

## QUESTION: 1

Name	Null Type
ORDER_ID	NUMBER(38)
PRODUCT_ID	NUMBER(38)
QUANTITY	NUMBER(38)
UNIT_PRICE	NUMBER(10,2)

Examine the description of the ORDER\_ITEMS table: Examine this incomplete query:SELECT DISTINCT quantity \* unit\_price total\_paid FROM order\_items ORDER BY ; Which two can replace so the query completes successfully?

- Option A : quantity
- Option B : quantity, unit\_price
- Option C : total\_paid
- Option D : product\_id
- Option E : quantity \* unit\_price

Correct Answer: B,E

### Explanation/Reference:

In a SELECT statement with DISTINCT, the ORDER BY clause can only order by expressions that are part of the SELECT list.

- A. quantity alone is not sufficient to replace as it is not included in the SELECT list after DISTINCT.
- B. This option can successfully replace because both quantity and unit\_price are used in the SELECT expression, and thus their individual values are valid for the ORDER BY clause.
- C. total\_paid is an alias for the expression quantity \* unit\_price, but it cannot be used in the ORDER BY clause because Oracle does not allow aliases of expressions in DISTINCT queries to be used in ORDER BY.
- D. product\_id is not included in the SELECT list after DISTINCT and thus cannot be used in ORDER BY.
- E. The expression quantity \* unit\_price is exactly what is selected, so it can replace and the query will complete successfully.

### References:

Oracle Database SQL Language Reference, 12c Release 1 (12.1): "ORDER BY Clause"

## QUESTION: 2

Which three statements are true about an ORDER BY clause?

- Option A : An ORDER BY clause always sorts NULL values last.
- Option B : An ORDER BY clause can perform a binary sort
- Option C : An ORDER BY clause can perform a linguistic sort
- Option D : By default an ORDERBY clause sorts rows in ascending order
- Option E : An ORDR BY clause will always precede a HAVING clause if both are used in the same top-level

Correct Answer: B,C,D

### Explanation/Reference:

In Oracle Database 12c, the behavior of the ORDER BY clause is guided by several rules:

Option B: An ORDER BY clause can perform a binary sort.

A binary sort is the default sorting mechanism in Oracle Database, which is based on the binary representation of the data.

Option C: An ORDER BY clause can perform a linguistic sort.

Oracle supports linguistic sorting through the use of the NLS\_SORT parameter, which can be set to various linguistic and cultural norms.

Option D: By default, an ORDER BY clause sorts rows in ascending order.

If no ASC or DESC keyword is specified, Oracle will sort the results in ascending order by default.

Options A and E are incorrect:

Option A is not universally true; the position of NULL values in the sort order can be controlled by settings or specific SQL syntax (NULLS FIRST or NULLS LAST).

Option E is incorrect as the HAVING clause filters groups after data has been grouped by the GROUP BY clause and cannot be logically placed before ORDER BY in processing order.

### QUESTION: 3

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

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

Correct Answer: A,C,D

Explanation/Reference:

Explanation: <http://docs.oracle.com/javadb/10.6.2.1/ref/rrefsqj13658.html>

### QUESTION: 4

Which statement is true regarding the default behavior of the ORDER BY clause?

- Option A : In a character sort, the values are case-sensitive.
- Option B : NULLs are not included in the sort operation
- Option C : Only columns that are specified in the SELECT list can be used in the ORDER BY clause
- Option D : Numeric values are displayed in descending order if they have decimal positions.
- Option E : Column aliases can be used in the ORDER BY clause.

Correct Answer: A

Explanation/Reference:

Explanation:

The ORDER BY clause performs case sensitive sorting with character values.

### QUESTION: 5

Examine this SQL statement:

```
DELETE FROM employees e
WHERE EXISTS
  (SELECT 'dummy'
   FROM emp_history
   WHERE employee_id = e.employee_id);
```

Which two are true?

- Option A : The subquery is not a correlated subquery.
- Option B : The subquery is executed before the DELETE statement is executed
- Option C : The DELETE statement executes successfully even if the subquery selects multiple rows
- Option D : The subquery is executed for every row in the EMPLOYEES table.
- Option E : All existing rows in the EMPLOYEES table are deleted

Correct Answer: B,D

## QUESTION: 6

Examine the data in the CUST NAME column of the CUSTOMERS table: CUST\_NAME-----  
 Renske Ladwig Jason Mallin Samuel McCain Allan MCEwen Irene Mikkilineni Julia Nayer  
 You want to display the CUST\_NAME values where the last name starts with Mc or MC. Which two WHERE clauses give the required result?

Option A :

WHERE INITCAP (SUBSTR(cust\_name, INSTR(cust\_name, '') + 1)) IN ('MC%', 'Mc%')

Option B :

WHERE UPPER (SUBSTR(cust\_name, INSTR(cust\_name, '') + 1)) LIKE UPPER('MC%')

Option C :

WHERE INITCAP(SUBSTR(cust\_name, INSTR(cust\_name, '') + 1)) LIKE 'Mc%'

Option D :

WHERE SUBSTR(cust\_name, INSTR(cust\_name, '') + 1) LIKE 'Mc%' OR 'MC%'

Option E :

WHERE SUBSTR(cust\_name, INSTR(cust\_name, '') + 1) LIKE 'Mc%'

Correct Answer: B,C

### Explanation/Reference:

To find customers whose last names start with "Mc" or "MC", we need to ensure our SQL query correctly

identifies and compares these prefixes regardless of case variations. Let's analyze the given options:

Option B: WHERE UPPER(SUBSTR(cust\_name, INSTR(cust\_name, ' ') + 1)) LIKE UPPER('

MC%') This clause uses UPPER to convert both the extracted substring (starting just after the first space, assuming it indicates the start of the last name) and the comparison string 'MC%' to uppercase. This ensures case-insensitive comparison. The LIKE operator is used to match any last names starting with "MC", which will correctly capture both "Mc" and "MC". This option is correct.

Option C: WHERE INITCAP(SUBSTR(cust\_name, INSTR(cust\_name, ' ') + 1)) LIKE 'Mc%'

This clause applies INITCAP to the substring, which capitalizes the first letter of each word and makes other letters lowercase. The result is compared to 'Mc%', assuming only the last name follows the space. This approach will match last names starting with "Mc" (like "McEwen"), but not "MC". However, considering we're looking for "Mc" specifically, this clause works under the assumption that "Mc" is treated as proper capitalization for these last names. Thus, it can also be considered correct, albeit less inclusive than option B.

The other options either use incorrect syntax or apply case-sensitive matches without ensuring that both "Mc" and "MC" are captured:

Option A: Contains syntax errors (unmatched quotes and wrong use of IN).

Option D: Uses case-sensitive match without combining both "Mc" and "MC".

Option E: Only matches "Mc", which is too specific.

## QUESTION: 7

Which two statements are true about \*\_TABLES views?

Option A : You must have SELECT privileges on a table to view it in ALL \_TABLES.

Option B : You must have SELECT privileges on a table to view it in DBA TABLES.

Option C : USER\_ TABLES displays all tables owned by the current user.

Option D : ALL TABLES displays all tables owned by the current user.

Option E : You must have SELECT privileges on a table to view it in USER TABLES.

Option F : All users can query DBA TABLES successfully.

Correct Answer: C,F

## Explanation/Reference:

Understanding Oracle's data dictionary views is crucial for database management:

Option C: USER\_TABLES displays all tables owned by the current user.

USER\_TABLES is a view that shows all tables owned by the user executing the query.

Option F: All users can query DBA\_TABLES successfully.

While all users can attempt to query DBA\_TABLES, only users with the appropriate privileges

can retrieve information from it. This option is tricky; "querying" is possible for all, but

"retrieving data" depends on privileges.

Options A, B, D, and E are not entirely correct:

Option A and Option E are incorrect because USER\_TABLES shows tables regardless of SELECT privileges.

Option B is incorrect; one does not need SELECT privileges on each table to view them in

DBA\_TABLES, but they need appropriate system privileges to access DBA\_TABLES.

