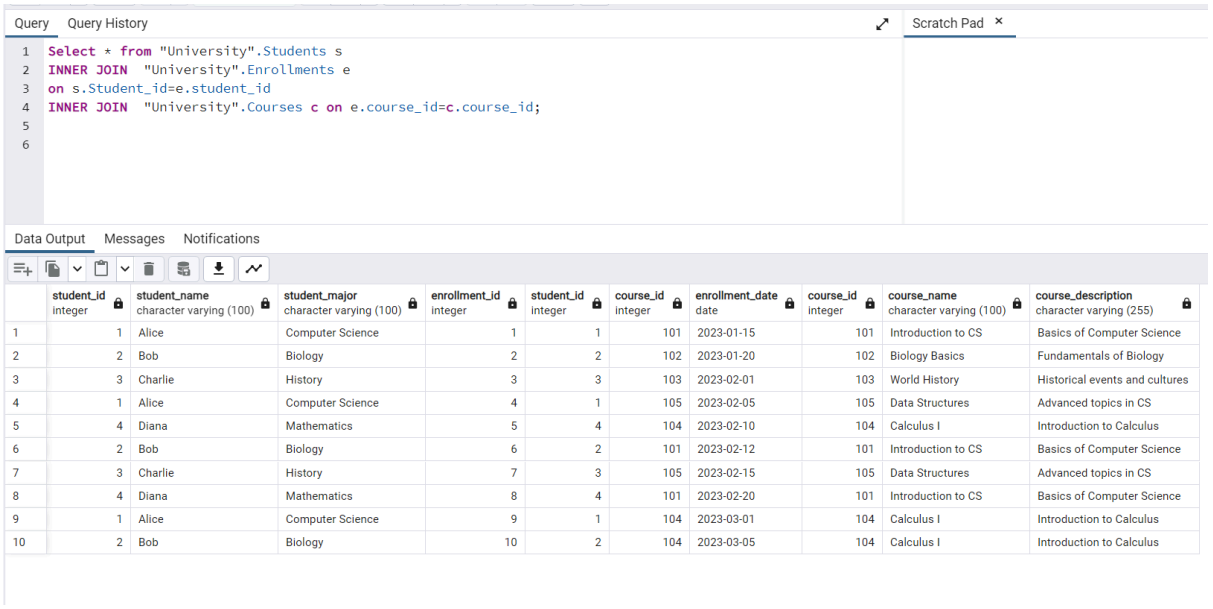
1.Inner-Join :



