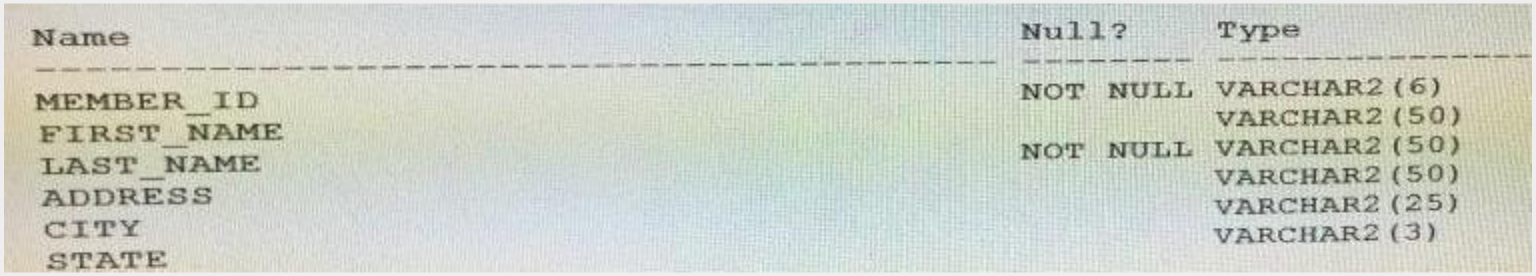
Question Results

Score 1 of 1

Question:

Examine the structure of the members table:



```
Name                                      Null?     Type
----------------------------------------- --------  -------------
MEMBER_ID                                 NOT NULL  VARCHAR2(6)
FIRST_NAME                                          VARCHAR2(50)
LAST_NAME                                 NOT NULL  VARCHAR2(50)
ADDRESS                                             VARCHAR2(50)
CITY                                                VARCHAR2(25)
STATE                                               VARCHAR2(3)
```

You want to display details of all members who reside in states starting with the letter A followed by exactly one character. Which SQL statement must you execute?

Response:

SELECT * FROM MEMBERS WHERE state LIKE 'A%';

SELECT * FROM MEMBERS WHERE state LIKE '%A_' ;

✓ SELECT * FROM MEMBERS WHERE state LIKE 'A_';

SELECT * FROM MEMBERS WHERE state LIKE 'A_%';

Score 1 of 1

Question:

You need to display the date 11-oct-2017 in words as 'Eleventh of October, Two Thousand Seventeen'.

Which SQL statement would give the required result?

Response:

SELECT TO_DATE (TO_CHAR ('11-oct-2017'), 'fmDdspth 'of Month, Year')) FROM DUAL;

SELECT TO_CHAR ('11-oct-2017', 'fmDdspth or Month, Year') FROM DUAL;

✓ SELECT TO_CHAR (TO_DATE ('11-oct-2017'), 'fmDdspth "of" Month, Year') FROM DUAL;

SELECT TO_CHAR (TO_DATE ('11-oct-2017'), 'fmDdthsp of Month, Year') FROM DUAL;

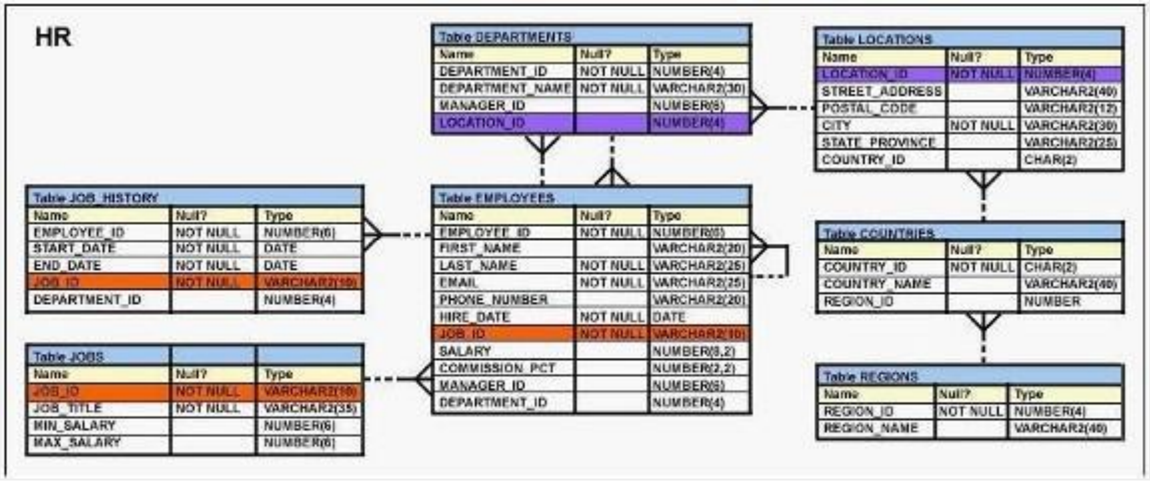
Score 1 of 1

Question:

View the Exhibit and examine the structure of the EMPLOYEES table. You want to display all employees and their managers having 100 as the MANAGER_ID.

You want the output in two columns: the first column would have the LAST_NAME of the managers and the second column would have LAST_NAME of the employees.

Which SQL statement would you execute?



HR

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

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

Table JOB_HISTORY		
Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
START_DATE	NOT NULL	DATE
END_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
DEPARTMENT_ID		NUMBER(4)

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

Table COUNTRIES		
Name	Null?	Type
COUNTRY_ID	NOT NULL	CHAR(2)
COUNTRY_NAME		VARCHAR2(40)
REGION_ID		NUMBER

Table JOBS		
Name	Null?	Type
JOB_ID	NOT NULL	VARCHAR2(10)
JOB_TITLE	NOT NULL	VARCHAR2(35)
MIN_SALARY		NUMBER(6)
MAX_SALARY		NUMBER(6)

Table REGIONS		
Name	Null?	Type
REGION_ID	NOT NULL	NUMBER(4)
REGION_NAME		VARCHAR2(40)

Response:

```
SELECT m.last_name "Manager", e.last_name "Employee"
FROM employees m JOIN employees e
WHERE m.employee_id = e.manager_id and AND e.manager_id = 100
```

```
SELECT m.last_name "Manager", e.last_name "Employee"
FROM employees m JOIN employees e
ON m.employee_id = e.manager_id
WHERE m.manager_id = 100;
```

✓

```
SELECT m.last_name "Manager", e.last_name "Employee"
FROM employees m JOIN employees e
ON m.employee_id = e.manager_id
WHERE e.manager_id = 100;
```

```
SELECT m.last_name "Manager", e.last_name "Employee"
FROM employees m JOIN employees e
ON e.employee_id = m.manager_id
WHERE m.manager_id = 100;
```

Score 1 of 1

Question:

View the Exhibit and examine the structure of CUSTOMERS table.

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?

