

## Mid Term Exam Semester 2

Group functions ignore null values.

The PAYMENT table contains these columns:

PAYMENT\_ID NUMBER(9) PK

PAYMENT\_DATE DATE

CUSTOMER\_ID NUMBER(9)

Which SELECT statement could you use to display the number of times each customer payment was made between January 1, 2003 and June 30, 2003 ?

```
SELECT customer_id, COUNT(payment_id)
FROM payment
WHERE payment_date BETWEEN '01-JAN-2003' AND '30-JUN-2003'
GROUP BY customer_id;
```

Evaluate this statement:

```
SELECT department_id, AVG(salary)
```

```
FROM employees
```

```
WHERE job_id <> 69879
```

```
GROUP BY job_id, department_id
```

```
HAVING AVG(salary) > 35000
```

```
ORDER BY department_id;
```

Which clause restricts the result?

```
WHERE job_id <> 69879
HAVING AVG(salary) > 35000
```

What is the correct order of clauses in a SELECT statement?

```
SELECT FROM WHERE GROUP BY HAVING ORDER BY
```

The MANUFACTURER table contains these columns:

MANUFACTURER\_ID NUMBER

MANUFACTURER\_NAME VARCHAR2(30)

TYPE VARCHAR2(25)

LOCATION\_ID NUMBER

You need to display the number of unique types of manufacturers at each location.

Which SELECT statement should you use?

```
SELECT location_id, COUNT(DISTINCT type)
FROM manufacturer
GROUP BY location_id;
```

Evaluate this SELECT statement:

```
SELECT COUNT(employee_id), department_id
```

```
FROM employees
```

```
GROUP BY department_id;
```

You only want to include employees who earn more than 15000.  
Which clause should you include in the SELECT statement?

**WHERE salary > 15000**

Evaluate this SELECT statement:

```
SELECT SUM(salary), department_id, manager_id  
FROM employees  
GROUP BY department_id, manager_id;
```

Which SELECT clause allows you to restrict the rows returned, based on a group function?

**HAVING SUM(salary) > 100000**

Group functions can be nested to a depth of?

**Two**

The EMPLOYEES table contains these columns:

ID\_NUMBER NUMBER Primary Key

NAME VARCHAR2 (30)

DEPARTMENT\_ID NUMBER

SALARY NUMBER (7,2)

HIRE\_DATE DATE

Evaluate this SQL statement:

```
SELECT id_number, name, department_id, SUM(salary)
```

```
FROM employees
```

```
WHERE salary > 25000
```

```
GROUP BY department_id, id_number, name
```

```
ORDER BY hire_date;
```

Why will this statement cause an error?

**The HIRE\_DATE column is NOT included in the GROUP BY clause.**

To control the order of rows returned using SET operators, the ORDER BY clause is used \_\_\_\_\_ and is placed in the \_\_\_\_\_ SELECT statement of the query.

**ONCE; LAST**

When using SET operators the names of the columns must be identified in all of the SELECT statements used in the query. True or False?

**False**

GROUPING SETS is another extension to the GROUP BY clause and is used to specify multiple groupings of data but provide a single result set. True or False?

**True**

CUBE can be applied to all aggregate functions including AVG, SUM, MIN, MAX, and COUNT. True or False?

True

You use ROLLUP to:

produce subtotal values

You use GROUPING function to:

Identify the extra row values created by either a ROLLUP or CUBE operation

Which statement is false?

The WITH clause decreases performance.

The Oracle server performs a correlated subquery when the subquery references a column from a table referred to in the parent. True or False?

True

Which answer is incorrect? The parent statement can be:

An INSERT statement

The WITH clause enables a SELECT statement to define the subquery block at the start of the query, process the block just once, label the results, and then refer to the results multiple times. True or False?

True

If a single-row subquery returns a null value and uses the equality comparison operator, what will the outer query return?

No rows

Which comparison operator can only be used with a single-row subquery?

<>

Which of the following is a valid reason why the query below will not execute successfully?

```
SELECT employee_id, last_name, salary
```

```
FROM employees
```

```
WHERE department_id =
```

```
(SELECT department_id FROM employees WHERE last_name like '%u%')
```

Single rather than multiple value operator used.

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

IN

You need to display all the players whose salaries are greater than or equal to John Brown's salary. Which comparison operator should you use?