Option D is incorrect as ALL\_TABLES displays tables accessible to the current user, not just owned by them.

## QUESTION: 8

Which three statements about roles are true?

Option A : Roles are assigned to roles using the ALTER ROLE Statement

Option B : A role is a named group of related privileges that can only be assigned to a user

Option C : Roles are assigned to users using the ALTER USER statement

Option D : A single role can be assigned to multiple users.

Option E : A single user can be assigned multiple roles

Option F : Privileges are assigned to a role using the ALTER ROLE statement.

Option G : Privileges are assigned to a role using the GRANT statement.

Correct Answer: A,D

### QUESTION: 9

Examine the description of the SALES1 table:

Name	Null	Type
SALES_ID	NOT NULL	NUMBER
STORE_ID	NOT NULL	NUMBER
ITEMS_ID		NUMBER
QUANTITY		NUMBER
SALES_DATE		DATE

SALES2 is a table with the same description as SALES1. Some sales data is duplicated in both tables. You want to display the rows from the SALES1 table which are not present in the SALES2 table. Which set operator generates the required output?

- Option A : SUBTRACT
- Option B : INTERSECT
- Option C : UNION ALL
- Option D : UNION
- Option E : MINUS

Correct Answer: A

### QUESTION: 10

Which three statements are true about the DESCRIBE command? (Choose three.)

Option A :

It can be used from SQL Developer.

Option B :

It can be used to display the structure of an existing view.

Option C :

It can be used only from SQL\*Plus.

Option D :

It displays the NOT NULL constraint for any columns that have that constraint.

Option E :

It displays all constraints that are defined for each column.

Option F :

It displays the PRIMARY KEY constraint for any column or columns that have that constraint.

Correct Answer: A,D,F

Explanation/Reference:

A: True. The DESCRIBE command can indeed be used from SQL Developer as well as other Oracle database tools such as SQL\*Plus. This command is used to display the structure of a table, view, or other object, showing information such as column names, data types, and whether a column is nullable.

B: True. The DESCRIBE command can be used to display the structure of an existing view, showing similar information as it would for a table. This includes the columns, their data types, and other pertinent details.

D: True. When DESCRIBE is used, it does display the NOT NULL constraints for columns that have this constraint. This is part of the basic information about the structure of a table or view that can help developers understand the requirements of the data stored therein.

## QUESTION: 11

Which statement will return a comma-separated list of employee names in alphabetical order for each department in the EMP table?

Option A :

```
SELECT deptno,LISTAGG(ename, ' , ') WITHIN GROUP AS employee_list FROM emp GROUP BY deptno;
```

Option B :

```
SELECT deptno,LISTAGG(ename, ' , ') WITHIN GROUP AS employee_list FROM emp GROUP BY deptno  
ORDER BY ename;
```

Option C :

```
SELECT deptno,LISTAGG(ename, ' , ') WITHIN GROUP (GROUP BY deptno) AS employee_list FROM emp  
ORDER BY ename;
```

Option D :

```
SELECT deptno,LISTAGG(ename, ' , ') WITHIN GROUP (ORDER BY ename) AS employee_list FROM emp  
GROUP BY deptno;
```

Correct Answer: D

Explanation/Reference:

The LISTAGG function is used in Oracle to aggregate strings from data in a group specified by the GROUP BY clause, producing a single row of concatenated values. The correct syntax also specifies an ORDER BY clause within the WITHIN GROUP parenthesis to sort the values in the concatenated list.

The correct query is:

```
SELECT deptno, LISTAGG(ename, ', ') WITHIN GROUP (ORDER BY ename) AS employee_list FROM  
emp GROUP BY deptno;
```

This statement will return a comma-separated list of employee names (ename) in alphabetical order for each department (deptno) in the EMP table.

## QUESTION: 12

Which statement is true regarding the INTERSECT operator?

Option A : The names of columns in all SELECT statements must be identical.

Option B : It ignores NULLs.

Option C : Reversing the order of the intersected tables alters the result.

Option D : The number of columns and data types must be identical for all SELECT statements in the query.

Correct Answer: D

Explanation/Reference:

INTERSECT Returns only the rows that occur in both queries' result sets, sorting them and removing duplicates.

The columns in the queries that make up a compound query can have different names, but the output result set will use the names of the columns in the first query.

### QUESTION: 13

ORD_ID	ITEN_NO	QTY
-----	-----	-----
1	111	10
1	222	20
1	333	30
2	333	30
2	444	40
3	111	40

Examine the data in the ORD\_ITEMS table:  
Evaluate this query:

```
SQL>SELECT item_no, AVG(qty)
FROM ord_items
HAVING AVG(qty) > MIN(qty) * 2
GROUP BY item_no;
```

Which statement is true regarding the result?

- Option A : It returns an error because the HAVING clause should be specified after the GROUP BY clause.
- Option B : It returns an error because all the aggregate functions used in the HAVING clause must be specified in the SELECT list.
- Option C : It displays the item nos with their average quantity where the average quantity is more than double the minimum quantity of that item in the table.
- Option D : It displays the item nos with their average quantity where the average quantity is more than double the overall minimum quantity of all the items in the table.

Correct Answer: C

### QUESTION: 14

Examine the structure proposed for the TRANSACTIONS table:

Name	Null?	Type
-----	-----	-----
TRANS_ID	NOT NULL	NUMBER (6)
CUST_NAME	NOT NULL	VARCHAR2 (20)
CUST_STATUS	NOT NULL	CHAR
TRANS_DATE	NOT NULL	DATE
TRANS_VALIDITY		VARCHAR2
CUST_CREDIT_LIMIT		NUMBER

Which two statements are true regarding the creation and storage of data in the above tablestructure? (Choose two.)

- Option A : The CUST\_STATUS column would store exactly one character
- Option B : The TRANS\_VALIDITY column would have a maximum size of one character.
- Option C : The CUST\_CREDIT\_LIMIT column would be able to store decimal values.
- Option D : The CUST\_STATUS column would give an error.

Option E :

The TRANS\_DATE column would be able to store day, month, century, year, hour, minutes, seconds, and fractions of seconds.

Option F : The TRANS\_VALIDITY column would give an error.

Correct Answer: A,C

## QUESTION: 15

Examine the description of the PRODUCTS table which contains data:

Name	Null?	Type
PROD ID	NOT NULL	NUMBER (2)
PROD NAME		VARCHAR2 (20)
EXPIRYDATE	NOT NULL	DATE

Which two are true?

Option A : The PROD ID column can be renamed.

Option B : The PROD\_ID column data type can be changed to VARCHAR2 (2).

Option C : The EXPIRY DATE column data type can be changed to TIME STAMP.

Option D : The EXPIRY DATE column cannot be dropped.

Option E : The PROD NAME column cannot have a DEFAULT clause added to it.

Correct Answer: A,C

Explanation/Reference:

A: True, the name of a column can be changed in Oracle using the ALTER TABLE ... RENAME

COLUMN command.

B: False, you cannot change a column's data type from NUMBER to VARCHAR2 if the table contains data, unless the change does not result in data loss or inconsistency.

C: True, it is possible to change a DATE data type column to TIMESTAMP because TIMESTAMP is an extension of DATE that includes fractional seconds. This operation is allowed if there is no data loss.

D: False, any column that is not part of a primary key or does not have a non-deferrable constraint can generally be dropped unless it contains data that does not allow for such a change.

E: False, the DEFAULT clause can be added to a column provided there is no data that contradicts the

default value or it doesn't violate any constraints.

These statements are verified against the Oracle Database 12c SQL documentation, specifically the sections on data types, the ALTER TABLE command, and the use of literals in SQL expressions.

