Oracle Database SQL (1Z0-071) - Mini

You got *18* of *36* possible points.
Your score was: 50 %

Question Results

Question: Score 0 of 1

Review the following diagrams of the SPARES table:

SPARES	
SPARE_ID	NUMBER (8)
PART_NO	VARCHAR2 (30 BYTE)
PART_NAME	VARCHAR2 (80 BYTE)
IX_01	

Also examine the diagrams of the tables PORT_INVENTORY, STORE_INVENTORY, and SHIP_INVENTORY, shown here.

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

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

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

Now consider the following SQL statement:

```
01   INSERT ALL
02     WHEN (SUBSTR(PART_NAME,1,4) = 'MED-') THEN
03       INTO STORE_INVENTORY (NUM, AISLE, PRODUCT, LAST_ORDER)
04       VALUES (SPARE_ID, 'Back', PART_NAME, SYSDATE)
05       INTO SHIP_INVENTORY (NUM, AISLE, PRODUCT, LAST_ORDER)
06       VALUES (SPARE_ID, 'Back', PART_NAME, SYSDATE)
07     WHEN (SUBSTR(PART_NAME,1,4) = 'ARR-') THEN
08       INTO PORT_INVENTORY (NUM, AISLE, PRODUCT, LAST_ORDER)
09       VALUES (SPARE_ID, 'Back', PART_NAME, SYSDATE)
10   SELECT SPARE_ID, PART_NO, PART_NAME
11   FROM   SPARES;
```

Regarding this SQL statement, which of the following statements is true?

Response:

The statement will fail because there is no ELSE clause.

✓ The statement will add a row returned from the SPARES table to the SHIP_INVENTORY table only if the WHEN condition on line 2 evaluates to true.

The statement will fail because it is missing a WHEN condition.

✗ The statement will add every row returned from the SPARES table to the SHIP_INVENTORY table.

Question: Score 1 of 1

View the Exhibit and examine PRODUCTS and ORDER_ITEMS tables.

PRODUCTS

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

ORDER_ITEMS

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

You executed the following query to display PRODUCT_NAME and the number of times the product has been ordered:

```
SELECT p.product_name, i.item_cnt
FROM (SELECT product_id, COUNT (*) item_cnt FROM order_items
GROUP BY product_id) i RIGHT OUTER JOIN products p ON i.product_id = p.product_id;
```

What would happen when the above statement is executed?

Response:

The statement would not execute because the ITEM_CNT alias cannot be displayed in the outer query.

✓ The statement would execute successfully to produce the required output.

The statement would not execute because the GROUP BY clause cannot be used in the inline view.

The statement would not execute because inline views and outer joins cannot be used together.

Question: Score 1 of 1

In which three situations does a transaction complete?

Response:

when a DELETEstatement is executed

✓ when a TRUNCATEstatement is executed after the pending transaction

✓ when a ROLLBACKcommand is executed

when a PL/SQL anonymous block is executed

✓ when a data definition language (DDL) statement is executed

Question: Score 0 of 1

Which statement correctly grants a system privilege?

Response:

✗
```
GRANT CREATE VIEW
ON table1 TO
user1;
```

```
GRANT CREATE SESSION
TO ALL;
```

```
GRANT ALTER TABLE
TO PUBLIC;
```

GRANT CREATE TABLE
TO user1, user2;

Question:

Score 0 of 1

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
     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);
```