>=

You need to create a report to display the names of products with a cost value greater than the average cost of all products. Which SELECT statement should you use?

```
SELECT product_name
FROM products
WHERE cost > (SELECT AVG(cost)
FROM product);
```

The TEACHERS and CLASS\_ASSIGNMENTS tables contain these columns:

TEACHERS  
TEACHER\_ID NUMBER(5) Primary Key  
NAME VARCHAR2 (25)  
SUBJECT\_ID NUMBER(5)

CLASS\_ASSIGNMENTS  
CLASS\_ID NUMBER (5) Primary Key  
TEACHER\_ID NUMBER (5)  
DATE  
MAX\_CAPACITY NUMBER (3)

All MAX\_CAPACITY values are greater than 10. Which two SQL statements correctly use subqueries?

```
SELECT *
FROM class_assignments
WHERE max_capacity = (SELECT AVG(max_capacity) FROM class_assignments);
```

```
SELECT *
FROM teachers
WHERE teacher_id = (SELECT teacher_id FROM teachers WHERE class_id = 45963);
```

If you use the equality operator (=) with a subquery, how many values can the subquery return?

Only 1

You need to remove a row from the EMPLOYEES table. Which statement would you use?

DELETE with a WHERE clause

You need to update both the DEPARTMENT\_ID and LOCATION\_ID columns in the EMPLOYEES table using one UPDATE statement. Which clause should you include in the UPDATE statement to update multiple columns?

The SET clause

What keyword in an UPDATE statement specifies the columns you want to change?

SET

Examine the structures of the PRODUCTS and SUPPLIERS tables:

SUPPLIERS

SUPPLIER\_ID NUMBER NOT NULL, Primary Key

SUPPLIER\_NAME VARCHAR2 (25)

ADDRESS VARCHAR2 (30)

CITY VARCHAR2 (25)

REGION VARCHAR2 (10)

POSTAL\_CODE VARCHAR2 (11)

PRODUCTS

PRODUCT\_ID NUMBER NOT NULL, Primary Key

PRODUCT\_NAME VARCHAR2 (25)

SUPPLIER\_ID NUMBER Foreign key to SUPPLIER\_ID of the SUPPLIERS table

CATEGORY\_ID NUMBER

QTY\_PER\_UNIT NUMBER

UNIT\_PRICE NUMBER (7,2)

QTY\_IN\_STOCK NUMBER

QTY\_ON\_ORDER NUMBER

REORDER\_LEVEL NUMBER

You want to delete any products supplied by the five suppliers located in Atlanta. Which

**DELETE FROM products**

**WHERE supplier\_id IN**

**(SELECT supplier\_id FROM suppliers WHERE UPPER(city) = 'ATLANTA');**

One of the sales representatives, Janet Roper, has informed you that she was recently married, and she has requested that you update her name in the employee database. Her new last name is Cooper. Janet is the only person with the last name of Roper that is employed by the company. The EMPLOYEES table contains these columns and all data is stored in lowercase:

EMPLOYEE\_ID NUMBER(10) PRIMARY KEY

LAST\_NAME VARCHAR2(20)

FIRST\_NAME VARCHAR2(20)

DEPARTMENT\_ID VARCHAR2 (20)

HIRE\_DATE DATE

SALARY NUMBER(10)

Which UPDATE statement will accomplish your objective?

**UPDATE employees**

**SET last\_name = 'cooper'**

**WHERE last\_name = 'roper';**

One of your employees was recently married. Her employee ID is still 189, however, her last name is now Rockefeller. Which SQL statement will allow you to reflect this change?

**UPDATE my\_employees SET last\_name = 'Rockefeller' WHERE employee\_ID = 189;**

Assume all the column names are correct. The following SQL statement will execute which of the following?

```
INSERT INTO departments (department_id, department_name, manager_id, location_id)
```

```
VALUES (70, 'Public Relations', 100, 1700);
```

70 will be inserted into the department\_id column

You need to display each employee's name in all uppercase letters. Which function should you use?

UPPER

You query the database with this SQL statement:

```
SELECT CONCAT(last_name, (SUBSTR(LOWER(first_name), 4))) "Default Password" FROM employees;
```

Which function will be evaluated first?

LOWER

Which SQL function is used to return the position where a specific character string begins within a larger character string?

INSTR

Which three statements about functions are true?

The SYSDATE function returns the Oracle Server date and time.

The ROUND number function rounds a value to a specified decimal place or the nearest whole number.

The SUBSTR character function returns a portion of a string beginning at a defined character position to a specified length.

The PRICE table contains this data:

```
PRODUCT_ID MANUFACTURER_ID
```

```
86950 59604
```

You query the database and return the value 95. Which script did you use?

```
SELECT SUBSTR(product_id, 3, 2)
```

```
FROM price
```

```
WHERE manufacturer_id = 59604;
```

You need to display the number of characters in each customer's last name. Which function should you use?

