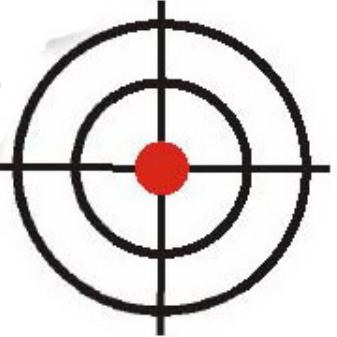


T  
R  
O  
Y  
T  
E  
C  
H  
E  
C  
H  
O  
N  
O  
G  
I  
E  
S

Troy Tec



TM

Oracle Database 11g : SQL  
Fundamentals I Exam  
Exam: 1Z0-051

*Edition: 6.0*

© 2013 - 2014 Test Killer, LTD All Rights Reserved

**QUESTION: 1**

Evaluate the SQL statement: TRUNCATE TABLE DEPT; Which three are true about the SQL statement? (Choose three.)

- A. It releases the storage space used by the table.
- B. It does not release the storage space used by the table.
- C. You can roll back the deletion of rows after the statement executes.
- D. You can NOT roll back the deletion of rows after the statement executes.
- E. An attempt to use DESCRIBE on the DEPT table after the TRUNCATE statement executes will display an error.
- F. You must be the owner of the table or have DELETE ANY TABLE system privileges to truncate the DEPT table

**Answer:** A, D, F

**Explanation:**

A: The TRUNCATE TABLE Statement releases storage space used by the table, D: Can not rollback the deletion of rows after the statement executes, F: You must be the owner of the table or have DELETE ANY TABLE system privilege to truncate the DEPT table. Incorrect answer: is not true is not true is not true Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 8-18

**QUESTION: 2**

You need to design a student registration database that contains several tables storing academic information. The STUDENTS table stores information about a student. The STUDENT\_GRADES table stores information about the student's grades. Both of the tables have a column named STUDENT\_ID. The STUDENT\_ID column in the STUDENTS table is a primary key. You need to create a foreign key on the STUDENT\_ID column of the STUDENT\_GRADES table that points to the STUDENT\_ID column of the STUDENTS table. Which statement creates the foreign key?

- A. CREATE TABLE student\_grades (student\_id NUMBER(12), semester\_end DATE, gpa NUMBER(4,3), CONSTRAINT student\_id\_fk REFERENCES (student\_id) FOREIGN KEY students(student\_id));
- B. CREATE TABLE student\_grades(student\_id NUMBER(12), semester\_end DATE, gpa NUMBER(4,3), student\_id\_fk FOREIGN KEY (student\_id) REFERENCES students(student\_id));
- C. CREATE TABLE student\_grades(student\_id NUMBER(12), semester\_end DATE, gpa NUMBER(4,3), CONSTRAINT FOREIGN KEY (student\_id) REFERENCES students(student\_id));

D. CREATE TABLE student\_grades(student\_id NUMBER(12),semester\_end DATE, gpa NUMBER(4,3), CONSTRAINT student\_id\_fk FOREIGN KEY (student\_id) REFERENCES students(student\_id));

**Answer:** D

**Explanation:**

CONSTRAINT name FOREIGN KEY (column\_name) REFERENCES table\_name (column\_name); Incorrect answer: invalid syntax invalid syntax invalid syntax Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 10-14

**QUESTION:** 3

Here is the structure and data of the CUST\_TRANS table: Exhibit:

CUST_TRANS		
Name	Null?	Type
CUSTNO	NOT NULL	CHAR(2)
TRANSDATE	DATE	
TRANSAMT		NUMBER(6,2)
CUSTNO	TRANSDATE	TRANSAMT
11	01-JAN-07	1000
22	01-FEB-07	2000
33	01-MAR-07	3000

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

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

**Answer:** A, C, D

**QUESTION:** 4

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

INVOICE			
Name	Null?	Type	
INV_NO	NOT NULL	NUMBER(3)	
INV_DATE		DATE	
CUST_ID		VARCHAR2(4)	
INV_AMT		NUMBER(8,2)	
<hr/>			
INV_NO	INV_DATE	CUST_ID	INV_AMT
1	01-APR-07	A1Q	1000
2	01-OCT-07	B1R	2000
3	01-FEB-07		3000

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

- A. SELECT MAX(inv\_date),MIN(cust\_id) FROM invoice;
- B. SELECT MAX(AVG(SYSDATE - inv\_date)) FROM invoice;
- C. SELECT (AVG(inv\_date)) FROM invoice;
- D. SELECT AVG(inv\_date - SYSDATE),AVG(inv)amt) FROM invoice;

**Answer:** A, D

### QUESTION: 5

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

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

**Answer:** A, B, C

### QUESTION: 6

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

Table CUSTOMERS		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2 (20)
CUST_LAST_NAME	NOT NULL	VARCHAR2 (40)
CUST_GENDER	NOT NULL	CHAR (1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER (4)
CUST_MARITAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

Using the CUSTOMERS table, you need to generate a report that shown the average credit limit for customers in WASHINGTON and NEW YORK. Which SQL statement would produce the required result?

A.

```
SELECT cust_city, AVG(cust_credit_limit) FROM customers
WHERE cust_city IN ('WASHINGTON','NEW YORK') GROUP BY
cust_credit_limit, cust_city;
```

B.

```
SELECT cust_city, AVG(cust_credit_limit) FROM customers
WHERE cust_city IN ('WASHINGTON','NEW YORK') GROUP BY
cust_city,cust_credit_limit;
```

C.

```
SELECT cust_city, AVG(cust_credit_limit) FROM customers
WHERE cust_city IN ('WASHINGTON','NEW YORK') GROUP BY cust_city;
```

D.

```
SELECT cust_city, AVG(NVL(cust_credit_limit,0)) FROM customers
WHERE cust_city IN ('WASHINGTON','NEW YORK');
```

**Answer:** C

### **Explanation:**

Creating Groups of Data: GROUP BY Clause Syntax You can use the GROUP BY clause to divide the rows in a table into groups. You can then use the group functions to return summary information for each group. In the syntax: group\_by\_expression Specifies the columns whose values determine the basis for grouping rows Guidelines

- If you include a group function in a SELECT clause, you cannot select individual results as well, unless the individual column appears in the GROUP BY clause. You receive an error message if you fail to include the column list in the GROUP BY clause.
- Using a WHERE clause, you can exclude rows before dividing them into groups.

- You must include the columns in the GROUP BY clause.
- You cannot use a column alias in the GROUP BY clause.

**QUESTION: 7**

Evaluate these two SQL statements: `SELECT last_name, salary, hire_date FROM EMPLOYEES ORDER BY salary DESC;` `SELECT last_name, salary, hire_date FROM EMPLOYEES ORDER BY 2 DESC;` What is true about them?

- A. The two statements produce identical results.
- B. The second statement returns a syntax error.
- C. There is no need to specify DESC because the results are sorted in descending order by default.
- D. The two statements can be made to produce identical results by adding a column alias for the salary column in the second SQL statement.

**Answer:** A**Explanation:**

the two statement produce identical results as ORDER BY 2 will take the second column as sorting column. Incorrect answer: there is no syntax error result are sorted in ascending order by default ORDER BY 2 will take the second column as sorting column. Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 2-22

**QUESTION: 8**

Where can sub queries be used? (Choose all that apply)

- A. field names in the SELECT statement
- B. the FROM clause in the SELECT statement
- C. the HAVING clause in the SELECT statement
- D. the GROUP BY clause in the SELECT statement
- E. the WHERE clause in only the SELECT statement
- F. the WHERE clause in SELECT as well as all DML statements

**Answer:** A, B, C, F**Explanation:**

SUBQUERIES can be used in the SELECT list and in the FROM, WHERE, and HAVING clauses of a query. A subquery can have any of the usual clauses for selection and projection. The following are required clauses:

- A SELECT list

■ A FROM clause

The following are optional clauses:

- WHERE
- GROUP BY
- HAVING

The subquery (or subqueries) within a statement must be executed before the parent query that calls it, in order that the results of the subquery can be passed to the parent.

**QUESTION: 9**

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

A.

SELECT TO\_CHAR(1890.55,'\$99G999D00') FROM DUAL;

B.

SELECT TO\_CHAR(1890.55,'\$9,999V99') FROM DUAL;

C.

SELECT TO\_CHAR(1890.55,'\$0G000D00') FROM DUAL;

D.

SELECT TO\_CHAR(1890.55,'\$99G999D99') FROM DUAL;

E.

SELECT TO\_CHAR(1890.55,'\$9,999D99') FROM DUAL;

**Answer:** A, C, D

**QUESTION: 10**

Evaluate the following SQL statement:

```
SQL> SELECT promo_id, promo_category
  FROM promotions
 WHERE promo_category = 'Internet' ORDER BY 2 DESC
 UNION
 SELECT promo_id, promo_category
  FROM promotions
 WHERE promo_category = 'TV'
 UNION
 SELECT promo_id, promo_category
  FROM promotions
 WHERE promo_category ='Radio';
```

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

- A. It produces an error because the ORDER BY clause should appear only at the end of a compound query—that is, with the last SELECT statement
- B. It executes successfully and displays rows in the descending order of PROMO\_CATEGORY
- C. It executes successfully but ignores the ORDER BY clause because it is not located at the end of the compound statement
- D. It produces an error because positional notation cannot be used in the ORDER BY clause with SET operators

**Answer:** A

**Explanation:**

Using the ORDER BY Clause in Set Operations The ORDER BY clause can appear only once at the end of the compound query. Component queries cannot have individual ORDER BY clauses. The ORDER BY clause recognizes only the columns of the first SELECT query. By default, the first column of the first SELECT query is used to sort the output in an ascending order.

**QUESTION: 11**

Which statement correctly describes SQL and /SQL\*Plus?

- A. Both SQL and /SQL\*plus allow manipulation of values in the database.
- B. /SQL\*Plus recognizes SQL statements and sends them to the server; SQL is the Oracle proprietary interface for executing SQL statements.
- C. /SQL\*Plus is a language for communicating with the Oracle server to access data; SQL recognizes SQL statements and sends them to the server.
- D. SQL manipulates data and table definitions in the database; /SQL\*Plus does not allow manipulation of values in the database.

**Answer:** A

**QUESTION: 12**

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

- A. string
- B. character
- C. integer
- D. calendar
- E. numeric
- F. translation

- G. date
- H. conversion

**Answer:** B, E, G, H

**Explanation:**

SQL have character, numeric, date, conversion function. Incorrect answer: SQL have character, numeric, date, conversion function. Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 3-3

**QUESTION:** 13

Examine the structure of the EMPLOYEES and NEW\_EMPLOYEES tables:

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

NEW_EMPLOYEES
EMPLOYEE_ID NUMBER Primary Key
NAME VARCHAR2(60)
```

Which MERGE statement is valid?

A.

```
MERGE INTO new_employees c
USING employees e
ON (c.employee_id = e.employee_id) WHEN MATCHED THEN
UPDATE SET
C.name = e.first_name ||','|| e.last_name
WHEN NOT MATCHED THEN INSERT
value
S(e.employee_id, e.first_name ||',
'||e.last_name);
```

B.