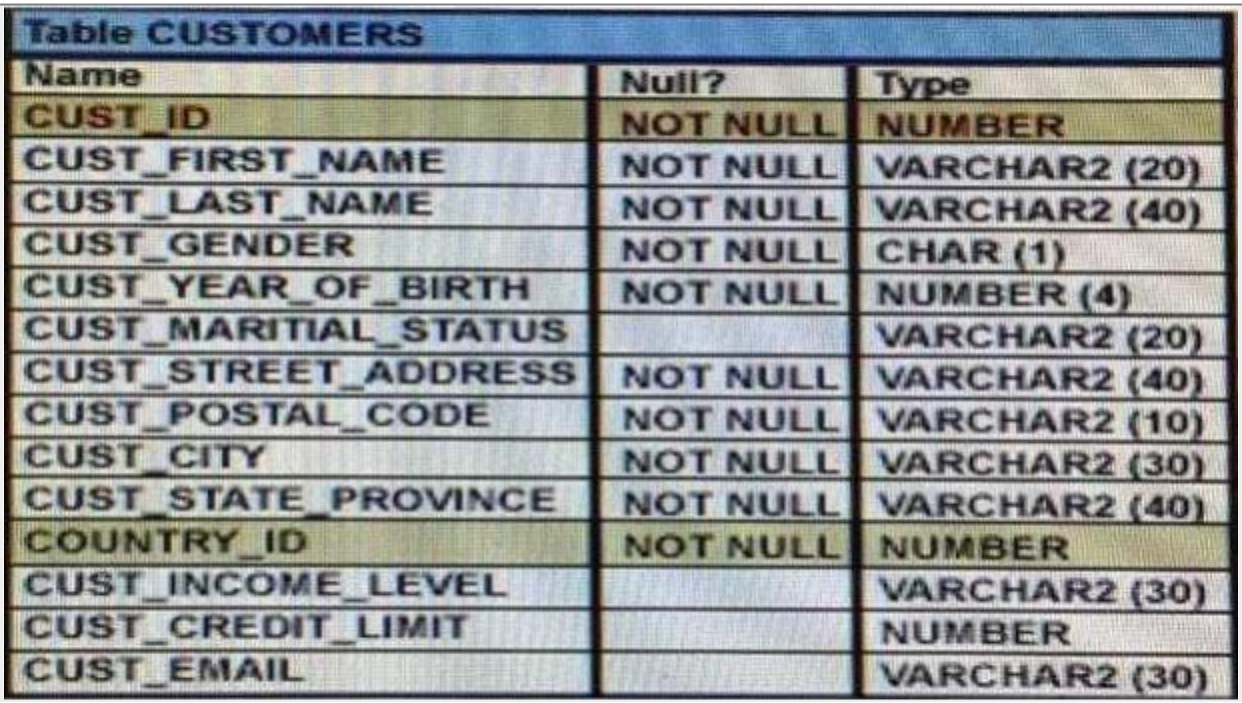


Table CUSTOMERS		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2 (20)
CUST_LAST_NAME	NOT NULL	VARCHAR2 (40)
CUST_GENDER	NOT NULL	CHAR (1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER (4)
CUST_MARITIAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT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

Response:

```
SELECT TO_CHAR (NVL(cust_credit_limit *.15), 'Not Available') "NEW CREDIT" FROM customers;

SELECT NVL(cust_credit_limit), 'Not Available') "NEW CREDIT" FROM customers;

SELECT NVL(cust_credit_limit *.15), 'Not Available') "NEW CREDIT" FROM customers;
```

✓ SELECT NVL (TO CHAR(cust_credit_limit *.15), 'Not Available') "NEW CREDIT" FROM customers;

Score 1 of 1

Question:

Which two statements are true regarding subqueries?
(Choose two.)

Response:

There is no limit on the number of subquery levels in the WHERE clause of a SELECT statement.

✓ A subquery can appear on either side of a comparison operator.

A subquery can be used only in SQL query statements.

✓ A subquery can retrieve zero or more rows.

Only two subqueries can be placed at one level.

Score 1 of 1

Question:

When you’re looking for a particular bit of data and you’re not sure where in the data dictionary it might be, a good starting point is:
(Choose the best answer.)

Response:

SELECT * FROM V$DATABASE;

SELECT * FROM GV_$START_HERE;

✓ SELECT * FROM DICTIONARY;

SELECT * FROM V$RESERVED_WORDS;

Score 1 of 1

Question:

You need to get information about columns in a table you do not own, nor do you have privileges to it. Which view can you query to get this information?

Response:

✓ DBA_TAB_COLUMNS

Can’t be done

ALL_COLUMNS

ALL_TAB_COLUMNS

Score 1 of 1

Question:

User account MUSKIE owns a table called CBAY. Which of the following statements can be executed by MUSKIE and enable user ONEILL to execute UPDATE statements on the CBAY table?
(Choose three.)

Response:

GRANT ALL TO ONEILL;

✓ GRANT INSERT, UPDATE ON CBAY TO ONEILL;

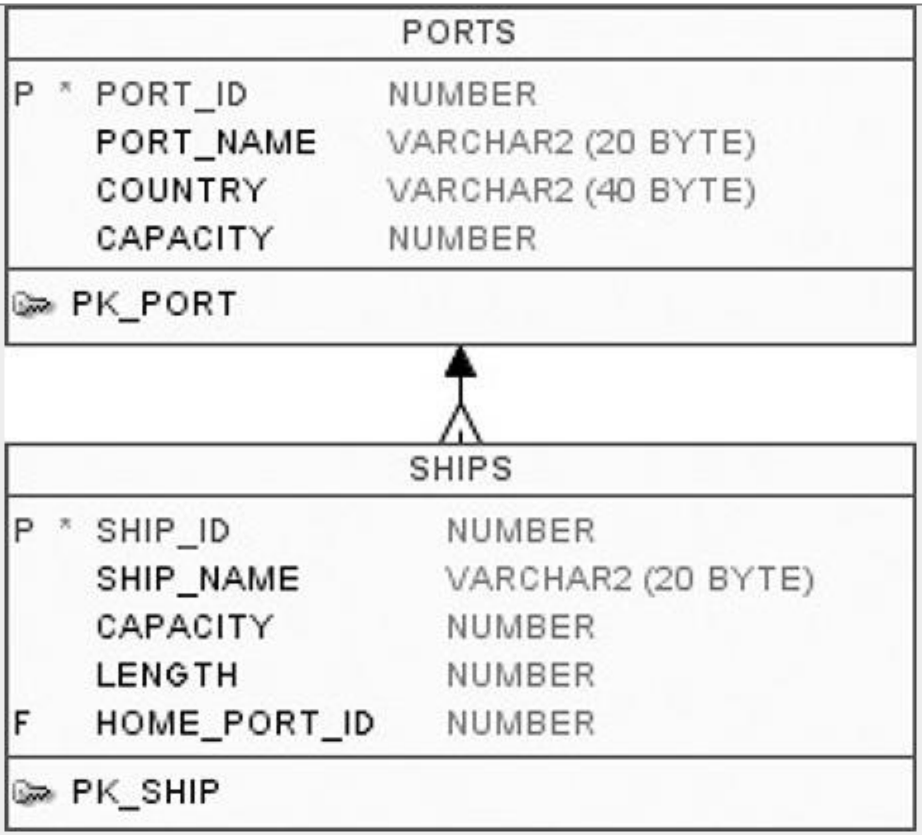
✓ GRANT ALL PRIVILEGES TO ONEILL;

✓ GRANT ALL ON CBAY TO ONEILL;