LENGTH

You issue this SQL statement:

```
SELECT INSTR ('organizational sales', 'al')
```

```
FROM dual;
```

Which value is returned by this command?

13

You issue this SQL statement:  
SELECT ROUND (1282.248, -2)  
FROM dual;  
What value does this statement produce?

1300

Evaluate this function: MOD (25, 2) Which value is returned?

1

Which two functions can be used to manipulate number or date column values, but NOT character column values?

TRUNC  
ROUND

You need to display the current year as a character value (for example: Two Thousand and One). Which element would you use?

YEAR

Which SELECT statement will NOT return a date value?

SELECT (SYSDATE - hire\_date) + 10\*8  
FROM employees;

You need to display the number of months between today's date and each employee's hiredate. Which function should you use?

MONTHS\_BETWEEN

You want to create a report that displays all orders and their amounts that were placed during the month of January. You want the orders with the highest amounts to appear first. Which query should you issue?

SELECT orderid, total  
FROM orders  
WHERE order\_date BETWEEN '01-jan-02' AND '31-jan-02'  
ORDER BY total DESC;

Evaluate this SELECT statement:

SELECT SYSDATE + 30  
FROM dual;  
Which value is returned by the query?

the current date plus 30 days

Which three statements concerning explicit data type conversions are true?

Use the TO\_DATE function to convert a character string to a date value.  
Use the TO\_NUMBER function to convert a character string of digits to a number.  
Use the TO\_CHAR function to convert a number or date value to character string.

The EMPLOYEES table contains these columns:

EMPLOYEE\_ID NUMBER(9)  
LAST\_NAME VARCHAR2 (25)  
FIRST\_NAME VARCHAR2 (25)  
SALARY NUMBER(6)

You need to create a report to display the salaries of all employees. Which script should you use to display the salaries in format: "\$45,000.00"?

```
SELECT TO_CHAR(salary, '$999,999.00')  
FROM employees;
```

Which statement concerning single row functions is true?

Single row functions can be nested.

Which SQL Statement should you use to display the prices in this format: "\$00.30"?

```
SELECT TO_CHAR(price, '$99,900.99') FROM product;
```

Which best describes the TO\_CHAR function?

The TO\_CHAR function can be used to display dates and numbers according to formatting conventions that are supported by Oracle.

The EMPLOYEES table contains these columns:

EMPLOYEE\_ID NUMBER(9)  
LAST\_NAME VARCHAR2 (25)  
FIRST\_NAME VARCHAR2 (25)  
HIRE\_DATE DATE

You need to display HIRE\_DATE values in this format: January 28, 2000

Which SELECT statement could you use?

```
SELECT TO_CHAR(hire_date, 'Month DD, YYYY')  
FROM employees;
```

Which of the following General Functions will return the first non-null expression in the expression list?

COALESCE

Which statement about group functions is true?

NVL, NVL2, and COALESCE can be used with group functions to replace null values.

The STYLES table contains this data:

STYLE_ID	STYLE_NAME	CATEGORY	COST
895840	SANDAL	85940	12.00
968950	SANDAL	85909	10.00
869506	SANDAL	89690	15.00
809090	LOAFER	89098	10.00
890890	LOAFER	89789	14.00
857689	HEEL	85940	11.00
758960	SANDAL	86979	

Evaluate this SELECT statement:



```
SELECT style_id, style_name, category, cost
FROM styles
WHERE style_name = 'SANDAL' AND NVL(cost, 0) < 15.00 ORDER BY category, cost;
Which result will the query provide?
STYLE_ID STYLE_NAME CATEGORY COST 968950 SANDAL 85909 10.00 895840
SANDAL 85940 12.00 758960 SANDAL 86979
```

When joining 3 tables in a SELECT statement, how many join conditions are needed in the

2

You have the following EMPLOYEES table:

```
EMPLOYEE_ID NUMBER(5) NOT NULL PRIMARY KEY
FIRST_NAME VARCHAR2(25)
LAST_NAME VARCHAR2(25)
ADDRESS VARCHAR2(35)
CITY VARCHAR2(25)
STATE VARCHAR2(2)
ZIP NUMBER(9)
TELEPHONE NUMBER(10)
DEPARTMENT_ID NUMBER(5) NOT NULL FOREIGN KEY
```

The BONUS table includes the following columns:

```
BONUS_ID NUMBER(5) NOT NULL PRIMARY KEY
ANNUAL_SALARY NUMBER(10)
BONUS_PCT NUMBER(3, 2)
EMPLOYEE_ID VARCHAR2(5) NOT NULL FOREIGN KEY
```

