Section 5 Programming

USING GROUP BY AND HAVING CLAUSES

Is the following statement correct?

```
SELECT department_id, AVG(salary)
FROM employees;
```

No, because a GROUP BY department_id clause is needed

Which of the following SQL statements could display the number of people with the same last name:

```
SELECT last_name, COUNT(last_name)
FROM EMPLOYEES
GROUP BY last_name;
```

How would you alter the following query to list only employees where there existed more than one per last_name with the same last name:

```
SELECT last_name, COUNT(employee_id)
FROM EMPLOYEES
GROUP BY last_name;
```

```
SELECT last_name, COUNT(last_name)
FROM EMPLOYEES
GROUP BY last_name
HAVING COUNT(last_name) > 1;
```

Read the following SELECT statement. Choose the column or columns that must be included in the GROUP BY clause.

```
SELECT COUNT(last_name), grade, gender
FROM STUDENTS
GROUP_BY ?????;
```

grade, gender

Is the following statement correct:

```
SELECT first_name, last_name, salary, department_id, COUNT(employee_id)
FROM employees
WHERE department_id = 50
GROUP BY last_name, first_name, department_id;
```

No, because the statement is missing salary in the GROUP BY clause

USING ROLLUP AND CUBE OPERATIONS, AND GROUPING SETS

Examine the following statement:

```
SELECT department_id, manager_id, job_id, SUM(salary)
```

FROM employees
GROUP BY GROUPING SETS(.......);
Select the correct GROUP BY GROUPING SETS clause from the following list:
GROUP BY GROUPING SETS ((department_id, manager_id), (department_id, job_id), (manager_id, job_id))

Examine the following statement:
SELECT department_id, manager_id, job_id, SUM(salary)
FROM employees
GROUP BY ROLLUP(department_id, manager_id)
What extra data will this query generate?
The statement will fail.

Examine the following statement:
SELECT department_id, manager_id, job_id, SUM(salary)
FROM employees
GROUP BY GROUPING SETS((department_id, manager_id), (department_id, job_id))
What data will this query generate?
Sum of salaries for (department_id, job_id) and (department_id, manager_id)

If you want to include subtotals and grand totals for all columns mentioned in a GROUP BY clause you should use which of the following extensions to the GROUP BY clause?
CUBE

USING SET OPERATORS

MINUS will give you rows from the first query not present in the second query? (True or False)
True

INTERSECT will give you the common rows found in both queries? (True or False)
True

Which ones of the following are correct SET operators?
UNION, MINUS
UNION ALL, INTERSECTION

The difference between UNION and UNION ALL is
UNION will remove duplicates, UNION ALL returns all rows from all queries