



Oracle Database SQL (1Z0-071) - Full

You got **65** of **71** possible points.

Your score: **92 %**

Question Results

Question:

Score 1 of 1

The set operators do not include which one of the following keywords?

Response:

MINUS

ALL

UNION



SET

Question:

Score 1 of 1

The data dictionary is owned by:

Response:

Each individual user



SYS

SYSTEM

PUBLIC

Question:

Score 1 of 1

All database data is stored in:

Response:

TABLES and VIEWS

None of the above

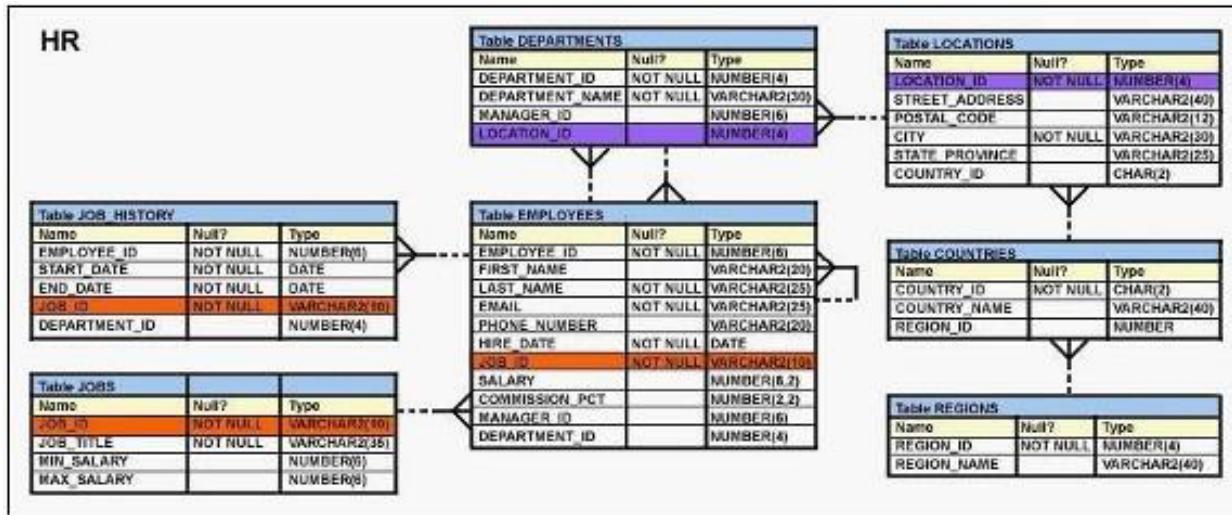
TABLES, VIEWS, and SEQUENCES

TABLES

Question:

Score 1 of 1

View the Exhibit and examine the description of the DEPARTMENTS and EMPLOYEES tables.



To retrieve data for all the employees for their EMPLOYEE_ID, FIRST_NAME, and DEPARTMENT NAME, the following SQL statement was written:

```

SELECT employee_id, first_name, department_name
FROM employees
NATURAL JOIN departments;
    
```

The desired output is not obtained after executing the above SQL statement. What could be the reason for this?

Response:

The NATURAL JOIN clause is missing the USING clause.

The DEPARTMENTS table is not used before the EMPLOYEES table in the FROM clause.

The EMPLOYEES and DEPARTMENTS tables have more than one column with the same column name and data type.

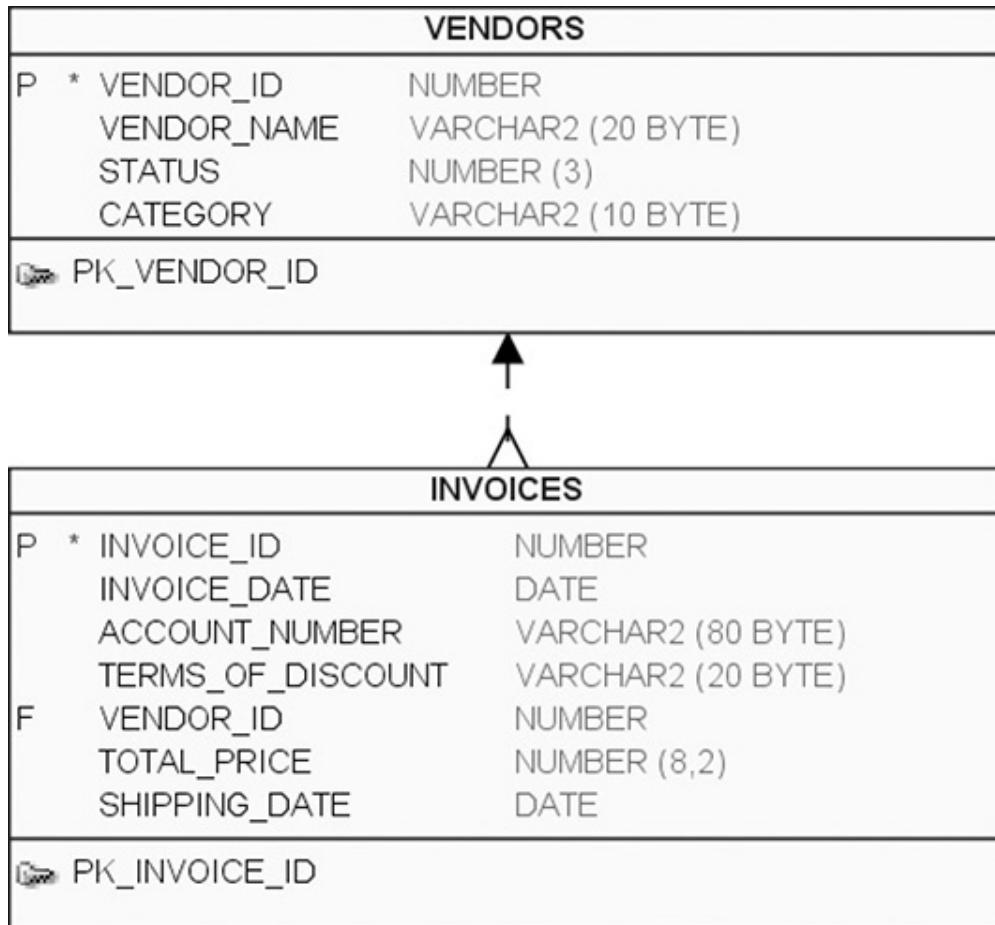
The table prefix is missing for the column names in the SELECT clause.

