Question Results

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

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

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

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

Score 0 of 1

Question:

Your user account owns an updatable view, BACKLOG, which is based on the table PROJECTS. You are tasked to give SELECT and UPDATE capabilities to another user account named MARINO.

Currently, MARINO has no privileges on either the table or the view. You want for MARINO to have the ability to grant SELECT on the view to other users as well.

Examine the following SQL code:

```
GRANT SELECT ON BACKLOG TO MARINO WITH GRANT OPTION;
GRANT UPDATE ON BACKLOG TO MARINO;
```

Which of the following statements is true?

Response:

✗ The statements will execute successfully, but MARINO will not be able to SELECT from the view because the PROJECTS table has not been granted to MARINO.

✓ The statements will execute successfully and perform as intended.

The statements will execute successfully, and MARINO will be able to SELECT from the view but not UPDATE the view.

The statements will fail, and MARINO will not be able to use the view.

Score 1 of 1

Question:

A multitable INSERT statement:

Response:

Will create any tables in which it attempts to INSERT but that do not yet exist

✓ Can use conditional logic

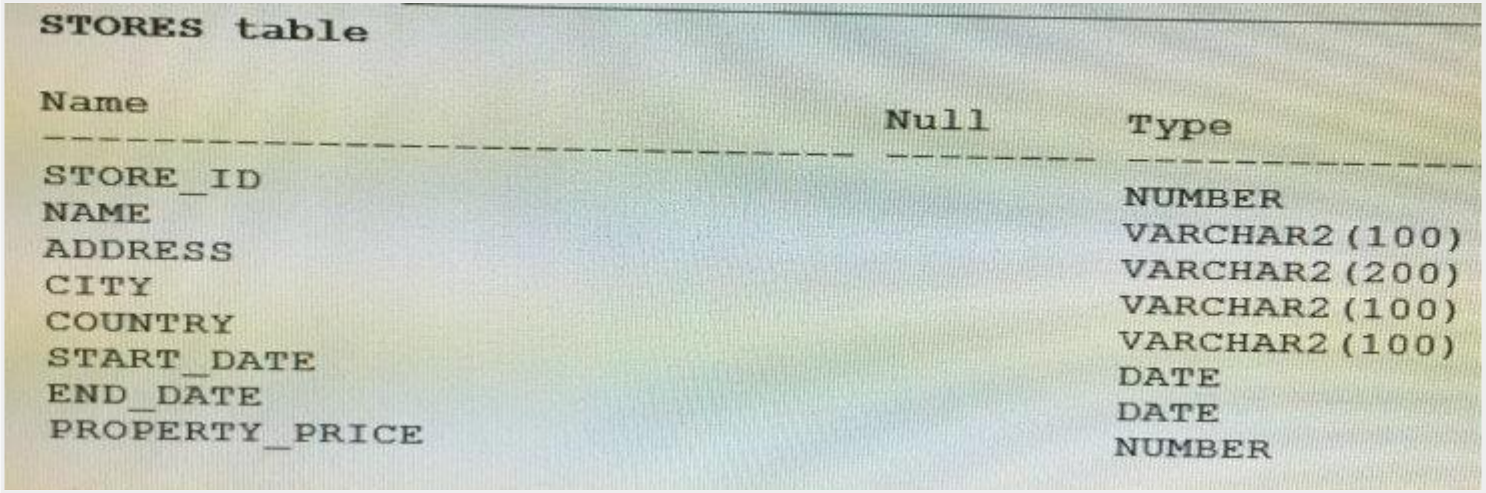
Can accomplish tasks that cannot otherwise be done in any combination of SQL statements

Is capable of inserting rows into nonupdatable views

Score 1 of 1

Question:

View the Exhibit and examine the structure of the stores table.

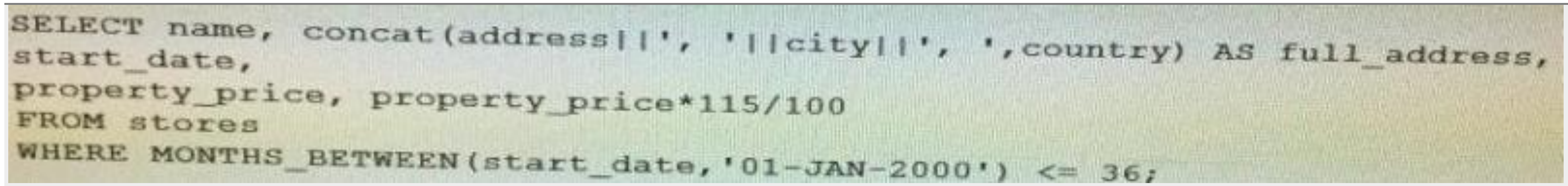


```
STORES table

Name                              Null      Type
--------------------------------- --------- ---------------
STORE_ID                                    NUMBER
NAME                                        VARCHAR2(100)
ADDRESS                                     VARCHAR2(200)
CITY                                        VARCHAR2(100)
COUNTRY                                     VARCHAR2(100)
START_DATE                                  DATE
END_DATE                                    DATE
PROPERTY_PRICE                              NUMBER
```

You want to display the name of the store along with the address, START_DATE, PROPERTV_PRICE, and the projected property price, which is 115% of the property price. The stores displayed must have START_DATE in the range of 36 months starting from 01- Jan-2000 and above.

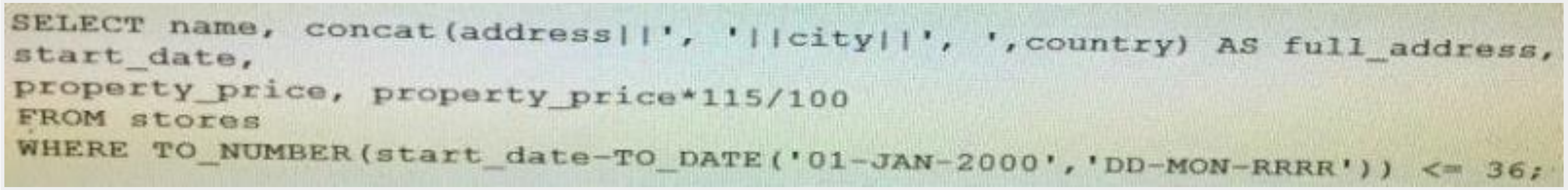
Which SQL statement would get the desired output?

A)
Exhibit



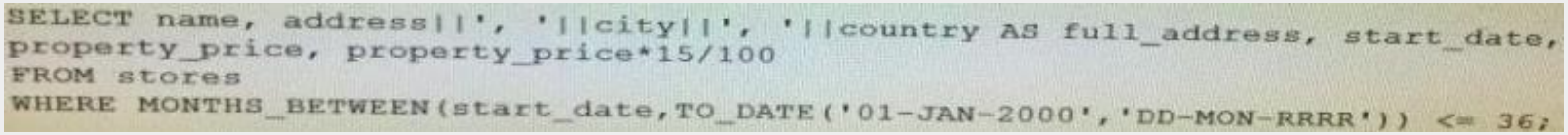
```
SELECT name, concat(address||', '||city||', ',country) AS full_address,
start_date,
property_price, property_price*115/100
FROM stores
WHERE MONTHS_BETWEEN(start_date,'01-JAN-2000') <= 36;
```

B)
Exhibit



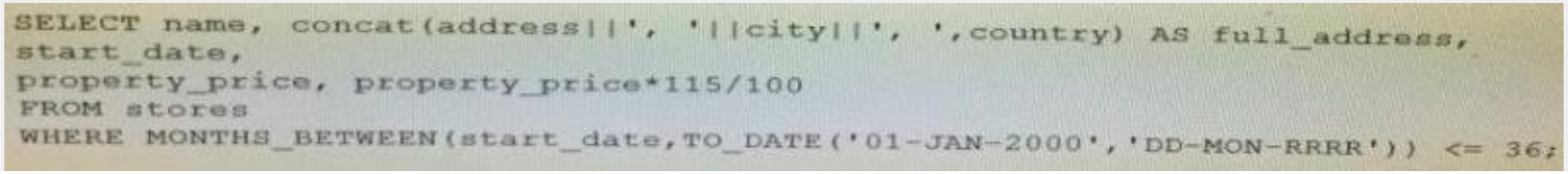
```
SELECT name, concat(address||', '||city||', ',country) AS full_address,
start_date,
property_price, property_price*115/100
FROM stores
WHERE TO_NUMBER(start_date-TO_DATE('01-JAN-2000','DD-MON-RRRR')) <= 36;
```

C)
Exhibit



```
SELECT name, address||', '||city||', '||country AS full_address, start_date,
property_price, property_price*15/100
FROM stores
WHERE MONTHS_BETWEEN(start_date,TO_DATE('01-JAN-2000','DD-MON-RRRR')) <= 36;
```

D)
Exhibit



```
SELECT name, concat(address||', '||city||', ',country) AS full_address,
start_date,
property_price, property_price*115/100
FROM stores
WHERE MONTHS_BETWEEN(start_date,TO_DATE('01-JAN-2000','DD-MON-RRRR')) <= 36;
```

Response:

Option A

Option C

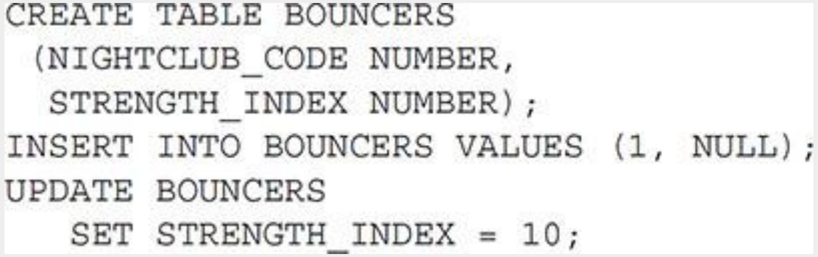
✓ Option D

Option B

Score 1 of 1

Question:

Review the following SQL statements:



```
CREATE TABLE BOUNCERS
 (NIGHTCLUB_CODE NUMBER,
  STRENGTH_INDEX NUMBER);
INSERT INTO BOUNCERS VALUES (1, NULL);
UPDATE BOUNCERS
   SET STRENGTH_INDEX = 10;
```

What is the end result of the SQL statements listed here?

Response:

✓ The BOUNCERS table will contain one row.

The BOUNCERS table will contain two rows.

None of the above.

The UPDATE will fail because there is no WHERE clause.

Score 1 of 1

Question:

Which statement is true regarding external tables?

Response:

✓ The CREATE TABLE AS SELECT statement can be used to unload data into regular table in the database from an external table.

The data and metadata for an external table are stored outside the database.