### QUESTION: 16

Examine the data in the NEW\_EMPLOYEES table:

EMPLOYEE_ID	NAME	DEPARTMENT_ID	MANAGER_ID	JOB_ID	SALARY
101	David	20	120	SA_REP	14000
102	Sam	10	105	CLERK	12500
103	Andrew	20	120	FIN_ADMIN	14200
104	Adrian	30	108	MAR_CLERK	12500
105	Maria	30	108	FIN_ADMIN	15000
106	Tracy	40	110	AD_ASST	13000
108	Kate	30	110	FIN_DIR	16500
110	Anne	40	120	EX_DIR	18000
120	Fran	20	110	SQ_DIR	16500

Examine the data in the EMPLOYEES table:

EMPLOYEE_ID	NAME	JOB_ID	SALARY
101	David	CLERK	14000
102	Sam	SA_REP	11500
104	Adrian	MAR_CLERK	12500
108	Kate	FIN_DIR	16500
110	Annie	EX_DIR	18000

Update existing employee details in the EMPLOYEES table with data from the NEW\_EMPLOYEES table. Add new employee details from the NEW\_EMPLOYEES table to the EMPLOYEES table. Which statement will do this?

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

Correct Answer: C

### QUESTION: 17

Which three statements are true about views in an Oracle database?

Option A :

A SELECT statement cannot contain a where clause when querying a view containing a WHERE clause in its defining query

Option B :

Rows inserted into a table using a view are retained in the table if the view is dropped

Option C :

Views can join tables only if they belong to the same schema.

Option D :

Views have no segment.

Option E :

Views have no object number.

Option F :

A view can be created that refers to a non-existent table in its defining query.

Option G :

Inserting one or more rows using a view whose defining query contains a GROUP BY clause will cause an error.

Correct Answer: B,D,F,G

Explanation/Reference:

A view is a virtual table based on a SQL query.

A. This is incorrect because a SELECT statement querying a view can contain a WHERE clause, regardless of the view's defining query. C. This is incorrect because views can join tables from different schemas, not just the same schema. B. Correct. The rows inserted into a base table via a view remain in the table even if the view is dropped because the view is just a window to the data in the base tables. D. Correct. Views do not require storage space other than for the definition of the view in the data dictionary, hence they have no segment. E. Incorrect. Views do not have object numbers because they are not database objects that occupy physical space. F. Correct. You can create a view that references non-existent tables; such a view would be considered invalid until the base table is created.

The Oracle Database Concepts guide provides information about views and their characteristics.

## QUESTION: 18

Examine this query and its output:

```
SELECT * FROM products;

PROD_ID  PROD_NAME  PROD_LIST
-----  -
101     Plate      10
102     Cup        20
103     Saucer     30
104     Knife     40
105     Fork
```

Examine this query with an incomplete

was WHERE clause:

```
SELECT prod_name
FROM products
WHERE prod_list <operator> (SELECT prod_list FROM products);
```

Which two are true about operators that can be used in the WHERE clause?

- Option A : Using not in or <> any will give the same result.
- Option B : Using in will display all the product names
- Option C : Using not in or <> all will give the same result
- Option D : Using <> any will display all the product names except the product named Fork
- Option E : Using <> any will display all the product names.

Correct Answer: C,D

## QUESTION: 19

Examine the commands used to create DEPARTMENT\_DETAILS and COURSE\_DETAILS tables:

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

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

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

- Option A : SELECT d.department\_id, c.course\_id FROM department\_details d RIGHT OUTER JOIN course\_details c ON (d.department\_id=c.department\_id);
- Option B : SELECT d.department\_id, c.course\_id FROM department\_details d LEFT OUTER JOIN course\_details c ON (d.department\_id=c.department\_id);
- Option C : SELECT d.department\_id, c.course\_id FROM course\_details c LEFT OUTER JOIN department\_details d ON (c.department\_id=d.department\_id);

Option D : SELECT d.department\_id, c.course\_id FROM department\_details d RIGHT OUTER JOIN course\_details c ON (c.department\_id=d.department\_id);

Correct Answer: B

### QUESTION: 20

Which two statements are true regarding the SQL GROUP BY clause? (Choose two.)

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

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

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

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

Option E : If the SELECT clause has an aggregating function, then columns without an aggregating function in the SELECT clause should be included in the GROUP BY clause.

Correct Answer: A,D

### QUESTION: 21

Which two statements are true about single row functions?

Option A : CONCAT: can be used to combine any number of values

Option B : MOD: returns the quotient of a division operation

Option C : CEIL: can be used for positive and negative numbers

Option D : FLOOR: returns the smallest integer greater than or equal to a specified number

Option E : TRUNC: can be used with NUMBER and DATE values

Correct Answer: C,E

Explanation/Reference:

Single-row functions operate on single rows and return one result per row. Let's look at each option in the context of Oracle 12c SQL:

A. CONCAT: This function can only combine two values at a time. If you need to concatenate more than two values, you have to nest CONCAT functions or use the || operator.

B. MOD: The MOD function returns the remainder of a division operation, not the quotient.

C. CEIL: This function returns the smallest integer that is greater than or equal to a specified number. It

works with both positive and negative numbers.

D. FLOOR: It returns the largest integer that is less than or equal to the specified number, not greater than or equal to.

E. TRUNC: This function can indeed be used with both NUMBER and DATE values to truncate them to a specified number of decimal places or to a particular component of a date.

References:

Oracle Database SQL Language Reference 12c Release 1 (12.1), Functions

Oracle Database SQL Language Reference 12c Release 1 (12.1), CONCAT Function

Oracle Database SQL Language Reference 12c Release 1 (12.1), MOD Function

Oracle Database SQL Language Reference 12c Release 1 (12.1), CEIL and FLOOR Functions

Oracle Database SQL Language Reference 12c Release 1 (12.1), TRUNC Function

## QUESTION: 22

Evaluate these commands which execute successfully `CREATE SEQUENCE ord_seq INCREMENT BY 1 START WITH 1 MAXVALUE 100000 CYCLE CACHE 5000`; Create table `ord_items(ord_no number(4) default ord_seq.nextval not null, item_no number(3), Qty number(3), Expiry_date date, Constraint it_pk primary key(ord_no, item_no), Constraint ord_fk foreign key (ord_no) references orders(ord_no))`; Which two statements are true about the ORD\_ITEMS table and the ORD\_SEQ sequence?

Option A : Any user inserting rows into table ORD\_ITEMS must have been granted access to sequence ORD\_SEQ.

Option B :

Column ORD\_NO gets the next number from sequence ORD\_SEQ whenever a row is inserted into ORD\_ITEMS and no explicit value is given for ORD\_NO.

Option C : Sequence ORD\_SEQ cycles back to 1 after every 5000 numbers and can cycle 20 times

Option D :

IF sequence ORD\_SEQ is dropped then the default value for column ORD\_NO will be NULL for rows inserted into ORD\_ITEMS.

Option E : Sequence ORD\_SEQ is guaranteed not to generate duplicate numbers.

Correct Answer: B,D

## Explanation/Reference:

Sequences and default values in Oracle play a crucial role in providing unique values for table columns.

Statement B is correct: When a row is inserted into ORD\_ITEMS without an explicit value for ORD\_NO, the ORD\_NO column gets the next number from the ORD\_SEQ sequence due to the DEFAULT ord\_seq.nextval clause.

Statement D is correct: If the ORD\_SEQ sequence is dropped, Oracle would not be able to get the next value for ORD\_NO from ORD\_SEQ, and unless another default is specified, the default for ORD\_NO would be NULL.

Statements A, C, and E are incorrect for the following reasons:

A is incorrect because the use of the sequence in the default clause of the table definition automatically grants the necessary permissions to use the sequence when inserting into the table.

C is incorrect because the sequence is defined to cycle when it reaches its MAXVALUE and not after every 5000 numbers. It is set to cycle, but the cycling event is triggered by the MAXVALUE limit.

E is incorrect because while a sequence is designed to produce unique numbers, if it cycles, it can potentially generate the same number again after reaching the MAXVALUE. This statement would only be true if there was no cycling.

## QUESTION: 23

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?