```
MERGE new_employees c
USING employees e
ON (c.employee_id = e.employee_id) WHEN EXISTS THEN
UPDATE SET
```

```

C.name = e.first_name ||','|| e.last_name WHEN NOT MATCHED THEN INSERT
valueS(e.employee_id, e.first_name ||',
'||e.last_name);
C.
MERGE INTO new_employees cUSING employees e
ON (c.employee_id = e.employee_id) WHEN EXISTS THEN
UPDATE SET
C.name = e.first_name ||','|| e.last_name
WHEN NOT MATCHED THEN INSERT
value
S(e.employee_id, e.first_name ||',
'||e.last_name);
D.
MERGE new_employees c
FROM employees e ON (c.employee_id = e.employee_id) WHEN MATCHED THEN
UPDATE SET
c.name = e.first_name ||','|| e.last_name
WHEN NOT MATCHED THEN INSERT INTO
new_employees valueS(e.employee_id, e.first_name ||',
'||e.last_name);

```

**Answer:** A

**Explanation:**

this is the correct MERGE statement syntax Incorrect answer: it should MERGE INTO table\_name it should be WHEN MATCHED THEN it should MERGE INTO table\_name Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 8-29

**QUESTION: 14**

Which view should a user query to display the columns associated with the constraints on a table owned by the user?

- A. USER\_CONSTRAINTS
- B. USER\_OBJECTS
- C. ALL\_CONSTRAINTS
- D. USER\_CONS\_COLUMNS
- E. USER\_COLUMNS

**Answer:** D

**Explanation:**

view the columns associated with the constraint names in the USER\_CONS\_COLUMNS view. Incorrect answer: table to view all constraints definition and names show all object name belong to user does not display column associated no such view Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 10-25

**QUESTION: 15**

The COMMISSION column shows the monthly commission earned by the employee. Exhibit

EMP_ID	DEPT_ID	COMMISSION
1	10	500
2	20	1000
3	10	
4	10	600
5	30	800
6	30	200
7	10	
8	20	300

Which two tasks would require sub queries or joins in order to be performed in a single step? (Choose two.)

- A. listing the employees who earn the same amount of commission as employee 3
- B. finding the total commission earned by the employees in department 10
- C. finding the number of employees who earn a commission that is higher than the average commission of the company
- D. listing the departments whose average commission is more than 600
- E. listing the employees who do not earn commission and who are working for department 20 in descending order of the employee ID
- F. listing the employees whose annual commission is more than 6000

**Answer:** A, C

**QUESTION: 16**

Examine the structure of the STUDENTS table:

STUDENT_ID	NUMBER	NOT NULL, Primary Key
STUDENT_NAME	VARCHAR2(30)	
COURSE_ID	VARCHAR2(10)	NOT NULL
MARKS	NUMBER	
START_DATE	DATE	
FINISH_DATE	DATE	

You need to create a report of the 10 students who achieved the highest ranking in the course INT SQL and who completed the course in the year 1999. Which SQL statement accomplishes this task?

- A. SELECT student\_id, marks, ROWNUM "Rank" FROM students  
WHERE ROWNUM <= 10  
AND finish\_date BETWEEN '01-JAN-99' AND '31-DEC-99'  
AND course\_id = 'INT\_SQL' ORDER BY marks DESC;
- B. SELECT student\_id, marks, ROWID "Rank" FROM students  
WHERE ROWID <= 10  
AND finish\_date BETWEEN '01-JAN-99' AND '31-DEC-99' AND course\_id =  
'INT\_SQL'  
ORDER BY marks;
- C. SELECT student\_id, marks, ROWNUM "Rank" FROM (SELECT student\_id,  
marks  
FROM students  
WHERE ROWNUM <= 10  
AND finish\_date BETWEEN '01-JAN-99' AND '31-DEC-  
99'  
AND course\_id = 'INT\_SQL' ORDER BY marks DESC);
- D. SELECT student\_id, marks, ROWNUM "Rank" FROM (SELECT student\_id,  
marks  
FROM students  
WHERE (finish\_date BETWEEN '01-JAN-99' AND '31-DEC-99') AND course\_id =  
'INT\_SQL'  
ORDER BY marks DESC) WHERE ROWNUM <= 10 ;
- E. SELECT student\_id, marks, ROWNUM "Rank" FROM (SELECT student\_id,  
marks  
FROM students  
ORDER BY marks) WHERE ROWNUM <= 10  
AND finish\_date BETWEEN '01-JAN-99' AND '31-DEC-99' AND course\_id =  
'INT\_SQL';

**Answer:** D

**QUESTION: 17**

Evaluate the following SQL statements: Exhibit:

```
CREATE TABLE orders
(ord_no NUMBER(2) CONSTRAINT ord_pk PRIMARY KEY,
ord_date DATE,
cust_id NUMBER(4));
```

Exhibit:

```
CREATE TABLE ord_items
(ord_no NUMBER(2),
item_no NUMBER(3),
qty NUMBER(3) CHECK (qty BETWEEN 100 AND 200),
expiry_date date CHECK (expiry_date > SYSDATE),
CONSTRAINT it_pk PRIMARY KEY (ord_no,item_no),
CONSTRAINT ord_fk FOREIGN KEY(ord_no) REFERENCES orders(ord_no));
```

The above command fails when executed. What could be the reason?

- A. The BETWEEN clause cannot be used for the CHECK constraint
- B. SYSDATE cannot be used with the CHECK constraint
- C. ORD\_NO and ITEM\_NO cannot be used as a composite primary key because ORD\_NO is also the FOREIGN KEY
- D. The CHECK constraint cannot be placed on columns having the DATE data type

**Answer:** B

**Explanation:**

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

**QUESTION: 18**

Evaluate the following SQL statements: DELETE FROM sales; There are no other uncommitted transactions on the SALES table. Which statement is true about the DELETE statement?

- A. It removes all the rows as well as the structure of the table
- B. It removes all the rows in the table and deleted rows cannot be rolled back
- C. It removes all the rows in the table and deleted rows can be rolled back
- D. It would not remove the rows if the table has a primary key

**Answer:** C

**QUESTION: 19**

Examine the structure of the EMPLOYEES table:

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

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

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

Which INSERT statement meets the above requirements?

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

**Answer:** D

**QUESTION: 20**

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

- A. A constraint can be disabled even if the constraint column contains data
- B. A constraint is enforced only for the INSERT operation on a table
- C. A foreign key cannot contain NULL values
- D. All constraints can be defined at the column level as well as the table level
- E. A columns with the UNIQUE constraint can contain NULL values

**Answer:** A, E

**QUESTION: 21**

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

- A. CURRVAL is used to refer to the last sequence number that has been generated
- B. DELETE <sequencename> would remove a sequence from the database
- C. The numbers generated by a sequence can be used only for one table

- D. When the MAXVALUE limit for a sequence is reached, you can increase the MAXVALUE limit by using the ALTER SEQUENCE statement
- E. When a database instance shuts down abnormally, the sequence numbers that have been cached but not used would be available once again when the database instance is restarted

**Answer:** A, D

**Explanation:**

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

**QUESTION: 22**

The ORDERS TABLE belongs to the user OE. OE has granted the SELECT privilege on the ORDERS table to the user HR. Which statement would create a synonym ORD so that HR can execute the following query successfully? `SELECT * FROM ord;`

- A. CREATE SYNONYM ord FOR orders; This command is issued by OE.
- B. CREATE PUBLIC SYNONYM ord FOR orders; This command is issued by OE.
- C. CREATE SYNONYM ord FOR oe.orders; This command is issued by the database administrator.
- D. CREATE PUBLIC SYNONYM ord FOR oe.orders; This command is issued by the database administrator.

**Answer:** D

**Explanation:**

Creating a Synonym for an Object To refer to a table that is owned by another user, you need to prefix the table name with the name of the user who created it, followed by a period. Creating a synonym eliminates the need to qualify the object name with the schema and provides you with an alternative name for a table, view, sequence,

procedure, or other objects. This method can be especially useful with lengthy object names, such as views. In the syntax: PUBLIC Creates a synonym that is accessible to all users synonym Is the name of the synonym to be created object Identifies the object for which the synonym is created Guidelines The object cannot be contained in a package. A private synonym name must be distinct from all other objects that are owned by the same user. If you try to execute the following command (alternative B, issued by OE): CREATE PUBLIC SYNONYM ord FOR orders; You will get an error. Error que empieza en la línea 693 del comando: create public synonym nuly for prueba\_null Error en la línea de comandos:693 Columna:0 Informe de error: Error SQL: ORA-01031: privilegios insuficientes 01031. 00000 - "insufficient privileges" The message gives you the answer: OE doesn't have enough privileges. However, if you give the necessary privileges (issued by DBA): GRANT CREATE PUBLIC SYNONYM TO OE; You won't have problems executing the command in the alternative B (issued by OE): CREATE PUBLIC SYNONYM ord FOR orders; Finally, if you need to be sure what system privileges you have in your active session, you can execute the following command (issued by OE): SELECT \* FROM USER\_PRIVS; (One of the rows must be: CREATE PUBLIC SYNONYM ).

### **QUESTION: 23**

Evaluate this SQL statement: SELECT e.emp\_name, d.dept\_name FROM employees e JOIN departments d USING (department\_id) WHERE d.department\_id NOT IN (10,40) ORDER BY dept\_name; The statement fails when executed. Which change fixes the error?

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

**Answer:** C, E

### **Explanation:**

Prefix the column in the ORDER BY Clause would cause the statement to succeed, assuming that the statement failed because the dept\_name existed in employee & department tables. Not C: Removing the alias from the columns in the SELECT clause would cause the Statement to fail if the columns existing in both tables.

### **QUESTION: 24**

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

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

**Answer:** D

**QUESTION:** 25

Evaluate the following SQL query;

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

What would be the outcome?

- A. 200
- B. 16
- C. 160
- D. 150
- E. 100

**Answer:** C

**Explanation:**

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

**QUESTION:** 26

Which two statements are true regarding single row functions? (Choose two.)

- A. They can be nested only to two levels

- B. They always return a single result row for every row of a queried table
- C. Arguments can only be column values or constant
- D. They can return a data type value different from the one that is referenced
- E. They accept only a single argument

**Answer:** B, D

**Explanation:**

A function is a program written to optionally accept input parameters, perform an operation, or return a single value. A function returns only one value per execution. Three important components form the basis of defining a function. The first is the input parameter list. It specifies zero or more arguments that may be passed to a function as input for processing. These arguments or parameters may be of differing data types, and some are mandatory while others may be optional. The second component is the data type of its resultant value. Upon execution, only one value is returned by the function. The third encapsulates the details of the processing performed by the function and contains the program code that optionally manipulates the input parameters, performs calculations and operations, and generates a return value.

**QUESTION: 27**

Which statement is true regarding the UNION operator?

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

**Answer:** D

**Explanation:**

The columns in the queries that make up a compound query can have different names, but the output result set will use the names of the columns in the first query. A compound query will by default return rows sorted across all the columns, from left to right. The only exception is UNION ALL, where the rows will not be sorted. The only place where an ORDER BY clause is permitted is at the end of the compound query.

Oracle Server and Set Operators

- Duplicate rows are automatically eliminated except in UNION ALL.
- Column names from the first query appear in the result.
- The output is sorted in ascending order by default except in UNION ALL.

**QUESTION: 28**

Which two statements are true regarding working with dates? (Choose two.)

- A. The default internal storage of dates is in the numeric format
- B. The RR date format automatically calculates the century from the SYSDATE function but allows the user to enter the century if required
- C. The default internal storage of dates is in the character format
- D. The RR date format automatically calculates the century from the SYSDATE function and does not allow the user to enter the century

**Answer:** A, B

**Explanation:**

Working with Dates The Oracle Database stores dates in an internal numeric format, representing the century, year, month, day, hours, minutes, and seconds. The default display and input format for any date is DD-MON-RR. RR Date Format The RR date format is similar to the YY element, but you can use it to specify different centuries. Use the RR date format element instead of YY so that the century of the return value varies according to the specified two digit year and the last two digits of the current year. The table in the slide summarizes the behavior of the RR element.

Current Year	Given Date	Interpreted (RR)	Interpreted (YY)
1994	27-OCT-95	1995	1995
1994	27-OCT-17	2017	1917
2001	27-OCT-17	2017	2017
2048	27-OCT-52	1952	2052
2051	27-OCT-47	2147	2047

Note the values shown in the last two rows of the above table. As we approach the middle of the century, then the RR behavior is probably not what you want. This data is stored internally as follows: CENTURY YEAR MONTH DAY HOUR MINUTE SECOND 19 87 06 17 17 10 43

**QUESTION:** 29

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

Table CUSTOMERS		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2 (20)
CUST_LAST_NAME	NOT NULL	VARCHAR2 (40)
CUST_GENDER	NOT NULL	CHAR (1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER (4)
CUST_MARITAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

NEW\_CUSTOMERS is a new table with the columns CUST\_ID, CUST\_NAME and CUST\_CITY that have the same data types and size as the corresponding columns in the CUSTOMERS table. Evaluate the following INSERT statement:

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

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

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

**Answer:** A

**Explanation:**

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

**QUESTION: 30**

View the Exhibit and examine the description for the CUSTOMERS table.

Table CUSTOMERS		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2 (20)
CUST_LAST_NAME	NOT NULL	VARCHAR2 (40)
CUST_GENDER	NOT NULL	CHAR (1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER (4)
CUST_MARITAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

You want to update the CUST\_CREDIT\_LIMIT column to NULL for all the customers, where CUST\_INCOME\_LEVEL has NULL in the CUSTOMERS table. Which SQL statement will accomplish the task?

A.

UPDATE customers

SET cust\_credit\_limit = NULL

WHERE CUST\_INCOME\_LEVEL = NULL;

B.

UPDATE customers

SET cust\_credit\_limit = NULL WHERE cust\_income\_level IS NULL;

C.

UPDATE customers

SET cust\_credit\_limit = TO\_NUMBER(NULL) WHERE cust\_income\_level = TO\_NUMBER(NULL);

D.

UPDATE customers

SET cust\_credit\_limit = TO\_NUMBER(' ',9999) WHERE cust\_income\_level IS NULL;

**Answer:** B

### QUESTION: 31

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

A. A sub query should retrieve only one row.

B. A sub query can retrieve zero or more rows.

C. A sub query can be used only in SQL query statements.

- D. Sub queries CANNOT be nested by more than two levels.
- E. A sub query CANNOT be used in an SQL query statement that uses group functions.
- F. When a sub query is used with an inequality comparison operator in the outer SQL statement, the column list in the SELECT clause of the sub query should contain only one column.

**Answer:** B, F

**Explanation:**

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

**QUESTION:** 32

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

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

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

A.

SELECT promo\_name, promo\_begin\_date FROM promotions  
WHERE promo\_begin\_date > ALL (SELECT MAX(promo\_begin\_date) FROM promotions )AND  
promo\_category = 'INTERNET';

B.

SELECT promo\_name, promo\_begin\_date FROM promotions WHERE  
 promo\_begin\_date IN (SELECT promo\_begin\_date FROM promotions  
 WHERE promo\_category='INTERNET');  
 C.  
 SELECT promo\_name, promo\_begin\_date FROM promotions WHERE  
 promo\_begin\_date > ALL (SELECT promo\_begin\_date FROM promotions  
 WHERE promo\_category = 'INTERNET');  
 D.  
 SELECT promo\_name, promo\_begin\_date FROM promotions WHERE  
 promo\_begin\_date > ANY (SELECT promo\_begin\_date FROM promotions  
 WHERE promo\_category = 'INTERNET');

**Answer:** C

**QUESTION:** 33

Which are /SQL\*Plus commands? (Choose all that apply.)

- A. INSERT
- B. UPDATE
- C. SELECT
- D. DESCRIBE
- E. DELETE
- F. RENAME

**Answer:** D

**Explanation:**

Describe is a valid iSQL\*Plus/ SQL\*Plus command. INSERT, UPDATE & DELETE are SQL DML Statements. A SELECT is an ANSI Standard SQL Statement not an iSQL\*Plus Statement. RENAME is a DDL Statement.

**QUESTION:** 34

Which two statements are true regarding the COUNT function? (Choose two.)

- A. COUNT(\*) returns the number of rows including duplicate rows and rows containing NULL value in any of the columns
- B. COUNT(cust\_id) returns the number of rows including rows with duplicate customer IDs and NULL value in the CUST\_ID column
- C. COUNT(DISTINCT inv\_amt) returns the number of rows excluding rows containing duplicates and NULL values in the INV\_AMT column

- D. A SELECT statement using COUNT function with a DISTINCT keyword cannot have a WHERE clause  
 E. The COUNT function can be used only for CHAR, VARCHAR2 and NUMBER data types

**Answer:** A, C

**Explanation:**

Using the COUNT Function The COUNT function has three formats: COUNT(\*) COUNT(expr) COUNT(DISTINCT expr) COUNT(\*) returns the number of rows in a table that satisfy the criteria of the SELECT statement, including duplicate rows and rows containing null values in any of the columns. If a WHERE clause is included in the SELECT statement, COUNT(\*) returns the number of rows that satisfy the condition in the WHERE clause. In contrast, COUNT(expr) returns the number of non-null values that are in the column identified by expr. COUNT(DISTINCT expr) returns the number of unique, non-null values that are in the column identified by expr.

**QUESTION: 35**

Examine the description of the EMP\_DETAILS table given below: Exhibit:

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

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

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

**Answer:** B, C

**Explanation:**

LONG Character data in the database character set, up to 2GB. All the functionality of LONG (and more) is provided by CLOB; LONGs should not be used in a modern database, and if your database has any columns of this type they should be converted to

CLOB. There can only be one LONG column in a table. Guidelines A LONG column is not copied when a table is created using a subquery. A LONG column cannot be included in a GROUP BY or an ORDER BY clause. Only one LONG column can be used per table. No constraints can be defined on a LONG column. You might want to use a CLOB column rather than a LONG column.

**QUESTION: 36**

Which CREATE TABLE statement is valid?

A.