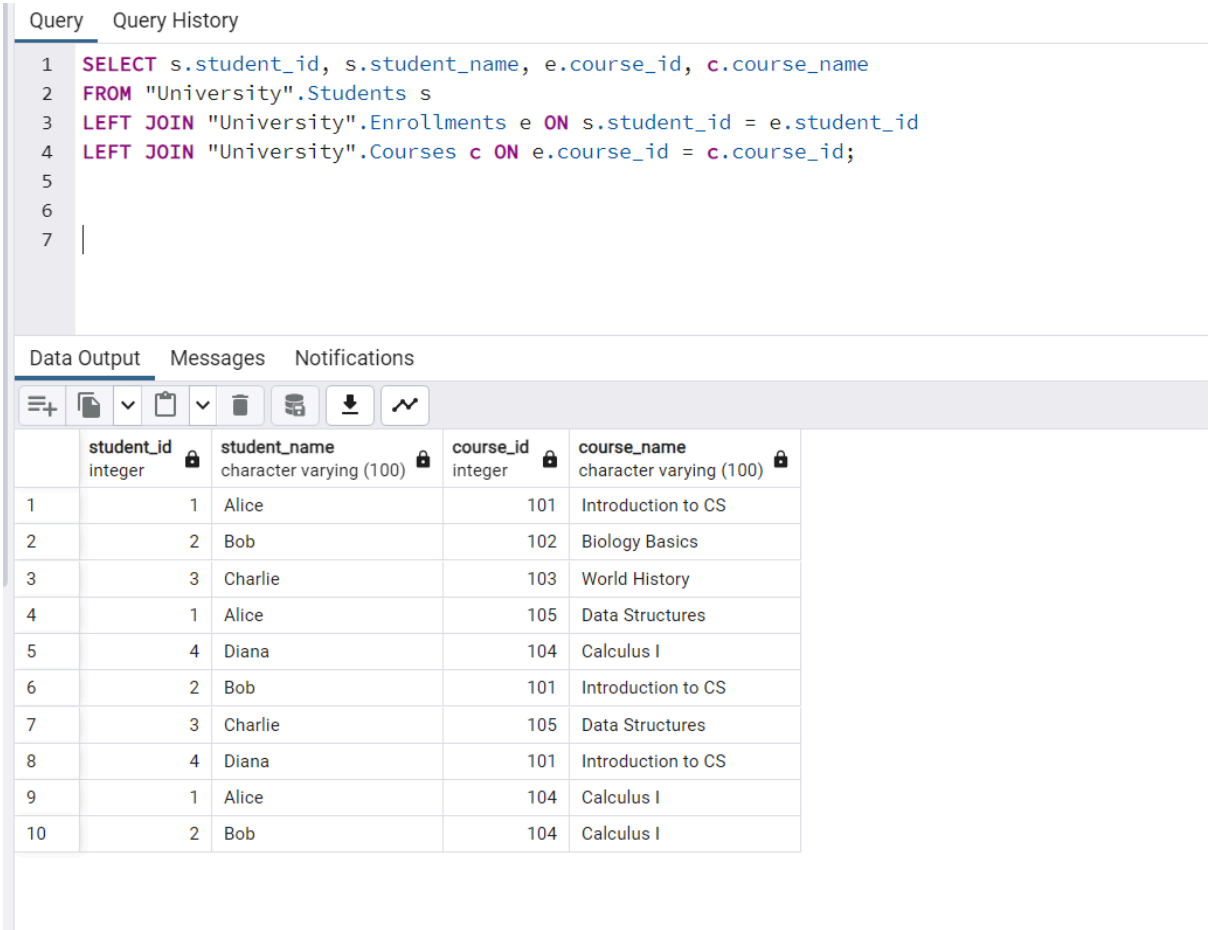
Query | Query History | Scratch Pad

```
Select * from "University".Students s
INNER JOIN  "University".Enrollments e
on s.Student_id=e.student_id
INNER JOIN  "University".Courses c on e.course_id=c.course_id;
```

Data Output | Messages | Notifications

	student_id integer	student_name character varying (100)	student_major character varying (100)	enrollment_id integer	student_id integer	course_id integer	enrollment_date date	course_id integer	course_name character varying (100)	course_description character varying (255)
1	1	Alice	Computer Science	1	1	101	2023-01-15	101	Introduction to CS	Basics of Computer Science
2	2	Bob	Biology	2	2	102	2023-01-20	102	Biology Basics	Fundamentals of Biology
3	3	Charlie	History	3	3	103	2023-02-01	103	World History	Historical events and cultures
4	1	Alice	Computer Science	4	1	105	2023-02-05	105	Data Structures	Advanced topics in CS
5	4	Diana	Mathematics	5	4	104	2023-02-10	104	Calculus I	Introduction to Calculus
6	2	Bob	Biology	6	2	101	2023-02-12	101	Introduction to CS	Basics of Computer Science
7	3	Charlie	History	7	3	105	2023-02-15	105	Data Structures	Advanced topics in CS
8	4	Diana	Mathematics	8	4	101	2023-02-20	101	Introduction to CS	Basics of Computer Science
9	1	Alice	Computer Science	9	1	104	2023-03-01	104	Calculus I	Introduction to Calculus
10	2	Bob	Biology	10	2	104	2023-03-05	104	Calculus I	Introduction to Calculus

2:Left-Join :



Query | Query History

```
SELECT s.student_id, s.student_name, e.course_id, c.course_name
FROM "University".Students s
LEFT JOIN "University".Enrollments e ON s.student_id = e.student_id
LEFT JOIN "University".Courses c ON e.course_id = c.course_id;
```