```
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);
```

Question: Score 1 of 1

Which of the following SQL statements can always be executed on any VIEW object?
(Choose all that apply.)

Response:

- INSERT
- DELETE
- UPDATE
- ✓ SELECT

Question: Score 0 of 1

The difference between dropping a column from a table with DROP and setting a column to be UNUSED is:

Response:

- An UNUSED column can be recovered.
- Nothing.
- ✓ The UNUSED column and its data are retained within the table's storage allocation and counts against the total limit on the number of columns the table is allowed to have.
- ✗ A column that is dropped with DROP no longer appears within the table's description as shown with the DESC or DESCRIBE statement, whereas a column that is set to UNUSED still appears in the table's structure as shown in the output of the DESC statement.

Question: Score 1 of 1

Which of the following topics are not included in the SQL Fundamentals I exam but are addressed on the SQL Associate exam?
(Choose all that apply.)

Response:

- ✓ External tables
- ✓ MERGE
- ✓ FLASHBACK
- ✓ Conversion functions

Question: Score 1 of 1

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

Response:

- None of the above
- ✓ Remove existing data from a database table
- Issue privileges to users
- Query data from a given table

Question: Score 0 of 1

Which three statements are true about the ALTER TABLE....DROP COLUMN....command?

Response:

- ✓ A column can be dropped only if another column exists in the table.
- ✓ A parent key column in the table cannot be dropped.
- ✗ A dropped column can be rolled back.

A column can be dropped only if it does not contain any data.

The column in a composite PRIMARY KEY with the CASCADE option can be dropped.

Question: Score 1 of 1

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

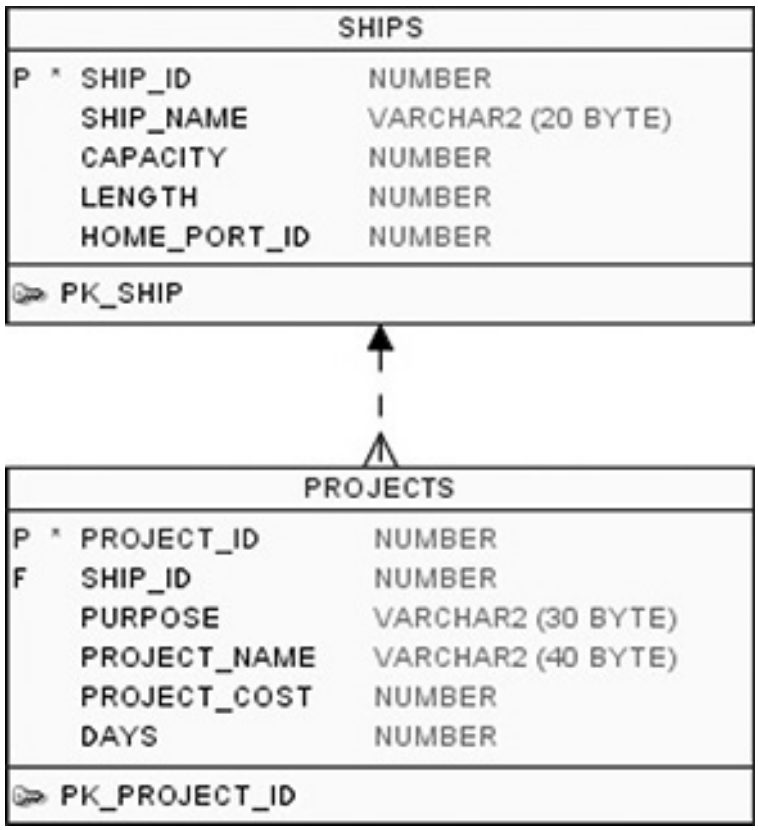
SELECT TO_CHAR(NVL(cust_credit_limit*1.15,'Not Available')) "NEW CREDIT" FROM customers;

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

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

Question: Score 0 of 1

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.

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 uses a GROUP BY clause.

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

Question: Score 1 of 1

The 1Z0-071 exam (which is the subject of this book) has been officially validated by Oracle Corporation against which of the following versions of the

Oracle database?
(Choose all that apply.)

Response:

- ✓ 11g
- 9i
- Every version
- ✓ 12c

Question: Score 0 of 1

Which subquery includes references to the parent query and thus cannot execute as a standalone query?
(Choose the best answer.)

Response:

- A multiple-column subquery
- ✗ A referential subquery
- ✓ A correlated subquery
- A scalar subquery

Question: Score 0 of 1

The output of a function may be used:
(Choose three.)

Response:

- ✓ As an input parameter value to an outer function.
- ✓ As an input value within the VALUES list of an INSERT statement.
- ✓ As a column of output in a SELECT statement.
- As an alternative to the keyword SET in an UPDATE statement.

Question: Score 1 of 1

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?

Response:

SELECT SUBSTR(cust_name, INSTR(cust_name,' ')+1) FROM customers
WHERE SUBSTR(cust_name, INSTR(cust_name,' ')+1) LIKE INITCAP('MC%');

✓ SELECT SUBSTR(cust_name, INSTR(cust_name,' ')+1) FROM customers
WHERE INITCAP(SUBSTR(cust_name, INSTR(cust_name,' ')+1)) LIKE 'Mc%';

SELECT SUBSTR(cust_name, INSTR(cust_name,' ')+1) FROM customers
WHERE INITCAP(SUBSTR(cust_name, INSTR(cust_name,' ')+1)) =
INITCAP('MC%');

SELECT SUBSTR(cust_name, INSTR(cust_name,' ')+1) FROM customers
WHERE INITCAP(SUBSTR(cust_name, INSTR(cust_name,' ')+1))='Mc';

Question: Score 0 of 1

Which two statements are true about Data Manipulation Language (DML) statements?

Response:

An UPDATE...SET... statement can modify multiple rows based on multiple conditions on a table.

A DELETE FROM..... statement can remove multiple rows based on multiple conditions on a table.

An INSERT INTO...VALUES..... statement can add a single row based on multiple conditions on a table.

✓ A DELETE FROM ..... statement can remove rows based on only a single condition on a table.

✓ AH INSERT INTO. . .VALUES. . statement can add multiple rows per execution to a table.

✗ An UPDATE...SET.... statement can modify multiple rows based on only a single condition on a table.

Question: Score 0 of 1

Evaluate the following SELECT statement and view the Exhibit to examine its output:

CONSTRAINT_NAME	CON	SEARCH_CONDITION	R_CONSTRAINT_NAME	DELETE_RULE	STATUS
ORDER_DATE_NN	C	"ORDER_DATE" IS NOT NULL			ENABLED
ORDER_CUSTOMER_ID_NN	C	"CUSTOMER_ID" IS NOT NULL			ENABLED
ORDER_MODE_LOV	C	order_mode in ('direct','online')			ENABLED
ORDER_TOTAL_MIN	C	order_total >= 0			ENABLED
ORDER_PK	P				ENABLED
ORDERS_CUSTOMER_ID	R		CUSTOMERS_ID	SET NULL	ENABLED
ORDERS_SALES_REP	R		EMP_EMP_ID	SET NULL	ENABLED

SELECT constraint_name, constraint_type, search_condition, r_constraint_name, delete_rule, status FROM user_constraints WHERE table_name = ORDERS

Which two statements are true about the output?

(Choose two.)

Response:

The R_CONSTRAINT_NAME column gives the alternative name for the constraint.

✔ The column DELETE_RULE decides the state of the related rows in the child table when the corresponding row is deleted from the parent table.

The STATUS column indicates whether the table is currently in use.

✔ In the second column, indicates a check constraint.

Question: Score 1 of 1

Which of the following SQL statements creates a table that will reject attempts to INSERT a row with NULL values entered into the POSITION_ID column?

Response:

```
CREATE TABLE POSITIONS
(POSITION_ID NUMBER(3),
 CONSTRAINT POSITION_CON REQUIRED (POSITION_ID));
```

None of the above

```
CREATE TABLE POSITIONS
(POSITION_ID NUMBER(3),
 CONSTRAINT POSITION_CON UNIQUE (POSITION_ID));
```

```
CREATE TABLE POSITIONS
(POSITION_ID NUMBER(3),
 CONSTRAINT POSITION_CON PRIMARY KEY (POSITION_ID));
```

Question: Score 0 of 1

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:

- 90.00.
- It will result in an execution error.
- 60.00.
- It will result in a syntax error because of the TO_CHAR function.

Question: Score 0 of 1

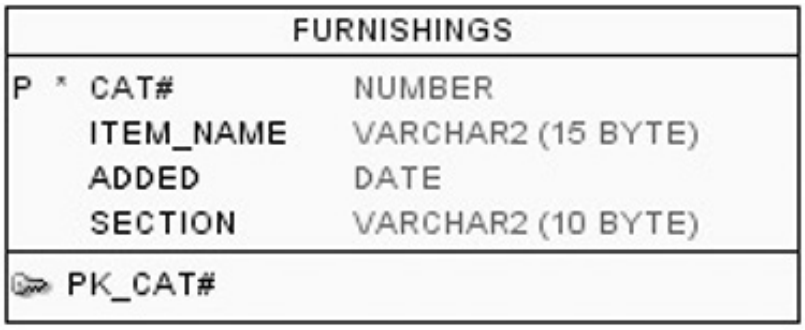
Review the first two illustrations as well as the ONLINE_SUBSCRIBERS table and then review this SQL code:

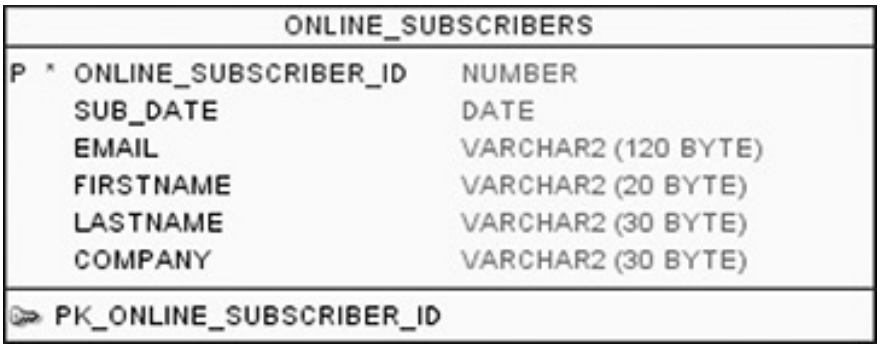
```
SELECT * FROM FURNISHING:

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
```





```
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 fail with a syntax error starting at line 4.

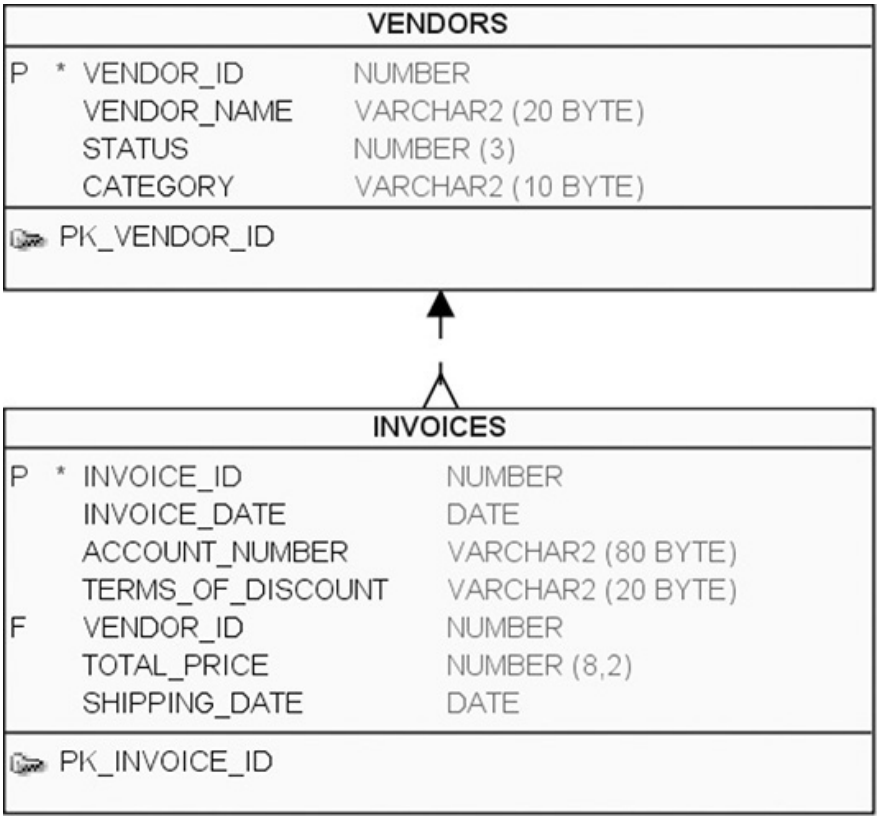
It will execute, but it will not perform as intended because the second SELECT statement within the subquery on line 6 will not execute; only the first SELECT in the subquery on line 4 will execute.

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

It will execute successfully.

Question: Score 1 of 1

Review the INVOICES and VENDORS tables.



Next review the following SQL statement:

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

Which of the following statements is true for the SQL statement?

Response:

It will fail with a syntax error because there is no ON clause.

It will fail with a syntax error on line 1 because VENDOR_ID is ambiguous.

✔ It will execute successfully.

It will fail with a syntax error on line 3 because of the parentheses around VENDOR_ID.