Score 1 of 1

Question:

Review the PORTS and SHIPS tables shown. Then review the following SQL code:



PORTS

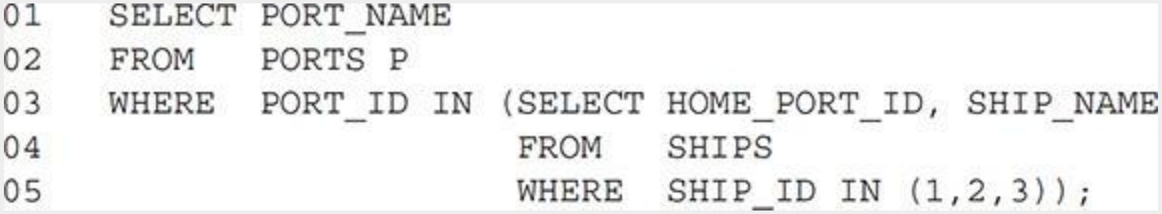
P	*	PORT_ID	NUMBER
		PORT_NAME	VARCHAR2 (20 BYTE)
		COUNTRY	VARCHAR2 (40 BYTE)
		CAPACITY	NUMBER

PK_PORT

SHIPS

P	*	SHIP_ID	NUMBER
		SHIP_NAME	VARCHAR2 (20 BYTE)
		CAPACITY	NUMBER
		LENGTH	NUMBER
F		HOME_PORT_ID	NUMBER

PK_SHIP



```
01   SELECT PORT_NAME
02   FROM   PORTS P
03   WHERE  PORT_ID IN (SELECT HOME_PORT_ID, SHIP_NAME
04                      FROM   SHIPS
05                      WHERE  SHIP_ID IN (1,2,3));
```

Which of the following is true of this statement?

Response:

The statement will fail with a syntax error because of line 5.

✓ The statement will fail with a syntax error because of line 3.

None of the above.

Whether the statement fails depends on how many rows are returned by the subquery in lines 3 through 5.

Score 1 of 1

Question:

Which two statements are true regarding constraints?

(Choose two.)

Response:

A constraint is enforced only for the INSERT operation on a table.

✓ A constraint can be disabled even if the constraint column contains data.

✓ A column with the UNIQUE constraint can contain NULL.

A foreign key cannot contain NULL values.

All the constraints can be defined at the column level as well as the table level

Score 1 of 1

Question:

If an ALTER TABLE . . . DROP COLUMN statement is executed against an underlying table upon which a view is based, the status of that view in the data dictionary changes to:

Response:

FLAG

COMPILE

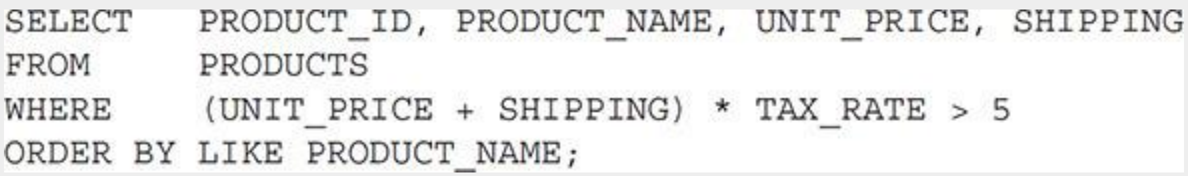
ALTERED

✓ INVALID

Score 1 of 1

Question:

Review this SELECT statement:



```
SELECT   PRODUCT_ID, PRODUCT_NAME, UNIT_PRICE, SHIPPING
FROM     PRODUCTS
WHERE    (UNIT_PRICE + SHIPPING) * TAX_RATE > 5
ORDER BY LIKE PRODUCT_NAME;
```

Assume all table and column references exist in the database. What can be said of this SELECT statement?

Response:

✓ The statement will fail to execute because the ORDER BY clause includes the word LIKE.

The statement will execute but not sort because the ORDER BY clause is wrong.

The statement will execute successfully and as intended.

None of the above.

Score 1 of 1

Question:

Assume you have a table ITEMS that includes a column STATUS. Which of the following statements is syntactically correct?
(Choose all that apply.)

Response:

✓ SELECT * FROM ITEMS FETCH NEXT 20 ROWS WITH TIES;

✓ SELECT * FROM ITEMS ORDER BY STATUS FETCH NEXT 20 ROWS WITH TIES;

SELECT * FROM ITEMS FETCH NEXT 20 % ROWS ONLY;

✓ SELECT * FROM ITEMS FETCH NEXT 20 PERCENT ROWS ONLY;

Score 1 of 1

Question:

Which of the following is a true statement?

Response:

If a query returns multiple rows, it may not be used as a subquery for a SELECT statement that uses a GROUP BY clause.

✓ A SELECT statement with a GROUP BY may use a subquery to return a value to the outermost WHERE clause.

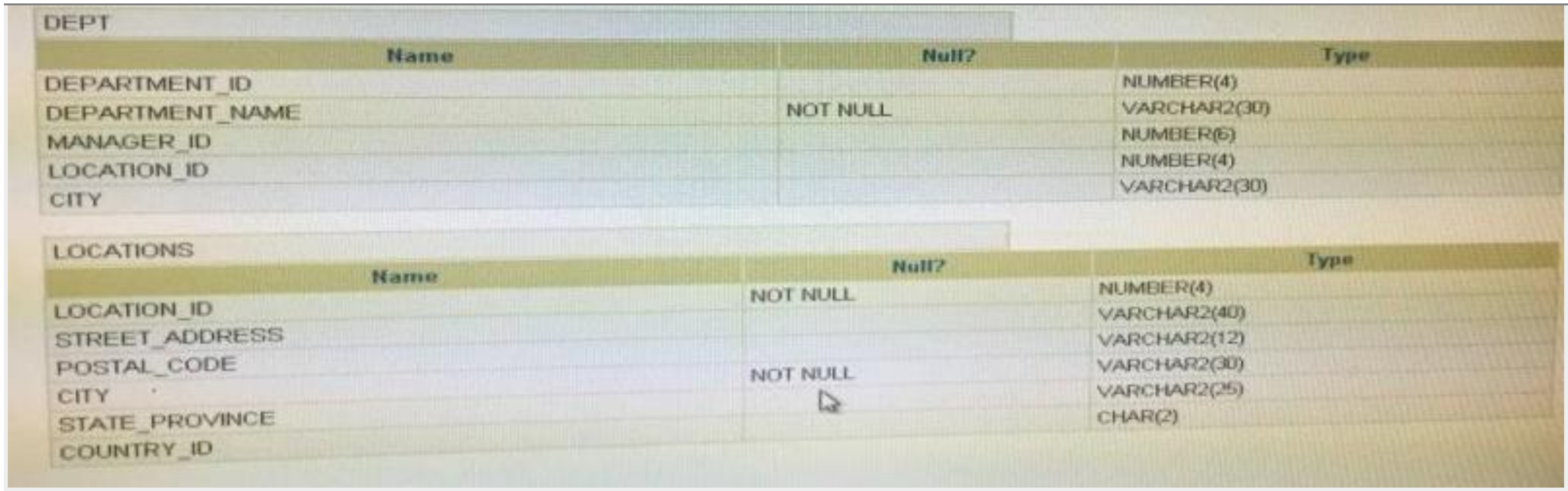
If a SELECT includes a GROUP BY clause, then any subquery used within the SELECT must also have a GROUP BY clause.

