

## GROUP BY

SQL GROUP BY - বিস্তৃত ব্যাখ্যা (বাংলায়)

টেবিল ডেটা:

teacher2 Table:

T\_id T\_name T\_dept T\_salary

201	Azad	CSE	1000
202	Kamal	EEE	2000
203	Hasan	CE	1000
204	Kamal	EEE	1000
205	Rony	CSE	2000
206	Rakib	CSE	4000

---

### 1. Basic GROUP BY (শৰ্ধু Grouping)

কোড:

```
SELECT T_dept  
FROM teacher2  
GROUP BY T_dept;
```

ব্যাখ্যা:

- **GROUP BY** একই ধরনের data-কে একসাথে গ্রুপ করে
- এখানে department অনুযায়ী group করা হচ্ছে
- একই department-এর সব rows একটা group-এ থাকবে
- আউটপুটে প্রতি department-এর জন্য **একটা মাত্র row** আসবে
- এটা মূলত **DISTINCT**-এর মতো কাজ করে যখন aggregate function না থাকে

আউটপুট:

T\_dept

CSE

EEE

CE

মনে রাখার কথা: একই value-গুলোকে একসাথে করে unique list দেখায়

---

## 2. GROUP BY with COUNT (গণনা করা)

কোড:

```
SELECT T_dept, COUNT(*) AS total_teachers  
FROM teacher2  
GROUP BY T_dept;
```

ব্যাখ্যা:

- COUNT(\*) প্রতি group-এ কতগুলো rows আছে তা গণনা করে
- প্রতি department-এ কতজন teacher আছে তা বের করছি
- CSE-তে ৩ জন, EEE-তে ২ জন, CE-তে ১ জন
- \* মানে সব columns count করবে (NULL-সহ)
- COUNT(column\_name) ব্যবহার করলে NULL skip করবে

আউটপুট:

T\_dept total\_teachers

CSE 3

EEE 2

CE 1

মনে রাখার কথা: COUNT = কতগুলো আছে

---

### 3. GROUP BY with SUM (মোগফল)

কোড:

```
SELECT T_dept, SUM(T_salary) AS total_salary  
FROM teacher2  
GROUP BY T_dept;
```

ব্যাখ্যা:

- **SUM()** প্রতি group-এর সব মান যোগ করে
- প্রতি department-এর মোট বেতন বের করছি
- CSE:  $1000 + 2000 + 4000 = \mathbf{7000}$
- EEE:  $2000 + 1000 = \mathbf{3000}$
- CE:  $1000 = \mathbf{1000}$
- সংখ্যাসূচক (numeric) column-এই শুধু SUM করা যায়

আউটপুট:

T\_dept total\_salary

CSE 7000

EEE 3000

CE 1000

মনে রাখার কথা: SUM = সব মিলিয়ে করে

---

### 4. GROUP BY with MIN (সর্বনিম্ন)

কোড:

```
SELECT T_dept, MIN(T_salary) AS min_salary  
FROM teacher2  
GROUP BY T_dept;
```

### ব্যাখ্যা:

- **MIN()** প্রতি group-এর সবচেয়ে ছোট মান খুঁজে বের করে
- প্রতি department-এ সবচেয়ে কম বেতন করে তা দেখছি
- CSE-তে সবচেয়ে কম: 1000 (Azad)
- EEE-তে সবচেয়ে কম: 1000 (Kamal)
- CE-তে শুধু একজন তাই: 1000 (Hasan)

### আউটপুট:

T\_dept min\_salary

CSE 1000

EEE 1000

CE 1000

মনে রাখার কথা: MIN = সবচেয়ে ছোট

---

### 5. GROUP BY with MAX (সর্বোচ্চ)

#### কোড:

```
SELECT T_dept, MAX(T_salary) AS max_salary  
FROM teacher2  
GROUP BY T_dept;
```

### ব্যাখ্যা:

- **MAX()** প্রতি group-এর সবচেয়ে বড় মান খুঁজে বের করে
- প্রতি department-এ সবচেয়ে বেশি বেতন করে তা দেখছি
- CSE-তে সবচেয়ে বেশি: 4000 (Rakib)
- EEE-তে সবচেয়ে বেশি: 2000 (Kamal - ID 202)
- CE-তে শুধু একজন তাই: 1000 (Hasan)