```
CREATE TABLE ord_details
(ord_no NUMBER(2) PRIMARY KEY, item_no NUMBER(3) PRIMARY KEY,
ord_date DATE NOT NULL);
```

B.

```
CREATE TABLE ord_details
(ord_no NUMBER(2) UNIQUE, NOT NULL, item_no NUMBER(3),
ord_date DATE DEFAULT SYSDATE NOT NULL);
```

C.

```
CREATE TABLE ord_details
(ord_no NUMBER(2), item_no NUMBER(3),
ord_date DATE DEFAULT NOT NULL, CONSTRAINT ord_uq UNIQUE (ord_no),
CONSTRAINT ord_pk PRIMARY KEY (ord_no));
```

D.

```
CREATE TABLE ord_details
(ord_no NUMBER(2), item_no NUMBER(3),
ord_date DATE DEFAULT SYSDATE NOT NULL, CONSTRAINT ord_pk
PRIMARY KEY (ord_no, item_no));
```

**Answer:** D

**Explanation:**

PRIMARY KEY Constraint A PRIMARY KEY constraint creates a primary key for the table. Only one primary key can be created for each table. The PRIMARY KEY constraint is a column or a set of columns that uniquely identifies each row in a table. This constraint enforces the uniqueness of the column or column combination and ensures that no column that is part of the primary key can contain a null value. Note: Because uniqueness is part of the primary key constraint definition, the Oracle server enforces the uniqueness by implicitly creating a unique index on the primary key column or columns.

**QUESTION: 37**

See the exhibit and examine the structure of the CUSTOMERS and GRADES tables:

**CUSTOMERS**

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

**GRADES**

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

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

A.

```
SELECT custname, grade
FROM customers, grades
WHERE (SELECT MAX(cust_credit_limit)
FROM customers) BETWEEN startval and endval;
```

B.

```
SELECT custname, grade
FROM customers, grades
WHERE (SELECT MAX(cust_credit_limit)
FROM customers) BETWEEN startval and endval
AND cust_credit_limit BETWEEN startval AND endval;
```

C.

```
SELECT custname, grade
FROM customers, grades
WHERE cust_credit_limit = (SELECT MAX(cust_credit_limit) FROM customers)
AND cust_credit_limit BETWEEN startval AND endval;
```

D.

```
SELECT custname, grade
FROM customers , grades
WHERE cust_credit_limit IN (SELECT MAX(cust_credit_limit) FROM customers)
AND MAX(cust_credit_limit) BETWEEN startval AND endval;
```

**Answer:** B, C

**QUESTION:** 38

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_MARITAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

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

- A. SELECT NVL(cust\_credit\_limit,'Not Available')\*.15 "NEW CREDIT" FROM customers;
- B. SELECT NVL(cust\_credit\_limit\*.15,'Not Available') "NEW CREDIT" FROM customers;
- C. SELECT TO\_CHAR(NVL(cust\_credit\_limit\*.15,'Not Available')) "NEW CREDIT" FROM customers;
- D. SELECT NVL(TO\_CHAR(cust\_credit\_limit\*.15),'Not Available') "NEW CREDIT" FROM customers;

**Answer:** D

**Explanation:**

NVL Function Converts a null value to an actual value: Data types that can be used are date, character, and number. Data types must match:

- NVL(commission\_pct,0)
- NVL(hire\_date,'01-JAN-97')
- NVL(job\_id,'No Job Yet')

**QUESTION: 39**

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

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

**Answer:** A, C

**QUESTION: 40**

Which two are true about aggregate functions? (Choose two.)

- A. You can use aggregate functions in any clause of a SELECT statement.
- B. You can use aggregate functions only in the column list of the select clause and in the WHERE clause of a SELECT statement.
- C. You can mix single row columns with aggregate functions in the column list of a SELECT statement by grouping on the single row columns.
- D. You can pass column names, expressions, constants, or functions as parameter to an aggregate function.
- E. You can use aggregate functions on a table, only by grouping the whole table as one single group.
- F. You cannot group the rows of a table by more than one column while using aggregate functions.

**Answer:** A, D

**QUESTION: 41**

See the structure of the PROGRAMS table:

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

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

- A. SELECT NVL(ADD\_MONTHS(END\_DATE,1),SYSDATE)  
FROM programs;
- B. SELECT TO\_DATE(NVL(SYSDATE-END\_DATE,SYSDATE)) FROM programs;
- C. SELECT NVL(MONTHS\_BETWEEN(start\_date,end\_date),'Ongoing') FROM programs;
- D. NVL(TO\_CHAR(MONTHS\_BETWEEN(start\_date,end\_date)),'Ongoing') FROM programs;

**Answer:** A, D

**Explanation:**

NVL Function Converts a null value to an actual value: Data types that can be used are date, character, and number. Data types must match:

- NVL(commission\_pct,0)
- NVL(hire\_date,'01-JAN-97')
- NVL(job\_id,'No Job Yet')

MONTHS\_BETWEEN(date1, date2): Finds the number of months between date1 and date2 . The result can be positive or negative. If date1 is later than date2, the result is positive; if date1 is earlier than date2, the result is negative. The noninteger part of the result represents a portion of the month. MONTHS\_BETWEEN returns a numeric value. - answer C NVL has different datatypes - numeric and strings, which is not possible! The data types of the original and if null parameters must always be compatible. They must either be of the same type, or it must be possible to implicitly convert if null to the type of the original parameter. The NVL function returns a value with the same data type as the original parameter.

**QUESTION:** 42

You issue the following command to drop the PRODUCTS table: SQL>DROP TABLE products; What is the implication of this command? (Choose all that apply.)

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

**Answer:** B, C, D

**QUESTION: 43**

Exhibit contains the structure of PRODUCTS table:

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

Evaluate the following query:

```
SQL> SELECT prod_name
  FROM products
 WHERE prod_id IN (SELECT prod_id FROM products
                    WHERE prod_list_price =
                      (SELECT MAX(prod_list_price)FROM products
                       WHERE prod_list_price <
                         (SELECT MAX(prod_list_price)FROM products)));
```

What would be the outcome of executing the above SQL statement?

- A. It produces an error
- B. It shows the names of products whose list price is the second highest in the table.
- C. It shows the names of all products whose list price is less than the maximum list price
- D. It shows the names of all products in the table

**Answer:** B

**QUESTION: 44**

See the Exhibits and examine the structures of PRODUCTS, SALES and CUSTOMERS table:

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

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

Table CUSTOMERS		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2 (20)
CUST_LAST_NAME	NOT NULL	VARCHAR2 (40)
CUST_GENDER	NOT NULL	CHAR (1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER (4)
CUST_MARITAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

You issue the following query:

```
SQL>SELECT p.prod_id,prod_name,prod_list_price,
           quantity_sold,cust_last_name
      FROM products p NATURAL JOIN sales s  NATURAL JOIN customers c
     WHERE prod_id =148;
```

Which statement is true regarding the outcome of this query?

- A. It produces an error because the NATURAL join can be used only with two tables
- B. It produces an error because a column used in the NATURAL join cannot have a qualifier
- C. It produces an error because all columns used in the NATURAL join should have a qualifier
- D. It executes successfully

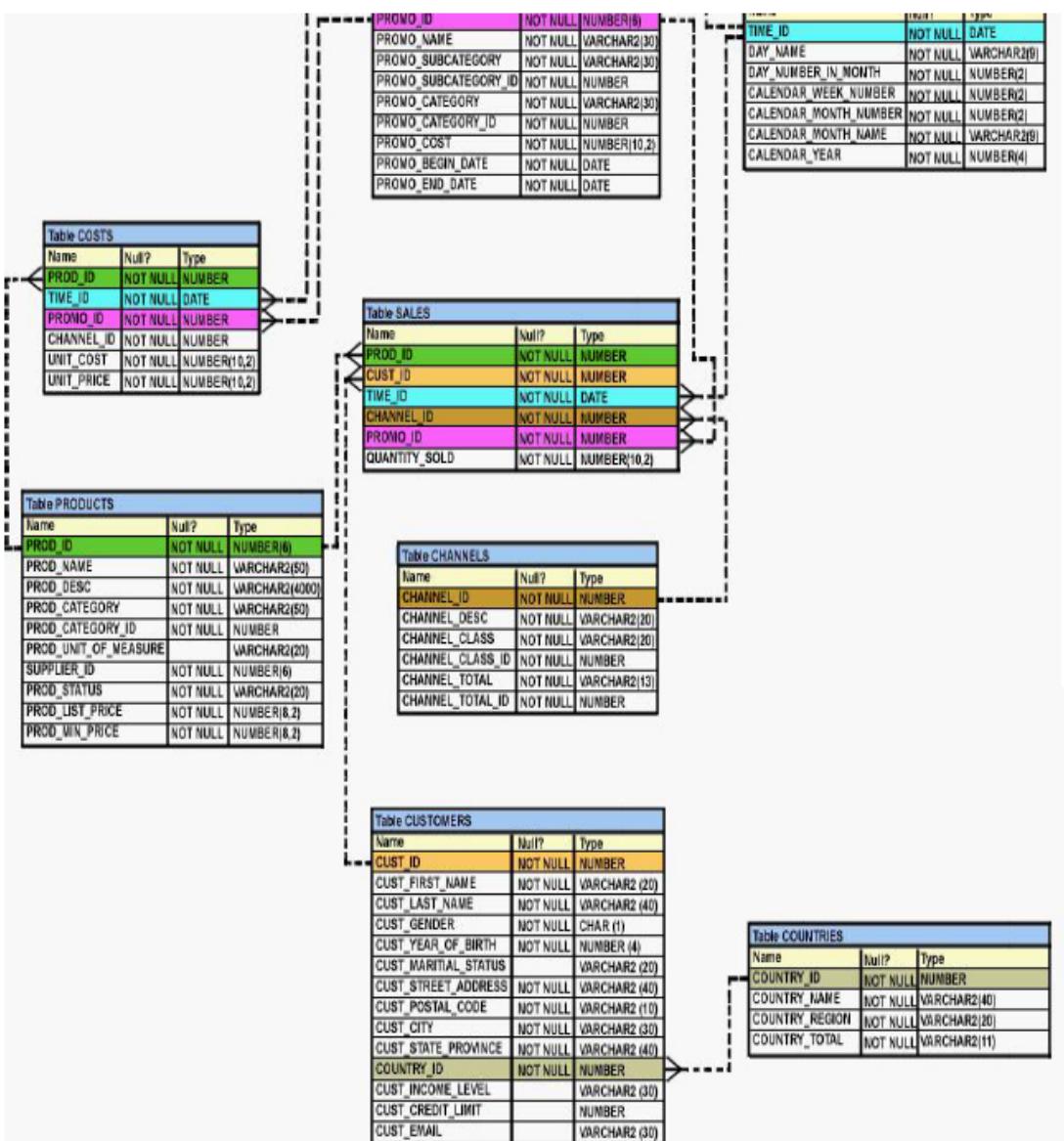
**Answer:** B

**Explanation:**

Creating Joins with the USING Clause Natural joins use all columns with matching names and data types to join the tables. The USING clause can be used to specify only those columns that should be used for an equijoin. The Natural JOIN USING Clause The format of the syntax for the natural JOIN USING clause is as follows: SELECT table1.column, table2.column FROM table1 JOIN table2 USING (join\_column1, join\_column2...); While the pure natural join contains the NATURAL keyword in its syntax, the JOIN...USING syntax does not. An error is raised if the keywords NATURAL and USING occur in the same join clause. The JOIN...USING clause allows one or more equijoin columns to be explicitly specified in brackets after the USING keyword. This avoids the shortcomings associated with the pure natural join. Many situations demand that tables be joined only on certain columns, and this format caters to this requirement.

**QUESTION:** 45

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



You want to create a SALE\_PROD view by executing the following SQL statements:

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

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

- A. The view will be created and you can perform DLM operations on the view
- B. The view will not be created because the join statements are not allowed for creating a view

- C. The view will not be created because the GROUP BY clause is not allowed for creating a view
- D. The view will be created but no DML operations will be allowed on the view

**Answer:** D

**Explanation:**

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

**QUESTION:** 46

Which three statements are true regarding the data types in Oracle Database 10g/11g?  
(Choose three.)

- A. The BLOB data type column is used to store binary data in an operating system file
- B. The minimum column width that can be specified for a VARCHAR2 data type column is one
- C. A TIMESTAMP data type column stores only time values with fractional seconds
- D. The value for a CHAR data type column is blank-padded to the maximum defined column width
- E. Only One LONG column can be used per table

**Answer:** B, D, E

**Explanation:**

■ LONG Character data in the database character set, up to 2GB. All the functionality of LONG (and more) is provided by CLOB; LONGs should not be used in a modern database, and if your database has any columns of this type they should be converted to CLOB. There can only be one LONG column in a table. DVARCHAR2 Variable-length character data, from 1 byte to 4KB. The data is stored in the database character set. The VARCHAR2 data type must be qualified with a number indicating the maximum length of the column. If a value is inserted into the column that is less than this, it is not a problem: the value will only take up as much space as it needs. If the value is longer than this maximum, the INSERT will fail with an error. VARCHAR2(size) Variable-length character data (A maximum size must be specified: minimum size is 1; maximum size is 4,000.) BLOB Like CLOB, but binary data that will not undergo character set conversion by Oracle Net. BFILE A locator pointing to a file stored on the operating system of the database server. The size of the files is limited to 4GB. TIMESTAMP This is length zero if the column is empty, or up to 11

bytes, depending on the precision specified. Similar to DATE, but with precision of up to 9 decimal places for the seconds, 6 places by default.

**QUESTION:** 47

Which three are true? (Choose three.)

- A. A MERGE statement is used to merge the data of one table with data from another.
- B. A MERGE statement replaces the data of one table with that of another.
- C. A MERGE statement can be used to insert new rows into a table.
- D. A MERGE statement can be used to update existing rows in a table.

**Answer:** A, C, D

**Explanation:**

The MERGE Statement allows you to conditionally insert or update data in a table. If the rows are present in the target table which match the join condition, they are updated if the rows are not present they are inserted into the target table

**QUESTION:** 48

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

- A. A sub query that defines a view cannot include the GROUP BY clause
- B. A view is created with the sub query having the DISTINCT keyword can be updated
- C. A Data Manipulation Language (DML) operation can be performed on a view that is created with the sub query having all the NOT NULL columns of a table
- D. A view that is created with the sub query having the pseudo column ROWNUM keyword cannot be updated

**Answer:** C, D

**Explanation:**

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

**QUESTION:** 49

Examine the structure of the EMPLOYEES and NEW\_EMPLOYEES tables:

**EMPLOYEES**

EMPLOYEE\_ID NUMBER Primary Key  
 FIRST\_NAME VARCHAR2(25)  
 LAST\_NAME VARCHAR2(25)  
 HIRE\_DATE DATE

**NEW EMPLOYEES**

EMPLOYEE\_ID NUMBER Primary Key  
 NAME VARCHAR2(60)

Which DELETE statement is valid?

- A. DELETE FROM employees  
 WHERE employee\_id = (SELECT employee\_id FROM employees);
- B. DELETE \* FROM employees  
 WHERE employee\_id=(SELECT employee\_id FROM new\_employees);
- C. DELETE FROM employees  
 WHERE employee\_id IN (SELECT employee\_id  
 FROM new\_employees  
 WHERE name = 'Carrey');
- D. DELETE \* FROM employees  
 WHERE employee\_id IN (SELECT employee\_id  
 FROM new\_employees  
 WHERE name = 'Carrey');

**Answer:** C

**QUESTION: 50**

View the Exhibits and examine the structures of the PROMOTIONS and SALES tables.

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

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

Evaluate the following SQL statements:

```
SQL>SELECT p.promo_id, p.promo_name, s.prod_id
  FROM sales s RIGHT OUTER JOIN promotions p
  ON (s.promo_id = p.promo_id);
```

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

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

**Answer:** D

**QUESTION:** 51

Examine the structure of the EMPLOYEES table:

<b>EMPLOYEE_ID</b>	<b>NUMBER</b>	Primary Key
<b>FIRST_NAME</b>	VARCHAR2(25)	
<b>LAST_NAME</b>	VARCHAR2(25)	
<b>HIRE_DATE</b>	DATE	

Which UPDATE statement is valid?

A.

```
UPDATE employees
SET first_name = 'John' SET last_name = 'Smith'
WHERE employee_id = 180;
```

B.

```
UPDATE employees
SET first_name = 'John', SET last_name = 'Smith'
WHERE employee_id = 180;
```

C.

```
UPDATE employee
SET first_name = 'John' AND last_name = 'Smith'
WHERE employee_id = 180;
```

D.

