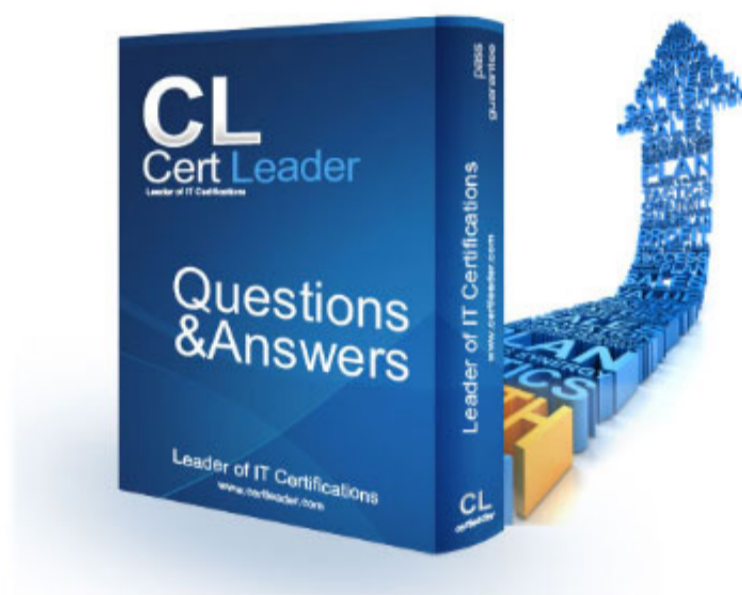


## 1Z0-051 Dumps

### Oracle Database: SQL Fundamentals I

<https://www.certleader.com/1Z0-051-dumps.html>



**NEW QUESTION 1**

The following data exists in the PRODUCTS table: PROD\_ID PROD\_LIST\_PRICE

123456 152525.99

You issue the following query:

```
SQL> SELECT RPAD(( ROUND(prod_list_price)), 10,'*')
```

```
FROM products
```

```
WHERE prod_id = 123456;
```

What would be the outcome?

- A. 152526\*\*\*\*
- B. \*\*152525.99
- C. 152525\*\*
- D. an error message

**Answer:** A

**Explanation:**

The LPAD(string, length after padding, padding string) and RPAD(string, length after padding, padding string) functions add a padding string of characters to the left or right of a string until it reaches the specified length after padding.

**NEW QUESTION 2**

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:** CD

**NEW QUESTION 3**

Evaluate the following SQL statement:

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.

**NEW QUESTION 4**

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:** AB

**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

untitled 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

#### NEW QUESTION 5

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**

#### NEW QUESTION 6

View the Exhibit and examine the structure of the PROMOTIONS, SALES, and CUSTOMER tables.

PROMOTIONS		
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(2)
PROMO_NAME		VARCHAR2(10)
PROMO_CAT		VARCHAR2(10)
PROMO_COST		NUMBER(8,2)
PROMO_BEGIN_DATE		DATE
PROMO_END_DATE		DATE
SALES		
Name	Null?	Type
PROD_ID	NOT NULL	NUMBER(3)
PROMO_ID	NOT NULL	NUMBER(3)
TIME_ID		DATE
QTY_SOLD		NUMBER(6,2)
CUST_ID	NOT NULL	NUMBER(2)
CUSTOMER		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER(3)
CUST_NAME		VARCHAR2(20)
CUST_ADDRESS		VARCHAR2(30)

You need to generate a report showing the promo name along with the customer name for all products that were sold during their promo campaign and before 30th October 2007.

You issue the following query:

Which statement is true regarding the above query?

- A. It executes successfully and gives the required resul
- B. It executes successfully but does not give the required resul
- C. It produces an error because the join order of the tables is incorrec
- D. It produces an error because equijoin and nonequijoin conditions cannot be used in the same SELECT statemen

**Answer: B**

#### NEW QUESTION 7

Examine the structure of the EMPLOYEES table:

EMPLOYEE_ID	NUMBER	Primary Key
FIRST_NAME	VARCHAR2(25)	
LAST_NAME	VARCHAR2(25)	
HIRE_DATE	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 = 'Smoth' 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**

#### NEW QUESTION 8

The STUDENT\_GRADES table has these columns:

STUDENT\_IDNUMBER(12)

SEMESTER\_ENDDATE

GPANUMBER(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

#### NEW QUESTION 9

Which three are true? (Choose three.)

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

**Answer:** ACD

**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

#### NEW QUESTION 10

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')

#### NEW QUESTION 10

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

### NEW QUESTION 13

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.

### NEW QUESTION 14

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 result
- 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:** Explanation: the two statements produce identical results as ORDER BY 2 will take the second column as sorting column.

Incorrect Answer: B there is no syntax error C results are sorted in ascending order by default D ORDER BY 2 will take the second column as sorting column. Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 2-22

### NEW QUESTION 17

View the Exhibit and examine the structure of the PRODUCTS table. You need to generate a report in the following format: CATEGORIES 5MP Digital Photo Camera's category is Photo Y Box's category is Electronics Envoy Ambassador's category is Hardware Which two queries would give the required output? (Choose two.)

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)

- A. SELECT prod\_name || q''s category is ' || prod\_category CATEGORIES FROM products;
- B. SELECT prod\_name || q['s 'category is ' || prod\_category CATEGORIES FROM products;
- C. SELECT prod\_name || q\'s\' || ' category is ' || prod\_category CATEGORIES FROM products;
- D. SELECT prod\_name || q'<s >' || 'category is ' || prod\_category CATEGORIES FROM products;

**Answer:** CD

**Explanation:**

So, how are words that contain single quotation marks dealt with? There are essentially two mechanisms available. The most popular of these is to add an additional single quotation mark next to each naturally occurring single quotation mark in the character string Oracle offers a neat way to deal with this type of character literal in the form of the alternative quote (q) operator. Notice that the problem is that Oracle chose the single quote characters as the special pair of symbols that enclose or wrap any other character literal. These character-enclosing symbols could have been anything other than single quotation marks. Bearing this in mind, consider the alternative quote (q) operator. The q operator enables you to choose from a set of possible pairs of wrapping symbols for character literals as alternatives to the single quote symbols. The options are any single-byte or multibyte character or the four brackets: (round brackets), {curly braces}, [squarebrackets], or <angle brackets>. Using the q operator, the character delimiter can effectively be changed from a single quotation mark to any other character. The syntax of the alternative quote operator is as follows: q'delimiter'character literal which may include the single quotes delimiter' where delimiter can be any character or bracket.

Alternative Quote (q) Operator

Specify your own quotation mark delimiter.

Select any delimiter.

Increase readability and usability.

SELECT department\_name || q'[ Department's Manager Id: ]'

|| manager\_id

AS "Department and Manager"

FROM departments;

Alternative Quote (q) Operator

Many SQL statements use character literals in expressions or conditions. If the literal itself contains a single quotation mark, you can use the quote (q) operator and select your own quotation mark delimiter.

You can choose any convenient delimiter, single-byte or multibyte, or any of the following

character pairs: [ ], { }, ( ), or < >. In the example shown, the string contains a single quotation mark, which is normally interpreted as a delimiter of a character string. By using the q operator, however, brackets [] are used as the quotation mark delimiters. The string between the brackets delimiters is interpreted as a literal character string.

**NEW QUESTION 21**

View the Exhibit and examine the structure of the PROMOTIONS table. Examine the following two SQL statements:

Statement 1

```
SQL>SELECT promo_category,SUM(promo_cost)
FROM promotions
WHERE promo_end_date-promo_begin_date > 30
GROUP BY promo_category;
```

Statement 2

```
SQL>SELECT promo_category,sum(promo_cost)
FROM promotions
GROUP BY promo_category
HAVING MIN(promo_end_date-promo_begin_date)>30;
```

Which statement is true regarding the above two SQL statements?

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

- A. statement 1 gives an error, statement 2 executes successfully
- B. statement 2 gives an error, statement 1 executes successfully
- C. statement 1 and statement 2 execute successfully and give the same output
- D. statement 1 and statement 2 execute successfully and give a different output

**Answer:** D

**NEW QUESTION 23**

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 statement
- 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 function
- 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:** BF

**Explanation:** 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: A sub query can retrieve zero or more rows C sub query is not SQL query statement D sub query can be nested E group function can be use with sub query

### NEW QUESTION 28

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 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:** AD

### NEW QUESTION 31

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:** CD

### 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

### NEW QUESTION 35

View the Exhibit and examine the structure of the CUSTOMERS table. Evaluate the following SQL statement:

```
SQL> SELECT cust_city, COUNT(cust_last_name)
FROM customers
WHERE cust_credit_limit > 1000
GROUP BY cust_city
HAVING AVG(cust_credit_limit) BETWEEN 5000 AND 6000;
```

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

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)

- A. It executes successfully
- B. It returns an error because the BETWEEN operator cannot be used in the HAVING clause
- C. It returns an error because WHERE and HAVING clauses cannot be used in the same SELECT statement
- D. It returns an error because WHERE and HAVING clauses cannot be used to apply conditions on the same column

Answer: A

### NEW QUESTION 39

You need to perform these tasks:

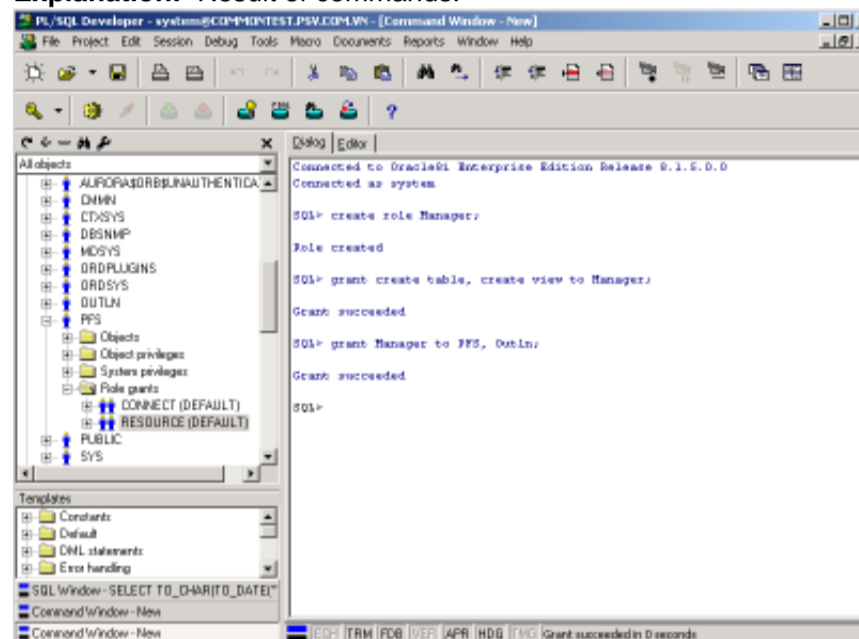
. Create and assign a MANAGER role to Blake and Clark . Grant CREATE TABLE and CREATE VIEW privileges to Blake and Clark

Which set of SQL statements achieves the desired results?