The only form of subquery permitted with a GROUP BY clause is a correlated subquery.

Score 1 of 1

Question:

View the Exhibit and examine the descriptions of the DEPT and LOCATIOMS tables.



DEPT

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

LOCATIONS

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

You want to update the CITY column of the DEPT table for all the rows with the corresponding value in the CITY column of the LOCATIONS table for each department.

Which SQL statement would you execute to accomplish the task?

Response:

```
UPDATE dept d
SET city = ANY (SELECT city FROM locations l);
```

✓

```
UPDATE dept d
SET city = (SELECT city FROM locations l
WHERE d.location_id = l.location_id);
```

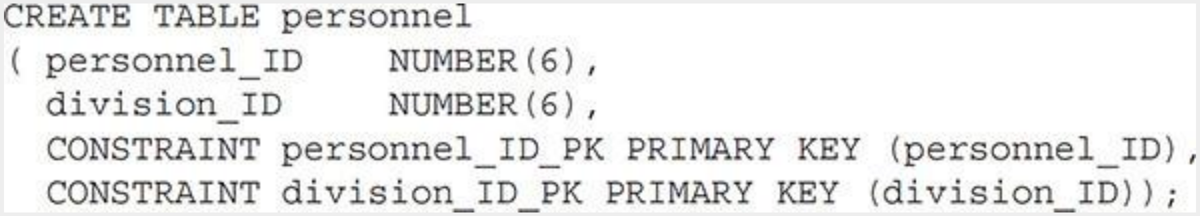
```
UPDATE dept d
SET city = (SELECT city FROM locations l) WHERE d.location_id = l.location_id;
```

```
UPDATE dept d
SET city = ALL (SELECT city FROM locations l
WHERE d.location_id = l.location_id);
```

Score 1 of 1

Question:

Review the following SQL statement:



```
CREATE TABLE personnel
( personnel_ID    NUMBER(6),
  division_ID     NUMBER(6),
  CONSTRAINT personnel_ID_PK PRIMARY KEY (personnel_ID),
  CONSTRAINT division_ID_PK PRIMARY KEY (division_ID));
```

Assume there is no table already called PERSONNEL in the database. What will be the result of an attempt to execute the preceding SQL statement?

Response:

The statement will successfully create the table and two primary keys.

✓ The statement will fail because you cannot create two primary key constraints on the table.

The statement will successfully create a single table and one composite primary key consisting of two columns.

The statement will successfully create the table and the first primary key but not the second.

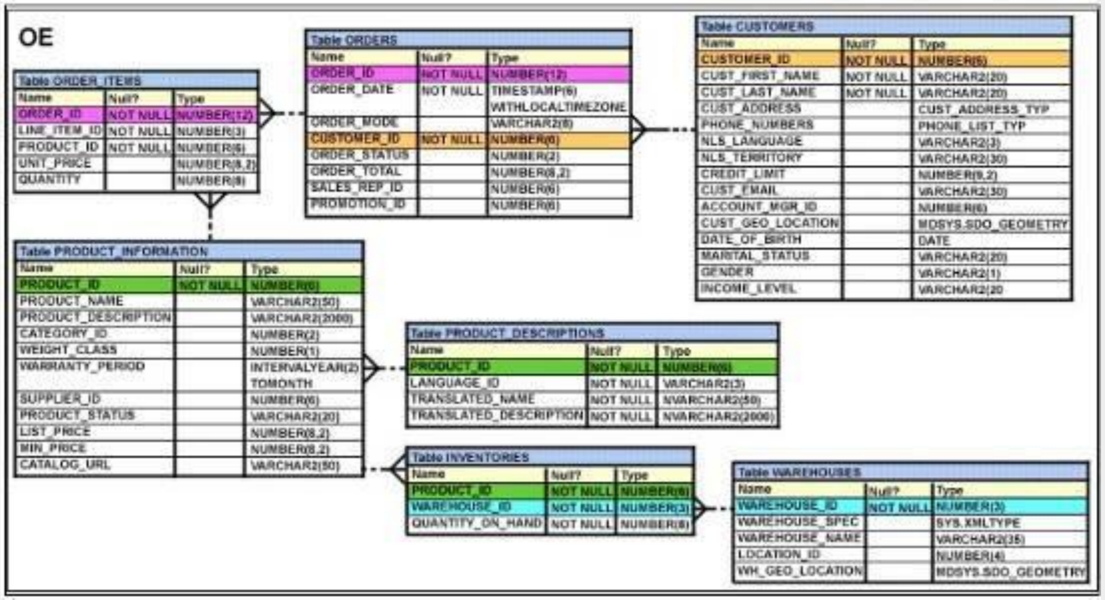
Score 1 of 1

Question:

View the Exhibit and examine the structure of the ORDERS table. The ORDER_ID column is the PRIMARY KEY in the ORDERS table. Evaluate the following CREATE TABLE command:

```
CREATE TABLE new_orders(ord_id, ord_date DEFAULT SYSDATE, cus_id) AS SELECT order_id.order_date,customer_id FROM orders;
```

Which statement is true regarding the above command?



OE

Table ORDER_ITEMS

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

Table ORDERS

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

Table CUSTOMERS

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

Table PRODUCT_INFORMATION

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

Table PRODUCT_DESCRIPTIONS

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

Table INVENTORIES

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

Table WAREHOUSES

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

Response:

The NEW_IDRDERS table would get created and all the constraints defined on the specified columns in the ORDERS table would be passed to the new table.

✓ The NEW_IDRDERS table would get created and only the NOT NULL constraint defined on the specified columns would be passed to the new table.

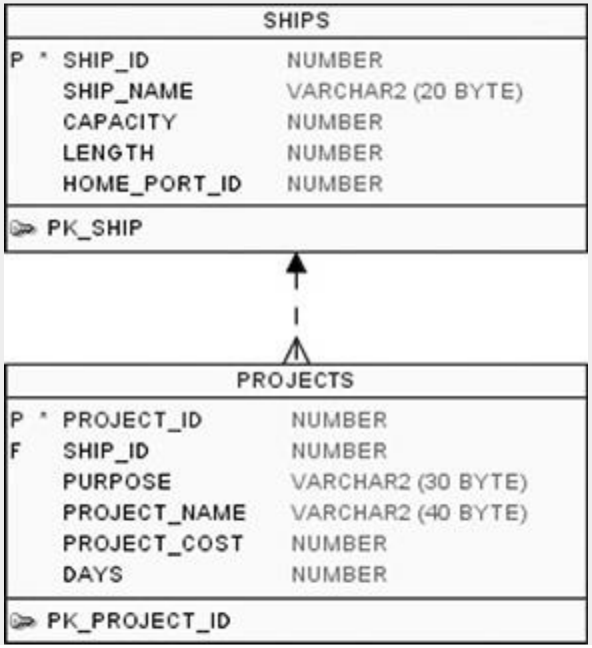
The NEW_IDRDERS table would not get created because the column names in the CREATE TABLE command and the SELECT clause do not match.