```
UPDATE employee
SET first_name = 'John', last_name = 'Smith'
WHERE employee_id = 180;
```

**Answer:** D

**QUESTION:** 52

View the Exhibit and evaluate structures of the SALES, PRODUCTS, and COSTS tables.

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

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

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

Evaluate the following SQL statements:

```
SQL>SELECT prod_id FROM products
INTERSECT
SELECT prod_id FROM sales
MINUS
SELECT prod_id FROM costs;
```

Which statement is true regarding the above compound query?

- A. It shows products that have a cost recorded irrespective of sales
- B. It shows products that were sold and have a cost recorded
- C. It shows products that were sold but have no cost recorded

- D. It reduces an error

**Answer:** C

**QUESTION: 53**

The PRODUCTS table has the following structure:

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

Evaluate the following two SQL statements:

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

Which statement is true regarding the outcome?

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

**Answer:** B

**Explanation:**

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

**QUESTION: 54**

Which statements are correct regarding indexes? (Choose all that apply.)

- A. For each data manipulation language (DML) operation performed, the corresponding indexes are automatically updated.
- B. A nondeferrable PRIMARY KEY or UNIQUE KEY constraint in a table automatically creates a unique index.
- C. A FOREIGN KEY constraint on a column in a table automatically creates a non unique key
- D. When a table is dropped, the corresponding indexes are automatically dropped

**Answer:** A, B, D

**QUESTION: 55**

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

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

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

A.

UPDATE promotions

SET promo\_cost = promo\_cost+ 100

WHERE TO\_CHAR(promo\_end\_date, 'yyyy') > '2000';

B.

SELECT promo\_begin\_date

FROM promotions

WHERE TO\_CHAR(promo\_begin\_date,'mon dd yy')='jul 01 98';

C.

UPDATE promotions

SET promo\_cost = promo\_cost+ 100

WHERE promo\_end\_date > TO\_DATE(SUBSTR('01-JAN-2000',8));

D.

SELECT TO\_CHAR(promo\_begin\_date,'dd/month') FROM promotions

WHERE promo\_begin\_date IN (TO\_DATE('JUN 01 98'), TO\_DATE('JUL 01 98'));

**Answer:** A, B

**QUESTION: 56**

Examine these statements: CREATE ROLE registrar; GRANT UPDATE ON student\_grades TO registrar; GRANT registrar to user1, user2, user3; What does this set of SQL statements do?

- A. The set of statements contains an error and does not work.
- B. It creates a role called REGISTRAR, adds the MODIFY privilege on the STUDENT\_GRADES object to the role, and gives the REGISTRAR role to three users.
- C. It creates a role called REGISTRAR, adds the UPDATE privilege on the STUDENT\_GRADES object to the role, and gives the REGISTRAR role to three users.
- D. It creates a role called REGISTRAR, adds the UPDATE privilege on the STUDENT\_GRADES object to the role, and creates three users with the role.
- E. It creates a role called REGISTRAR, adds the UPDATE privilege on three users, and gives the REGISTRAR role to the STUDENT\_GRADES object.
- F. It creates a role called STUDENT\_GRADES, adds the UPDATE privilege on three users, and gives the UPDATE role to the registrar.

**Answer:** C

**Explanation:**

the statement will create a role call REGISTRAR, grant UPDATE on student\_grades to registrar, grant the role to user1,user2 and user3. Incorrect answer: the statement does not contain error there is no MODIFY privilege statement does not create 3 users with the role privilege is grant to role then grant to user privilege is grant to role then grant to user

**QUESTION: 57**

Examine the structure of the MARKS table: Exhibit:

Name	Null?	Type
STUDENT_ID	NOT NULL	VARCHAR2(4)
STUDENT_NAME		VARCHAR2(25)
SUBJECT1		NUMBER(3)
SUBJECT2		NUMBER(3)
SUBJECT3		NUMBER(3)

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

- A. SELECT student\_name,subject1 FROM marks WHERE subject1 > AVG(subject1);
- B. SELECT student\_name,SUM(subject1) FROM marks WHERE student\_name LIKE 'R%';
- C. SELECT SUM(subject1+subject2+subject3) FROM marks WHERE student\_name IS NULL;
- D. SELECT SUM(DISTINCT NVL(subject1,0)), MAX(subject1) FROM marks WHERE subject1 > subject2;

**Answer:** C, D

**QUESTION: 58**

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

Exhibit:

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

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

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

**Answer:** D

**QUESTION:** 59

Evaluate the following two queries: Exhibit:

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

Exhibit:

```
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?

- A. Performance would improve in query 2 only if there are null values in the CUST\_CREDIT\_LIMIT column
- B. Performance would degrade in query 2
- C. There would be no change in performance
- D. Performance would improve in query 2

**Answer:** C

**Explanation:**

Note: The IN operator is internally evaluated by the Oracle server as a set of OR conditions, such as a=value1 or a=value2 or a=value3. Therefore, using the IN operator has no performance benefits and is used only for logical simplicity.

**QUESTION:** 60

View the Exhibit and examine the structure of the CUSTOMERS and CUST\_HISTORY tables.

**CUSTOMERS**

Name	Null?	Type
CUST_ID	NOT NULL	NUMBER (4)
CUST_NAME		VARCHAR2 (20)
CUST_ADDRESS		VARCHAR2 (30)
CUST_CITY		VARCHAR2 (20)

**CUST\_HISTORY**

Name	Null?	Type
CUST_ID	NOT NULL	NUMBER (4)
CUST_NAME		VARCHAR2 (20)
CUST_CITY		VARCHAR2 (20)
CHANGE_DATE		DATE

The CUSTOMERS table contains the current location of all currently active customers. The CUST\_HISTORY table stores historical details relating to any changes in the location of all current as well as previous customers who are no longer active with the company. You need to find those customers who have never changed their address. Which SET operator would you use to get the required output?

- A. INTERSECT
- B. UNION ALL
- C. MINUS
- D. UNION

**Answer:** C

**QUESTION: 61**

You created an ORDERS table with the following description: Exhibit:

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

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

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

**Answer:** B

**QUESTION:** 62

Which object privileges can be granted on a view?

- A. none
- B. DELETE, INSERT, SELECT
- C. ALTER, DELETE, INSERT, SELECT
- D. DELETE, INSERT, SELECT, UPDATE

**Answer:** D

**Explanation:**

Object privilege on VIEW is DELETE, INSERT, REFERENCES, SELECT and UPDATE. Incorrect answer: Object privilege on VIEW is DELETE, INSERT, REFERENCES, SELECT and UPDATE Object privilege on VIEW is DELETE, INSERT, REFERENCES, SELECT and UPDATE Object privilege on VIEW is DELETE, INSERT, REFERENCES, SELECT and UPDATE Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 13-12

**QUESTION:** 63

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

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

**Answer:** C, D, E

**QUESTION:** 64

View the Exhibit and examine the structure of the CUSTOMERS table. Exhibit:

Table CUSTOMERS		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2 (20)
CUST_LAST_NAME	NOT NULL	VARCHAR2 (40)
CUST_GENDER	NOT NULL	CHAR (1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER (4)
CUST_MARITAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

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

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

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

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

**Answer:** C

**QUESTION: 65**

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

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

**Answer:** E

**QUESTION: 66**

The STUDENT\_GRADES table has these columns:

STUDENT_ID	NUMBER(12)
SEMESTER_END	DATE
GPA	NUMBER(4,3)

Which statement finds students who have a grade point average (GPA) greater than 3.0 for the calendar year 2001?

- A. SELECT student\_id, gpa  
 FROM student\_grades  
 WHERE semester\_end BETWEEN '01-JAN-2001' AND '31-DEC-2001' OR gpa > 3.;
- B. SELECT student\_id, gpa  
 FROM student\_grades  
 WHERE semester\_end BETWEEN '01-JAN-2001' AND '31-DEC-2001' AND gpa gt 3.0;
- C. SELECT student\_id, gpa  
 FROM student\_grades  
 WHERE semester\_end BETWEEN '01-JAN-2001' AND '31-DEC-2001' AND gpa > 3.0;
- D. SELECT student\_id, gpa  
 FROM student\_grades  
 WHERE semester\_end BETWEEN '01-JAN-2001' AND '31-DEC-2001' OR gpa > 3.0;
- E. SELECT student\_id, gpa  
 FROM student\_grades  
 WHERE semester\_end > '01-JAN-2001' OR semester\_end < '31-DEC-2001' AND gpa >= 3.0;

**Answer:** C

**QUESTION:** 67

View the Exhibit and examine the data in the PROMOTIONS table.

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

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

A.

```
SELECT promo_category FROM promotions MINUS  
SELECT promo_category  
FROM promotions  
WHERE promo_subcategory = 'discount';
```

B.

```
SELECT promo_category FROM promotions INTERSECT  
SELECT promo_category  
FROM promotions  
WHERE promo_subcategory = 'discount';
```

C.

```
SELECT promo_category FROM promotions MINUS  
SELECT promo_category  
FROM promotions  
WHERE promo_subcategory <> 'discount';
```

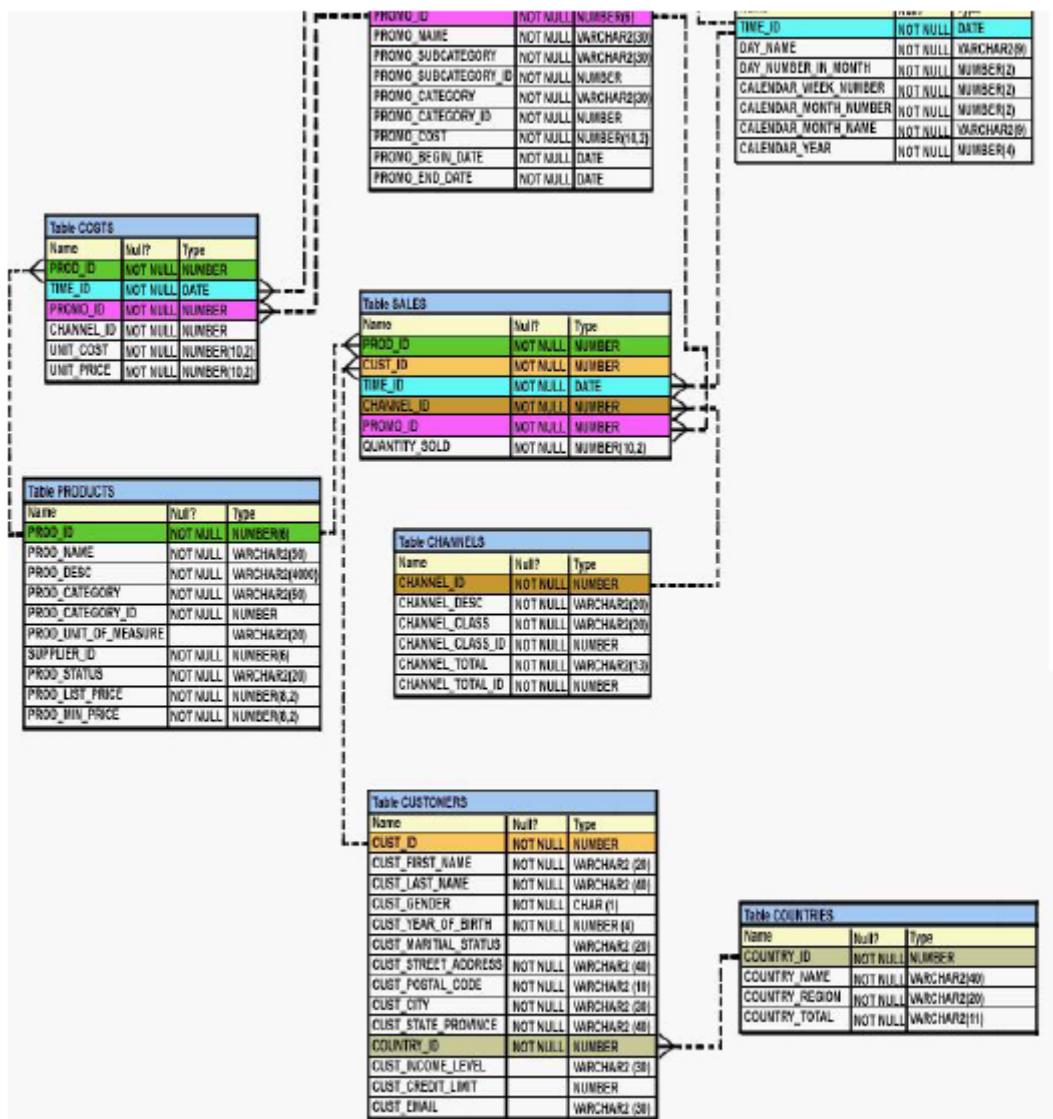
D.

```
SELECT promo_category  
FROM promotions  
INTERSECT  
SELECT promo_category  
FROM promotions  
WHERE promo_subcategory <> 'discount';
```

