**Assignment 3**

**Overview:** This is a continuation to the previous assignment, In this part of the assignment, you will practice relational algebra operations on your University Database System (UDS) schema. You will perform various operations such as select, project, rename, union, intersection, set difference, cartesian product, and join to retrieve meaningful information from your database. Additionally, you will combine multiple relational algebra operations to answer more complex queries.

1. Relational Algebra Operations:

Perform the following relational algebra operations on your database schema. For each query, provide the appropriate relational algebra syntax and describe how the operation works. After that, write the SQL equivalent and show the output to demonstrate the correctness of your relational algebra queries.

* **Select (σ) Operation:** Retrieve all students enrolled in the "Database Systems" or any other course.
  + Relational Algebra Syntax(Example):



NOTE: The **Select** operation (σ) is used to filter rows based on a condition. In this case, we are filtering for students who are enrolled in the "Database Systems" course.

* **Project (π) Operation:** Retrieve the names and email addresses of all faculty members.

**Explanation:** The Project operation (π) is used to select specific columns from a table. Here, we are selecting the Name and Email columns from the Faculty table.

* **Rename (ρ) Operation:** Rename the Faculty table to Professors, next reverse the change.

**Explanation:** The Rename operation (ρ) is used to rename a relation (table). In this case, we are renaming the Faculty table to Professors.

* **Union (∪) Operation:** Retrieve a list of all students and faculty who have an email address.

**Explanation:** The Union operation (∪) combines the results of two relations, keeping only distinct rows. Here, we are combining the Name and Email columns of both Student and Faculty tables to get a list of all students and faculty with email addresses.

* **Intersection (∩) Operation:** Retrieve the list of faculty members who also teach the "Database Systems" course.

**Explanation**: The Intersection operation (∩) finds common rows between two relations. Here, we are finding faculty members who teach the "Database Systems" course by joining the Faculty and Course tables, filtering for the course "Database Systems," and then performing the intersection with the list of all faculty members.

* **Set Difference (−) Operation:** Retrieve the list of students not enrolled in the "Database Systems" course.

**Explanation:** The Set Difference operation (−) finds rows that are present in one relation but not in another. In this case, we are finding students who are not enrolled in the "Database Systems" course by subtracting the set of students enrolled in this course from the set of all students.

* **Cartesian Product (×) Operation:** Retrieve the Cartesian product of Faculty and Course.

**Explanation**: The Cartesian Product operation (×) returns all combinations of rows from the two relations. Here, we are performing the Cartesian product of the Faculty and Course tables, meaning that each faculty member is paired with every course in the database.

* **Join (⨝) Operation:** Retrieve all students enrolled in courses, along with course details.

**Explanation:** The Join operation (⨝) is used to combine related tuples from two or more relations based on a common attribute. In this case, we are joining the Student and Enrollment tables on StudentID, and then joining the Enrollment and Course tables on CourseID, to retrieve all students enrolled in courses along with the course details.

1. Complex Queries:

For each query, provide the appropriate relational algebra syntax and describe how the operation works. After that, write the SQL equivalent and show the output to demonstrate the correctness of your relational algebra queries.

* Retrieve the names of students enrolled in courses taught by "Dr. Smith" or any other faculty listed in the table.

* Retrieve the names of faculty members who teach courses in the "Computer Science" department for example.

1. Deliverables (How and what to submit):
   * **Submit your relational algebra queries** in the specified syntax for each operation (Select, Project, Rename, Union, etc.). Include the following:
     + Relational Algebra Queries for each of the tasks in Part 1 and Part 2.
     + SQL equivalents for the queries (to demonstrate how they would be implemented in a relational database).
     + Screenshots of the SQL queries and their outputs for each operation.

* **Explanation of Queries:** For each query, provide a brief explanation of:
  + The relational algebra operation being performed (e.g., Select, Project, Join, etc.).
  + How the operation works in the context of the database schema.
  + The relationship between the operations and the data being queried.

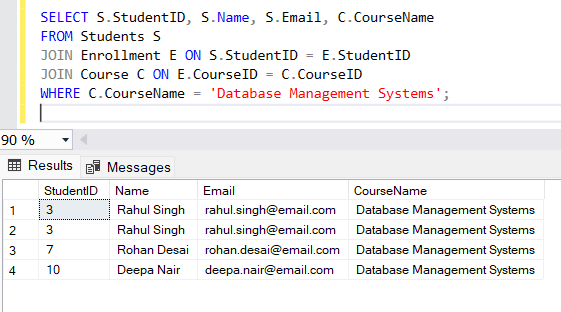
**Answers:**

**1. Select (σ) Operation:**

Retrieve all students enrolled in the "Database Management Systems" course.