Data Output | Messages | Notifications

	student_id integer	student_name character varying (100)	course_id integer	course_name character varying (100)
1	1	Alice	101	Introduction to CS
2	2	Bob	102	Biology Basics
3	3	Charlie	103	World History
4	1	Alice	105	Data Structures
5	4	Diana	104	Calculus I
6	2	Bob	101	Introduction to CS
7	3	Charlie	105	Data Structures
8	4	Diana	101	Introduction to CS
9	1	Alice	104	Calculus I
10	2	Bob	104	Calculus I

3:Right-Join:

```sql
SELECT s.student_id, s.student_name, e.course_id, c.course_name
FROM "University".Students s
RIGHT JOIN "University".Enrollments e ON s.student_id = e.student_id
RIGHT JOIN "University".Courses c ON e.course_id = c.course_id;
```

Data Output | Messages | Notifications

	student_id integer	student_name character varying (100)	course_id integer	course_name character varying (100)
1	1	Alice	101	Introduction to CS
2	2	Bob	102	Biology Basics
3	3	Charlie	103	World History
4	1	Alice	105	Data Structures
5	4	Diana	104	Calculus I
6	2	Bob	101	Introduction to CS
7	3	Charlie	105	Data Structures
8	4	Diana	101	Introduction to CS
9	1	Alice	104	Calculus I
10	2	Bob	104	Calculus I

Total rows: 10 of 10 Query complete 00:00:00.061

4.Self-Join:

Query | Query History

```
1  SELECT s1.student_name AS student1, s2.student_name AS student2, c.course_name
2  FROM "University".Enrollments e1
3  INNER JOIN "University".Enrollments e2 ON e1.course_id = e2.course_id AND e1.student_id <> e2.student_id
4  INNER JOIN "University".Students s1 ON e1.student_id = s1.student_id
5  INNER JOIN "University".Students s2 ON e2.student_id = s2.student_id
6  INNER JOIN "University".Courses c ON e1.course_id = c.course_id;
7
8
9
```

Data Output | Messages | Notifications

	student1 character varying (100)	student2 character varying (100)	course_name character varying (100)
1	Alice	Bob	Introduction to CS
2	Alice	Diana	Introduction to CS
3	Bob	Alice	Introduction to CS
4	Bob	Diana	Introduction to CS
5	Diana	Alice	Introduction to CS
6	Diana	Bob	Introduction to CS
7	Alice	Diana	Calculus I
8	Alice	Bob	Calculus I
9	Diana	Alice	Calculus I
10	Diana	Bob	Calculus I
11	Bob	Alice	Calculus I
12	Bob	Diana	Calculus I
13	Charlie	Alice	Data Structures
14	Alice	Charlie	Data Structures

5.Complex-Join:

Query | Query History | Scratch

```
1  SELECT DISTINCT s.student_name
2  FROM "University".Students s
3  INNER JOIN  "University".Enrollments e1 ON s.student_id = e1.student_id
4  INNER JOIN "University".Courses c1 ON e1.course_id = c1.course_id
5  LEFT JOIN "University".Enrollments e2 ON s.student_id = e2.student_id
6  LEFT JOIN "University".Courses c2 ON e2.course_id = c2.course_id AND c2.course_name = 'Data Structures'
7  WHERE c1.course_name = 'Introduction to CS'
8  AND c2.course_id IS NULL;
9
```

Data Output | Messages | Notifications

	student_name character varying (100)
1	Alice
2	Bob
3	Diana

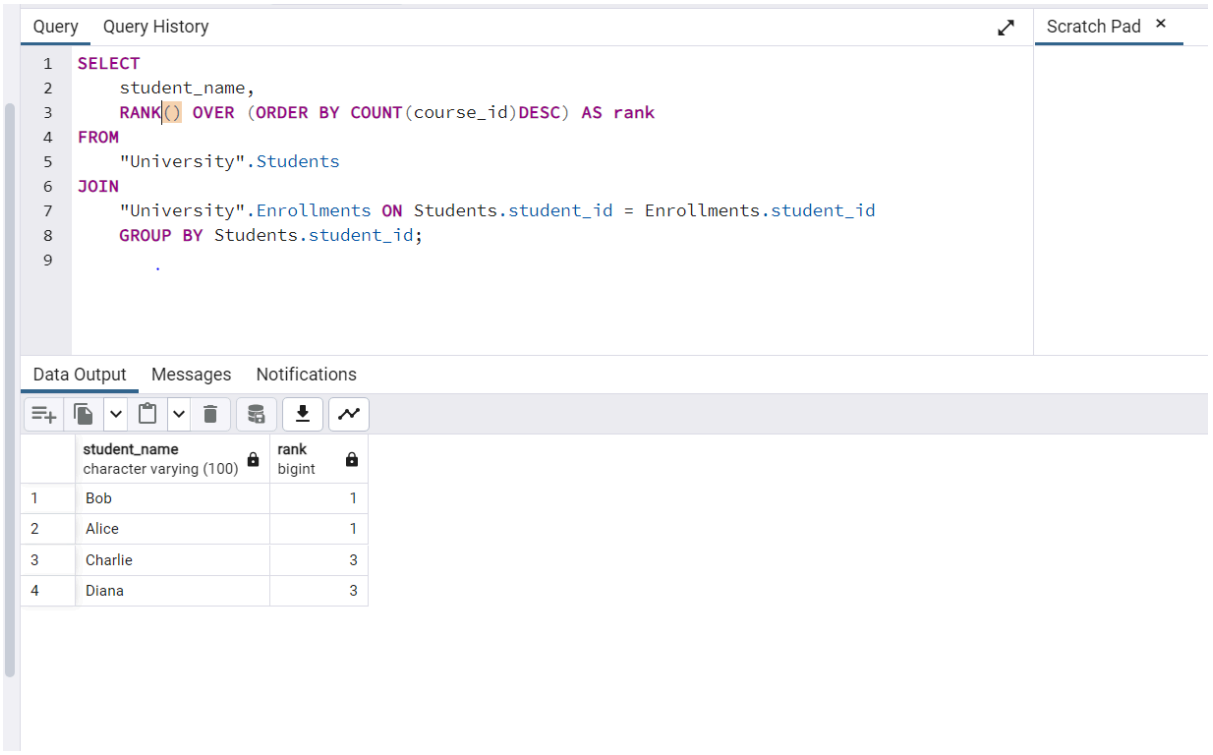
Row_Number:

```sql
SELECT
    student_name,
    enrollment_date,
    ROW_NUMBER() OVER (ORDER BY enrollment_date) AS row_num
FROM
    "University".Students
JOIN
    "University".Enrollments ON Students.student_id = Enrollments.student_id;
```

Output Messages Notifications

student_name character varying (100)	enrollment_date date	row_num bigint
Alice	2023-01-15	1
Bob	2023-01-20	2
Charlie	2023-02-01	3
Alice	2023-02-05	4
Diana	2023-02-10	5
Bob	2023-02-12	6
Charlie	2023-02-15	7
Diana	2023-02-20	8
Alice	2023-03-01	9
Bob	2023-03-05	10

Rank:


Query
Query History
Scratch Pad
1 SELECT
2 student_name,
3 RANK() OVER (ORDER BY COUNT(course_id)DESC) AS rank
4 FROM
5 "University".Students
6 JOIN
7 "University".Enrollments ON Students.student_id = Enrollments.student_id
8 GROUP BY Students.student_id;
9
Data Output
Messages
Notifications
student_name character varying (100)
rank bigint
1 Bob 1
2 Alice 1
3 Charlie 3
4 Diana 3

Dense_Rank:

```sql
SELECT
    c.course_id,
    c.course_name,
    COUNT(e.enrollment_id) AS enrollment_count,
    DENSE_RANK() OVER (ORDER BY COUNT(e.enrollment_id) DESC) AS dense_rank
FROM
    "University".Courses c
LEFT JOIN "University".Enrollments e ON c.course_id = e.course_id
GROUP BY
    c.course_id, c.course_name
ORDER BY
    dense_rank;
```

Data Output | Messages | Notifications

	course_id [PK] integer	course_name character varying (100)	enrollment_count bigint	dense_rank bigint
1	101	Introduction to CS	3	1
2	104	Calculus I	3	1
3	105	Data Structures	2	2
4	102	Biology Basics	1	3
5	103	World History	1	3