**Answer:** A, D

**QUESTION:** 68

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



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

Exhibit:

```

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

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

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

**Answer:** B

**QUESTION:** 69

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

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

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

```
SQL>SELECT AVG(CASE
    WHEN promo_cost BETWEEN 0 AND 2000 AND promo_category='A'
        THEN promo_cost
    ELSE null END) 'CAT_2000A',
    AVG(CASE
        WHEN promo_cost BETWEEN 2001 AND 5000 AND promo_category='A'
            THEN promo_cost
        ELSE null END) "CAT_5000A"
FROM promotions;
```

What would be the outcome?

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

**Answer:** B

**Explanation:**

CASE Expression Facilitates conditional inquiries by doing the work of an IF-THEN-ELSE statement: CASE expr WHEN comparison\_expr1 THEN return\_expr1 [WHEN comparison\_expr2 THEN return\_expr2 WHEN comparison\_exprn THEN return\_exprn ELSE else\_expr] END

**QUESTION:** 70

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

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

**Answer:** A, D

**Explanation:**

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

ON clause to specify arbitrary conditions or specify columns to join. – ANSWER C  
 The join condition is separated from other search conditions. ANSWER D The ON clause makes code easy to understand.

**QUESTION:** 71

Examine the structure of the EMPLOYEES table:

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

Which INSERT statement is valid?

A.

INSERT INTO employees (employee\_id, first\_name, last\_name, hire\_date) VALUES ( 1000, 'John', 'Smith', '01/01/01');

B.

INSERT INTO employees(employee\_id, first\_name, last\_name, hire\_date) VALUES ( 1000, 'John', 'Smith', '01 January 01');

C.

INSERT INTO employees(employee\_id, first\_name, last\_name, Hire\_date) VALUES ( 1000, 'John', 'Smith', To\_date('01/01/01'));

D.

INSERT INTO employees(employee\_id, first\_name, last\_name, hire\_date) VALUES ( 1000, 'John', 'Smith', 01-Jan-01);

**Answer:** D

**Explanation:**

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

**QUESTION:** 72

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

A.

GRANT select ON dept

TO ALL\_USERS;  
 B.  
 GRANT select ON dept  
 TO ALL;  
 C.  
 GRANT QUERY ON dept  
 TO ALL\_USERS  
 D.  
 GRANT select ON dept  
 TO PUBLIC;

**Answer:** D

**Explanation:**

view the columns associated with the constraint names in the USER\_CONS\_COLUMNS view. Incorrect answer: table to view all constraints definition and names show all object name belong to user does not display column associated no such view Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 10-25

**QUESTION:** 73

Which is an iSQL\*Plus command?

- A. INSERT
- B. UPDATE
- C. SELECT
- D. DESCRIBE
- E. DELETE
- F. RENAME

**Answer:** D

**Explanation:**

The only SQL\*Plus command in this list: DESCRIBE. It cannot be used as SQL command. This command returns a description of tablename, including all columns in that table, the datatype for each column and an indication of whether the column permits storage of NULL values. Incorrect answer: INSERT is not a SQL\*PLUS command UPDATE is not a SQL\*PLUS command SELECT is not a SQL\*PLUS command DELETE is not a SQL\*PLUS command RENAME is not a SQL\*PLUS command Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 7

**QUESTION: 74**

You work as a database administrator at ABC.com. You study the exhibit carefully. Exhibit

Using the PROMOTIONS table, you need to display the names of all promos done after January 1, 2001 starting with the latest promo. Which query would give the required result? (Choose all that apply.)

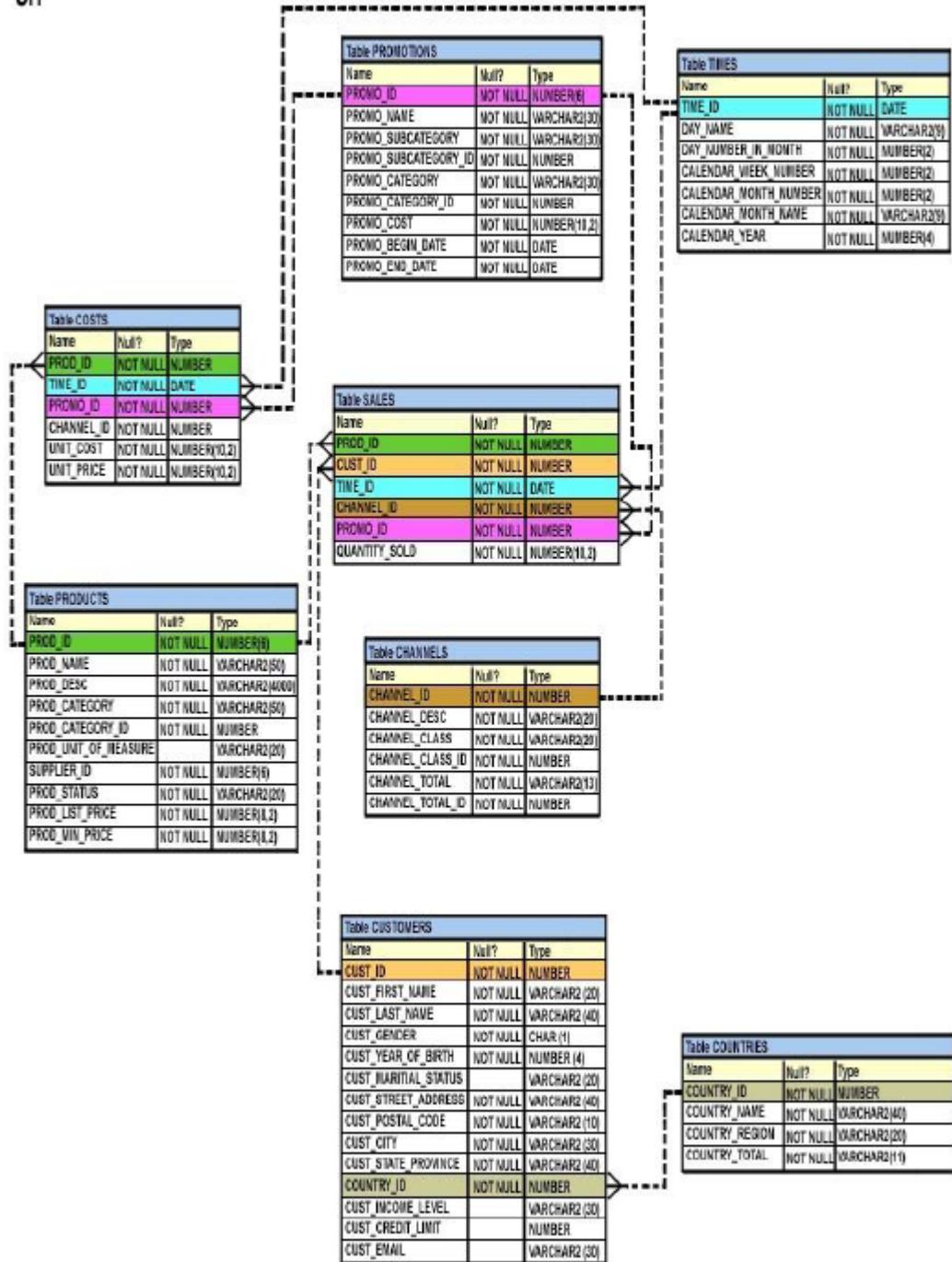
- A.    SELECT promo\_name,promo\_begin\_date  
FROM promotions  
WHERE promo\_begin\_date > '01-JAN-01' ORDER BY 1 DESC;
- B.    SELECT promo\_name,promo\_begin\_date "START DATE" FROM promotions  
WHERE promo\_begin\_date > '01-JAN-01' ORDER BY "START DATE" DESC;
- C.    SELECT promo\_name,promo\_begin\_date  
FROM promotions  
WHERE promo\_begin\_date > '01-JAN-01' ORDER BY 2 DESC;
- D.    SELECT promo\_name,promo\_begin\_date  
FROM promotions  
WHERE promo\_begin\_date > '01-JAN-01' ORDER BY promo\_name DESC;

**Answer:** B, C

**QUESTION: 75**

See the Exhibit and examine the structure of the SALES, CUSTOMERS, PRODUCTS and ITEMS tables:

SH



The PROD\_ID column is the foreign key in the SALES table, which references the PRODUCTS table. Similarly, the CUST\_ID and TIME\_ID columns are also foreign keys in the SALES table referencing the CUSTOMERS and TIMES tables, respectively. Evaluate the following the CREATE TABLE command:  
Exhibit:

```

CREATE TABLE new_sales(prod_id, cust_id, order_date DEFAULT SYSDATE)
AS
SELECT prod_id, cust_id, time_id
FROM sales;

```

Which statement is true regarding the above command?

- A. The NEW\_SALES table would not get created because the column names in the CREATE TABLE command and the SELECT clause do not match
- B. The NEW\_SALES table would get created and all the NOT NULL constraints defined on the specified columns would be passed to the new table
- C. The NEW\_SALES table would not get created because the DEFAULT value cannot be specified in the column definition
- D. The NEW\_SALES table would get created and all the FOREIGN KEY constraints defined on the specified columns would be passed to the new table

**Answer:** B

**Explanation:**

Creating a Table Using a Subquery Create a table and insert rows by combining the CREATE TABLE statement and the AS subquery option. CREATE TABLE table [(column, column...)] AS subquery; Match the number of specified columns to the number of subquery columns. Define columns with column names and default values. Guidelines The table is created with the specified column names, and the rows retrieved by the SELECT statement are inserted into the table. The column definition can contain only the column name and default value. If column specifications are given, the number of columns must equal the number of columns in the subquery SELECT list. If no column specifications are given, the column names of the table are the same as the column names in the subquery. The column data type definitions and the NOT NULL constraint are passed to the new table. Note that only the explicit NOT NULL constraint will be inherited. The PRIMARY KEY column will not pass the NOT NULL feature to the new column. Any other constraint rules are not passed to the new table. However, you can add constraints in the column definition.

**QUESTION:** 76

The CUSTOMERS table has the following structure: Exhibit:

Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2(20)
CUST_LAST_NAME	NOT NULL	VARCHAR2(30)
CUST_INCOME_LEVEL		VARCHAR2(30)
CUST_CREDIT_LIMIT		NUMBER

You need to write a query that does the following task:

- \* Display the first name and tax amount of the customers. Tax is 5% of their credit limit
  - \* Only those customers whose income level has a value should be considered
  - \* Customers whose tax amount is null should not be considered
- Which statement accomplishes all the required tasks?

A.

```
SELECT cust_first_name, cust_credit_limit * .05 AS TAX_AMOUNT FROM
customers
WHERE cust_income_level IS NOT NULL AND
tax_amount IS NOT NULL;
```

B.

```
SELECT cust_first_name, cust_credit_limit * .05 AS TAX_AMOUNT FROM
customers
WHERE cust_income_level IS NOT NULL AND
cust_credit_limit IS NOT NULL;
```

C.

```
SELECT cust_first_name, cust_credit_limit * .05 AS TAX_AMOUNT FROM
customers
WHERE cust_income_level <> NULL AND
tax_amount <> NULL;
```

D.