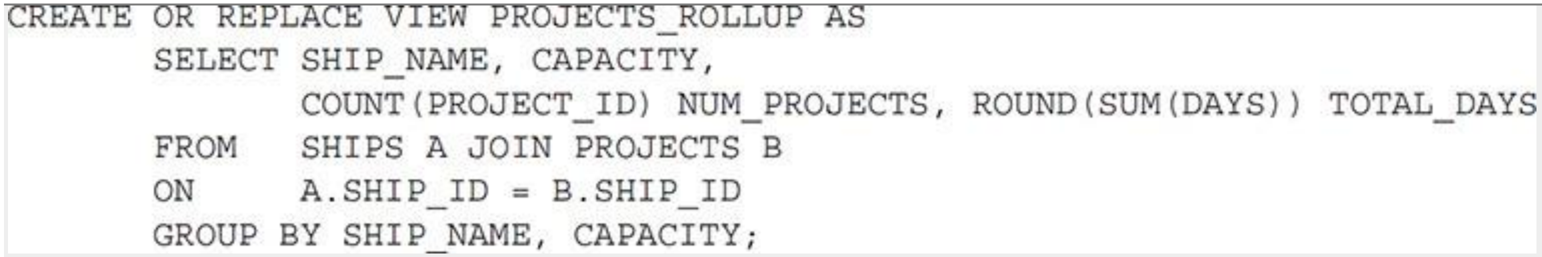
The NEW_IDRDERS table would not get created because the DEFAULT value cannot be specified in the column definition.

Score 1 of 1

Question:

Review the illustration and the following SQL code:





```
CREATE OR REPLACE VIEW PROJECTS_ROLLUP AS
       SELECT SHIP_NAME, CAPACITY,
              COUNT(PROJECT_ID) NUM_PROJECTS, ROUND(SUM(DAYS)) TOTAL_DAYS
       FROM   SHIPS A JOIN PROJECTS B
       ON     A.SHIP_ID = B.SHIP_ID
       GROUP BY SHIP_NAME, CAPACITY;
```

What can be said of this code?

Response:

After the view is created, a valid SELECT and valid INSERT statement will work on the PROJECTS_ROLLUP view.

The attempt to create the view will fail because you cannot create a VIEW with a SELECT statement that uses a GROUP BY clause.

✓ After the view is created, a valid SELECT statement will work on the PROJECTS_ROLLUP view, but an INSERT will not.

The attempt to create the view will fail because you cannot create a VIEW with a SELECT statement that is a join.

Score 1 of 1

Question:

You execute the following commands:

SQL > DEFINE hiredate = '01-APR-2011'

SQL >SELECT employee_id, first_name, salary

```
FROM employees
WHERE hire_date > '&hiredate'
AND manager_id > &mgr_id;
```

For which substitution variables are you prompted for the input?

Response:

both the substitution variables ''hiredate' and 'mgr_id'.

✓ only 'mgr_id'

none, because no input required

only hiredate'

Score 1 of 1

Question:

Which of the following can a correlated subquery be used in?
(Choose three.)

Response:

The FROM clause of a DELETE statement

✓ The SET clause of an UPDATE statement

✓ The WHERE clause of an UPDATE statement

✓ The WHERE clause of a DELETE statement

Score 1 of 1

Question:

Which statement is true about SQL query processing in an Oracle database instance?

Response:

During row source generation, rows that satisfy the query are retrieved from the database and stored in memory.

During parsing, a SQL statement containing literals in the WHERE clause that has been executed by any session and which is cached in memory, is always reused for the current execution.

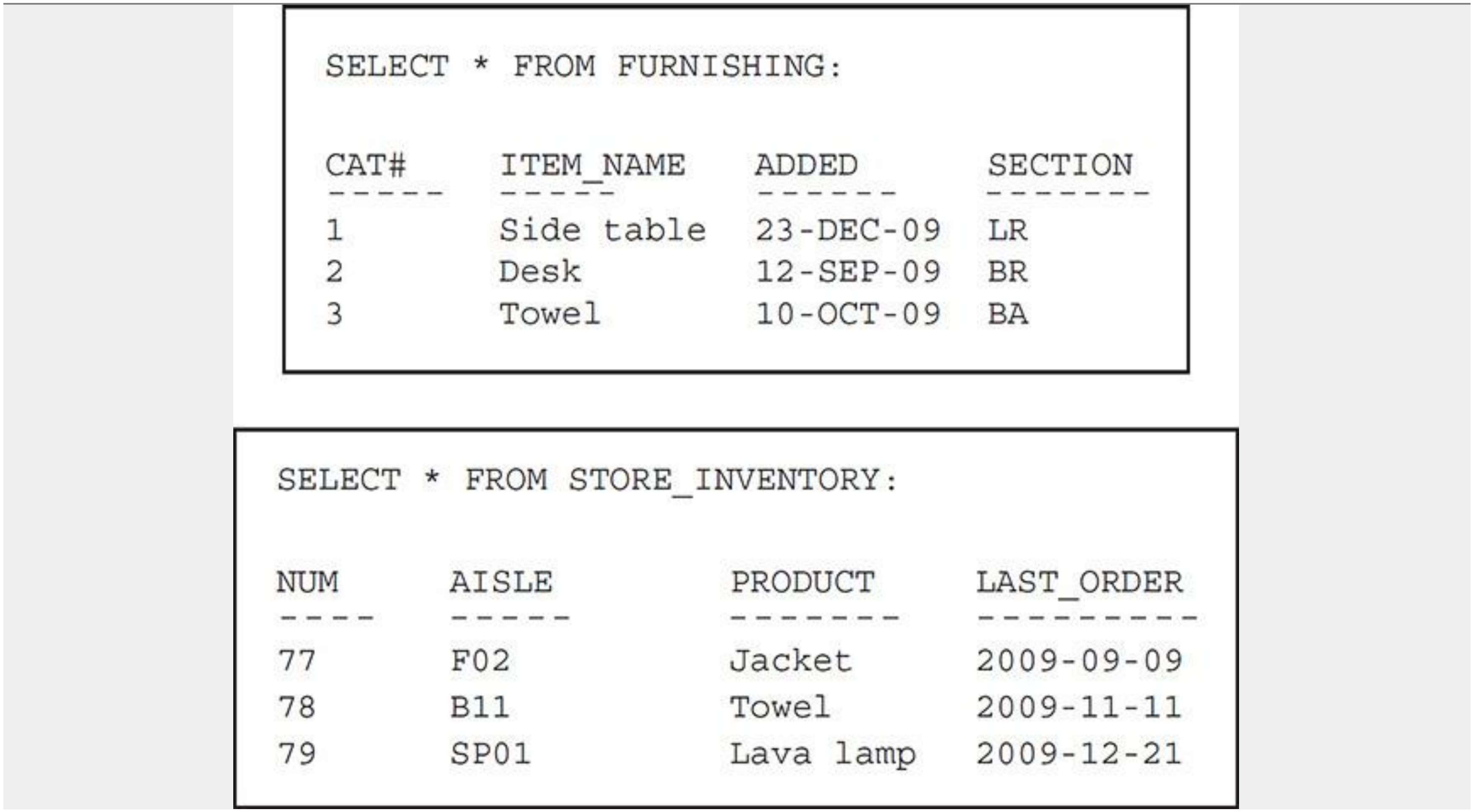
✓ During executing, the oracle server may read data from storage if the required data is not already in memory.

