Certkiller.1z0-061.75.QA

Number: 1z0-061
Passing Score: 800
Time Limit: 120 min
File Version: 3.4

Cert-Killer

1z0-061

Oracle Database 12c: SQL Fundamentals

- Provided information is extremely useful for you.
- By study this, I am reached my goal easily. Thanks a lot.
- I purchased the study guide package. I thought it was a great resource for studying for the exam.
- Awesome collection for preparing a certification exam; really your inputs are appreciable.
- If you want to score high marks in exam then go for this dump. I have personally used it and trust me I scored more than 90% in my exam.
- These are the most accurate study questions. Just focus on these and sit in your exam.
- The test guide is very realistic. If you have never taken an actual guide before this is extremely valuable and reduces the exam stress.
- Best of luck guys, Use this Exam and pass your Oracle Certification now.

Exam A

QUESTION 1
View the Exhibit and examine the structure of the 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 product
                         WHERE prod_list_price <
                               (SELECT MAX(prod_list_price)FROM p
```

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

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

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 2
View the Exhibits and examine the structures of the products, sales, and customers tables.

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

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 need to generate a report that gives details of the customer's last name, name of the product, and the quantity sold for a customers in 'Tokyo'.

Which two queries give the required result?

A)
```
SELECT c.cust_last_name,p.prod_name, s.quantity_sold
FROM sales s JOIN products p
USING(prod_id)
JOIN customers c
USING(cust_id)
WHERE c.cust_city='Tokyo';
```

B)
```
SELECT c.cust_last_name, p.prod_name, s.quantity_sold
FROM products p JOIN sales s JOIN customers c
ON(p.prod_id=s.prod_id)
ON(s.cust_id=c.cust_id)
WHERE c.cust_city='Tokyo';
```

C)
```
SELECT c.cust_last_name, p.prod_name, s.quantity_sold
FROM products p JOIN sales s
ON(p.prod_id=s.prod_id)
JOIN customers c
ON(s.cust_id=c.cust_id)
AND c.cust_city='Tokyo';
```

D)
```
SELECT c.cust_id,c.cust_last_name,p.prod_id, p.prod_name, s.qu
FROM products p JOIN sales s
USING(prod_id)
JOIN customers c
USING(cust_id)
WHERE c.cust_city='Tokyo';
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: AC
Section: (none)
Explanation

Explanation/Reference:

QUESTION 3
Examine the create table statements for the stores and sales tables.

SQL> CREATE TABLE stores(store_id NUMBER(4) CONSTRAINT store_id_pk PRIMARY KEY, store_name VARCHAR2(12), store_address VARCHAR2(20), start_date DATE);

SQL> CREATE TABLE sales(sales_id NUMBER(4) CONSTRAINT sales_id_pk PRIMARY KEY, item_id NUMBER(4), quantity NUMBER(10), sales_date DATE, store_id NUMBER(4), CONSTRAINT store_id_fk FOREIGN KEY(store_id) REFERENCES stores(store_id));

You executed the following statement:

SQL> DELETE from stores
WHERE store_id=900;

The statement fails due to the integrity constraint error:

ORA-02292: integrity constraint (HR.STORE_ID_FK) violated

Which three options ensure that the statement will execute successfully?

A. Disable the primary key in the STORES table.
B. Use CASCADE keyword with DELETE statement.
C. DELETE the rows with STORE_ID = 900 from the SALES table and then delete rows from STORES table.
D. Disable the FOREIGN KEY in SALES table and then delete the rows.
E. Create the foreign key in the SALES table on SALES_ID column with on DELETE CASCADE option.

Correct Answer: ACD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 4
You issued the following command:

SQL> DROP TABLE employees;

Which three statements are true?

A. All uncommitted transactions are committed.
B. All indexes and constraints defined on the table being dropped are also dropped.
C. Sequences used in the employees table become invalid.
D. The space used by the employees table is reclaimed immediately.
E. The employees table can be recovered using the rollback command.
F. The employees table is moved to the recycle bin.

Correct Answer: BCF
Section: (none)
Explanation

Explanation/Reference:
Reference: http://www.sqlcourse.com/drop.html

QUESTION 5
Examine the data in the CUST_NAME column of the customers table.

```
CUST_NAME
------------------
Renske Ladwig
Jason Mallin
Samuel McCain
Allan MCEwen
Irene Mikkilineni
Julia Nayer
```

You need to display customers' second names where the second name starts with "Mc" or "MC."

Which query gives the required output?

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

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

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

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

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 6
You want to create a sales table with the following column specifications and data types:
SALESID: Number
STOREID: Number
ITEMID: Number
QTY: Number, should be set to 1 when no value is specified SLSDATE: Date, should be set to current date when no value is specified PAYMENT: Characters up to 30 characters, should be set to CASH when no value is specified

Which statement would create the table?

A)
```
CREATE TABLE sales(
salesid NUMBER(4),
storeid NUMBER(4),
Itemid NUMBER(4),
qty NUMBER DEFAULT = 1,
slsdate DATE DEFAULT SYSDATE,
payment VARCHAR2(30) DEFAULT = "CASH");
```

B)
```
CREATE TABLE sales(
salesid NUMBER(4),
storeid NUMBER(4),
itemid NUMBER(4),
QTY NUMBER DEFAULT 1,
slsdate DATE DEFAULT SYSDATE,
payment VARCHAR2(30) DEFAULT 'CASH');
```

C)
```
CREATE TABLE sales(
salesid NUMBER(4),
storeid NUMBER(4),
itemid NUMBER(4),
qty NUMBER DEFAULT 1,
slsdate DATE DEFAULT 'SYSDATE',
payment VARCHAR2(30) DEFAULT CASH);
```

D)
```
CREATE TABLE sales(
salesid NUMBER(4),
storeid NUMBER(4),
itemid NUMBER(4),
qty NUMBER DEFAULT = 1,
slsdate DATE DEFAULT SYSDATE,
payment VARCHAR2(30) DEFAULT = "CASH");
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 7
You want to display 5 percent of the rows from the sales table for products with the lowest AMOUNT_SOLD and also want to include the rows that have the same AMOUNT_SOLD even if this causes the output to exceed 5 percent of the rows.

Which query will provide the required result?

A)
```
SELECT prod_id,cust_id,amount_sold
FROM sales
ORDER BY amount_sold
FETCH FIRST 5 PERCENT ROWS ONLY;
```

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

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

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

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 8
Evaluate the following query:

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

What would be the outcome?

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

Correct Answer: C
Section: (none)
Explanation

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

omitted, n defaults to zero

QUESTION 9
Using the customers table, you need to generate a report that shows 50% of each credit amount in each income level. The report should NOT show any repeated credit amounts in each income level.

Which query would give the required result?

A)
```
SELECT cust_income_level, DISTINCT cust_credit_limit * 0.50
   AS "50% Credit Limit"
FROM customers;
```

B)
```
SELECT DISTINCT cust_income_level, DISTINCT cust_credit_limit * 0.50
   AS "50% Credit Limit"
FROM customers;
```

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

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

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:
Duplicate Rows
Unless you indicate otherwise, SQL displays the results of a query without eliminating the duplicate rows.
To eliminate duplicate rows in the result, include the DISTINCT keyword in the SELECT clause immediately after the SELECT keyword.
You can specify multiple columns after the DISTINCT qualifier. The DISTINCT qualifier affects all the selected columns, and the result is every distinct combination of the columns.

QUESTION 10
Examine the structure of the employees table.

