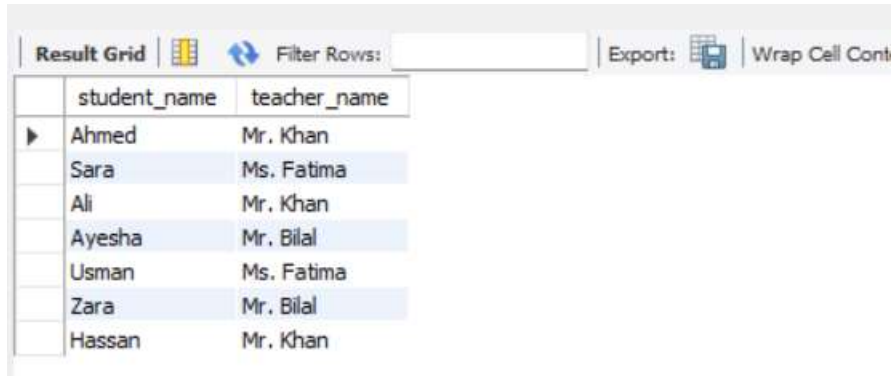


MYSQL TASK OUTPUTS

8. Show student names with their teacher's name for each class.

```
SELECT s.name AS student_name, t.name AS teacher_name  
FROM Students s  
JOIN Classes c ON s.class_id = c.class_id  
JOIN Teachers t ON c.teacher_id = t.teacher_id;
```

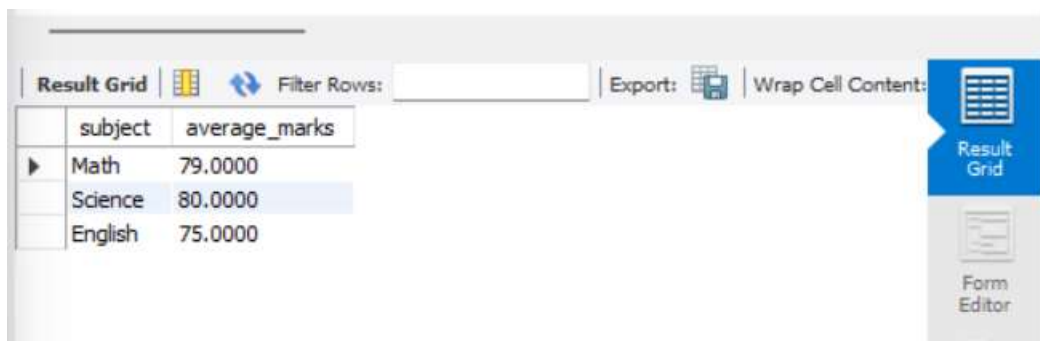


The screenshot shows a database application interface with a 'Result Grid' tab. The grid displays the results of a query, showing student names and their corresponding teacher names. The interface includes a 'Filter Rows' field, an 'Export' button, and a 'Wrap Cell Content' option.

student_name	teacher_name
Ahmed	Mr. Khan
Sara	Ms. Fatima
Ali	Mr. Khan
Ayesha	Mr. Bilal
Usman	Ms. Fatima
Zara	Mr. Bilal
Hassan	Mr. Khan

9. Find the average marks for each subject.

```
SELECT subject, AVG(marks) AS average_marks  
FROM Marks  
GROUP BY subject;
```



The screenshot shows a database application interface with a 'Result Grid' tab. The grid displays the results of a query, showing the average marks for each subject. The interface includes a 'Filter Rows' field, an 'Export' button, and a 'Wrap Cell Content' option. A 'Result Grid' button and a 'Form Editor' button are also visible on the right side of the interface.