During optimization, execution plans are formulated based on the statistics gathered by the database instance, and the lowest cost plan is selected for execution.

Score 1 of 1

Question:

Review the first two illustrations and then review this SQL code:

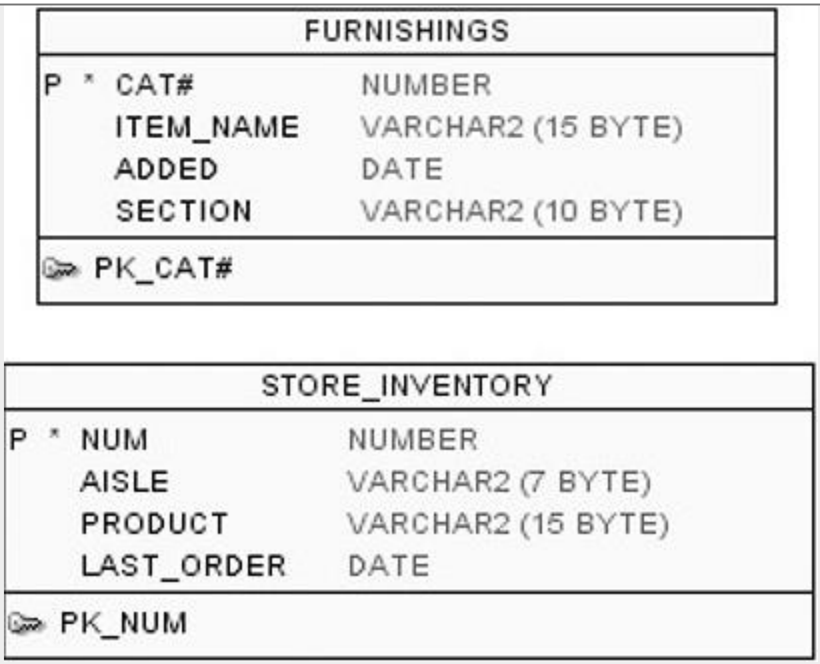


```
SELECT * FROM FURNISHING:

CAT#      ITEM_NAME    ADDED       SECTION
-----     -----        ------      --------
1         Side table   23-DEC-09   LR
2         Desk         12-SEP-09   BR
3         Towel        10-OCT-09   BA
```

```
SELECT * FROM STORE_INVENTORY:

NUM       AISLE          PRODUCT     LAST_ORDER
----      ------         --------    ----------
77        F02            Jacket      2009-09-09
78        B11            Towel       2009-11-11
79        SP01           Lava lamp   2009-12-21
```



FURNISHINGS

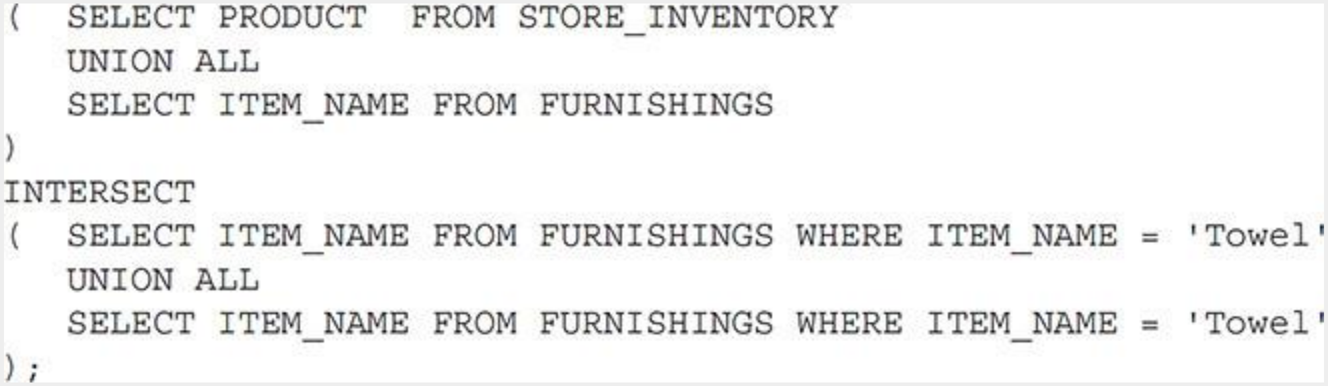
P	*	CAT#	NUMBER
		ITEM_NAME	VARCHAR2 (15 BYTE)
		ADDED	DATE
		SECTION	VARCHAR2 (10 BYTE)

PK_CAT#

STORE_INVENTORY

P	*	NUM	NUMBER
		AISLE	VARCHAR2 (7 BYTE)
		PRODUCT	VARCHAR2 (15 BYTE)
		LAST_ORDER	DATE

PK_NUM



```
(   SELECT PRODUCT  FROM STORE_INVENTORY
    UNION ALL
    SELECT ITEM_NAME FROM FURNISHINGS
)
INTERSECT
(   SELECT ITEM_NAME FROM FURNISHINGS WHERE ITEM_NAME = 'Towel'
    UNION ALL
    SELECT ITEM_NAME FROM FURNISHINGS WHERE ITEM_NAME = 'Towel'
);
```

How many rows will result from this code?

Response:

6

2

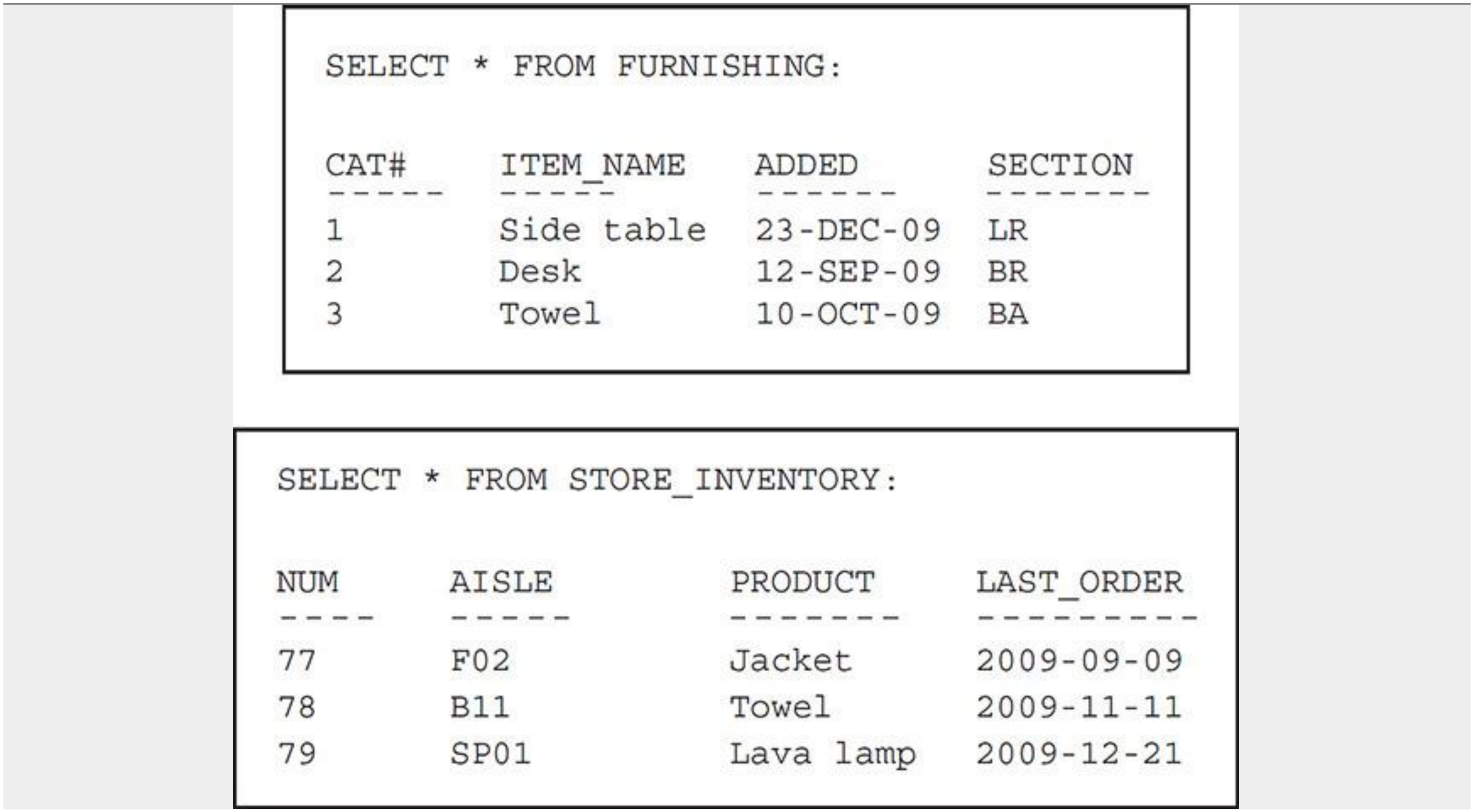
✓ 1

4

Score 1 of 1

Question:

Review the first two illustrations; then review this SQL code:

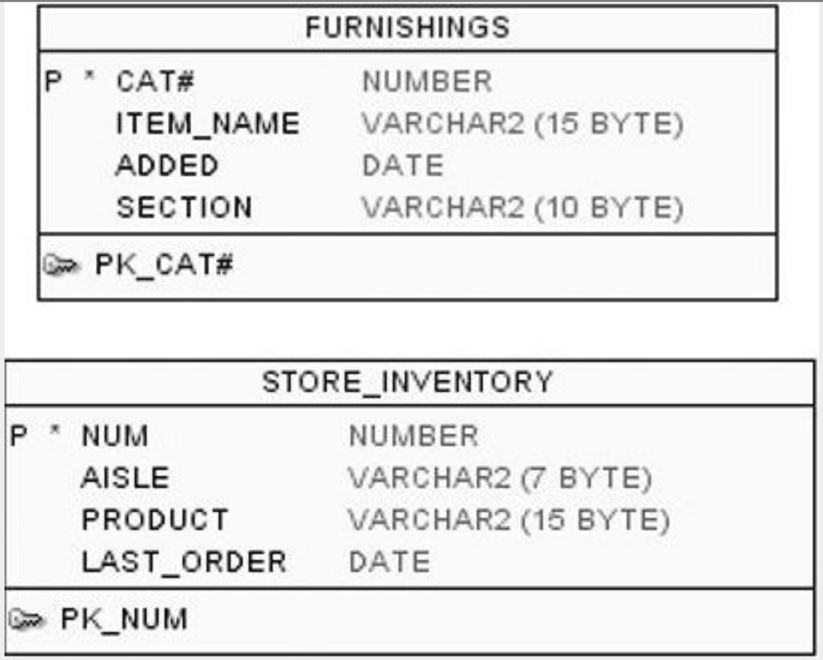


```
SELECT * FROM FURNISHING:

CAT#      ITEM_NAME    ADDED       SECTION
-----     -----        ------      -------
1         Side table   23-DEC-09   LR
2         Desk         12-SEP-09   BR
3         Towel        10-OCT-09   BA
```

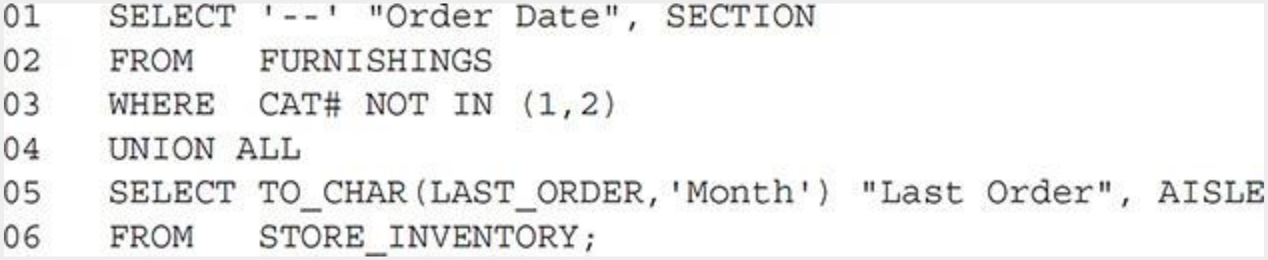
```
SELECT * FROM STORE_INVENTORY:

NUM      AISLE        PRODUCT      LAST_ORDER
----     -----        -------      ----------
77       F02          Jacket       2009-09-09
78       B11          Towel        2009-11-11
79       SP01         Lava lamp    2009-12-21
```



FURNISHINGS		
P *	CAT#	NUMBER
	ITEM_NAME	VARCHAR2 (15 BYTE)
	ADDED	DATE
	SECTION	VARCHAR2 (10 BYTE)
	PK_CAT#	

STORE_INVENTORY		
P *	NUM	NUMBER
	AISLE	VARCHAR2 (7 BYTE)
	PRODUCT	VARCHAR2 (15 BYTE)
	LAST_ORDER	DATE
	PK_NUM	



```
SELECT '--' "Order Date", SECTION
FROM   FURNISHINGS
WHERE  CAT# NOT IN (1,2)
UNION ALL
SELECT TO_CHAR(LAST_ORDER,'Month') "Last Order", AISLE
FROM   STORE_INVENTORY;
```

Which of the following are valid ORDER BY clauses for this query?
(Choose two.)

Response:

ORDER BY "Last Order"

ORDER BY AISLE

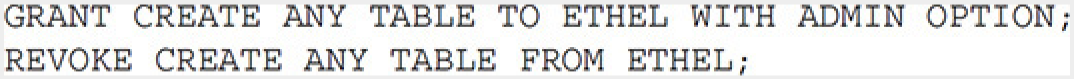
✓ ORDER BY 1

✓ ORDER BY SECTION

Score 1 of 1

Question:

You are logged in to user account FRED and have been tasked with granting privileges to the user account ETHEL. You execute the following SQL statements:



```
GRANT CREATE ANY TABLE TO ETHEL WITH ADMIN OPTION;
REVOKE CREATE ANY TABLE FROM ETHEL;
```

Assuming both statements execute successfully, what is the result?