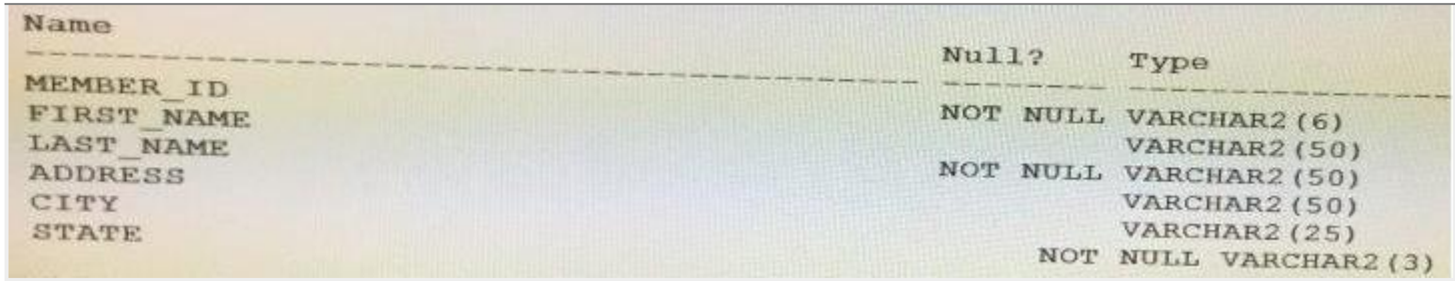
The default REJECT LIMIT for external tables is UNLIMITED.

ORACLE_LOADER and ORACLE_DATAPUMP have exactly the same functionality when used with an external table.

Score 1 of 1

Question:

Examine the structure of the members table:

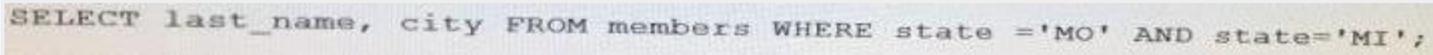


```
Name
--------------------------------------------   Null?      Type
MEMBER_ID                                      --------   ----------------
FIRST_NAME                                     NOT NULL   VARCHAR2(6)
LAST_NAME                                                 VARCHAR2(50)
ADDRESS     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

Which query can be used to display the last names and city names only for members from the states MO and MI?

A)

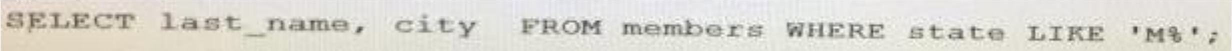
Exhibit



```
SELECT last_name, city FROM members WHERE state ='MO' AND state='MI';
```

B)

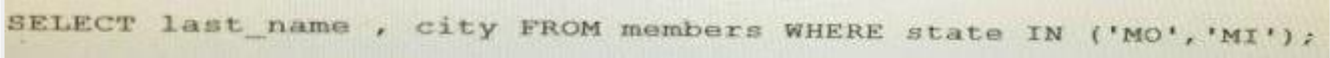
Exhibit



```
SELECT last_name, city  FROM members WHERE state LIKE 'M%';
```

C)

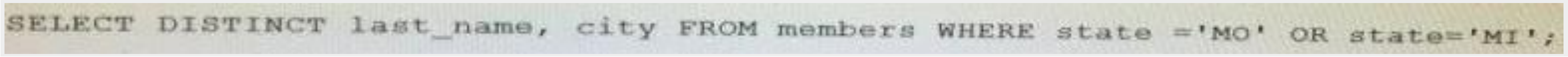
Exhibit



```
SELECT last_name , city FROM members WHERE state IN ('MO','MI');
```

D)

Exhibit



```
SELECT DISTINCT last_name, city FROM members WHERE state ='MO' OR state='MI';
```

Response:

Option A

✓ Option C

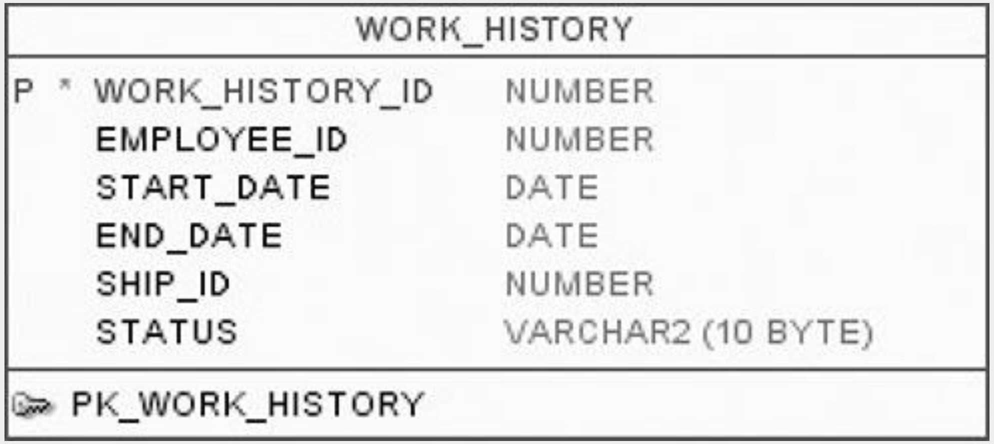
Option D

Option B

Score 0 of 1

Question:

Review this WORK_HISTORY table.



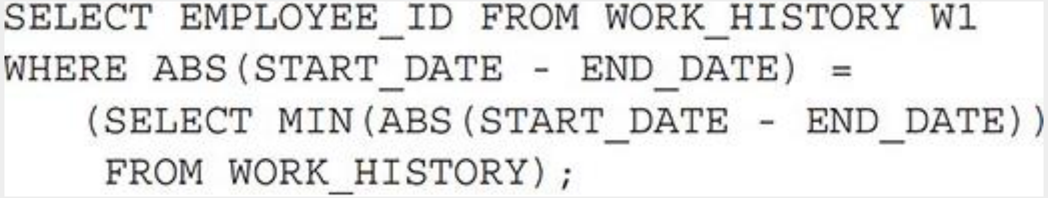
WORK_HISTORY		
P *	WORK_HISTORY_ID	NUMBER
	EMPLOYEE_ID	NUMBER
	START_DATE	DATE
	END_DATE	DATE
	SHIP_ID	NUMBER
	STATUS	VARCHAR2 (10 BYTE)
	PK_WORK_HISTORY	

Your task is to create a query that will list—for each ship—all of the EMPLOYEE_ID values for all the employees who have the shortest work history for their ship.

In other words, if there are two ships, you want to list all the employees assigned to the first ship who have the shortest work history, all the employees assigned to the second ship who have the shortest work history, and so on.

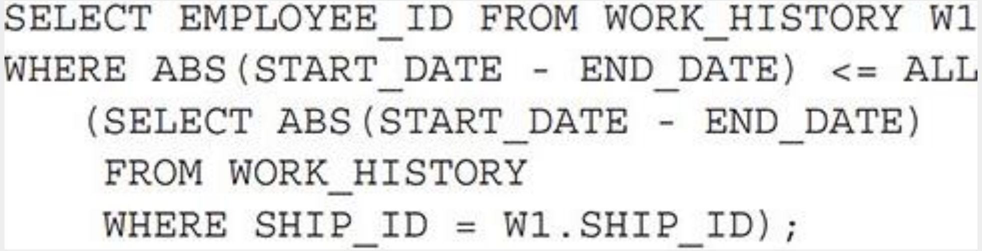
Which of the following queries will accomplish this task?
(Choose two.)

Response:



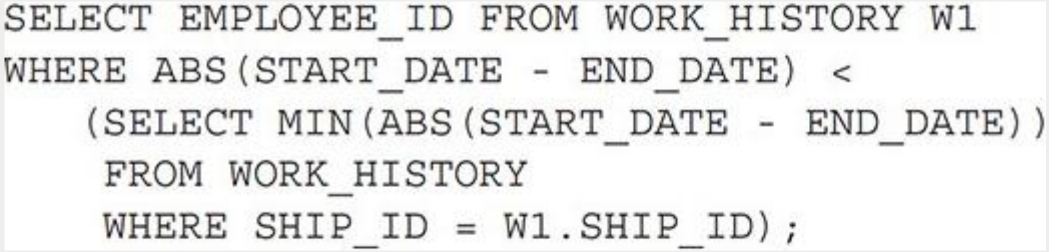
```
SELECT EMPLOYEE_ID FROM WORK_HISTORY W1
WHERE ABS(START_DATE - END_DATE) =
   (SELECT MIN(ABS(START_DATE - END_DATE))
    FROM WORK_HISTORY);
```

✓



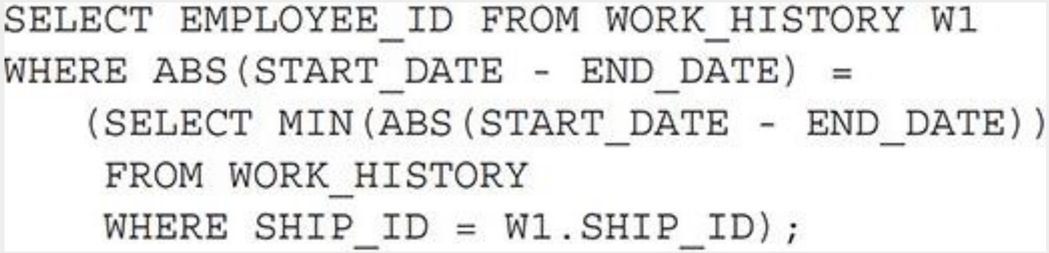
```
SELECT EMPLOYEE_ID FROM WORK_HISTORY W1
WHERE ABS(START_DATE - END_DATE) <= ALL
   (SELECT ABS(START_DATE - END_DATE)
    FROM WORK_HISTORY
    WHERE SHIP_ID = W1.SHIP_ID);
```

✗



```
SELECT EMPLOYEE_ID FROM WORK_HISTORY W1
WHERE ABS(START_DATE - END_DATE) <
   (SELECT MIN(ABS(START_DATE - END_DATE))
    FROM WORK_HISTORY
    WHERE SHIP_ID = W1.SHIP_ID);
```

✓

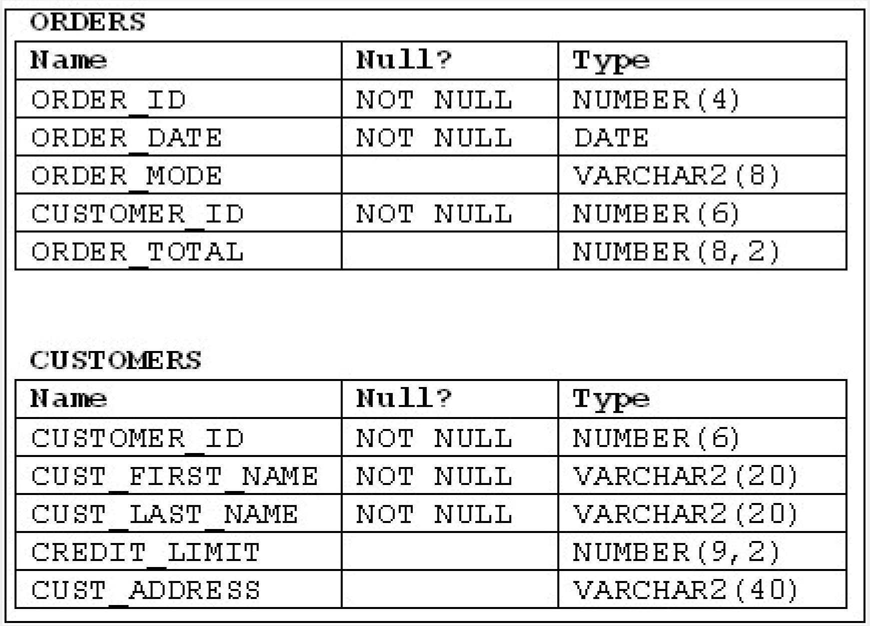