Question:

Score 1 of 1

Review the illustration. Which of the following is a syntactically correct outer join query?

(Choose two.)

**Response:**

```
SELECT VENDOR_NAME, INVOICE_DATE
FROM VENDORS OUTER JOIN INVOICES
ON VENDORS.VENDOR_ID = INVOICES.VENDOR_ID;
```

✓

```
SELECT VENDOR_NAME, INVOICE_DATE
FROM VENDORS LEFT JOIN INVOICES
ON VENDORS.VENDOR_ID = INVOICES.VENDOR_ID;
```

```
SELECT VENDOR_NAME, INVOICE_DATE
FROM VENDORS FULL OUTER JOIN INVOICES
ON VENDORS.VENDOR_ID = INVOICES.VENDOR_ID;
```

✓

```
SELECT VENDOR_NAME, INVOICE_DATE
FROM VENDORS RIGHT OUTER JOIN INVOICES
ON VENDORS.VENDOR_ID = INVOICES.VENDOR_ID;
```

Question:

Score 1 of 1

Which normal form is a table in if it has no multi-valued attributes and no partial dependencies?

Response:

First normal form



Second normal form

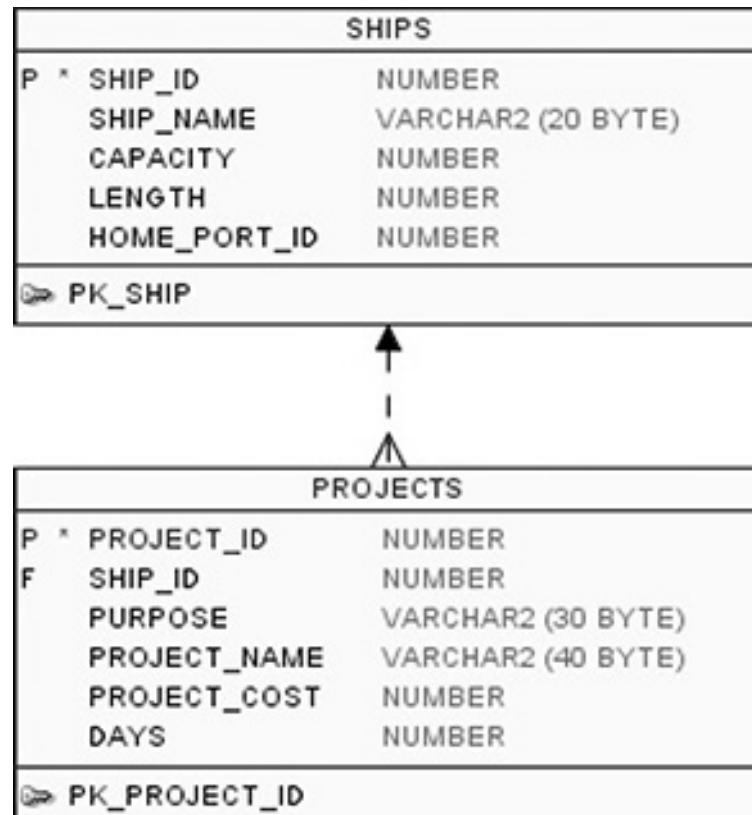
Fourth normal form

Third normal form

Question:

Score 1 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:**

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

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.

Question:

Score 1 of 1

Which of the following is true about aggregate functions?

(Choose two.)

Response:

Return one value for each group of rows specified in a SELECT statement.



Are also called group functions.

Will cause a run-time error when used in SELECT statements that return zero rows or one row.

Can operate only with numeric data.

Question:

Score 1 of 1

You attempt to execute the following SQL statement:

```
CREATE TABLE VENDORS
  (VENDOR_ID    NUMBER,
   VENDOR_NAME  VARCHAR2,
   CATEGORY     CHAR);
```

Which one of the following is true?**Response:**

The execution succeeds, and the table is created.

The execution fails because there is no precision indicated for CHAR.



The execution fails because there is no precision indicated for VARCHAR2.

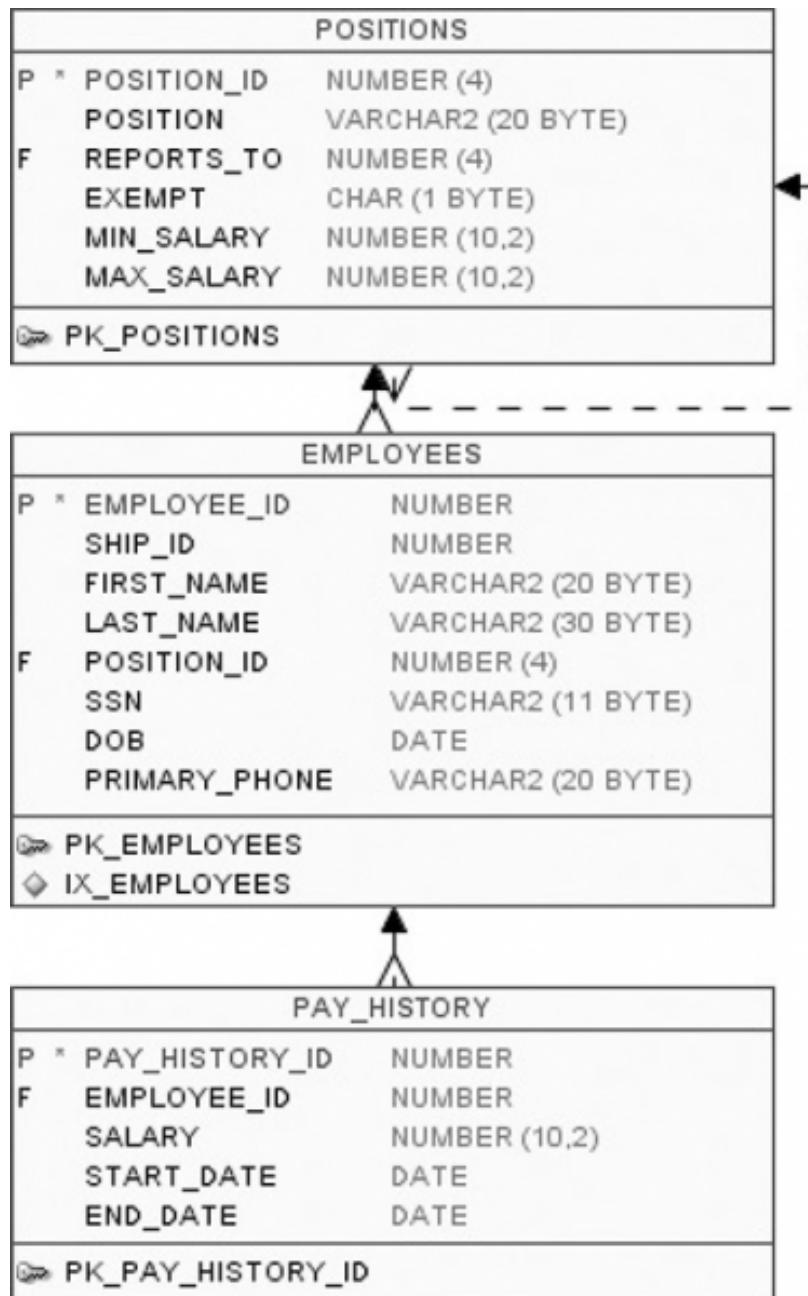
The execution fails because there is no precision indicated for NUMBER.

