📄 Full MySQL Notes (Post-CASE)

👨‍💻 With Output (Tabular Format)

✅ STEP 1: Create Table `students` & Insert Data

```
CREATE TABLE students (
  id INT PRIMARY KEY,
  name VARCHAR(50),
  age INT,
  grade VARCHAR(5),
  city VARCHAR(50)
);

INSERT INTO students (id, name, age, grade, city) VALUES
(1, 'Rahul', 18, 'A', 'Delhi'),
(2, 'Priya', 19, 'B', 'Mumbai'),
(3, 'Aman', 17, 'A', 'Pune'),
(4, 'Neha', 18, 'C', 'Delhi'),
(5, 'Ravi', 20, 'B', 'Chennai');
```

✅ ORDER BY

```
SELECT * FROM students ORDER BY age ASC;
```

📤 Output:

id	name	age	grade	city
3	Aman	17	A	Pune
1	Rahul	18	A	Delhi
4	Neha	18	C	Delhi
2	Priya	19	B	Mumbai
5	Ravi	20	B	Chennai

✅ ORDER BY + LIMIT

```
SELECT * FROM students ORDER BY age DESC LIMIT 3;
```

📤 Output:

id	name	age	grade	city
5	Ravi	20	B	Chennai
2	Priya	19	B	Mumbai

id	name	age	grade	city
1	Rahul	18	A	Delhi

✅ GROUP BY + COUNT()

```
SELECT city, COUNT(*) AS total_students
FROM students
GROUP BY city;
```

📤 Output:

city	total_students
Delhi	2
Mumbai	1
Pune	1
Chennai	1

```
SELECT grade, COUNT(*) AS grade_count
FROM students
GROUP BY grade;
```

📤 Output:

grade	grade_count
A	2
B	2
C	1

✅ UPDATE Queries

🔼 Update Priya's city:

```
UPDATE students
SET city = 'Bangalore'
WHERE name = 'Priya';

SELECT name, city FROM students;
```

📤 Output:

name	city
Rahul	Delhi
Priya	Bangalore
Aman	Pune

name	city
Neha	Delhi
Ravi	Chennai

🔻 Increase everyone's age by 1:

```
UPDATE students
SET age = age + 1;

SELECT name, age FROM students;
```

📤 Output:

name	age
Rahul	19
Priya	20
Aman	18
Neha	19
Ravi	21

✅ DELETE Queries

🔻 Delete Neha:

```
DELETE FROM students
WHERE name = 'Neha';

SELECT * FROM students;
```

📤 Output:

id	name	age	grade	city
1	Rahul	19	A	Delhi
2	Priya	20	B	Bangalore
3	Aman	18	A	Pune
5	Ravi	21	B	Chennai

🔻 Delete students from Delhi:

```
DELETE FROM students
WHERE city = 'Delhi';

SELECT * FROM students;
```

📤 Output:

id	name	age	grade	city
2	Priya	20	B	Bangalore
3	Aman	18	A	Pune
5	Ravi	21	B	Chennai

✅ AGGREGATE FUNCTIONS

🔹 Total Age of All Students

```
SELECT SUM(age) AS total_age FROM students;
```

📤 Output:

total_age
59

🔹 Average Age

```
SELECT AVG(age) AS average_age FROM students;
```

📤 Output:

average_age
19.66

🔹 Max & Min Age

```
SELECT MAX(age) AS oldest, MIN(age) AS youngest FROM students;
```

📤 Output:

oldest	youngest
21	18

📘 MySQL – Aggregate Functions Notes with Table & Output

□ Step 1: Create Sample Table

```
CREATE TABLE students (
  id INT,
  name VARCHAR(50),
  gender VARCHAR(10),
  age INT
);
```

□ Step 2: Insert Sample Data

```
INSERT INTO students (id, name, gender, age) VALUES
(1, 'Ankit', 'Male', 18),
(2, 'Riya', 'Female', 20),
(3, 'Aman', 'Male', 21),
(4, 'Sneha', 'Female', 18),
(5, 'Rohit', 'Male', 20);
```

📊 Data in `students` Table:

id	name	gender	age
1	Ankit	Male	18
2	Riya	Female	20
3	Aman	Male	21
4	Sneha	Female	18
5	Rohit	Male	20

✅ 1. Total Age – `SUM()`

```
SELECT SUM(age) AS total_age FROM students;
```

👉 Sari age ka total karega.

📤 Output:

total_age
97

✅ 2. Average Age – `AVG()`

```
SELECT AVG(age) AS average_age FROM students;
```

👉 Sab students ki average age nikalta hai.

📤 Output:

average_age
19.4

✅ 3. Max & Min Age – `MAX()` / `MIN()`

```
SELECT MAX(age) AS oldest, MIN(age) AS youngest FROM students;
```

👉 Sabse bada aur sabse chhota age show karega.

📤 Output:

oldest	youngest
21	18

✅ 4. Total Students – `COUNT(*)`

```
SELECT COUNT(*) AS total_students FROM students;
```

👉 Kitne students total hain wo batata hai.

📤 Output:

total_students
5

✅ 5. Count by Gender – `GROUP BY`

```
SELECT gender, COUNT(*) AS total FROM students GROUP BY gender;
```

👉 Har gender me kitne students hain wo show karega.

📤 Output:

gender	total
Male	3
Female	2

✅ 6. Average Age by Gender – `GROUP BY + AVG ()`

```
SELECT gender, AVG(age) AS avg_age FROM students GROUP BY gender;
```

☞ Har gender ki alag average age show karta hai.

📤 Output:

gender	avg_age
Male	19.66
Female	19.00

✅ 7. Filter Groups – `HAVING`

```
SELECT gender, COUNT (*) AS total
FROM students
GROUP BY gender
HAVING total > 1;
```

☞ Sirf un genders ko dikhata hai jinke students 1 se zyada hain.

📤 Output:

gender	total
Male	3
Female	2

✅ 8. Unique Genders – `COUNT(DISTINCT)`

```
SELECT COUNT(DISTINCT gender) AS unique_genders FROM students;
```

☞ Table me kitne unique gender values hain wo count karega.

📤 Output:

unique_genders
2

Yeh chapter complete