```
SELECT EMPLOYEE_ID FROM WORK_HISTORY W1
WHERE ABS(START_DATE - END_DATE) =
   (SELECT MIN(ABS(START_DATE - END_DATE))
    FROM WORK_HISTORY
    WHERE SHIP_ID = W1.SHIP_ID);
```

Score 1 of 1

Question:

View the Exhibit and examine the structure of ORDERS and CUSTOMERS tables.



ORDERS

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

CUSTOMERS

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

Which INSERT statement should be used to add a row into the ORDERS table for the customer whose CUST LAST NAME is Roberts and CREDIT LIMIT is 600?

Response:

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

```
INSERT INTO orders (order_id, order_date, order_mode,
(SELECT customer id
FROM customers
WHERE cust_last_name='Roberts' AND credit_limit=600), order_total);
VALUES (1,'10-mar-2007', 'direct', &customer_id, 1000);
```

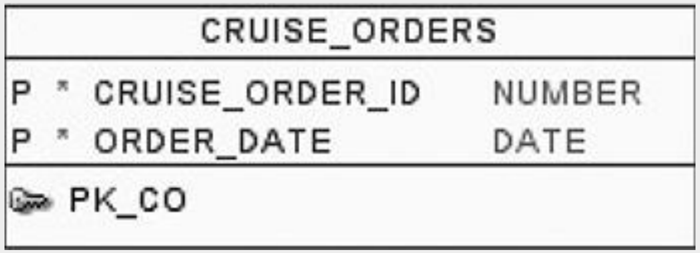
```
INSERT INTO orders (order_id, order_date, order_mode,
(SELECT customer_id
FROM customers
WHERE cust_last_name='Roberts' AND credit_limit=600), order_total);
VALUES (1,'10-mar-2007', 'direct', &customer_id, 1000);
```

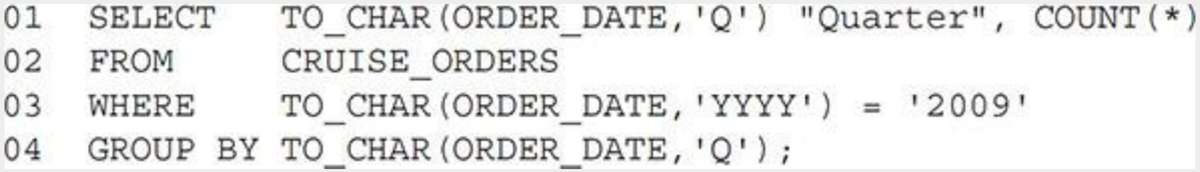
```
  INSERT INTO orders
  VALUES (1,'10-mar-2007', 'direct',
✓ (SELECT customer_id
  FROM customers
  WHERE cust_last_name='Roberts' AND credit_limit=600), 1000);
```

Score 1 of 1

Question:

Review the illustration and then look at the SQL code that follows:





```
01   SELECT   TO_CHAR(ORDER_DATE,'Q') "Quarter", COUNT(*)
02   FROM     CRUISE_ORDERS
03   WHERE    TO_CHAR(ORDER_DATE,'YYYY') = '2009'
04   GROUP BY TO_CHAR(ORDER_DATE,'Q');
```

Recall that the 'Q' format model is for quarter, so TO_CHAR using a DATE data type with the 'Q' format mask is translating the date into the quarter in which it falls—1, 2, 3, or 4.

Given that, which of the following statements is true of the SQL statement?

Response:

It will fail because of a syntax error in line 4 since you cannot use the TO_CHAR function in the GROUP BY clause.

None of the above.

It will fail because of a syntax error in line 1 since you cannot use the TO_CHAR function with the COUNT aggregate function.

✓ It will execute and show the number of orders in the CRUISE_ORDERS table for each quarter in the year 2009.

Score 0 of 1

Question:

An invisible index is an index on one or more columns in a table:

Response:

✗ And is updated for any SELECT statements performed on the table

✓ And is updated for any DELETE statements performed on the table

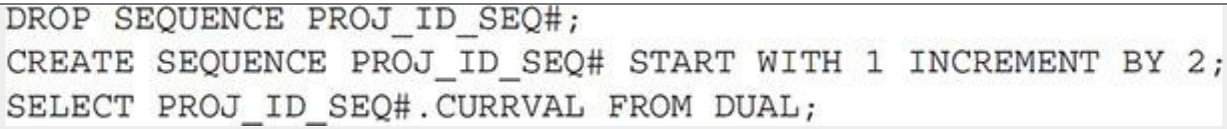
Where all the columns must be invisible

Where at least one of the columns must be invisible

Score 1 of 1

Question:

Review this code:



```
DROP SEQUENCE PROJ_ID_SEQ#;
CREATE SEQUENCE PROJ_ID_SEQ# START WITH 1 INCREMENT BY 2;
SELECT PROJ_ID_SEQ#.CURRVAL FROM DUAL;
```

What will result from these SQL statements?

Response:

✓ The SELECT statement will fail because you cannot reference the CURRVAL pseudocolumn of a sequence until after you have referenced NEXTVAL for the sequence in a session.

The SELECT statement will fail because the sequence can be referenced only in an INSERT statement.

The SELECT statement will display a value of 3.

The SELECT statement will display a value of 1.

Score 1 of 1

Question:

If you focus on trying to achieve the minimum passing grade requirement for the exam, you can study more efficiently.

Response:

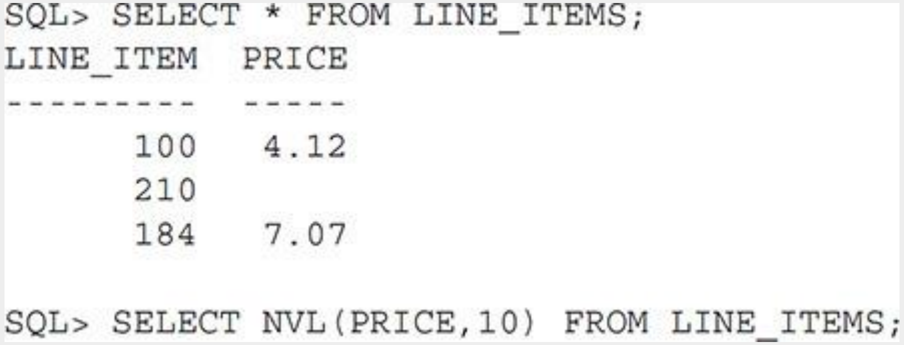
True

✓ False

Score 1 of 1

Question:

Consider the following query, its output, and a subsequent query:



```
SQL> SELECT * FROM LINE_ITEMS;
LINE_ITEM  PRICE
---------  -----
      100   4.12
      210
      184   7.07

SQL> SELECT NVL(PRICE,10) FROM LINE_ITEMS;
```

What is true of the final query shown previously?

Response:

It will return “no rows found” because there is no PRICE of 10.

✓ It will return three rows, but it will not change the price for line items 100 and 184.

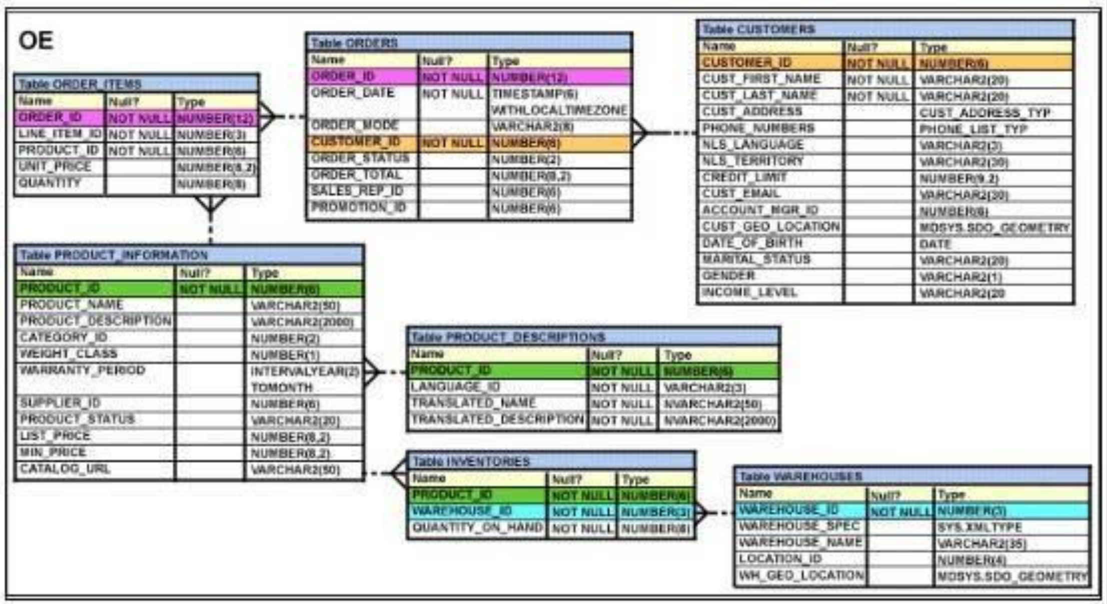
It will return no rows because there is no PRICE of 10.

It will return only the row where LINE_ITEM is 210.

Score 0 of 1

Question:

View the Exhibit and examine the data in ORDERS and ORDER_ITEMS tables. You need to create a view that displays the ORDER ID, ORDER_DATE, and the total number of items in each order.



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

Which CREATE VIEW statement would create the view successfully?

Response:

✓

```
CREATE OR REPLACE VIEW ord_vu
AS SELECT o.order_id, o.order_date, COUNT(i.line_item_id) "NO OF ITEMS"
FROM orders o JOIN order_items i ON (o.order_id = i.order_id)
GROUP BY o.order_id,o.order_date;
```

```
CREATE OR REPLACE VIEW ord_vu (order_id,order_date) AS SELECT
```

```
o.order_id, o.order_date, COUNT(i.line_item_id) "NO OF ITEMS"
FROM orders o JOIN order_items i ON (o.order_id = i.order_id)
GROUP BY o.order_id,o.order_date;
```

