

ExamSheets.com

Leaders of the Pack

Exam Title: Introduction to Oracle9i: SQL

Exam Number: 1Z0-007

Version Dated: 05-02-2003

Note: The questions and answers in these ExamSheets.com products reflect the opinion of the exam experts that the material is current as of the version date. Even if the material has not changed but the date has been updated, it is because after careful research, the material was determined to be current and applicable to the exams.

Study this material in detail and entirety, understanding the answers and the questions, and be prepared fully before taking the final exam.

Good Luck from ExamSheets.com

Q1.

You need to give the MANAGER role the ability to select from, insert into, and modify existing rows in the STUDENT_GRADES table. Anyone given this MANAGER role should be able to pass those privileges on to others.

Which statement accomplishes this?

- A. GRANT select, insert, update
ON student_grades
TO manager;
- B. GRANT select, insert, update
ON student_grades
TO ROLE manager;
- C. GRANT select, insert, modify
ON student_grades
TO manager
WITH GRANT OPTION;
- D. GRANT select, insert, update
ON student_grades
TO manager
WITH GRANT OPTION;
- E. GRANT select, insert, update
ON student_grades
TO ROLE manager
WITH GRANT OPTION;
- F. F.GRANT select, insert, modify
ON student_grades
TO ROLE manager
WITH GRANT OPTION;

Answer: D.

Q2.

Examine the description of the EMPLOYEES table:

EMP_ID	NUMBER(4)	NOT NULL
LAST_NAME	VARCHAR2(30)	NOT NULL
FIRST_NAME	VARCHAR2(30)	
DEPT_ID	NUMBER(2)	
JOB_CAT	VARCHARD2(30)	
SALARY	NUMBER(8,2)	

Which statement shows the maximum salary paid in each job category of each department?

- A. SELECT dept_id, job_cat, MAX(salary)
FROM employees

```

WHERE salary > MAX(salary);
B. SELECT dept_id, job_cat, MAX(salary)
FROM employees
GROUP BY dept_id, job_cat;
C. SELECT dept_id, job_cat, MAX(salary)
FROM employees;
D. SELECT dept_id, job_cat, MAX(salary)
FROM employees
GROUP BY dept_id;
E. SELECT dept_id, job_cat, MAX(salary)
FROM employees
GROUP BY dept_id, job_cat, salary;

```

Answer: B.

Q3.

Examine the structure of the EMPLOYEES and DEPARTMENTS tables:

EMPLOYEES

Column name	Data type	Remarks
EMPLOYEE_ID	NUMBER	NOT NULL, Primary Key
EMP_NAME	VARCHAR2 (30)	
JOB_ID	VARCHAR2 (20)	
SALARY	NUMBER	
MGR_ID	NUMBER	References EMPLOYEE_ID COLUMN
DEPARTMENT ID	NUMBER	Foreign key to DEPARTMENT ID column of the DEPARTMENTS table

DEPARTMENTS

Column name	Data type	Remarks
DEPARTMENT_ID	NUMBER	NOT NULL, Primary Key
DEPARTMENT_NAME	VARCHAR2 (30)	
MGR_ID	NUMBER	References MGR_ID column of the EMPLOYEES table

Evaluate this SQL statement:

```

SELECT employee_id, e.department_id, department_name,
       salary
FROM employees e, departments d
WHERE e.department_id = d.department_id;

```

Which SQL statement is equivalent to the above SQL statement?

```

A. SELECT employee_id, department_id, department_name,
       salary
FROM employees
WHERE department_id IN (SELECT department_id
                       FROM departments);

```

B. SELECT employee_id, department_id, department_name,
salary
FROM employees
NATURAL JOIN departments;
C. SELECT employee_id, d.department_id, department_name,
salary
FROM employees e
JOIN departments d
ON e.department_id = d.department_id;
D. SELECT employee_id, department_id, department_name,
Salary
FROM employees
JOIN departments
USING (e.department_id, d.department_id);

Answer: C.

Q4.

Examine the SQL statements that creates ORDERS table:

```
CREATE TABLE orders
(SER_NO          NUMBER UNIQUE,
ORDER_ID         NUMBER,
ORDER_DATE       DATE NOT NULL
STATUS           VARCHAR2(10)
CHECK (status IN ('CREDIT','CASH')),
PROD_ID_NUMBER
REFERENCES PRODUCTS (PRODUCT_ID),
ORD_TOTAL NUMBER,
PRIMARY KEY (order id, order date));
```

For which columns would an index be automatically created when you execute the above SQL statement? (Choose two)

- A. SER_NO
- B. ORDER_ID
- C. STATUS
- D. PROD_ID
- E. ORD_TOTAL
- F. Composite index on ORDER_ID and ORDER_DATE

Answer: A, F.

Q5.

For which two constraints does the Oracle Server implicitly create a unique index? (Choose two.)

- A. NOT NULL
- B. PRIMARY KEY
- C. FOREIGN KEY
- D. CHECK
- E. UNIQUE

Answer: B, E.

Q6.

Examine the structure of the STUDENTS table:

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

You need to create a report of the 10 students who achieved the highest ranking in the course INT SQL and who completed the course in the year 1999.

Which SQL statement accomplishes this task?

- A.

```
SELECT student_id, marks, ROWNUM "Rank"
FROM students
WHERE ROWNUM <= 10
AND finish_date BETWEEN '01-JAN-99' AND '31-DEC-99'
AND course_id = 'INT_SQL'
ORDER BY marks DESC;
```
- B.

```
SELECT student_id, marks, ROWID "Rank"
FROM students
WHERE ROWID <= 10
AND finish_date BETWEEN '01-JAN-99' AND '31-DEC-99'
AND course_id = 'INT_SQL'
ORDER BY marks;
```
- C.

```
SELECT student_id, marks, ROWNUM "Rank"
FROM (SELECT student_id, marks
      FROM students
      WHERE ROWNUM <= 10
      AND finish_date BETWEEN '01-JAN-99' AND
                           '31-DEC-99'
      AND course_id = 'INT_SQL'
      ORDER BY marks DESC);
```
- D.

```
SELECT student_id, marks, ROWNUM "Rank:
FROM (SELECT student_id, marks
      FROM students
```

```

ORDER BY marks)
WHERE ROWNUM <= 10
AND finish_date BETWEEN '01-JAN-99' AND '31-DEC-99'
AND course_id = 'INT_SQL';

```

Answer: D.

Q7.

Examine the structure of the EMPLOYEES table:

Column name	Data type	Remarks
EMPLOYEE_ID	NUMBER	NOT NULL, Primary Key
LAST_NAME	VARCNAR2(30)	
FIRST_NAME	VARCNAR2(30)	
JOB_ID	NUMBER	
SAL	NUMBER	
MGR_ID	NUMBER	References EMPLOYEE_ID column
DEPARTMENT_ID	NUMBER	

You need to create an index called NAME_IDX on the first name and last name fields of the EMPLOYEES table. Which SQL statement would you use to perform this task?

- A. CREATE INDEX NAME_IDX (first_name, last_name);
- B. CREATE INDEX NAME_IDX (first_name AND last_name);
- C. CREATE INDEX NAME_IDX
ON (first_name, last_name);
- D. CREATE INDEX NAME_IDX
ON employees (first_name AND last_name);
- E. CREATE INDEX NAME_IDX
ON employees(first_name, last_name);
- F. CREATE INDEX NAME_IDX
FOR employees(first_name, last_name);

Answer: E.

Q8.

In which case would you use a FULL OUTER JOIN?

- A. Both tables have NULL values.
- B. You want all unmatched data from one table.
- C. You want all matched data from both tables.
- D. You want all unmatched data from both tables.
- E. One of the tables has more data than the other.
- F. You want all matched and unmatched data from only one table.

Answer: D.

Q9.

Which statement describes the ROWID data type?

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

Answer: E.

Q10.

Which two statements complete a transaction? (Choose two)

- A. DELETE employees;
- B. DESCRIBE employees;
- C. ROLLBACK TO SAVEPOINT C;
- D. GRANT SELECT ON employees TO SCOTT;
- E. ALTER TABLE employees
SET UNUSED COLUMN sal;
- F. Select MAX(sal)
FROM employees
WHERE department_id = 20;

Answer: C, E.

Q11.

Which substitution variable would you use if you want to reuse the variable without prompting the user each time?

- A. &
- B. ACCEPT
- C. PROMPT
- D. &&

Answer: D.

Q12.

In which scenario would TOP N analysis be the best solution?

- A. You want to identify the most senior employee in the company.
- B. You want to find the manager supervising the largest number of employees.
- C. You want to identify the person who makes the highest salary for all employees.
- D. You want to rank the top three sales representatives who have sold the maximum number of products.

Answer: D.

Q13.

You need to change the definition of an existing table. The COMMERCIALS table needs its DESCRIPTION column changed to hold varying length characters up to 2000 bytes. The column can currently hold 1000 bytes per value. The table contains 20000 rows. Which statement is valid?

- A. ALTER TABLE commercials
MODIFY (description CHAR2(2000));
- B. ALTER TABLE commercials
CHANGE (description CHAR2(2000));
- C. ALTER TABLE commercials
CHANGE (description VARCHAR2(2000));
- D. ALTER TABLE commercials
MODIFY (description VARCHAR2(2000));
- E. You cannot increase the size of a column if the table has rows.

Answer: D.

Q14.

Which SELECT statement should you use to extract the year from the system date and display it in the format "1998"?

- A. SELECT TO_CHAR(SYSDATE, 'yyyy')
FROM dual;
- B. SELECT TO_DATE(SYSDATE, 'yyyy')
FROM dual;
- C. SELECT DECODE(SUBSTR(SYSDATE, 8), 'YYYY')
FROM dual;
- D. SELECT DECODE(SUBSTR(SYSDATE, 8), 'year')
FROM dual;
- E. SELECT TO_CHAR(SUBSTR(SYSDATE, 8, 2), 'yyyy')
FROM dual;

Answer: A.

Q15.

Which clause should you use to exclude group results?

- A. WHERE
- B. HAVING
- C. RESTRICT
- D. GROUP BY
- E. ORDER BY

Answer: B.

Q16.

You need to calculate the total of all salaries in the accounting department. Which group function should you use?

- A. MAX
- B. MIN
- C. SUM
- D. COUNT
- E. TOTAL
- F. LARGEST

Answer: C.

Q17.

Which two are character manipulation functions? (Choose two.)

- A. TRIM
- B. REPLACE
- C. TRUNC
- D. TO_DATE
- E. MOD
- F. CASE

Answer: A, B.

Q18.

You need to perform certain data manipulation operations through a view called EMP_DEPT_VU, which you previously created.

You want to look at the definition of the view (the SELECT statement on which the view was create.) How do you obtain the definition of the view?

- A. Use the DESCRIBE command in the EMP_DEPT_VU view.
- B. Use the DEFINE VIEW command on the EMP_DEPT_VU view.
- C. Use the DESCRIBE VIEW command on the EMP_DEPT_VU view.
- D. Query the USER_VIEWS data dictionary view to search for the EMP_DEPT_VU view.
- E. Query the USER_SOURCE data dictionary view to search for the EMP_DEPT_VU view.
- F. Query the USER_OBJECTS data dictionary view to search for the EMP_DEPT_VU view.

Answer: D.

Q19.

Examine the structure of the EMPLOYEES, DEPARTMENTS, and TAX tables.

EMPLOYEES

EMPLOYEE_ID	NUMBER	NOT NULL, Primary Key
EMP_NAME	VARCHAR2 (30)	
JOB_ID	VARCHAR2 (20)	
SALARY	NUMBER	
MGR_ID	NUMBER	References EMPLOYEE_ID column
DEPARTMENT_ID	NUMBER	Foreign key to DEPARTMENT_ID column of the DEPARTMENTS table

DEPARTMENTS

DEPARTMENT_ID	NUMBER	NOT NULL, Primary Key
DEPARTMENT_NAME	VARCHAR2 (30)	
MGR_ID	NUMBER	References MGR_ID column of the EMPLOYEES table

TAX

MIN_SALARY	NUMBER	
MAX_SALARY	NUMBER	
TAX_PERCENT	NUMBER	

For which situation would you use a nonequijoin query?

- A. To find the tax percentage for each of the employees.
- B. To list the name, job id, and manager name for all the employees.
- C. To find the name, salary, and department name of employees who are not working with Smith.
- D. To find the number of employees working for the Administrative department and earning less than 4000.

- E. To display name, salary, manager ID, and department name of all the employees, even if the employees do not have a department ID assigned.

Answer: A.

Q20.

Which four are correct guidelines for naming database tables? (Choose four)

- A. Must begin with either a number or a letter.
- B. Must be 1-30 characters long.
- C. Should not be an Oracle Server reserved word.
- D. Must contain only A-Z, a-z, 0-9, _, *, and #.
- E. Must contain only A-Z, a-z, 0-9, _, \$, and #.
- F. Must begin with a letter.

Answer: B, C, E, F.

Q21.

Which three statements about subqueries are true? (Choose three)

- A. A single row subquery can retrieve only one column and one row.
- B. A single row subquery can retrieve only one row but many columns.
- C. A multiple row subquery can retrieve multiple rows and multiple columns.
- D. A multiple row subquery can be compared by using the ">" operator.
- E. A single row subquery can use the IN operator.
- F. A multiple row subquery can use the "=" operator.

Answer: B, C, D.

Q22.

Which SQL statement generates the alias Annual Salary for the calculated column SALARY*12?

- A. `SELECT ename, salary*12 'Annual Salary'`
`FROM employees;`
- B. `SELECT ename, salary*12 "Annual Salary"`
`FROM employees;`
- C. `SELECT ename, salary*12 AS Annual Salary`
`FROM employees;`
- D. `SELECT ename, salary*12 AS INITCAP("ANNUAL SALARY")`
`FROM employees`

Answer: B.

Q23.

Examine the structure of the EMPLOYEES table:

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

You issue these statements:

```
CREATE table new_emp ( employee_id NUMBER, name VARCHAR2(30));
INSERT INTO new_emp SELECT employee_id , last_name from employees;
Savepoint s1;
UPDATE new_emp set name = UPPER(name);
Savepoint s2;
Delete from new_emp;
Rollback to s2;
Delete from new_emp where employee_id =180;
UPDATE new_emp set name = 'James';
Rollback to s2;
UPDATE new_emp set name = 'James' WHERE employee_id =180;
Rollback;
```

At the end of this transaction, what is true?

- A. You have no rows in the table.
- B. You have an employee with the name of James.
- C. You cannot roll back to the same savepoint more than once.
- D. Your last update fails to update any rows because employee ID 180 was already deleted.

Answer: A.

Q24.

Which are DML statements? (Choose all that apply)

- A. COMMIT...
- B. MERGE...
- C. UPDATE...
- D. DELETE...
- E. CREATE...
- F. DROP...

Answer: B, C, D.

Q25.

Which two tasks can you perform by using the TO_CHAR function? (Choose two)

- A. Convert 10 to 'TEN'
- B. Convert '10' to 10
- C. Convert '10' to '10'
- D. Convert 'TEN' to 10
- E. Convert a date to a character expression
- F. Convert a character expression to a date

Answer: C, E.

Q26.

Examine the data from the EMP table:

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

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

Which three tasks would require subqueries or joins in order to perform in a single step? (Choose three)

- A. Deleting the records of employees who do not earn commission.
- B. Increasing the commission of employee 3 by the average commission earned in department 20.
- C. Finding the number of employees who do NOT earn commission and are working for department 20.
- D. Inserting into the table a new employee 10 who works for department 20 and earns a commission that is equal to the commission earned by employee 3.
- E. Creating a table called COMMISSION that has the same structure and data as the columns EMP_ID and COMMISSIONS of the EMP table.
- F. Decreasing the commission by 150 for the employees who are working in department 30 and earning a commission of more than 800.

Answer: B, D, F.

Q27.

You need to create a view EMP_VU. The view should allow the users to manipulate the records of only the employees that are working for departments 10 or 20.
Which SQL statement would you use to create the view EMP_VU?

- A.

```
CREATE VIEW emp_vu AS
  SELECT *
  FROM employees
  WHERE department_id IN (10,20);
```
- B.

```
CREATE VIEW emp_vu AS
  SELECT *
  FROM employees
  WHERE department_id IN (10,20)
  WITH READ ONLY;
```
- C.

```
CREATE VIEW emp_vu AS
  SELECT *
  FROM employees
  WHERE department_id IN (10,20)
  WITH CHECK OPTION;
```
- D.

```
CREATE FORCE VIEW emp_vu AS
  SELECT *
  FROM employees
  WHERE department_id IN (10,20);
```
- E.

```
CREATE FORCE VIEW emp_vu AS
  SELECT *
  FROM employees
  WHERE department_id IN (10,20)
  NO UPDATE;
```

Answer: C.

Q28.

Evaluate this SQL statement:

```
SELECT e.employee_id, (.15* e.salary) + (.5 * e.commission_pct)
      + (s.sales amount * (.35 * e.bonus)) AS CALC_VALUE
FROM employees e, sales s
WHERE e.employee_id = s.emp_id;
```

What will happen if you remove all the parentheses from the calculation?

- A. The value displayed in the CALC_VALUE column will be lower.
- B. The value displayed in the CALC_VALUE column will be higher.
- C. There will be no difference in the value displayed in the CALC_VALUE column.

D. An error will be reported.

Answer: C.

Q29.

You would like to display the system date in the format "Monday, 01 June, 2001". Which SELECT statement should you use?

- A. SELECT TO_DATE(SYSDATE, 'FMDAY, DD Month, YYYY')
FROM dual;
- B. SELECT TO_CHAR(SYSDATE, 'FMDD, DY Month, 'YYY')
FROM dual;
- C. SELECT TO_CHAR(SYSDATE, 'FMDay, DD Month, YYYY')
FROM dual;
- D. SELECT TO_CHAR(SYSDATE, 'FMDY, DDD Month, YYYY')
FROM dual;
- E. SELECT TO_DATE(SYSDATE, 'FMDY, DDD Month, YYYY')
FROM dual;

Answer: C.

Q30.

The EMP table has these columns:

ENAME	VARCHAR2 (35)
SALARY	NUMBER (8, 2)
HIRE_DATE	DATE

Management wants a list of names of employees who have been with the company for more than five years. Which SQL statement displays the required results?

- A. SELECT ENAME
FROM EMP
WHERE SYSDATE-HIRE_DATE > 5;
- B. SELECT ENAME
FROM EMP
WHERE HIRE_DATE-SYSDATE > 5;
- C. SELECT ENAME
FROM EMP
WHERE (SYSDATE-HIRE_DATE)/365 > 5;
- D. SELECT ENAME
FROM EMP
WHERE (SYSDATE-HIRE_DATE)* 365 > 5;

Answer: C.

Q31.

You added a PHONE_NUMBER column of NUMBER data type to an existing EMPLOYEES table. The EMPLOYEES table already contains records of 100 employees. Now, you want to enter the phone numbers of each of the 100 employees into the table. Some of the employees may not have a phone number available. Which data manipulation operation do you perform?

- A. MERGE
- B. INSERT
- C. UPDATE
- D. ADD
- E. ENTER
- F. You cannot enter the phone numbers for the existing employee records.

Answer: C.

Q32.

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

- A. A single row subquery can retrieve data from only one table.
- B. A SQL query statement cannot display data from table B that is referred to in its subquery, unless table B is included in the main query's FROM clause.
- C. A SQL query statement can display data from table B that is referred to in its subquery, without including table B in its own FROM clause.
- D. A single row subquery can retrieve data from more than one table.
- E. A single row subquery cannot be used in a condition where the LIKE operator is used for comparison.
- F. A multiple-row subquery cannot be used in a condition where the LIKE operator is used for comparison.

Answer: B, D.

Q33.

The STUDENT_GRADES table has these columns:

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


The registrar has requested a report listing the students' grade point averages (GPA), sorted from highest grade point average to lowest within each semester, starting from the earliest date. Which statement accomplishes this?

- A. `SELECT student_id, semester_end, gpa
FROM student_grades
ORDER BY semester_end DESC, gpa DESC;`
- B. `SELECT student_id, semester_end, gpa
FROM student_grades
ORDER BY semester_end ASC, gpa ASC;`
- C. `SELECT student_id, semester_end, gpa
FROM student_grades
ORDER BY semester_end, gpa DESC;`
- D. `SELECT student_id, semester_end, gpa
FROM student_grades
ORDER BY gpa DESC, semester_end DESC;`
- E. `SELECT student_id, semester_end, gpa
FROM student_grades
ORDER BY gpa DESC, semester_end ASC;`

Answer: C.

Q34.

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

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

Answer: C, D.

Q35.

The STUDENT_GRADES table has these columns:

STUDENT_ID	NUMBER(12)
SEMESTER_END	DATE

GPA NUMBER (4,3)

The registrar has asked for a report on the average grade point average (GPA) for students enrolled during semesters that end in the year 2000. Which statement accomplish this?

- A. SELECT AVERAGE(gpa)
FROM student_grades
WHERE semester_end > '01-JAN-2000' and semester end < 31-DEC-2000';
- B. SELECT COUNT(gpa)
FROM student_grades
WHERE semester_end > '01-JAN-2000' and semester end < '31-DEC-2000';
- C. SELECT MIN(gpa)
FROM student_grades
WHERE semester_end > '01-JAN-2000' and semester end < '31-DEC-2000';
- D. SELECT AVG(gpa)
FROM student_grades
WHERE semester_end BETWEEN '01-JAN-2000' and '31-DEC-2000';
- E. SELECT SUM(gpa)
FROM student_grades
WHERE semester_end > '01-JAN-2000' and semester end < '31-DEC-2000';
- F. SELECT MEDIAN(gpa)
FROM student_grades
WHERE semester end > '01-JAN-2000' and semester end < '31-DEC-2000';

Answer: D.

Q36.

Examine the structure of the EMPLOYEES and NEW_EMPLOYEES tables:

<u>EMPLOYEES</u>		
EMPLOYEE_ID	NUMBER	Primary Key
FIRST_NAME	VARCHAR2(25)	
LAST_NAME	VARCHAR2(25)	
HIRE_DATE	DATE	
<u>NEW_EMPLOYEES</u>		
EMPLOYEE_ID	NUMBER	Primary Key
NAME	VARCHAR2(60)	

Which UPDATE statement is valid?

- A. UPDATE new_employees SET name = (Select last_name||
first_name

```

FROM employees
Where employee_id
=180)
WHERE employee_id =180;
B. UPDATE new_employees SET name = (SELECT
    last_name||first_name
    FROM employees)
WHERE employee_id =180;
C. UPDATE new_employees SET name = (SELECT last_name||
    first_name
    FROM employees
    WHERE employee_id
    =180)
WHERE employee_id =(SELECT employee_id
    FROM new_employees);
D. UPDATE new_employees SET name = (SELECT last name||
    first_name
    FROM employees
    WHERE employee_id=
    (SELECT employee_id
    FROM new_employees))
    WHERE employee_id
    =180;

```

Answer: A.

Q37.

You created a view called EMP_DEPT_VU that contains three columns from the EMPLOYEES and DEPARTMENTS tables:

EMPLOYEE_ID, EMPLOYEE_NAME AND DEPARTMENT_NAME.

The DEPARTMENT_ID column of the EMPLOYEES table is the foreign key to the primary key DEPARTMENT_ID column of the DEPARTMENTS table.

You want to modify the view by adding a fourth column, MANAGER_ID of NUMBER data type from the EMPLOYEES tables.

How can you accomplish this task?

- A. ALTER VIEW emp_dept_vu (ADD manager_id NUMBER);
- B. MODIFY VIEW emp_dept_vu (ADD manager_id NUMBER);
- C. ALTER VIEW emp_dept_vu AS
 SELECT employee_id, employee_name,
 department_name, manager_id
 FROM employee e, departments d
 WHERE e.department_id = d.department_id;
- D. MODIFY VIEW emp_dept_vu AS
 SELECT employee_id, employee_name,
 department_name, manager_id
 FROM employees e, departments d

- ```
WHERE e.department_id = d.department_id;
```
- E. CREATE OR REPLACE VIEW emp\_dept\_vu AS  
SELECT employee\_id, employee\_name,  
department\_name, manager\_id  
FROM employees e, departments d  
WHERE e.department\_id = d.department\_id;
- F. You must remove the existing view first, and then run the CREATE VIEW command with a new column list to modify a view.

**Answer: E.**

**Q38.**

You want to display the titles of books that meet these criteria:

1. Purchased before January 21, 2001
2. Price is less than \$500 or greater than \$900

You want to sort the results by their data of purchase, starting with the most recently bought book. Which statement should you use?

- A. SELECT book\_title  
FROM books  
WHERE price between 500 and 900  
AND purchase\_date < '21-JAN-2001'  
ORDER BY purchase\_date;
- B. SELECT book\_title  
FROM books  
WHERE price IN (500,900)  
AND purchase\_date < '21-JAN-2001'  
ORDER BY purchase date ASC;
- C. SELECT book\_title  
FROM books  
WHERE price < 500 or > 900  
AND purchase\_date < '21-JAN-2001'  
ORDER BY purchase date DESC;
- D. SELECT book\_title  
FROM books  
WHERE (price < 500 OR price > 900)  
AND purchase\_date < '21-JAN-2001'  
ORDER BY purchase date DESC;

**Answer: D.**

**Q39.**

Evaluate the SQL statement:

```
SELECT ROUND (TRUNC (MOD (1600,10) ,-1) ,2)
FROM dual;
```

What will be displayed?

- A. 0
- B. 1
- C. 0.00
- D. An error statement

**Answer: A.**

**Q40.**

Which data dictionary table should you query to view the object privileges granted to the user on specific columns?

- A. USER\_TAB\_PRIVS\_MADE
- B. USER\_TAB\_PRIVS\_RECD
- C. USER\_COL\_PRIVS\_MADE
- D. USER\_COL\_PRIVS\_RECD

**Answer: D.**

**Q41.**

Which three are DATETIME data types that can be used when specifying column definitions? (Choose three.)

- A. TIMESTAMP
- B. INTERVAL MONTH TO DAY
- C. INTERVAL DAY TO SECOND
- D. INTERVAL YEAR TO MONTH
- E. TIMESTAMP WITH DATABASE TIMEZONE

**Answer: A, C, D.**

**Q42.**

Evaluate the SQL statement:

```
1 SELECT a.emp_name, a.sal, a.dept_id, b.maxsal
```

```
2 FROM employees a,
3 (SELECT dept_id, MAX(sal) maxsal
4. FROM employees
5 GROUP BY dept_id) b
6 WHERE a.dept_id = b.dept_id
7 AND a.sal < b.maxsal;
```

What is the result of the statement?

- A. The statement produces an error at line 1.
- B. The statement produces an error at line 3.
- C. The statement produces an error at line 6.
- D. The statement returns the employee name, salary, department ID, and maximum salary earned in the department of the employee for all departments that pay less salary then the maximum salary paid in the company.
- E. The statement returns the employee name, salary, department ID, and maximum salary earned in the department of the employee for all employees who earn less than the maximum salary in their department.

**Answer: E.**

**Q43.**

The EMPLOYEE tables has these columns:

|                |               |
|----------------|---------------|
| LAST_NAME      | VARCHAR2 (35) |
| SALARY         | NUMBER (8,2)  |
| COMMISSION_PCT | NUMBER (5,2)  |

You want to display the name and annual salary multiplied by the commission\_pct for all employees. For records that have a NULL commission\_pct, a zero must be displayed against the calculated column.

Which SQL statement displays the desired results?

- A. `SELECT last_name, (salary * 12) * commission_pct  
FROM EMPLOYEES;`
- B. `SELECT last_name, (salary * 12) * IFNULL(commission_pct, 0)  
FROM EMPLOYEES;`
- C. `SELECT last_name, (salary * 12) * NVL2(commission_pct, 0)  
FROM EMPLOYEES;`
- D. `SELECT last_name, (salary * 12) * NVL(commission_pct, 0)  
FROM EMPLOYEES;`

**Answer: D.**

**Q44.**

Management has asked you to calculate the value  $12 * salary * commission\_pct$  for all the employees in the EMP table. The EMP table contains these columns:

|                |                        |
|----------------|------------------------|
| LAST_NAME      | VARCNAR2 (35) NOT NULL |
| SALARY         | NUMBER (9, 2) NOT NULL |
| COMMISSION_PCT | NUMBER (4, 2)          |

Which statement ensures that a value is displayed in the calculated columns for all employees?

- A. `SELECT last_name, 12*salary*commison_pct  
FROM emp;`
- B. `SELECT last_name, 12*salary* (commission_pct,0)  
FROM emp;`
- C. `SELECT last_name, 12*salary*(nvl(commission_pct,0))  
FROM emp;`
- D. `SELECT last_name, 12*salary*(decode(commission_pct,0))  
FROM emp;`

**Answer: C.**

**Q45.**

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

- A. A view can be created as read only.
- B. A view can be created as a join on two or more tables.
- C. A view cannot have an ORDER BY clause in the SELECT statement.
- D. A view cannot be created with a GROUP BY clause in the SELECT statement.
- E. A view must have aliases defined for the column names in the SELECT statement.

**Answer: A, B.**

**Q46.**

Examine the data in the EMPLOYEES table:

| LAST_NAME | DEPARTMENT_ID | SALARY |
|-----------|---------------|--------|
| Getz      | 10            | 3000   |
| Davis     | 20            | 1500   |
| King      | 20            | 2200   |
| Davis     | 30            | 5000   |
| ...       |               |        |

Which three subqueries work? (Choose three)

A. SELECT \*  
FROM employees  
where salary > (SELECT MIN(salary)  
FROM employees  
GROUP BY department.id);

B. SELECT \*  
FROM employees  
WHERE salary = (SELECT AVG(salary)  
FROM employees  
GROUP BY department\_id);

C. SELECT distinct department\_id  
FROM employees  
Where salary > ANY (SELECT AVG(salary)  
FROM employees  
GROUP BY department\_id);

D. SELECT department\_id  
FROM employees  
WHERE SALARY > ALL (SELECT AVG(salary)  
FROM employees  
GROUP BY department\_id);

E. SELECT last\_name  
FROM employees  
Where salary > ANY (SELECT MAX(salary)  
FROM employees  
GROUP BY department\_id);

F. SELECT department\_id  
FROM employees  
WHERE salary > ALL (SELECT AVG(salary)  
FROM employees  
GROUP BY AVG(SALARY));

**Answer: C, D, E.**

**Q47.**

Examine the data in the EMPLOYEES and DEPARTMENTS tables.

**EMPLOYEES**

| LAST NAME | DEPARTMENT ID | SALARY |
|-----------|---------------|--------|
| Getz      | 10            | 3000   |
| Davis     | 20            | 1500   |
| King      | 20            | 2200   |
| Davis     | 30            | 5000   |
| Kochhar   |               | 5000   |

**DEPARTMENTS**

| DEPARTMENT ID | DEPARTMENT NAME |
|---------------|-----------------|
| 10            | Sales           |



|    |                |
|----|----------------|
| 20 | Marketing      |
| 30 | Accounts       |
| 40 | Administration |

You want to retrieve all employees, whether or not they have matching departments in the departments table. Which query would you use?

- A. `SELECT last_name, department_name  
FROM employees , departments(+);`
- B. `SELECT last_name, department_name  
FROM employees JOIN departments (+);`
- C. `SELECT last_name, department_name  
FROM employees(+) e JOIN departments d  
ON (e.department_id = d.department_id);`
- D. `SELECT last_name, department_name  
FROM employees e  
RIGHT OUTER JOIN departments d ON (e.department_id =  
d.department_id);`
- E. `SELECT last_name, department_name  
FROM employees(+) , departments  
ON (e.department_id = d.department_id);`
- F. `SELECT last_name, department_name  
FROM employees e LEFT OUTER  
JOIN departments d ON (e.department_id = d.department_id);`

**Answer: F.**

#### Q48.

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

- A. Constraint names must start with SYS\_C.
- B. All constraints must be defines at the column level.
- C. Constraints can be created after the table is created.
- D. Constraints can be created at the same time the table is created.
- E. Information about constraints is found in the VIEW\_CONSTRAINTS dictionary view.

**Answer: C, D.**

#### Q49.

Evaluate the SQL statement

```
DROP TABLE DEPT;
```

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

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

**Answer: A, B, D, E.**

**Q50.**

You need to produce a report for mailing labels for all customers. The mailing label must have only the customer name and address. The CUSTOMERS table has these columns:

|              |               |          |
|--------------|---------------|----------|
| CUST_ID      | NUMBER(4)     | NOT NULL |
| CUST_NAME    | VARCHAR2(100) |          |
| CUST_ADDRESS | VARCHAR2(150) |          |
| CUST_PHONE   | VARCHAR2(20)  |          |

Which SELECT statement accomplishes this task?

- A. `SELECT*`  
`FROM customers;`
- B. `SELECT name, address`  
`FROM customers;`
- C. `SELECT id, name, address, phone`  
`FROM customers;`
- D. `SELECT cust_name, cust_address`  
`FROM customers;`
- E. `SELECT cust_id, cust_name, cust_address, cust_phone`  
`FROM customers;`

**Answer: D.**

**Q51.**

The DBA issues this SQL command:

```
CREATE USER scott
IDENTIFIED BY tiger;
```

What privileges does the user Scott have at this point?

- A. No privileges.

- B. Only the SELECT privilege.
- C. Only the CONNECT privilege.
- D. All the privileges of a default user.

**Answer: A.**

**Q52.**

Which constraint can be defines only at the column level?

- A. UNIQUE
- B. NOT NULL
- C. CHECK
- D. PRIMARY KEY
- E. FOREIGN KEY

**Answer: B.**

**Q53.**

Which statement creates a new user?

- A. CREATE USER susan;
- B. CREATE OR REPLACE USER susan;
- C. CREATE NEW USER susan  
DEFAULT;
- D. CREATE USER susan  
IDENTIFIED BY blue;
- E. CREATE NEW USER susan  
IDENTIFIED by blue;
- F. CREATE OR REPLACE USER susan  
IDENTIFIED BY blue;

**Answer: D.**

**Q54.**

Which is an /SQL\*Plus command?

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

F. RENAME

**Answer: D.**

**Q55.**

The EMPLOYEES table contains these columns:

|                |               |
|----------------|---------------|
| LAST_NAME      | VARCHAR2 (25) |
| SALARY         | NUMBER (6,2)  |
| COMMISSION_PCT | NUMBER (6)    |

You need to write a query that will produce these results:

1. Display the salary multiplied by the commission\_pct.
2. Exclude employees with a zero commission\_pct.
3. Display a zero for employees with a null commission value.

Evaluate the SQL statement:

```
SELECT LAST_NAME, SALARY*COMMISSION_PCT
FROM EMPLOYEES
WHERE COMMISSION_PCT IS NOT NULL;
```

What does the statement provide?

- A. All of the desired results
- B. Two of the desired results
- C. One of the desired results
- D. An error statement

**Answer: C.**

**Q56.**

Scott issues the SQL statements:

```
CREATE TABLE dept
(deptno NUMBER(2),
dname VARCHAR2(14),
loc VARCHAR2(13));
GRANT SELECT
ON DEPT
TO SUE;
```

If Sue needs to select from Scott's DEPT table, which command should she use?

- A. `SELECT *  
FROM DEPT;`
- B. `SELECT *  
FROM SCOTT.DEPT;`
- C. `SELECT *  
FROM DBA.SCOTT.DEPT;`
- D. `SELECT *  
FROM ALL_USERS  
WHERE USER_NAME = 'SCOTT'  
AND TABLE_NAME = 'DEPT';`

**Answer: B.**

**Q57.**

You define a multiple-row subquery in the WHERE clause of an SQL query with a comparison operator "=".

What happens when the main query is executed?

- A. The main query executes with the first value returned by the subquery.
- B. The main query executes with the last value returned by the subquery.
- C. The main query executes with all the values returned by the subquery.
- D. The main query fails because the multiple-row subquery cannot be used with the comparison operator
- E. You cannot define a multiple-row subquery in the WHERE clause of a SQL query.

**Answer: D.**

**Q58.**

You need to display the last names of those employees who have the letter "A" as the second character in their names.

Which SQL statement displays the required results?

- A. `SELECT last_name  
FROM EMP  
WHERE last_name LIKE '_A%';`
- B. `SELECT last_name  
FROM EMP  
WHERE last name = '*A%'`
- C. `SELECT last_name  
FROM EMP  
WHERE last name = '_A%';`
- D. `SELECT last_name`

```
FROM EMP
WHERE last name LIKE '*A%'
```

**Answer: A.**

**Q59.**

Which operator can be used with a multiple-row subquery?

- A. =
- B. LIKE
- C. BETWEEN
- D. NOT IN
- E. IS
- F. <>

**Answer: D.**

**Q60.**

Which two statements about sequences are true? (Choose two)

- A. You use a NEXTVAL pseudo column to look at the next possible value that would be generated from a sequence, without actually retrieving the value.
- B. You use a CURRVAL pseudo column to look at the current value just generated from a sequence, without affecting the further values to be generated from the sequence.
- C. You use a NEXTVAL pseudo column to obtain the next possible value from a sequence by actually retrieving the value from the sequence.
- D. You use a CURRVAL pseudo column to generate a value from a sequence that would be used for a specified database column.
- E. If a sequence starting from a value 100 and incremented by 1 is used by more than one application, then all of these applications could have a value of 105 assigned to their column whose value is being generated by the sequence.
- F. You use REUSE clause when creating a sequence to restart the sequence once it generates the maximum value defined for the sequence.

**Answer: B, C.**

**Q61.**

When should you create a role? (Choose two)

- A. To simplify the process of creating new users using the CREATE USER xxx IDENTIFIED by yyy statement.

- B. To grant a group of relate privileges to a user.
- C. When the number of people using the database is very high.
- D. To simplify the process of granting and revoking privileges.
- E. To simplify profile maintenance for a user who is constantly traveling.

**Answer: B, D.**

**Q62.**

In which scenario would an index be most useful?

- A. The indexed column is declared as NOT NULL.
- B. The indexed columns are used in the FROM clause.
- C. The indexed columns are part of an expression.
- D. The indexed column contains a wide range of values.

**Answer: D.**

**Q63.**

Mary has a view called EMP\_DEPT\_LOC\_VU that was created based on the EMPLOYEES, DEPARTMENTS, and LOCATIONS tables. She granted SELECT privilege to Scott on this view. Which option enables Scott to eliminate the need to qualify the view with the name MARY.EMP\_DEP\_LOC\_VU each time the view is referenced?

- A. Scott can create a synonym for the EMP\_DEPT\_LOC\_VU bus using the command:  
`CREATE PRIVATE SYNONYM EDL_VU  
FOR mary.EMP DEPT_LOC_VU;`  
then he can prefix the columns with this synonym.
- B. Scott can create a synonym for the EMP\_DEPT\_LOC\_VU by using the command:  
`CREATE SYNONYM EDL_VU  
FOR mary.EMP_DEPT_LOC_VU;`  
then he can prefix the columns with this synonym.
- C. Scott can create a synonym for the EMP\_DEPT\_LOC\_VU by using the command:  
`CREATE LOCAL SYNONYM EDL_VU  
FOR mary.EMP DEPT_LOC_VU;`  
then he can prefix the columns with this synonym.
- D. Scott can create a synonym for the EMP\_DEPT\_LOC\_VU by using the command:  
`CREATE SYNONYM EDL_VU  
ON mary(EMP_DEPT_LOC_VU);`  
then he can prefix the columns with this synonym.
- E. Scott cannot create a synonym because synonyms can be created only for tables.
- F. Scott cannot create any synonym for Mary's view. Mary should create a private synonym for the view and grant SELECT privilege on that synonym to Scott.

**Answer: B.**

**Q64.**

Which two statements are true regarding the ORDER BY clause? (Choose two)

- A. The sort is in ascending by order by default.
- B. The sort is in descending order by default.
- C. The ORDER BY clause must precede the WHERE clause.
- D. The ORDER BY clause is executed on the client side.
- E. The ORDER BY clause comes last in the SELECT statement.
- F. The ORDER BY clause is executed first in the query execution.

**Answer: A, E.**

**Q65.**

The CUSTOMERS table has these columns:

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

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

Which two statements find the number of customers? (Choose two.)

- A. `SELECT TOTAL(*)`  
`FROM customers;`
- B. `SELECT COUNT(*)`  
`FROM customers;`
- C. `SELECT TOTAL(customer_id)`  
`FROM customers;`
- D. `SELECT COUNT(customer_id)`  
`FROM customers;`
- E. `SELECT COUNT(customers)`  
`FROM customers;`
- F. `SELECT TOTAL(customer_name)`  
`FROM customers;`

**Answer: B, D.**



**Q66.**

From SQL\*Plus, you issue this SELECT statement:

```
SELECT*
From orders;
```

You use this statement to retrieve data from a data table for \_\_\_\_\_. (Choose all that apply)

- A. Updating
- B. Viewing
- C. Deleting
- D. Inserting
- E. Truncating

**Answer: B, D.**

**Q67.**

Evaluate these two SQL statements:

```
SELECT last_name, salary , hire_date
FROM EMPLOYEES
ORDER BY salary DESC;
SELECT last_name, salary, hire_date
FROM EMPLOYEES
ORDER BY 2 DESC;
```

What is true about them?

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

**Answer: A.**

**Q68.**

Which two statements accurately describe a role? (Choose two.)

- A. A role can be given to a maximum of 1000 users.

- B. A user can have access to a maximum of 10 roles.
- C. A role can have a maximum of 100 privileges contained in it.
- D. Privileges are given to a role by using the CREATE ROLE statement.
- E. A role is a named group of related privileges that can be granted to the user.
- F. A user can have access to several roles, and several users can be assigned the same role.

**Answer: E, F.**

**Q69.**

The ORDERS table has these columns:

|             |              |          |
|-------------|--------------|----------|
| ORDER_ID    | NUMBER(4)    | NOT NULL |
| CUSTOMER_ID | NUMBER(12)   | NOT NULL |
| ORDER_TOTAL | NUMBER(10,2) |          |

The ORDERS table tracks the Order number, the order total, and the customer to whom the Order belongs. Which two statements retrieve orders with an inclusive total that ranges between 100.00 and 2000.00 dollars? (Choose two.)

- A. `SELECT customer_id, order_id, order_total  
FROM orders  
RANGE ON order_total (100 AND 2000) INCLUSIVE;`
- B. `SELECT customer_id, order_id, order_total  
FROM orders  
HAVING order_total BETWEEN 100 and 2000;`
- C. `SELECT customer_id, order_id, order_total  
FROM orders  
WHERE order_total BETWEEN 100 and 2000;`
- D. `SELECT customer_id, order_id, order_total  
FROM orders  
WHERE order_total >= 100 and <= 2000;`
- E. `SELECT customer_id, order_id, order_total  
FROM orders  
WHERE order_total >= 100 and order_total <= 2000;`

**Answer: C, E.**

**Q70.**

Examine the structure of the EMPLOYEES and NEW EMPLOYEES tables:

| <u>EMPLOYEES</u> |              |             |
|------------------|--------------|-------------|
| EMPLOYEE_ID      | NUMBER       | Primary Key |
| FIRST_NAME       | VARCHAR2(25) |             |
| LAST_NAME        | VARCHAR2(25) |             |

```

HIRE_DATE DATE
NEW_EMPLOYEES
EMPLOYEE_ID NUMBER Primary Key
NAME VARCHAR2(60)

```

Which MERGE statement is valid?

```

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

B. MERGE new_employees c
 USING employees e
 ON (c.employee_id = e.employee_id)
 WHEN EXIST THEN
 UPDATE SET
 c.name = e.first_name || ',' || e.last_name
 WHEN NOT MATCHED THEN
 INSERT VALUES(e.employee_id, e.first_name || ',
 ' || e.last_name);

C. MERGE INTO new employees c
 USING employees e
 ON (c.employee_id = e.employee_id)
 WHEN EXISTS THEN
 UPDATE SET
 c.name = e.first_name || ',' || e.last_name
 WHEN NOT MATCHED THEN
 INSERT VALUES(e.employee_id, e.first_name || ',
 ' || e.last_name);

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

```

**Answer: A.**

**Q71.**

Examine the structure if the EMPLOYEES table:

| Column name   | Data Type    | Remarks                                                      |
|---------------|--------------|--------------------------------------------------------------|
| EMPLOYEE_ID   | NUMBER       | NOT NULL, Primary Key                                        |
| EMP_NAME      | VARCHAR2(30) |                                                              |
| JOB_ID        | VARCHAR2(20) | NOT NULL                                                     |
| SAL           | NUMBER       |                                                              |
| MGR_ID        | NUMBER       | References EMPLOYEE_ID column                                |
| DEPARTMENT_ID | NUMBER       | Foreign key to DEPARTMENT_ID column of the DEPARTMENTS table |

You need to create a view called EMP\_VU that allows the user to insert rows through the view. Which SQL statement, when used to create the EMP\_VU view, allows the user to insert rows?

- A. CREATE VIEW emp\_Vu AS  
SELECT employee\_id, emp\_name,  
department\_id  
FROM employees  
WHERE mgr\_id IN (102, 120);
- B. CREATE VIEW emp\_Vu AS  
SELECT employee\_id, emp\_name, job\_id  
department\_id  
FROM employees  
WHERE mgr\_id IN (102, 120);
- C. CREATE VIEW emp\_Vu AS  
SELECT department\_id, SUM(sal) TOTALSAL  
FROM employees  
WHERE mgr\_id IN (102, 120)  
GROUP BY department\_id;
- D. CREATE VIEW emp\_Vu AS  
SELECT employee\_id, emp\_name, job\_id,  
DISTINCT department\_id  
FROM employees;

**Answer: B.**

### Q72.

Which three SELECT statements displays 2000 in the format “\$2,000.00”? (Choose three)

- A. SELECT TO CNAR(2000, '\$#,###.##')  
FROM dual;
- B. SELECT TO CNAR(2000, '\$0,000.00')  
FROM dual;
- C. SELECT TO CNAR(2000, '\$9,999.00')  
FROM dual;
- D. SELECT TO CNAR(2000, '\$9,999.99')  
FROM dual;
- E. SELECT TO CNAR(2000, '\$2,000.00')

```
FROM dual;
F. SELECT TO CNAR(2000, '$N,NNN.NN')
FROM dual;
```

**Answer: B, C, D.**

**Q73.**

Which statement explicitly names a constraint?

```
A. ALTER TABLE student_grades
ADD
FOREIGN KEY (student_id) REFERENCES students(student_id);
B. ALTER TABLE student_grades
ADD CONSTRAINT NAME = student_id_fk
FOREIGN KEY (student_id) REFERENCES students(student_id);
C. ALTER TABLE student_grades
ADD CONSTRAINT student_id_fk
FOREIGN KEY (student_id) REFERENCES students(student_id);
D. ALTER TABLE student grades
ADD NAMED CONSTRAINT student_id_fk
FOREIGN KEY (student_id) REFERENCES students(student_id);
E. ALTER TABLE student grades
ADD NAME student_id_fk
FOREIGN KEY (student_id) REFERENCES students(student_id);
```

**Answer: C.**

**Q74.**

Examine the description of the MARKS table:

|              |              |
|--------------|--------------|
| STD_ID       | NUMBER(4)    |
| STUDENT_NAME | VARCHAR2(30) |
| SUBJ1        | NUMBER(3)    |
| SUBJ2        | NUMBER(3)    |

SUBJ1 and SUBJ2 indicate the marks obtained by a student in two subjects.

Examine this SELECT statement based on the MARKS table:

```
SELECT subj1+subj2 total_marks, std_id
FROM marks
WHERE subj1 > AVG(subj1) AND subj2 > AVG(subj2)
ORDER BY total_marks;
```

What is the result of the SELECT statement?

- A. The statement executes successfully and returns the student ID and sum of all marks for each student who obtained more than the average mark in each subject.
- B. The statement returns an error at the SELECT clause.
- C. The statement returns an error at the WHERE clause.
- D. The statement returns an error at the ORDER BY clause.

**Answer: C.**

**Q75.**

Evaluate the set of SQL statements:

```
CREATE TABLE dept
(deptno NUMBER(2),
dname VARCHAR2(14),
loc VARCHAR2(13));
ROLLBACK;
DESCRIBE DEPT
```

What is true about the set?

- A. The DESCRIBE DEPT statement displays the structure of the DEPT table.
- B. The ROLLBACK statement frees the storage space occupies by the DEPT table.
- C. The DESCRIBE DEPT statement returns an error ORA-04043: object DEPT does not exist.
- D. The DESCRIBE DEPT statement displays the structure of the DEPT table only if there is a COMMIT statement introduced before the ROLLBACK statement.

**Answer: A.**

**Q76.**

Examine the data in the EMPLOYEES and DEPARTMENTS tables:

**EMPLOYEES**

| EMPLOYEE_ID | EMP_NAME | DEPT_ID | MGR_ID | JOB_ID   | SALARY |
|-------------|----------|---------|--------|----------|--------|
| 101         | Smith    | 20      | 120    | SA_REP   | 4000   |
| 102         | Martin   | 10      | 105    | CLERK    | 2500   |
| 103         | Chris    | 20      | 120    | IT_ADMIN | 4200   |
| 104         | John     | 30      | 108    | HR_CLERK | 2500   |
| 105         | Diana    | 30      | 108    | IT_ADMIN | 5000   |
| 106         | Smith    | 40      | 110    | AD_ASST  | 3000   |
| 108         | Jennifer | 30      | 110    | HR_DIR   | 6500   |
| 110         | Bob      | 40      |        | EX_DIR   | 8000   |
| 120         | Ravi     | 20      | 110    | SA*DIR   | 6500   |

## DEPARTMENTS

| DEPARTMENT_ID | DEPARTMENT_NAME |
|---------------|-----------------|
| 10            | Admin           |
| 20            | Education       |
| 30            | IT              |
| 40            | Human Resources |

Also examine the SQL statements that create the EMPLOYEES and DEPARTMENTS tables:

```
CREATE TABLE departments
(department_id NUMBER PRIMARY KEY,
department_name VARCHAR2(30));
CREATE TABLE employees
(EMPLOYEE_ID NUMBER PRIMARY KEY,
EMP_NAME VARCHAR2(20),
DEPT_ID NUMBER REFERENCES
departments(department_id),
MGR_ID NUMBER REFERENCES
employees(employee_id),
MGR_ID NUMBER REFERENCES
employees(employee_id),
JOB_ID VARCHAR2(15),
SALARY NUMBER);
ON the EMPLOYEES,
```

On the EMPLOYEES table, EMPLOYEE\_ID is the primary key.  
MGR\_ID is the ID of managers and refers to the EMPLOYEE\_ID.  
DEPT\_ID is foreign key to DEPARTMENT\_ID column of the DEPARTMENTS table.  
On the DEPARTMENTS table, DEPARTMENT\_ID is the primary key.  
Examine this DELETE statement:

```
DELETE
FROM departments
WHERE department id = 40;
```

What happens when you execute the DELETE statement?

- A. Only the row with department ID 40 is deleted in the DEPARTMENTS table.
- B. The statement fails because there are child records in the EMPLOYEES table with department ID 40.
- C. The row with department ID 40 is deleted in the DEPARTMENTS table. Also the rows with employee IDs 110 and 106 are deleted from the EMPLOYEES table.
- D. The row with department ID 40 is deleted in the DEPARTMENTS table. Also the rows with employee IDs 106 and 110 and the employees working under employee 110 are deleted from the EMPLOYEES table.
- E. The row with department ID 40 is deleted in the DEPARTMENTS table. Also all the rows in the EMPLOYEES table are deleted.
- F. The statement fails because there are no columns specifies in the DELETE clause of the DELETE statement.

**Answer: B.**

**Q77.**

Examine the description of the STUDENTS table:

|            |              |
|------------|--------------|
| STD_ID     | NUMBER(4)    |
| COURSE_ID  | VARCHAR2(10) |
| START_DATE | DATE         |
| END_DATE   | DATE         |

Which two aggregate functions are valid on the START\_DATE column? (Choose two)

- A. SUM(start\_date)
- B. AVG(start\_date)
- C. COUNT(start\_date)
- D. AVG(start\_date, end\_date)
- E. MIN(start\_date)
- F. MAXIMUM(start\_date)

**Answer: C, E.**

**Q78.**

The database administrator of your company created a public synonym called HR for the HUMAN\_RESOURCES table of the GENERAL schema, because many users frequently use this table.

As a user of the database, you created a table called HR in your schema. What happens when you execute this query?

```
SELECT *
FROM HR;
```

- A. You obtain the results retrieved from the public synonym HR created by the database administrator.
- B. You obtain the results retrieved from the HR table that belongs to your schema.
- C. You get an error message because you cannot retrieve from a table that has the same name as a public synonym.
- D. You obtain the results retrieved from both the public synonym HR and the HR table that belongs to your schema, as a Cartesian product.
- E. You obtain the results retrieved from both the public synonym HR and the HR table that belongs to your schema, as a FULL JOIN.

**Answer: B.**



**Q79.**

Examine the structure of the EMPLOYEES table:

|             |               |             |
|-------------|---------------|-------------|
| EMPLOYEE_ID | NUMBER        | Primary Key |
| FIRST_NAME  | VARCNAR2 (25) |             |
| LAST_NAME   | VARCNAR2 (25) |             |

Which three statements inserts a row into the table? (Choose three)

- A. INSERT INTO employees  
VALUES ( NULL, 'John', 'Smith');
- B. INSERT INTO employees( first\_name, last\_name)  
VALUES ('John', 'Smith');
- C. INSERT INTO employees  
VALUES ('1000', 'John', NULL);
- D. INSERT INTO employees(first\_name,last\_name, employee\_id)  
VALUES ( 1000, 'John', 'Smith');
- E. INSERT INTO employees (employee\_id)  
VALUES (1000);
- F. INSERT INTO employees (employee\_id, first\_name, last\_name)  
VALUES ( 1000, 'John', '' );

**Answer: C, E, F.**

**Q80.**

You are granted the CREATE VIEW privilege. What does this allow you to do?

- A. Create a table view.
- B. Create a view in any schema.
- C. Create a view in your schema.
- D. Create a sequence view in any schema.
- E. Create a view that is accessible by everyone.
- F. Create a view only if it is based on tables that you created.

**Answer: C.**

**Q81.**

The EMPLOYEES table has these columns:

|           |               |
|-----------|---------------|
| LAST NAME | VARCHAR2 (35) |
| SALARY    | NUMBER (8, 2) |

HIRE\_DATE          DATE

Management wants to add a default value to the SALARY column. You plan to alter the table by using this SQL statement:

```
ALTER TABLE EMPLOYEES
MODIFY (SALARY DEFAULT 5000);
```

What is true about your ALTER statement?

- A. Column definitions cannot be altered to add DEFAULT values.
- B. A change to the DEFAULT value affects only subsequent insertions to the table.
- C. Column definitions cannot be altered to add DEFAULT values for columns with a NUMBER data type.
- D. All the rows that have a NULL value for the SALARY column will be updated with the value 5000.

**Answer: B.**

**Q82.**

Which three are true regarding the use of outer joins? (Choose three.)

- A. You cannot use IN operator in a condition that involves an outerjoin.
- B. You use (+) on both sides of the WHERE condition to perform an outerjoin.
- C. You use (\*) on both sides of the WHERE condition to perform an outerjoin.
- D. You use an outerjoin to see only the rows that do not meet the join condition.
- E. In the WHERE condition, you use (+) following the name of the column in the table without matching rows, to perform an outerjoin.
- F. You cannot link a condition that is involved in an outerjoin to another condition by using the OR operator.

**Answer: D, E, F.**

**Q83.**

What does the TRUNCATE statement do?

- A. Removes the table
- B. Removes all rows from a table
- C. Shortens the table to 10 rows
- D. Removes all columns from a table
- E. Removes foreign keys from a table

**Answer: B.**

**Q84.**

What is true about the WITH GRANT OPTION clause?

- A. It allows a grantee DBA privileges.
- B. It is required syntax for object privileges.
- C. It allows privileges on specified columns of tables.
- D. It is used to grant an object privilege on a foreign key column.
- E. It allows the grantee to grant object privileges to other users and roles.

**Answer: E.**

**Q85.**

A subquery can be used to \_\_\_\_\_.

- A. Create groups of data
- B. Sort data in a specific order
- C. Convert data to a different format
- D. Retrieve data based on an unknown condition

**Answer: D.**

**Q86.**

Which three statements correctly describe the functions and use of constraints? (Choose three.)

- A. Constraints provide data independence.
- B. Constraints make complex queries easy.
- C. Constraints enforce rules at the view level.
- D. Constraints enforce rules at the table level.
- E. Constraints prevent the deletion of a table if there are dependencies.
- F. Constraints prevent the deletion of an index if there are dependencies.

**Answer: C, D, E.**

**Q87.**

What is true about joining tables through an equijoin?

- A. You can join a maximum of two tables through an equijoin.

- B. You can join a maximum of two columns through an equijoin.
- C. You specify an equijoin condition in the SELECT or FROM clauses of a SELECT statement.
- D. To join two tables through an equijoin, the columns in the join condition must be primary key and foreign key columns.
- E. You can join n tables (all having single column primary keys) in a SQL statement by specifying a minimum of n-1 join conditions.

**Answer: E.**

**Q88.**

Examine the data from the ORDERS and CUSTOMERS tables.

**ORDERS**

| ORD_ID | ORD_DATE    | CUST_ID | ORD_TOTAL |
|--------|-------------|---------|-----------|
| 100    | 12-JAN-2000 | 15      | 10000     |
| 101    | 09-MAR-2000 | 40      | 8000      |
| 102    | 09-MAR-2000 | 35      | 12500     |
| 103    | 15-MAR-2000 | 15      | 12000     |
| 104    | 25-JUN-2000 | 15      | 6000      |
| 105    | 18-JUL-2000 | 20      | 5000      |
| 106    | 18-JUL-2000 | 35      | 7000      |
| 107    | 21-JUL-2000 | 20      | 6500      |
| 109    | 04-AUG-2000 | 10      | 8000      |

**CUSTOMERS**

| CUST_ID | CUST_NAME | CITY          |
|---------|-----------|---------------|
| 10      | Smith     | Los Angeles   |
| 15      | Bob       | San Francisco |
| 20      | Martin    | Chicago       |
| 25      | Mary      | New York      |
| 30      | Rina      | Chicago       |
| 35      | Smith     | New York      |
| 40      | Lind      | New York      |

Evaluate the SQL statement:

```
SELECT *
FROM orders
WHERE cust_id = (SELECT cust_id
 FROM customers
 WHERE cust_name = 'Smith');
```

What is the result when the query is executed?

A.

| ORD_ID | ORD_DATE    | CUST_ID | ORD_TOTAL |
|--------|-------------|---------|-----------|
| 102    | 09-MAR-2000 | 35      | 12500     |

|     |             |    |      |
|-----|-------------|----|------|
| 106 | 18-JUL-2000 | 35 | 7000 |
| 108 | 04-AUG-2000 | 10 | 8000 |

B.

| ORD_ID | ORD_DATE    | CUST_ID | ORD_TOTAL |
|--------|-------------|---------|-----------|
| 102    | 09-MAR-2000 | 35      | 12500     |
| 106    | 18-JUL-2000 | 35      | 7000      |

C.

| ORD_ID | ORD_DATE    | CUST_ID | ORD_TOTAL |
|--------|-------------|---------|-----------|
| 108    | 04-AUG-2000 | 10      | 8000      |

D. The query fails because the subquery returns more than one row.

E. The query fails because the outer query and the inner query are using different tables.

**Answer: D.**

### Q89.

Evaluate this SQL statement:

```
SELECT e.EMPLOYEE_ID, e.LAST_NAME, e.DEPARTMENT_ID, d.DEPARTMENT_NAME
FROM EMP e, DEPARTMENT d
WHERE e.DEPARTMENT_ID = d.DEPARTMENT_ID;
```

In the statement, which capabilities of a SELECT statement are performed?

- A. Selection, projection, join
- B. Difference, projection, join
- C. Selection, intersection, join
- D. Intersection, projection, join
- E. Difference, projection, product

**Answer: A.**

### Q90.

Examine the structure of the EMPLOYEES table:

|               |               |             |
|---------------|---------------|-------------|
| EMPLOYEE_ID   | NUMBER        | Primary Key |
| FIRST_NAME    | VARCHAR2 (25) |             |
| LAST_NAME     | VARCHAR2 (25) |             |
| DEPARTMENT_ID | NUMBER        |             |
| SALARY        | NUMBER        |             |

What is the correct syntax for an inline view?

- A. SELECT a.last\_name, a.salary, a.department\_id, b.maxsal

```
FROM employees a,
 (SELECT department_id, max(salary)maxsal
 FROM employees
 GROUP BY department_id) b
WHERE a.department_id = b.department_id
AND a.salary < b.maxsal;
B. SELECT a.last_name, a.salary, a.department_id
FROM employees a
WHERE a.department_id IN
 (SELECT department_id
 FROM employees b
 GROUP BY department_id having salary =
 (SELECT max(salary) from employees
 WHERE a.department_id = b.department_id);
C. SELECT a.last_name, a.salary, a.department_id
FROM employees a
WHERE a.salary =
 (SELECT max(salary)
 FROM employees b
 WHERE a.department_id = b.department_id);
D. SELECT a.last_name, a.salary, a.department_id
FROM employees a
WHERE (a.department_id, a.salary) IN
 (SELECT department_id, a.salary) IN
 (SELECT department_id max(salary)
 FROM employees b
 GROUP BY department_id
 ORDER BY department_id);
```

**Answer: A.**

**Q91.**

Which clause would you use in a SELECT statement to limit the display to those employees whose salary is greater than 5000?

- A. ORDER BY SALARY > 5000
- B. GROUP BY SALARY > 5000
- C. HAVING SALARY > 5000
- D. WHERE SALARY > 5000

**Answer: D.**

**Q92.**

Examine the structure of the EMPLOYEES table:

|               |              |                  |
|---------------|--------------|------------------|
| EMPLOYEE_ID   | NUMBER       | NOT NULL         |
| EMP_NAME      | VARCHAR2(30) |                  |
| JOB_ID        | VARCHAR2(20) | DEFAULT 'SA_REP' |
| SAL           | NUMBER       |                  |
| COMM_PCT      | NUMBER       |                  |
| MGR_ID        | NUMBER       |                  |
| DEPARTMENT_ID | NUMBER       |                  |

You need to update the records of employees 103 and 115. The UPDATE statement you specify should update the rows with the values specified below:

JOB\_ID: Default value specified for this column definition.  
 SAL: Maximum salary earned for the job ID SA\_REP.  
 COMM\_PCT: Default value specified for this commission percentage column, if any.  
 If no default value is specified for the column, the value should be NULL.  
 DEPARTMENT\_ID: Supplied by the user during run time through substitution variable.

Which UPDATE statement meets the requirements?

- A. UPDATE employees  
 SET job\_id = DEFAULT  
 AND Sal = (SELECT MAX(sal)  
           FROM employees  
           WHERE job\_id = 'SA\_REP')  
 AND comm\_pct = DEFAULT  
 AND department\_id = &did  
 WHERE employee\_id IN (103,115);
- B. UPDATE employees  
 SET job\_id = DEFAULT  
 AND Sal = MAX(sal)  
 AND comm\_pct = DEFAULT OR NULL  
 AND department\_id = &did  
 WHERE employee\_id IN (103,115)  
 AND job\_id = 'SA\_REP';
- C. UPDATE employees  
 SET job\_id = DEFAULT,  
     Sal = (SELECT MAX(sal)  
           FROM employees  
           WHERE job\_id = 'SA\_REP'),  
     comm\_pct = DEFAULT,  
     department\_id = &did  
 WHERE employee\_id IN (103,115);
- D. UPDATE employees  
 SET job\_id = DEFAULT,  
     Sal = MAX(sal),  
     comm\_pct = DEFAULT,  
     department\_id = &did  
 WHERE employee\_id IN (103,115)

```

AND job_id = 'SA_REP';
E. UPDATE employees
SET job_id = DEFAULT,
 Sal = (SELECT MAX(sal)
 FROM employees
 WHERE job_id = 'SA_REP')
 comm_pct = DEFAULT OR NULL,
 department_id = &did
WHERE employee_id IN (103,115);

```

**Answer: C.**

**Q93.**

Examine the structures of the EMPLOYEES and TAX tables.

#### EMPLOYEES

|               |               |                                                              |
|---------------|---------------|--------------------------------------------------------------|
| EMPLOYEE_ID   | NUMBER        | NOT NULL, Primary Key                                        |
| EMP_NAME      | VARCHAR2 (30) |                                                              |
| JOB_ID        | VARCHAR2 (20) |                                                              |
| SALARY        | NUMBER        |                                                              |
| MGR_ID        | NUMBER        | References EMPLOYEE_ID column                                |
| DEPARTMENT_ID | NUMBER        | Foreign key to DEPARTMENT_ID column of the DEPARTMENTS table |

#### TAX

|             |        |                                       |
|-------------|--------|---------------------------------------|
| MIN_SALARY  | NUMBER |                                       |
| MAX_SALARY  | NUMBER |                                       |
| TAX_PERCENT | NUMBER | Percentage tax for given salary range |

You need to find the percentage tax applicable for each employee. Which SQL statement would you use?

- A. 

```
SELECT employee_id, salary, tax_percent
FROM employees e, tax t
WHERE e.salary BETWEEN t.min_salary AND t.max_salary;
```
- B. 

```
SELECT employee_id, salary, tax_percent
FROM employees e, tax t
WHERE e.salary > t.min_salary, tax_percent
```
- C. 

```
SELECT employee_id, salary, tax_percent
FROM employees e, tax t
WHERE MIN(e.salary) = t.min_salary
AND MAX(e.salary) = t.max_salary
```
- D. You cannot find the information because there is no common column between the two tables.

**Answer: A.**



**Q94.**

Which two are attributes of /SQL\*Plus? (Choose two)

- A. /SQL\*Plus commands cannot be abbreviated.
- B. /SQL\*Plus commands are accessed from a browser.
- C. /SQL\*Plus commands are used to manipulate data in tables.
- D. /SQL\*Plus commands manipulate table definitions in the database.
- E. /SQL\*Plus is the Oracle proprietary interface for executing SQL statements.

**Answer: C, D.****Q95.**

Which four statements correctly describe functions that are available in SQL? (Choose four)

- A. INSTR returns the numeric position of a named character.
- B. NVL2 returns the first non-null expression in the expression list.
- C. TRUNCATE rounds the column, expression, or value to n decimal places.
- D. DECODE translates an expression after comparing it to each search value.
- E. TRIM trims the heading of trailing characters (or both) from a character string.
- F. NVL compares two expressions and returns null if they are equal, or the first expression if they are not equal.
- G. NULLIF compares two expressions and returns null if they are equal, or the first expression if they are not equal.

**Answer: A, D, E, G.****Q96.**

Examine the data in the EMPLOYEES table.

EMPLOYEES

| EMPLOYEE_ID | EMP_NAME | DEPT_ID | MGR_ID | JOB_ID   | SALARY |
|-------------|----------|---------|--------|----------|--------|
| 101         | Smith    | 20      | 120    | SA_REP   | 4000   |
| 102         | Martin   | 10      | 105    | CLERK    | 2500   |
| 103         | Chris    | 20      | 120    | IT_ADMIN | 4200   |
| 104         | John     | 30      | 108    | HR_CLERK | 2500   |
| 105         | Diana    | 30      | 108    | IT_ADMIN | 5000   |
| 106         | Smith    | 40      | 110    | AD_ASST  | 3000   |
| 108         | Jennifer | 30      | 110    | HR_DIR   | 6500   |
| 110         | Bob      | 40      |        | EK_DIR   | 8000   |
| 120         | Revi     | 20      | 110    | SA_DIR   | 6500   |

On the EMPLOYEES table, EMPLOYEE\_ID is the primary key. MGR\_ID is the ID of managers and refers to the EMPLOYEE\_ID. The JOB\_ID column is a NOT NULL column.

Evaluate this DELETE statement:

```
DELETE employee_id, salary, job_id
FROM employees
WHERE dept_id = 90;
```

Why does the DELETE statement fail when you execute it?

- A. There is no row with dept\_id 90 in the EMPLOYEES table.
- B. You cannot delete the JOB\_ID column because it is a NOT NULL column.
- C. You cannot specify column names in the DELETE clause of the DELETE statement.
- D. You cannot delete the EMPLOYEE\_ID column because it is the primary key of the table.

**Answer: C.**

**Q97.**

The CUSTOMERS table has these columns:

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

Which statement finds the rows in the CUSTOMERS table that do not have a postal code?

- A. 

```
SELECT customer_id, customer_name
FROM customers
WHERE postal_code CONTAINS NULL;
```
- B. 

```
SELECT customer_id, customer_name
FROM customers
WHERE postal_code = '_____';
```
- C. 

```
SELECT customer_id, customer_name
FROM customers
WHERE postal_code IS NULL;
```
- D. 

```
SELECT customer_id, customer_name
FROM customers
WHERE postal code IS NVL;
```
- E. 

```
SELECT customer_id, customer_name
FROM customers
```

WHERE postal\_code = NULL;

**Answer: C.**

**Q98.**

Examine the data in the EMPLOYEES table.

| LAST NAME | DEPARTMENT ID | SALARY |
|-----------|---------------|--------|
| Geiz      | 10            | 3000   |
| Davis     | 20            | 1500   |
| King      | 20            | 2200   |
| Davis     | 30            | 5000   |
| ---       |               |        |

Which three subqueries work? (Choose three.)

A. SELECT \*  
FROM employees  
where salary > (SELECT MIN(salary)  
FROM employees  
GROUP BY department\_id);

B. SELECT \*  
FROM employees  
WHERE salary = (SELECT AVG(salary)  
FROM employees  
GROUP BY department\_id);

C. SELECT distinct department\_id  
FROM employees  
WHERE salary > ANY (SELECT AVG(salary)  
FROM employees  
GROUP BY department\_id);

D. SELECT department\_id  
FROM employees  
WHERE salary > ALL (SELECT AVG(salary)  
FROM employees  
GROUP BY department\_id);

E. SELECT last\_name  
FROM employees  
WHERE salary > ANY (SELECT MAX(salary)  
FROM employees  
GROUP BY department\_id);

F. SELECT department\_id  
FROM employees  
WHERE salary > ALL (SELECT AVG(salary)  
FROM employees  
GROUP BY AVG(SALARY));

**Answer: C, D, E.**

**Q99.**

What is necessary for your query on an existing view to execute successfully?

- A. The underlying tables must have data.
- B. You need SELECT privileges on the view.
- C. The underlying tables must be in the same schema.
- D. You need SELECT privileges only on the underlying tables.

**Answer: B.**

**Q100.**

Which SELECT statement will the result 'ello world' from the string 'HelloWorld'?

- A. SELECT SUBSTR( 'HelloWorld',1) FROM dual;
- B. SELECT INITCAP(TRIM( 'HelloWorld', 1,1)) FROM dual;
- C. SELECT LOWER(SUBSTR('Hello World', 1, 1) FROM dual;
- D. SELECT LOWER(SUBSTR('Hello World', 2, 1) FROM dual;
- E. SELECT LOWER(TRIM( 'H' FROM 'Hello World')) FROM dual;

**Answer: E.**

**Q101.**

Examine the data in the EMPLOYEES and EMP\_HIST tables:

EMPLOYEES

| EMPLOYEE_ID | NAME     | DEPT_ID | MGR_ID | JOB_ID   | SALARY |
|-------------|----------|---------|--------|----------|--------|
| 101         | Smith    | 20      | 120    | SA_REP   | 4000   |
| 102         | Martin   | 10      | 105    | CLERK    | 2500   |
| 103         | Chris    | 20      | 120    | IT_ADMIN | 4200   |
| 104         | John     | 30      | 108    | HR_CLERK | 2500   |
| 105         | Diana    | 30      | 108    | IT_ADMIN | 5000   |
| 106         | Smith    | 40      | 110    | AD_ASST  | 3000   |
| 108         | Jennifer | 30      | 110    | HR_DIR   | 6500   |
| 110         | Bob      | 40      |        | EX_DIR   | 8000   |
| 120         | Ravi     | 20      | 110    | SA_DIR   | 6500   |

EMP\_HIST

| EMPLOYEE_ID | NAME | JOB_ID | SALARY |
|-------------|------|--------|--------|
|-------------|------|--------|--------|

|     |          |          |      |
|-----|----------|----------|------|
| 101 | Smith    | SA CLERK | 2000 |
| 103 | Chris    | IT CLERK | 2200 |
| 104 | John     | HR CLERK | 2000 |
| 106 | Smith    | AD ASST  | 3000 |
| 108 | Jennifer | HR_MGR   | 4500 |
|     |          |          |      |

The EMP\_HIST table is updated at the end of every year. The employee ID, name, job ID, and salary of each existing employee are modified with the latest data. New employee details are added to the table.

Which statement accomplishes this task?

```

A. UPDATE emp_hist
SET employee_id, name, job_id, salary =
 (SELECT employee_id, name, job_id, salary
FROM employees)
WHERE employee_id IN
 (SELECT employee_id
FROM employees);
B. MERGE INTO emp_hist eh
USING employees e
ON (eh.employee_id = e.employee_id)
WHEN MATCHED THEN
UPDATE SET eh.name = e.name,
 eh.job_id = e.job_id,
 eh.salary = e.salary
WHEN NOT MATCHED THEN
INSERT VALUES (e.employee_id, e.name,
e.job_id, e.salary);
C. MERGE INTO emp_hist eh
USING employees e
ON (eh.employee_id = e.employee_id)
WHEN MATCHED THEN
UPDATE emp_hist SET eh.name = e.name,
 eh.job_id = e.job_id,
 eh.salary = e.salary
WHEN NOT MATCHED THEN
INSERT INTO emp_hist
VALUES (e.employee_id, e.name, e.job_id, e.salary);
D. MERGE INTO emp_hist eh
USING employees e
WHEN MATCHED THEN
UPDATE emp_hist SET eh.name = e.name,
 eh.job_id = e.job_id,
 eh.salary = e.salary
WHEN NOT MATCHED THEN
INSERT INTO emp_hist
VALUES (e.employee_id, e.name, e.job_id, e.salary);

```

**Answer: B.**

**Q102.**

Examine the data of the EMPLOYEES table.

EMPLOYEES (EMPLOYEE\_ID is the primary key. MGR\_ID is the ID of managers and refers to the EMPLOYEE\_ID)

| EMPLOYEE ID | EMP NAME | DEPT ID | MGR ID | JOB ID   | SALARY |
|-------------|----------|---------|--------|----------|--------|
| 101         | Smith    | 20      | 120    | SA_REP   | 4000   |
| 102         | Martin   | 10      | 105    | CLERK    | 2500   |
| 103         | Chris    | 20      | 120    | IT_ADMIN | 4200   |
| 104         | John     | 30      | 108    | HR_CLERK | 2500   |
| 105         | Diana    | 30      | 108    | HR_MGR   | 5000   |
| 106         | Bryan    | 40      | 110    | AD_ASST  | 3000   |
| 108         | Jennifer | 30      | 110    | HR_DIR   | 6500   |
| 110         | Bob      | 40      |        | EX_DIR   | 8000   |
| 120         | Ravi     | 20      | 110    | SA_DIR   | 6500   |

Which statement lists the ID, name, and salary of the employee, and the ID and name of the employee's manager, for all the employees who have a manager and earn more than 4000?

- A. SELECT employee\_id "Emp\_id", emp\_name "Employee",  
salary,  
employee\_id "Mgr\_id", emp\_name "Manager"  
FROM employees  
WHERE salary > 4000;
- B. SELECT e.employee\_id "Emp\_id", e.emp\_name "Employee",  
e.salary,  
m.employee\_id "Mgr\_id", m.emp\_name "Manager"  
FROM employees e, employees m  
WHERE e.mgr\_id = m.mgr\_id  
AND e.salary > 4000;
- C. SELECT e.employee\_id "Emp\_id", e.emp\_name "Employee",  
e.salary,  
m.employee\_id "Mgr\_id", m.emp\_name "Manager"  
FROM employees e, employees m  
WHERE e.mgr\_id = m.employee\_id  
AND e.salary > 4000;
- D. SELECT e.employee\_id "Emp\_id", e.emp\_name "Employee",  
e.salary,  
m.mgr\_id "Mgr\_id", m.emp\_name "manager"  
FROM employees e, employees m  
WHERE e.mgr\_id = m.employee\_id  
AND e.salary > 4000;
- E. SELECT e.employee\_id "Emp\_id", e.emp\_name "Employee",  
e.salary,  
m.mgr\_id "Mgr\_id", m.emp\_name "Manager"  
FROM employees e, employees m

```
WHERE e.employee_id = m.employee_id
AND e.salary > 4000;
```

**Answer: C.**

**Q103.**

Examine the description of the EMPLOYEES table:

|            |              |          |
|------------|--------------|----------|
| EMP_ID     | NUMBER(4)    | NOT NULL |
| LAST_NAME  | VARCHAR2(30) | NOT NULL |
| FIRST_NAME | VARCHAR2(30) |          |
| DEPT_ID    | NUMBER(2)    |          |
| JOB_CAT    | VARCHAR2(30) |          |
| SALARY     | NUMBER(8,2)  |          |

Which statement shows the department ID, minimum salary, and maximum salary paid in that department, only of the minimum salary is less than 5000 and the maximum salary is more than 15000?

- A. SELECT dept\_id, MIN(salary), MAX(salary)  
FROM employees  
WHERE MIN(salary) < 5000 AND MAX(salary) > 15000;
- B. SELECT dept\_id, MIN(salary), MAX(salary)  
FROM employees  
WHERE MIN(salary) < 5000 AND MAX(salary) > 15000  
GROUP BY dept\_id;
- C. SELECT dept\_id, MIN(salary), MAX(salary)  
FROM employees  
HAVING MIN(salary) < 5000 AND MAX(salary) > 15000;
- D. SELECT dept\_id, MIN(salary), MAX(salary)  
FROM employees  
GROUP BY dept\_id  
HAVING MIN(salary) < 5000 AND MAX(salary) < 15000;
- E. SELECT dept\_id, MIN(salary), MAX(salary)  
FROM employees  
GROUP BY dept\_id, salary  
HAVING MIN(salary) < 5000 AND MAX(salary) > 15000;

**Answer: E.**

**Q104.**

In a SELECT statement that includes a WHERE clause, where is the GROUP BY clause placed in the SELECT statement?

- A. Immediately after the SELECT clause
- B. Before the WHERE clause
- C. Before the FROM clause
- D. After the ORDER BY clause
- E. After the WHERE clause

**Answer: E.**

**Q105.**

The EMP table contains these columns:

|               |               |
|---------------|---------------|
| LAST_NAME     | VARCHAR2 (25) |
| SALARY        | NUMBER (6, 2) |
| DEPARTMENT_ID | NUMBER (6)    |

You need to display the employees who have not been assigned to any department.  
You write the SELECT statement:

```
SELECT LAST_NAME, SALARY, DEPARTMENT_ID
FROM EMP
WHERE DEPARTMENT_ID = NULL;
```

What is true about this SQL statement?

- A. The SQL statement displays the desired results.
- B. The column in the WHERE clause should be changed to display the desired results.
- C. The operator in the WHERE clause should be changed to display the desired results.
- D. The WHERE clause should be changed to use an outer join to display the desired results.

**Answer: C.**

**Q106.**

Which /SQL\*Plus feature can be used to replace values in the WHERE clause?

- A. Substitution variables
- B. Replacement variables
- C. Prompt variables
- D. Instead-of variables
- E. This feature cannot be implemented through /SQL\*Plus.

**Answer: A.**



**Q107.**

You need to modify the STUDENTS table to add a primary key on the STUDENT\_ID column. The table is currently empty.

Which statement accomplishes this task?

- A. ALTER TABLE students  
ADD PRIMARY KEY student\_id;
- B. ALTER TABLE students  
ADD CONSTRAINT PRIMARY KEY (student\_id);
- C. ALTER TABLE students  
ADD CONSTRAINT stud\_id\_pk PRIMARY KEY student\_id;
- D. ALTER TABLE students  
ADD CONSTRAINT stud\_id\_pk PRIMARY KEY (student\_id);
- E. ALTER TABLE students  
MODIFY CONSTRAINT stud\_id\_pk PRIMARY KEY (student\_id);

**Answer: D.**

**Q108.**

Which SQL statement defines the FOREIGN KEY constraint on the DEPTNO column of the EMP table?

- A. CREATE TABLE EMP  
(empno NUMBER(4),  
ename VARNAR2(35),  
deptno NUMBER(7,2) NOT NULL  
CONSTRAINT emp\_deptno\_fk FOREIGN KEY deptno  
REFERENCES dept deptno);
- B. CREATE TABLE EMP  
(empno NUMBER(4),  
ename VARNAR2(35),  
deptno NUMBER(7,2)  
CONSTRAINT emp\_deptno\_fk REFERENCES dept (deptno));
- C. CREATE TABLE EMP  
(empno NUMBER(4)  
ename VARCHAR2(35),  
deptno NUMBER(7,2) NOT NULL,  
CONSTRAINT emp\_deptno\_fk REFERENCES dept (deptno)  
FOREIGN KEY (deptno));
- D. CREATE TABLE EMP (empno NUMBER(4),  
ename VARNAR2(35),  
deptno NUMBER(7,2) FOREIGN KEY  
CONSTRAINT emp\_deptno\_fk REFERENCES dept (deptno));

**Answer: B.**

**Q109.**

Examine the data from the ORDERS and CUSTOMERS table.

**ORDERS**

| ORD_ID | ORD_DATE    | CUST_ID | ORD_TOTAL |
|--------|-------------|---------|-----------|
| 100    | 12-JAN-2000 | 15      | 10000     |
| 101    | 09-MAR-2000 | 40      | 8000      |
| 102    | 09-MAR-2000 | 35      | 12500     |
| 103    | 15-MAR-2000 | 15      | 12000     |
| 104    | 25-JUN-2000 | 15      | 6000      |
| 105    | 18-JUL-2000 | 20      | 5000      |
| 106    | 18-JUL-2000 | 35      | 7000      |
| 107    | 21-JUL-2000 | 20      | 6500      |
| 108    | 04-AUG-2000 | 10      | 8000      |

**CUSTOMERS**

| CUST_ID | CUST_NAME | CITY          |
|---------|-----------|---------------|
| 10      | Smith     | Los Angeles   |
| 15      | Bob       | San Francisco |
| 20      | Martin    | Chicago       |
| 25      | Mary      | New York      |
| 30      | Rina      | Chicago       |
| 35      | Smith     | New York      |
| 40      | Linda     | New York      |

Which SQL statement retrieves the order ID, customer ID, and order total for the orders that are placed on the same day that Martin places his orders?

- A. `SELECT ord_id, cust_id, ord_total  
FROM orders, customers  
WHERE cust_name='Martin'  
AND ord_date IN ('18-JUL-2000','21-JUL-2000');`
- B. `SELECT ord_id, cust_id, ord_total  
FROM orders  
Where ord_date IN (SELECT ord_date  
FROM orders  
WHERE cust_id = (SELECT cust_id  
FROM customers  
WHERE cust_name =  
'Martin'));`
- C. `SELECT ord_id, cust_id, ord_total  
FROM orders  
Where ord_date IN (SELECT ord_date  
FROM orders, customers  
Where cust_name = 'Martin');`

```
D. SELECT ord_id, cust_id, ord_total
FROM orders
WHERE cust_id IN (SELECT cust_id
 FROM customers
 WHERE cust name = 'Martin');
```

**Answer: B.**

**Q110.**

Which syntax turns an existing constraint on?

- A. ALTER TABLE table\_name  
ENABLE constraint\_name;
- B. ALTER TABLE table\_name  
STATUS = ENABLE CONSTRAINT constraint\_name;
- C. ALTER TABLE table\_name  
ENABLE CONSTRAINT constraint\_name;
- D. ALTER TABLE table\_name  
STATUS ENABLE CONSTRAINT constraint\_name;
- E. ALTER TABLE table\_name  
TURN ON CONSTRAINT constraint\_name;
- F. ALTER TABLE table\_name  
TURN ON CONSTRAINT constraint\_name;

**Answer: C.**

**Q111.**

Examine the structure of the EMPLOYEES, DEPARTMENTS, and LOCATIONS tables.

**EMPLOYEES**

|               |               |                                                              |
|---------------|---------------|--------------------------------------------------------------|
| EMPLOYEE_ID   | NUMBER        | NOT NULL, Primary Key                                        |
| EMP_NAME      | VARCHAR2 (30) |                                                              |
| JOB_ID        | VARCHAR2 (20) |                                                              |
| SALARY        | NUMBER        |                                                              |
| MGR_ID        | NUMBER        | References EMPLOYEE_ID column                                |
| DEPARTMENT_ID | NUMBER        | Foreign key to DEPARTMENT_ID column of the DEPARTMENTS table |

**DEPARTMENTS**

|                 |               |                                                 |
|-----------------|---------------|-------------------------------------------------|
| DEPARTMENT_ID   | NUMBER        | NOT NULL, Primary Key                           |
| DEPARTMENT_NAME | VARCHAR2 (30) |                                                 |
| MGR_ID          | NUMBER        | References MGR_ID column of the EMPLOYEES table |

|             |        |                                                          |
|-------------|--------|----------------------------------------------------------|
| LOCATION_ID | NUMBER | Foreign key to LOCATION_ID column of the LOCATIONS table |
|-------------|--------|----------------------------------------------------------|

## LOCATIONS

|             |               |                       |
|-------------|---------------|-----------------------|
| LOCATION_ID | NUMBER        | NOT NULL, Primary Key |
| CITY        | VARCHAR2 (30) |                       |

Which two SQL statements produce the name, department name, and the city of all the employees who earn more than 10000? (Choose two)

- A. `SELECT emp_name, department_name, city  
FROM employees e  
JOIN departments d  
USING (department_id)  
JOIN locations l  
USING (location_id)  
WHERE salary > 10000;`
- B. `SELECT emp_name, department_name, city  
FROM employees e, departments d, locations l  
JOIN ON(e.department_id = d.department_id)  
AND (d.location_id = l.location_id)  
AND salary > 10000;`
- C. `SELECT emp_name, department_name, city  
FROM employees e, departments d, locations l  
WHERE salary > 10000;`
- D. `SELECT emp_name, department_name, city  
FROM employees e, departments d, locations l  
WHERE e.department_id = d.department_id  
AND d.location_id = l.location_id  
AND salary > 10000;`
- E. `SELECT emp_name, department_name, city  
FROM employees e  
NATURAL JOIN departments, locations  
WHERE salary > 10000;`

**Answer: B, D.**

**Q112.**

Examine the data of the EMPLOYEES table.

EMPLOYEES (EMPLOYEE\_ID is the primary key. MGR\_ID is the ID of managers and refers to the EMPLOYEE\_ID)

| EMPLOYEE_ID | EMP NAME | DEPT ID | MGR_ID | JOB ID   | SALARY |
|-------------|----------|---------|--------|----------|--------|
| 101         | Smith    | 20      | 120    | SA_REP   | 4000   |
| 102         | Martin   | 10      | 105    | CLERK    | 2500   |
| 103         | Chris    | 20      | 120    | IT_ADMIN | 4200   |
| 104         | John     | 30      | 108    | HR_CLERK | 2500   |

|     |          |    |     |         |      |
|-----|----------|----|-----|---------|------|
| 105 | Diana    | 30 | 108 | HR_MGR  | 5000 |
| 106 | Bryan    | 40 | 110 | AD_ASST | 3000 |
| 108 | Jennifer | 30 | 110 | HR_DIR  | 6500 |
| 110 | Bob      | 40 |     | EX_DIR  | 8000 |
| 120 | Ravi     | 20 | 110 | SA_DIR  | 6500 |

Evaluate this SQL statement:

```
SELECT e.employee_id "Emp_id", e.emp_name "Employee", e.salary,
 m.employee_id "Mgr_id", m.emp_name "Manager"
FROM employees e, employees m
WHERE e.mgr_id = m.employee_id
AND e.salary > 4000;
```

What is its output?

A.

| EMP_id | EMPLOYEE | SALARY | Mgr_id | Manager  |
|--------|----------|--------|--------|----------|
| 110    | Bob      | 8000   |        | Bob      |
| 120    | Ravi     | 6500   | 110    | Ravi     |
| 108    | Jennifer | 6500   | 110    | Jennifer |
| 103    | Chris    | 4200   | 120    | Chris    |
| 105    | Diana    | 5000   | 108    | Diana    |

B.

| EMP_id | EMPLOYEE | SALARY | Mgr_id | Manager  |
|--------|----------|--------|--------|----------|
| 120    | Ravi     | 6500   | 110    | Bob      |
| 108    | Jennifer | 6500   | 110    | Bob      |
| 103    | Chris    | 4200   | 120    | Ravi     |
| 105    | Diana    | 5000   | 108    | Jennifer |

C.

| EMP_id | EMPLOYEE | SALARY | Mgr_id | Manager  |
|--------|----------|--------|--------|----------|
| 110    | Bob      | 8000   |        |          |
| 120    | Ravi     | 6500   | 110    | Bob      |
| 108    | Jennifer | 6500   | 110    | Bob      |
| 103    | Chris    | 4200   | 120    | Ravi     |
| 105    | Diana    | 5000   | 108    | Jennifer |

D

| EMP_id | EMPLOYEE | SALARY | Mgr_id | Manager  |
|--------|----------|--------|--------|----------|
| 110    | Bob      | 8000   | 110    | Bob      |
| 120    | Ravi     | 6500   | 120    | Ravi     |
| 108    | Jennifer | 6500   | 108    | Jennifer |
| 103    | Chris    | 4200   | 103    | Chris    |
| 105    | Diana    | 5000   | 105    | Dina     |

E. The SQL statement produces an error.

**Answer: B.**