Task 1: Using Comparison and Logical Operators

Question:
Write a SQL query to retrieve the `emp_id`, `last_name`, and `salary` of employees whose salary is between 2,000 and 5,000 and do not have a manager ID of 101 or 200.

Instructions:

1. Use the `SELECT` statement to specify the columns: `emp_id`, `last_name`, and `salary`.
2. Filter the results using the `WHERE` clause with the `BETWEEN` operator to set the salary range.
3. Use the `NOT IN` clause to exclude certain manager IDs.
4. Combine conditions using the `AND` logical operator.

```
SELECT emp_id, last_name, salary FROM employees WHERE salary BETWEEN
2000 AND 5000 AND emp_id NOT IN (101,200);
```

Task 2: Using JOINs and Aliases

Question:
Write a SQL query to display the employee names along with their respective department names. Use aliases for table names for better readability.

Instructions:

1. Use the `SELECT` statement to specify the columns: `employee.name` and `department.name`.
2. Use the `FROM` clause to include the tables `employees` and `departments`.
3. Use an `INNER JOIN` to connect the `employees` and `departments` tables based on the department IDs.
4. Use table aliases (e.g., `e` for employees, `d` for departments) to shorten the table names in the query.
5. Order the results by department name in ascending order.

```
SELECT  e.name, d.name  FROM employees e INNER JOIN departments d ON e.id = d.eid
ORDER  BY d.name ASC ;
```

Task 3: Aggregate Functions and GROUP BY

Question:
Write a SQL query to find the number of employees and the average salary for each department. Ensure that the results are grouped by department ID.

Instructions:

1. Use the `SELECT` statement to specify the department ID, the count of employees, and the average salary.
2. Use the `GROUP BY` clause to group the results by department ID.
3. Use the `COUNT()` function to find the number of employees in each department.
4. Use the `AVG()` function to calculate the average salary in each department.

SELECT COUNT(id), AVG(salary) FROM employees GROUP BY did;