```
Name                                      Null?    Type
----------------------------------------- -------- ------
EMPLOYEE_ID                               NOT NULL NUMBE
FIRST_NAME                                         VARCH
LAST_NAME                                 NOT NULL VARCH
EMAIL                                     NOT NULL VARCH
PHONE_NUMBER                                       VARCH
HIRE_DATE                                 NOT NULL DATE
JOB_ID                                    NOT NULL VARCH
SALARY                                             NUMBE
COMMISSION_PCT                                     NUMBE
MANAGER_ID                                         NUMBE
DEPARTMENT_ID                                      NUMBE
```

You want to display the maximum and minimum salaries of employees hired 1 year ago.

Which two statements would get the correct output?

A)
```
SELECT MIN(salary),MAX(salary)
FROM (SELECT salary
      FROM employees
      WHERE hire_date < SYSDATE-365) ;
```

B)
```
SELECT minsal, maxsal
FROM (SELECT MIN(salary) minsal, MAX(salary) maxsal
      FROM employees
      WHERE hire_date < SYSDATE-365
      GROUP BY MIN(salary), MAX(salary)) ;
```

C)
```
SELECT minsal, maxsal
FROM (SELECT MIN(salary) minsal, MAX(salary) maxsal
      FROM employees
      WHERE hire_date < SYSDATE-365)
GROUP BY maxsal, minsal ;
```

D)
```
SELECT MIN(salary) minsal, MAX(salary) maxsal
FROM employees
WHERE hire_date < SYSDATE-365
GROUP BY MIN(salary), MAX(salary);
```

A. Option A
B. Option B

C. Option C
D. Option D

Correct Answer: BD
Section: (none)
Explanation

Explanation/Reference:
http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/index.jsp?topic= %2Fcom.ibm.db2z10.doc.sqlref%2Fsrc%2Ftpc%2Fdb2z_sql_subselectexamples.htm

QUESTION 11
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 subquery is null?

A. It produces an error.
B. It executes but returns no rows.
C. It generates output for null as well as the other values produced by the subquery.
D. It ignores the null value and generates output for the other values produced by the subquery.

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:
Answer is Modified.

QUESTION 12
You issue the following command to drop the products table:

SQL> DROP TABLE products;

Which three statements are true about the implication of this command?

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

Correct Answer: ABD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 13
Which statement is true regarding the default behavior of the order by clause?

A. In a character sort, the values are case-sensitive.
B. NULL values are not considered at all by the sort operation.
C. Only those columns that are specified in the select list can be used in the order by clause.
D. Numeric values are displayed from the maximum to the minimum value if they have decimal positions.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:
Explanation:
Character Strings and Dates
Character strings and date values are enclosed with single quotation marks. Character values are case-sensitive and date values are format-sensitive.
The default date display format is DD-MON-RR.

QUESTION 14
Examine the structure of the employees table:

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
EMAIL	NOT NULL	VARCHAR2(25)
PHONE_NUMBER		VARCHAR2(20)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

There is a parent/child relationship between EMPLOYEE_ID and MANAGER_ID.

You want to display the name, joining date, and manager for all the employees. Newly hired employees are yet to be assigned a department or a manager. For them, 'No Manager1 should be displayed in the manager column.
Which SQL query gets the required output?

A)
```
SELECT e.last_name, e.hire_date, NVL(m.last_name, 'No Manager') Manager
FROM employees e JOIN employees m
ON (e.manager_id = m.employee_id);
```

B)
```
SELECT e.last_name, e.hire_date, NVL(m.last_name, 'No Manager') Manager
FROM employees e LEFT OUTER JOIN employees m
ON (e.manager_id = m.employee_id);
```

C)
```
SELECT e.last_name, e.hire_date, NVL(m.last_name, 'No Manager') Manager
FROM employees e RIGHT OUTER JOIN employees m
ON (e.manager_id = m.employee_id);
```

D)
```
SELECT e.last_name, e.hire_date, NVL(m.last_name, 'No Manager') Manager
FROM employees e NATURAL JOIN employees m
ON (e.manager_id = m.employee_id);
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:
Reference: http://ivrainbow65.blogspot.com/

QUESTION 15
The customers table has the following structure:

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

You need to write a query that does the following tasks:
1. Display the first name and tax amount of the customers. Tax is 5% of their credit limit.
2. Only those customers whose income level has a value should be considered.
3. Customers whose tax amount is null should not be considered.

Which statement accomplishes all the required tasks?

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

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

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

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

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 16
View the Exhibits and examine the structures of the 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)

Which two SQL statements would give the same output?

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;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: AC
Section: (none)
Explanation

Explanation/Reference:

QUESTION 17
View the Exhibit and examine the description of SALES and PROMOTIONS tables.