Response:

ETHEL has the system privilege CREATE ANY TABLE because the WITH ADMIN OPTION clause wasn’t included in the REVOKE statement.

ETHEL no longer has the system privilege CREATE ANY TABLE but still has the right to grant the CREATE ANY TABLE system privilege to any other user since the WITH ADMIN OPTION clause was omitted. Furthermore, ETHEL may grant the CREATE ANY TABLE privilege to herself because of the WITH ADMIN OPTION clause.

ETHEL no longer has the system privilege CREATE ANY TABLE but still has the right to grant the CREATE ANY TABLE system privilege to any other user, since the WITH

ADMIN OPTION clause was omitted from the REVOKE statement. However, ETHEL may not grant the CREATE ANY TABLE privilege to herself.

✓ ETHEL does not have the system privilege CREATE ANY TABLE or the right to grant the CREATE ANY TABLE system privilege to any other user.

Score 1 of 1

Question:

You have two tables. One table is called CUSTOMERS. Another is called PURCHASES, and it records a list of customer transactions.

Your goal is to create a SELECT statement that will show all customers by last name in alphabetical order, along with any purchases they may have made in the past two weeks, as recorded in the PURCHASES table.

It’s possible that many customers have made no purchases in the past two weeks, but you still want them included in the output. Both tables contain a column called CUSTOMER_ID.

Which of the following will be true of the SELECT statement you’ll need to create?
(Choose two.)

Response:

It will be an inner join.

✓ It will be an outer join.

✓ It will be an equijoin.

It will be a cross-join.

Score 1 of 1

Question:

What is one of the purposes of DDL?
(Choose the best answer.)

Response:

✓ Remove existing data from a database table

None of the above

Issue privileges to users

Query data from a given table

Score 1 of 1

Question:

See the Exhibit and Examine the structure of the CUSTOMERS table:

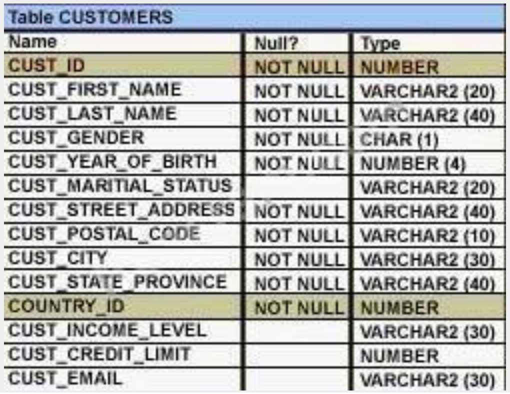


Table CUSTOMERS

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?

Response:

✓ SELECT NVL(TO_CHAR(cust_credit_limit*1.15),'Not Available') "NEW CREDIT" FROM customers;

SELECT TO_CHAR(NVL(cust_credit_limit*1.15,'Not Available')) "NEW CREDIT" FROM customers;

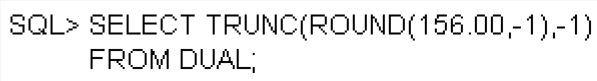
SELECT NVL(cust_credit_limit,'Not Available')*1.15 "NEW CREDIT" FROM customers;

SELECT NVL(cust_credit_limit*1.15,'Not Available') "NEW CREDIT" FROM customers;

Score 1 of 1

Question:

Evaluate the following SQL query;



```
SQL> SELECT TRUNC(ROUND(156.00,-1),-1)
     FROM DUAL;
```

What would be the outcome?

Response:

✓ 160

100

16

150

200

Score 1 of 1

Question:

The difference between an INNER and an OUTER join is:

Response:

The OUTER join relates a table to tables in other user accounts; the INNER does not.

The INNER join relates a table to itself; the OUTER join relates a table to other tables.

The INNER runs on data inside the table; the OUTER runs on data outside of the table.

✓ The INNER join displays rows that match in all joined tables; the OUTER join shows data that doesn’t necessarily match.

Score 1 of 1

Question:

The WITH clause can be used to name a subquery. Which of the following is also true?
(Choose two.)

Response:

The name of the subquery can be invoked from within the subquery that is named.

✓ The name of the subquery can be used in the SELECT statement following the WITH clause.

✓ The name of the subquery can be joined to other tables in the SELECT statement following the WITH clause.

The name of the subquery is stored in the database by the WITH statement and can be referenced by other SQL statements in later sessions.

Score 1 of 1

Question:

Which of the following can a subquery be used in?
(Choose all that apply.)

Response:

✓ A WHERE clause in a SELECT statement

✓ An INSERT statement's SELECT

✓ An inline view

A GRANT statement

Score 1 of 1

Question:

Which two statements are true regarding multiple-row subqueries?

(Choose two.)

Response:

✓ They should not be used with the NOT IN operator in the main query if NULL is likely to be a part of the result of the subquery.

They use the < ALL operator to imply less than the maximum.

✓ They can contain group functions.

They always contain a subquery within a subquery.

They can be used to retrieve multiple rows from a single table only.

Score 1 of 1

Question:

Which two statements are true regarding the execution of the correlated subqueries?
(Choose two.)

Response:

The outer query executes only once for the result returned by the inner query.

The nested query executes first and then the outer query executes.

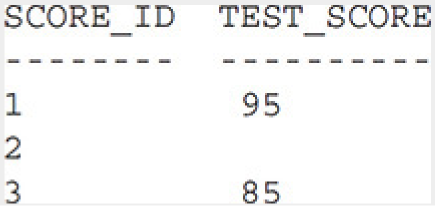
✓ The nested query executes after the outer query returns the row.

✓ Each row returned by the outer query is evaluated for the results returned by the inner query.

Score 1 of 1

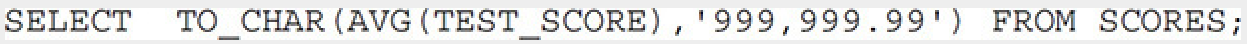
Question:

Review the following data listing from a table SCORES:



```
SCORE_ID  TEST_SCORE
--------  ----------
1          95
2
3          85
```

Now consider the following query:



```
SELECT  TO_CHAR(AVG(TEST_SCORE),'999,999.99') FROM SCORES;
```

What will be the result of this query?

Response:

It will result in an execution error.

60.00.

It will result in a syntax error because of the TO_CHAR function.

✓ 90.00.

Score 1 of 1

Question:

Which two statements are true about sequences crated in a single instance Oracle database?

Response:

When the database instance shuts down abnormally, sequence numbers that have been cached but not used are available again when the instance is restarted.

✓ When the MAXVALUE limit for a sequence is reached, it can be increased by using the ALTER SEQUENCE statement.

DELETE <sequencename> would remove a sequence from the database.

✓ CURRVAL is used to refer to the most recent sequence number that has been generated for a particular sequence.

The numbers generated by an explicitly defined sequence can only be used to insert data in one table.

Score 1 of 1

Question:

When does a transaction complete?

(Choose all that apply.)

Response:

When a PL/SQL anonymous block is executed

✓ When a ROLLBACK command is executed

✓ When a TRUNCATE statement is executed after the pending transaction

When a DELETE statement is executed

✓ When a data definition language statement is executed