SQL Coding Assessment - 21.07.25

Table Creation and Inserting Values:

```
use CollegeDB;
CREATE TABLE Students (
  student_id INT PRIMARY KEY,
  name VARCHAR(50),
  department_id INT
);
CREATE TABLE Marks (
  mark_id INT PRIMARY KEY,
  student_id INT,
  subject VARCHAR(50),
  marks INT,
  FOREIGN KEY (student_id) REFERENCES Students(student_id)
);
INSERT INTO Students VALUES (1, 'Aarav', 101);
INSERT INTO Students VALUES (2, 'Ishita', 102);
INSERT INTO Students VALUES (3, 'Ravi', 101);
INSERT INTO Students VALUES (4, 'Meena', 103);

INSERT INTO Marks VALUES (1, 1, 'Maths', 85);
INSERT INTO Marks VALUES (2, 1, 'English', 78);
INSERT INTO Marks VALUES (3, 2, 'Maths', 92);
INSERT INTO Marks VALUES (4, 3, 'English', 60);
INSERT INTO Marks VALUES (5, 4, 'Maths', 55);
```

Results | Messages

	student_id	name	department_id
1	1	Aarav	101
2	2	Ishita	102
3	3	Ravi	101
4	4	Me...	103

	mark_id	student_id	subject	marks
1	1	1	Maths	85
2	2	1	English	78
3	3	2	Maths	92
4	4	3	English	60
5	5	4	Maths	55

1. Find average marks for each department

```
SELECT s.department_id, AVG(m.marks) AS average_marks
FROM Students s
JOIN Marks m ON s.student_id = m.student_id
GROUP BY s.department_id;
```

Output :

Results | Messages

	department_id	average_marks
1	101	74
2	102	92
3	103	55

This query joins Students and Marks using student_id. It groups the data by department_id so all students in one department are grouped together. Then AVG() calculates the average marks scored by students in each department.

2. Find minimum marks in each department

```
SELECT s.department_id, MIN(m.marks) AS min_marks
FROM Students s
JOIN Marks m ON s.student_id = m.student_id
GROUP BY s.department_id;
```

Output :

Results | Messages

	department_id	min_marks
1	101	60
2	102	92
3	103	55

This time I am using MIN() , to find the lowest mark scored in each department.

3. Find maximum marks in each department

```
SELECT s.department_id, MAX(m.marks) AS max_marks
FROM Students s
JOIN Marks m ON s.student_id = m.student_id
GROUP BY s.department_id;
```

Output :

Results | Messages

	department_id	max_marks
1	101	85
2	102	92
3	103	55

This Finds the highest marks scored by the student in each department using MAX().

4. Total marks scored by each department

```
SELECT s.department_id, SUM(m.marks) AS total_marks
FROM Students s
JOIN Marks m ON s.student_id = m.student_id
GROUP BY s.department_id;
```

Output :

Results | Messages

	department_id	total_marks
1	101	223
2	102	92
3	103	55

I am adding up all the marks scored by students in each department using SUM().

5. How many subjects were written per department

```
SELECT s.department_id, COUNT(m.subject) AS total_subjects
FROM Students s
JOIN Marks m ON s.student_id = m.student_id
GROUP BY s.department_id;
```

Output :

Results | Messages

	department_id	total_subjects
1	101	3
2	102	1
3	103	1

It counts how many subjects exist per department. It helps understand how active each department is.

6. Average marks only for students who scored 60 or more

```sql
SELECT s.department_id, AVG(m.marks) AS avg_passing_marks
FROM Students s
JOIN Marks m ON s.student_id = m.student_id
WHERE m.marks >= 60
GROUP BY s.department_id;
```

Output :

Results | Messages

	department_id	avg_passing_marks
1	101	74
2	102	92

I have added a filter so only students who scored 60 or more are considered.This is useful when you want to calculate averages for passing students only.