**Relational Algebra Expression**:



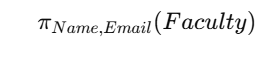
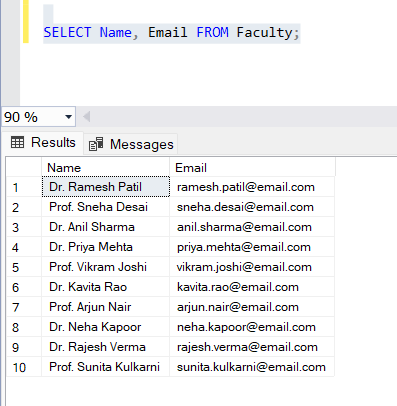


--------------------------------------------------------------------------------------------------------------------------------------

**2. Project (π) Operation**

Retrieve the names and email addresses of all faculty members.

**Relational Algebra Expression:**

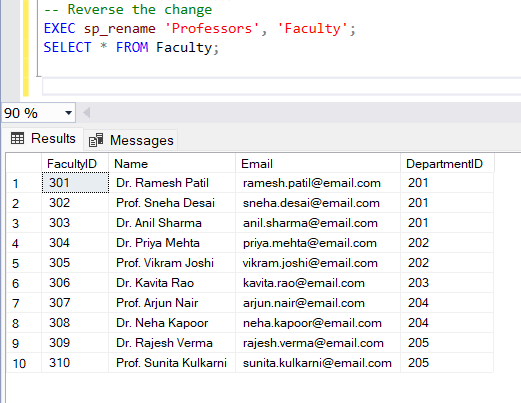
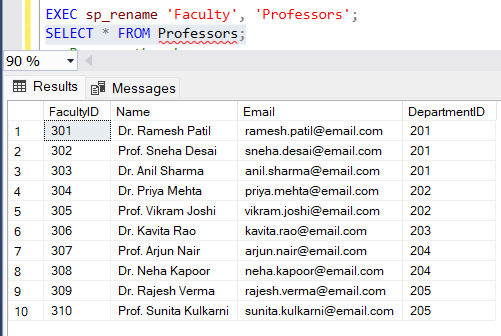
--------------------------------------------------------------------------------------------------------------------------------------

**3. Rename (ρ) Operation**

Rename the Faculty table to Professors, next reverse the change.

**Relational Algebra Expression:**





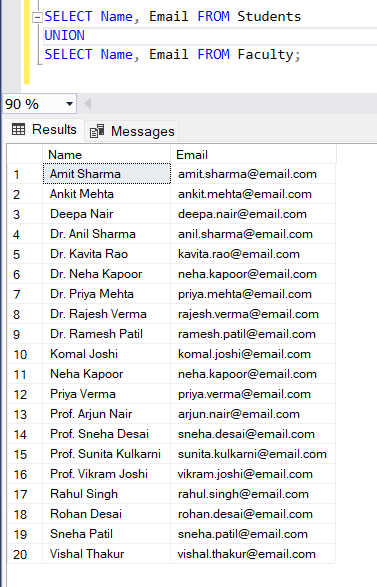
--------------------------------------------------------------------------------------------------------------------------------------

**4. Union (∪) Operation**

Retrieve a list of all students and faculty who have an email address.

**Relational Algebra Expression:**





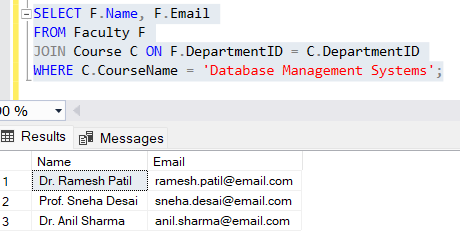
--------------------------------------------------------------------------------------------------------------------------------------

**5. Intersection (∩) Operation**

Retrieve the list of faculty members who also teach the "Database Management Systems" course.

**Relational Algebra Expression:**





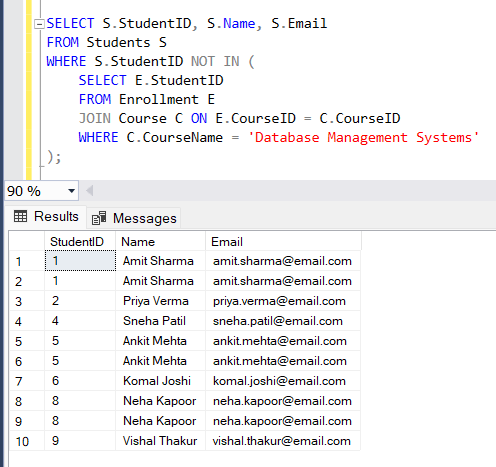
--------------------------------------------------------------------------------------------------------------------------------------

**6. Set Difference (−) Operation**

Retrieve the list of students not enrolled in the "Database Systems" course.