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

Correct Answer: C

## QUESTION: 24

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

### DEPT

Name	Null?	Type
DEPARTMENT_ID		NUMBER(4)
DEPARTMENT_NAME	NOT NULL	VARCHAR2(30)
MANAGER_ID		NUMBER(6)
LOCATION_ID		NUMBER(4)
CITY		VARCHAR2(30)

### LOCATIONS

Name	Null?	Type
LOCATION_ID	NOT NULL	NUMBER(4)
STREET_ADDRESS		VARCHAR2(40)
POSTAL_CODE		VARCHAR2(12)
CITY	NOT NULL	VARCHAR2(30)
STATE_PROVINCE		VARCHAR2(25)
COUNTRY_ID		CHAR(2)

You want to update the CITY column of the DEPT table for all the rows with the corresponding value in the CITY column of the LOCATIONS table for each department. Which SQL statement would you execute to accomplish the task?

- Option A : UPDATE dept dSET city = ALL (SELECT cityFROM locations IWHERE d.location\_id = I.location\_id);
- Option B : UPDATE dept dSET city = (SELECT cityFROM locations I)WHERE d.location\_id = I.location\_id;
- Option C : UPDATE dept dSET city = ANY (SELECT cityFROM locations I)
- Option D : UPDATE dept dSET city = (SELECT cityFROM locations IWHERE d.location\_id = I.location\_id);

Correct Answer: D

### QUESTION: 25

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

```
CUST NAME
-----
*** De Haan
*** Manuel Urman
```

and display the \* symbol inplace of the first name as follows:  
queries give the required output?

Which two

- Option A : SELECT LPAD(SUBSTR(cust\_name, INSTR(cust\_name, ' '),LENGTH(cust\_name),'\*') "CUST NAME"FROM customersWHERE INSTR(cust\_name, ' ',1,2)<>0;
- Option B : SELECT LPAD(SUBSTR(cust\_name, INSTR(cust\_name, ' '),LENGTH(cust\_name),'\*') "CUST NAME"FROM customersWHERE INSTR(cust\_name, ' ',-1,2)<>0;
- Option C : SELECT LPAD(SUBSTR(cust\_name, INSTR (cust\_name ' '),LENGTH(cust\_name) - INSTR(cust\_name, ' '), '\*') "CUST NAME"FROM customersWHERE INSTR(cust\_name, ' ',1,-2)<>0;
- Option D : SELECT LPAD(SUBSTR(cust\_name, INSTR (cust\_name ' '),LENGTH(cust\_name) - INSTR(cust\_name, ' '), '\*') "CUST NAME"FROM customersWHERE INSTR(cust\_name, ' ',1,2)<>0;

Correct Answer: A,B

### QUESTION: 26

Which three statements are true about inner and outer joins?

- Option A : A full outer join returns matched and unmatched rows.
- Option B : A full outer join must use Oracle syntax.
- Option C : Outer joins can be used when there are multiple join conditions on two tables.
- Option D : Outer joins can only be used between two tables per query.
- Option E : An inner join returns matched rows.
- Option F : A left or right outer join returns only unmatched rows.

Correct Answer: A,E

Explanation/Reference:

A: True. A full outer join does indeed return both matched and unmatched rows from both tables involved in the join. It combines the results of both left and right outer joins.

E: True. An inner join, by definition, returns rows that have matching values in both tables. Rows from both tables that do not match are not returned in an inner join result set.

Inner joins match rows from the joined tables based on the join condition, while outer joins include all rows from one or both tables regardless of whether a matching row exists in the other table.

References: The Oracle SQL documentation explains different types of joins, including inner joins, left and right outer joins, and full outer joins, clarifying how they differ in the result sets they produce.

## QUESTION: 27

Which statement is true about the INTERSECT operator used in compound queries?

Option A : It processes NULLS in the selected columns.

Option B : INTERSECT is of lower precedence than UNION or UNION ALL.

Option C : It ignores NULLS.

Option D : Multiple INTERSECT operators are not possible in the same SQL statement.

Correct Answer: A

Explanation/Reference:

For the question about the INTERSECT operator in SQL:

A. It processes NULLS in the selected columns: The INTERSECT operator compares two SELECT statements and returns rows that exist in both queries. It considers NULLS during this process, and NULLS in corresponding columns must match for rows to be considered equal. This means if both selected columns in the intersecting queries have NULLS, those rows will be included in the output.

Incorrect options:

B: INTERSECT has higher precedence than UNION and UNION ALL, not lower.

C: It does not ignore NULLs; rather, it processes them, as explained.

D: Multiple INTERSECT operators are indeed possible in the same SQL statement, allowing for complex compound queries.

## QUESTION: 28

Which statement is true regarding the UNION operator?

Option A : By default, the output is not sorted.

Option B : Null values are not ignored during duplicate checking.

Option C : Names of all columns must be identical across all select statements.

Option D : The number of columns selected in all select statements need not be the same.

Correct Answer: B

## QUESTION: 29

In which three situations does a new transaction always start?

Option A :

When issuing a SELECT FOR UPDATE statement after a CREATE TABLE AS SELECT statement was issued in the same session

Option B :

When issuing a CREATE INDEX statement after a CREATE TABLE statement completed unsuccessfully in the same session

Option C : When issuing a TRUNCATE statement after a SELECT statement was issued in the same session

Option D : When issuing a CREATE TABLE statement after a SELECT statement was issued in the same session

Option E :

When issuing the first Data Manipulation Language (DML) statement after a COMMIT or ROLLBACK statement was issued in the same session

Option F : When issuing a DML statement after a DML statement filed in the same session.

Correct Answer: A,C,E

Explanation/Reference:

Substitution variables in Oracle are used to replace a value dynamically during the execution of SQL statements. The behavior of these variables is well-documented:

C. A substitution variable prefixed with & always prompts only once for a value in a session: This is true. In a session, when you use a single ampersand (&), SQL\*Plus or SQL Developer will prompt for the value the first time the variable is encountered. The value for this variable will then be reused for the remainder of the session unless it is redefined.

D. A substitution variable can be used with any clause in a SELECT statement: Substitution variables can be placed in any part of a SQL statement, including the SELECT, WHERE, GROUP BY, ORDER BY, etc. They are not limited to any specific clause.

References:

Oracle SQL\*Plus User's Guide and Reference, which discusses substitution variables.

## QUESTION: 30

User HR has CREATE SESSION, CREATE ANY TABLE and UNLIMITED TABLESPACE privileges. User SCOTT has CREATE SESSION, CREATE TABLE and UNLIMITED TABLESPACE Privileges. HR successfully executes this statement:

```
CREATE TABLE scott.products (  
  prod_id NUMBER(2),  
  prod_name VARCHAR2(20));
```

HR attempts to execute:

```
1. INSERT INTO scott.products VALUES (1, 'LAPTOP');
```

SCOTT attempts to execute:

```
2. SELECT * FROM products;  
3. INSERT INTO scott.products VALUES (2, 'HDD');  
4. CREATE SYNONYM prod FOR products;
```

Which will execute successfully?

Option A : 1 only

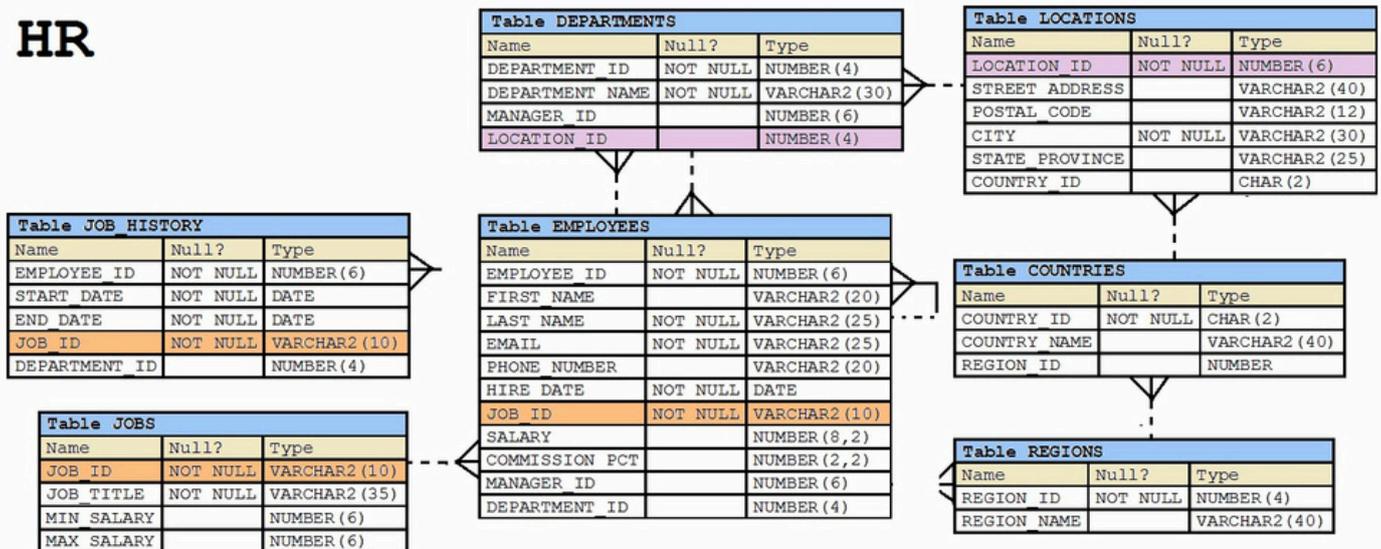
- Option B : 2 and 3 only
- Option C : 1,2 and 3
- Option D : 2, 3 and 4