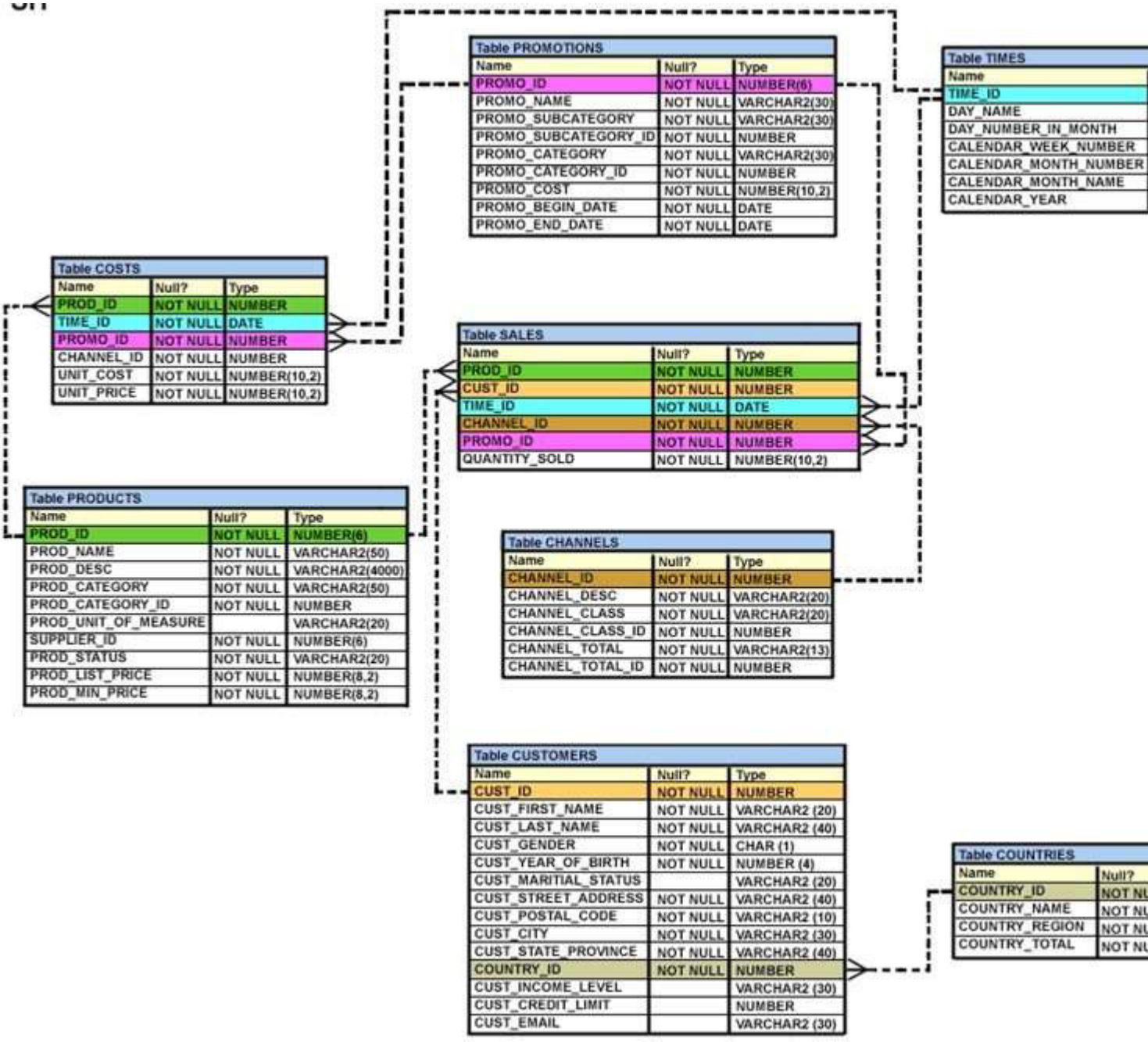

Table PROMOTIONS

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

Table TIMES

Name
TIME_ID
DAY_NAME
DAY_NUMBER_IN_MONTH
CALENDAR_WEEK_NUMBER
CALENDAR_MONTH_NUMBER
CALENDAR_MONTH_NAME
CALENDAR_YEAR

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)

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 CHANNELS

Name	Null?	Type
CHANNEL_ID	NOT NULL	NUMBER
CHANNEL_DESC	NOT NULL	VARCHAR2(20)
CHANNEL_CLASS	NOT NULL	VARCHAR2(20)
CHANNEL_CLASS_ID	NOT NULL	NUMBER
CHANNEL_TOTAL	NOT NULL	VARCHAR2(13)
CHANNEL_TOTAL_ID	NOT NULL	NUMBER

Table CUSTOMERS

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

Table COUNTRIES

Name	Null?
COUNTRY_ID	NOT NU
COUNTRY_NAME	NOT NU
COUNTRY_REGION	NOT NU
COUNTRY_TOTAL	NOT NU

You want to delete rows from the sales table, where the PROMO_NAME column in the promotions table has either blowout sale of everyday low prices as values.

Which three delete statements are valid?

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)
```
DELETE
FROM sales
WHERE promo_id IN (SELECT promo_id
                   FROM promotions
                   WHERE promo_name IN ('blowout sale','everyday low price'))
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: BCD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 18
Which three tasks can be performed using SQL functions built into Oracle Database?

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

Correct Answer: ABC
Section: (none)
Explanation

Explanation/Reference:

QUESTION 19
You want to display 5 percent of the employees with the highest salaries in the EMPLOYEES table.

Which query will generate the required result?

A)
```
SELECT employee_id, last_name, salary
FROM employees
ORDER BY salary
FETCH FIRST 5 PERCENT ROWS ONLY;
```

B)
```
SELECT employee_id, last_name, salary
FROM employees
ORDER BY salary DESC
FETCH FIRST 5 PERCENT ROWS ONLY;
```

C)
```
SELECT employee_id, last_name, salary
FROM employees
ORDER BY salary DESC
FETCH FIRST 5 PERCENT ROWS ONLY WITH TIES;
```

D)
```
SELECT employee_id, last_name, salary
FROM employees
ORDER BY salary DESC
FETCH 5 PERCENT ROWS ONLY;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 20
Which three SQL statements would display the value 1890.55 as $1, 890.55?

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;
```

A. Option A
B. Option B
C. Option C
D. Option D
E. Option E

Correct Answer: ADE
Section: (none)
Explanation

Explanation/Reference:

QUESTION 21
View the Exhibit and evaluate the structure and data in the CUST_STATUS table.

```
CUST_STATUS
Name            Null?       Type
--------------- --------    ---------
CUSTNO          NOT NULL    NUMBER(2)
AMT_SPENT                   NUMBER(10,2)
CREDIT_LIMIT                NUMBER(10,2)
```

```
CUSTNO      AMT_SPENT    CREDIT_LIMIT
------- -------------- -----------------
1           1000         1000
2           2000         2500
3                        3000
4           3000         2800
```

You issue the following SQL statement:

```
SQL> SELECT custno, NVL2(NULLIF(amt_spent, credit_limit), 0, 1000)"BONUS"
     FROM cust_status;
```

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

A. It produces an error because the AMT_SPENT column contains a null value.
B. It displays a bonus of 1000 for all customers whose AMT_SPENT is less than CREDIT_LIMIT.
C. It displays a bonus of 1000 for all customers whose AMT_SPENT equals CREDIT_LIMIT, or AMT_SPENT is null.
D. It produces an error because the TO_NUMBER function must be used to convert the result of the NULLIF function before it can be used by the NVL2 function.

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:
Explanation:
The NULLIF Function
The NULLIF function tests two terms for equality. If they are equal the function returns a null, else it returns the first of the two terms tested. The NULLIF function takes two mandatory parameters of any data type. The syntax is NULLIF(ifunequal, comparison_term), where the parameters ifunequal and comparison_term are compared. If they are identical, then NULL is returned. If they differ, the ifunequal parameter is returned.

QUESTION 22
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

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 23
Examine the types and examples of relationships that follow:

1. One-to-one a) Teacher to students
2. One-to-many b) Employees to Manager
3. Many-to-one c) Person to SSN
4. Many-to-many d) Customers to products

Which option indicates the correctly matched relationships?

A. 1-a, 2-b, 3-c, and 4-d
B. 1-c, 2-d, 3-a, and 4-b
C. 1-c, 2-a, 3-b, and 4-d
D. 1-d, 2-b, 3-a, and 4-c

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:
Corrected.

QUESTION 24
Which normal form is a table in if it has no multi-valued attributes and no partial dependencies?

A. First normal form
B. Second normal form
C. Third normal form
D. Fourth normal form

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 25
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

Evaluate the following SQL statement:

```
SQL>SELECT promo_name,CASE
                      WHEN promo_cost >=(SELECT AVG(pr
                                         FROM promotio
                                         WHERE promo_c
                      THEN 'HIGH'
                      ELSE 'LOW'
                      END COST_REMARK
FROM promotions;
```

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

A. It shows COST_REMARK for all the promos in the table.
B. It produces an error because the SUBQUERY gives an error.
C. It shows COST_REMARK for all the promos in the promo category 'TV'
D. It produces an error because SUBQUERIES cannot be used with the case expression.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 26
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_MARITIAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

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

Which SQL statement would produce the required result?

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;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:
Explanation:
NVL Function
Converts a null value to an actual value:
Data types that can be used are date, character, and number.
Data types must match:
NVL(commission_pct, 0)
NVL(hire_date, '01-JAN-97')

NVL(job_id, 'No Job Yet')

QUESTION 27
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_MARITIAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

NEW_CUSTOMERS is a new table with the columns CUST_ID, CUST_NAME and CUST_CITY that have the same data types and size as the corresponding columns in the customers table.

Evaluate the following insert statement:

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

The insert statement fails when executed.

What could be the reason?

A. The values clause cannot be used in an INSERT with a subquery.
B. Column names in the NEW_CUSTOMERS and CUSTOMERS tables do not match.
C. The where clause cannot be used in a subquery embedded in an INSERT statement.
D. The total number of columns in the NEW_CUSTOMERS table does not match the total number of columns in the CUSTOMERS table.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:
Explanation:
Copying Rows from Another Table
Write your INSERT statement with a subquery:
Do not use the VALUES clause.

Match the number of columns in the INSERT clause to those in the subquery. Inserts all the rows returned by the subquery in the table, sales_reps.

QUESTION 28
Examine the structure and data of the CUST_TRANS table:

```
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-mon-rr in the CUST_TRANS table. Which three SQL statements would execute successfully?