You want to determine the amount of each employee's bonus, as a calculation of salary times bonus. Which of the following queries should you issue?

```
SELECT e.first_name, e.last_name, b.annual_salary * b. bonus_pct FROM employees
e, bonus b WHERE e.employee_id = b.employee_id;
```

You have been asked to create a report that lists all corporate customers and all orders that they have placed. The customers should be listed alphabetically beginning with the letter 'A', and their corresponding order totals should be sorted from the highest amount to the lowest amount. Which of the following statements should you issue?

```
SELECT c.custid, c.companyname, o.orderdate, o. custid, o.amount FROM customers
c, orders o WHERE c.custid = o.custid ORDER BY companyname, amount DESC;
```

Evaluate this SQL statement:

```
SELECT e.employee_id, e.last_name, e.first_name, d.department_name
FROM employees e, departments d
WHERE e.department_id = d.department_id AND employees.department_id > 5000
ORDER BY 4;
```

Which clause contains a syntax error?

```
AND employees.department_id > 5000 (*)
```

You need to create a report that lists all employees in department 10 (Sales) whose salary is not equal to \$25,000 per year. Which query should you issue to accomplish this task?

```
SELECT last_name, first_name, salary
FROM employees
WHERE salary != 25000 AND department_id = 10;
```

What happens when you create a Cartesian product?

All rows from one table are joined to all rows of another table

Which statement about outer joins is true?

The OR operator cannot be used to link outer join conditions.

The EMPLOYEE\_ID column in the EMPLOYEES table corresponds to the EMPLOYEE\_ID column of the ORDERS table. The EMPLOYEE\_ID column in the ORDERS table contains null values for rows that you need to display. Which type of join should you use to display the data?

outer join

Using Oracle Proprietary join syntax, which two operators can be used in an outer join condition using the outer join operator (+)?

AND and =

Which statement about a natural join is true?

Columns with the same names must have the same precision and datatype.

Which of the following best describes a natural join?

A join between two tables that includes columns that share the same name, datatypes and length

The following SQL statement will produce what output?

```
SELECT last_name, department_name
FROM employees
CROSS JOIN departments;
```

A Cartesian product between the two tables.

For which condition would you use an equijoin query with the USING keyword?

You need to perform a join of the CUSTOMER and ORDER tables but limit the number of columns in the join condition.

Below find the structure of the CUSTOMERS and SALES\_ORDER tables:

```
CUSTOMERS
CUSTOMER_ID NUMBER NOT NULL, Primary Key
CUSTOMER_NAME VARCHAR2 (30)
CONTACT_NAME VARCHAR2 (30)
CONTACT_TITLE VARCHAR2 (20)
```

ADDRESS VARCHAR2 (30)  
CITY VARCHAR2 (25)  
REGION VARCHAR2 (10)  
POSTAL\_CODE VARCHAR2 (20)  
COUNTRY\_ID NUMBER Foreign key to COUNTRY\_ID column of the COUNTRY table  
PHONE VARCHAR2 (20)  
FAX VARCHAR2 (20)  
CREDIT\_LIMIT NUMBER(7,2)

SALES\_ORDER  
ORDER\_ID NUMBER NOT NULL, Primary Key  
CUSTOMER\_ID NUMBER Foreign key to CUSTOMER\_ID column of the CUSTOMER table

ORDER\_DT DATE

ORDER\_AMT NUMBER (7,2)

SHIP\_METHOD VARCHAR2 (5)

You need to create a report that displays customers without a sales order. Which statement could you use?

```
SELECT c.customer_name  
FROM customers c  
WHERE c.customer_id not in (SELECT s.customer_id FROM sales_order s);
```

The primary advantage of using JOIN ON is:

It permits columns with different names to be joined

Evaluate this SELECT statement:

```
SELECT a.last_name || ', ' || a.first_name as "Patient", b.last_name || ', ' || b.first_name  
as "Physician", c.admission
```

```
FROM patient a
```

```
JOIN physician b
```

```
ON (b.physician_id = c.physician_id)
```

```
JOIN admission c
```

```
ON (a.patient_id = c.patient_id);
```

Which clause generates an error?

ON (b.physician\_id = c.physician\_id)

What should be included in a SELECT statement to return NULL values from all tables?

full outer joins

Which query will retrieve all the rows in the EMPLOYEES table, even if there is no match in the DEPARTMENTS table?

```
SELECT e.last_name, e.department_id, d.department_name
```

```
FROM employees e
```

```
LEFT OUTER JOIN departments d ON (e.department_id = d.department_id);
```

Which type of join returns rows from one table that have NO direct match in the other

table?

outer join

Evaluate this SELECT statement:

```
SELECT MIN(hire_date), department_id  
FROM employees  
GROUP BY department_id;
```

Which values are displayed?