Correct Answer: B

### QUESTION: 31

View the exhibit and examine the structure of the EMPLOYEES table.

**HR**



You want to select all employees having 100 as their MANAGER\_ID manages and their manager. You want output in two columns: the first column should have the employee's manager's LAST\_NAME and the second column should have the employee's LAST\_NAME. Which SQL statement would you execute?

- Option A : SELECT m.last\_name 'Manager', e.last\_name 'Employee' FROM employees m JOIN employees e ON m.employee\_id = e.manager\_id WHERE m.manager\_id=100;
- Option B : SELECT m.last\_name 'Manager', e.last\_name 'Employee' FROM employees m JOIN employees e ON m.employee\_id = e.manager\_id WHERE e.manager\_id=100;
- Option C : SELECT m.last\_name 'Manager', e.last\_name 'Employee' FROM employees m JOIN employees e ON e.employee\_id = m.manager\_id WHERE m.manager\_id=100;
- Option D : SELECT m.last\_name 'Manager', e.last\_name 'Employee' FROM employees m JOIN employees e WHERE m.employee\_id = e.manager\_id AND e.manager\_id=100

Correct Answer: B

### QUESTION: 32

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

## CUSTOMERS

Name	Null?	Type
CUSTOMER_ID	NOT NULL	NUMBER (6)
CUST_NAME		VARCHAR2 (20)
CUST_EMAIL		VARCHAR2 (30)
INCOME_LEVEL		VARCHAR2 (20)

CUSTOMER\_VU is a view based on CUSTOMERS\_BR1 table which has the same structure as CUSTOMERS table. CUSTOMERS need to be updated to reflect the latest information about the customers. What is the error in the following MERGE statement?

```
MERGE INTO customers c
  USING customer_vu cv
  ON (c.customer_id = cv.customer_id)
WHEN MATCHED THEN
  UPDATE SET
    c.customer_id = cv.customer_id,
    c.cust_name = cv.cust_name,
    c.cust_email = cv.cust_email,
    c.income_level = cv.income_level
WHEN NOT MATCHED THEN
  INSERT VALUES (cv.customer_id, cv.cust_name, cv.cust_email, cv.income_level)
  WHERE cv.income_level >100000;
```

Option A : The CUSTOMER\_ID column cannot be updated. Option B : The INTO clause is misplaced in the command. Option C : The WHERE clause cannot be used with INSERT. Option D : CUSTOMER\_VU cannot be used as a data source.

Correct Answer: A

## QUESTION: 33

Which two statements are true about transactions in the Oracle Database server?

Option A : An uncommitted transaction commits automatically if the user exits SQL\*Plus  
Option B : Data Manipulation Language (DML) statements always start a new transaction.  
Option C : A user can always see uncommitted updates made by the same user in a different session.  
Option D :  
A Data Definition Language (DDL) statement does a commit automatically only for the data dictionary updates caused by the DDL

Option E : A session can always see uncommitted updates made by itself.  
Option F :

If a session has an uncommitted transaction, then a DDL statement issue a COMMIT before starting a new transaction.

Correct Answer: A,B,E

Explanation/Reference:

A. Incorrect. An uncommitted transaction is not automatically committed if the user exits SQL\*Plus. It is rolled back unless otherwise specified. B. Correct. A DML statement such as INSERT, UPDATE, or DELETE will implicitly start a new transaction if there is no current transaction running in the session. C. Incorrect. A user cannot see uncommitted updates made by another session; a session can only see its own uncommitted changes. D. Incorrect. A Data Definition Language (DDL) statement automatically commits any outstanding transactions in the session, not just the changes to the data dictionary. E. Correct. A session can always see its own uncommitted updates. This is because Oracle uses a multiversion consistency model, allowing each user to see a consistent view of the data including their own changes. F. Incorrect but close. If a session has an uncommitted transaction, then a DDL statement does issue a COMMIT before executing and starting a new implicit transaction, not just for the data dictionary updates but for all pending changes in the session.

This information can be verified in the Oracle Database SQL Language Reference and Oracle Database Concepts documentation, which discuss transaction management and the behavior of DML and DDL statements within transactions.

## QUESTION: 34

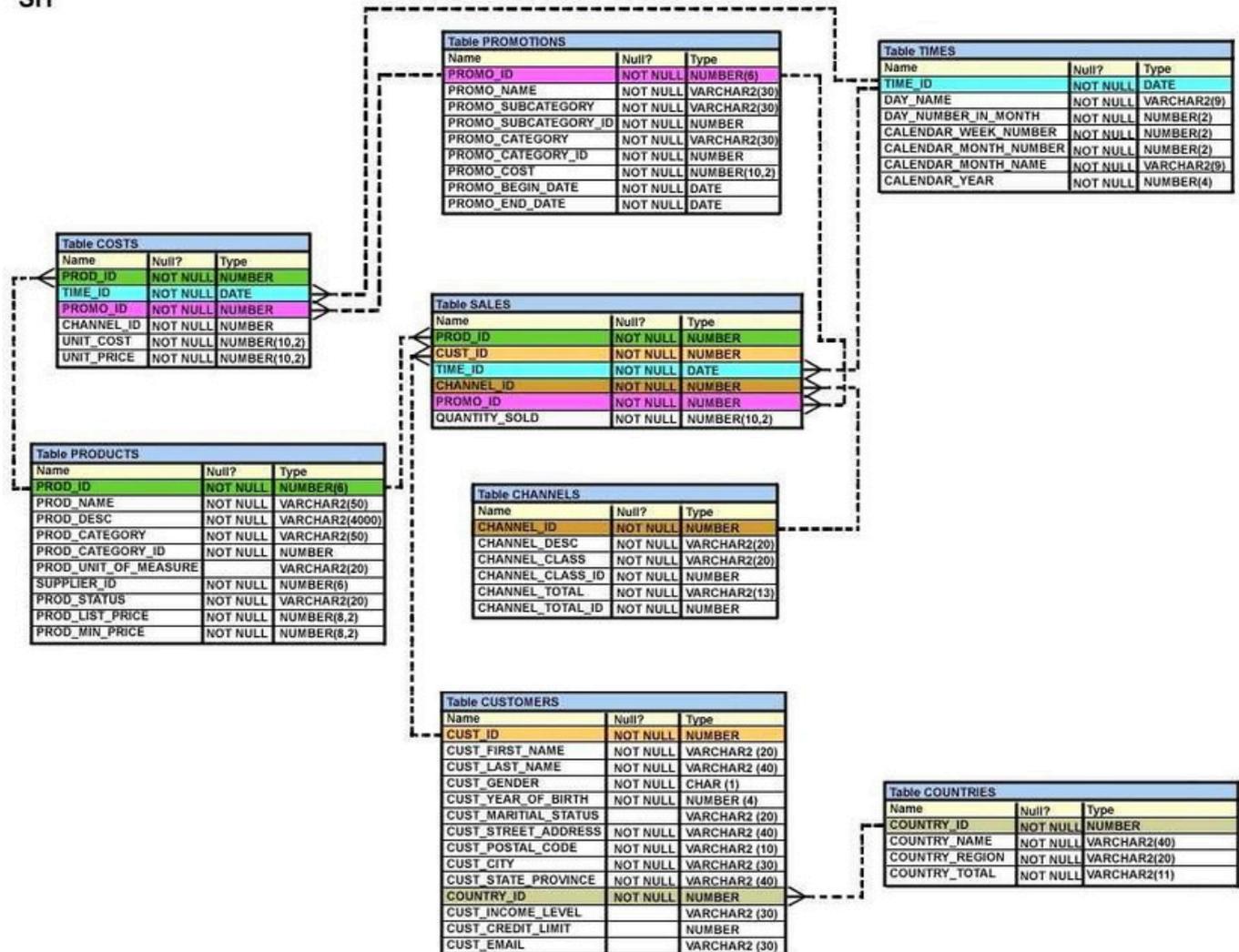
Which two statements are true regarding savepoints? (Choose two.)