**আউটপুট:**

**T\_dept max\_salary**

CSE 4000

EEE 2000

CE 1000

**মনে রাখার কথা:** MAX = সবচেয়ে বড়

---

## 6. GROUP BY with AVG (গড়)

**কোড:**

```
SELECT T_dept, AVG(T_salary) AS avg_salary  
FROM teacher2  
GROUP BY T_dept;
```

**ব্যাখ্যা:**

- **AVG()** প্রতি group-এর গড় (average) বের করে
- প্রতি department-এর গড় বেতন বের করছি
- CSE:  $(1000 + 2000 + 4000) \div 3 = 2333.33$
- EEE:  $(2000 + 1000) \div 2 = 1500.00$
- CE:  $1000 \div 1 = 1000.00$
- দশমিক (decimal) আকারে আসবে

**আউটপুট:**

**T\_dept avg\_salary**

CSE 2333.33

EEE 1500.00

**T\_dept avg\_salary**

CE 1000.00

মনে রাখার কথা: AVG = গড়

---

## 7. GROUP BY with WHERE (শর্ত দিয়ে filtering)

কোড:

```
SELECT T_dept, COUNT(*) AS total_teachers  
FROM teacher2  
WHERE T_id > 201  
GROUP BY T_dept;
```

ব্যাখ্যা:

- WHERE clause grouping-এর আগে কাজ করে
- প্রথমে T\_id > 201 শর্ত অনুযায়ী rows filter করা হয়
- শুধু T\_id 202, 203, 204, 205, 206 নেওয়া হয় (201 বাদ)
- তারপর সেগুলো department অনুযায়ী group করা হয়
- WHERE = individual rows filter করে

Filter হওয়ার পর ডেটা:

**T\_id T\_name T\_dept T\_salary**

202	Kamal	EEE	2000
203	Hasan	CE	1000
204	Kamal	EEE	1000
205	Rony	CSE	2000
206	Rakib	CSE	4000

আউটপুট:

### **T\_dept total\_teachers**

CSE      2

EEE      2

CE        1

**মনে রাখার কথা:** WHERE = grouping-এর আগে filter

---

### **8. GROUP BY with HAVING (group filter করা)**

**কোড:**

```
SELECT T_dept, SUM(T_salary) AS total_salary
```

```
FROM teacher2
```

```
GROUP BY T_dept
```

```
HAVING SUM(T_salary) > 2000;
```

**ব্যাখ্যা:**

- **HAVING** clause grouping-এর পরে কাজ করে
- প্রথমে department অনুযায়ী group করা হয়
- তারপর যে groups-এর মোট বেতন 2000-এর বেশি শুধু সেগুলো দেখানো হয়
- HAVING = **groups** filter করে ( aggregate functions-এর উপর শর্ত )
- WHERE ব্যবহার করা যাবে না কারণ SUM() aggregate function

**সব groups (HAVING এর আগে):**

### **T\_dept total\_salary**

CSE      7000 ✓

EEE      3000 ✓

CE        1000 X

**আউটপুট:**

**T\_dept total\_salary**

CSE 7000

EEE 3000

**মনে রাখার কথা:** HAVING = grouping-এর পরে filter

---

## 9. GROUP BY Multiple Columns (একাধিক column)

**কোড:**

```
SELECT T_dept, T_salary, COUNT(*) AS count_teachers  
FROM teacher2  
GROUP BY T_dept, T_salary;
```

**ব্যাখ্যা:**

- একাধিক column দিয়ে group করা যায়
- একই department এবং একই salary যাদের তারা একই group-এ থাকবে
- প্রতি department-এ কোন বেতনে কতজন আছে তা দেখা যাবে
- Combination অনুযায়ী group হয়