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;

Correct Answer: ACD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 29
Examine the data in the ORD_ITEMS table:

```
ORD_NO    ITEM_NO   QTY
------    -------   ------
1         111       10
1         222       20
1         333       30
2         333       30
2         444       40
3         111       40
```

Evaluate the following 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 outcome of the above query?

A. It gives an error because the having clause should be specified after the group by clause.
B. It gives an error because all the aggregate functions used in the having clause must be specified in the select list.
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.
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
Section: (none)
Explanation

Explanation/Reference:

QUESTION 30
YOU need to display the date ll-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, ye
FROM DUAL;
```

C)
```
SELECT TO_CHAR(TO_DATE('11-oct-2007'), 'fmDdthsp "of" Month,
FROM DUAL;
```

D)
```
SELECT TO_DATE(TO_CHAR('11-oct-2007','fmDdspth ''of'' Month,
FROM DUAL;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 31
You need to create a table with the following column specifications:
1. Employee ID (numeric data type) for each employee
2. Employee Name (character data type) that stores the employee name
3. Hire date, which stores the date of joining the organization for each employee
4. Status (character data type), that contains the value 'active1 if no data is entered
5. Resume (character large object [CLOB] data type), which contains the resume submitted by the employee

Which is the correct syntax to create this table?

A)
```
CREATE TABLE EMP_1
  (emp_id NUMBER(4),
emp_name VARCHAR2(25),
start_date DATE,
e_status VARCHAR2(10) DEFAULT 'ACTIVE',
resume CLOB(200));
```

B)
```
CREATE TABLE 1_EMP
  (emp_id NUMBER(4),
emp_name VARCHAR2(25),
start_date DATE,
emp_status VARCHAR2(10) DEFAULT 'ACTIVE',
resume CLOB);
```

C)
```
CREATE TABLE EMP_1
  (emp_id NUMBER(4),
emp_name VARCHAR2(25),
start_date DATE,
emp_status VARCHAR2(10) DEFAULT "ACTIVE",
resume CLOB);
```

D)
```
CREATE TABLE EMP_1
  (emp_id  NUMBER,
emp_name VARCHAR2(25),
start_date DATE,
emp_status VARCHAR2(10) DEFAULT 'ACTIVE',
resume CLOB);
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:
Explanation:
CLOB Character data (up to 4 GB)
NUMBER [(p, s)] Number having precision p and scale s (Precision is the total number of decimal digits and

scale is the number of digits to the right of the decimal point; precision can range from 1 to 38, and scale can range from 84 to 127.)

QUESTION 32
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?

A)
```
SELECT promo_name, promo_cost, promo_begin_date
FROM promotions
WHERE promo_category = 'post' AND promo_begin_date < '01-01-00';
```

B)
```
SELECT promo_name, promo_cost, promo_begin_date
FROM promotions
WHERE promo_cost LIKE 'post%' AND promo_begin_date < '01-01-2000';
```

C)
```
SELECT promo_name, promo_cost, promo_begin_date
FROM promotions
WHERE promo_category LIKE 'P%' AND promo_begin_date < '1-JANUARY-00';
```

D)
```
SELECT promo_name, promo_cost, promo_begin_date
FROM promotions
WHERE promo_category LIKE '%post%' AND promo_begin_date < '1-JAN-00';
```

A. Option A
B. Option B
C. Option C

D. Option D

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 33
Examine the data in the PROMO_BEGIN_DATE column of the promotions table:

```
PROMO_BEGIN_DATE
----------------
04-jan-00
10-jan-00
15-dec-99
18-oct-98
22-aug-99
```

You want to display the number of promotions started in 1999 and 2000.

Which query gives the correct output?

A)
```
SELECT SUM(DECODE(SUBSTR(promo_begin_date,8),'00',1,0)) "2000",
SUM(DECODE(SUBSTR(promo_begin_date,8),'99',1,0)) "1999"
FROM promotions;
```

B)
```
SELECT SUM(CASE TO_CHAR(promo_begin_date,'yyyy') WHEN '99' THEN 1
ELSE 0 END) "1999",SUM(CASE TO_CHAR(promo_begin_date,'yyyy') WHEN
ELSE 0 END) "2000"
FROM promotions;
```

C)
```
SELECT COUNT(CASE TO_CHAR(promo_begin_date,'yyyy') WHEN '99' THEN
ELSE 0 END) "1999",COUNT(CASE TO_CHAR(promo_begin_date,'yyyy') WHE
ELSE 0 END) "2000"
FROM promotions;
```

D)
```
SELECT COUNT(DECODE(SUBSTR(TO_CHAR(promo_begin_date,'yyyy'), 8), '
COUNT(DECODE(SUBSTR(TO_CHAR(promo_begin_date,'yyyy'), 8),'2000', 1
0)) "2000"
FROM promotions;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: A
Section: (none)

Explanation

Explanation/Reference:

QUESTION 34
View the Exhibit and examine the structure of the 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)

All products have a list price.

You issue the following command to display the total price of each product after a discount of 25% and a tax of 15% are applied on it. Freight charges of $100 have to be applied to all the products.

```
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;
```

What would be the outcome if all the parentheses are removed from the above statement?

A. It produces a syntax error.
B. The result remains unchanged.
C. The total price value would be lower than the correct value.
D. The total price value would be higher than the correct value.

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 35
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_MARITIAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

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?

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;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 36
You want to display the date for the first Monday of the next month and issue the following command:

```
SQL>SELECT TO_CHAR(NEXT_DAY(LAST_DAY(SYSDATE),'MON'),
                   'dd "is the first Monday for" fmmonth rrrr')
FROM DUAL;
```

What is the outcome?

A. It executes successfully and returns the correct result.
B. It executes successfully but does not return the correct result.
C. It generates an error because TO_CHAR should be replaced with TO_DATE.
D. It generates an error because rrrr should be replaced by rr in the format string.
E. It generates an error because fm and double quotation marks should not be used in the format string.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 37
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';
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:
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

QUESTION 38
View the Exhibit and examine the data in the products table.

PROD_ID	PROD_NAME	PROD_CATEGORY	PROD_MIN_PRICE	P
101	Envoy 256MB - 40GB	Hardware	6000	N
102	Y Box	Electronics	9000	
103	DVD-R Disc, 4.7 GB	Software/Other	2000	N
104	Documentation Set - Spanish	Software/Other	4000	

You need to display product names from the products table that belong to the 'software/other' category with minimum prices as either S2000 or S4000 and no unit of measure.

You issue the following query:

```
SQL>SELECT prod_name, prod_category, prod_min_price
    FROM products
    WHERE prod_category LIKE '%Other%' AND (prod_min_price = 2000 OR
                prod_min_price = 4000) AND prod_unit_of_measure <> '';
```

Which statement is true regarding the above query?