Question:

Score 1 of 1

Review the illustration. Which of the following is a valid self-join statement?

(Choose all that apply.)



Response:

```
SELECT P1.POSITION_ID, P1.MIN_SALARY, P1.MAX_SALARY
FROM   POSITIONS P1 SELF JOIN POSITIONS P2
ON     P1.REPORTS_TO = P2.POSITION_ID;
```

 SELECT P1.POSITION_ID, P1.MIN_SALARY, P1.MAX_SALARY
FROM POSITIONS P1 RIGHT OUTER JOIN POSITIONS P2
ON P1.REPORTS_TO = P2.POSITION_ID;

 SELECT P1.POSITION_ID, P1.MIN_SALARY, P1.MAX_SALARY
FROM POSITIONS P1 INNER JOIN POSITIONS P2
ON P1.REPORTS_TO = P2.POSITION_ID;

 SELECT P1.POSITION_ID, P1.MIN_SALARY, P1.MAX_SALARY
FROM POSITIONS P1 JOIN POSITIONS P2
ON P1.REPORTS_TO = P2.POSITION_ID;

Question:

Score 1 of 1

Examine the following query:

```
SQL> SELECT prod_id, amount_sold
  FROM sales
 ORDER BY amount_sold
  FETCH FIRST 5 PERCENT ROWS ONLY;
```

What is the output of this query?**Response:**

It results in an error because the ORDER BY clause should be the last clause.

It displays the first 5 percent of the rows from the SALES table.

It displays 5 percent of the products with the highest amount sold.



It displays 5 percent of the products with the lowest amount sold.

Question:

Score 1 of 1

Which three statements are true about multiple-row subqueries?**Response:**

They can return multiple columns as well as rows.



They can contain group functions and GROUP BY and HAVING clauses.

They cannot contain a subquery within a subquery

They can contain group functions and the GROUP BY clause, but not the HAVING clause.



They can contain a subquery within a subquery.

They can return only one column but multiple rows.

Question:

Score 1 of 1

You issued this command:

CHOOSE THREE SQL > DROP TABLE employees;

Which three statements are true?

Response:



The EMPLOYEES table may be moved to the recycle bin.

The EMPLOYEES table can be recovered using the ROLLBACK command.

Sequences used in the EMPLOYEES table become invalid.



All indexes and constraints defined on the table being dropped are also dropped.

The space used by the EMPLOYEES table is always reclaimed immediately.



If there is an uncommitted transaction in the session, it is committed.

Question:

Score 1 of 1

Review the following illustrations:

```
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

Next, review the following SQL code:

```

01   SELECT    TO_CHAR(A.LAST_ORDER, 'RRRR-MM-DD')
02   FROM      STORE_INVENTORY A
03   ORDER BY 1
04   UNION
05   SELECT    ADDED
06   FROM      FURNISHINGS;
```

What will result from an attempt to execute this SQL statement?

Response:



It will fail with a syntax error on line 3 because you cannot use an ORDER BY in this context.

It will fail because of the table alias in lines 1 and 2, which cannot be used in this context.

It will execute successfully.

It will fail with a syntax error because of the TO_CHAR conversion function on line 1.

Question:

Score 1 of 1

Review the following SQL statements:

```
CREATE TABLE INSTRUCTORS
  (INSTRUCTOR_ID NUMBER,
   EXEMPT      VARCHAR2(5),
   VACATION     NUMBER,
   PAY_RATE     NUMBER);
INSERT INTO INSTRUCTORS VALUES (1, 'YES', NULL, 25);
INSERT INTO INSTRUCTORS VALUES (2, NULL,  NULL, NULL);
UPDATE INSTRUCTORS
  SET EXEMPT      = 'YES',
      SET VACATION    = 15
 WHERE PAY_RATE < 50;
```

What can be said of the statements listed here?

Response:



At least one of the statements will not execute.

None of the above.

Two rows will be updated.

One row will be updated.

Question:

Score 1 of 1

Review the diagrams and examine the following statement:

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

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	

```

01  INSERT
02    WHEN (PART_NO < 500) THEN
03      INTO STORE_INVENTORY (NUM, PRODUCT)
04        VALUES (SPARE_ID, PART_NAME)
05      INTO PORT_INVENTORY (NUM, PRODUCT)
06        VALUES (SPARE_ID, PART_NAME)
07    WHEN (PART_NO >= 500) THEN
08      INTO SHIP_INVENTORY (NUM, PRODUCT)
09        VALUES (SPARE_ID, PART_NAME)
10    SELECT SPARE_ID, PART_NO, PART_NAME
11    FROM   SPARES;

```

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

Response:

If the first WHEN condition in line 2 is true, the INTO clause in line 3 and line 4 will be executed, after which processing will skip to the next row returned by the subquery.

If the first WHEN condition in line 2 is true, the WHEN condition in line 7 will not be evaluated.



Regardless of whether the first WHEN condition is true, the second WHEN condition will be evaluated.

No matter which WHEN condition is true, the INTO clause in line 5 will be executed regardless.

Question:

Score 1 of 1

When is a query considered a multirow subquery?

(Choose the best answer.)

Response:

If it returns numeric data, regardless of the number of rows of data it returns



If it returns multiple rows at the time of execution

If it may or may not return multiple rows, as determined by its WHERE clause

All of the above

Question:

Score 1 of 1

Which of the following SQL statements can always be executed on any VIEW object?

(Choose all that apply.)

Response:

DELETE

UPDATE

INSERT



SELECT

Question:

Score 1 of 1

Review this SQL statement: SELECT SUBSTR('2009',1,2) || LTRIM('1124','1') FROM DUAL; What will be the result of the SQL statement?

Response:

2024

20124

221

A syntax error

Question:

Score 1 of 1

Review the following illustration:

CRUISE_ORDERS		
P	*	CRUISE_ORDER_ID NUMBER
P	*	ORDER_DATE DATE
PK_CO		

Now review this SQL statement:

```
SELECT CRUISE_ORDER_ID, COUNT(ORDER_DATE)
FROM   CRUISE_ORDERS;
```

What can be said of this statement?**Response:**

It will execute successfully but not produce any meaningful output.

It will fail to execute because ORDER_DATE is a date data type, and no aggregate function can work with a date data type.

There is nothing wrong with the SQL statement.



It will fail to execute because it mixes scalar and aggregate data in the select list.

Question:

Score 1 of 1

View the Exhibit for the structure of the STUDENT and FACULTY tables.

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

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

Statement 1

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

Statement 2

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

Which statement is true regarding the outcome?

Response:

 Both statements 1 and 2 execute successfully and give the same required result.

Only statement 1 executes successfully and gives the required result.

Both statements 1 and 2 execute successfully and give different results.

Only statement 2 executes successfully and gives the required result.

Question:

Score 0 of 1

See the diagrams. You want to merge rows from the PORT_INVENTORY table into the SHIP_INVENTORY table. You start with the following SQL statement:

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

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

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

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

```

01 MERGE INTO SHIP_INVENTORY A
02 USING PORT_INVENTORY B
03 ON (A.NUM = B.NUM)
04 WHEN NOT MATCHED THEN INSERT
05     (A.NUM, A.AISLE, A.PRODUCT, A.LAST_ORDER)
06     VALUES
07     (B.NUM, B.AISLE, B.PRODUCT, B.LAST_ORDER)
08 WHERE TO_CHAR(A.LAST_ORDER, 'RRRR') = '2019';

```

What will this SQL statement do?

Response:

It will add rows from PORT_INVENTORY to SHIP_INVENTORY that do not already exist in SHIP_INVENTORY, regardless of the value for LAST_ORDER.

It will fail with a syntax error because you must have an ELSE clause.

 It will fail with a syntax error because you cannot reference the target table (SHIP_INVENTORY) in the WHERE clause in line 8.

 It will add rows from PORT_INVENTORY to SHIP_INVENTORY that do not already exist in SHIP_INVENTORY, limited to LAST_ORDER values from the year 2019.

Question:

Score 1 of 1

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

Response:

 INDEX

ROWID

VIEW

LOOKUP

Question:

Score 1 of 1

Examine the structure of the members table:

Name	Null?	Type
MEMBER_ID	NOT NULL	VARCHAR2 (6)
FIRST_NAME	NOT NULL	VARCHAR2 (50)
LAST_NAME		VARCHAR2 (50)
ADDRESS		VARCHAR2 (50)

You execute the SQL statement:

```
SQL> SELECT member_id, ' ', first_name, ' ', last_name "ID FIRSTNAME LASTNAME" FROM
members;
```

What is the outcome?

Response:

It fails because the space specified in single quotation marks after the first two column names is invalid.

 It executes successfully and displays the column details in three separate columns and replaces only the last column heading with the alias.

It fails because the alias name specified after the column names is invalid.

It executes successfully and displays the column details in a single column with only the alias column heading.

Question:

Score 1 of 1

Which of the following data dictionary views does not have an OWNER column?

Response:

USER_TABLES



DBA_CONS_COLUMNS

ALL_INDEXES

All of the above

Question:

Score 1 of 1

Which of the following is true about ROLES?

Response:

Roles are in the same namespace as TABLES.

Roles are in the same namespace as CONSTRAINTS.

Roles are schema objects but only when created from within a user account.



Roles are in the same namespace as USERS.

Question:

Score 1 of 1

What can be granted to a role?

(Choose all that apply.)

Response:

None of the above



Object privileges



System privileges

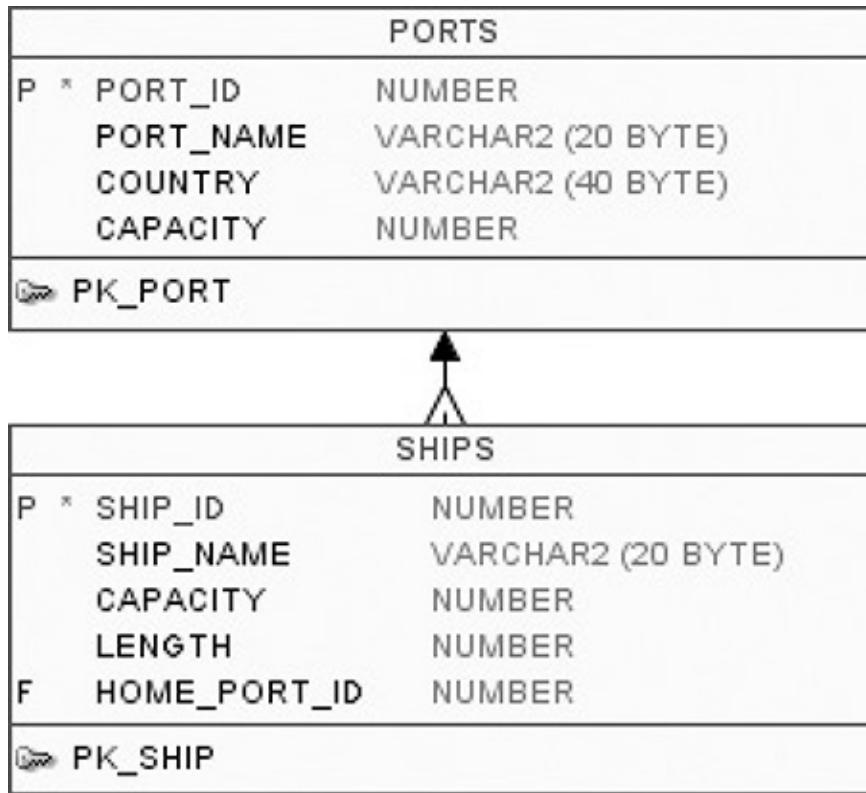


Roles

Question:

Score 1 of 1

Review the illustration and the following SQL code:



```

01  DELETE FROM PORTS P
02  WHERE  PORT_ID NOT EXISTS (SELECT PORT_ID
03                      FROM   SHIPS
04                      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 a syntax error on line 4.