```
CREATE OR REPLACE VIEW ord_vu
AS SELECT o.order_id, o.order_date, COUNT(i.line_item_id)ll’ NO OF ITEMS'
FROM orders o JOIN order_items i
ON (o.order_id = i.order_id) GROUP BY o.order_id,o.order_date WITH CHECK
OPTION;
```

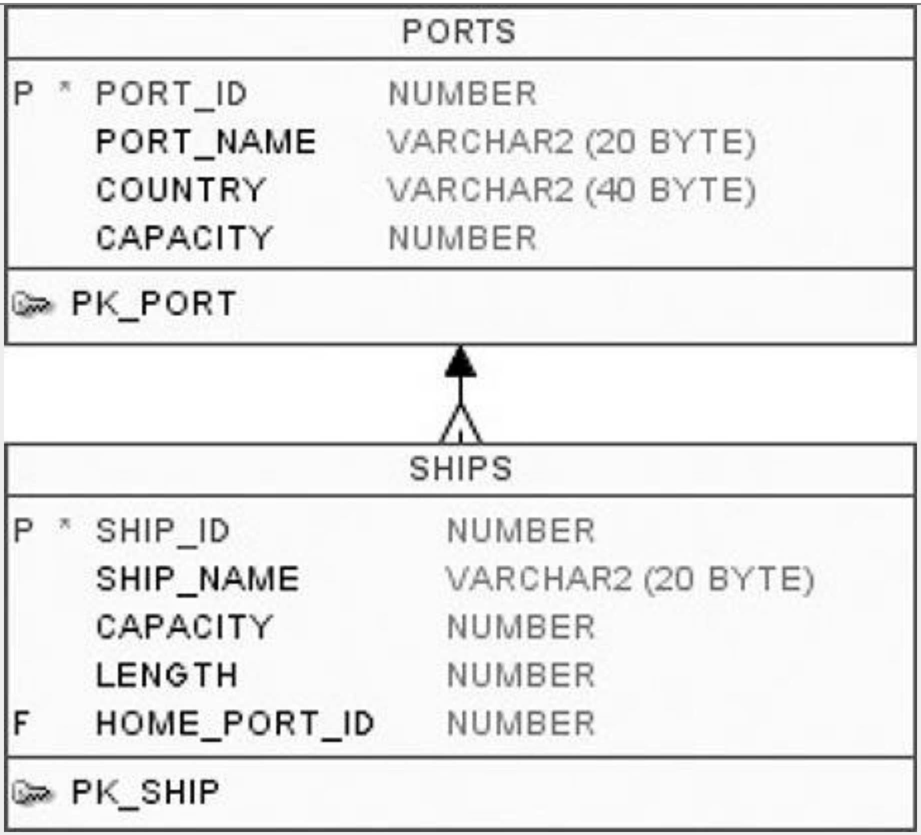
✗

```
CREATE OR REPLACE VIEW ord_vu
AS SELECT o.order_id, o.order_date, COUNT(i.line_item_id) FROM orders o
JOIN order_items i ON (o.order_id = i.order_id) GROUP BY
o.order_id,o.order_date;
```

Score 1 of 1

Question:

Review the illustration and the following SQL code:



PORTS

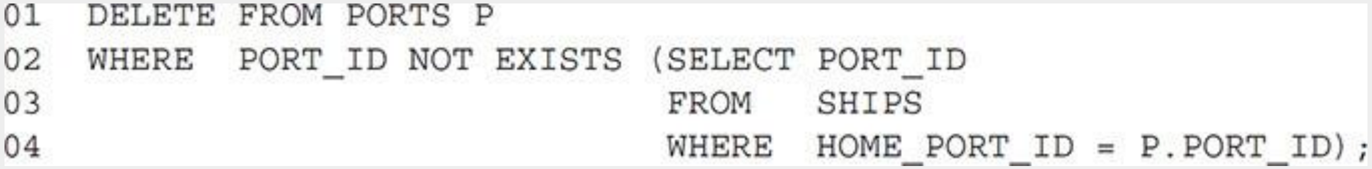
P	*	PORT_ID	NUMBER
		PORT_NAME	VARCHAR2 (20 BYTE)
		COUNTRY	VARCHAR2 (40 BYTE)
		CAPACITY	NUMBER

PK_PORT

SHIPS

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

PK_SHIP



```
DELETE FROM PORTS P
WHERE  PORT_ID NOT EXISTS (SELECT PORT_ID
                           FROM   SHIPS
                           WHERE  HOME_PORT_ID = P.PORT_ID);
```

The code is attempting to delete any row in the PORTS table that is not a home port for any ship in the SHIPS table, as indicated by the HOME_PORT_ID column.

In other words, only keep the PORTS rows that are currently the HOME_PORT_ID value for a ship in the SHIPS table; get rid of all other PORT rows. That’s the intent of the SQL statement.

What will result from an attempt to execute the preceding SQL statement?

Response:

✓ It will fail because of a syntax error on line 2.

It will fail because of an execution error in the subquery.

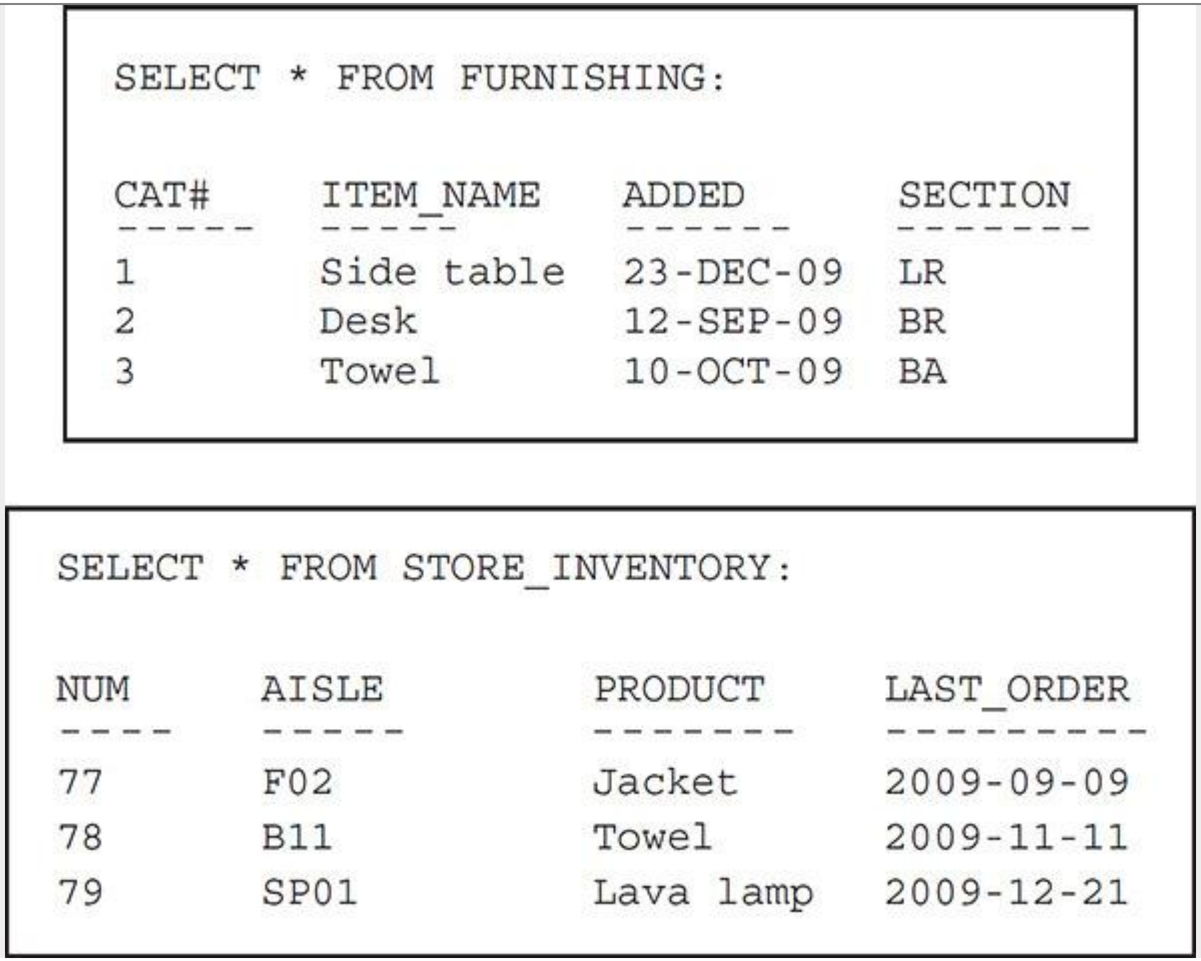
It will execute successfully and perform as intended.

It will fail because of a syntax error on line 4.

Score 1 of 1

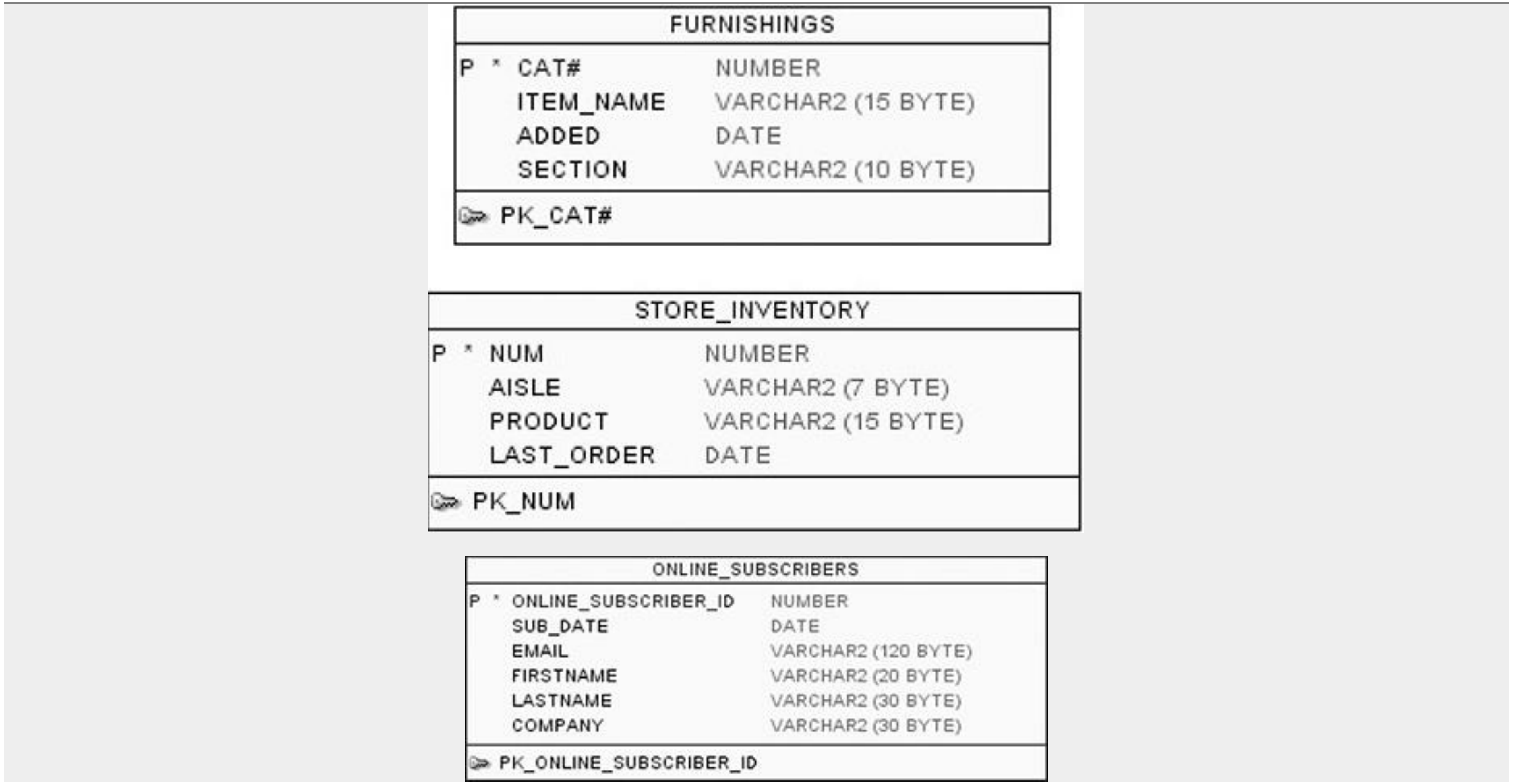
Question:

Review the first two illustrations as well as the ONLINE_SUBSCRIBERS table 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
```