A. It executes successfully but returns no result.
B. It executes successfully and returns the required result.
C. It generates an error because the condition specified for PROD_UNIT_OF_MEASURE is not valid.
D. It generates an error because the condition specified for the prod category column is not valid.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 39
You need to generate a list of all customer last names with their credit limits from the customers table.

Those customers who do not have a credit limit should appear last in the list.

Which two queries would achieve the required result?

A)
```
SELECT cust_last_name, cust_credit_limit
FROM customers
ORDER BY cust_credit_limit DESC;
```

B)
```
SELECT cust_last_name, cust_credit_limit
FROM customers
ORDER BY cust_credit_limit;
```

C)
```
SELECT cust_last_name, cust_credit_limit
FROM customers
ORDER BY cust_credit_limit NULLS LAST;
```

D)
```
SELECT cust_last_name, cust_credit_limit
FROM customers
ORDER BY cust_last_name, cust_credit_limit NULLS LAST;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: BC
Section: (none)
Explanation

Explanation/Reference:
Explanation:
If the ORDER BY clause is not used, the sort order is undefined, and the Oracle server may not fetch rows in the same order for the same query twice. Use the ORDER BY clause to display the rows in a specific order.
Note: Use the keywords NULLS FIRST or NULLS LAST to specify whether returned rows containing null values should appear first or last in the ordering sequence. ANSWER C Sorting
The default sort order is ascending:
· Numeric values are displayed with the lowest values first (for example, 1 to 999). · Date values are displayed with the earliest value first (for example, 01-JAN-92 before 01-JAN-
95).
· Character values are displayed in the alphabetical order (for example, "A" first and "Z" last). · Null values are

displayed last for ascending sequences and first for descending sequences.
- ANSWER B
· You can also sort by a column that is not in the SELECT list.

QUESTION 40
Examine the structure of the customers table:

```
Name                 Null?          Type
-----------------    -----------    ------------
CUSTNO               NOT NULL       NUMBER(3)
CUSTNAME             NOT NULL       VARCHAR2(25)
CUSTADDRESS                         VARCHAR2(35)
CUST_CREDIT_LIMIT                   NUMBER(5)
```

CUSTNO is the primary key in the table. You want to find out if any customers' details have been entered more than once using different CUSTNO, by listing all the duplicate names.

Which two methods can you use to get the required result?

A. Self-join
B. Subquery
C. Full outer-join with self-join
D. Left outer-join with self-join
E. Right outer-join with self-join

Correct Answer: AB
Section: (none)
Explanation

Explanation/Reference:

QUESTION 41
Evaluate the following query:

```
SQL> SELECT promo_name || q'{'s start date was \}' || promo_begin_date
                          AS "Promotion Launches"
FROM promotions;
```

What would be the outcome of the above query?

A. It produces an error because flower braces have been used.
B. It produces an error because the data types are not matching.
C. It executes successfully and introduces an 's at the end of each PROMO_NAME in the output.
D. It executes successfully and displays the literal "{'s start date was \} * for each row in the output.

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:
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 multi byte, 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.

QUESTION 42
View the Exhibit and examine the data in the employees table:

EMPLOYEES

EMPLOYEE_ID	EMPLOYEE_NAME	MANAGER_ID	SALARY	DE
7369	SMITH	7902	800	
77698	ALLEN		1600	
7902	WARD		1250	
7654	MARTIN	7698	1250	

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
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?

A. INNER JOIN

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

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 43
View the Exhibit and examine the structures of the employees and departments tables.

EMPLOYEES

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(10,2)
COMMISSION		NUMBER(6,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

DEPARTMENTS

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

You want to update the employees table as follows:
-Update only those employees who work in Boston or Seattle (locations 2900 and 2700). -Set department_id for these employees to the department_id corresponding to London (location_id 2100).
-Set the employees' salary in iocation_id 2100 to 1.1 times the average salary of their department.
-Set the employees' commission in iocation_id 2100 to 1.5 times the average commission of their department.

You issue the following command:

```
SQL>UPDATE employees
     SET department_id =
          (SELECT department_id
              FROM departments
              WHERE location_id = 2100),
          (salary, commission) =
          (SELECT 1.1*AVG(salary), 1.5*AVG(commission)
            FROM employees, departments
            WHERE departments.location_id IN(2900,2700,2100))
          WHERE department_id IN
          (SELECT department_id
            FROM departments
            WHERE location_id = 2900
                OR location_id = 2700);
```

What is the outcome?

A. It executes successfully and gives the correct result.
B. It executes successfully but does not give the correct result.
C. It generates an error because a subquery cannot have a join condition in an update statement.
D. It generates an error because multiple columns (SALARY, COMMISSION) cannot be specified together in an update statement.

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 44
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,'$9999999
FROM transactions
WHERE TO_NUMBER(TO_DATE(trans_date,'hh24')) < 12 AND COALESCE(trans_amt,NULL)<>N
```

B)
```
SELECT TO_CHAR(trans_date,'dd-mon-yyyy hh24:mi:ss'), NVL(TO_CHAR(trans_amt,'$999
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_am
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(t
FROM transactions
WHERE TO_DATE(trans_date,'hh24') < 12;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 45
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.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 46
You need to produce a report where each customer's credit limit has been incremented by $1000. In the output, the customer's last name should have the heading Name and the incremented credit limit should be labeled New credit Limit. The column headings should have only the first letter of each word in uppercase.

Which statement would accomplish this requirement?

A)
```
SELECT cust_last_name Name, cust_credit_limit + 1000
      "New Credit Limit"
FROM customers;
```

B)
```
SELECT cust_last_name AS Name, cust_credit_limit + 1000
       AS New Credit Limit
FROM customers;
```

C)
```
SELECT cust_last_name AS "Name", cust_credit_limit + 1000
       AS "New Credit Limit"
FROM customers;
```

D)
```
SELECT INITCAP(cust_last_name) "Name", cust_credit_limit + 1000
       INITCAP("NEW CREDIT LIMIT")
FROM customers;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:
Explanation:
A column alias:
- Renames a column heading
- Is useful with calculations
- Immediately follows the column name (There can also be the optional AS keyword between the column name and the alias.)
- Requires double quotation marks if it contains spaces or special characters, or if it is case sensitive.

QUESTION 47

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));
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: D
Section: (none)
Explanation

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

QUESTION 48
You need to list the employees in DEPARTMENT_ID 30 in a single row, ordered by HIRE_DATE.

Examine the sample output:

```
Emp_list                                                    E
------------------------------------------------------------ -
Raphaely; Khoo; Tobias; Baida; Himuro; Colmenares           (
```

Which query will provide the required output?