**Relational Algebra Expression:**





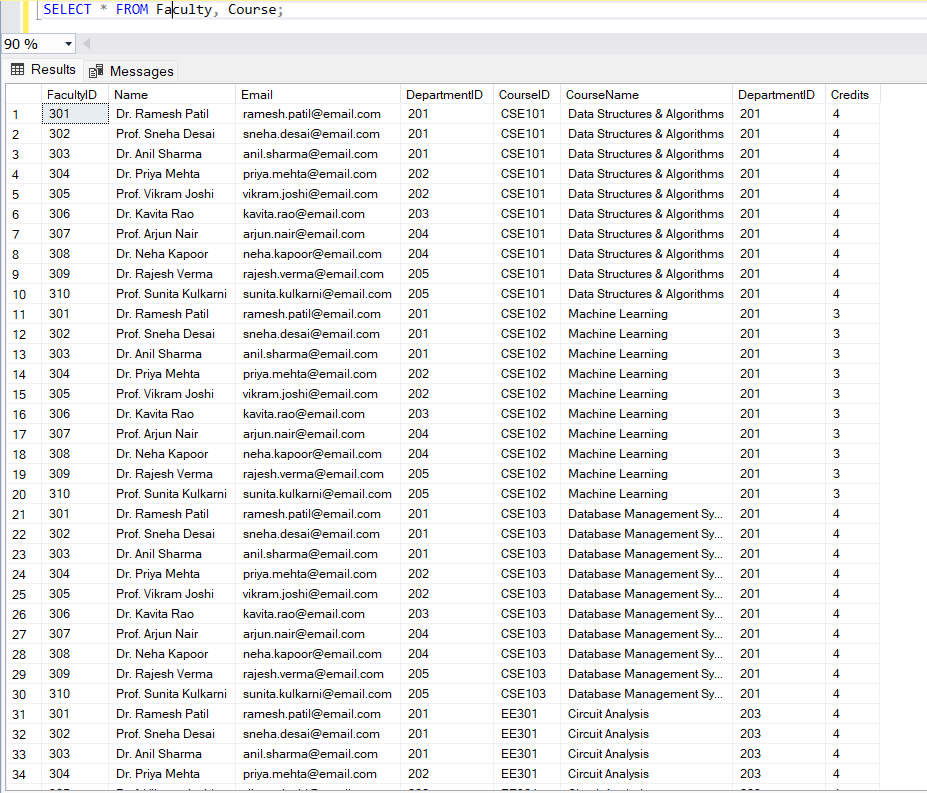
--------------------------------------------------------------------------------------------------------------------------------------

**7. Cartesian Product (×) Operation**

Retrieve the Cartesian product of Faculty and Course.

**Relational Algebra Expression:**



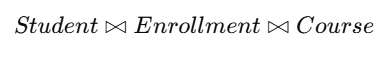
There are total 100 rows.

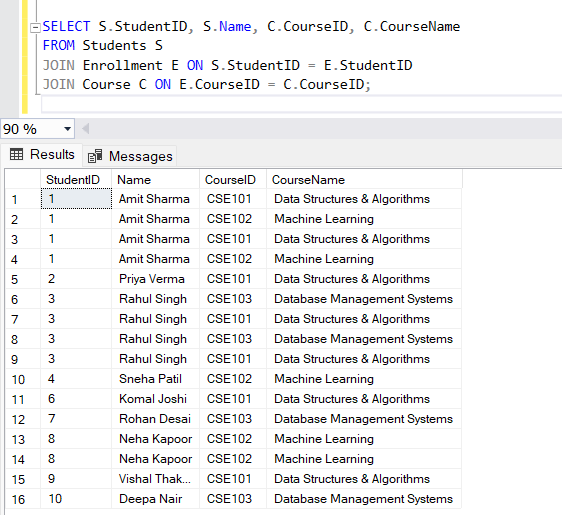
--------------------------------------------------------------------------------------------------------------------------------------

**8. Join (⨝) Operation**

Retrieve all students enrolled in courses, along with course details.

**Relational Algebra Expression:**





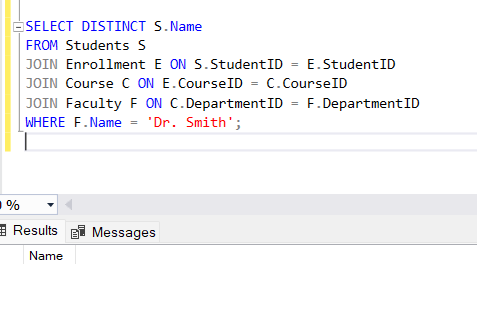
--------------------------------------------------------------------------------------------------------------------------------------

**Complex Queries**

1. Students enrolled in courses taught by "Dr. Smith"

**Relational Algebra Expression:**



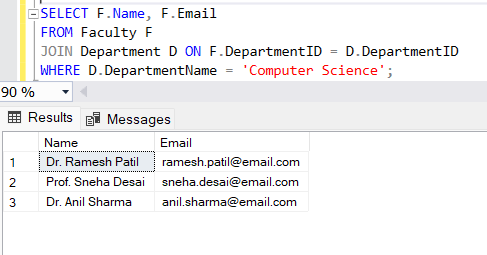


--------------------------------------------------------------------------------------------------------------------------------------

**2. Faculty teaching in "Computer Science"**

**Relational Algebra Expression:**





--------------------------------------------------------------------------------------------------------------------------------------

--------------------------------------------------------------------------------------------------------------------------------------