- A. CREATE ROLE manager; GRANT create table, create view TO manager; GRANT manager TO BLAKE,CLARK;
- B. CREATE ROLE manager; GRANT create table, create voew TO manager; GRANT manager ROLE TO BLAKE,CLARK;
- C. GRANT manager ROLE TO BLAKE,CLARK; GRANT create table, create voew TO BLAKE CLARK; \*\*\*MISSING\*\*\*

Answer: A

**Explanation:** Result of commands:



### NEW QUESTION 40

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

The following query is written to retrieve all those product IDs from the SALES table that have more than 55000 sold and have been ordered more than 10 times.

```
SQL> SELECT prod_id
FROM sales
WHERE quantity_sold > 55000 AND COUNT(*)>10
GROUP BY prod_id
HAVING COUNT(*)>10;
```

Which statement is true regarding this SQL statement?

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)

- A. It executes successfully and generates the required resul
- B. It produces an error because COUNT(\*) should be specified in the SELECT clause als
- C. It produces an error because COUNT(\*) should be only in the HAVING clause and not in the WHERE claus
- D. It executes successfully but produces no result because COUNT(prod\_id) should be used instead of COUNT(\*)).

Answer: C

**Explanation:**

Restricting Group Results with the HAVING Clause

You use the HAVING clause to specify the groups that are to be displayed, thus further restricting the groups on the basis of aggregate information.

In the syntax, group\_condition restricts the groups of rows returned to those groups for which the specified condition is true.

The Oracle server performs the following steps when you use the HAVING clause:

- Rows are grouped.
  - The group function is applied to the group.
  - The groups that match the criteria in the HAVING clause are displayed.
- The HAVING clause can precede the GROUP BY clause, but it is recommended that you place the GROUP BY clause first because it is more logical. Groups are formed and group functions are calculated before the HAVING clause is applied to the groups in the SELECT list.

Note: The WHERE clause restricts rows, whereas the HAVING clause restricts groups.

#### NEW QUESTION 42

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**

#### NEW QUESTION 47

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

#### NEW QUESTION 50

View the Exhibit and examine the structure of the CUSTOMERS table. Evaluate the query statement:

```
SQL> SELECT cust_last_name, cust_city, cust_credit_limit
FROM customers
WHERE cust_last_name BETWEEN 'A' AND 'C' AND cust_credit_limit BETWEEN
1000 AND 3000;
```

What would be the outcome of the above statement?

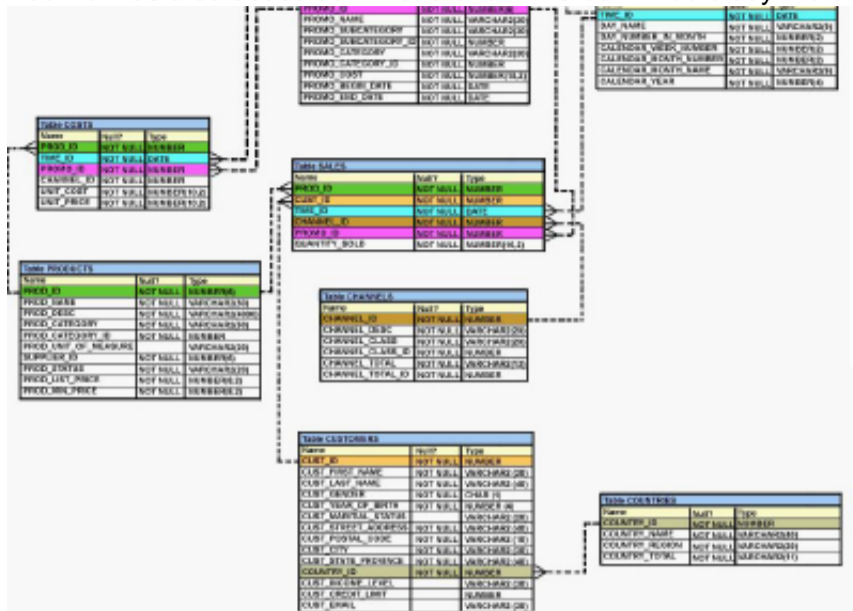
Table CUSTOMERS		
Name	Null?	Type
CUST_ID	NOT NULL	NUMBER
CUST_FIRST_NAME	NOT NULL	VARCHAR2 (20)
CUST_LAST_NAME	NOT NULL	VARCHAR2 (40)
CUST_GENDER	NOT NULL	CHAR (1)
CUST_YEAR_OF_BIRTH	NOT NULL	NUMBER (4)
CUST_MARITAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT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

- A. It executes successfully
- B. It produces an error because the condition on CUST\_LAST\_NAME is invalid
- C. It executes successfully only if the CUST\_CREDIT\_LIMIT column does not contain any null value
- D. It produces an error because the AND operator cannot be used to combine multiple BETWEEN clauses

**Answer: A**

### NEW QUESTION 55

You work as a database administrator at ABC.com. You study the exhibit carefully and examine the structure of CUSTOMRS 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**

### NEW QUESTION 59

Which one is a system privilege?

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

**Answer: E**

### NEW QUESTION 61

Evaluate the following SQL statements: Exhibit:

```
SELECT INTERVAL '300' MONTH,
       INTERVAL '54-2' YEAR TO MONTH,
       INTERVAL '11:12:10.1234567' HOUR TO SECOND
FROM dual;
```

Which is the correct output of the above query?

- A. +00-300, +54-02, +00 11:12:10.123457
- B. +00-300, +00-650, +00 11:12:10.123457
- C. +25-00, +54-02, +00 11:12:10.123457
- D. +25-00, +00-650, +00 11:12:10.123457

**Answer: C**

### NEW QUESTION 65

The PART\_CODE column in the SPARES table contains the following list of values:

```
PART_CODE
-----
A%_WQ123
A%BWQ123
AB_WQ123
```

Evaluate the following query:

```
SQL> SELECT part_code
FROM spares
WHERE part_code LIKE '%\%_WQ12%' ESCAPE '\';
```

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

- A. It produces an error
- B. It displays all values
- C. It displays only the values A%\_WQ123 and AB\_WQ123.
- D. It displays only the values A%\_WQ123 and A%BWQ123.
- E. It displays only the values A%BWQ123 and AB\_WQ123.

**Answer:** D

**Explanation:**

Combining Wildcard Characters

The % and \_ symbols can be used in any combination with literal characters. The example in the slide displays the names of all employees whose last names have the letter "o" as the second character.

ESCAPE Identifier

When you need to have an exact match for the actual % and \_ characters, use the ESCAPE identifier. This option specifies what the escape character is. If you want to search for strings that contain SA\_, you can use the following SQL statement: SELECT employee\_id, last\_name, job\_id FROM employees WHERE job\_id LIKE '%SA\\_%' ESCAPE '\';

**NEW QUESTION 67**

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:

B Non alphanumeric character such as "\*" is discouraged in Oracle table name.

D Table name must begin with a letter.

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 9-4

**NEW QUESTION 72**

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

Table PROMOTIONS		
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(8)
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

Exhibit:

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:** AB

**NEW QUESTION 76**

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

### NEW QUESTION 77

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

You have been asked to produce a report on the CUSTOMERS table showing the customers details sorted in descending order of the city and in the descending order of their income level in each city. Which query would accomplish this task?

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)

- A. SELECT cust\_city, cust\_income\_level, cust\_last\_name FROM customers ORDER BY cust\_city desc, cust\_income\_level DESC;
- B. SELECT cust\_city, cust\_income\_level, cust\_last\_name FROM customers ORDER BY cust\_income\_level desc, cust\_city DESC;
- C. SELECT cust\_city, cust\_income\_level, cust\_last\_name FROM customers ORDER BY (cust\_city, cust\_income\_level) DESC;
- D. SELECT cust\_city, cust\_income\_level, cust\_last\_name FROM customers ORDER BY cust\_city, cust\_income\_level DESC;

**Answer:** A

### NEW QUESTION 81

Which arithmetic operations can be performed on a column by using a SQL function that is built into Oracle database? (Choose three.)

- A. addition
- B. subtraction
- C. raising to a power
- D. finding the quotient
- E. finding the lowest value

**Answer:** ACE

### NEW QUESTION 85

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:** AB

### NEW QUESTION 87

Examine the statement:

Create synonym emp for hr.employees;

What happens when you issue the statement?

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

**Answer:** D

### NEW QUESTION 88

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

- A. The UNIQUE constraint does not permit a null value for the column
- B. A UNIQUE index gets created for columns with PRIMARY KEY and UNIQUE constraint
- C. The PRIMARY KEY and FOREIGN KEY constraints create a UNIQUE index
- D. The NOT NULL constraint ensures that null values are not permitted for the column

**Answer:** BD

**Explanation:**

B: A unique constraint can contain null values because null values cannot be compared to anything.