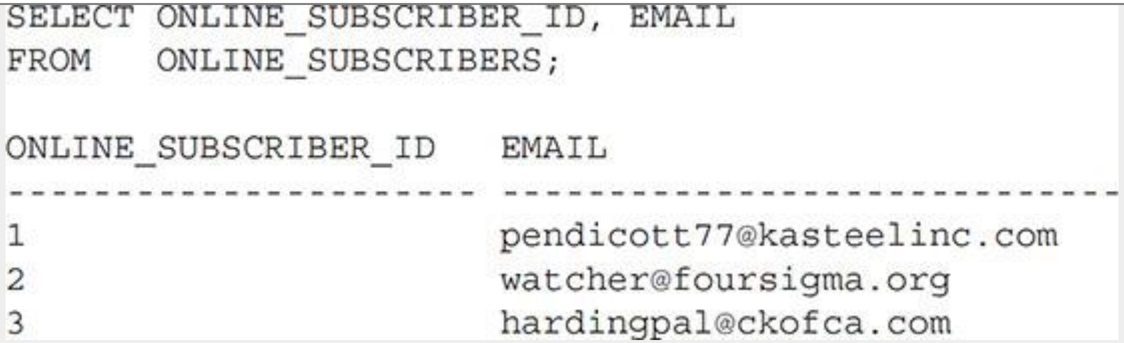
```
SELECT * FROM STORE_INVENTORY:

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



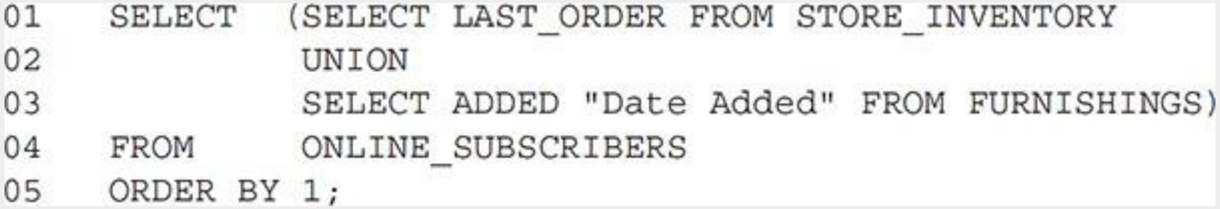
FURNISHINGS			
P	*	CAT#	NUMBER
		ITEM_NAME	VARCHAR2 (15 BYTE)
		ADDED	DATE
		SECTION	VARCHAR2 (10 BYTE)
PK_CAT#			

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

ONLINE_SUBSCRIBERS			
P	*	ONLINE_SUBSCRIBER_ID	NUMBER
		SUB_DATE	DATE
		EMAIL	VARCHAR2 (120 BYTE)
		FIRSTNAME	VARCHAR2 (20 BYTE)
		LASTNAME	VARCHAR2 (30 BYTE)
		COMPANY	VARCHAR2 (30 BYTE)
PK_ONLINE_SUBSCRIBER_ID			



```
SELECT ONLINE_SUBSCRIBER_ID, EMAIL
FROM   ONLINE_SUBSCRIBERS;

ONLINE_SUBSCRIBER_ID   EMAIL
---------------------- ----------------------------
1                      pendicott77@kasteelinc.com
2                      watcher@foursigma.org
3                      hardingpal@ckofca.com
```



```
01   SELECT  (SELECT LAST_ORDER FROM STORE_INVENTORY
02            UNION
03            SELECT ADDED "Date Added" FROM FURNISHINGS)
04   FROM     ONLINE_SUBSCRIBERS
05   ORDER BY 1;
```

What will happen when this SQL statement is executed?

Response:

It will execute and display one column under the “Date Added” heading.

It will execute, but the UNION will not work as expected.

✓ It will fail with an execution error on line 1.

It will execute and display one column under the “LAST_ORDER” heading.

Score 1 of 1

Question:

Evaluate the following SQL statement:

```
SELECT product_name || 'it's not available for order' FROM product_information
WHERE product_status = 'obsolete';
```

You received the following error while executing the above query:

```
ERROR: ORA-01756: quoted string not properly terminated
```

What would you do to execute the query successfully?

Response:

Enclose the character literal string in the SELECT clause within the double quotation marks.

✓ Use Quote (q) operator and delimiter to allow the use of single quotation mark in the literal character string.

Use escape character to negate the single quotation mark inside the literal character string in the SELECT clause.

Do not enclose the character literal string in the SELECT clause within the single quotation marks.

Score 1 of 1

Question:

The BOOKS_TRANSACTIONS table exists in your database. Examine the SQL statement:

```
SQL>SELECT * FROM books_transactionsORDER BY 3;
```

What is the outcome on execution?

Response:

Rows are displayed in the order that they are stored in the table only for the first three rows.

✓ Rows are displayed sorted in ascending order of the values in the third column in the table.

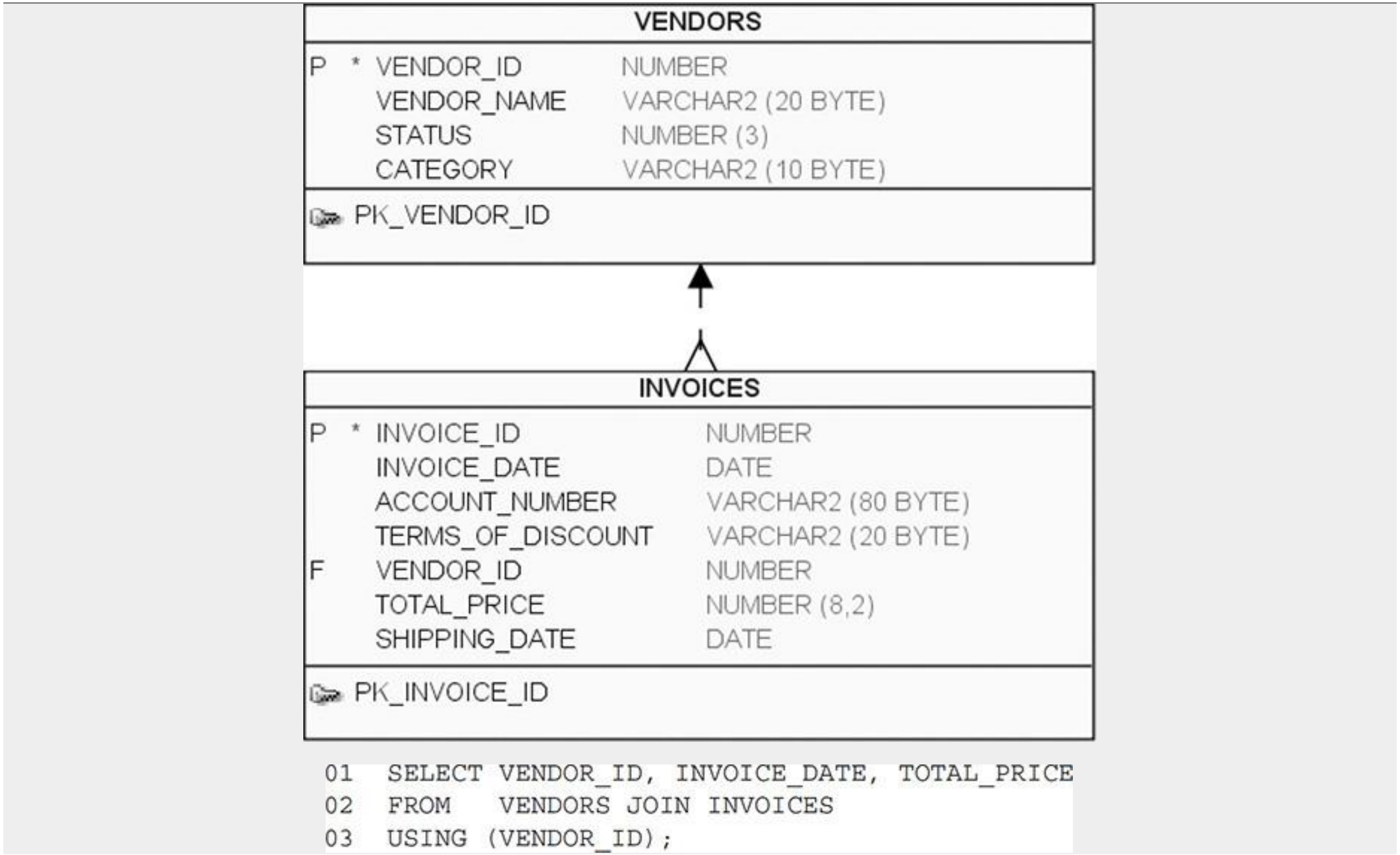
Rows are displayed in the order that they are stored in the table only for the three rows with the lowest values in the key column.

The execution tails unless the numeral 3 in the order by clause is replaced by a column name,

Score 1 of 1

Question:

Review the illustration and then review the following SQL statement:



VENDORS

P	*	VENDOR_ID	NUMBER
		VENDOR_NAME	VARCHAR2 (20 BYTE)
		STATUS	NUMBER (3)
		CATEGORY	VARCHAR2 (10 BYTE)

PK_VENDOR_ID

INVOICES

P	*	INVOICE_ID	NUMBER
		INVOICE_DATE	DATE
		ACCOUNT_NUMBER	VARCHAR2 (80 BYTE)
		TERMS_OF_DISCOUNT	VARCHAR2 (20 BYTE)
F		VENDOR_ID	NUMBER
		TOTAL_PRICE	NUMBER (8,2)
		SHIPPING_DATE	DATE

PK_INVOICE_ID