A)
```
SELECT LISTAGG(last_name)
WITHIN GROUP ORDER BY (hire_date) "Emp_list", MIN(hire_date) "Earliest"
FROM employees
WHERE department_id = 30;
```

B)
```
SELECT LISTAGG(last_name, '; ')
WITHIN GROUP (ORDER BY hire_date) "Emp_list", MIN(hire_date) "Earliest"
FROM employees
WHERE department_id = 30;
```

C)
```
SELECT LISTAGG(last_name, '; ') "Emp_list", MIN(hire_date) "Earliest"
FROM employees
WHERE department_id = 30
WITHIN GROUP ORDER BY hire_date;
```

D)
```
SELECT LISTAGG(last_name, '; ') "EMP_LIST", MIN(hire_date) "Earliest"
FROM employees
WHERE department_id = 30
ORDER BY hire_date;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:
Reference: http://docs.oracle.com/cd/E11882_01/server.112/e10592/functions089.htm

QUESTION 49
Which two statements are true regarding the count function?

A. The count function can be used only for CHAR, VARCHAR2, and NUMBER data types.
B. Count (*) returns the number of rows including duplicate rows and rows containing null value in any of the columns.
C. Count (cust_id) returns the number of rows including rows with duplicate customer IDs and NULL value in the CUST_ID column.
D. Count (distinct inv_amt) returns the number of rows excluding rows containing duplicates and NULL values in the INV_AMT column.
E. A select statement using the COUNT function with a DISTINCT keyword cannot have a where clause.

Correct Answer: BD
Section: (none)
Explanation

Explanation/Reference:
Explanation:
Using the COUNT Function
The COUNT function has three formats:
COUNT(*)

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

QUESTION 50
Which two statements are true regarding single row functions?

A. MOD: returns the quotient of a division
B. TRUNC: can be used with number and date values
C. CONCAT: can be used to combine any number of values
D. SYSDATE: returns the database server current date and time
E. INSTR: can be used to find only the first occurrence of a character in a string
F. TRIM: can be used to remove all the occurrences of a character from a string

Correct Answer: BD
Section: (none)
Explanation

Explanation/Reference:
Explanation:
ROUND: Rounds value to a specified decimal
TRUNC: Truncates value to a specified decimal
MOD: Returns remainder of division
SYSDATE is a date function that returns the current database server date and time.

Date-Manipulation Functions
Date functions operate on Oracle dates. All date functions return a value of the DATE data type except MONTHS_BETWEEN, which returns a numeric value. MONTHS_BETWEEN(date1, date2): Finds the number of months between date1 and date2. The result can be positive or negative. If date1 is later than date2, the result is positive; if date1 is earlier than date2, the result is negative. The noninteger part of the result represents a portion of the month.
ADD_MONTHS(date, n): Adds n number of calendar months to date. The value of n must be an integer and can be negative.
NEXT_DAY(date, 'char'): Finds the date of the next specified day of the week ('char') following date. The value of char may be a number representing a day or a character string. LAST_DAY(date): Finds the date of the last day of the month that contains date The above list is a subset of the available date functions. ROUND and TRUNC number functions can also be used to manipulate the date values as shown below:
ROUND(date[, 'fmt']): Returns date rounded to the unit that is specified by the format model fmt. If the format model fmt is omitted, date is rounded to the nearest day. TRUNC(date[, 'fmt']): Returns date with the time portion of the day truncated to the unit that is specified by the format model fmt. If the format model fmt is omitted, date is truncated to the nearest day.

The CONCAT Function
The CONCAT function joins two character literals, columns, or expressions to yield one larger character expression. Numeric and date literals are implicitly cast as characters when they occur as parameters to the CONCAT function. Numeric or date expressions are evaluated before being converted to strings ready to be concatenated. The CONCAT function takes two parameters. Its syntax is CONCAT(s1, s2), where s1 and s2 represent string literals, character column values, or expressions resulting in character values.
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
The TRIM function literally trims off leading or trailing (or both) character strings from a given source string:

QUESTION 51
Evaluate the following SQL statement:

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

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

A. It executes successfully and displays rows in the descending order of PROMO_CATEGORY.
B. It produces an error because positional notation cannot be used in the order by clause with set operators.
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 the order by clause should appear only at the end of a compound query-that is, with the last select statement.

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 52
View the Exhibit and examine the structure of the promotions table.

You need to generate a report of all promos from the promotions table based on the following conditions:
1. The promo name should not begin with 'T' or 'N'.
2. The promo should cost more than $20000.
3. The promo should have ended after 1st January 2001.

Which where clause would give the required result?

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

Correct Answer: WHERE promo_name NOT LIKE 'T%' AND promo_name NOT LIKE 'N%' AND promo_cost > 20000 AND promo_end_date > '1-JAN-01'
Section: (none)
Explanation

Explanation/Reference:

QUESTION 53
View the Exhibit for the structure of the student and faculty tables.

STUDENT

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

FACULTY

Name	Null?	Type
FACULTY_ID	NOT NULL	NUMBER(2)
FACULTY_NAME		VARCHAR2(20)
LOCATION ID		NUMBER(2)

You need to display the faculty name followed by the number of students handled by the faculty at the base location.

Examine the following two SQL statements:

Statement 1

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

Statement 2

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

Which statement is true regarding the outcome?

A. Only statement 1 executes successfully and gives the required result.
B. Only statement 2 executes successfully and gives the required result.
C. Both statements 1 and 2 execute successfully and give different results.
D. Both statements 1 and 2 execute successfully and give the same required result.

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:
Valid Answer.

QUESTION 54
View the Exhibit and examine the structure of the product, component, and PDT_COMP tables.
In product table, PDTNO is the primary key.

In component table, COMPNO is the primary key.
In PDT_COMP table, <PDTNO, COMPNO) is the primary key, PDTNO is the foreign key referencing PDTNO in product table and COMPNO is the foreign key referencing the COMPNO in component table.

You want to generate a report listing the product names and their corresponding component names, if the component names and product names exist.

Evaluate the following query:

SQL>SELECT pdtno, pdtname, compno, compname
FROM product _____ pdt_comp
USING (pdtno) _____ component USING (compno) WHERE compname IS NOT NULL;

Which combination of joins used in the blanks in the above query gives the correct output?

PRODUCT

Name	Null?	Type
PDTNO	NOT NULL	NUMBER(3)
PDTNAME		VARCHAR2(25)
QTY		NUMBER(6,2)

COMPONENT

Name	Null?	Type
COMPNO	NOT NULL	NUMBER(4)
COMPNAME		VARCHAR2(25)
QTY		NUMBER(6,2)

PDT_COMP

Name	Null?	Type
PDTNO	NOT NULL	NUMBER(2)
COMPNO	NOT NULL	NUMBER(3)

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

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 55
You want to create a table employees in which the values of columns EMPLOYEES_ID and LOGIN_ID must be unique and not null. Which two SQL statements would create the required table?

A)
```
CREATE TABLE employees(
employee_id NUMBER,
login_id NUMBER,
employee_name VARCHAR2(25),
hire_date DATE,
CONSTRAINT emp_id_pk PRIMARY KEY (employee_id, login_id));
```

B)
```
CREATE TABLE employees(
employee_id NUMBER CONSTRAINT emp_id_pk PRIMARY KEY,
login_id NUMBER UNIQUE,
employee_name VARCHAR2(25),
hire_date DATE);
```

C)
```
CREATE TABLE employees(
employee_id NUMBER,
login_id NUMBER,
employee_name VARCHAR2(100),
hire_date DATE,
CONSTRAINT emp_id_uk UNIQUE (employee_id, login_id));
```

D)
```
CREATE TABLE employees(
employee_id NUMBER,
login_id NUMBER,
employee_name VARCHAR2(100),
hire_date DATE,
CONSTRAINT emp_id_uk UNIQUE (employee_id, login_id),
CONSTRAINT emp_id_nn NOT NULL (employee_id, login_id));
```

D)
```
CREATE TABLE employees(
employee_id NUMBER,
login_id NUMBER,
employee_name VARCHAR2(100),
hire_date DATE,
CONSTRAINT emp_id_uk UNIQUE (employee_id, login_id),
CONSTRAINT emp_id_nn NOT NULL (employee_id, login_id));
```

E)
```
CREATE TABLE employees(
employee_id NUMBER CONSTRAINT emp_id_nn NOT NULL,
login_id NUMBER CONSTRAINT login_id_nn NOT NULL,
employee_name VARCHAR2(100),
hire_date DATE,
CONSTRAINT emp_num_id_uk UNIQUE (employee_id, login_id));
```

A. Option A

B. Option B
C. Option C
D. Option D
E. Option E
F. Option F

Correct Answer: DE
Section: (none)
Explanation

Explanation/Reference:

QUESTION 56
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.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:

QUESTION 57
Evaluate the following SQL commands:

```
SQL>CREATE SEQUENCE ord_seq
    INCREMENT BY 10
    START WITH 120
    MAXVALUE 9999
    NOCYCLE;

SQL>CREATE TABLE ord_items
    (ord_no NUMBER(4)DEFAULT ord_seq.NEXTVAL NOT NULL,
    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 command to create a table fails. Identify the two reasons for the SQL statement failure?