D: The NOT NULL constraint ensures that null values are not permitted for the column

Incorrect Answer: A statement is not true C statement is not true

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 10-9

**NEW QUESTION 90**

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

Table PROMOTIONS		
Name	Null?	Type
PROMO_ID	NOT NULL	NUMBER(8)
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

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

**Answer:** AD

**NEW QUESTION 94**

Examine the structure and data of the CUSTJTRANS table:

CUSTJTRANS

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-mon-rr in the CUSTJTRANS table. Which three SQL statements would execute successfully? (Choose three.)

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

**Answer:** ACD

**NEW QUESTION 99**

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

COSTS			
PROD_ID	PROMO_ID	UNIT_COST	UNIT_PRICE
14	111	900	1129
15	333	875	1075
16	333	700	900
17	444	1000	1150

You need to generate a report that displays the IDs of all products in the COSTS table whose unit price is at least 25% more than the unit cost. The details should be displayed in the descending order of 25% of the unit cost. You issue the following query:

```
SQL>SELECT prod_id
FROM costs
WHERE unit_price >= unit_cost * 1.25
ORDER BY unit_cost * 0.25 DESC;
```

Which statement is true regarding the above query?

- A. It executes and produces the required result
- B. It produces an error because an expression cannot be used in the ORDER BY clause
- C. It produces an error because the DESC option cannot be used with an expression in the ORDER BY clause
- D. It produces an error because the expression in the ORDER BY clause should also be specified in the SELECT clause

**Answer:** A

#### NEW QUESTION 103

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

#### NEW QUESTION 105

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):

#### NEW QUESTION 108

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

- A. SELECT SYSDATE - '01-JAN-2007' FROM DUAL;
- B. SELECT SYSDATE - TOJDATE(X)1/JANUARY/2007") FROM DUAL;
- C. SELECT SYSDATE - TOJDATE('01-JANUARY-2007') FROM DUAL;
- D. SELECT TO\_CHAR(SYSDATE
- E. 'DD-MON-YYYY') - '01-JAN-2007' FROM DUAL;
- F. SELECT TO\_DATE(SYSDATE
- G. 'DD/MONTH/YYYY') - '01/JANUARY/2007' FROM DUAL;

**Answer:** BC

#### NEW QUESTION 109

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

**NEW QUESTION 110**

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

Using the PROMOTIONS table, you need to find out the names and cost of all the promos done on 'TV' and 'internet' that ended in the time interval 15th March '00 to 15th October '00.

Which two queries would give the required result? (Choose two.)

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

- A. SELECT promo\_name, promo\_cost FROM promotions WHERE promo\_category IN ('TV', 'internet') AND promo\_end\_date BETWEEN '15-MAR-00' AND '15-OCT-00';
- B. SELECT promo\_name, promo\_cost FROM promotions WHERE promo\_category = 'TV' OR promo\_category = 'internet' AND promo\_end\_date >='15-MAR-00' OR promo\_end\_date <='15-OCT-00';
- C. SELECT promo\_name, promo\_cost FROM promotions WHERE (promo\_category BETWEEN 'TV' AND 'internet') AND (promo\_end\_date IN ('15-MAR-00','15-OCT-00'));
- D. SELECT promo\_name, promo\_cost FROM promotions WHERE (promo\_category = 'TV' OR promo\_category = 'internet') AND (promo\_end\_date >='15-MAR-00' AND promo\_end\_date <='15-OCT-00');

**Answer: AD**

**NEW QUESTION 115**

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: AC**

**Explanation:**

The NVL2 Function

The NVL2 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.

**NEW QUESTION 117**

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:** AD

#### NEW QUESTION 118

You need to display the first names of all customers from the CUSTOMERS table that contain the character 'e' and have the character 'a' in the second last position.  
Which query would give the required output?

- A. SELECT cust\_first\_name FROM customers WHERE INSTR(cust\_first\_name, 'e')<>0 AND SUBSTR(cust\_first\_name, -2, 1)='a';
- B. SELECT cust\_first\_name FROM customers WHERE INSTR(cust\_first\_name, 'e')<>" AND SUBSTR(cust\_first\_name, -2, 1)='a';
- C. SELECT cust\_first\_name FROM customers WHERE INSTR(cust\_first\_name, 'e')IS NOT NULL AND SUBSTR(cust\_first\_name, 1,-2)='a';
- D. SELECT cust\_first\_name FROM customers WHERE INSTR(cust\_first\_name, 'e')<>0 AND SUBSTR(cust\_first\_name, LENGTH(cust\_first\_name),-2)='a';

**Answer:** A

#### Explanation:

The SUBSTR(string, start position, number of characters) function accepts three parameters and returns a string consisting of the number of characters extracted from the source string, beginning at the specified start position:

substr('http://www.domain.com',12,6) = domain

The position at which the first character of the returned string begins.

When position is 0 (zero), then it is treated as 1.

When position is positive, then the function counts from the beginning of string to find the first character.

When position is negative, then the function counts backward from the end of string.

substring\_length

The length of the returned string. SUBSTR calculates lengths using characters as defined by the input character set. SUBSTRB uses bytes instead of characters. SUBSTRC uses Unicode complete characters.

SUBSTR2 uses UCS2 code points. SUBSTR4 uses UCS4 code points.

When you do not specify a value for this argument, then the function

The INSTR(source string, search item, [start position],[nth occurrence of search item])

function returns a number that represents the position in the source string, beginning from the given start position, where the nth occurrence of the search item begins:

instr('http://www.domain.com','.',1,2) = 18

#### NEW QUESTION 120

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:** AE

#### NEW QUESTION 125

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

**NEW QUESTION 129**

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

**NEW QUESTION 130**

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', 'miDdspth "of" Mont
- B. Year') FROM DUAL;
- C. SELECT TO\_CHAR(TO\_DATE('11-oct-2007', 'X', 'miDdspth of month, year') FROM DUAL;
- D. SELECT TO\_CHAR(TO\_DATE('11-oct-2007'), 'miDdspth "of" Mont
- E. Year') FROM DUAL;
- F. SELECT TO\_CHAR(TO\_DATE('11-oct-2007', 'fmmiDdspth "of" Mont
- G. Year')) FROM DUAL;

**Answer:** A

**NEW QUESTION 134**

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 column
- 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 function

**Answer:** AD

**NEW QUESTION 135**

Which statement is true regarding sub queries?

- A. The LIKE operator cannot be used with single-row subqueries
- B. The NOT IN operator is equivalent to IS NULL with single-row subqueries
- C. =ANY and =ALL operators have the same functionality in multiple-row subqueries
- D. The NOT operator can be used with IN, ANY, and ALL operators in multiple-row subqueries

**Answer:** D

**Explanation:**

Using the ANY Operator in Multiple-Row Subqueries

The ANY operator (and its synonym, the SOME operator) compares a value to each value returned by a subquery.

<ANY means less than the maximum.

>ANY means more than the minimum.

=ANY is equivalent to IN

Using the ALL Operator in Multiple-Row Subqueries

The ALL operator compares a value to every value returned by a subquery.

>ALL means more than the maximum and

<ALL means less than the minimum.

The NOT operator can be used with IN, ANY, and ALL operators.

**NEW QUESTION 136**

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:** BC

#### NEW QUESTION 140

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. Guidelines

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 month

MON Three-letter abbreviation of the month

MONTH Full name of the month

YYYY Full year in numbers

YEAR Year spelled out (in English)

#### NEW QUESTION 144

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

#### NEW QUESTION 148

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:** BD

#### 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.

#### NEW QUESTION 150

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_SEQ, EMP_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

#### NEW QUESTION 152

View the Exhibit and examine the structure of ORD and ORD\_ITEMS tables.

The ORD\_NO column is PRIMARY KEY in the ORD table and the ORD\_NO and ITEM\_NO columns are composite PRIMARY KEY in the ORD\_ITEMS table.

Which two CREATE INDEX statements are valid? (Choose two.)

ORD

Name	Null?	Type
ORD_NO	NOT NULL	NUMBER(2)
ORD_DATE		DATE
CUST_ID		NUMBER(4)

ORD\_ITEMS

Name	Null?	Type
ORD_NO	NOT NULL	NUMBER(2)
ITEM_NO	NOT NULL	NUMBER(3)
QTY		NUMBER(8,2)

- A. CREATE INDEX ord\_idx1 ON ord(ord\_no);
- B. CREATE INDEX ord\_idx2 ON ord\_items(ord\_no);
- C. CREATE INDEX ord\_idx3 ON ord\_items(item\_no);
- D. CREATE INDEX ord\_idx4 ON ord,ord\_items(ord\_no, ord\_date,qty);

**Answer:** BC

**Explanation:** How Are Indexes Created?

You can create two types of indexes.

Unique index: The Oracle server automatically creates this index when you define a column in a table to have a PRIMARY KEY or a UNIQUE constraint. The name of the index is the name that is given to the constraint.

Nonunique index: This is an index that a user can create. For example, you can create the FOREIGN KEY column index for a join in a query to improve the speed of retrieval.

Note: You can manually create a unique index, but it is recommended that you create a unique constraint, which implicitly creates a unique index.