Question: Score 0 of 1

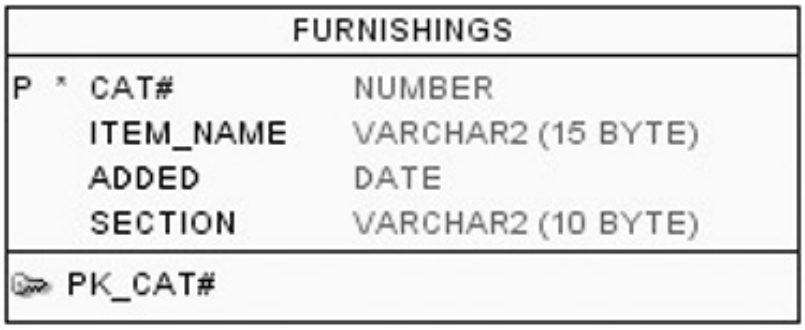
Review the first two illustrations as well as the ONLINE_SUBSCRIBERS table and then review this SQL code:

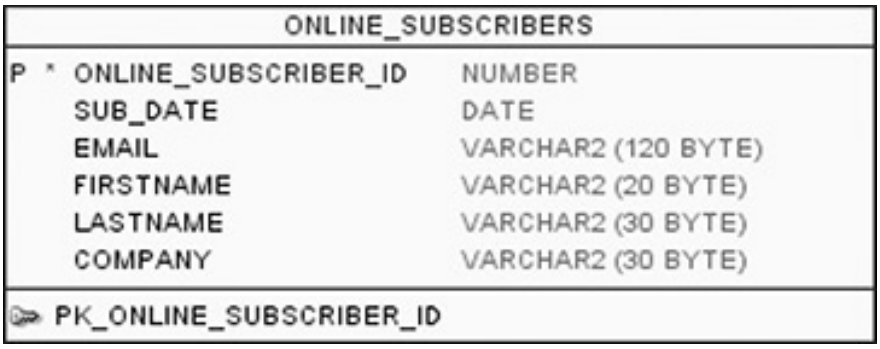
```
SELECT * FROM FURNISHING:

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
```





```
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 PRODUCT FROM STORE_INVENTORY
02            INTERSECT
03            SELECT ITEM_NAME FROM FURNISHINGS)
04   FROM     ONLINE_SUBSCRIBERS;
```

What will happen when this SQL statement is executed?

Response:

- ✓ It will execute and repeat the value 'Towel' for each row of the ONLINE_SUBSCRIBERS table.
- It will fail with a general syntax error.
- ✗ It will fail with an execution error.
- It will execute, but the INTERSECT will not work correctly.

Question: Score 1 of 1

Examine the commands used to createDEPARTMENT_DETAILS andCOURSE_DETAILS:

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

You want to generate a list of all department IDs along with any course IDs that may have been assigned to them. Which SQL statement must you use?

A)
Exhibit

```
SELECT d.department_id, c.course_id FROM department_details d RIGHT OUTER JOIN
course_details c ON (d.department_id=c.department_id);
```

B)
Exhibit

```
SELECT d.department_id, c.course_id FROM department_details d LEFT OUTER JOIN
course_details c ON (d.department_id=c.department_id);
```

C)
Exhibit

```
SELECT d.department_id, c.course_id FROM course_details c LEFT OUTER JOIN
department_details d ON (c.department_id=d.department_id);
```

D)
Exhibit

```
SELECT d.department_id, c.course_id FROM department_details d RIGHT OUTER JOIN
course_details c ON (c.department_id=d.department_id);
```

Response:

- Option D
- ✓ Option B
- Option A
- Option C

Question: Score 1 of 1

The database object that stores lookup information to speed up querying in tables is:

Response:

- ✓ INDEX
- ROWID
- LOOKUP
- VIEW

Question: Score 0 of 1

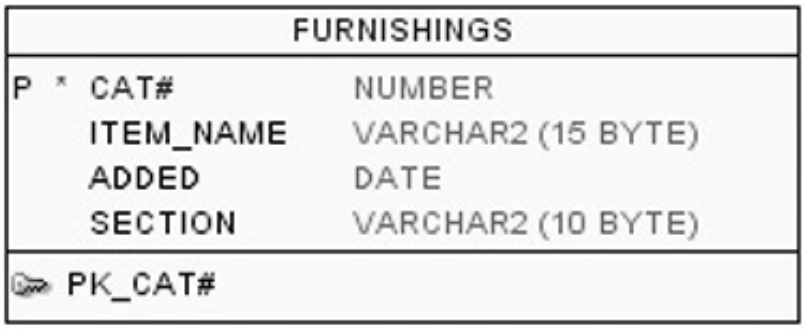
Review the first two illustrations and then review this SQL code:

```
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
```




```
(  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

4

✓ 1

✗ 2

Question: Score 0 of 1

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

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

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

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

Question:

Score 1 of 1

Which of the following is true of functions?

Response:

There is no consistent answer to whether they return a value or not.

✔ They always return a value.

They often return a value.

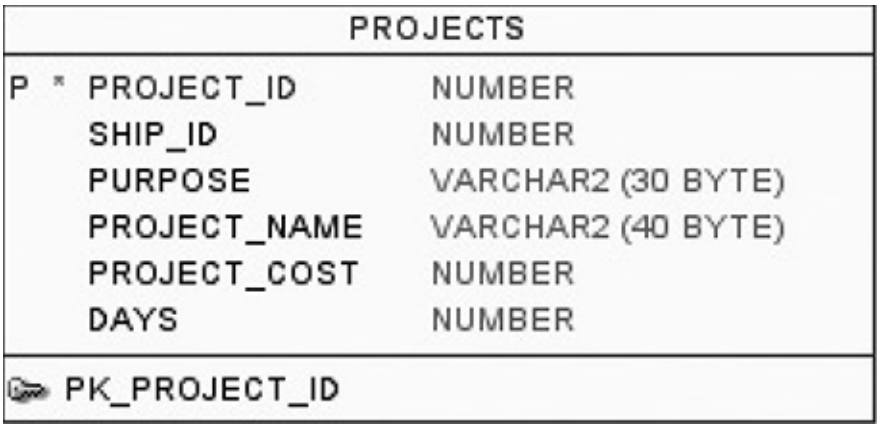
They never return a value.

Question:

Score 1 of 1

Review the illustration. Your task is to define a SELECT statement that groups rows according to their value for PURPOSE and, for each purpose, adds up the values stored in DAYS.

Which one of the following queries will perform this task?



Response:

```
SELECT SUM(DAYS), PURPOSE
FROM PROJECTS
GROUP BY PURPOSE, SUM(DAYS);
```

```
SELECT PURPOSE, RANK(DAYS) ON (ORDER BY)
FROM PROJECTS
GROUP BY PURPOSE;
```

```
SELECT PURPOSE, COUNT(DAYS)
FROM PROJECTS
GROUP BY PURPOSE;
```

✔

```
SELECT SUM(DAYS), PURPOSE
FROM PROJECTS
GROUP BY PURPOSE;
```

Question: Score 1 of 1

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

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

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

Question: Score 1 of 1

Which statement is true about an inner join specified in the WHERE clause of a query?

Response:

It requires the column names to be the same in all tables used for the join conditions.

It must have primary-key and foreign-key constraints defined on the columns used in the join condition.

It is applicable for only equijoin conditions.

✔ It is applicable for equijoin and nonequijoin conditions.

Question: Score 1 of 1

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:

- Can’t be done
- ✓ DBA_TAB_COLUMNS
- ALL_COLUMNS
- ALL_TAB_COLUMNS

Question: Score 1 of 1

Which three tasks can be performed using SQL functions built into Oracle Database?

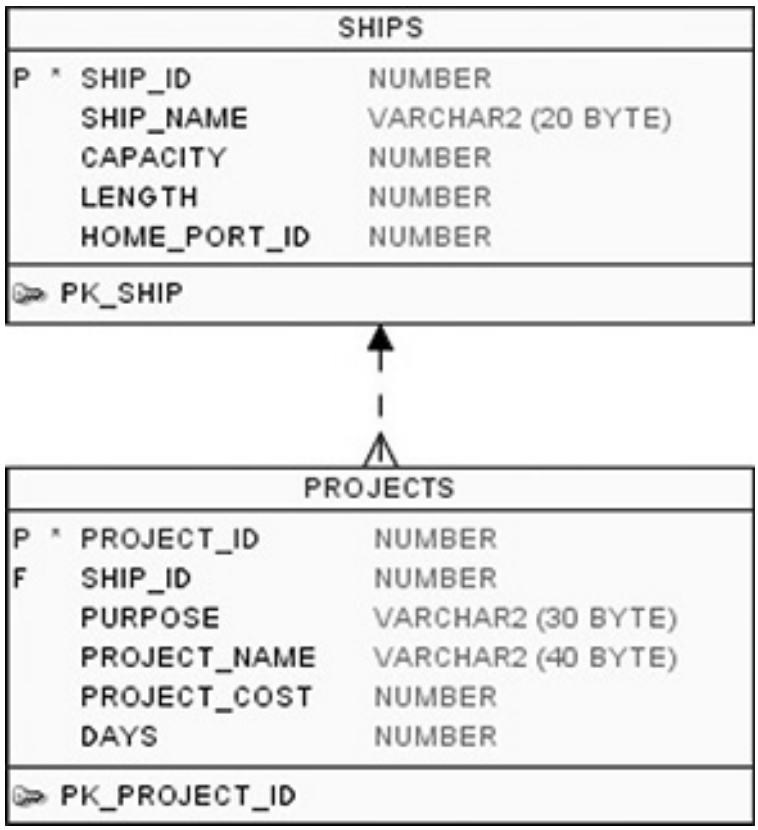
(Choose three.)

Response:

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

Question: Score 0 of 1

Review the illustration and the following SQL code:




```
01  CREATE OR REPLACE VIEW MAJOR_PROJECTS AS
02    SELECT PROJECT_ID, SHIP_ID, PROJECT_NAME, PROJECT_COST
03    FROM   PROJECTS
04    WHERE  PROJECT_COST > 10000;
05
06  INSERT INTO MAJOR_PROJECTS
07   (PROJECT_ID, SHIP_ID, PROJECT_NAME, PROJECT_COST)
08   VALUES
09   ((SELECT MAX(PROJECT_ID)+1 FROM PROJECTS),
10    (SELECT MAX(SHIP_ID) FROM SHIPS),
11    'Small Project',
12    500);
```

What will result from an attempt to execute these two SQL statements?

Response:

- ✓ The CREATE and INSERT statements will successfully execute.
- The INSERT statement will fail because the PROJECT_COST value being inserted is not consistent with the WHERE clause on line 4.
- The INSERT statement will fail because of an error on lines 9 and 10.
- ✗ The CREATE statement will fail because it omits the PURPOSE column from the PROJECTS table.

Question: Score 0 of 1

You have a table FURNISHINGS and are told to grant DELETE privileges on the table to user HEARST. Examine the following SQL statements:

```
GRANT DELETE ON FURNISHINGS TO HEARST;
CREATE ROLE MGR;
GRANT DELETE ON FURNISHINGS TO MGR;
GRANT MGR TO HEARST;
```

Now you are told to change the privileges given to HEARST so that HEARST can no longer execute DELETE statements on the FURNISHINGS table.

Which of the following will accomplish the goal?
(Choose the best answer.)

Response:

- ✗ REVOKE DELETE ON FURNISHINGS FROM MGR;
- ✓ REVOKE DELETE ON FURNISHINGS FROM HEARST, MGR;
- REVOKE DELETE ON FURNISHINGS FROM HEARST;
- None of the above

Question: Score 0 of 1

View the Exhibit and examine the details of the PRODUCT_INFORMATION table.

PRODUCT_NAME	CATEGORY_ID	SUPPLIER_ID
Inkjet C/8/HQ	12	102094
Inkjet C/4	12	102090
LaserPro 600/6/BW	12	102087
LaserPro 1200/8/BW	12	102099
Inkjet B/6	12	102096
Industrial 700/HD	12	102086
Industrial 600/DQ	12	102088
Compact 400/LQ	12	102087
Compact 400/DQ	12	102088
HD 12GB /R	13	102090
HD 10GB /I	13	102071
HD 12GB @7200 /SE	13	102057
HD 18.2GB @10000 /E	13	102078
HD 18.2GB@10000 /I	13	102050
HD 18GB /SE	13	102083
HD 6GB /I	13	102072
HD 8.2GB @5400	13	102093

You have the requirement to display PRODUCT_NAME and LIST_PRICE from the table where the CATEGORYJD column has values 12 or 13, and the SUPPLIER_ID column has the value 102088.

You executed the following SQL statement:

```
SELECT product_name, list_price FROM product_information
WHERE (category_id = 12 AND category_id = 13) AND supplier_id = 102088;
```

Which statement is true regarding the execution of the query?

Response:

 It would execute but the output would return no rows.

 It would not execute because the entire WHERE clause condition is not enclosed within the parentheses.

It would not execute because the same column has been used in both sides of the AND logical operator to form the condition.

It would execute and the output would display the desired result.