The earliest hire date in each department.

Which statement about the GROUP BY clause is true?

The first column listed in the GROUP BY clause is the most major grouping.

Evaluate this SELECT statement:

```
SELECT MAX(salary), department_id  
FROM employees  
GROUP BY department_id;
```

Which values are displayed?

The highest salary in each department.

What will the following SQL Statement do?

```
SELECT job_id, COUNT(*)  
FROM employees  
GROUP BY job_id;
```

Displays each job id and the number of people assigned to that job id.

Which group function would you use to display the lowest value in the SALES\_AMOUNT column?

MIN

Group functions return a value for \_\_\_\_\_ and \_\_\_\_\_ null values in their computations.

a row set, ignore

The EMPLOYEES table contains these columns:

```
EMPLOYEE_ID NUMBER(9)  
LAST_NAME VARCHAR2(20)  
FIRST_NAME VARCHAR2(20)  
SALARY NUMBER(9,2)  
HIRE_DATE DATE  
BONUS NUMBER(7,2)  
COMM_PCT NUMBER(4,2)
```

Which three functions could be used with the HIRE\_DATE, LAST\_NAME, or SALARY columns?

MAX

MIN

COUNT

Which group function would you use to display the total of all salary values in the EMPLOYEE table?

**SUM**

You need to calculate the standard deviation for the cost of products produced in the Birmingham facility. Which group function will you use?

**STDDEV**

You need to calculate the average salary of employees in each department. Which group function will you use?

**AVG**

You need to compute the total salary for all employees in department 10. Which group function will you use?

**SUM**

Which aggregate function can be used on a column of the DATE data type?

**MAX**

Evaluate this SQL statement:

```
SELECT COUNT (amount)
```

```
FROM inventory;
```

What will occur when the statement is issued?

**The statement will count the number of rows in the INVENTORY table where the AMOUNT column is not null.**

Which SELECT statement will calculate the number of rows in the PRODUCTS table?

**SELECT COUNT (\*) FROM products;**

The EMPLOYEES table contains these columns:

```
EMPLOYEE_ID NUMBER(9)
```

```
LAST_NAME VARCHAR2(20)
```

```
FIRST_NAME VARCHAR2(20)
```

```
SALARY NUMBER(7,2)
```

```
DEPARTMENT_ID NUMBER(9)
```

You need to display the number of employees whose salary is greater than \$50,000?

Which SELECT would you use?

**SELECT \* FROM employees**

**WHERE salary < 50000; SELECT COUNT(\*) FROM employees WHERE salary**

**< 50000; SELECT COUNT(\*) FROM employees WHERE salary > 50000;**

Examine the data from the LINE\_ITEM table:

```
LINE_ITEM_ID ORDER_ID PRODUCT_ID PRICE DISCOUNT
```

```
890898 847589 848399 8.99 0.10
```

```
768385 862459 849869 5.60 0.05
```

867950 985490 945809 5.60  
954039 439203 438925 5.25 0.15  
543949 349302 453235 4.50

You query the LINE\_ITEM table and a value of 3 is returned. Which SQL statement did you execute?

**SELECT COUNT(discount) FROM line\_item;**

Evaluate this SELECT statement:

```
SELECT COUNT(emp_id), mgr_id, dept_id
FROM employee
WHERE status = 'I'
GROUP BY dept_id
HAVING salary > 30000
ORDER BY 2;
```

Why does this statement return a syntax error?

**MGR\_ID must be included in the GROUP BY clause.**

The PRODUCTS table contains these columns:

```
PROD_ID NUMBER(4)
PROD_NAME VARCHAR(20)
PROD_CAT VARCHAR2(15)
PROD_PRICE NUMBER(5)
PROD_QTY NUMBER(4)
```

You need to identify the minimum product price in each product category. Which statement could you use to accomplish this task?

**SELECT prod\_cat, MIN (prod\_price)
FROM products
GROUP BY prod\_cat;**

You want to write a report that returns the average salary of all employees in the company, sorted by departments. The EMPLOYEES table contains the following columns:

```
EMPLOYEES:
EMPLOYEE_ID NUMBER(10) PRIMARY KEY
LAST_NAME VARCHAR2(20)
FIRST_NAME VARCHAR2(20)
DEPARTMENT VARCHAR2(20)
HIRE_DATE DATE
SALARY NUMBER(10)
```

Which SELECT statement will return the information that you require?