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

It will execute successfully and perform as intended.

Question:

Score 1 of 1

Conversion functions cannot be used to:

Response:

Transform data

Format date values

 Convert columns to new data types

 Create user-defined data types

Question:

Score 1 of 1

Which two statements are true regarding the SQL GROUP BY clause?

Response:

 Using the WHERE clause before the GROUP BY clause excludes rows before creating groups.

You can use a column alias in the GROUP BY clause.

 if the SELECT clause has an aggregating function, then columns without an aggregating function in the SELECT clause should be included in the GROUP BY clause.

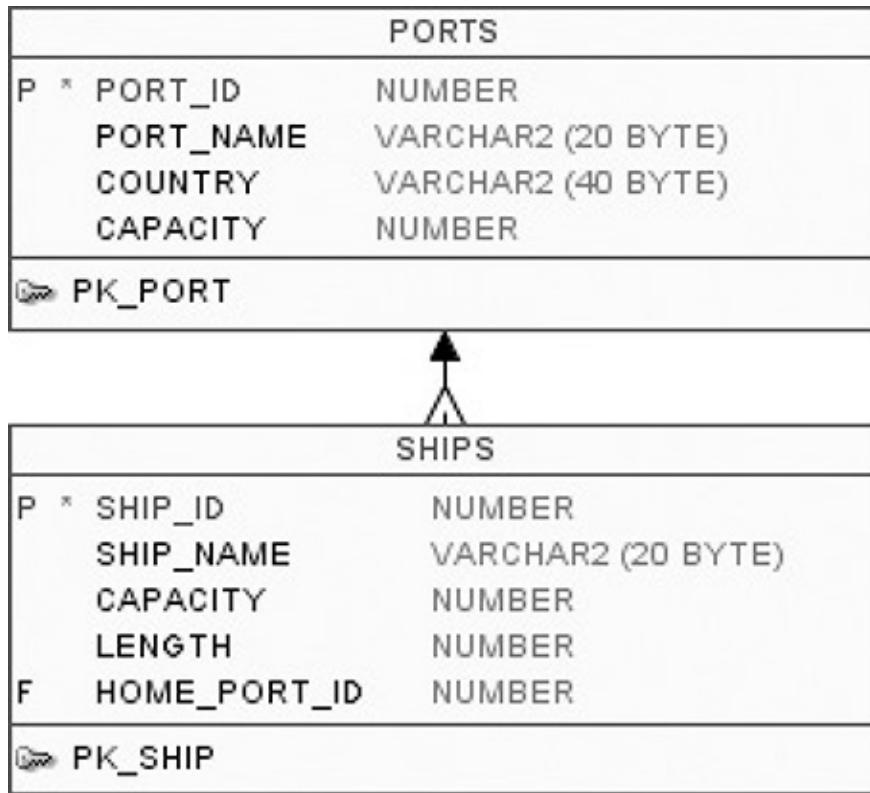
The GROUP BY clause is mandatory if you are using an aggregating function in the SELECT clause.

Using the WHERE clause after the GROUP BY clause excludes rows after creating groups.

Question:

Score 0 of 1

Review the illustration and the following SQL code:



```

01 UPDATE PORTS P
02 SET CAPACITY = CAPACITY + 1
03 WHERE EXISTS (SELECT *
04                   FROM SHIPS
05                   WHERE HOME_PORT_ID = P.PORT_ID);
  
```

The PORTS table has 15 rows. The SHIPS table has 20 rows. Each row in PORTS has a unique value for PORT_ID. Each PORT_ID value is represented in the HOME_PORT_ID column of at least one row of the SHIPS table.

What can be said of this UPDATE statement?

Response:



The value for CAPACITY will increase by 20 for each of the 15 rows in the PORTS table.

The statement will fail to execute because of an error in the syntax.

The value for CAPACITY will not increase.

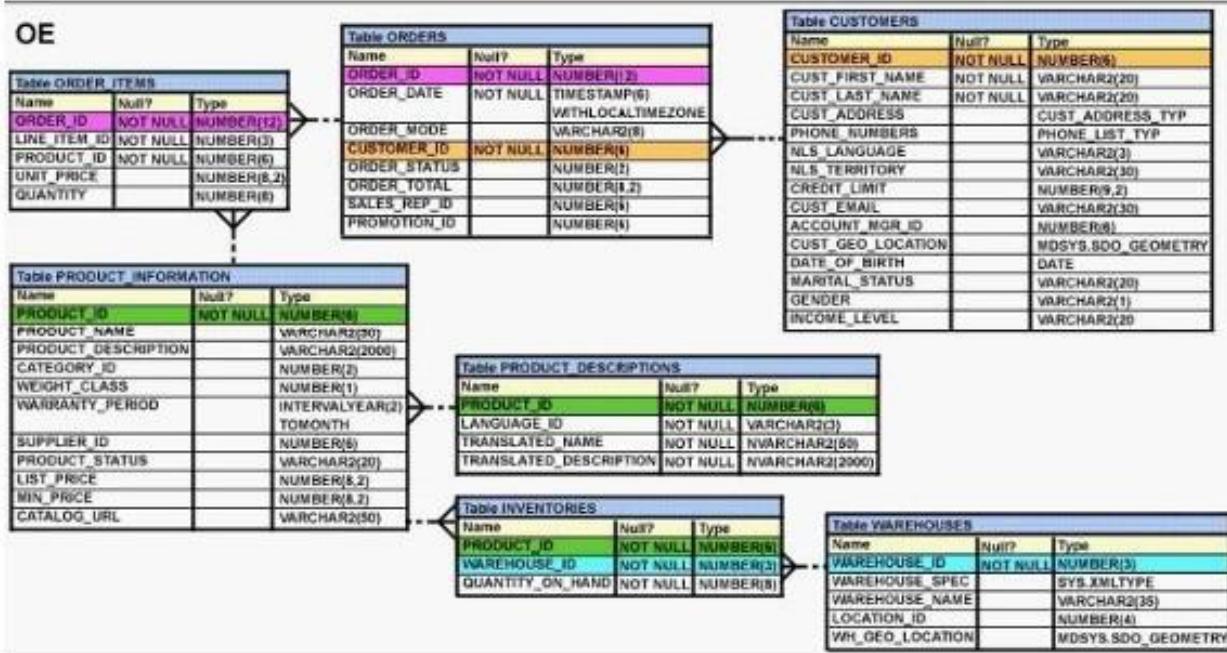


The value for CAPACITY will increase once for each of the 15 rows in the PORTS table.

