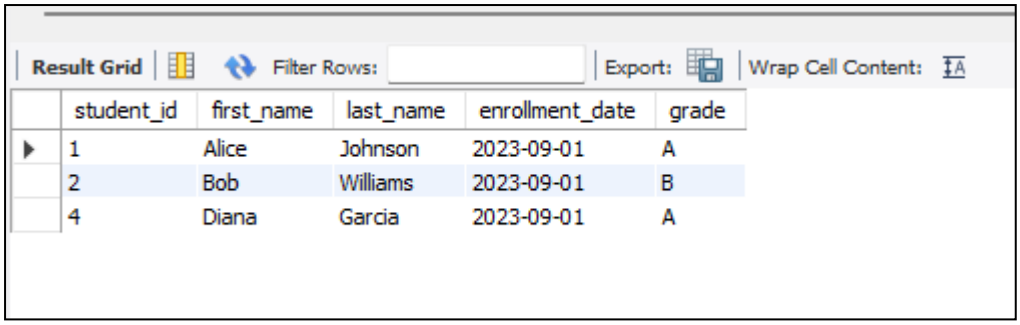
**Query 1: Retrieve All Students in a Specific Course**

**Purpose:**
Find all students enrolled in "Introduction to Programming" (CS101)

```
SELECT s.student_id, s.first_name, s.last_name, e.enrollment_date, e.grade
FROM students s
JOIN enrollments e ON s.student_id = e.student_id
WHERE e.course_id = 1;
```

**Explanation:**

1. Joins students and enrollments tables using student_id
2. Filters for course_id = 1 (CS101)
3. Returns:
    - Student identification details
    - Enrollment date
    - Current grade (if assigned)




| student_id | first_name | last_name | enrollment_date | grade |
|---|---|---|---|---|
| 1 | Alice | Johnson | 2023-09-01 | A |
| 2 | Bob | Williams | 2023-09-01 | B |
| 4 | Diana | Garcia | 2023-09-01 | A |

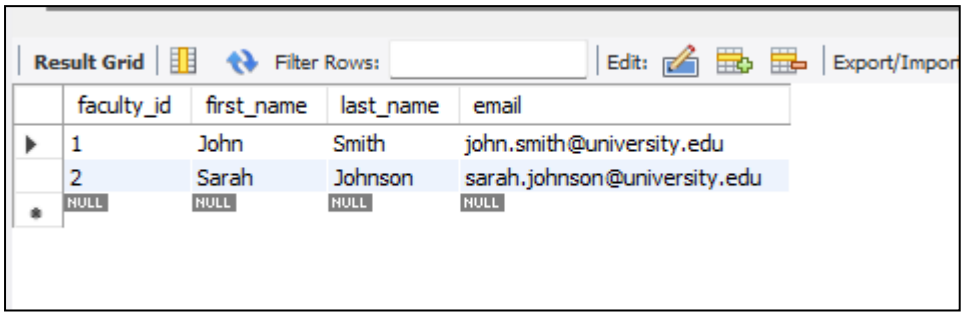
**Query 2: Find Faculty in a Department**

**Purpose:**
List all Computer Science department faculty

```
SELECT f.faculty_id, f.first_name, f.last_name, f.email
FROM faculty f
WHERE f.department_id = 1;
```

**Explanation:**

1. Scans the faculty table
2. Filters for department_id = 1
3. Returns faculty contact information




| faculty_id | first_name | last_name | email |
|---|---|---|---|
| 1 | John | Smith | john.smith@university.edu |
| 2 | Sarah | Johnson | sarah.johnson@university.edu |
| NULL | NULL | NULL | NULL |

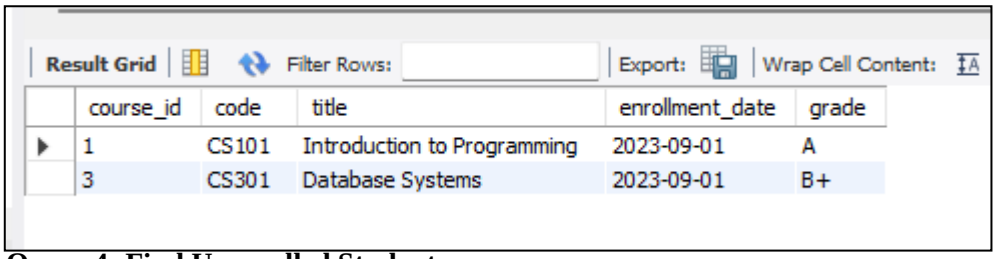
**Query 3: List Courses for a Student**

**Purpose:**
Show all courses taken by Alice Johnson (student_id=1)

```
SELECT c.course_id, c.code, c.title, e.enrollment_date, e.grade
FROM courses c
JOIN enrollments e ON c.course_id = e.course_id
WHERE e.student_id = 1;
```

**Explanation:**

1. Joins courses and enrollments tables
2. Filters for student_id = 1
3. Returns course details with grades




| course_id | code | title | enrollment_date | grade |
|---|---|---|---|---|
| 1 | CS101 | Introduction to Programming | 2023-09-01 | A |
| 3 | CS301 | Database Systems | 2023-09-01 | B+ |

**Query 4: Find Unenrolled Students**

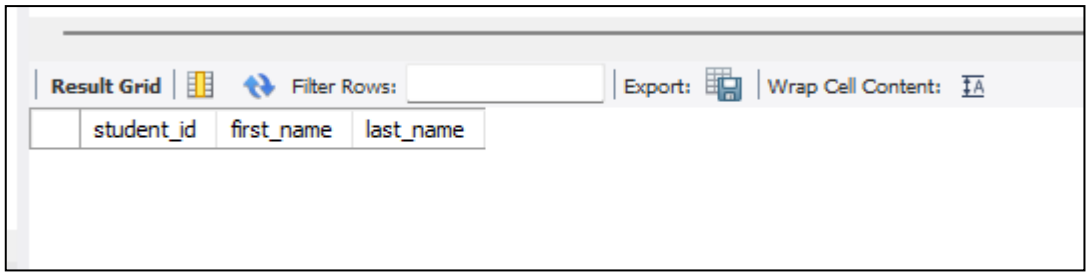
**Purpose:**
Identify students not registered for any courses

```
SELECT s.student_id, s.first_name, s.last_name
FROM students s
LEFT JOIN enrollments e ON s.student_id = e.student_id
WHERE e.enrollment_id IS NULL;
```

**Explanation:**

1. Left join preserves all students
2. IS NULL finds records with no enrollment

3. Returns basic student info




**Query 5: Calculate Course Average Grade**

**Purpose:**
**Compute grade average for CS101**

```sql
SELECT c.code, c.title,
       AVG(CASE
           WHEN e.grade = 'A' THEN 4.0
           WHEN e.grade = 'A-' THEN 3.7
           WHEN e.grade = 'B+' THEN 3.3
           WHEN e.grade = 'B' THEN 3.0
           WHEN e.grade = 'B-' THEN 2.7
           WHEN e.grade = 'C+' THEN 2.3
           WHEN e.grade = 'C' THEN 2.0
           WHEN e.grade = 'C-' THEN 1.7
           WHEN e.grade = 'D+' THEN 1.3
           WHEN e.grade = 'D' THEN 1.0
           ELSE 0.0
       END) as average_grade_points
FROM courses c
JOIN enrollments e ON c.course_id = e.course_id
WHERE c.course_id = 1
GROUP BY c.course_id;
```

**Explanation:**

1. Converts letter grades to numeric values
2. Calculates weighted average
3. Returns single-row summary