```
SELECT cust_first_name, cust_credit_limit * .05 AS TAX_AMOUNT FROM
customers
WHERE (cust_income_level,tax_amount) IS NOT NULL;
```

**Answer:** B

### QUESTION: 77

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

- A. SELECT TO\_CHAR('11-oct-2007', 'fmDdspth "of" Month, Year') FROM DUAL;

- B. SELECT TO\_CHAR(TO\_DATE('11-oct-2007'), 'fmDdspth of month, year') FROM DUAL;
- C. SELECT TO\_CHAR(TO\_DATE('11-oct-2007'), 'fmDdthsp "of" Month, Year') FROM DUAL;
- D. SELECT TO\_DATE(TO\_CHAR('11-oct-2007','fmDdspth "of" Month, Year')) FROM DUAL;

**Answer:** C

**Explanation:**

Using the TO\_CHAR Function with Dates TO\_CHAR converts a datetime data type to a value of VARCHAR2 data type in the format specified by the format\_model. A format model is a character literal that describes the format of datetime stored in a character string. For example, the datetime format model for the string '11-Nov-1999' is 'DD-Mon-YYYY'. You can use the TO\_CHAR function to convert a date from its default format to the one that you specify.

- The format model must be enclosed with single quotation marks and is case-sensitive.
- The format model can include any valid date format element. But be sure to separate the date value from the format model with a comma.
- The names of days and months in the output are automatically padded with blanks.
- To remove padded blanks or to suppress leading zeros, use the fill mode fm element.

Elements of the Date Format Model

----- DY Three-letter abbreviation of the day of the week DAY Full name of the day of the week DD Numeric day of the month MM Two-digit value for the mon h MON Three-letter abbreviation of the month MONTH Full name of the month YYYY Full year in numbers YEAR Year spelled out (in English)

**QUESTION:** 78

Which statement is true regarding the INTERSECT operator?

- A. It ignores NULL values
- B. The number of columns and data types must be identical for all SELECT statements in the query
- C. The names of columns in all SELECT statements must be identical
- D. Reversing the order of the intersected tables the result

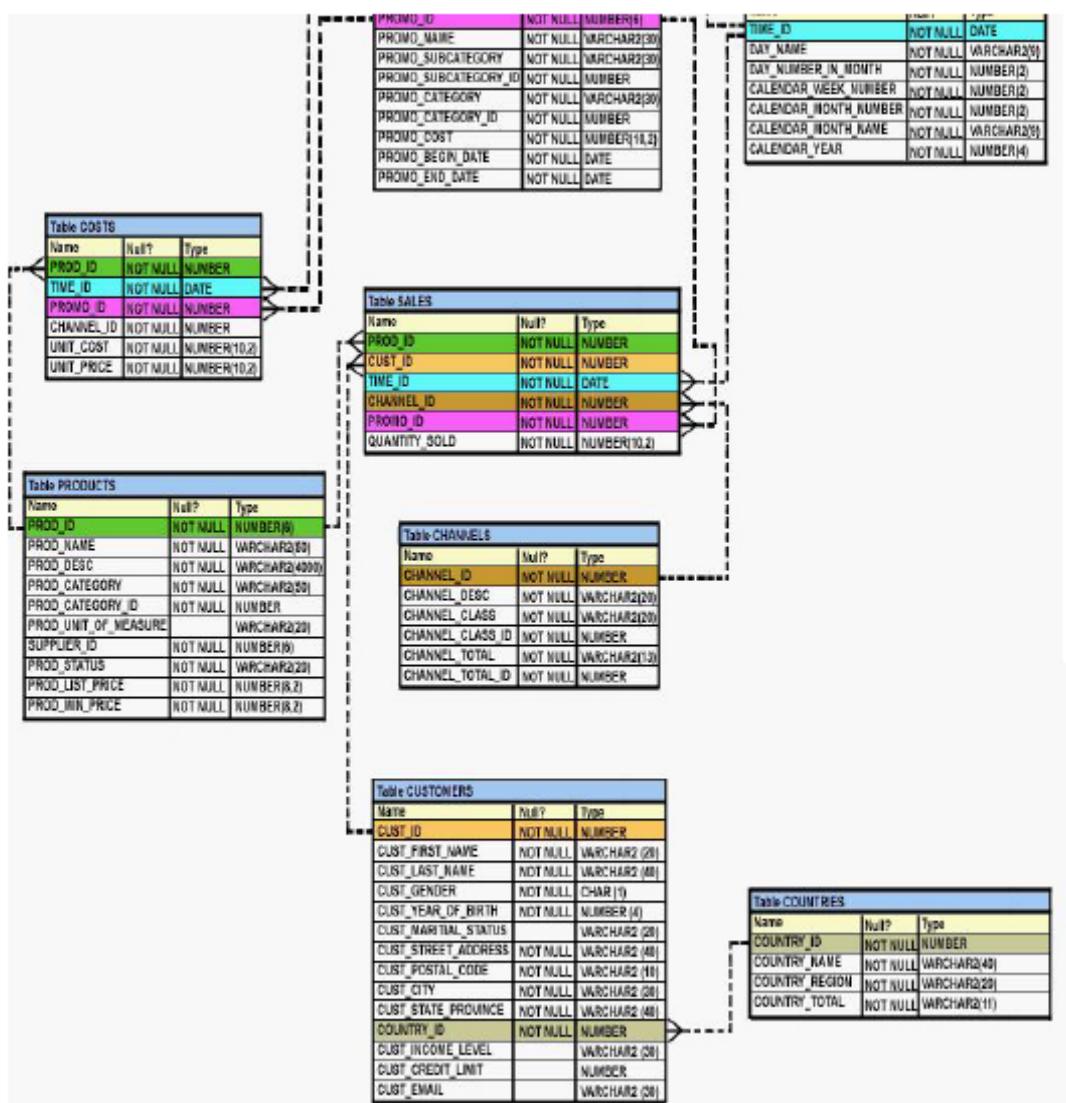
**Answer:** B

**Explanation:**

INTERSECT Returns only the rows that occur in both queries' result sets, sorting them and removing duplicates. The columns in the queries that make up a compound query can have different names, but the output result set will use the names of the columns in the first query.

### QUESTION: 79

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



Evaluate the following SQL statement: Exhibit:

```

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

```

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

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

**Answer:** A

**QUESTION:** 80

The STUDENT\_GRADES table has these columns: STUDENT\_INUMBER(12) SEMESTER\_ENDATE GPNUMBER(4,3) The registrar has asked for a report on the average grade point average (GPA), sorted from the highest grade point average to each semester, starting from the earliest date. Which statement accomplish this?

- A.  

```

SELECT student_id, semester_end, gpa
FROM student_grades
ORDER BY semester_end DESC, gpa DESC;

```
- B.  

```

SELECT student_id, semester_end, gpa
FROM student_grades
ORDER BY semester_end, gpa ASC

```
- C.  

```

SELECT student_id, semester_end, gpa
FROM student_grades
ORDER BY gpa DESC, semester_end ASC;

```
- D.  

```

SELECT student_id, semester_end, gpa
FROM student_grades

```

ORDER BY gpa DESC, semester\_end DESC;  
E.  
SELECT student\_id, semester\_end, gpa  
FROM student\_grades  
ORDER BY gpa DESC, semester\_end ASC;  
F.  
SELECT student\_id, semester\_end, gpa  
FROM student\_grades  
ORDER BY semester\_end, gpa DESC

**Answer:** F

**QUESTION:** 81

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

<b>CUSTOMERS</b>			
CUST_NO	CUST_NAME	CUST_CITY	CUST_CREDIT_LIMIT
101	KING	NEW YORK	100000
102	GREEN	BOSTON	150000
103	SCOTT	LONDON	
104	SMITH	BOSTON	

Evaluate the following query: Exhibit:

```
SQL> SELECT cust_name AS "NAME", cust_credit_limit/2 AS MIDPOINT,
MIDPOINT+100 AS "MAX
LOWER LIMIT"
FROM customers;
```

The above query produces an error on execution. What is the reason for the error?

- A. An alias cannot be used in an expression
- B. The alias MIDPOINT should be enclosed within double quotation marks for the CUST\_CREDIT\_LIMIT/2 expression
- C. The MIDPOINT +100 expression gives an error because CUST\_CREDIT\_LIMIT contains NULL values
- D. The alias NAME should not be enclosed within double quotation marks

**Answer:** A

**QUESTION: 82**

Which statement is true regarding synonyms?

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

**Answer:** D

**QUESTION: 83**

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

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

Examine the structure of PRODUCTS table. Using the PRODUCTS table, you issue the following query to generate the names, current list price and discounted list price for all those products whose list price fails below \$10 after a discount of 25% is applied on it. Exhibit:

```
SQL>SELECT prod_name, prod_list_price,
           prod_list_price - (prod_list_price * .25) "DISCOUNTED_PRICE"
      FROM products
     WHERE discounted_price < 10;
```

The query generates an error. What is the reason of generating error?

- A. The column alias should be put in uppercase and enclosed within double quotation marks in the WHERE clause
- B. The parenthesis should be added to enclose the entire expression
- C. The column alias should be replaced with the expression in the WHERE clause

D. The double quotation marks should be removed from the column alias

**Answer:** C

**Explanation:**

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

**QUESTION:** 84

Which one is a system privilege?

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

**Answer:** E

**QUESTION:** 85

What is true about sequences?

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

**Answer:** C

**QUESTION:** 86

Examine the structure of the INVOICE table: Exhibit:

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

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

- A. SELECT inv\_no,NVL2(inv\_date,'Pending','Incomplete') FROM invoice;
- B. SELECT inv\_no,NVL2(inv\_amt,inv\_date,'Not Available') FROM invoice;
- C. SELECT inv\_no,NVL2(inv\_date,sysdate-inv\_date,sysdate) FROM invoice;
- D. SELECT inv\_no,NVL2(inv\_amt,inv\_amt\*.25,'Not Available') FROM invoice;

**Answer:** A, C

**Explanation:**

The NVL Function

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

**QUESTION: 87**

View the Exhibit and examine the description for the CUSTOMERS table.

Table CUSTOMERS		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2 (20)
CUST_LAST_NAME	NOT NULL	VARCHAR2 (40)
CUST_GENDER	NOT NULL	CHAR (1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER (4)
CUST_MARITAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

You want to update the CUST\_INCOME\_LEVEL and CUST\_CREDIT\_LIMIT columns for the customer with the CUST\_ID 2360. You want the value for the CUST\_INCOME\_LEVEL to have the same value as that of the customer with the CUST\_ID 2560 and the CUST\_CREDIT\_LIMIT to have the same value as that of the customer with CUST\_ID 2566. Which UPDATE statement will accomplish the task?

A.

```
UPDATE customers
SET cust_income_level = (SELECT cust_income_level
FROM customers
WHERE cust_id = 2560),
cust_credit_limit = (SELECT cust_credit_limit
FROM customers WHERE cust_id = 2566) WHERE cust_id=2360;
```

B.

```
UPDATE customers
SET (cust_income_level,cust_credit_limit) = (SELECT
cust_income_level, cust_credit_limit
FROM customers
WHERE cust_id=2560 OR cust_id=2566) WHERE cust_id=2360;
```

C.

```
UPDATE customers
SET (cust_income_level,cust_credit_limit) = (SELECT
cust_income_level, cust_credit_limit
FROM customers
WHERE cust_id IN(2560, 2566) WHERE cust_id=2360;
```

D.