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

## QUESTION: 35

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

SH



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

- Option A : `DELETE FROM products WHERE prod_id = (SELECT prod_id FROM sales WHERE time_id - 3*365 <= SYSDATE);`
- Option B : `DELETE FROM products WHERE prod_id = (SELECT prod_id FROM sales WHERE SYSDATE >= time_id - 3*365);`
- Option C : `DELETE FROM products WHERE prod_id IN (SELECT prod_id FROM sales WHERE SYSDATE - 3*365 >= time_id);`
- Option D : `DELETE FROM products WHERE prod_id IN (SELECT prod_id FROM sales WHERE time_id >= SYSDATE - 3*365);`

## QUESTION: 36

Which two are true about the data dictionary?

Option A : Base tables in the data dictionary have the prefix DBA\_.

Option B :

All user actions are recorded in the data dictionary.

Option C :

The data dictionary is constantly updated to reflect changes to database objects, permissions, and data.

Option D :

All users have permissions to access all information in the data dictionary by default

Option E :

The SYS user owns all base tables and user-accessible views in the data dictionary.

Correct Answer: C,E

Explanation/Reference:

C. True, the data dictionary is constantly updated to reflect changes to the metadata of the database objects, permissions, and structures, among other things. E. True, the SYS user owns all base tables in the data dictionary. These base tables underlie all data dictionary views that are accessible by the users.

A, B, and D are not correct because: A. Base tables do not necessarily have the prefix DBA\_; instead, DBA\_ is a prefix for administrative views that are accessible to users with DBA privileges. B. The data dictionary records metadata about the actions, not the actions themselves. D. Not all users have access to all information in the data dictionary. Access is controlled by privileges.

References:

Oracle documentation on data dictionary and dynamic performance views: Oracle Database Reference

Understanding the Oracle data dictionary: Oracle Database Concepts

### QUESTION: 37

Examine the data in the EMP table:

ENO	ENAME	SAL	DEPTNO
1001	John	12000	10
1002	Sam	40000	20
1003	Daniel	12000	20
1004	Andrea	5000	10

You execute this query: `SELECT deptno AS "Department", AVG(sal) AS AverageSalary, MAX(sal) AS "Max Salary" FROM emp WHERE sal >= 12000 GROUP BY "Department" ORDER BY AverageSalary;` Why does an error occur?

- Option A : An alias name must not be used in an ORDER BY clause.
- Option B : An alias name must not contain space characters.
- Option C : An alias name must not be used in a GROUP BY clause.
- Option D : An alias name must always be specified in quotes.

Correct Answer: C

Explanation/Reference:

C. True. The error occurs because the alias "Department" used in the GROUP BY clause is enclosed in double quotes, which makes it case-sensitive. However, the column deptno is not originally created with double quotes in the table definition, so you cannot refer to it with a case-sensitive alias in the GROUP BY clause. Oracle interprets "Department" as a string literal, not a column alias, in the GROUP BY clause. A is incorrect because you can use an alias in an ORDER BY clause. B is incorrect because an alias can contain space characters if it is quoted. D is incorrect because an alias does not always have to be specified in quotes, only when it includes special characters or spaces or if it is case-sensitive.

### QUESTION: 38

Which is the default column or columns for sorting output from compound queries using SET operators such as INTERSECT in a SQL statement?

- Option A : The first column in the last SELECT of the compound query
- Option B : The first NUMBER column in the first SELECT of the compound query

- Option C : The first VARCHAR2 column in the first SELECT of the compound query
- Option D : The first column in the first SELECT of the compound query
- Option E : The first NUMBER or VARCHAR2 column in the last SELECT of the compound query

Correct Answer: D

**Explanation/Reference:**

For the sorting of output in compound queries (INTERSECT, UNION, etc.):

D. The first column in the first SELECT of the compound query: By default, Oracle does not

automatically sort the results of SET operations unless an ORDER BY clause is explicitly stated.

However, if an ORDER BY is implied or specified without explicit columns, the default sorting would

logically involve the first column specified in the first SELECT statement of the compound query.

**QUESTION: 39**

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

PROMO_NAME	PROMO_CATEGORY	PROMO_COST	PROMO_BEGIN_DATE
NO PROMOTION #	NO PROMOTION	0	01-JAN-99
newspaper promotion #16-108	newspaper	200	23-DEC-00
post promotion #20-232	post	300	25-SEP-98
newspaper promotion #16-349	newspaper	400	10-JUL-98
internet promotion #14-471	internet	600	26-FEB-00
TV promotion #13-448	TV	1100	06-AUG-00
internet promotion #25-86	internet	1400	20-SEP-98
TV promotion #12-49	TV	1500	10-AUG-00
post promotion #21-166	post	2000	25-SEP-98
newspaper promotion #19-210	newspaper	2100	19-MAR-99
post promotion #20-282	post	2300	06-DEC-00
newspaper promotion #16-327	newspaper	2800	09-APR-99
internet promotion #29-289	internet	3000	01-NOV-98
TV promotion #12-252	TV	3100	20-JUN-98
magazine promotion #26-258	magazine	3200	04-MAY-00

PROMO\_BEGIN\_DATE is stored in the default date format, dd-mon-rr. You need to produce a report that provides the name, cost, and start date of all promos in the POST category that were launched before January 1, 2000. Which SQL statement would you use?

- Option A : SELECT promo\_name, promo\_cost, promo\_begin\_date FROM promotions WHERE promo\_category = 'post' AND promo\_begin\_date < '01-01-00';
- Option B : SELECT promo\_name, promo\_cost, promo\_begin\_date FROM promotions WHERE promo\_category LIKE 'P%' AND promo\_begin\_date < '1-JANUARY-00';
- Option C : SELECT promo\_name, promo\_cost, promo\_begin\_date FROM promotions WHERE promo\_cost

LIKE 'post%' AND promo\_begin\_date < '01-01-2000';

Option D : SELECT promo\_name, promo\_cost, promo\_begin\_date FROM promotions WHERE promo\_category LIKE '%post%' AND promo\_begin\_date < '1-JAN-00';

Correct Answer: D

## QUESTION: 40

Examine the description of the PRODUCT\_STATUS table:

Name	Null?	Type
PROD_ID	NOT NULL	NUMBER(2)
STATUS	NOT NULL	VARCHAR2(15)

The STATUS column contains the values 'IN STOCK' or 'OUT OF STOCK' for each row. Which two queries will execute successfully?

