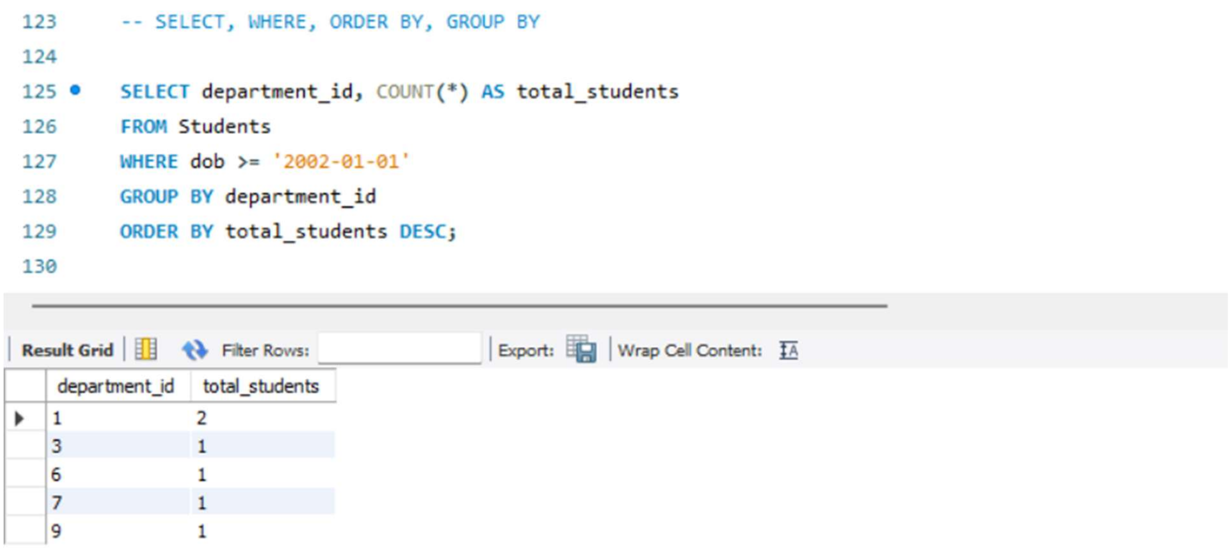
College Dataset [SQL]

Presented By: Archi Dabhi

QUERIES OUTPUT:

➢ Use SELECT, WHERE, ORDER BY, GROUP BY

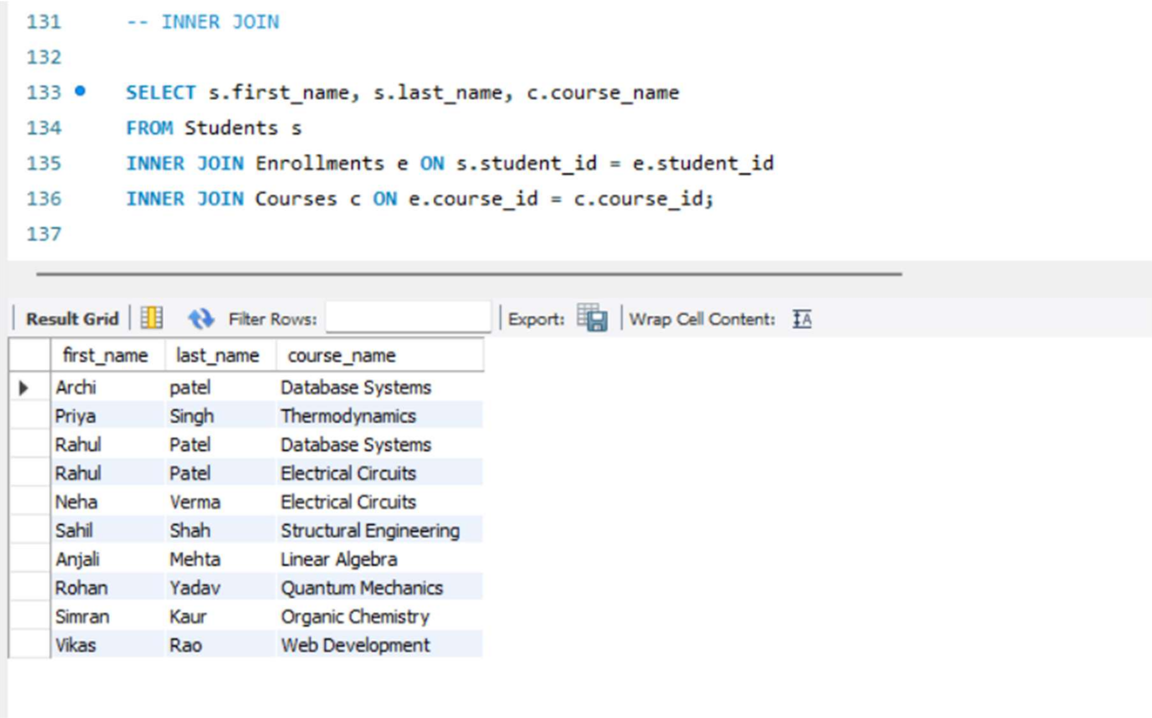


```
-- SELECT, WHERE, ORDER BY, GROUP BY

SELECT department_id, COUNT(*) AS total_students
FROM Students
WHERE dob >= '2002-01-01'
GROUP BY department_id
ORDER BY total_students DESC;

```

department_id	total_students
1	2
3	1
6	1
7	1
9	1

➢ Use JOINS (INNER, LEFT, RIGHT)

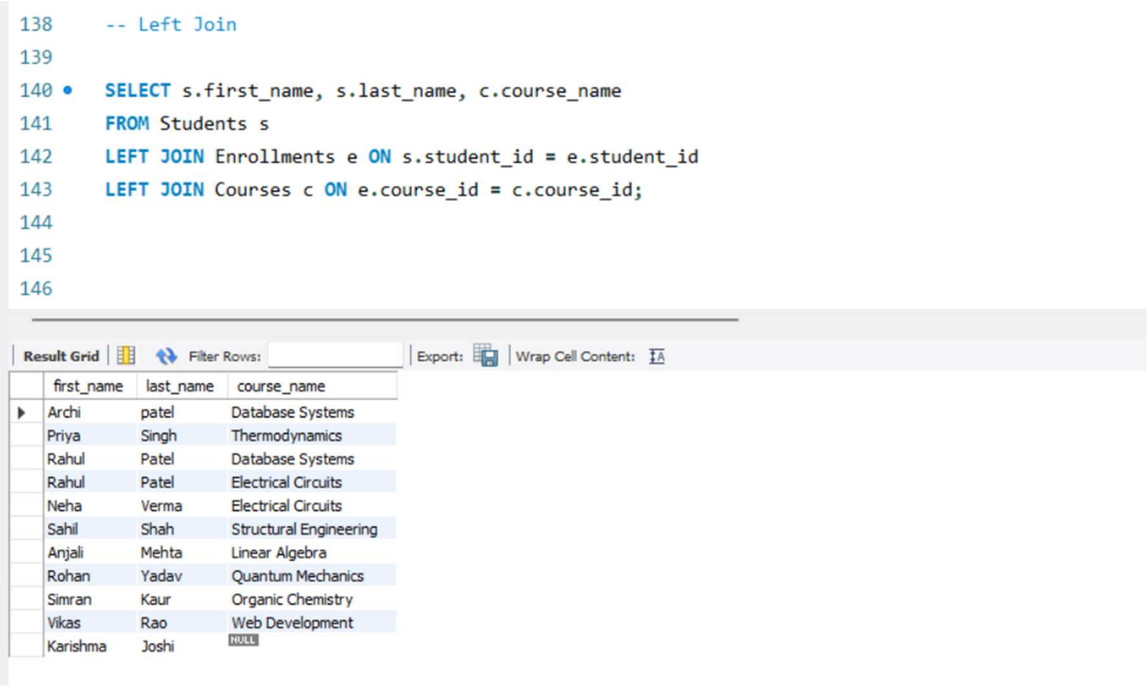
# INNER



```sql
-- INNER JOIN

SELECT s.first_name, s.last_name, c.course_name
FROM Students s
INNER JOIN Enrollments e ON s.student_id = e.student_id
INNER JOIN Courses c ON e.course_id = c.course_id;
```

first_name	last_name	course_name
Archi	patel	Database Systems
Priya	Singh	Thermodynamics
Rahul	Patel	Database Systems
Rahul	Patel	Electrical Circuits
Neha	Verma	Electrical Circuits
Sahil	Shah	Structural Engineering
Anjali	Mehta	Linear Algebra
Rohan	Yadav	Quantum Mechanics
Simran	Kaur	Organic Chemistry
Vikas	Rao	Web Development

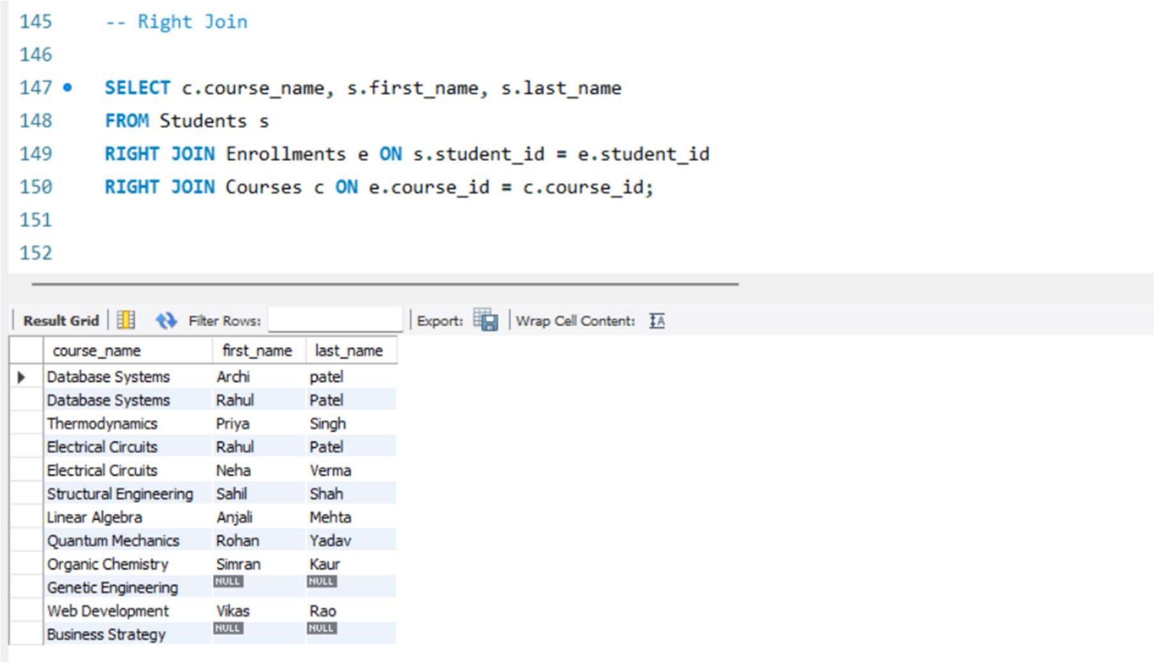
# LEFT



```sql
-- Left Join

SELECT s.first_name, s.last_name, c.course_name
FROM Students s
LEFT JOIN Enrollments e ON s.student_id = e.student_id
LEFT JOIN Courses c ON e.course_id = c.course_id;
```

first_name	last_name	course_name
Archi	patel	Database Systems
Priya	Singh	Thermodynamics
Rahul	Patel	Database Systems
Rahul	Patel	Electrical Circuits
Neha	Verma	Electrical Circuits
Sahil	Shah	Structural Engineering
Anjali	Mehta	Linear Algebra
Rohan	Yadav	Quantum Mechanics
Simran	Kaur	Organic Chemistry
Vikas	Rao	Web Development
Karishma	Joshi	NULL

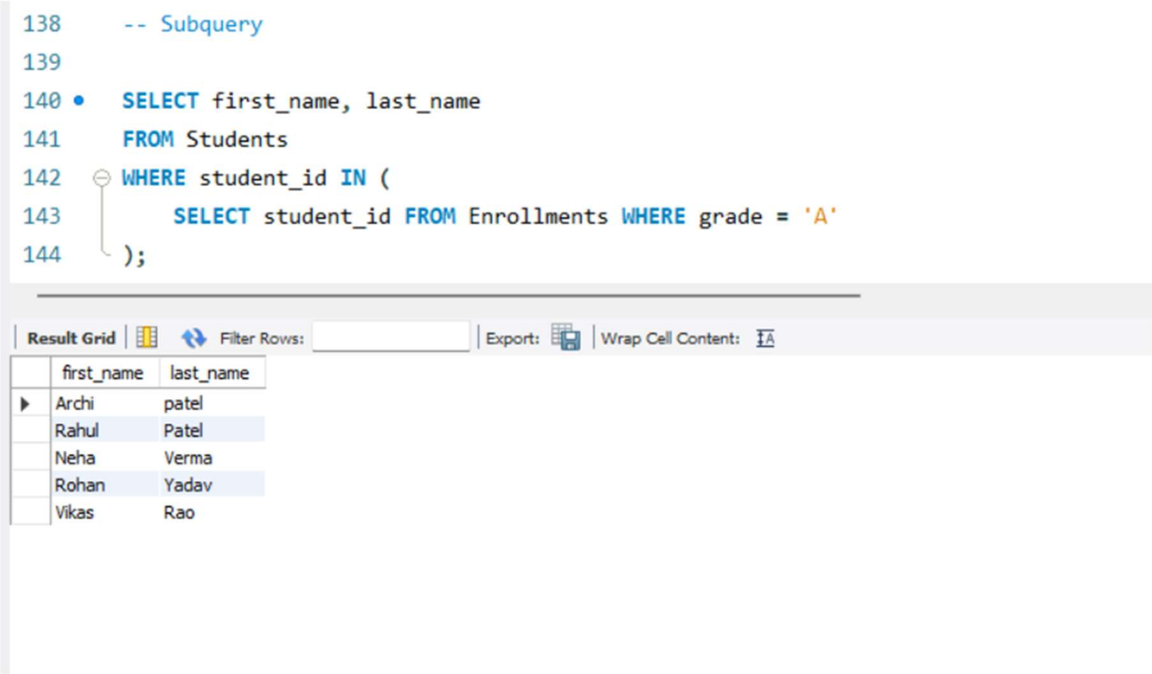
#RIGHT



```sql
-- Right Join

SELECT c.course_name, s.first_name, s.last_name
FROM Students s
RIGHT JOIN Enrollments e ON s.student_id = e.student_id
RIGHT JOIN Courses c ON e.course_id = c.course_id;
```

course_name	first_name	last_name
Database Systems	Archi	patel
Database Systems	Rahul	Patel
Thermodynamics	Priya	Singh
Electrical Circuits	Rahul	Patel
Electrical Circuits	Neha	Verma
Structural Engineering	Sahil	Shah
Linear Algebra	Anjali	Mehta
Quantum Mechanics	Rohan	Yadav
Organic Chemistry	Simran	Kaur
Genetic Engineering	NULL	NULL
Web Development	Vikas	Rao
Business Strategy	NULL	NULL

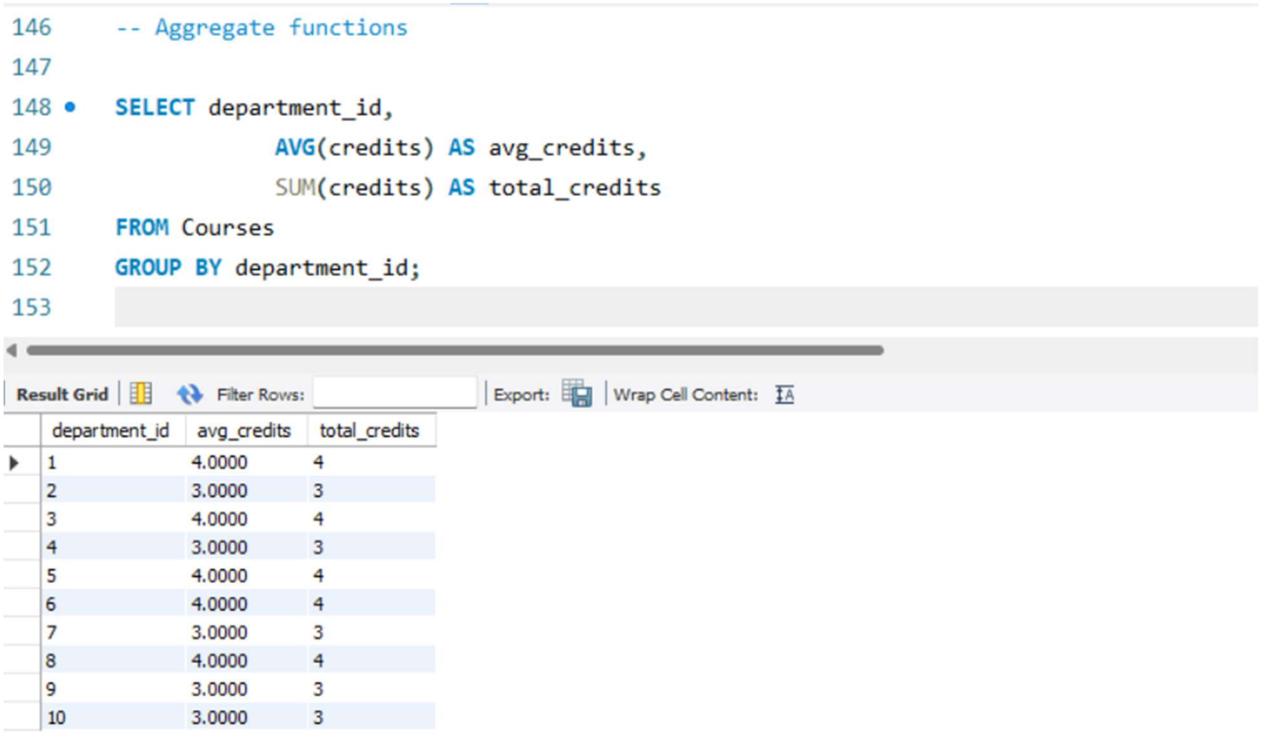
➢ Write subqueries



```sql
-- Subquery

SELECT first_name, last_name
FROM Students
WHERE student_id IN (
    SELECT student_id FROM Enrollments WHERE grade = 'A'
);
```

first_name	last_name
Archi	patel
Rahul	Patel
Neha	Verma
Rohan	Yadav
Vikas	Rao

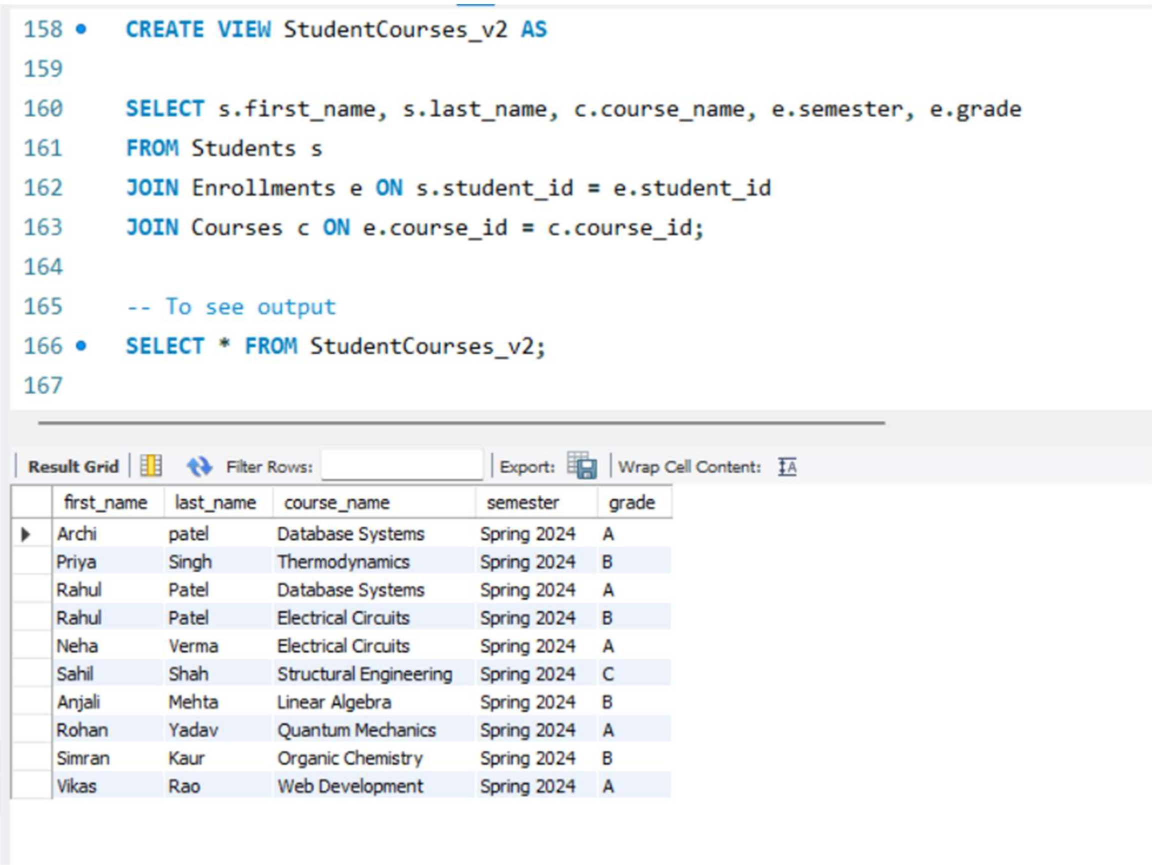
➢ Use aggregate functions (SUM, AVG)



```sql
-- Aggregate functions

SELECT department_id,
            AVG(credits) AS avg_credits,
            SUM(credits) AS total_credits
FROM Courses
GROUP BY department_id;

```

department_id	avg_credits	total_credits
1	4.0000	4
2	3.0000	3
3	4.0000	4
4	3.0000	3
5	4.0000	4
6	4.0000	4
7	3.0000	3
8	4.0000	4
9	3.0000	3
10	3.0000	3

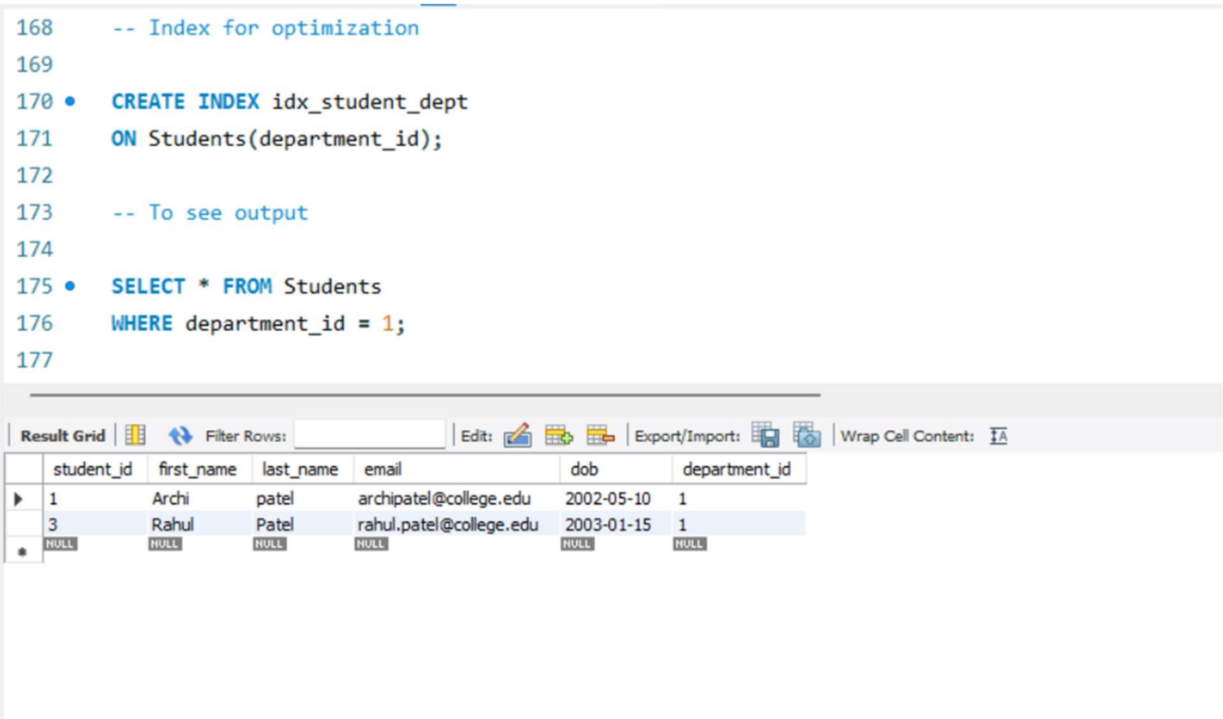
➢ Create views for analysis



```
CREATE VIEW StudentCourses_v2 AS

SELECT s.first_name, s.last_name, c.course_name, e.semester, e.grade
FROM Students s
JOIN Enrollments e ON s.student_id = e.student_id
JOIN Courses c ON e.course_id = c.course_id;

-- To see output
SELECT * FROM StudentCourses_v2;

```

first_name	last_name	course_name	semester	grade
Archi	patel	Database Systems	Spring 2024	A
Priya	Singh	Thermodynamics	Spring 2024	B
Rahul	Patel	Database Systems	Spring 2024	A
Rahul	Patel	Electrical Circuits	Spring 2024	B
Neha	Verma	Electrical Circuits	Spring 2024	A
Sahil	Shah	Structural Engineering	Spring 2024	C
Anjali	Mehta	Linear Algebra	Spring 2024	B
Rohan	Yadav	Quantum Mechanics	Spring 2024	A
Simran	Kaur	Organic Chemistry	Spring 2024	B
Vikas	Rao	Web Development	Spring 2024	A

➢ Optimize queries with indexes



```sql
-- Index for optimization

CREATE INDEX idx_student_dept
ON Students(department_id);

-- To see output

SELECT * FROM Students
WHERE department_id = 1;

```

student_id	first_name	last_name	email	dob	department_id
1	Archi	patel	archipatel@college.edu	2002-05-10	1
3	Rahul	Patel	rahul.patel@college.edu	2003-01-15	1
NULL	NULL	NULL	NULL	NULL	NULL