```
01  SELECT VENDOR_ID, INVOICE_DATE, TOTAL_PRICE
02  FROM   VENDORS JOIN INVOICES
03  USING (VENDOR_ID);
```

What kind of join is this?
(Choose two.)

Response:

OUTER

NATURAL

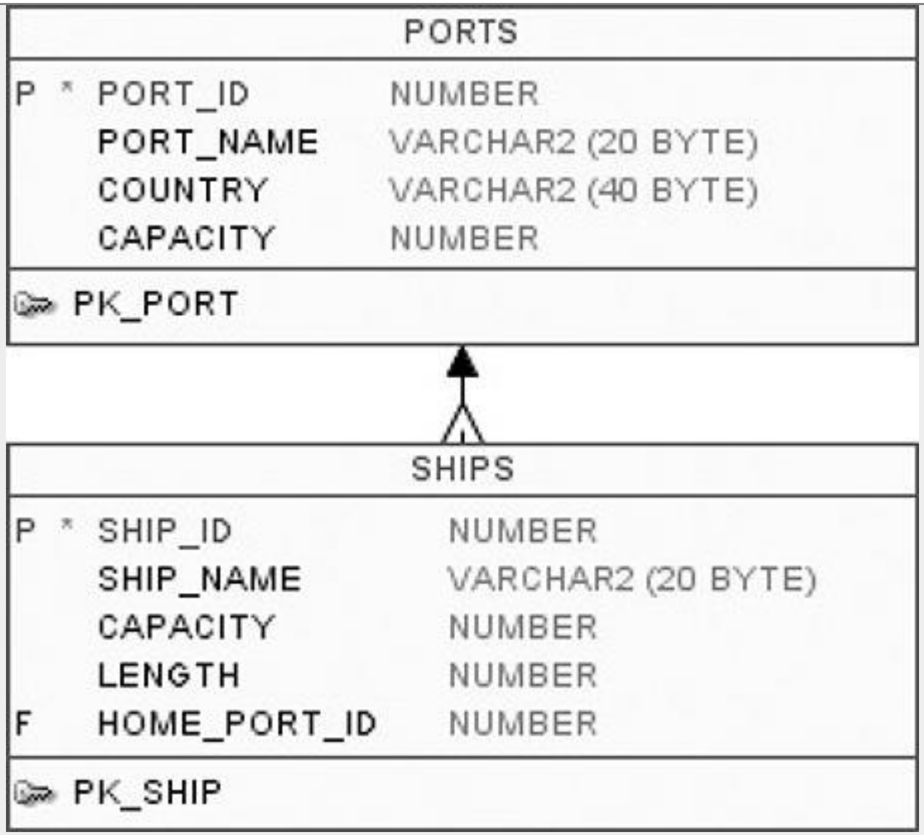
✓ Equijoin

✓ INNER

Score 1 of 1

Question:

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



PORTS

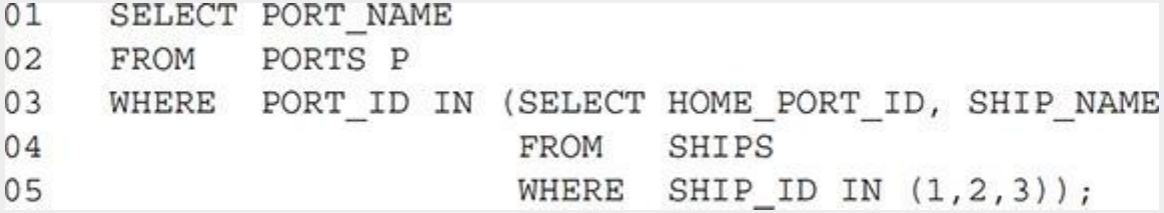
P	*	PORT_ID	NUMBER
		PORT_NAME	VARCHAR2 (20 BYTE)
		COUNTRY	VARCHAR2 (40 BYTE)
		CAPACITY	NUMBER

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:

None of the above.

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

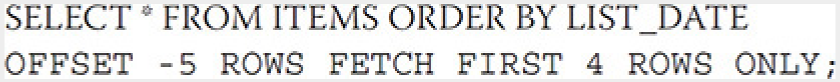
✓ The statement will fail with a syntax error because of line 3.

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

Score 1 of 1

Question:

Consider the following statement:



```
SELECT * FROM ITEMS ORDER BY LIST_DATE
OFFSET -5 ROWS FETCH FIRST 4 ROWS ONLY;
```

Assume you have a table ITEMS with a column LIST_DATE. What is the result of an attempt to execute the statement?

Response:

It will fail with a syntax error because of the use of a negative number with OFFSET.

It will fail with a syntax error because of the use of FIRST and OFFSET together.

It will sort the rows by LIST_DATE and return only the last four rows.

✓ It will sort the rows by LIST_DATE and return only the first four rows.

Score 1 of 1

Question:

Which three statements are true regarding the data types?

Response:

✓ Only one LONG column can be used per table.

✓ The minimum column width that can be specified for a varchar2 data type column is one.

ATIMESTAMP data type column stores only time values with fractional seconds.

The BLOB data type column is used to store binary data in an operating system file.

✓ The value for a CHAR data type column is blank-padded to the maximum defined column width.

Score 1 of 1

Question:

The first DROP operation is performed on PRODUCTS table using the following command:

```
DROP TABLE products PURGE;
```

Then you performed the FLASHBACK operation by using the following command:

FLASHBACK TABLE products TO BEFORE DROP;

Which statement describes the outcome of the FLASHBACK command?

Response:

It recovers only the table structure.

It recovers the table structure, data, and the indexes.

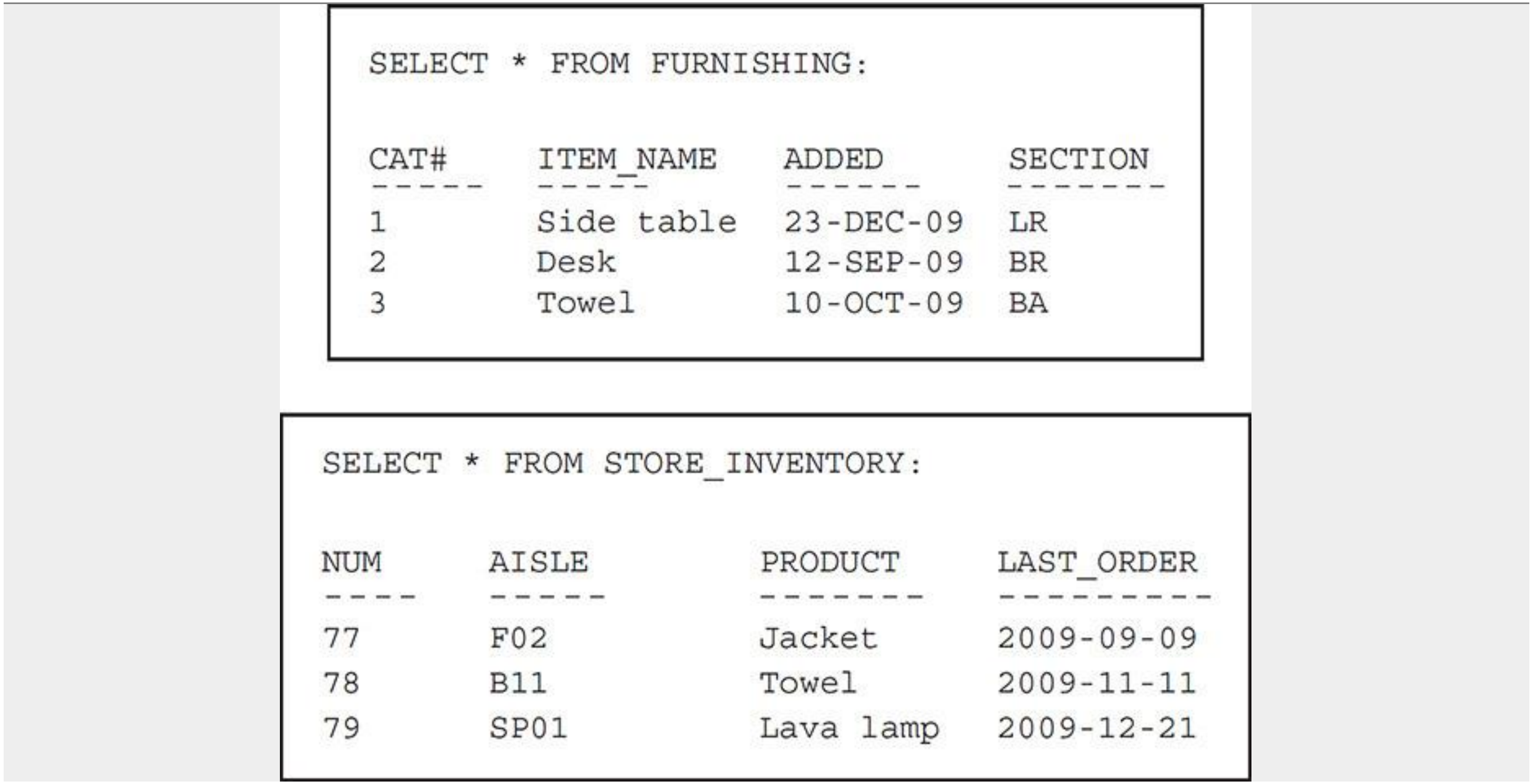
✓ It is not possible to recover the table structure, data, or the related indexes.

It recovers the table structure and data but not the related indexes.

Score 1 of 1

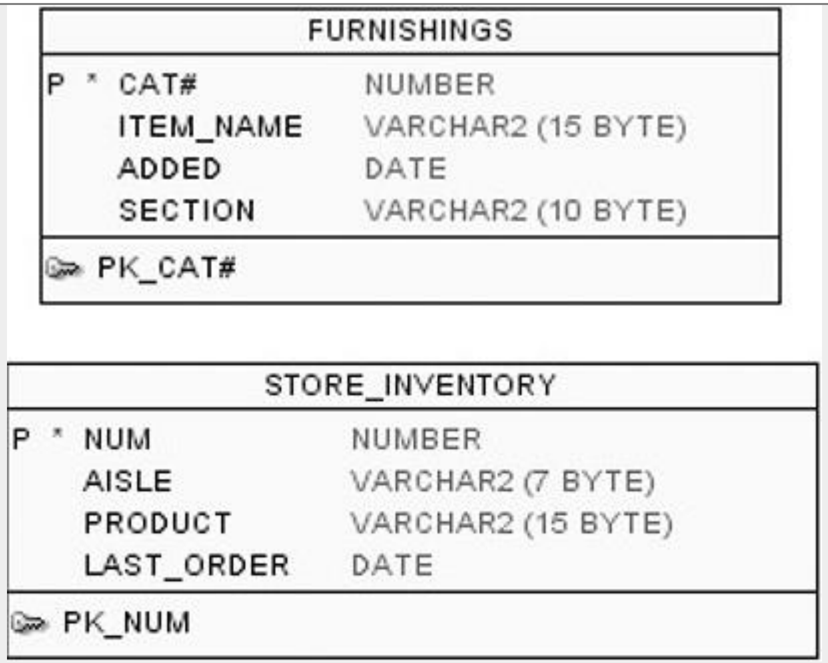
Question:

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



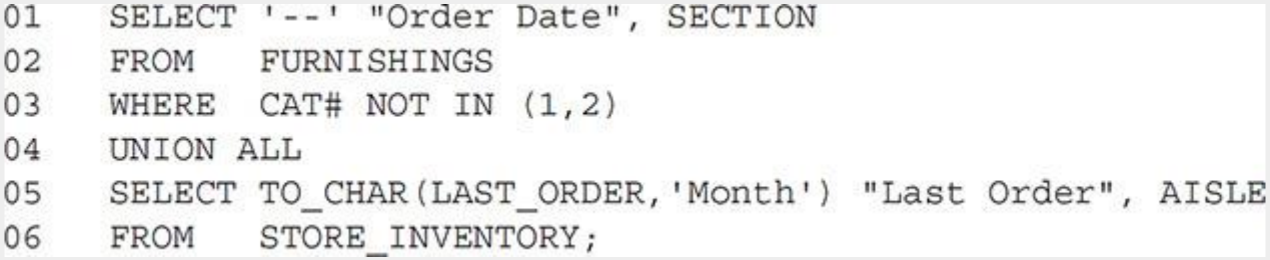
```
SELECT * FROM FURNISHING:

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 SECTION

✓ ORDER BY 1

ORDER BY AISLE

ORDER BY "Last Order"

Score 0 of 1

Question:

The CASCADE keyword, when used with TRUNCATE:

Response:

✓ None of the above

Is required if the table has any dependent child tables

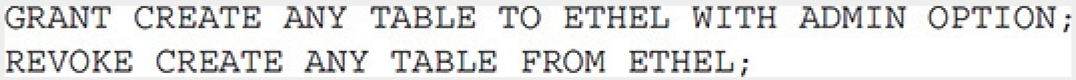
Can be used with the optional DEPENDENCY keyword

✗ Will ensure that future attempts to insert rows to the table will be rejected if they satisfy the TRUNCATE table’s WHERE clause

Score 0 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 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 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 does not have the system privilege CREATE ANY TABLE or the right to grant the CREATE ANY TABLE system privilege to any other user.

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

Score 0 of 1

Question:

Review this SQL statement: SELECT TRUNC(ROUND(ABS(-1.7),2)) FROM DUAL; What will be the result of the SQL statement?

Response:

−1

✓ 1

✗ −2

2

Score 0 of 1

Question:

Which of the following statements is true about HAVING?
(Choose two.)

Response:

✗ It cannot reference an expression unless that expression is first referenced in the GROUP BY clause.

It must occur after the GROUP BY clause.

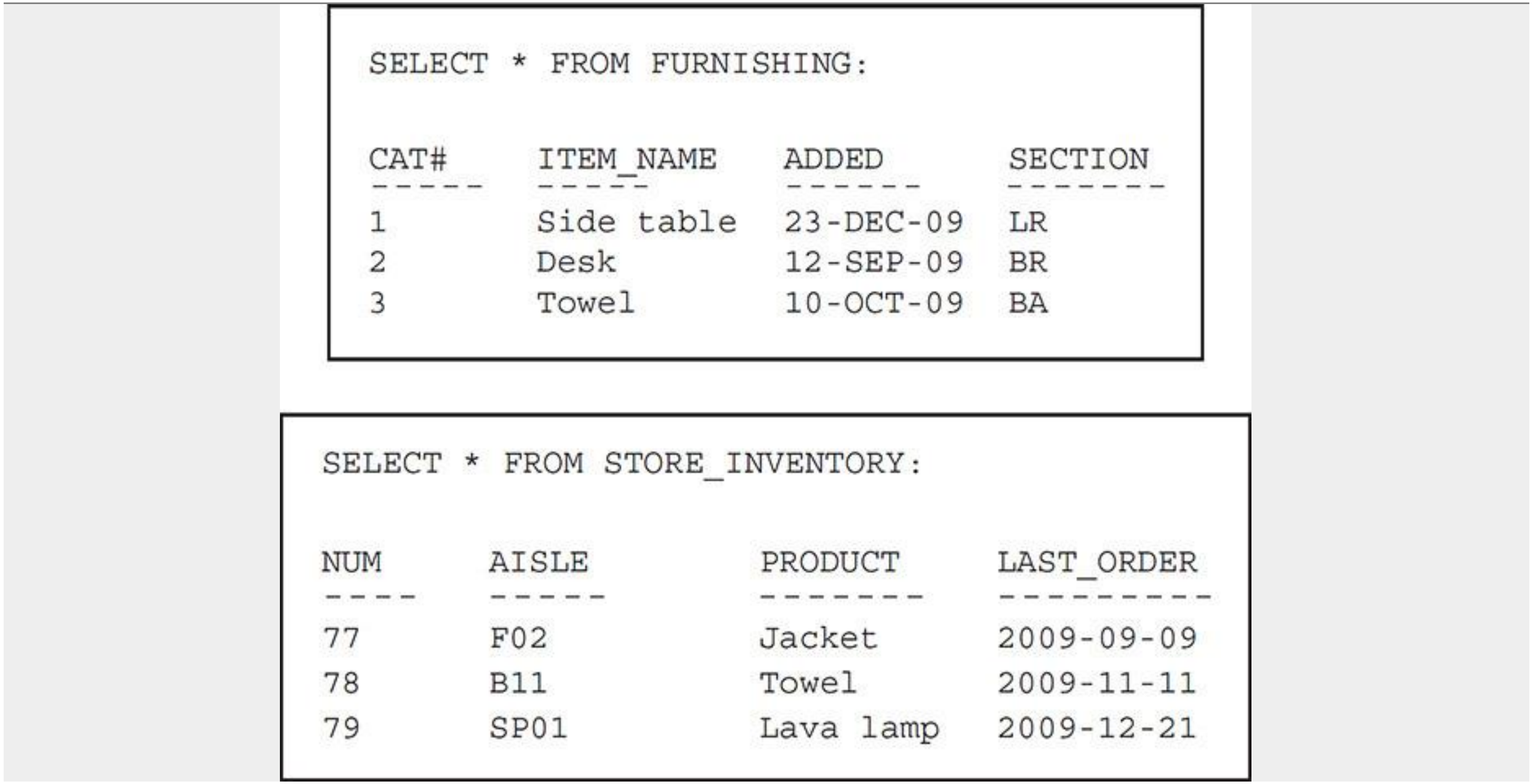
✓ It must occur after the WHERE clause.

✓ It can be used only in the SELECT statement.

Score 1 of 1

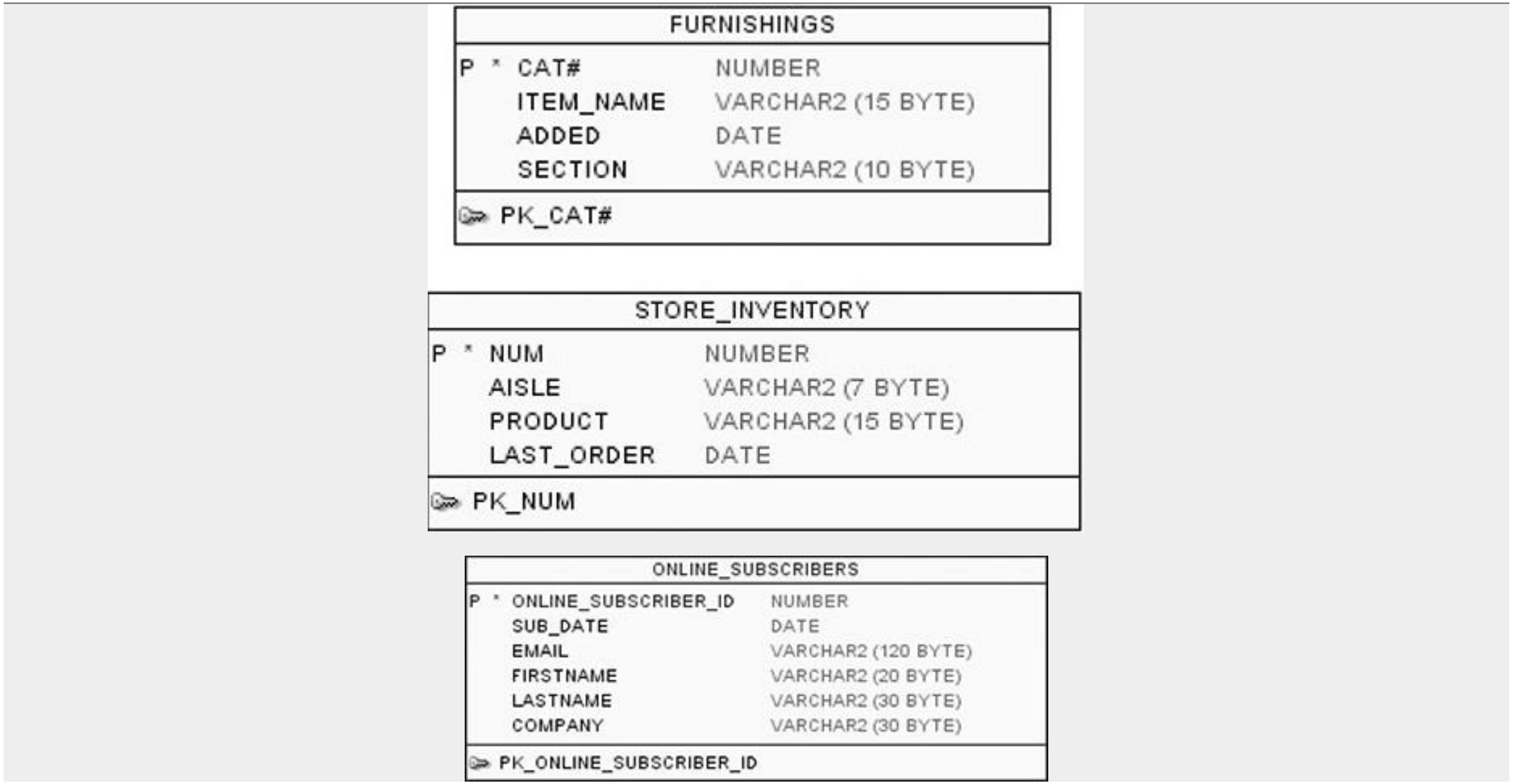
Question:

Review the first two illustrations as well as the ONLINE_SUBSCRIBERS table 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

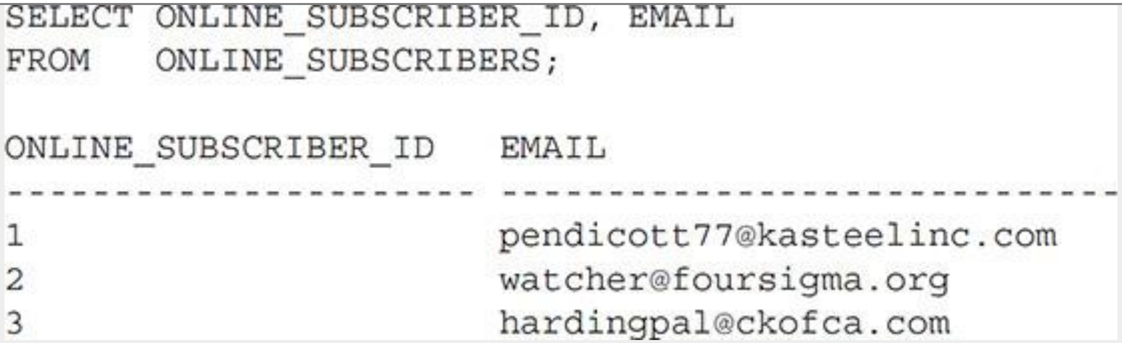
P	*	NUM	NUMBER
		AISLE	VARCHAR2 (7 BYTE)
		PRODUCT	VARCHAR2 (15 BYTE)
		LAST_ORDER	DATE

PK_NUM

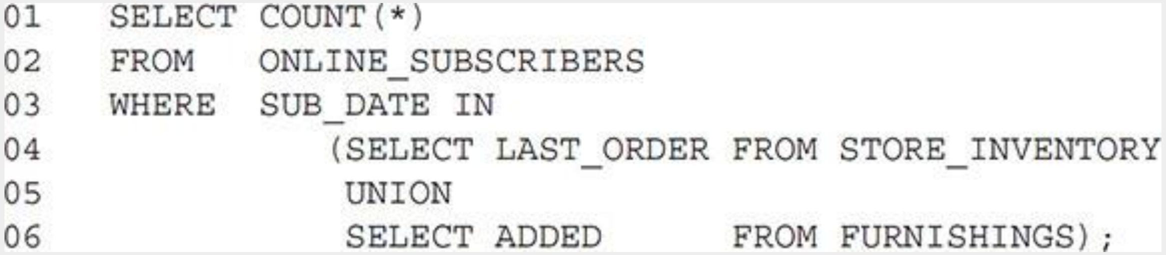
ONLINE_SUBSCRIBERS

P	*	ONLINE_SUBSCRIBER_ID	NUMBER
		SUB_DATE	DATE
		EMAIL	VARCHAR2 (120 BYTE)
		FIRSTNAME	VARCHAR2 (20 BYTE)
		LASTNAME	VARCHAR2 (30 BYTE)
		COMPANY	VARCHAR2 (30 BYTE)

PK_ONLINE_SUBSCRIBER_ID



```
SELECT ONLINE_SUBSCRIBER_ID, EMAIL
FROM   ONLINE_SUBSCRIBERS;

ONLINE_SUBSCRIBER_ID   EMAIL
---------------------- -----------------------------
1                      pendicott77@kasteelinc.com
2                      watcher@foursigma.org
3                      hardingpal@ckofca.com
```



```
01   SELECT COUNT(*)
02   FROM   ONLINE_SUBSCRIBERS
03   WHERE  SUB_DATE IN
04             (SELECT LAST_ORDER FROM STORE_INVENTORY
05              UNION
06              SELECT ADDED      FROM FURNISHINGS);
```

What will happen when this SQL statement is executed?

Response:

✓ It will execute successfully.

It will fail with a syntax error because you cannot use an aggregate function like COUNT(*) in line 1 in this context.

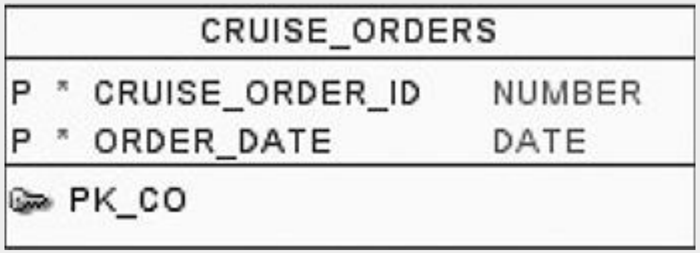
It will execute, but it will not perform as intended because the second SELECT statement within the subquery on line 6 will not execute; only the first SELECT in the subquery on line 4 will execute.

It will fail with a syntax error starting at line 4.

Score 1 of 1

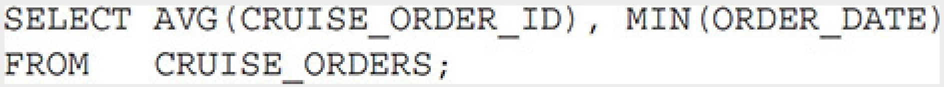
Question:

Review the illustration and then review the following SQL statement:



CRUISE_ORDERS

```
P * CRUISE_ORDER_ID    NUMBER
P * ORDER_DATE         DATE

PK_CO
```



```
SELECT AVG(CRUISE_ORDER_ID), MIN(ORDER_DATE)
FROM   CRUISE_ORDERS;
```

What will result from an attempt to execute this SQL statement on the CRUISE_ORDERS table?

Response:

✓ It will execute and perform as intended.

It will fail with an execution error if the table contains only one row.

It will fail with an execution error because you cannot use the AVG function on a PRIMARY KEY column.

It will fail with an execution error because you cannot use the MIN function on a DATE data type.

Score 1 of 1

Question:

Examine the business rule:
Each student can take up multiple projects and each project can have multiple students.

You need to design an Entity Relationship Model (ERD) for optimal data storage and allow for generating reports in this format:
STUDENT_ID FIRST_NAME LAST_NAME PROJECT_ID PROJECT_NAME PROJECT_TASK

Which two statements are true in this scenario?

Response:

The ERD must have a 1:M relationship between the STUDENTS and PROJECTS entities.

PROJECT_ID must be the primary key in the PROJECTS entity and foreign key in the STUDENTS entity

✓ The ERD must have a M:M relationship between the STUDENTS and PROJECTS entities that must be resolved into 1:M relationships.

✓ An associative table must be created with a composite key of STUDENT_ID and PROJECT_ID, which is the foreign key linked to the STUDENTS and PROJECTS entities.

STUDENT_ID must be the primary key in the STUDENTS entity and foreign key in the PROJECTS entity.

Score 0 of 1

Question:

You can add your own comments to the data dictionary with the COMMENT statement using which of the following? (Choose two.)

Response:

✓ COLUMN

✗ SEQUENCE

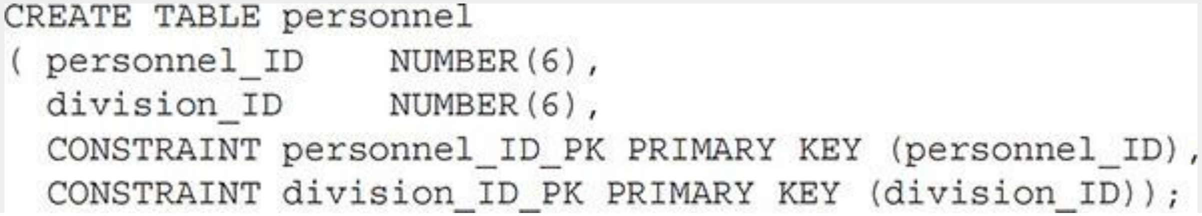
✓ TABLE

✗ INDEX

Score 0 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 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.

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

Score 1 of 1

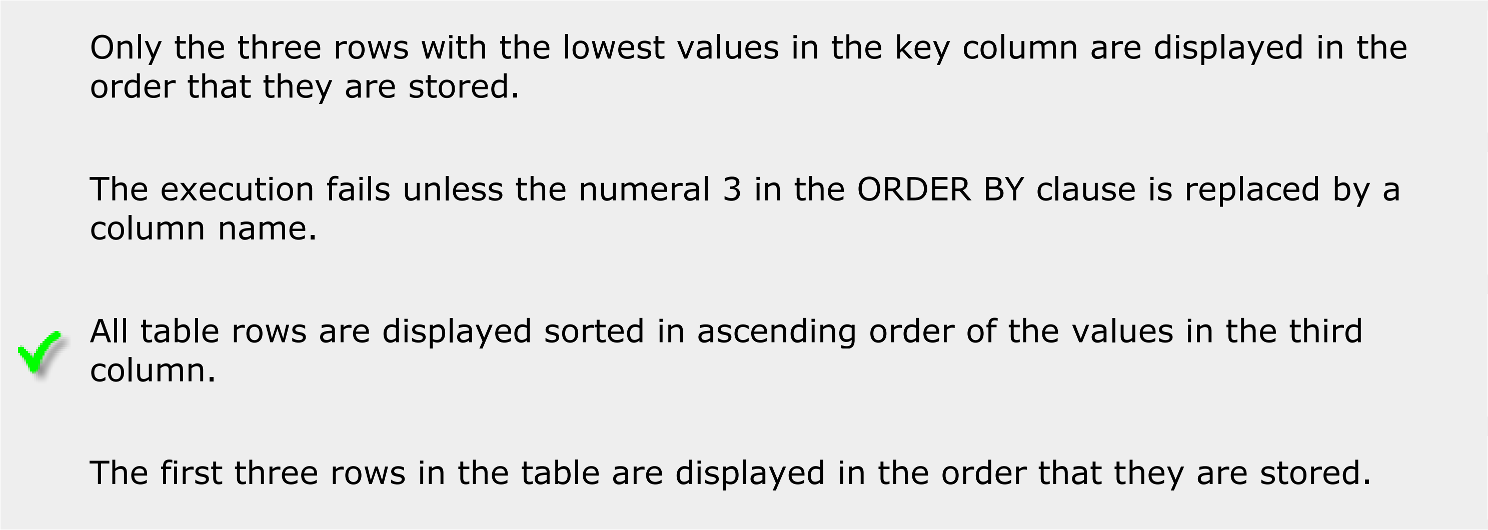
Question:

The BOOKS_TRANSACTIONS table exists in your schema in this database. You execute this SQL statement when connected to your schema in your database instance.

```
SQL> SELECT * FROM books transactions ORDER BY 3;
```

What is the result?

Response:



Only the three rows with the lowest values in the key column are displayed in the order that they are stored.

The execution fails unless the numeral 3 in the ORDER BY clause is replaced by a column name.

✓ All table rows are displayed sorted in ascending order of the values in the third column.

The first three rows in the table are displayed in the order that they are stored.