Option A :

SELECT prod\_id "CURRENT AVAILABILITY" || q('s not available)' FROM product\_status WHERE status = 'OUT OF STOCK';

Option B : SELECT prod\_id || q's not available" FROM product\_status WHERE status='OUT OF STOCK';

Option C :

SELECT prod\_id || q('s not available)' "CURRENT AVAILABILITY" FROM product\_status WHERE status = 'OUT OF STOCK';

Option D : SELECT prod\_id || q('s not available)' FROM product\_status WHERE status = 'OUT OF STOCK';

Option E :

SELECT prod\_id || q('s not available)' 'CURRENT AVAILABILITY' FROM product\_status WHERE status = 'OUT OF STOCK';

Option F : SELECT prod\_id || q"s not available" FROM product\_status WHERE status = 'OUT OF STOCK';

Correct Answer: C,D

Explanation/Reference:

Queries that will execute successfully given the PRODUCT\_STATUS table:

C. SELECT prod\_id || q('s not available)' "CURRENT AVAILABILITY" FROM

product\_status WHERE status = 'OUT OF STOCK'; This query will execute successfully because it

uses the q quote operator correctly to handle the inclusion of single quotes in the string.

D. `SELECT prod_id || q('s not available)' FROM product_status WHERE status = 'OUT OF`

`STOCK'`; Similar to the previous query, it uses the q quote operator correctly and will execute without syntax errors.

Options A, B, E, and F are incorrect because of the incorrect or incomplete use of the quote operator:

A is incorrect because it has an additional single quote before the FROM clause.

B is incorrect because of a missing ending single quote after the q quote operator.

E and F are incorrect because they both use the q quote operator incorrectly; `q('s not available)'` is not a valid use of the quote operator, and `q"s not available"` incorrectly uses double quotes instead of single quotes.

#### QUESTION: 41

Which statement is true regarding the default behavior of the ORDER BY clause?

Option A : In a character sort, the values are case-sensitive.

Option B : NULLs are not included in the sort operation.

Option C : Only columns that are specified in the SELECT list can be used in the ORDER BY clause.

Option D : Numeric values are displayed in descending order if they have decimal positions.

Correct Answer: A

#### QUESTION: 42

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

Option A : A sequence's unallocated cached values are lost if the instance shuts down.

Option B : Two or more tables cannot have keys generated from the same sequence.

Option C : A sequence number that was allocated can be rolled back if a transaction fails.

Option D : A sequence can issue duplicate values.

Option E : Sequences can always have gaps.

Option F : A sequence can only be dropped by a DBA.

Correct Answer: A,C,E

## Explanation/Reference:

Sequences are database objects used to generate unique numeric identifiers. Here's the correct understanding of sequences in Oracle:

A: Correct. Cached sequence values that are not yet used are lost if the database instance shuts down, as the cache is held in memory.

B: Incorrect. The same sequence can be used to generate keys for more than one table. There is no such limitation.

C: Correct. If a transaction using a sequence number is rolled back, the sequence number that was used or allocated is not reused.

D: Incorrect. By their nature and configuration, sequences are designed to avoid issuing duplicate values unless explicitly designed to cycle.

E: Correct. Sequences may have gaps, which can occur due to caching, sequence increment settings, or if a transaction using a sequence number is rolled back.

F: Incorrect. A sequence can be dropped by any user with adequate privileges, not just a DBA.

## QUESTION: 43

Examine the description of the EMPLOYEES table:



Which two statements will run successfully?

- Option A : `SELECT 'The first_name is ' || first_name || ' FROM employees ;`
- Option B : `SELECT 'The first_name is '''||first_name ||''' FROM employees ;`
- Option C : `SELECT 'The first_name is "' ||first_name||'" FROM employees ;`
- Option D : `SELECT 'The first_name is '| first_name|' FROM employees;`

Option E : SELECT 'The first\_name is \'' || first\_name || \'' FROM employees;

Correct Answer: B,D

#### QUESTION: 44

View the Exhibit and examine the details of the PRODUCT\_INFORMATION table. Exhibit  
SQL> SELECT d.department\_id FROM course\_details c INNER JOIN  
department\_details d ON c.department\_id<>d.department\_id;

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

You must display PRODUCT\_NAME from the table where the CATEGORY\_ID column has values 12 or 13, and the SUPPLIER\_ID column has the value 102088. You executed this SQL statement:

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

Which statement is true regarding the execution?

- Option A : It would not execute because the entire WHERE clause is not enclosed within parentheses.
- Option B : It would execute but would return no rows.
- Option C : It would not execute because the same column has been used twice with the AND logical operator.
- Option D : It would execute and return the desired result.

Correct Answer: B

## QUESTION: 45

Which statement is true regarding external tables?

Option A : The CREATE TABLE AS SELECT statement can be used to upload data into a normal table in the database from an external table.

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

Option C : The default REJECT LIMIT for external tables is UNLIMITED.

Option D : ORACLE\_LOADER and ORACLE\_DATAPUMP have exactly the same functionality when used with an external table.

Correct Answer: A

Explanation/Reference:

Explanation: [https://docs.oracle.com/cd/B28359\\_01/server.111/b28310/tables013.htm](https://docs.oracle.com/cd/B28359_01/server.111/b28310/tables013.htm)

## QUESTION: 46

Examine the data in the COLORS table:

RGB_HEX VALUE	COLOR NAME
FE0000	red
00FF00	green
0000FF	blue

Examine the data in the BRICKS table:

BRICK ID	COLOR_RGB_HEX_VALUE
1	FF0000
2	00FF00
3	FFFFFF

Which two queries return all the rows from COLORS?

Option A : SELECT

FROM bricks b

RIGHT JOIN colors c

ON b. color\_rgb\_hex\_value = c. rgb\_hex\_value;

Option B : SELECT

EROM colors C

LEFT JOIN bricks #

USING (rgb\_hex\_value) ;

Option C : SELECT

FROM bricks b

FULL JOIN colors C

ON b. color\_rgb\_hex\_value = c. rgb\_hex\_value;

Option D : SELECT \*

EROM bricks | b

JOIN colors C

ON b. color\_rgb\_hex\_value =c. rgb\_hex\_value;

Option E : SELECT

EROM colors C

LEET JOIN bricks b

ON b. color\_rgb\_hex\_value = c. rgb\_hex.

value

WHERE b. brick\_id > 0;

Correct Answer: A,B

Explanation/Reference:

The queries that will return all the rows from the COLORS table are those that ensure every record from COLORS is selected, regardless of whether there's a matching record in the BRICKS table:

Option A:

SELECT \* FROM bricks b RIGHT JOIN colors c ON b.color\_rgb\_hex\_value = c.

rgb\_hex\_value;

A right join will return all the rows from the right table (COLORS), with the matching rows from the left table (BRICKS). If there is no match, NULL will be returned for columns from BRICKS.

Option B:

```
SELECT * FROM colors c LEFT JOIN bricks b USING (rgb_hex_value);
```

A left join will return all the rows from the left table (COLORS), with the matching rows from the right table (BRICKS). If there is no match, NULL will be returned for columns from BRICKS. The USING clause indicates a join condition where columns with the same names are compared for equality.

Options C, D, and E will not return all the rows from the COLORS table:

Option C: The full join will return all rows from both tables, but it is not restricted to only the rows from COLORS.

Option D: An inner join will return only the matching rows between both tables, not all rows from COLORS.

Option E: This is a left join, which would typically return all rows from COLORS, but the WHERE clause restricts the result set to only those rows from COLORS that have a matching BRICK\_ID in BRICKS which is greater than 0, potentially excluding rows from COLORS.

#### QUESTION: 47

The PRODUCTS table has the following structure.

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

Evaluate the following two SQL statements: SQL>SELECT prod\_id, NVL2 (prod\_expiry\_date, prod\_expiry\_date + 15, ' ') FROM products; SQL>SELECT prod\_id, NVL (prod\_expiry\_date, prod\_expiry\_date + 15) FROM products; Which statement is true regarding the outcome?

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

Correct Answer: A

## Explanation/Reference:

### Using the NVL2 Function

The NVL2 function examines the first expression. If the first expression is not null, the NVL2 function returns the second expression. If the first expression is null, the third expression is returned.