Question:

Score 1 of 1

View the Exhibit and examine the description of the PRODUCT_INFORMATION table. Which SQL statement would retrieve from the table the number of products having LIST_PRICE as NULL?

**Response:**

```
SELECT COUNT (list_price)
FROM product_information
WHERE list_price i= NULL
```

```
SELECT COUNT (list_price)
FROM product_information
WHERE list_price is NULL
```

✓

```
SELECT COUNT (NVL(list_price, 0))
FROM product_information
WHERE list_price is NULL
```

```
SELECT COUNT (DISTINCT list_price)
FROM product_information
WHERE list_price is NULL
```

Question:

Score 1 of 1

Review the following SQL statement:

```
CREATE TABLE shipping_Order
( order_ID      NUMBER,
  order_Year    CHAR(2),
  customer_ID  NUMBER,
  CONSTRAINT shipping_Order PRIMARY KEY (order_ID, order_Year));
```

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

Response:



The statement will succeed: the table will be created, and the primary key will also be created.

The table will be created, but the primary key constraint will not be created because the name does not include the _PK suffix.

The statement will fail because the data type for ORDER_YEAR is a CHAR, and CHAR data types aren't allowed in a PRIMARY KEY constraint.

The statement will fail because there is no precision for the ORDER_ID column's data type.

Question:

Score 1 of 1

Which of the following data dictionary views contains information about grants on tables that have been made by other users to your user account, as well as grants on tables that have been made by your user account to other user accounts?

Response:

USER_TABLES

USER_TAB_COLUMNS

ALL_TAB_PRIVS_REC



USER_TAB_PRIVS

Question:

Score 1 of 1

Which of the following keywords cannot be used with the CREATE SEQUENCE statement?

Response:

MAXVALUE



JOIN

CYCLE

INCREMENT

Question:

Score 1 of 1

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

DEPT			
Name	Null?	Type	
DEPARTMENT_ID	NOT NULL	NUMBER(4)	
DEPARTMENT_NAME		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 = ALL (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 = ANY (SELECT city FROM locations l);
```

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

Question:

Score 1 of 1

Which of the following is the system privilege that is required as a minimum to allow a user account to log in to the database?

Response:

```
CREATE ANY LOGIN
```

```
CREATE TABLE
```

```
CREATE ANY SESSION
```

```
CREATE SESSION
```

Question:

Score 1 of 1

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

PROJECTS	
P *	PROJECT_ID
	NUMBER
	SHIP_ID
	NUMBER
	PURPOSE
	VARCHAR2 (30 BYTE)
	PROJECT_NAME
	VARCHAR2 (40 BYTE)
	PROJECT_COST
	NUMBER
	DAYS
	NUMBER
PK	PROJECT_ID

```

01   SELECT   COUNT(COUNT(PROJECT_COST))
02   FROM     PROJECTS
03   GROUP BY PURPOSE;

```

What will happen if you try to execute this query on the PROJECTS table?

Response:

It will succeed and display one row.

It will fail with a syntax error because line 1 is not correct.

It will fail with an execution error because you cannot use a VARCHAR2 column in a GROUP BY clause.

It will succeed and display one row for each different value in the PURPOSE column.

Question:

Score 1 of 1

Which statement is true about Data Manipulation Language (DML)?

Response:

Each DML statement forms a transaction by default.

DML automatically disables foreign key constraints when modifying primary key values in the parent table.



A transaction can consist of one or more DML statements.

DML disables foreign key constraints when deleting primary key values in the parent table, only when the ON DELETE CASCADE option is set for the foreign key

constraint.

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:

ALL_TAB_COLUMNS

ALL_COLUMNS

DBA_TAB_COLUMNS

Can't be done

Question:**Score 1 of 1**

Evaluate the following two queries:

```
SQL> SELECT cust_last_name, cust_city  
FROM customers  
WHERE cust_credit_limit IN (1000, 2000, 3000);
```

```
SQL> SELECT cust_last_name, cust_city  
FROM customers  
WHERE cust_credit_limit = 1000 or cust_credit_limit = 2000 or  
cust_credit_limit = 3000
```

Which statement is true regarding the above two queries?

Response:

Performance would improve in query 2.

There would be no change in performance.

Performance would improve in query 2 only if there are null values in the CUST_CREDIT_LIMIT column.

Performance would degrade in query 2.

Question:**Score 1 of 1**

Which statement is true about transactions?**Response:**

Each Data Definition Language (DDL) statement executed forms a single transaction.

A set of Data Manipulation Language (DML) statements executed in a sequence ending with a SAVEPOINT forms a single transaction.

A set of DDL statements executed in a sequence ending with a COMMIT forms a single transaction.

A combination of DDL and DML statements executed in a sequence ending with a COMMIT forms a single transaction.

Question:

Score 1 of 1

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 DICTIONARY;

SELECT * FROM V\$RESERVED_WORDS;

SELECT * FROM GV\$_START_HERE;

SELECT * FROM V\$DATABASE;

Question:

Score 1 of 1

Review the following series of SQL statements:

```

CREATE TABLE SUPPLIES_01
(
    SUPPLY_ID NUMBER(7),
    SUPPLIER  VARCHAR2(30),
    ACCT_NO   VARCHAR2(50));
CREATE INDEX IX_SU_01 ON SUPPLIES_01(ACCT_NO);
DROP TABLE SUPPLIES_01;
CREATE TABLE SUPPLIES_02
(
    SUPPLY_ID NUMBER(7),
    SUPPLIER  VARCHAR2(30),
    ACCT_NO   VARCHAR2(50));
CREATE INDEX IX_SU_02 ON SUPPLIES_02(ACCT_NO, SUPPLIER);

```

Assuming there are no objects already in existence named SUPPLIES_01 or SUPPLIES_02 prior to the execution of the preceding statements, what database objects will result from these statements?

Response:

A table called SUPPLIES_02 and two indexes called IX_SU_01 and IX_SU_02

None of the above

A table called SUPPLIES_02 and nothing else



A table called SUPPLIES_02 and an index called IX_SU_02

Question:

Score 1 of 1

Which of the following symbols is most likely to be used in a SELECT statement using a non-equijoin?

Response:

<>

None of the above

!=



<=

Question:

Score 1 of 1

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

SQL>SELECT * FROM books_transactions ORDER 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 in the order that they are stored in the table only for the three rows with the lowest values in the key column.



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

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

Question:

Score 1 of 1

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

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



The statements will execute successfully and perform as intended.

Question:

Score 1 of 1

Using the CUSTOMERS table, you need to generate a report that shows 50% of each credit amount in each income level. The report should NOT show any repeated credit amounts in each income level.

Which query would give the required result?

Response:



SELECT DISTINCT cust_income_level || ' ' || cust_credit_limit * 0.50 AS "50% Credit Limit" FROM customers;

SELECT DISTINCT cust_income_level, DISTINCT cust_credit_limit * 0.50 AS '50% Credit Limit' FROM customers, IT;

SELECT cust_income_level, DISTINCT cust_credit_limit * 0.50 AS '50% Credit Limit' FROM customers;

SELECT cust_income_level || ' ' || cust_credit_limit * 0.50 AS '50% Credit Limit'
FROM customers;

Question:

Score 1 of 1

The following are the steps for a correlated subquery, listed in random order:

- 1) The WHERE clause of the outer query is evaluated.
- 2) The candidate row is fetched from the table specified in the outer query.
- 3) The procedure is repeated for the subsequent rows of the table, till all the rows are processed.
- 4) Rows are returned by the inner query, after being evaluated with the value from the candidate row in the outer query.

Identify the option that contains the steps in the correct sequence in which the Oracle server evaluates a correlated subquery.

Response:



2,4,1,3

2,1,4,3

4,1,2,3

4,2,1,3

Question:

Score 1 of 1

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 the first primary key but not the second.

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



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

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

Question:**Score 1 of 1****Which of the following is a true statement?****Response:**

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

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

Question:**Score 1 of 1****The DECODE expression always ends with:****Response:**

Both of the above

-  A default expression to return if no other value matched the source expression

The keyword END

Neither of the above

Question:

Score 1 of 1

User HARDING owns a table TEAPOT. User HARDING then executes the following SQL statements to give access to the table to user ALBERT:

```
CREATE PUBLIC SYNONYM TEAPOT FOR HARDING.TEAPOT;
CREATE ROLE DOME;
GRANT DOME TO ALBERT;
GRANT SELECT ON TEAPOT TO DOME;
```

Which of the following statements can user ALBERT now execute on the TEAPOT table?

Response:

SELECT * FROM DOME.HARDING.TEAPOT;

None of the above

-  SELECT * FROM HARDING.TEAPOT;

SELECT * FROM HARDING.DOME.TEAPOT;

Question:

Score 1 of 1

In which three situations does a transaction complete?

Response:

-  when a data definition language (DDL) statement is executed
-  when a TRUNCATEstatement is executed after the pending transaction
-  when a ROLLBACKcommand is executed

when a DELETEstatement is executed

when a PL/SQL anonymous block is executed

Question:

Score 1 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

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	

```

01  SELECT '--', SECTION
02  FROM  FURNISHINGS
03  WHERE  CAT# NOT IN (1,2)
04  UNION ALL
05  SELECT TO_CHAR(LAST_ORDER, 'Month'), AISLE
06  FROM  STORE_INVENTORY;

```

How many rows will result from this query?**Response:**

It will not execute because it will fail with a syntax error.

6



4

0

Question:

Score 1 of 1

Evaluate the following SQL statements that are issued in the given order:

```

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

```

What would be the status of the foreign key EMP_MGR_FK?**Response:**

It would remain disabled and has to be enabled manually using the ALTER TABLE command.

It would be automatically enabled and immediate.

It would remain disabled and can be enabled only by dropping the foreign key constraint and recreating it.

It would be automatically enabled and deferred.

Question:

Score 1 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

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

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



It will execute successfully.

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.

Question:

Score 1 of 1

Which one of the following is a DML statement?

Response:



UPDATE

ADD

MODIFY

ALTER

Question:

Score 0 of 1

You issue this command which succeeds:

SQL> DROP TABLE products;

Which three statements are true?

Response:



Any uncommitted transaction in the session is committed.

Table data is deleted but the table structure is retained.



All existing views and synonyms that refer to the table are invalidated but retained.



All the table's indexes if any exist, are invalidated but retained.



Table data and the table structure are deleted.

Question:

Score 0 of 1

You want to display 5 percent of the rows from the sales table for products with the lowestAMOUNT_SOLD and also want to include the rows that have the sameAMOUNT_SOLD even if this causes the output to exceed 5 percent of the rows.

Which query will provide the required result?

Response:



SELECT prod_id, cust_id, amount_sold FROM sales
ORDER BY amount_sold
FETCH FIRST 5 PERCENT ROWS WITH TIES ONLY;

SELECT prod_id, cust_id, amount_sold FROM sales
ORDER BY amount_sold
FETCH FIRST 5 PERCENT ROWS ONLY WITH TIES;



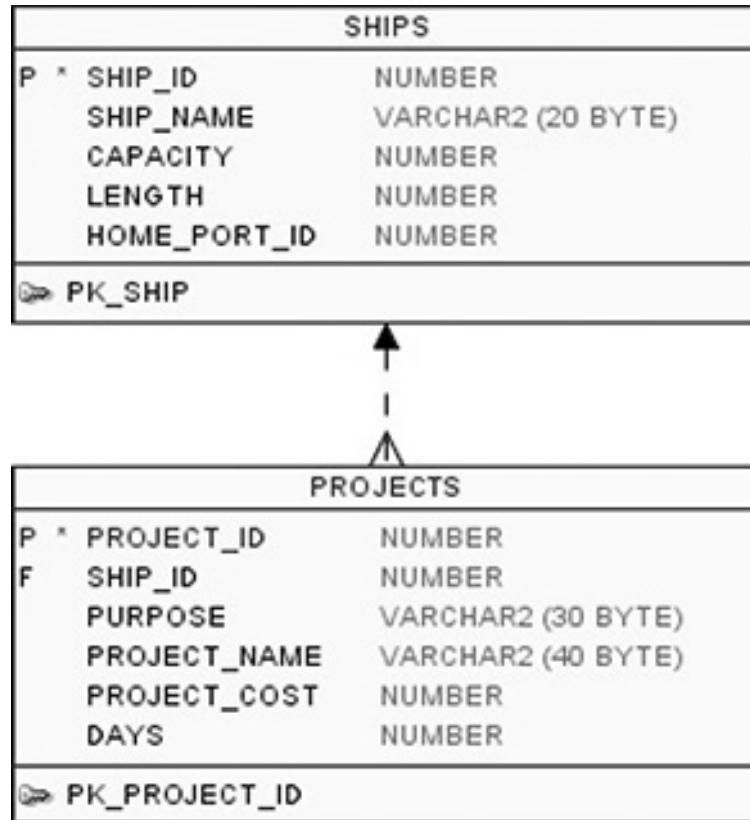
SELECT prod_id, cust_id, amount_sold FROM sales
ORDER BY amount_sold
FETCH FIRST 5 PERCENT ROWS WITH TIES;

SELECT prod_id, cust_id, amount_sold FROM sales
ORDER BY amount sold

FETCH FIRST 5 PERCENT ROWS ONLY;

Question:

Score 1 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 INSERT statement will fail because of an error on lines 9 and 10.



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 CREATE statement will fail because it omits the PURPOSE column from the PROJECTS table.

Question:

Score 1 of 1

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

Response:

-  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.

Nothing.

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.

An UNUSED column can be recovered.

Question:

Score 0 of 1

A role:**Response:**

-  Can be created by a user only if that user has the CREATE ROLE system privilege

-  Takes the place of privileges automatically so that any privilege granted to a role supersedes any grants that have already been granted directly to a user

Can be granted to a user, who can be granted only one role at a time

Cannot be given the same name as a table

Question:

Score 1 of 1

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

Question:

Score 1 of 1

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 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.

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

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

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

Question:

Score 1 of 1

The CASCADE keyword, when used with TRUNCATE:**Response:**

Is required if the table has any dependent child tables



None of the above

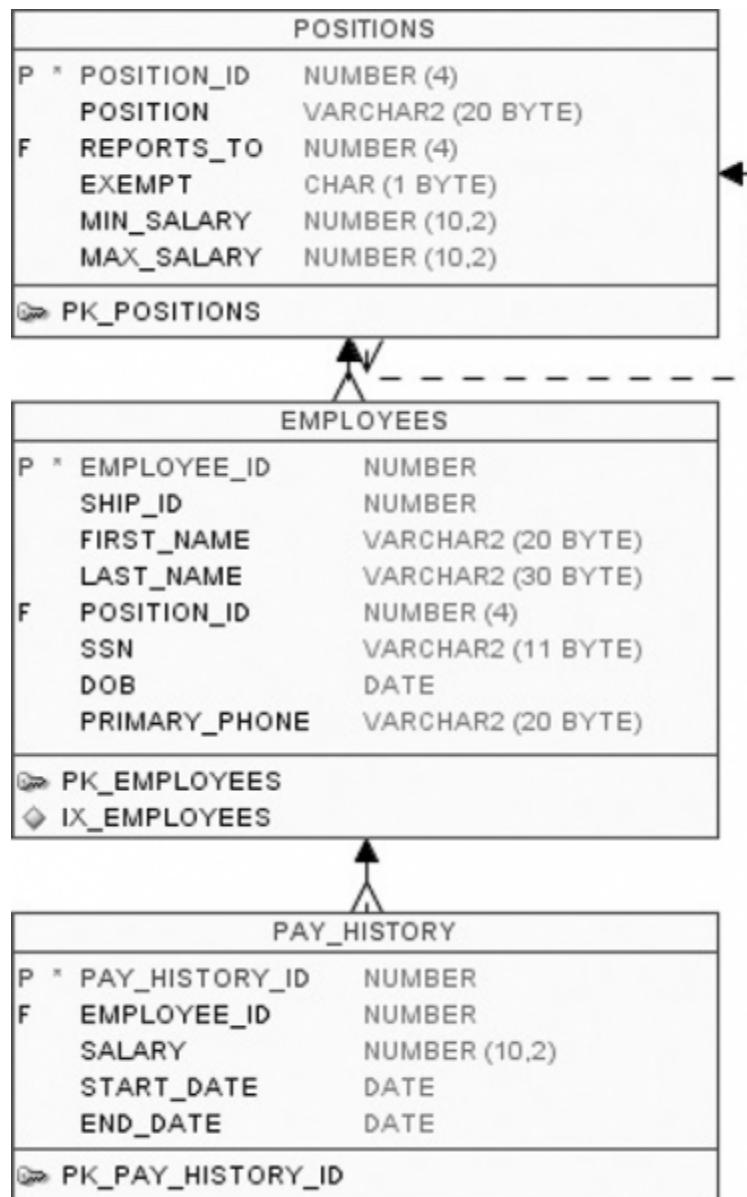
Will ensure that future attempts to insert rows to the table will be rejected if they satisfy the TRUNCATE table's WHERE clause

Can be used with the optional DEPENDENCY keyword

Question:

Score 1 of 1

Review the POSITIONS, EMPLOYEES, and PAY_HISTORY tables.



Review the following SQL statement:

```
SELECT LAST_NAME, POSITION, SALARY
FROM   POSITIONS P JOIN EMPLOYEES E ON P.POSITION_ID = E.POSITION_ID
          JOIN PAY_HISTORY PH ON E.EMPLOYEE_ID = PH.EMPLOYEE_ID;
```

Which of the following is true for the SQL statement?

(Choose two.)

Response:

It is an outer join.



It will execute successfully.



It connects three tables.

It will fail because there are no table aliases.

Question:

Score 1 of 1

The unique identifier of a row in a database table is a(n):**Response:**

ID

Column

Primary column



Primary key

Question:

Score 1 of 1

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	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 D

Option C

Option B

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

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

STORE_INVENTORY	
P *	NUM
	NUMBER
	AISLE
	VARCHAR2 (7 BYTE)
	PRODUCT
	VARCHAR2 (15 BYTE)
	LAST_ORDER
	DATE
PK	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	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 fail with an execution error on line 1.



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

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

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

Question:

Score 1 of 1

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		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?

Response:



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

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

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

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

[Money Back Guarantee](#) | [Testimonial](#) | [FAQs](#) | [Privacy Policy](#) | [Terms and Conditions](#) | [About Us](#) | [Join Us](#) |
[Contact Us](#)