**SELECT AVG (salary)
FROM employees
GROUP BY department;**

Evaluate this SELECT statement:

```
SELECT SUM(salary), department_id, department_name
```

```
FROM employees
WHERE department_id = 1
GROUP BY department;
Which clause of the SELECT statement contains a syntax error?
GROUP BY
```

The EMPLOYEES table contains the following columns:

```
EMPLOYEE_ID NUMBER(10) PRIMARY KEY
LAST_NAME VARCHAR2(20)
FIRST_NAME VARCHAR2(20)
DEPARTMENT VARCHAR2(20)
HIRE_DATE DATE
SALARY NUMBER(10)
```

You want to create a report that includes each employee's last name, employee identification number, date of hire and salary. The report should include only those employees who have been with the company for more than one year and whose salary exceeds \$40,000. Which of the following SELECT statements will accomplish this task?

```
SELECT employee_id, last_name, hire_date, salary
FROM employees
WHERE salary > 40000
AND (sysdate-hire_date) / 365 > 1;
```

The PRODUCTS table contains these columns:

```
PRODUCT_ID NUMBER(9) PK
CATEGORY_ID VARCHAR2(10)
LOCATION_ID NUMBER(9)
DESCRIPTION VARCHAR2(30)
COST NUMBER(7,2)
PRICE NUMBER(7,2)
QUANTITY NUMBER
```

You display the total of the extended costs for each product category by location. You need to include only the products that have a price less than \$25.00. The extended cost of each item equals the quantity value multiplied by the cost value. Which SQL statement will display the desired result?

```
SELECT category_id, SUM(cost * quantity) TOTAL, location_id FROM products
WHERE price < 25.00 GROUP BY category_id, location_id;
```

Which statement about the GROUP BY clause is true?

**To exclude rows before dividing them into groups using the GROUP BY clause, you should use a WHERE clause.**

The PLAYERS and TEAMS tables contain these columns:

```
PLAYERS PLAYER_ID NUMBER NOT NULL, Primary Key LAST_NAME VARCHAR2
(30) NOT NULL FIRST_NAME VARCHAR2 (25) NOT NULL TEAM_ID NUMBER
POSITION VARCHAR2 (25)
TEAMS TEAM_ID NUMBER NOT NULL, Primary Key TEAM_NAME VARCHAR2 (25)
```

You need to create a report that lists the names of each team with more than three goal keepers. Which SELECT statement will produce the desired result?

```
SELECT t.team_name, COUNT(p.player_id)
FROM players p JOIN teams t ON (p.team_id = t.team_id)
WHERE UPPER(p.position) = 'GOAL KEEPER'
GROUP BY t.team_name HAVING COUNT(p.player_id) > 3;
```

The EMPLOYEES and ORDERS tables contain these columns:

EMPLOYEES

EMPLOYEE\_ID NUMBER(10) NOT NULL PRIMARY KEY

FIRST\_NAME VARCHAR2(30)

LAST\_NAME VARCHAR2(30)

ADDRESS VARCHAR2(25)

CITY VARCHAR2(20)

STATE VARCHAR2(2)

ZIP NUMBER(9)

TELEPHONE NUMBER(10)

ORDERS

ORDER\_ID NUMBER(10) NOT NULL PRIMARY KEY

EMPLOYEE\_ID NUMBER(10) NOT NULL FOREIGN KEY

ORDER\_DATE DATE

TOTAL NUMBER(10)

Which SELECT statement will return all orders generated by a sales representative named Franklin during the year 2001?

```
SELECT order_id, total
FROM ORDERS
WHERE employee_id = (SELECT employee_id FROM employees WHERE last_name = 'Franklin')
AND order_date BETWEEN '01-jan-01' AND '31-dec-01';
```

Using a subquery in which of the following clauses will return a syntax error?

You can use subqueries in all of the above clauses.

Which of the following is TRUE regarding the order of subquery execution?

The subquery executes once before the main query

Which statement about subqueries is true?

Subqueries are often used in a WHERE clause to return values for an unknown conditional value.

Which operator can be used with subqueries that return only one row?

LIKE

Examine the structure of the EMPLOYEE, DEPARTMENT, and ORDERS tables.