### Syntax

NVL2(expr1, expr2, expr3)

In the syntax:

Expr1 is the source value or expression that may contain a null

Expr2 is the value that is returned if expr1 is not null

Expr3 is the value that is returned if expr1 is null

## QUESTION: 48

Examine the description of the EMPLOYEES table:

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(4)
LAST_NAME	NOT NULL	VARCHAR2(100)
SALARY	NOT NULL	NUMBER(6,2)
DEPARTMENT_ID	NOT NULL	NUMBER(4)

Examine this

query:

```
1 SELECT e.last_name,  
2        e.salary,  
3        a.avg_sal  
4 FROM employees e  
5 WHERE e.salary > (SELECT AVG(a.salary) AS avg_sal  
6                   FROM employees a  
7                   WHERE a.department_id = e.department_id)  
8 ORDER BY e.last_name;
```

Which line produces an error?

- Option A : Line 7
- Option B : Line 8
- Option C : Line 3
- Option D : Line 5

Correct Answer: C

## Explanation/Reference:

In the provided SQL query, the issue arises from the alias 'a.avg\_sal' which is defined in the subquery but is

being referenced in the SELECT list of the outer query. This is not permitted in SQL as the scope of the alias defined in the subquery is only within that subquery.

Here is the breakdown of the code and the error:

Line 1: Correct syntax for initiating a SELECT statement.

Line 2: Refers to 'e.salary', which is a correct reference to the 'salary' column using alias 'e' for the employees table.

Line 3: 'a.avg\_sal' attempts to reference an alias that is defined in the subquery within the outer query, which is not allowed. This is because 'avg\_sal' is defined in the subquery's SELECT list and cannot be referenced outside of it. The correct way to include the average salary from the subquery in the SELECT list of the main query would be to repeat the subquery or to use a join that includes the average salary.

Line 5-7: The subquery itself is correctly formed; it computes the average salary for the same department.

Line 8: The ORDER BY clause is properly referencing 'e.last\_name', which is defined in the outer query.

Therefore, the error occurs at Line 3 where 'a.avg\_sal' is not a valid reference in the SELECT list of the main query because it is defined in the subquery.

The rules of scope for aliases in subqueries are specified in the Oracle Database SQL Language Reference 12c documentation. Subquery aliases cannot be referenced outside their subquery.

## QUESTION: 49

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

Option A : The ORDER BY Clause can be used in a subquery.

Option B : A subquery can be used in the FROM clause of a SELECT statement.

Option C : If a subquery returns NULL, the main query may still return rows.

Option D : A subquery can be placed in a WHERE clause, a GROUP BY clause, or a HAVING clause.

Option E : Logical operators, such as AND, OR and NOT, cannot be used in the WHERE clause of a subquery.

## QUESTION: 50

Which two are true about self joins?

- Option A : They are always equijoins.
- Option B : They require the NOT EXISTS operator in the join condition.
- Option C : They have no join condition.
- Option D : They can use INNER JOIN and LEFT JOIN.
- Option E : They require table aliases.
- Option F : They require the EXISTS operator in the join condition.

Correct Answer: D,E

Explanation/Reference:

Self joins in Oracle Database 12c SQL have these characteristics:

Option D: They can use INNER JOIN and LEFT JOIN.

Self joins can indeed use various join types, including inner and left outer joins. A self join is a regular join, but the table is joined with itself.

Option E: They require table aliases.

When a table is joined to itself, aliases are required to distinguish between the different instances of the same table within the same query.

Options A, B, C, and F are incorrect:

Option A is incorrect because self joins can be non-equijoins as well.

Option B is incorrect because self joins do not require the NOT EXISTS operator. They may require a condition, but NOT EXISTS is not a necessity.

Option C is incorrect because a join condition is needed to relate the two instances of the same table in a self join.

Option F is incorrect for the same reason as B; the EXISTS operator is not a requirement for self joins.

### QUESTION: 51

Which three statements are true about performing Data Manipulation Language (DML) operations on a view with no INSTEAD OF triggers defined? (Choose three.)

Option A : Insert statements can always be done on a table through a view.

Option B : Views cannot be used to add rows to an underlying table if the table has columns with NOT NULL constraints lacking default values which are not referenced in the defining query of the view

Option C : Views cannot be used to query rows from an underlying table if the table has a PRIMARY KEY and the PRIMARY KEY columns are not referenced in the defining query of the view.

Option D : Delete statements can always be done on a table through a view

Option E : The WITH CHECK clause has no effect when deleting rows from the underlying table through the view.

Option F : Views cannot be used to add or modify rows in an underlying table if the defining query of the view contains the DISTINCT keyword.

Correct Answer: A,C,D

### QUESTION: 52

Which two statements are true? (Choose two.)

Option A :

The USER SYNONYMS view can provide information about private synonyms.

Option B : The user SYSTEM owns all the base tables and user-accessible views of the data dictionary.

Option C :

All the dynamic performance views prefixed with V\$ are accessible to all the database users.

Option D :

The USER OBJECTS view can provide information about the tables and views created by the user only.

Option E :

DICTIONARY is a view that contains the names of all the data dictionary views that the user can access.

Correct Answer: A,E

## Explanation/Reference:

A. The USER SYNONYMS view can provide information about private synonyms: This view contains information about private synonyms that have been created by the user. It lists all synonyms that are accessible to the user in their schema.

E. DICTIONARY is a view that contains the names of all the data dictionary views that the user can access: This view lists the names of all data dictionary views available to the user, providing a directory of useful database metadata.

Incorrect options:

B: The SYSTEM user does not own all base tables of the data dictionary; some are owned by SYS.

C: Not all users have access to dynamic performance views prefixed with V\$; access is restricted based on privileges.

D: The USER\_OBJECTS view lists all objects owned by the user, not just tables and views but also other types of schema objects like procedures and functions.

## QUESTION: 53

Which two are true about the MERGE statement?

Option A : The WHEN NOT MATCHED clause can be used to specify the deletions to be performed.

Option B : The WHEN NOT MATCHED clause can be used to specify the inserts to be performed.

Option C : The WHEN MATCHED clause can be used to specify the inserts to be performed.

Option D : The WHEN NOT MATCHED clause can be used to specify the updates to be performed.

Option E : The WHEN MATCHED clause can be used to specify the updates to be performed.

Correct Answer: B,E

## Explanation/Reference:

The correct answers regarding the MERGE statement are:

B. The WHEN NOT MATCHED clause can be used to specify the inserts to be performed. This is true. When a row from the source does not match any row in the target, the WHEN NOT MATCHED clause is where you specify the insert operation.

E. The WHEN MATCHED clause can be used to specify the updates to be performed. This is true as well. The WHEN MATCHED clause is where you specify the update (or delete) operation to be performed when the source and target rows match.

Options A, C, and D are incorrect:

A is incorrect because WHEN NOT MATCHED does not handle deletions, it is for inserts.

C is incorrect as inserts are not specified in the WHEN MATCHED clause but in the WHEN NOT MATCHED clause.

D is incorrect because updates are specified in the WHEN MATCHED clause, not the WHEN NOT MATCHED clause.

## QUESTION: 54

Examine the description of the PRODUCT\_DETAILS table:

NAME	NULL	TYPE
PRODUCT_ID	NOT NULL	NUMBER (2)
PRODUCT_NAME	NOT NULL	VARCHAR2 (25)
PRODUCT_PRICE		NUMBER (8, 2)
EXPIRY_DATE		DATE

Which two statements are true?

Which

Option A :

PRODUCT\_ID can be assigned the PEIMARY KEY constraint.

Option B :

EXPIRY\_DATE cannot be used in arithmetic expressions.

Option C :

EXPIRY\_DATE contains the SYSDATE by default if no date is assigned to it

Option D :

PRODUCT\_PRICE can be used in an arithmetic expression even if it has no value stored in it

Option E : PRODUCT\_PRICE contains the value zero by default if no value is assigned to it.

Option F :

PRODUCT\_NAME cannot contain duplicate values.

Correct Answer: A,D

**Explanation/Reference:**

A. PRODUCT\_ID can be assigned the PRIMARY KEY constraint.

In Oracle Database 12c, a PRIMARY KEY constraint is a combination of a NOT NULL constraint and a unique constraint. It ensures that the data contained in a column, or a group of columns, is unique among all the rows in the table and not null. Given the PRODUCT\_ID is marked as NOT NULL, it is a candidate for being a primary key because we can assume that it is intended to uniquely identify each product in the table.