#### NEW QUESTION 156

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

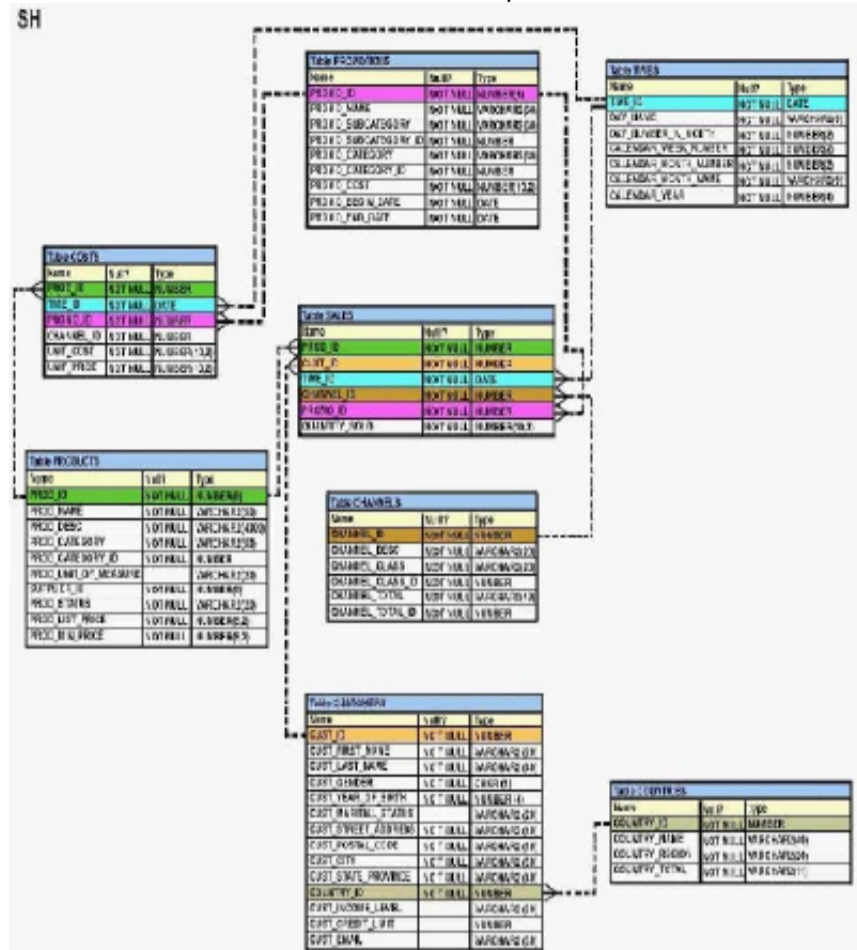
You can update multiple columns in the SET clause of an UPDATE statement by writing multiple subqueries. The syntax is as follows:

```
UPDATE table
SET column =
(SELECT column
FROM table
WHERE condition)
[ ,
column =
(SELECT column
FROM table
```

WHERE condition)]  
[WHERE condition ] :

## NEW QUESTION 158

View the Exhibit and examine the description for the PRODUCTS and SALES table.



PROD\_ID is a primary key in the PRODUCTS table and foreign key in the SALES table. You want to remove all the rows from the PRODUCTS table for which no sale was done for the last three years. Which is the valid DELETE statement?

- A. DELETE FROM products WHERE prod\_id = (SELECT prod\_id FROM sales WHERE time\_id - 3\*365 = SYSDATE );  
 B. DELETE FROM products WHERE prod\_id = (SELECT prod\_id FROM sales WHERE SYSDATE >= time\_id - 3\*365 );  
 C. DELETE FROM products WHERE prod\_id IN (SELECT prod\_id FROM sales WHERE SYSDATE - 3\*365 >= time\_id);  
 D. DELETE FROM products WHERE prod\_id IN (SELECT prod\_id FROM sales WHERE time\_id >= SYSDATE - 3\*365 );

**Answer: C**

## NEW QUESTION 160

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.

## NEW QUESTION 161

Evaluate the following SQL statement:

```
SQL> SELECT cust_id, cust_last_name FROM customers WHERE cust_credit_limit IN (select cust_credit_limit FROM customers WHERE cust_city='Singapore');
```

Which statement is true regarding the above query if one of the values generated by the sub query is NULL?

- A. It produces an error  
B. It executes but returns no row  
C. It generates output for NULL as well as the other values produced by the sub quer  
D. It ignores the NULL value and generates output for the other values produced by the sub quer

**Answer: C**

## NEW QUESTION 165

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

#### NEW QUESTION 167

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:** BC

#### 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.

#### NEW QUESTION 171

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

- A. It can be used to join a maximum of three table
- B. It can be used to restrict the number of columns used in a NATURAL join
- C. It can be used to access data from tables through equijoins as well as nonequijoin
- D. It can be used to join tables that have columns with the same name and compatible data type

**Answer:** BD

#### Explanation:

NATURAL JOIN operation A NATURAL JOIN is a JOIN operation that creates an implicit join clause for you based on the common columns in the two tables being joined. Common columns are columns that have the same name in both tables. If the SELECT statement in which the NATURAL JOIN operation appears has an asterisk (\*) in the select list, the asterisk will be expanded to the following list of columns (in this order):

All the common columns

Every column in the first (left) table that is not a common column

Every column in the second (right) table that is not a common column

An asterisk qualified by a table name (for example, COUNTRIES.\*) will be expanded to every column of that table that is not a common column.

If a common column is referenced without being qualified by a table name, the column reference points to the column in the first (left) table if the join is an INNER JOIN or a LEFT OUTER JOIN. If it is a RIGHT OUTER JOIN, unqualified references to a common column point to the column in the second (right) table.

Syntax

TableExpression NATURAL [ { LEFT | RIGHT } [ OUTER ] | INNER ] JOIN {

TableViewOrFunctionExpression |

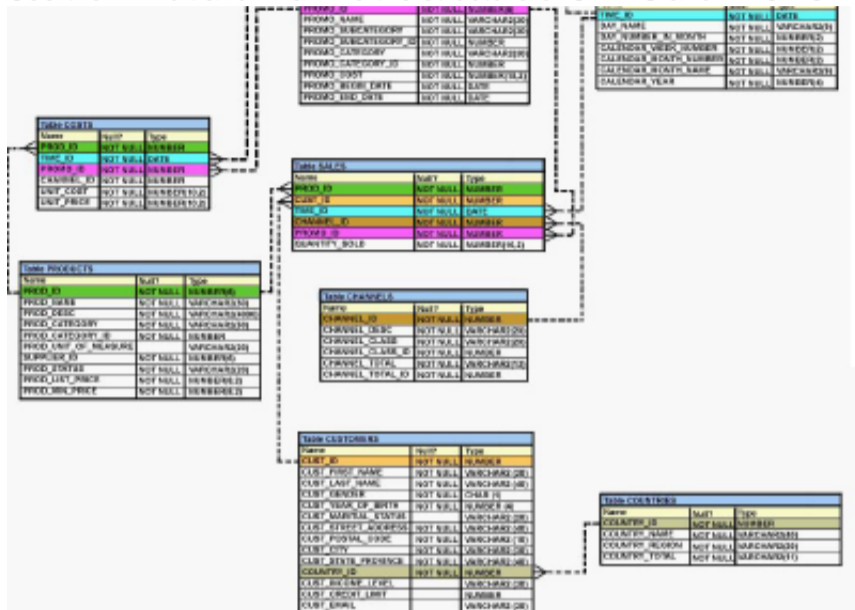
( TableExpression ) }

Examples

If the tables COUNTRIES and CITIES have two common columns named COUNTRY and COUNTRY\_ISO\_CODE, the following two SELECT statements are equivalent:  
SELECT \* FROM COUNTRIES NATURAL JOIN CITIES  
SELECT \* FROM COUNTRIES JOIN CITIES  
USING (COUNTRY, COUNTRY\_ISO\_CODE)

#### NEW QUESTION 172

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 PROMOTIONS 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:** BCD

#### NEW QUESTION 176

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:** BCD

#### NEW QUESTION 177

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 that 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:** AC

#### NEW QUESTION 179

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

```
SQL>SELECT prod_name, prod_list_price -{(prod_list_price*(25/100))
+(prod_list_price -(prod_list_price*(25/100))*(15/100))+100
AS "TOTAL PRICE"
FROM products;
```

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**

#### NEW QUESTION 180

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?

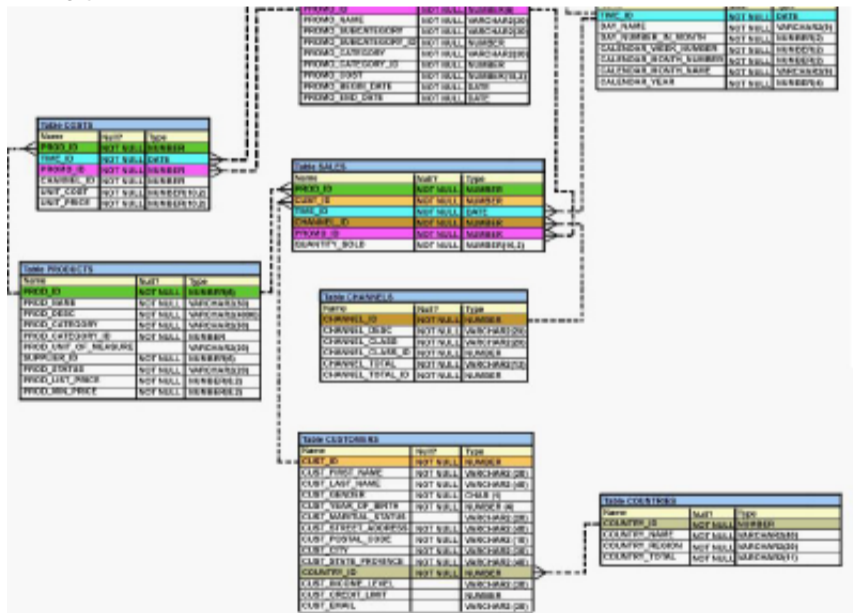
- A. SELECT student\_id, gpa FROM student\_grades WHERE semester\_end BETWEEN '01-JAN-2001' AND '31-DEC-2001' OR gpa > 3.;
- B. SELECT student\_id, gpa FROM student\_grades WHERE semester\_end BETWEEN '01-JAN-2001' AND '31-DEC-2001' AND gpa gt 3.0;
- C. SELECT student\_id, gpa FROM student\_grades WHERE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

#### NEW QUESTION 184

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

### NEW QUESTION 188

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

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

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

There is only one customer with the CUST\_LAST\_NAME column having value Roberts. Which INSERT statement should be used to add a row into the ORDERS table for the customer whose CUST\_LAST\_NAME is Roberts and CREDIT\_LIMIT is 600?

