Brightlearn Data Analytics

Exercise 2: SQL Aggregate Functions & SQL Operators

1. List all distinct departments in the students table.
   SQL QUERY:

   ```
   SELECT DISTINCT department
   FROM students;
   ```

   Expected Output:

Department
IT
HR
Finance

2. Get the average age of students per department.
   SQL QUERY:

   ```
   SELECT department, AVG(age) AS avg_age
   FROM students
   GROUP BY department;
   ```

   Expected Output:

department	Avg_age
IT	20.5
HR	22.0
Finance	23.0

3. Show departments with more than 1 student.
   SQL QUERY:

   ```
   SELECT department, COUNT(*) AS student_count
   FROM students
   GROUP BY department
   HAVING COUNT(*) > 1;
   ```

   Expected Output:

department	Student_count
IT	2
HR	2

4. Get all students whose age is between 21 and 23.
   SQL QUERY:

```
SELECT student_id, name, age, department
FROM students
WHERE age BETWEEN 21 AND 23;
```

Expected Output:

Student_id	name	age	department
2	Bob	22	HR
3	Charlie	21	IT
4	Diana	23	Finance
5	Eve	22	HR

5. List all students in the IT or HR department who are older than 21.
SQL QUERY:

```
SELECT student_id, name, age, department
FROM students
WHERE (department IN ('IT','HR')) AND age > 21;
```

Expected Output:

Student_id	name	age	department
2	Bob	22	HR
5	Eve	22	HR

6. Show total credits per department, only for departments with more than 5 total credits.
SQL QUERY:

```
SELECT department, SUM(credits) AS total_credits
FROM courses
GROUP BY department
HAVING SUM(credits) > 5;
```

Expected Output:

department	Total_credits
IT	11

7. List all courses that do not have 4 credits.
SQL QUERY:

```
SELECT course_id, course_name, department, credits
FROM courses
WHERE credits <> 4;
```

Expected Output:

Course_id	Course_name	department	credits
101	SQL Basics	IT	3
104	Excel	Finance	2
105	Statistics	HR	3

8. Show the top 3 courses by credits in descending order.
SQL QUERY:

```
SELECT course_id, course_name, credits
FROM courses
ORDER BY credits DESC
LIMIT 3;
```

Expected Output:

Course_id	Course_name	credits
102	Python	4
103	Data Science	4
101	SQL Basics	3

9. Get the maximum, minimum, and average grade across all enrollments.
SQL QUERY:

```
SELECT MAX(grade) AS max_grade,
       MIN(grade) AS min_grade,
       AVG(grade) AS avg_grade
FROM enrollments;
```

Expected Output:

Max_grade	Min_grade	Avg_grade
90	78	84.6

10. Count how many enrollments exist per course.
SQL QUERY:

```
SELECT course_id, COUNT(*) AS enrollment_count
FROM enrollments
GROUP BY course_id;
```

Expected Output:

Course_id	Enrolment_count
101	1
102	1
103	1
104	1
105	1

11. Find total salary and total bonus per department.
SQL QUERY:

```
SELECT department,
       SUM(salary) AS total_salary,
       SUM(bonus) AS total_bonus
FROM salaries
GROUP BY department;
```

Expected Output:

department	Total_salary	Total_bonus
IT	122000	10500
HR	109000	7500
Finance	70000	6000

12. Show departments where average salary is above 55,000.

SQL QUERY:

```
SELECT department, AVG(salary) AS avg_salary
FROM salaries
GROUP BY department
HAVING AVG(salary) > 55000;
```

Expected Output:

department	avg_salary
IT	61000
Finance	70000

13. List employees whose salary plus bonus is greater than 60,000.

SQL QUERY:

```
SELECT employee_id, name, salary, bonus,
       (salary + bonus) AS total_compensation
FROM salaries
WHERE (salary + bonus) > 60000;
```

Expected Output:

Employee_id	name	salary	bonus	Total_compensation
1	Tom	60000	5000	65000
3	Spike	70000	6000	76000
4	Tyke	62000	5500	67500

14. Show total and average budget per department. Only include departments with average budget above 70,000.

SQL QUERY:

```
SELECT department,
       SUM(budget) AS total_budget,
       AVG(budget) AS avg_budget
FROM projects
GROUP BY department
HAVING AVG(budget) > 70000;
```

Expected Output:

department	Total_budget	Avg_budget
IT	270000	135000
Finance	80000	80000

15. List all projects with budgets between 50,000 and 120,000, excluding the Marketing department.

SQL QUERY:

```
SELECT project_id, project_name, department, budget
FROM projects
WHERE budget BETWEEN 50000 AND 120000
  AND department <> 'Marketing';
```

Expected Output:

Project_id	Project_name	department	budget
1	AI App	IT	120000
2	Payroll System	Finance	80000
5	HR Portal	HR	50000