```
UPDATE customers
SET (cust_income_level,cust_credit_limit) = (SELECT
cust_income_level, cust_credit_limit
FROM customers
WHERE cust_id=2560 AND cust_id=2566) WHERE cust_id=2360;
```

**Answer:** A

**Explanation:**

Updating Two Columns with a Subquery You can update multiple columns in the SET clause of an UPDATE statement by writing multiple subqueries. The syntax is as follows: UPDATE table SET column = (SELECT column FROM table WHERE condition) [ , column = (SELECT column FROM table WHERE condition)] [WHERE condition] ;

**QUESTION:** 88

A SELECT statement can be used to perform these three functions:

1. Choose rows from a table.
  2. Choose columns from a table
  3. Bring together data that is stored in different tables by creating a link between them.
- Which set of keywords describes these capabilities?

- A. difference, projection, join

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

**Answer:** B

**Explanation:**

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

**QUESTION:** 89

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

INVOICE			
Name	Null?	Type	
INV_NO	NOT NULL	NUMBER(3)	
INV_DATE		DATE	
CUST_ID		VARCHAR2(4)	
INV_AMT		NUMBER(8,2)	
INV_NO	INV_DATE	CUST_ID	INV_AMT
-----	-----	-----	-----
1	01-APR-07	A1Q	1000
2	01-OCT-07	B1R	2000
3	01-FEB-07		3000

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

- A. SELECT MAX(inv\_date),MIN(cust\_id) FROM invoice;
- B. SELECT AVG(inv\_date-SYSDATE),AVG(inv\_amt) FROM invoice;
- C. SELECT MAX(AVG(SYSDATE-inv\_date)) FROM invoice;
- D. SELECT AVG(inv\_date) FROM invoice;

**Answer:** A, B

**QUESTION:** 90

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

Exhibit:

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

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

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

**Answer:** A

**QUESTION:** 91

Which is a valid CREATE TABLE statement?

- A. CREATE TABLE EMP9\$# AS (empid number(2));
- B. CREATE TABLE EMP\*123 AS (empid number(2));
- C. CREATE TABLE PACKAGE AS (packid number(2));
- D. CREATE TABLE 1EMP\_TEST AS (empid number(2));

**Answer:** A

**Explanation:**

Table names and column names must begin with a letter and be 1-30 characters long. Characters A- Z,a-z, 0-9, \_, \$ and # (legal characters but their use is discouraged). Incorrect answer: Non alphanumeric character such as "\*" is discourage in Oracle table name. Table name must begin with a letter. Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 9-4

**QUESTION:** 92

Evaluate the following SQL statements: Exhibit:

```
CREATE TABLE employees
(employee_id    NUMBER(2) PRIMARY KEY,
 last_name     VARCHAR2(25) NOT NULL,
 department_id NUMBER(2)NOT NULL,
 job_id        VARCHAR2(8),
 salary        NUMBER(10,2));
```

You issue the following command to create a view that displays the IDs and last names of the sales staff in the organization.

Exhibit:

```
CREATE OR REPLACE VIEW sales_staff_vu AS
SELECT employee_id,
last_name,job_id
FROM employees
WHERE job_id LIKE 'SA_%' WITH CHECK OPTION;
```

Which two statements are true regarding the above view? (Choose two.)

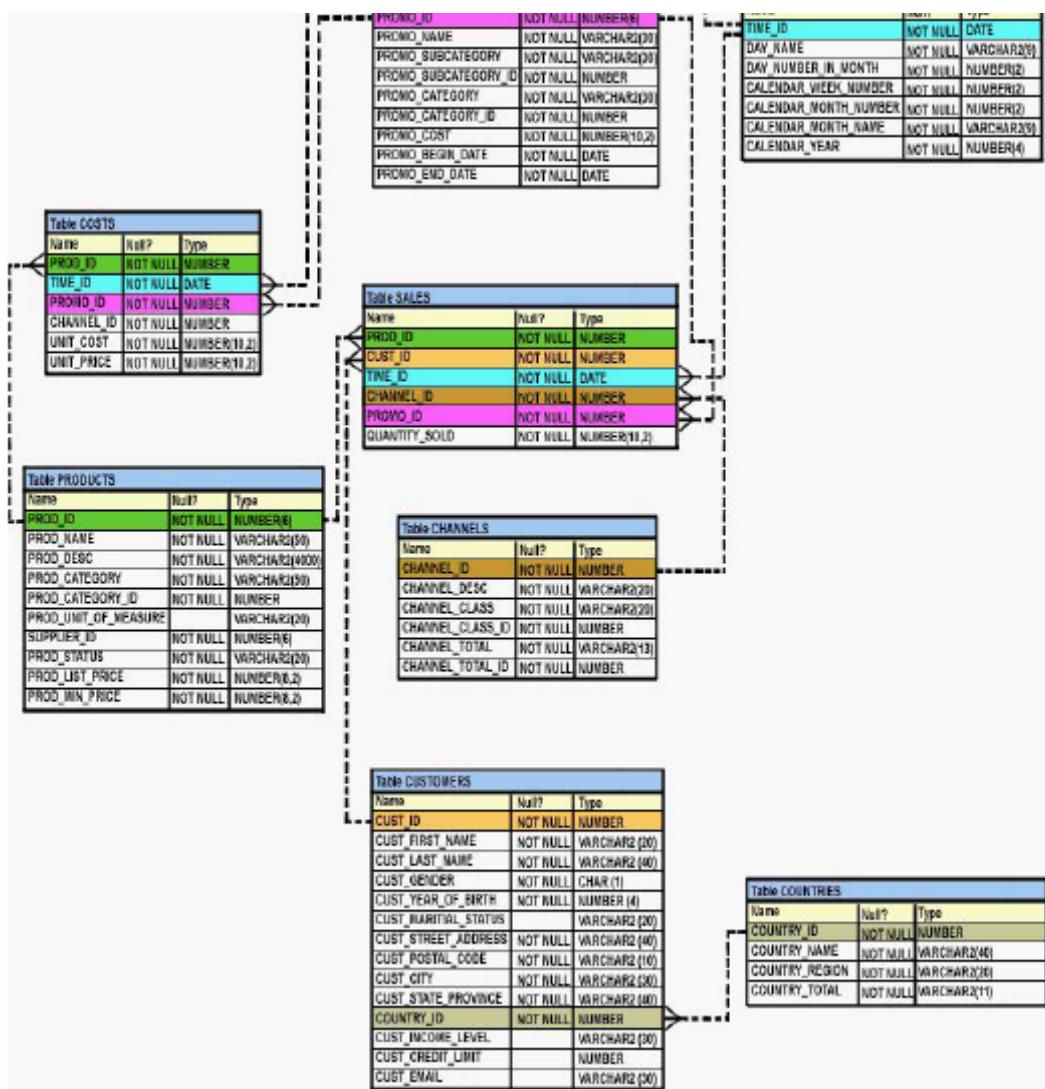
- A. It allows you to update job IDs of the existing sales staff to any other job ID in the EMPLOYEES table
- B. It allows you to delete details of the existing sales staff from the EMPLOYEES table
- C. It allows you to insert rows into the EMPLOYEES table
- D. It allows you to insert IDs, last names, and job IDs of the sales staff from the view if it is used in multitable INSERT statements

**Answer:** B, D

**QUESTION:** 93

See the Exhibit and Examine the structure of SALES and PROMOTIONS tables:  
Exhibit:

You want to delete rows from the SALES table, where the PROMO\_NAME column in the



P

ROMOTIONS table has either blowout sale or everyday low price as values. Which DELETE statements are valid? (Choose all that apply.)

A. DELETE

FROM sales

WHERE promo\_id = (SELECT promo\_id

FROM promotions

WHERE promo\_name = 'blowout sale') AND promo\_id = (SELECT promo\_id FROM promotions

WHERE promo\_name = 'everyday low price');

B.

DELETE FROM sales

WHERE promo\_id = (SELECT promo\_id

FROM promotions

WHERE promo\_name = 'blowout sale') OR promo\_id = (SELECT promo\_id FROM promotions

WHERE promo\_name = 'everyday low price');

C.

DELETE FROM sales  
WHERE promo\_id IN (SELECT promo\_id  
FROM promotions  
WHERE promo\_name = 'blowout sale' OR promo\_name = 'everyday low price');

D.

D DELETE FROM sales  
WHERE promo\_id IN (SELECT promo\_id  
FROM promotions  
WHERE promo\_name IN ('blowout sale','everyday low price'));

**Answer:** B, C, D

**QUESTION: 94**

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

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

**Answer:** A, C, D

**QUESTION: 95**

You want to create an ORD\_DETAIL table to store details for an order placed having the following business requirement:

- 1) The order ID will be unique and cannot have null values.
  - 2) The order date cannot have null values and the default should be the current date.
  - 3) The order amount should not be less than 50.
  - 4) The order status will have values either shipped or not shipped.
  - 5) The order payment mode should be cheque, credit card, or cash on delivery (COD).
- Which is the valid DDL statement for creating the ORD\_DETAIL table?

A.

```
CREATE TABLE ord_details
(ord_id NUMBER(2) CONSTRAINT ord_id_nn NOT NULL, ord_date DATE
DEFAULT SYSDATE NOT NULL,
ord_amount NUMBER(5, 2) CONSTRAINT ord_amount_min
```

```

CHECK (ord_amount > 50),
ord_status VARCHAR2(15) CONSTRAINT ord_status_chk CHECK (ord_status IN
('Shipped', 'Not Shipped')), ord_pay_mode VARCHAR2(15) CONSTRAINT
ord_pay_chk CHECK (ord_pay_mode IN ('Cheque', 'Credit Card',
'Cash On Delivery')));

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

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

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

```

**Answer:** C

**QUESTION: 96**

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

- A. Multiple columns or expressions can be compared between the main query and sub query

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

**Answer:** A, B, D

**QUESTION:** 97

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

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

- A. SELECT promo\_id, DECODE(NVL(promo\_cost,0), promo\_cost, promo\_cost \* 0.25, 100) "Discount" FROM promotions;
- B. SELECT promo\_id, DECODE(promo\_cost, 10000, DECODE(promo\_category, 'G1', promo\_cost \*.25, NULL), NULL) "Catcost" FROM promotions;
- C. SELECT promo\_id, DECODE(NULLIF(promo\_cost, 10000), NULL, promo\_cost\*.25, 'N/A') "Catcost" FROM promotions;
- D. SELECT promo\_id, DECODE(promo\_cost, >10000, 'High', <10000, 'Low') "Range" FROM promotions;

**Answer:** A, B

**Explanation:**

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

**QUESTION:** 98

Which SQL statement displays the date March 19, 2001 in a format that appears as “Nineteenth of March 2001 12:00:00 AM”?

- A. SELECT

```

TO_CHAR(TO_DATE('19-Mar-2001', 'DD-Mon-YYYY'), 'fmDdspth
"of" Month YYYY fmHH:MI:SS AM') NEW_DATE FROM dual;
B. SELECT
TO_CHAR(TO_DATE('19-Mar-2001', 'DD-Mon-YYYY'), 'Ddspth
"of" Month YYYY fmHH:MI:SS AM') NEW_DATE FROM dual;
C. SELECT
TO_CHAR(TO_DATE('19-Mar-2001', 'DD-Mon-YYYY'), 'fmDdspth "of" Month
YYYY HH:MI:SS AM') NEW_DATE FROM dual;
D. SELECT
TO_CHAR(TO_DATE('19-Mar-2001', 'DD-Mon-YYYY'), 'fmDdspth "of" Month
YYYYfmtHH:HI:SS AM') NEW_DATE FROM dual;

```

**Answer:** A

**QUESTION: 99**

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

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

**Answer:** B, C, D

**QUESTION: 100**

For which action can you use the TO\_DATE function?

- A. Convert any date literal to a date
- B. Convert any numeric literal to a date
- C. Convert any character literal to a date
- D. Convert any date to a character literal
- E. Format '10-JAN-99' to 'January 10 1999'

**Answer:** C