TASK 2:

- **CREATE TABLE STUDENT:-**

```
CREATE TABLE students (
student_id INT PRIMARY KEY,
name VARCHAR(100),
score INT,
class VARCHAR(50),
exam_date DATE
);
```

- **INSERTING THE DATA IN TABLE:-**

```
INSERT INTO students (student_id, name, score, class, exam_date)
VALUES
(1, 'Bhavesh', 85, 'Math', '2025-01-15'),
(2, 'Sushil', 78, 'Math', '2025-01-15'),
(3, 'Payal', 92, 'Math', '2025-01-15'),
(4, 'Avira', 75, 'Science', '2025-01-16'),
(5, 'Chhaku', 82, 'Science', '2025-01-16'),
(6, 'Pranali', 82, 'Math', '2025-01-20'),
(7, 'Tanu', 91, 'Math', '2025-01-20');
```

student_id	name	score	class	exam_date
1	Bhavesh	85	Math	2025-01-15
2	Sushil	78	Math	2025-01-15
3	Payal	92	Math	2025-01-15
4	Avira	75	Science	2025-01-16
5	Chhaku	82	Science	2025-01-16
6	Pranali	82	Math	2025-01-20
7	Tanu	91	Math	2025-01-20

- **PERFORM ADVANCED DATA ANALYSIS:-** Rank Students by Score in Each Class

```
SELECT student_id,
name,
score,
class,
RANK() OVER (PARTITION BY class ORDER BY score DESC) AS rank_within_class
FROM students;
```

student_id	name	score	class	rank_within_class
3	Payal	92	Math	1
7	Tanu	91	Math	2
1	Bhavesh	85	Math	3
6	Pranali	82	Math	4
2	Sushil	78	Math	5
5	Chhaku	82	Science	1
4	Avira	75	Science	2

- **SUBQUERY EXAMPLE:-** Find Students Who Scored Above the Average in Their Class

```
SELECT student_id,
name,
score,
class
FROM students
WHERE score > (
SELECT AVG(score)
FROM students
WHERE class = students.class
);
```

student_id	name	score	class
1	Bhavesh	85	Math
3	Payal	92	Math
7	Tanu	91	Math

- **COMMON TABLE EXPRESSION (CTE):** Top 3 Students by Score in Each Class

```
WITH ranked_students AS (
SELECT student_id,
name,
score,
class,
RANK() OVER (PARTITION BY class ORDER BY score DESC) AS rank_within_class
FROM students
)
SELECT student_id,
name,
score,
class
FROM ranked_students
WHERE rank_within_class <= 3;
```

student_id	name	score	class
3	Payal	92	Math
7	Tanu	91	Math
1	Bhavesh	85	Math
5	Chhaku	82	Science
4	Avira	75	Science

- **WINDOW FUNCTION :-** Calculate Cumulative Score for Each Student

```
SELECT student_id,
name,
score,
exam_date,
SUM(score) OVER (PARTITION BY student_id ORDER BY exam_date) AS cumulative_score
FROM students;
```

student_id	name	score	exam_date	cumulative_score
1	Bhavesh	85	2025-01-15	85
2	Sushil	78	2025-01-15	78
3	Payal	92	2025-01-15	92
4	Avira	75	2025-01-16	75
5	Chhaku	82	2025-01-16	82
6	Pranali	82	2025-01-20	82
7	Tanu	91	2025-01-20	91

- **SUBQEURY**:- Find Students Who Scored Above the Overall Average Score

```
SELECT student_id,
name,
score,
class
FROM students
WHERE score > (
SELECT AVG(score)
FROM students
);
```

student_id	name	score	class
1	Bhavesh	85	Math
3	Payal	92	Math
7	Tanu	91	Math

- **CTE AND WINDOW FUNCTION:**- Find Students Who Scored Above the Class Average and Rank Them.

```
WITH class_avg_score AS (
SELECT class,
AVG(score) AS avg_score
FROM students
GROUP BY class
)
SELECT s.student_id,
s.name,
```

```
s.score,
s.class,
RANK() OVER (PARTITION BY s.class ORDER BY s.score DESC) AS rank_within_class
FROM students s
JOIN class_avg_score cas
ON s.class = cas.class
WHERE s.score > cas.avg_score;
```

student_id	name	score	class	rank_within_class
3	Payal	92	Math	1
7	Tanu	91	Math	2
5	Chhaku	82	Science	1

- **Find the Best Scorer in Each Class Using SUBQEURY**

```
SELECT student_id,
name,
score,
class
FROM students
WHERE (class, score) IN (
SELECT class, MAX(score)
FROM students
GROUP BY class
);
```

student_id	name	score	class
3	Payal	92	Math
5	Chhaku	82	Science