A. You cannot use SYSDATE in the condition of a check constraint.
B. You cannot use the BETWEEN clause in the condition of a check constraint.
C. You cannot use the NEXTVAL sequence value as a default value for a column.
D. You cannot use ORD_NO and ITEM_NO columns as a composite primary key because ORD_NO is also the foreign key.

Correct Answer: AC
Section: (none)
Explanation

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

(...
Salary NUMBER(8, 2) CONSTRAINT emp_salary_min
CHECK (salary > 0),

QUESTION 58
Examine the data in the ename and hiredate columns of the employees table:

```
EMPLOYEES

Name                 Null?     Type
-------------------- --------- --------------
EMPNO                NOT NULL  NUMBER(4)
ENAME                          VARCHAR2(10)
JOB                            VARCHAR2(9)
HIREDATE                       DATE
SAL                            NUMBER(7,2)
COMM                           NUMBER(7,2)
DEPTNO                         NUMBER(2)
```

```
ENAME      HIREDATE
---------- ---------
SMITH      17-DEC-80
ALLEN      20-FEB-81
WARD       22-FEB-81
```

You want to generate a list of user IDs as follows:

```
USERID
-----------
Smi17DEC80
All20FEB81
War22FEB81
```

You issue the following query:

```
SQL>SELECT CONCAT(SUBSTR(INITCAP(ename),1,3), REPLACE(hiredate,'-')
FROM employees;
```

What is the outcome?

A. It executes successfully and gives the correct output.
B. It executes successfully but does not give the correct output.
C. It generates an error because the REPLACE function is not valid.

D. It generates an error because the SUBSTR function cannot be nested in the CONCAT function.

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:
Explanation:
REPLACE (text, search_string, replacement_string)
Searches a text expression for a character string and, if found, replaces it with a specified replacement string
The REPLACE Function
The REPLACE function replaces all occurrences of a search item in a source string with a replacement term and returns the modified source string. If the length of the replacement term is different from that of the search item, then the lengths of the returned and source strings will be different. If the search string is not found, the source string is returned unchanged. Numeric and date literals and expressions are evaluated before being implicitly cast as characters when they occur as parameters to the REPLACE function.
The REPLACE function takes three parameters, with the first two being mandatory. Its syntax is REPLACE (source string, search item, [replacement term]). If the replacement term parameter is omitted, each occurrence of the search item is removed from the source string. In other words, the search item is replaced by an empty string. . The following queries illustrate the REPLACE function with numeric and date expressions:
Query 1: select replace(10000-3, '9', '85') from dual Query 2: select replace(sysdate, 'DEC', 'NOV') from dual

QUESTION 59
Examine the structure of the sales table:

```
Name                          Null?    Type
----------------------------- -------- --------------
PRODUCT_ID                    NOT NULL NUMBER(10)
CUSTOMER_ID                   NOT NULL NUMBER(10)
TIME_ID                       NOT NULL DATE
CHANNEL_ID                    NOT NULL NUMBER(5)
PROMO_ID                      NOT NULL NUMBER(5)
QUANTITY_SOLD                 NOT NULL NUMBER(10,2)
PRICE                                  NUMBER(10,2)
AMOUNT_SOLD                   NOT NULL NUMBER(10,2)
```

Evaluate the following create table statement:

```
SQL> CREATE TABLE sales1 (prod_id, cust_id, quantity_sold, price)
     AS
     SELECT product_id, customer_id, quantity_sold, price
     FROM sales
     WHERE 1=2;
```

Which two statements are true about the creation of the SALES1 table?

A. The SALES1 table is created with no rows but only a structure.
B. The SALES1 table would have primary key and unique constraints on the specified columns.
C. The SALES1 table would not be created because of the invalid where clause.
D. The SALES1 table would have not null and unique constraints on the specified columns.
E. The SALES1 table would not be created because column-specified names in the select and create table clauses do not match,

Correct Answer: AD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 60
View the Exhibit and examine the structure of the SALES table.

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)

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?

A. It executes successfully and generates the required result.
B. It produces an error because count(*) should be specified in the SELECT clause also.
C. It produces an error because count{*) should be only in the HAVING clause and not in the WHERE clause.
D. It executes successfully but produces no result because COUNT (prod_id) should be used instead of COUNT (*).

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:
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:
1. Rows are grouped.
2. The group function is applied to the group.
3. 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.

QUESTION 61
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_
FROM products);
```

B)
```
SELECT prod_name
FROM products
WHERE (unit_price * qty_in_hand) = (SELECT MAX(unit_price * qty_
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 *
FROM products
GROUP BY prod_name);
```

D)
```
SELECT prod_name
FROM products
WHERE (unit_price * qty_in_hand) = (SELECT MAX(SUM(unit_price *
FROM products)
GROUP BY prod_name;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:
Corrected,

QUESTION 62

Which two statements are true regarding subqueries?

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

Correct Answer: AD
Section: (none)
Explanation

Explanation/Reference:
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.

QUESTION 63
View the Exhibit and examine the data in the PROMO_NAME and PROMO_END_DATE columns of the promotions table, and the required output format.

```
PROMO_NAME                    PROMO_END_DATE
----------------------------- --------------------
post promotion #20-343        19-JUN-99
post promotion #20-274        16-JUL-99
TV promotion #12-530          13-APR-99
post promotion #17-157        29-JUN-99
TV promotion #12-481          05-JAN-00
newspaper promotion #19-4     16-AUG-98
everyday low price            01-JAN-99
```

```
OUTPUT
-------

PROMO_NAME                    LAST_DAY
----------------------------- ------------------------------
post promotion #20-343        Saturday, June 19, 1999
post promotion #20-274        Friday, July 16, 1999
TV promotion #12-530          Tuesday, April 13, 1999
post promotion #17-157        Tuesday, June 29, 1999
TV promotion #12-481          Wednesday, January 05, 2000
newspaper promotion #19-4     Sunday, August 16, 1998
everyday low price            Friday, January 01, 1999
```

Which two queries give the correct result?

