:A change to the DEFAULT value affects only subsequent insertions to the table. Existing rows will not be affected.
OCP Introduction to Oracle 9i: SQL Exam Guide, Jason Couchman, p. 219-224 Chapter 5: Creating Oracle Database Objects

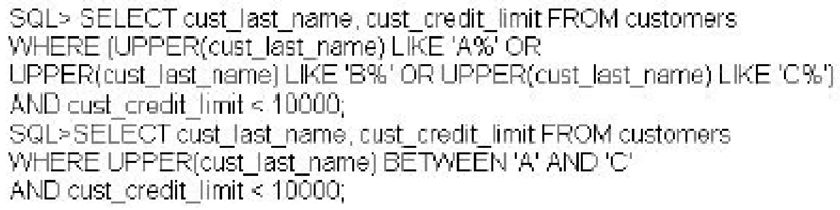
NEW QUESTION 304
In which two cases would you use an outer join? (Choose two.)

A. The tables being joined have NOT NULL column
B. The tables being joined have only matched dat
C. The columns being joined have NULL value
D. The tables being joined have only unmatched dat
E. The tables being joined have both matched and unmatched dat
F. Only when the tables have a primary key/foreign key relationshi

Answer: CE

Explanation:
You use an outer join to also see rows that do not meet the join condition.
Incorrect Answer: Ameet a join condition Bmeet a join condition Dmeet non join condition only Fdoes not take into consideration of primary key and foreign key relationship
Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 4-17

NEW QUESTION 306
View the Exhibit and examine the structure of the CUSTOMERS table.
You want to generate a report showing the last names and credit limits of all customers whose last names start with A, B, or C, and credit limit is below 10, 000.
Evaluate the following two queries:



```
SQL> SELECT cust_last_name, cust_credit_limit FROM customers
WHERE (UPPER(cust_last_name) LIKE 'A%' OR
UPPER(cust_last_name) LIKE 'B%' OR UPPER(cust_last_name) LIKE 'C%')
AND cust_credit_limit < 10000;
SQL>SELECT cust_last_name, cust_credit_limit FROM customers
WHERE UPPER(cust_last_name) BETWEEN 'A' AND 'C'
AND cust_credit_limit < 10000;
```

Which statement is true regarding the execution of the above queries?

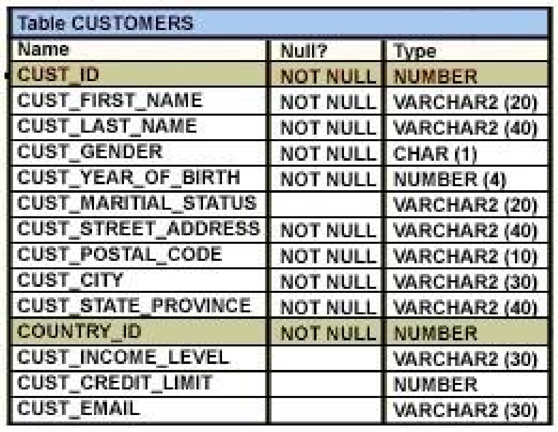


Table CUSTOMERS

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_MARITIAL_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. Only the first query gives the correct resul
B. Only the second query gives the correct resul
C. Both execute successfully and give the same resul
D. Both execute successfully but do not give the required resul

Answer: A

NEW QUESTION 311
......