EMPLOYEE



EMPLOYEE\_ID NUMBER(9)  
LAST\_NAME VARCHAR2(25)  
FIRST\_NAME VARCHAR2(25)  
DEPARTMENT\_ID NUMBER(9)

DEPARTMENT  
DEPARTMENT\_ID NUMBER(9)  
DEPARTMENT\_NAME VARCHAR2(25)  
CREATION\_DATE DATE

ORDERS  
ORDER\_ID NUMBER(9)  
EMPLOYEE\_ID NUMBER(9)  
DATE DATE  
CUSTOMER\_ID NUMBER(9)

You want to display all employees who had an order after the Sales department was established. Which of the following constructs would you use?

**a single-row subquery**

Examine the following EMPLOYEES table:

EMPLOYEES  
EMPLOYEE\_ID NUMBER(9)  
LAST\_NAME VARCHAR2(25)  
FIRST\_NAME VARCHAR2(25)  
DEPARTMENT\_ID NUMBER(9)  
SUPERVISOR\_ID NUMBER(9)

You need to produce a report that contains all employee-related information for those employees who have Brad Carter as a supervisor. However, you are not sure which supervisor ID belongs to Brad Carter. Which query should you issue to accomplish this task?

**SELECT \*  
FROM employees  
WHERE supervisor\_id =  
(SELECT employee\_id  
FROM employees  
WHERE last\_name = 'Carter');**

Which best describes a single-row subquery?

**a query that returns only one row from the inner SELECT statement**

Which best describes a multiple-row subquery?

**A query that returns one or more rows from the inner SELECT statement**

What is wrong with the following query?

SELECT employee\_id, last\_name  
FROM employees

WHERE salary =  
(SELECT MIN(salary) FROM employees GROUP BY department\_id);  
Subquery returns more than one row and single row comparison operator is used.

Evaluate this SELECT statement:

```
SELECT customer_id, name  
FROM customer  
WHERE customer_id IN  
(SELECT customer_id  
FROM customer  
WHERE state_id = 'GA' AND credit_limit > 500.00);
```

What would happen if the inner query returned null?

No rows would be returned by the outer query.

Evaluate this SQL statement:

```
SELECT employee_id, last_name, salary  
FROM employees  
WHERE department_id IN  
(SELECT department_id  
FROM employees  
WHERE salary > 30000 AND salary < 50000);
```

Which values will be displayed?

All employees who work in a department with employees who earn more than \$30,000, but less than \$50,000.

Evaluate this SELECT statement:

```
SELECT student_id, last_name, first_name FROM student WHERE major_id NOT IN  
(SELECT major_id FROM majors WHERE department_head_id = 30 AND title =  
'ADJUNCT');
```

What would happen if the inner query returned a NULL value row?

No rows would be returned from the STUDENT table.

Which of the following statements contains a comparison operator that is used to restrict rows based on a list of values returned from an inner query?

All of the above.

A multiple-row operator expects how many values?

One or more

What would happen if you attempted to use a single-row operator with a multiple-row subquery?

An error would be returned.

Which operator or keyword cannot be used with a multiple-row subquery?

=

You need to display all the products that cost more than the maximum cost of every product produced in Japan. Which multiple-row comparison operator could you use?  
>ANY

Evaluate the structure of the EMPLOYEE and DEPART\_HIST tables:

EMPLOYEE:

EMPLOYEE\_ID NUMBER(9)  
LAST\_NAME VARCHAR2(25)  
FIRST\_NAME VARCHAR2(25)  
DEPARTMENT\_ID NUMBER(9)  
MANAGER\_ID NUMBER(9)  
SALARY NUMBER(7,2)

DEPART\_HIST:

EMPLOYEE\_ID NUMBER(9)  
OLD\_DEPT\_ID NUMBER(9)  
NEW\_DEPT\_ID NUMBER(9)  
CHANGE\_DATE DATE

You want to generate a list of employees who are in department 10, but used to be in department 15.

Which query should you use?

```
SELECT employee_id, last_name, first_name, department_id
FROM employee
WHERE (employee_id, department_id) IN
(SELECT employee_id, new_dept_id
FROM depart_hist
WHERE old_dept_id = 15) AND new_dept_id = 10;
```

The PRODUCTS table contains these columns:

PRODUCT\_ID NUMBER NOT NULL  
PRODUCT\_NAME VARCHAR2 (25)  
SUPPLIER\_ID NUMBER NOT NULL  
LIST\_PRICE NUMBER (7,2)  
COST NUMBER (5,2)  
QTY\_IN\_STOCK NUMBER(4)  
LAST\_ORDER\_DT DATE NOT NULL DEFAULT SYSDATE

Which INSERT statement will execute successfully?

```
INSERT INTO products VALUES (2958, 'Cable', 8690, 7.09, 4.04, 700);
```

The PRODUCTS table contains these columns:

PROD\_ID NUMBER(4)  
PROD\_NAME VARCHAR2(25)  
PROD\_PRICE NUMBER(3)

You want to add the following row data to the PRODUCTS table:

- (1) a NULL value in the PROD\_ID column
- (2) "6-foot nylon leash" in the PROD\_NAME column

(3) "10" in the PROD\_PRICE column

You issue this statement:

```
INSERT INTO products  
VALUES (null, '6-foot nylon leash', 10);  
What row data did you add to the table?
```

The row was created with the correct data in all three columns.