A)
```
SELECT promo_name, TO_CHAR(promo_end_date,'Day') || ', ' ||
TO_CHAR(promo_end_date,'Month') || ' ' ||
TO_CHAR(promo_end_date,'DD, YYYY') AS last_day
FROM promotions;
```

B)
```
SELECT promo_name,TO_CHAR (promo_end_date,'fxDay') || ', ' ||
TO_CHAR(promo_end_date,'fxMonth') || ' ' ||
TO_CHAR(promo_end_date,'fxDD, YYYY') AS last_day
FROM promotions;
```

C)
```
SELECT promo_name, TRIM(TO_CHAR(promo_end_date,'Day')) || ', ' ||
TRIM(TO_CHAR(promo_end_date,'Month')) || ' ' ||
TRIM(TO_CHAR(promo_end_date,'DD, YYYY')) AS last_day
FROM promotions;
```

D)
```
SELECT promo_name,TO_CHAR(promo_end_date,'fmDay') || ', ' || TO_CHAR(promo_end_dat
TO_CHAR(promo_end_date,'fmDD, YYYY') AS last_day
FROM promotions;
```

A. Option A
B. Option B

C. Option C
D. Option D

Correct Answer: CD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 64
Which statement is true regarding the UNION operator?

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

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:
Explanation:
The SQL UNION query allows you to combine the result sets of two or more SQL SELECT statements. It removes duplicate rows between the various SELECT statements. Each SQL SELECT statement within the UNION query must have the same number of fields in the result sets with similar data types.

QUESTION 65
In which three situations does a transaction complete?

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

Correct Answer: BDE
Section: (none)
Explanation

Explanation/Reference:

QUESTION 66
Which two statements are true regarding constraints?

A. A table can have only one primary key and one foreign key.
B. A table can have only one primary key but multiple foreign keys.
C. Only the primary key can be defined at the column and table levels.
D. The foreign key and parent table primary key must have the same name.
E. Both primary key and foreign key constraints can be defined at both column and table levels.

Correct Answer: BE
Section: (none)

Explanation

Explanation/Reference:

QUESTION 67
You issue the following command to alter the country column in the departments table:

```
SQL> ALTER TABLE departments
     MODIFY (country DEFAULT 'USA');
```

Which statement is true?

A. It produces an error because column definitions cannot be altered to add default values.
B. It executes successfully and all the rows that have a null value for the country column will be updated with the value 'USA'.
C. It executes successfully. The modification to add the default value takes effect only from subsequent insertions to the table.
D. It produces an error because the data type for the column is not specified.

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:
Still Valid.

QUESTION 68
Examine the structure of the orders table:

```
Name            Null?     Type
 -------------- --------  -------------

 ORDER_ID       NOT NULL   NUMBER(12)
 ORDER_DATE     NOT NULL   TIMESTAMP(6)
 CUSTOMER_ID    NOT NULL   NUMBER(6)
 ORDER_STATUS              NUMBER(2)
 ORDER_TOTAL               NUMBER(8,2)
```

You want to find the total value of all the orders for each year and issue the following command:

```
SQL>SELECT TO_CHAR(order_date,'rr'), SUM(order_total)
FROM orders
GROUP BY TO_CHAR(order_date,'yyyy');
```

Which statement is true regarding the outcome?

A. It executes successfully and gives the correct output.
B. It gives an error because the TO_CHAR function is not valid.
C. It executes successfully but does not give the correct output.
D. It gives an error because the data type conversion in the SELECT list does not match the data type conversion in the GROUP BY clause.

Correct Answer: D
Section: (none)

Explanation

Explanation/Reference:

QUESTION 69
Which statement adds a column called salary to the employees table having 100 rows, which cannot contain null?

A)
```
ALTER TABLE EMPLOYEES
ADD SALARY NUMBER(8,2) NOT NULL;
```

B)
```
ALTER TABLE EMPLOYEES
ADD SALARY NUMBER(8,2) DEFAULT NOT NULL;
```

C)
```
ALTER TABLE EMPLOYEES
ADD SALARY NUMBER(8,2) DEFAULT 0 NOT NULL;
```

D)
```
ALTER TABLE EMPLOYEES
ADD SALARY NUMBER(8,2) DEFAULT CONSTRAINT p_nn NOT NULL;
```

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: A
Section: (none)
Explanation

Explanation/Reference:
Reference: http://www.comp.nus.edu.sg/~ooibc/courses/sql/ddl_table.htm (see changing table structures)

QUESTION 70
Which two statements are true regarding constraints?

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

Correct Answer: BD
Section: (none)
Explanation

Explanation/Reference:

QUESTION 71
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 table structure?

A. The CUST_STATUS column would give an error.
B. The TRANS_VALIDITY column would give an error.
C. The CUST_STATUS column would store exactly one character.
D. The CUST_CREDIT_LIMIT column would not be able to store decimal values.
E. The TRANS_VALIDITY column would have a maximum size of one character.
F. The TRANS_DATE column would be able to store day, month, century, year, hour, minutes, seconds, and fractions of seconds

Correct Answer: BC
Section: (none)
Explanation

Explanation/Reference:
Explanation:
VARCHAR2(size)Variable-length character data (A maximum size must be specified: minimum size is 1; maximum size is 4, 000.)
CHAR [(size)] Fixed-length character data of length size bytes (Default and minimum size is 1; maximum size is 2, 000.)
NUMBER [(p, s)] Number having precision p and scale s (Precision is the total number of decimal digits and scale is the number of digits to the right of the decimal point; precision can range from 1 to 38, and scale can range from 84 to 127.) DATE Date and time values to the nearest second between January 1, 4712 B.C., and December 31, 9999 A.D.

QUESTION 72
View the Exhibit and examine the structure of the 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)

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 falls below $10 after a discount of 25% is applied on it.

```
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 for the error?

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

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference:

QUESTION 73
You execute the following commands:

```
SQL> DEFINE hiredate = '01-APR-2011'

SQL> SELECT employee_id, first_name,salary
     FROM employees
     WHERE hire_date > '&hiredate'
     AND manager_id > &mgr_id;
```

For which substitution variables are you prompted for the input?

A. None, because no input required
B. Both the substitution variables 'hiredate' and 'mgr_id\
C. Only 'hiredate'
D. Only 'mgr_id'

Correct Answer: B
Section: (none)
Explanation

Explanation/Reference:

QUESTION 74
You need to create a table for a banking application. One of the columns in the table has the following requirements:

1) You want a column in the table to store the duration of the credit period.
2) 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 functions.
3) The maximum period of the credit provision in the application is 30 days.
4) 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. DATE
B. NUMBER
C. TIMESTAMP
D. INTERVAL DAY TO SECOND
E. INTERVAL YEAR TO MONTH

Correct Answer: D
Section: (none)
Explanation

Explanation/Reference:

QUESTION 75
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_MARITIAL_STATUS		VARCHAR2 (20)
CUST_STREET_ADDRESS	NOT NULL	VARCHAR2 (40)
CUST_POSTAL_CODE	NOT NULL	VARCHAR2 (10)
CUST_CITY	NOT NULL	VARCHAR2 (30)
CUST_STATE_PROVINCE	NOT NULL	VARCHAR2 (40)
COUNTRY_ID	NOT NULL	NUMBER
CUST_INCOME_LEVEL		VARCHAR2 (30)
CUST_CREDIT_LIMIT		NUMBER
CUST_EMAIL		VARCHAR2 (30)

Using the customers table, you need to generate a report that shows the average credit limit for customers in Washington and NEW YORK.

Which SQL statement would produce the required result?

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

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

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

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

A. Option A
B. Option B
C. Option C
D. Option D

Correct Answer: C
Section: (none)
Explanation

Explanation/Reference: