



**DBMS :- IT-214**  
**Project :-Merit Mapper**  
**(Academic Year - 2025)**

**Group Members:-**

Namra Kachhia - 202301129  
Meet Jain - 202301073  
Namra Sanandiya - 202301130  
Shubh Raj - 202301125  
Rishank Dhudhat - 202301068

# QUERIES IN SQL

1. The top 7 rankers have secured admissions across various colleges and branches; the details of their allotted colleges and respective branches are as follows.

Query

Query History

```
1 SELECT
2     s.StudentID,
3     s.FirstName,
4     s.LastName,
5     s.AIR,
6     cf.CollegeName,
7     b.BranchName
8 FROM
9     Student s
10    JOIN ChoicesFilled cf ON s.StudentID = cf.StudentID
11    JOIN Branch b ON cf.BranchID = b.BranchID
12    JOIN Result r ON cf.ChoiceID = r.ChoiceID
13 WHERE
14     s.AIR IN (
15         SELECT AIR
16         FROM Student
17         ORDER BY AIR ASC
18         LIMIT 7
19     )
20    AND r.Status = 'Confirmed'
21 ORDER BY
22     s.AIR ASC;
```

Data Output

Messages

Notifications

SQL

	studentid integer	firstname character varying (50)	lastname character varying (50)	air integer	collegename character varying (100)	branchname character varying (100)
1	2025001	Aarav	Sharma	155	IIT Delhi	Computer Science
2	2025002	Saanvi	Patel	160	IIT Bombay	Computer Science
3	2025003	Vivaan	Gupta	165	IIT Madras	Computer Science
4	2025004	Anika	Reddy	170	IIT Delhi	Computer Science
5	2025005	Ishaan	Mehta	175	IIT Bombay	Computer Science
6	2025006	Diya	Singh	180	IIT Delhi	Computer Science
7	2025007	Arjun	Nair	185	IIT Roorkee	Computer Science

```
SELECT
s.StudentID,
s.FirstName,
s.LastName,
s.AIR,
cf.CollegeName,
b.BranchName
FROM
Student s
JOIN ChoicesFilled cf ON s.StudentID = cf.StudentID
JOIN Branch b ON cf.BranchID = b.BranchID
JOIN Result r ON cf.ChoiceID = r.ChoiceID
WHERE
s.AIR IN (
SELECT AIR
```

```

        FROM Student
        ORDER BY AIR ASC
        LIMIT 7
    )
    AND r.Status = 'Confirmed'
ORDER BY
    s.AIR ASC;

```

2. The cutoff ranks (categorywise) for each of top 5 IITs is as follows:

```

WITH AdmittedStudents AS (
    SELECT
        cf.CollegeName,
        s.Category,
        MAX(s.AIR) AS CutoffRank
    FROM
        Student s
        JOIN ChoicesFilled cf ON s.StudentID = cf.StudentID
        JOIN Result r ON cf.ChoiceID = r.ChoiceID
    WHERE
        cf.CollegeName IN (
            'IIT Madras',
            'IIT Delhi',
            'IIT Bombay',
            'IIT Kanpur',
            'IIT Kharagpur'
        )
        AND r.Status = 'Confirmed'
        AND r.RoundID = (
            SELECT MAX(RoundID)
            FROM Result r2
            WHERE r2.ChoiceID = r.ChoiceID
        )
    GROUP BY
        cf.CollegeName,
        s.Category
)
SELECT
    CollegeName,
    Category,
    CutoffRank
FROM
    AdmittedStudents
ORDER BY
    CollegeName,

```

Category;

```
1  WITH AdmittedStudents AS (  
2      SELECT  
3          cf.CollegeName,  
4          s.Category,  
5          MAX(s.AIR) AS CutoffRank  
6      FROM  
7          Student s  
8      JOIN ChoicesFilled cf ON s.StudentID = cf.StudentID  
9      JOIN Result r ON cf.ChoiceID = r.ChoiceID  
10     WHERE  
11         cf.CollegeName IN (  
12             'IIT Madras',  
13             'IIT Delhi',  
14             'IIT Bombay',  
15             'IIT Kanpur',  
16             'IIT Kharagpur'  
17         )  
18         AND r.Status = 'Confirmed'  
19         AND r.RoundID = (  
20             SELECT MAX(RoundID)
```

Data Output Messages Notifications

	collegename character varying (100)	category character varying (50)	cutoffrank integer
1	IIT Bombay	General	175
2	IIT Delhi	General	170
3	IIT Delhi	SC	180
4	IIT Kanpur	General	2200
5	IIT Kharagpur	General	2350
6	IIT Kharagpur	SC	2050
7	IIT Kharagpur	ST	2300
8	IIT Madras	General	165
9	IIT Madras	ST	1900

3) Cutoff ranks for CS(categorywise) in all IITs are as follows:

```
WITH AdmittedStudents AS (  
  SELECT  
    cf.CollegeName,  
    s.Category,  
    MAX(s.AIR) AS CutoffRank  
  FROM  
    Student s  
    JOIN ChoicesFilled cf ON s.StudentID = cf.StudentID  
    JOIN Branch b ON cf.BranchID = b.BranchID  
    JOIN Result r ON cf.ChoiceID = r.ChoiceID  
  WHERE  
    b.BranchName = 'Computer Science'  
    AND r.Status = 'Confirmed'  
    AND r.RoundID = (  
      SELECT MAX(RoundID)  
      FROM Result r2  
      WHERE r2.ChoiceID = r.ChoiceID  
    )  
  GROUP BY  
    cf.CollegeName,  
    s.Category  
)  
SELECT  
  CollegeName,  
  Category,  
  CutoffRank  
FROM  
  AdmittedStudents  
ORDER BY  
  CollegeName,  
  Category;
```

Query Query History

```
1 WITH AdmittedStudents AS (  
2     SELECT  
3         cf.CollegeName,  
4         s.Category,  
5         MAX(s.AIR) AS CutoffRank  
6     FROM  
7         Student s  
8     JOIN ChoicesFilled cf ON s.StudentID = cf.StudentID  
9     JOIN Branch b ON cf.BranchID = b.BranchID  
10    JOIN Result r ON cf.ChoiceID = r.ChoiceID  
11    WHERE  
12        b.BranchName = 'Computer Science'  
13        AND r.Status = 'Confirmed'  
14        AND r.RoundID = (  
15            SELECT MAX(RoundID)  
16            FROM Result r2  
17            WHERE r2.ChoiceID = r.ChoiceID  
18        )  
19    GROUP BY  
20        cf.CollegeName
```

Data Output Messages Notifications

	collegename character varying (100)	category character varying (50)	cutoffrank integer
1	IIT Bombay	General	175
2	IIT Delhi	General	170
3	IIT Delhi	SC	180
4	IIT Hyderabad	General	2100
5	IIT Kanpur	General	1950
6	IIT Kharagpur	SC	2050
7	IIT Madras	General	165
8	IIT Madras	ST	1900
9	IIT Roorkee	General	185

4) The number of people that got their dream college:

Output: 17 Students

```
WITH FirstPreference AS (  
    SELECT  
        StudentID,  
        CollegeName,  
        ROW_NUMBER() OVER (PARTITION BY StudentID ORDER BY ChoiceID) AS PreferenceRank  
    FROM  
        ChoicesFilled  
)  
AdmittedStudents AS (  
    SELECT  
        cf.StudentID,  
        cf.CollegeName AS AdmittedCollege  
    FROM  
        ChoicesFilled cf  
    JOIN Result r ON cf.ChoiceID = r.ChoiceID  
    WHERE  
        r.Status = 'Confirmed'  
        AND r.RoundID = (  
            SELECT MAX(RoundID)  
            FROM Result r2  
            WHERE r2.ChoiceID = r.ChoiceID  
        )  
)  
SELECT  
    COUNT(DISTINCT a.StudentID) AS StudentsWithFirstPreference  
FROM  
    AdmittedStudents a  
    JOIN FirstPreference fp ON a.StudentID = fp.StudentID  
WHERE  
    fp.PreferenceRank = 1  
    AND a.AdmittedCollege = fp.CollegeName;
```

Query Query History

```
1 WITH FirstPreference AS (  
2     SELECT  
3         StudentID,  
4         CollegeName,  
5         ROW_NUMBER() OVER (PARTITION BY StudentID ORDER BY ChoiceID) AS PreferenceRank  
6     FROM  
7         ChoicesFilled  
8 ),  
9 AdmittedStudents AS (  
10     SELECT  
11         cf.StudentID,  
12         cf.CollegeName AS AdmittedCollege  
13     FROM  
14         ChoicesFilled cf  
15     JOIN Result r ON cf.ChoiceID = r.ChoiceID  
16     WHERE  
17         r.Status = 'Confirmed'  
18         AND r.RoundID = (  
19             SELECT MAX(RoundID)  
20             FROM Result r2  
21             WHERE r2.ChoiceID = r.ChoiceID  
22         )  
23 )  
24 SELECT  
25     COUNT(DISTINCT a.StudentID) AS StudentsWithFirstPreference  
26 FROM  
27     AdmittedStudents a  
28     JOIN FirstPreference fp ON a.StudentID = fp.StudentID  
29 WHERE  
30     fp.PreferenceRank = 1  
31     AND a.AdmittedCollege = fp.CollegeName;
```

Data Output Messages Notifications



studentswithfirstpreference	
bigint	
1	17



## 5) Percentage of students who got their college in their home state:

```
WITH AdmittedStudents AS (
  SELECT
    s.StudentID,
    s.State AS StudentState,
    c.State AS CollegeState
  FROM
    Student s
  JOIN ChoicesFilled cf ON s.StudentID = cf.StudentID
  JOIN Result r ON cf.ChoiceID = r.ChoiceID
  JOIN College c ON cf.CollegeName = c.CollegeName
  WHERE
    r.Status = 'Confirmed'
    AND r.RoundID = (
      SELECT MAX(RoundID)
      FROM Result r2
      WHERE r2.ChoiceID = r.ChoiceID
    )
),
HomeStateAdmissions AS (
  SELECT
    COUNT(DISTINCT StudentID) AS HomeStateCount
  FROM
    AdmittedStudents
  WHERE
    StudentState = CollegeState
),
TotalAdmitted AS (
  SELECT
    COUNT(DISTINCT StudentID) AS TotalAdmittedCount
  FROM
    AdmittedStudents
)
SELECT
  hsa.HomeStateCount,
  ta.TotalAdmittedCount,
  ROUND(
    ((hsa.HomeStateCount::NUMERIC / ta.TotalAdmittedCount) * 100)::NUMERIC,
    2
  ) AS HomeStatePercentage
FROM
  HomeStateAdmissions hsa,
  TotalAdmitted ta
WHERE
  ta.TotalAdmittedCount > 0;
```

Query Query History

```
1  WITH AdmittedStudents AS (  
2      SELECT  
3          s.StudentID,  
4          s.State AS StudentState,  
5          c.State AS CollegeState  
6      FROM  
7          Student s  
8      JOIN ChoicesFilled cf ON s.StudentID = cf.StudentID  
9      JOIN Result r ON cf.ChoiceID = r.ChoiceID  
10     JOIN College c ON cf.CollegeName = c.CollegeName  
11     WHERE  
12         r.Status = 'Confirmed'  
13         AND r.RoundID = (  
14             SELECT MAX(RoundID)  
15             FROM Result r2  
16             WHERE r2.ChoiceID = r.ChoiceID  
17         )  
18 ),  
19 HomeStateAdmissions AS (
```

Data Output Messages Notifications

	homestatecount bigint	totaladmittedcount bigint	homestatepercentage numeric
1	3	25	12.00

## 6) Students who did not get any seat in Round 1.

```
SELECT S.StudentID, S.FirstName, S.LastName
FROM Student S
WHERE NOT EXISTS (
    SELECT 1
    FROM ChoicesFilled CF
    JOIN Result R ON CF.ChoiceID = R.ChoiceID
    WHERE S.StudentID = CF.StudentID AND R.RoundID = 1 AND R.Status = 'Confirmed'
);
```

Query		Query History	
1	SELECT	S.StudentID, S.FirstName, S.LastName	
2	FROM	Student S	
3	WHERE NOT EXISTS (		
4	SELECT 1		
5	FROM	ChoicesFilled CF	
6	JOIN Result R ON	CF.ChoiceID = R.ChoiceID	
7	WHERE	S.StudentID = CF.StudentID AND R.RoundID = 1 AND R.Status = 'Confirmed'	
8	);		
9			

Data Output		Messages		Notifications	
studentid	firstname	lastname			
[PK] integer	character varying (50)	character varying (50)			
1	2025015	Aryan	Desai		
2	2025017	Samar	Malhotra		
3	2025018	Tara	Sen		
4	2025019	Veer	Rathore		
5	2025020	Ira	Joshi		
6	2025021	Rudra	Kumar		
7	2025022	Kyra	Sharma		
8	2025023	Shaurya	Patel		
9	2025024	Myra	Gupta		
10	2025025	Vihaan	Reddy		
11	2025026	Riya	Mehta		
12	2025027	Ayaan	Nair		
13	2025028	Sia	Das		
14	2025029	Neil	Verma		
15	2025030	Tia	Pandey		
16	2025031	Krish	Chauhan		
17	2025032	Esha	Bose		
18	2025033	Rohan	Saxena		
19	2025034	Nia	Khan		
20	2025035	Yash	Desai		

```
SELECT Category, AVG(JEE_Score) AS AverageScore
FROM Student
GROUP BY Category;
```

Query

Query History

1

2

3

4

SELECT Category, AVG(JEE\_Score) AS AverageScore  
FROM Student  
GROUP BY Category;

Data Output

Messages

Notifications

≡+

▼

▼

SQL

	category character varying (50)	averagescore numeric
1	ST	166.000000000000000000
2	SC	173.800000000000000000
3	General	181.7307692307692308

8) Students who did not pay the registration fees and got out of the admission process:

```
SELECT StudentID, FirstName, LastName
FROM Student
WHERE StudentID NOT IN (
    SELECT StudentID FROM Payments
);
```

Query		Query History	
1	▼	SELECT StudentID, FirstName, LastName	
2		FROM Student	
3		WHERE StudentID NOT IN (	
4		SELECT StudentID FROM Payments	
5		);	
6			

Data Output		Messages	Notifications
<div></div>			
	studentid [PK] integer	firstname character varying (50)	lastname character varying (50)
1	2025015	Aryan	Desai
2	2025025	Vihaan	Reddy
3	2025035	Yash	Desai

9) Top 3 IITs with most no. of preference:

```
SELECT CF.CollegeName, COUNT(*) AS PreferenceCount
FROM ChoicesFilled CF
WHERE CF.CollegeName LIKE 'IIT%'
GROUP BY CF.CollegeName
ORDER BY PreferenceCount DESC
LIMIT 3;
```

Query		Query History	
1	SELECT	CF.CollegeName, COUNT(*) AS PreferenceCount	
2	FROM	ChoicesFilled CF	
3	WHERE	CF.CollegeName LIKE 'IIT%'	
4	GROUP BY	CF.CollegeName	
5	ORDER BY	PreferenceCount DESC	
6	LIMIT	3;	
7			

Data Output		Messages	Notifications
	collegename character varying (100)		preferencecount bigint
1	IIT Hyderabad		23
2	IIT Guwahati		17
3	IIT Gandhinagar		14

10) Colleges which provide Aerospace Engineering or Data Science and Artificial Intelligence

```
SELECT O.CollegeName, B.BranchName
FROM Offers O
JOIN Branch B ON O.BranchID = B.BranchID
WHERE O.BranchID IN (107, 108)
ORDER BY O.CollegeName, O.BranchID;
```

Query Query History

```
1 SELECT O.CollegeName, B.BranchName
2 FROM Offers O
3 JOIN Branch B ON O.BranchID = B.BranchID
4 WHERE O.BranchID IN (107, 108)
5 ORDER BY O.CollegeName, O.BranchID;
6
```

Data Output Messages Notifications

	collegename character varying (100)	branchname character varying (100)
1	IIT Bombay	Aerospace Engineering
2	IIT Bombay	Data Science and Artificial Intelligen...
3	IIT Delhi	Data Science and Artificial Intelligen...
4	IIT Hyderabad	Data Science and Artificial Intelligen...
5	IIT Kanpur	Aerospace Engineering
6	IIT Kharagpur	Aerospace Engineering
7	IIT Madras	Aerospace Engineering
8	IIT Madras	Data Science and Artificial Intelligen...

## 11) Category-Wise Round-Wise Closing Rank per IIT and Branch:

```
WITH AllottedStudents AS (  
    SELECT  
        S.StudentID,  
        S.AIR,  
        S.Category,  
        R.RoundID,  
        CF.CollegeName,  
        CF.BranchID  
    FROM Student S  
    JOIN ChoicesFilled CF ON S.StudentID = CF.StudentID  
    JOIN Result R ON CF.ChoiceID = R.ChoiceID  
    WHERE R.Status = 'Confirmed'  
)  
IITsOnly AS (  
    SELECT *  
    FROM AllottedStudents  
    WHERE CollegeName LIKE 'IIT%'  
)  
MaxRanks AS (  
    SELECT  
        CollegeName,  
        BranchID,  
        Category,  
        RoundID,  
        MAX(AIR) AS ClosingRank  
    FROM IITsOnly  
    GROUP BY CollegeName, BranchID, Category, RoundID  
)  
SELECT  
    M.CollegeName,  
    C.Address AS CollegeAddress,  
    B.BranchName,  
    M.Category,  
    M.RoundID,  
    M.ClosingRank  
FROM MaxRanks M  
JOIN College C ON M.CollegeName = C.CollegeName  
JOIN Branch B ON M.BranchID = B.BranchID  
ORDER BY M.CollegeName, M.BranchID, M.Category, M.RoundID;
```



Query		Query History					
10	JOIN ChoicesFilled CF ON S.StudentID = CF.StudentID						
11	JOIN Result R ON CF.ChoiceID = R.ChoiceID						
12	WHERE R.Status = 'Confirmed'						
13	),						
14	IITsOnly AS (						
15	SELECT *						
16	FROM AllottedStudents						
17	WHERE CollegeName LIKE 'IIT%'						
18	),						
Data Output		Messages					
		Notifications					
		SQL					
	collegename character varying (100)	collegeaddress text	branchname character varying (100)	category character varying (50)	roundid integer	closingrank integer	
1	IIT Bombay	Powai	Computer Science	General	1	175	
2	IIT Delhi	Hauz Khas	Computer Science	General	1	170	
3	IIT Delhi	Hauz Khas	Computer Science	SC	1	180	
4	IIT Gandhinagar	Palaj	Civil Engineering	SC	3	3050	
5	IIT Gandhinagar	Palaj	Chemical Engineering	General	2	2650	
6	IIT Gandhinagar	Palaj	Chemical Engineering	SC	2	2800	
7	IIT Guwahati	Amingaon	Electrical Engineering	General	2	2400	
8	IIT Guwahati	Amingaon	Electrical Engineering	ST	3	2900	
9	IIT Guwahati	Amingaon	Chemical Engineering	General	3	2850	
10	IIT Hyderabad	Kandi	Computer Science	General	1	2100	
11	IIT Hyderabad	Kandi	Electrical Engineering	SC	2	2450	
12	IIT Hyderabad	Kandi	Civil Engineering	General	3	3150	
13	IIT Kanpur	Kalyanpur	Computer Science	General	1	1950	
14	IIT Kanpur	Kalyanpur	Electrical Engineering	General	1	2200	
15	IIT Kanpur	Kalyanpur	Electronics and Communication Engineeri...	General	1	2000	
16	IIT Kharagpur	Kharagpur	Computer Science	SC	1	2050	
17	IIT Kharagpur	Kharagpur	Mechanical Engineering	General	3	2350	
18	IIT Kharagpur	Kharagpur	Electrical Engineering	General	1	2150	
19	IIT Kharagpur	Kharagpur	Electrical Engineering	ST	1	2300	
20	IIT Madras	Sardar Patel Ro...	Computer Science	General	1	165	
21	IIT Madras	Sardar Patel Ro...	Computer Science	ST	1	1900	
22	IIT Roorkee	Roorkee Campus	Computer Science	General	1	185	

12) For each reason for the calls made to the help desk, the total number of calls for that reason is displayed :

```
SELECT Call_for AS Reason, COUNT(*) AS NumberOfCalls
FROM Calls
GROUP BY Call_for
ORDER BY NumberOfCalls DESC;
```

Query		Query History	
1	▼	SELECT Call_for AS Reason, COUNT(*) AS NumberOfCalls	
2		FROM Calls	
3		GROUP BY Call_for	
4		ORDER BY NumberOfCalls DESC;	
5			
Data Output		Messages	
		Notifications	
	reason character varying (50) 🔒	numberofcalls bigint 🔒	
1	documents	7	
2	cutoffquery	5	
3	paymentquery	4	
4	allocationquery	4	