You have been instructed to add a new customer to the CUSTOMERS table. Because the new customer has not had a credit check, you should not add an amount to the CREDIT column.

The CUSTOMERS table contains these columns:

```
CUST_ID NUMBER(10)  
COMPANY VARCHAR2(30)  
CREDIT NUMBER(10)  
POC VARCHAR2(30)  
LOCATION VARCHAR2(30)
```

Which two INSERT statements will accomplish your objective?

```
INSERT INTO customers (cust_id, company, poc, location)  
VALUES (200, 'InterCargo', 'tflanders', 'samerica');
```

```
INSERT INTO customers  
VALUES (200, 'InterCargo', null, 'tflanders', 'samerica');
```

Which statement about the VALUES clause of an INSERT statement is true?

If no column list is specified, then the values must be in the order the columns are specified in the table.

You need to update the area code of employees that live in Atlanta . Evaluate this partial UPDATE statement:

```
UPDATE employee  
SET area_code = 770
```

Which of the following should you include in your UPDATE statement to achieve the desired results?

```
WHERE city = 'Atlanta';
```

What would happen if you issued a DELETE statement without a WHERE clause?

All the rows in the table would be deleted.

You need to delete a record in the EMPLOYEES table for Tim Jones, whose unique employee identification number is 348. The EMPLOYEES table contains these columns:

```
EMPLOYEE_ID NUMBER(5) PRIMARY KEY  
LAST_NAME VARCHAR2(20)  
FIRST_NAME VARCHAR2(20)  
ADDRESS VARCHAR2(30)  
PHONE NUMBER(10)
```

Which DELETE statement will delete the appropriate record without deleting any additional records?

**DELETE FROM employees WHERE employee\_id = 348;**

Which two commands can be used to modify existing data in a database row?

**MERGE  
UPDATE**

When the WHERE clause is missing in a DELETE statement, what is the result?

**All rows are deleted from the table.**

Which of the following represents the correct syntax for an INSERT statement?

**INSERT INTO customers VALUES (3178, 'J.', 'Smith', '123 Main Street', 'Nashville', 'TN', '37777');**

The EMPLOYEES table contains the following columns:

EMPLOYEE\_ID NUMBER(10) PRIMARY KEY  
LAST\_NAME VARCHAR2(20)  
FIRST\_NAME VARCHAR2(20)  
DEPARTMENT\_ID NUMBER(10)  
HIRE\_DATE DATE  
SALARY NUMBER(9,2)  
BONUS NUMBER(9,2)

You want to execute one DML statement to change the salary of all employees in department 10 to equal the new salary of employee id 89898. Currently, all employees in department 10 have the same salary value. Which statement should you execute?

**UPDATE employees  
SET salary = (SELECT salary FROM employees WHERE employee\_id = 89898)  
WHERE department\_id = 10;**

You want to enter a new record into the CUSTOMERS table. Which two commands can be used to create new rows?

**INSERT, MERGE**

Examine the structures of the PLAYERS, MANAGERS, and TEAMS tables:

PLAYERS  
PLAYER\_ID NUMBER Primary Key  
LAST\_NAME VARCHAR2 (30)  
FIRST\_NAME VARCHAR2 (25)  
TEAM\_ID NUMBER  
MGR\_ID NUMBER  
SIGNING\_BONUS NUMBER(9,2)  
SALARY NUMBER(9,2)

MANAGERS  
MANAGER\_ID NUMBER Primary Key

LAST\_NAME VARCHAR2 (20)  
FIRST\_NAME VARCHAR2 (20)  
TEAM\_ID NUMBER

TEAMS

TEAM\_ID NUMBER Primary Key  
TEAM\_NAME VARCHAR2 (20)  
OWNER\_LAST\_NAME VARCHAR2 (20)  
OWNER\_FIRST\_NAME VARCHAR2 (20)

Which situation would require a subquery to return the desired result?

To display the names of the managers for all the teams owned by a given owner

Evaluate this statement: DELETE FROM customer; Which statement is true?

The statement deletes all the rows from the CUSTOMER table