- A. INSERT INTO orders VALUES (1,'10-mar-2007', 'direct', (SELECT customer\_id FROM customers WHERE cust\_last\_name='Roberts' AND credit\_limit=600), 1000);
- B. INSERT INTO orders (order\_id,order\_date,order\_mode, (SELECT customer\_id FROM customers WHERE cust\_last\_name='Roberts' AND credit\_limit=600),order\_total) VALUES(1,'10-mar-2007', 'direct', &&customer\_id, 1000);
- C. INSERT INTO(SELECT o.order\_id, o.order\_date,o.order\_mode,c.customer\_id, o.order\_total FROM orders o, customers c WHERE o.customer\_id = c.customer\_id AND c.cust\_last\_name='Roberts' ANDc.credit\_limit=600 ) VALUES (1,'10-mar-2007', 'direct',(SELECT customer\_id FROM customers WHERE cust\_last\_name='Roberts' AND credit\_limit=600), 1000);
- D. INSERT INTO orders (order\_id,order\_date,order\_mode, (SELECT customer\_id FROM customers WHERE cust\_last\_name='Roberts' AND credit\_limit=600),order\_total) VALUES(1,'10-mar-2007', 'direct', &customer\_id, 1000);

Answer: A

### NEW QUESTION 189

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: BCD

### NEW QUESTION 192

Examine the structure and data in the PRIC E\_LIST table: Name Null? Type

PROD\_D NOT NULL NUMBER(3)

PROD\_PRICE VARCHAR2(10)

PROD\_ID PROD PRICE

100 \$234.55

101 \$6,509.75

102 \$1,234

in the same format as the PROD\_PRICE. Which SQL statement would give the required result?

- A. SELECT TO\_CHAR(prod\_price\* .25.'\$99.999.99') FROM PRICEJLIST;
- B. SELECT TO\_CHAR(TO\_NUMBER(prod\_price)\* .25.'\$99.999.00') FROM PRICE\_LIST;
- C. SELECT TO\_CRAR(TO\_NUMBER(prod\_price.'S99.999.99')\* .25.'\$99.999.00') FROM PRICE\_LIST;
- D. SELECT TO\_NUMBER(TO\_NUMBER(prod\_price.,\$99.999.99')\* .25/\$99.999.00') FROM PRICE\_LIST;

Answer: C

### NEW QUESTION 193

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

#### NEW QUESTION 196

View the Exhibit to examine the description for the SALES table. Which views can have all DML operations performed on it? (Choose all that apply.)

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)

- A. CREATE VIEW v3 AS SELECT \* FROM SALES WHERE cust\_id = 2034 WITH CHECK OPTION;
- B. CREATE VIEW v1 AS SELECT \* FROM SALES WHERE time\_id <= SYSDATE - 2\*365 WITH CHECK OPTION;
- C. CREATE VIEW v2 AS SELECT prod\_id, cust\_id, time\_id FROM SALES WHERE time\_id <= SYSDATE - 2\*365 WITH CHECK OPTION;
- D. CREATE VIEW v4 AS SELECT prod\_id, cust\_id, SUM(quantity\_sold) FROM SALES WHERE time\_id <= SYSDATE - 2\*365 GROUP BY prod\_id, cust\_id WITH CHECK OPTION;

**Answer:** AB

#### Explanation:

Creating a View You can create a view by embedding a subquery in the CREATE VIEW statement. In the syntax: CREATE [OR REPLACE] [FORCE|NOFORCE] VIEW view [(alias[, alias]...)] AS subquery [WITH CHECK OPTION [CONSTRAINT constraint]] [WITH READ ONLY [CONSTRAINT constraint]]; OR REPLACE Re-creates the view if it already exists FORCE Creates the view regardless of whether or not the base tables exist NOFORCE Creates the view only if the base tables exist (This is the default.) View Is the name of the view alias Specifies names for the expressions selected by the view's query (The number of aliases must match the number of expressions selected by the view.) subquery Is a complete SELECT statement (You can use aliases for the columns in the SELECT list.) WITH CHECK OPTION Specifies that only those rows that are accessible to the view can be inserted or updated ANSWER D constraint Is the name assigned to the CHECK OPTION constraint WITH READ ONLY Ensures that no DML operations can be performed on this view 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 – ANSWER C

#### NEW QUESTION 197

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:** ACD

#### NEW QUESTION 201

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**

#### NEW QUESTION 202

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.

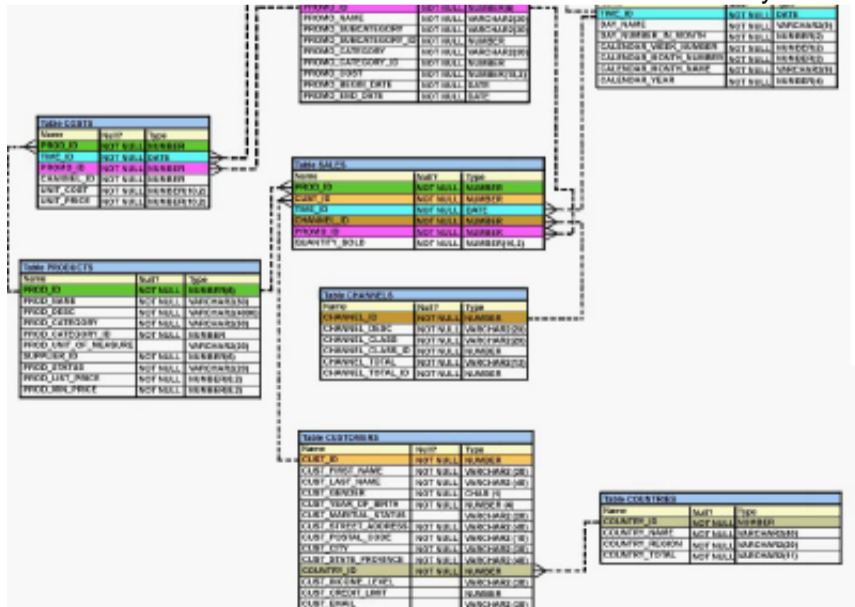
Which two SQL statements would give the required output? (Choose two.)

- 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: AC**

#### NEW QUESTION 203

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 DML 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

**NEW QUESTION 208**

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

CUSTOMERS			
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**

**NEW QUESTION 212**

View the Exhibit and examine the structure of the PROMOTIONS table. Evaluate the following SQL statement:

```
SQL>SELECT promo_category, AVG(promo_cost) Avg_Cost,
AVG(promo_cost)* 25 Avg_Overhead
FROM promotions
WHERE UPPER(promo_category) IN ('TV', 'INTERNET','POST')
GROUP BY Avg_Cost
ORDER BY Avg_Overhead;
```

The above query generates an error on execution.

Which clause in the above SQL statement causes the error?

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

- A. WHERE
- B. SELECT
- C. GROUP BY
- D. ORDER BY

**Answer: C**

**NEW QUESTION 213**

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:** BD

#### NEW QUESTION 214

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.

#### NEW QUESTION 217

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),

#### NEW QUESTION 219

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: A Object privilege on VIEW is DELETE, INSERT, REFERENCES, SELECT and UPDATE B Object privilege on VIEW is DELETE, INSERT, REFERENCES, SELECT and UPDATE C Object privilege on VIEW is DELETE, INSERT, REFERENCES, SELECT and UPDATE

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 13-12

#### NEW QUESTION 224

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)

Which two tasks would require subqueries or joins to be executed in a single statement? (Choose two.)

- A. listing of customers who do not have a credit limit and were born before 1980
- B. finding the number of customers, in each city, whose marital status is 'married'
- C. finding the average credit limit of male customers residing in 'Tokyo' or 'Sydney'
- D. listing of those customers whose credit limit is the same as the credit limit of customers residing in the city 'Tokyo'
- E. finding the number of customers, in each city, whose credit limit is more than the average credit limit of all the customers

**Answer:** DE

#### Explanation:

Describe the Types of Problems That the Subqueries Can Solve There are many situations where you will need the result of one query as the input for another.

Use of a Subquery Result Set for Comparison Purposes Which employees have a salary that is less than the average salary? This could be answered by two statements, or by a single statement with a subquery. The following example uses two statements: select avg(salary) from employees; select last\_name from employees where salary < result\_of\_previous\_query ;

Alternatively, this example uses one statement with a subquery:

select last\_name from employees where salary < (select avg(salary) from employees);

In this example, the subquery is used to substitute a value into the WHERE clause of the parent query: it is returning a single value, used for comparison with the rows retrieved by the parent query.

The subquery could return a set of rows. For example, you could use the following to find all departments that do actually have one or more employees assigned to them:

select department\_name from departments where department\_id in  
(select distinct(department\_id) from employees);

#### NEW QUESTION 227

Examine the structure of the TRANSACTIONS table:

Name Null Type  
TRANS\_ID NOT NULL NUMBER(3)  
CUST\_NAME VARCHAR2(30)  
TRANS\_DATE TIMESTAMP  
TRANS\_AMT NUMBER(10,2)

You want to display the date, time, and transaction amount of transactions that where done before 12 noon.

The value zero should be displayed for transactions where the transaction amount has not been entered.

Which query gives the required result?

- A. SELECT TO\_CHAR(trans\_date,'dd-mon-yyyy hh24:mi:ss'), TO\_CHAR(trans\_amt,'\$99999999D99') FROM transactions WHERE TO\_NUMBER(TO\_DATE(trans\_date,'hh24')) < 12 AND COALESCE(trans\_amt,NULL)<>NULL;
- B. SELECT TO\_CHAR(trans\_date,'dd-mon-yyyy hh24:mi:ss'), NVL(TO\_CHAR(trans\_amt,'\$99999999D99'),0) FROM transactions WHERE TO\_CHAR(trans\_date,'hh24') < 12;
- C. SELECT TO\_CHAR(trans\_date,'dd-mon-yyyy hh24:mi:ss'), COALESCE(TO\_NUMBER(trans\_amt,'\$99999999.99'),0) FROM transactions WHERE TO\_DATE(trans\_date,'hh24') < 12;

D. SELECT TO\_DATE (trans\_date,'dd-mon-yyyy hh24:mi:ss'), NVL2(trans\_amt,TO\_NUMBER(trans\_amt,'\$99999999.99'), 0) FROM transactions WHERE TO\_DATE(trans\_date,'hh24') < 12;

**Answer:** B

#### NEW QUESTION 231

The STUDENT\_GRADES table has these columns:

STUDENT\_ID NUMBER(12)

SEMESTER\_END DATE

GPA NUMBER(4,3)

Which statement finds the highest grade point average (GPA) per semester?

- A. SELECT MAX(gpa) FROM student\_grades WHERE gpa IS NOT NULL;
- B. SELECT (gpa) FROM student\_grades GROUP BY semester\_end WHERE gpa IS NOT NULL;
- C. SELECT MAX(gpa) FROM student\_grades WHERE gpa IS NOT NULL GROUP BY semester\_end;
- D. SELECT MAX(gpa) GROUP BY semester\_end WHERE gpa IS NOT NULL FROM student\_grades;
- E. SELECT MAX(gpa) FROM student\_grades GROUP BY semester\_end WHERE gpa IS NOT NULL;

**Answer:** C

**Explanation:** Explanation: For highest gpa value MAX function is needed, for result with per semester GROUP BY clause is needed

Incorrect Answer: A per semester condition is not included B result would not display the highest gpa value D invalid syntax error E invalid syntax error Refer:

Introduction to Oracle9i: SQL, Oracle University Study Guide, 5-7

#### NEW QUESTION 235

View the Exhibit and examine the structure of the PRODUCTS, SALES, and SALE\_SUMMARY tables.

SALE\_VW is a view created using the following command:

```
SQL>CREATE VIEW sale_vw AS
```

```
SELECT prod_id, SUM(quantity_sold) QTY_SOLD
```

```
FROM sales GROUP BY prod_id;
```

You issue the following command to add a row to the SALE\_SUMMARY table:

```
SQL>INSERT INTO sale_summary
```

```
SELECT prod_id, prod_name, qty_sold FROM sale_vw JOIN products
```

```
USING (prod_id) WHERE prod_id = 16;
```

What is the outcome?

- A. It executes successfully
- B. It gives an error because a complex view cannot be used to add data into the SALE\_SUMMARY table
- C. It gives an error because the column names in the subquery and the SALE\_SUMMARY table do not match
- D. It gives an error because the number of columns to be inserted does not match with the number of columns in the SALE\_SUMMARY table

**Answer:** D

#### NEW QUESTION 236

Top N analysis requires \_\_\_\_ and \_\_\_\_\_. (Choose two.)

- A. the use of ROWID
- B. a GROUP BY clause
- C. an ORDER BY clause
- D. only an inline view
- E. an inline view and an outer query

**Answer:** CE

#### Explanation:

The correct statement for Top-N Analysis SELECT [column\_list], ROWNUM FROM (SELECT [column\_list]

FROM table

ORDER BY Top-N\_column)

WHERE ROWNUM <= N;

Incorrect Answer:

ROWID is not required

B GROUP BY clause is not required

D Must have inline view and outer query.

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 11-23

#### NEW QUESTION 240

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

- A. Savepoints are effective only for COMMIT
- B. Savepoints may be used to ROLLBACK
- C. Savepoints can be used for only DML statement
- D. Savepoints are effective for both COMMIT and ROLLBACK
- E. Savepoints can be used for both DML and DDL statement

**Answer:** BC

#### NEW QUESTION 241

View the Exhibits and examine the structures of the PRODUCTS and SALES tables. Which two SQL statements would give the same output? (Choose two.)

- A. SELECT prod\_id FROM products INTERSECT SELECT prod\_id FROM sales;
- B. SELECT prod\_id FROM products MINUS SELECT prod\_id FROM sales;
- C. SELECT DISTINCT p.prod\_id FROM products p JOIN sales s ON p.prod\_id=s.prod\_id;
- D. SELECT DISTINCT p.prod\_id FROM products p JOIN sales s ON p.prod\_id <> s.prod\_id;

**Answer:** AC

#### NEW QUESTION 244

View the Exhibit and examine the data in the PROMO\_CATEGORY and PROMO\_COST columns of the PROMOTIONS table.

Evaluate the following two queries:

SQL>SELECT DISTINCT promo\_category to\_char(promo\_cost)"code"

FROM promotions

ORDER BY code;

SQL>SELECT DISTINCT promo\_category promo\_cost "code"

FROM promotions

ORDER BY 1;

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

**PROMOTIONS**

PROMO_CATEGORY	PROMO_COST
-----	-----
radio	97200
newspaper	97800
TV	97600
post	98000
internet	98200
TV	98300
internet	98700
newspaper	98500
magazine	98400
radio	99100
post	99000

- A. Only the first query executes successfull
- B. Only the second query executes successfull
- C. Both queries execute successfully but give different result
- D. Both queries execute successfully and give the same resul

**Answer:** B

#### Explanation:

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

#### NEW QUESTION 248

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

- A. SELECT TO\_CHAR(1890.55,'\$0G000D00') FROM DUAL;
- B. SELECT TO\_CHAR(1890.55,'\$9,999V99') FROM DUAL;
- C. SELECT TO\_CHAR(1890.55,'\$99,999D99') FROM DUAL;
- D. SELECT TO\_CHAR(1890.55,'\$99G999D00') FROM DUAL;
- E. SELECT TO\_CHAR(1890.55,'\$99G999D99') FROM DUAL;

**Answer:** ADE

#### NEW QUESTION 252

Evaluate the SQL statement

DROP TABLE DEPT;

Which four statements are true of the SQL statement? (Choose four)

- A. You cannot roll back this statemen
- B. All pending transactions are committe
- C. All views based on the DEPT table are delete
- D. All indexes based on the DEPT table are droppe
- E. All data in the table is deleted, and the table structure is also delete
- F. All data in the table is deleted, but the structure of the table is retain
- G. All synonyms based on the DEPT table are delete

**Answer:** ABDE

#### Explanation:

You cannot roll back DROP TABLE statement. All pending transactions related on this table are committed. If the table is dropped, Oracle automatically drops any index, trigger

and constraint associated with the table as well. All data in the table is deleted, and the table structure is also deleted.

Incorrect Answers

C: All views based on the DEPT table become invalid, but they are not deleted.

F: All data in the table is deleted, and the table structure is also deleted. Command

TRUNCATE deletes all data in the table, but does not delete the structure of the table.

G: All synonyms based on the DEPT table are not deleted after dropping the table.

OCP Introduction to Oracle 9i: SQL Exam Guide, Jason Couchman, p. 225 Chapter 5: Creating Oracle Database Objects

#### NEW QUESTION 257

Which SQL statement returns a numeric value?

A. SELECT ADD\_MONTHS(MAX(hire\_date), 6) FROM EMP;

B. SELECT ROUND(hire\_date) FROM EMP;

C. SELECT sysdate-hire\_date FROM EMP;

D. SELECT TO\_NUMBER(hire\_date + 7) FROM EMP;

**Answer: C**

#### Explanation:

DATE value subtract DATE value will return numeric value.

Incorrect Answer: A does not return numeric value B does not return numeric value D does not return numeric value

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 9-13

#### NEW QUESTION 262

View the Exhibits and examine PRODUCTS and SALES tables.

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 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)

You issue the following query to display product name and the number of times the product has been sold:

```
SQL>SELECT p.prod_name, i.item_cnt FROM (SELECT prod_id, COUNT(*) item_cnt FROM sales GROUP BY prod_id) i RIGHT OUTER JOIN products p
ON i.prod_id = p.prod_id;
```

What happens when the above statement is executed?

A. The statement executes successfully and produces the required output

B. The statement produces an error because ITEM\_CNT cannot be displayed in the outer query

C. The statement produces an error because a subquery in the FROM clause and outer-joins cannot be used together

D. The statement produces an error because the GROUP BY clause cannot be used in a subquery in the FROM clause

**Answer: A**

#### NEW QUESTION 265

Which four are attributes of single row functions? (Choose four.)

A. cannot be nested

B. manipulate data items

C. act on each row returned

D. return one result per row

E. accept only one argument and return only one value

F. accept arguments which can be a column or an expression

**Answer: BCDF**

#### Explanation:

manipulate data items, act on each row returned, return one result per row, and accept arguments that can be a column or expression.

Incorrect Answer: A is not single row attributes E functions can accept more than one argument, e.g NVL2

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 3-5

**NEW QUESTION 270**

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

- A. A subquery can retrieve zero or more row
- B. Only two subqueries can be placed at one leve
- C. A subquery can be used only in SQL query statement
- D. A subquery can appear on either side of a comparison operato
- E. There is no limit on the number of subquery levels in the WHERE clause of a SELECT statemen

**Answer:** AD

**Explanation:**

Using a Subquery to Solve a Problem Suppose you want to write a query to find out who earns a salary greater than Abel's salary. To solve this problem, you need two queries: one to find how much Abel earns, and a second query to find who earns more than that amount. You can solve this problem by combining the two queries, placing one query inside the other query. The inner query (or subquery) returns a value that is used by the outer query (or main query). Using a subquery is equivalent to performing two sequential queries and using the result of the first query as the search value in the second query. Subquery Syntax A subquery is a SELECT statement that is embedded in the clause of another SELECT statement. You can build powerful statements out of simple ones by using subqueries. They can be very useful when you need to select rows from a table with a condition that depends on the data in the table itself. You can place the subquery in a number of SQL clauses, including the following: WHERE clause HAVING clause FROM clause In the syntax: operator includes a comparison condition such as >, =, or IN Note: Comparison conditions fall into two classes: single-row operators (>, =, >=, <, <>, <=) and multiple-row operators (IN, ANY, ALL, EXISTS). The subquery is often referred to as a nested SELECT, sub-SELECT, or inner SELECT statement. The subquery generally executes first, and its output is used to complete the query condition for the main (or outer) query. Guidelines for Using Subqueries Enclose subqueries in parentheses. Place subqueries on the right side of the comparison condition for readability. (However, the subquery can appear on either side of the comparison operator.) Use single-row operators with single-row subqueries and multiple-row operators with multiple-row subqueries. Subqueries can be nested to an unlimited depth in a FROM clause but to "only" 255 levels in a WHERE clause. They can be used in the SELECT list and in the FROM, WHERE, and HAVING clauses of a query.

**NEW QUESTION 272**

View the Exhibit and examine the data in the EMPLOYEES table:

You want to display all the employee names and their corresponding manager names.

Evaluate the following query:

```
SQL> SELECT e.employee_name "EMP NAME", m.employee_name "MGR NAME"
```

```
FROM employees e _____ employees m
```

```
ON e.manager_id = m.employee_id;
```

Which JOIN option can be used in the blank in the above query to get the required output?

Exhibit:

- A. only inner JOIN
- B. only FULL OUTER JOIN
- C. only LEFT OUTER JOIN
- D. only RIGHT OUTER JOIN

**Answer:** C

**NEW QUESTION 275**

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

In the CUSTOMERS table, the CUST\_LAST\_NAME column contains the values 'Anderson' and 'Ausson'.

You issue the following query:

```
SQL> SELECT LOWER(REPLACE(TRIM('son' FROM cust_last_name),'An','O'))
```

```
FROM CUSTOMERS
```

```
WHERE LOWER(cust_last_name) LIKE 'a%n';
```

What would be the outcome?

- A. 'Oder' and 'Aus'
- B. an error because the TRIM function specified is not valid
- C. an error because the LOWER function specified is not valid
- D. an error because the REPLACE function specified is not valid

**Answer:** B

**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 The TRIM Function The TRIM function removes characters from the beginning or end of character literals, columns or expressions to yield one potentially shorter character item. Numeric and date literals are automatically cast as characters when they occur as parameters to the TRIM function. Numeric or date expressions are evaluated first before being converted to strings ready to be trimmed. The TRIM function takes a parameter made up of an optional and a mandatory component. Its syntax is TRIM ([trailing|leading|both] trimstring from s). The string to be trimmed (s) is mandatory. The following points list the rules governing the use of this function:

TRIM(s) removes spaces from both sides of the input string.

TRIM(trailing trimstring from s) removes all occurrences of trimstring from the end of the string s if it is present. TRIM(leading trimstring from s) removes all occurrences of trimstring from the beginning of the string s if it is present.

TRIM(both trimstring from s) removes all occurrences of trimstring from the beginning and end of the string s if it is present.

The following queries illustrate the usage of this function:

Query 1: select trim(trailing 'e' from 1+2.14||' is pie') from dual

Query 2: select trim(both '\*' from '\*\*\*\*\*Hidden\*\*\*\*\*') from dual

Query 3: select trim(1 from sysdate) from dual

ORA-30001: trim set should have only one character

30001. 00000 - "trim set should have only one character"

\*Cause: Trim set contains more or less than 1 character. This is not allowed in TRIM

function.  
REPLACE(text, search\_string, replacement\_string)  
Searches a text expression for a character string and, if found, replaces it with a specified replacement string

**NEW QUESTION 279**

In the CUSTOMERS table, the CUST\_CITY column contains the value 'Paris' for the CUST\_FIRST\_NAME 'ABIGAIL'.

Evaluate the following query:

```
SQL> SELECT INITCAP(cust_first_name || ' ' ||  
UPPER(SUBSTR(cust_city,-LENGTH(cust_city),2)))  
FROM customers  
WHERE cust_first_name = 'ABIGAIL';
```

What would be the outcome?

- A. Abigail PA
- B. Abigail Pa
- C. Abigail IS
- D. an error message

**Answer:** B

**NEW QUESTION 281**

Examine the structure of the EMPLOYEES and DEPARTMENTS tables:

You want to create a report displaying employee last names, department names, and locations. Which query should you use to create an equi-join?

- A. SELECT last\_name, department\_name, location\_id FROM employees , departments ;
- B. SELECT employees.last\_name, departments.department\_name, departments.location\_id FROM employees e, departments D WHERE e.department\_id =d.department\_id;
- C. SELECT e.last\_name, d.department\_name, d.location\_id FROM employees e, departments D WHERE manager\_id =manager\_id;
- D. SELECT e.last\_name, d.department\_name, d.location\_id FROM employees e, departments D WHERE e.department\_id =d.department\_id;

**Answer:** D

**Explanation:**

Equijoins are also called simple joins or inner joins. Equijoin involve primary key and foreign key.

Incorrect Answer: A there is no join B invalid syntax C does not involve the join in the primary and foreign key

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 4-8

**NEW QUESTION 282**

Examine the structure of the PRODUCTS table:

Name	Null	Type
PROD_ID	NOT NULL	NUMBER(4)
PROD_NAME		VARCHAR2(20)
PROD_STATUS		VARCHAR2(6)
QTY_IN_HAND		NUMBER(8,2)
UNIT_PRICE		NUMBER(10,2)

You want to display the names of the products that have the highest total value for UNIT\_PRICE \*QTY\_IN\_HAND.

Which SQL statement gives the required output?

- A. SELECT prod\_name FROM products WHERE (unit\_price \* qty\_in\_hand) = (SELECT MAX(unit\_price \* qty\_in\_hand) FROM products);
- B. SELECT prod\_name FROM products WHERE (unit\_price \* qty\_in\_hand) = (SELECT MAX(unit\_price \* qty\_in\_hand) FROM products GROUP BY prod\_name);
- C. SELECT prod\_name FROM products GROUP BY prod\_name HAVING MAX(unit\_price \* qty\_in\_hand) = (SELECT MAX(unit\_price \* qty\_in\_hand) FROM products GROUP BY prod\_name);
- D. SELECT prod\_name FROM products WHERE (unit\_price \* qty\_in\_hand) = (SELECT MAX(SUM(unit\_price \* qty\_in\_hand)) FROM products) GROUP BY prod\_name;

**Answer:** A

**NEW QUESTION 285**

Which describes the default behavior when you create a table?

- A. The table is accessible to all user
- B. Tables are created in the public schem
- C. Tables are created in your schem
- D. Tables are created in the DBA schem
- E. You must specify the schema when the table is create

**Answer:** C

**Explanation:**

sorted by highest to lowest is DESCENDING order

Incorrect Answer: Agrant the table privilege to PUBLIC Blogin as sysoper Dlogin as DBA or sysdba Eno such option is allow.

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 9-9

#### NEW QUESTION 289

View the Exhibit and examine the structure and data in the INVOICE table.

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

Which statements are true regarding data type conversion in expressions used in queries? (Choose all that apply.)

- A. inv\_amt ='0255982' : requires explicit conversion
- B. inv\_date > '01-02-2008' : uses implicit conversion
- C. CONCAT(inv\_amt,inv\_date) : requires explicit conversion
- D. inv\_date = '15-february-2008' : uses implicit conversion
- E. inv\_no BETWEEN '101' AND '110' : uses implicit conversion

**Answer:** DE

#### Explanation:

In some cases, the Oracle server receives data of one data type where it expects data of a different data type. When this happens, the Oracle server can automatically convert the data to the expected data type. This data type conversion can be done implicitly by the Oracle server or explicitly by the user. Explicit data type conversions are performed by using the conversion functions. Conversion functions convert a value from one data type to another. Generally, the form of the function names follows the convention data type TO data type. The first data type is the input data type and the second data type is the output. Note: Although implicit data type conversion is available, it is recommended that you do the explicit data type conversion to ensure the reliability of your SQL statements.

#### NEW QUESTION 290

The CUSTOMERS table has these columns:

CUSTOMER_ID	NUMBER(4)	NOT NULL
CUSTOMER_NAME	VARCHAR2(100)	NOT NULL
STREET_ADDRESS	VARCHAR2(150)	
CITY_ADDRESS	VARHCAR2(50)	
STATE_ADDRESS	VARCHAR2(50)	
PROVINCE_ADDRESS	VARCHAR2(50)	
COUNTRY_ADDRESS	VARCHAR2(50)	
POSTAL_CODE	VARCHAR2(12)	
CUSTOMER_PHONE	VARCHAR2(20)	

The CUSTOMER\_ID column is the primary key for the table.

You need to determine how dispersed your customer base is.

Which expression finds the number of different countries represented in the CUSTOMERS table?

- A. COUNT(UPPER(country\_address))
- B. COUNT(DIFF(UPPER(country\_address)))
- C. COUNT(UNIQUE(UPPER(country\_address)))
- D. COUNT DISTINCT UPPER(country\_address)
- E. COUNT(DISTINCT (UPPER(country\_address)))

**Answer:** E

#### NEW QUESTION 295

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

You want to generate a report showing the last names and credit limits of all customers

whose last names start with A, B, or C, and credit limit is below 10,000.

Evaluate the following two queries:

```
SQL> SELECT cust_last_name, cust_credit_limit FROM customers
WHERE (UPPER(cust_last_name) LIKE 'A%' OR
UPPER(cust_last_name) LIKE 'B%' OR UPPER(cust_last_name) LIKE 'C%')
AND cust_credit_limit < 10000;
SQL> SELECT cust_last_name, cust_credit_limit FROM customers
WHERE UPPER(cust_last_name) BETWEEN 'A' AND 'C'
AND cust_credit_limit < 10000;
```

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

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)

- A. Only the first query gives the correct resul
- B. Only the second query gives the correct resul
- C. Both execute successfully and give the same resul
- D. Both execute successfully but do not give the required resul

**Answer:** A

#### NEW QUESTION 299

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

- A. Views can be created only from table
- B. Views can be created from tables or other view
- C. Only simple views can use indexes existing on the underlying table
- D. Both simple and complex views can use indexes existing on the underlying table
- E. Complex views can be created only on multiple tables that exist in the same schem
- F. Complex views can be created on multiple tables that exist in the same or different schema

**Answer:** BDF

#### Explanation:

Creating a Sequence (continued)

CYCLE | NOCYCLE Specifies whether the sequence continues to generate values after reaching its maximum or minimum value

(NOCYCLE is the default option.)

CACHE n | NOCACHE Specifies how many values the Oracle server preallocates and keeps in memory (By default, the Oracle server caches 20 values.)

#### NEW QUESTION 300

View the Exhibit and examine the structure and data in the INVOICE table.

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

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

**Answer:** BD

#### Explanation:

Using the AVG and SUM Functions You can use the AVG, SUM, MIN, and MAX functions against the columns that can store numeric data. The example in the slide displays the average, highest, lowest, and sum of monthly salaries for all sales representatives Using the MIN and MAX Functions You can use the MAX and MIN functions for numeric, character, and date data types. The example in the slide displays the most junior and most senior employees.

#### NEW QUESTION 302

Which statement describes the ROWID data type?

- A. Binary data up to 4 gigabyte
- B. Character data up to 4 gigabyte
- C. Raw binary data of variable length up to 2 gigabyte
- D. Binary data stored in an external file, up to 4 gigabyte
- E. A hexadecimal string representing the unique address of a row in its tabl

**Answer:** E

#### Explanation:

The ROWID datatype stores information related to the disk location of table rows. They also uniquely identify the rows in your table. The ROWID datatype is stored as a hexadecimal string.

Incorrect Answers

A:It is not a binary data. The ROWID datatype is a hexadecimal string.

B:It is not a character data. The ROWID datatype is a hexadecimal string.

C:It is not a raw binary data. The ROWID datatype is a hexadecimal string.

D:It is not binary data stored in an external file. The ROWID datatype is a hexadecimal

string.  
OCP Introduction to Oracle 9i: SQL Exam Guide, Jason Couchman, p. 216  
Chapter 5: Creating Oracle Database Objects

**NEW QUESTION 304**

Examine the description of the CUSTOMERS table:

CUSTOMER_ID	NUMBER(4)	NOT NULL
CUSTOMER_NAME	VARCHAR2(100)	NOT NULL
STREET_ADDRESS	VARCHAR2(150)	
CITY_ADDRESS	VARCHAR2(50)	
STATE_ADDRESS	VARCHAR2(50)	
PROVINCE_ADDRESS	VARCHAR2(50)	
COUNTRY_ADDRESS	VARCHAR2(50)	
POSTAL_CODE	VARCHAR2(12)	
CUSTOMER_PHONE	VARCHAR2(20)	

The CUSTOMER\_ID column is the primary key for the table.  
Which statement returns the city address and the number of customers in the cities Los Angeles or San Francisco?

- A. SELECT city\_address, COUNT(\*) FROM customers WHERE city\_address IN ( 'Los Angeles', 'San Fransisco');
- B. SELECT city\_address, COUNT (\*) FROMcustomers WHERE city address IN ( 'Los Angeles', 'San Fransisco') GROUP BY city\_address;
- C. SELECT city\_address, COUNT(customer\_id) FROMcustomers WHERE city\_address IN ( 'Los Angeles', 'San Fransisco') GROUP BYcity\_address, customer\_id;
- D. SELECT city\_address, COUNT (customer\_id) FROM customers GROUP BY city\_address IN ( 'Los Angeles', 'San Fransisco');

**Answer:** B

**Explanation:**

Not C: The customer ID in the GROUP BY clause is wrong

**NEW QUESTION 308**

Examine this statement:

```
SELECT student_id, gpa FROM student_grades WHERE gpa > &&value;
```

You run the statement once, and when prompted you enter a value of 2.0. A report is produced. What happens when you run the statement a second time?

- A. An error is returne
- B. You are prompted to enter a new valu
- C. A report is produced that matches the first report produce
- D. You are asked whether you want a new value or if you want to run the report based on the previous valu

**Answer:** C

**Explanation:**

use the double-ampersand if you want to reuse the variable value without prompting the user each time.

Incorrect Answer: A is not an error

B&& will not prompt user for second time D&& will not ask the user for new value

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 7-13

**NEW QUESTION 311**

What is true about sequences?

- A. Once created, a sequence belongs to a specific schem
- B. Once created, a sequence is linked to a specific tabl
- C. Once created, a sequence is automatically available to all user
- D. Only the DBA can control which sequence is used by a certain tabl
- E. Once created, a sequence is automatically used in all INSERT and UPDATE statement

**Answer:** A

**NEW QUESTION 315**

View the Exhibit and examine the structure of the SALES and PRODUCTS tables.

#### SALES

Name	Null?	Type
PROD_ID	NOT NULL	NUMBER (3)
CUST_ID	NOT NULL	NUMBER (4)
TIME_ID		DATE
QTY_SOLD		NUMBER (10,2)

#### PRODUCTS

Name	Null?	Type
PROD_ID	NOT NULL	NUMBER (3)
PROD_NAME		VARCHAR2 (30)
PROD_LIST_PRICE		NUMBER (8,2)

In the SALES table, PROD\_ID is the foreign key referencing PROD\_ID in the PRODUCTS table. You want to list each product ID and the number of times it has been sold.

Evaluate the following query:

```
SQL>SELECT p.prod_id, COUNT(s.prod_id)
FROM products p _____ sales s
ON p.prod_id = s.prod_id
GROUP BY p.prod_id;
```

Which two JOIN options can be used in the blank in the above query to get the required output? (Choose two.)

- A. JOIN
- B. FULL OUTER JOIN
- C. LEFT OUTER JOIN
- D. RIGHT OUTER JOIN

**Answer:** BC

#### NEW QUESTION 319

What does the FORCE option for creating a view do?

- A. creates a view with constraints
- B. creates a view even if the underlying parent table has constraints
- C. creates a view in another schema even if you don't have privileges
- D. creates a view regardless of whether or not the base tables exist

**Answer:** D

#### Explanation:

create a view regardless of whether or not the base tables exist.

Incorrect Answer: Athe option is not valid Bthe option is not valid Cthe option is not valid

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 11-3

#### NEW QUESTION 324

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: AObject privilege on VIEW is DELETE, INSERT, REFERENCES, SELECT and UPDATE BObject privilege on VIEW is DELETE, INSERT, REFERENCES, SELECT and UPDATE CObject privilege on VIEW is DELETE, INSERT, REFERENCES, SELECT and UPDATE

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 13-12

#### NEW QUESTION 327

Examine the data in the CUST\_NAME column of the CUSTOMERS table. CUST\_NAME

Lex De Haan Renske Ladwig Jose Manuel Urman

Jason Mallin

You want to extract only those customer names that have three names and display the \* symbol in place of the first name as follows:

CUST NAME

\*\*\* De Haan

\*\*\*\* Manuel Urman

Which two queries give the required output? (Choose two.)

- A. SELECT LPAD(SUBSTR(cust\_name, INSTR(cust\_name, ' '), LENGTH(cust\_name), '\*')) "CUST NAME" FROM customers WHERE INSTR(cust\_name, ' ', 1, 2) <> 0;
- B. SELECT LPAD(SUBSTR(cust\_name, INSTR(cust\_name, ' '), LENGTH(cust\_name), '\*')) "CUST NAME" FROM customers WHERE INSTR(cust\_name, ' ', -1, 2) <> 0;
- C. SELECT LPAD(SUBSTR(cust\_name, INSTR(cust\_name, ' '), LENGTH(cust\_name) - INSTR(cust\_name, ' '), '\*')) "CUST NAME" FROM customers WHERE INSTR(cust\_name, ' ', -1, -2) <> 0;

D. SELECT LPAD(SUBSTR(cust\_name,INSTR(cust\_name,' ')),LENGTH(cust\_name)-INSTR(cust\_name,' '),\*) "CUST NAME" FROM customers WHERE INSTR(cust\_name, ' ',1,2)<>0 ;

**Answer:** AB

#### NEW QUESTION 332

Evaluate the following CREATE SEQUENCE statement:

```
CREATE SEQUENCE seq1
```

```
START WITH 100
```

```
INCREMENT BY 10
```

```
MAXVALUE 200
```

```
CYCLE
```

```
NOCACHE;
```

The SEQ1 sequence has generated numbers up to the maximum limit of 200. You issue the following SQL statement:

```
SELECT seq1.nextval FROM dual;
```

What is displayed by the SELECT statement?

- A. 1
- B. 10
- C. 100
- D. an error

**Answer:** A

#### Explanation:

But why the answer is not "C" ? Because you didn't specify the MINVALUE for the sequence. If you check the sequence definition that you created it will have the default value of 1, which it reverts to when cycling. If you wanted to keep the minimum value you would need to specify it in the sequence creation. sequence Is the name of the sequence generator INCREMENT BY n Specifies the interval between sequence numbers, where n is an integer (If this clause is omitted, the sequence increments by 1.) START WITH n Specifies the first sequence number to be generated (If this clause is omitted, the sequence starts with 1.)

MAXVALUE n Specifies the maximum value the sequence can generate NOMAXVALUE Specifies a maximum value of 10<sup>27</sup> for an ascending sequence and –1 for a descending sequence (This is the default option.) MINVALUE n Specifies the minimum sequence value NOMINVALUE Specifies a minimum value of 1 for an ascending sequence and –(10<sup>26</sup>) for a descending sequence (This is the default option.)

CYCLE | NOCYCLE Specifies whether the sequence continues to generate values after reaching its maximum or minimum value (NOCYCLE is the default option.)

CACHE n | NOCACHE Specifies how many values the Oracle server preallocates and keeps in memory (By default, the Oracle server caches 20 values.)

#### NEW QUESTION 337

Which is an iSQL\*Plus command?

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

**Answer:** D

**Explanation:** Explanation: The only SQL\*Plus command in this list : DESCRIBE. It cannot be used as SQL command. This command returns a description of table name, 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: A INSERT is not a SQL\*PLUS command B UPDATE is not a SQL\*PLUS command C SELECT is not a SQL\*PLUS command E DELETE is not a SQL\*PLUS command F RENAME is not a SQL\*PLUS command

Refer: Introduction to Oracle9i: SQL, Oracle University Study Guide, 7

#### NEW QUESTION 342

.....

## Thank You for Trying Our Product

\* 100% Pass or Money Back

All our products come with a 90-day Money Back Guarantee.

\* One year free update

You can enjoy free update one year. 24x7 online support.

\* Trusted by Millions

We currently serve more than 30,000,000 customers.

\* Shop Securely

All transactions are protected by VeriSign!

**100% Pass Your 1Z0-051 Exam with Our Prep Materials Via below:**

<https://www.certleader.com/1Z0-051-dumps.html>