subject	average_marks
Math	79.0000
Science	80.0000
English	75.0000

12. List names of students who scored more than the average marks.

```
SELECT s.name, m.marks  
FROM Students s  
JOIN Marks m ON s.student_id = m.student_id  
WHERE m.marks > (
```

```
SELECT AVG(marks) FROM Marks
);
```

	name	marks
▶	Ahmed	88
	Ali	90
	Usman	95
	Zara	85

17. Get names of students who have not received marks in "English".

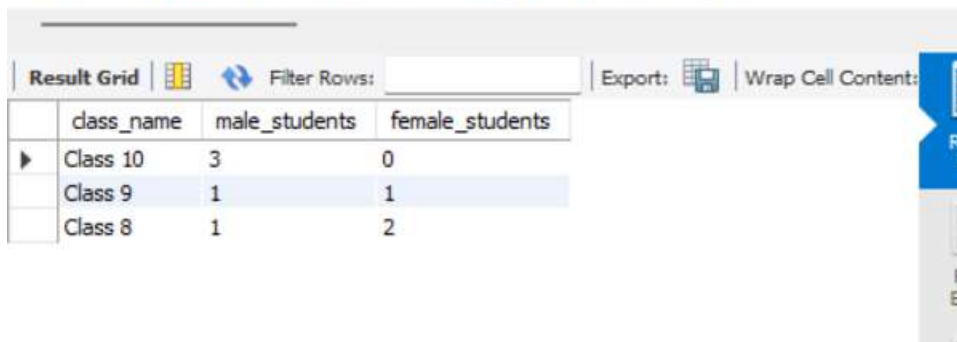
```
SELECT name
FROM Students
WHERE student_id NOT IN (
    SELECT student_id FROM Marks WHERE subject = 'English'
);
```

	name
▶	Ahmed
	Sara
	Ali
	Usman
	Hassan
	Ali

18. Display each class name with the total number of male and female students.

```
SELECT c.class_name,
    SUM(CASE WHEN s.gender = 'Male' THEN 1 ELSE 0 END) AS male_students,
    SUM(CASE WHEN s.gender = 'Female' THEN 1 ELSE 0 END) AS female_students
FROM Classes c
JOIN Students s ON c.class_id = s.class_id
GROUP BY c.class_name;
```

118 JOIN Marks m ON s.student_id = m.student_id



The screenshot shows a database query result grid. The grid has four columns: 'class_name', 'male_students', and 'female_students'. There are three rows of data: 'Class 10' with 3 male and 0 female students, 'Class 9' with 1 male and 1 female student, and 'Class 8' with 1 male and 2 female students. The interface includes a 'Result Grid' tab, a 'Filter Rows' search bar, and buttons for 'Export' and 'Wrap Cell Content'.

	class_name	male_students	female_students
▶	Class 10	3	0
	Class 9	1	1
	Class 8	1	2

19. Get a list of students with total marks across all subjects, ordered from highest to lowest.

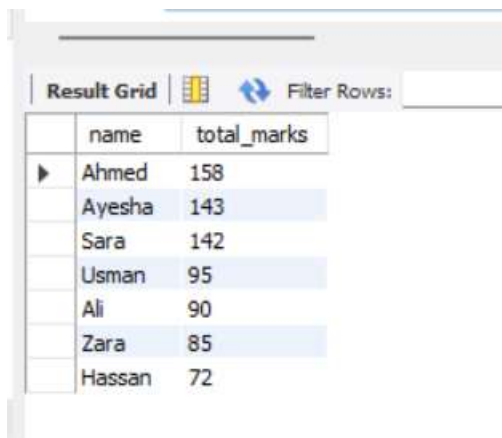
```
SELECT s.name, SUM(m.marks) AS total_marks
```

```
FROM Students s
```

```
JOIN Marks m ON s.student_id = m.student_id
```

```
GROUP BY s.name
```

```
ORDER BY total_marks DESC;
```



The screenshot shows a database query result grid. The grid has two columns: 'name' and 'total_marks'. There are eight rows of data, ordered from highest to lowest total marks: Ahmed (158), Ayesha (143), Sara (142), Usman (95), Ali (90), Zara (85), and Hassan (72). The interface includes a 'Result Grid' tab, a 'Filter Rows' search bar, and a 'Wrap Cell Content' button.

	name	total_marks
▶	Ahmed	158
	Ayesha	143
	Sara	142
	Usman	95
	Ali	90
	Zara	85
	Hassan	72