**আউটপুট:**

**T\_dept T\_salary count\_teachers**

CSE 1000 1

CSE 2000 1

CSE 4000 1

EEE 2000 1

EEE 1000 1

**T\_dept T\_salary count\_teachers**

CE      1000      1

**ব্যাখ্যা:**

- CSE-তে 1000 টাকা বেতন পান = 1 জন (Azad)
- CSE-তে 2000 টাকা বেতন পান = 1 জন (Rony)
- CSE-তে 4000 টাকা বেতন পান = 1 জন (Rakib)
- EEE-তে 2000 টাকা বেতন পান = 1 জন (Kamal-202)
- EEE-তে 1000 টাকা বেতন পান = 1 জন (Kamal-204)

**মনে রাখার কথা:** Multiple columns = আরো specific grouping

---

## **10. GROUP BY with ORDER BY (সাজানো)**

**কোড:**

```
SELECT T_dept, SUM(T_salary) AS total_salary  
FROM teacher2  
GROUP BY T_dept  
ORDER BY total_salary DESC;
```

**ব্যাখ্যা:**

- **ORDER BY** grouping-এর পরে results সাজায়
- এখানে মোট বেতন অনুযায়ী **বড় থেকে ছোট (DESC)** সাজানো হচ্ছে
- Execution order: GROUP BY → calculate aggregates → ORDER BY
- ORDER BY aggregate column (total\_salary) ব্যবহার করা যায়

**আউটপুট:**

### **T\_dept total\_salary**

CSE 7000

EEE 3000

CE 1000

**মনে রাখার কথা:** ORDER BY = results সাজানোর জন্য (সবার শেষে)

---

### **11. GROUP BY Specific Department (নির্দিষ্ট dept)**

**কোড:**

```
SELECT T_dept, MAX(T_salary) AS max_salary_CSE  
FROM teacher2  
WHERE T_dept = 'CSE'  
GROUP BY T_dept;
```

**ব্যাখ্যা:**

- **WHERE** দিয়ে প্রথমে শুধু CSE department filter করা হয়
- তারপর সেই department-এর সর্বোচ্চ বেতন বের করা হয়
- যেহেতু একটা department-ই আছে, তাই একটা row আসবে
- CSE-তে সবচেয়ে বেশি বেতন পান Rakib (4000)

**Filter হওয়ার পর ডেটা:**

### **T\_id T\_name T\_dept T\_salary**

201 Azad CSE 1000

205 Rony CSE 2000

206 Rakib CSE 4000

**আউটপুট:**

T\_dept max\_salary\_CSE

CSE 4000

মনে রাখার কথা: WHERE দিয়ে নির্দিষ্ট group-এর data নেওয়া

---

## 12. Multiple Aggregate Functions একসাথে

কোড:

```
SELECT T_dept,
       COUNT(*) AS total_teachers,
       SUM(T_salary) AS total_salary,
       AVG(T_salary) AS avg_salary,
       MIN(T_salary) AS min_salary,
       MAX(T_salary) AS max_salary
  FROM teacher2
 GROUP BY T_dept;
```

ব্যাখ্যা:

- একই query-তে **একাধিক aggregate functions** ব্যবহার করা যায়
- প্রতি department-এর জন্য সব statistics একসাথে দেখা যাবে
- খুব useful যখন comprehensive report চাই

আউটপুট:

T\_dept total\_teachers total\_salary avg\_salary min\_salary max\_salary

CSE	3	7000	2333.33	1000	4000
EEE	2	3000	1500.00	1000	2000
CE	1	1000	1000.00	1000	1000

মনে রাখার কথা: Multiple aggregates = complete analysis

---

## Aggregate Functions তালিকা:

Function কাজ      উদাহরণ

**COUNT()** গণনা করে COUNT(\*) = 3

**SUM()** যোগফল SUM(salary) = 7000

**AVG()** গড় AVG(salary) = 2333.33

**MIN()** সর্বনিম্ন MIN(salary) = 1000

**MAX()** সর্বোচ্চ MAX(salary) = 4000

---

## WHERE vs HAVING:

বিষয়                  WHERE                  HAVING

কথন কাজ করে      GROUP BY-এর আগে      GROUP BY-এর পরে

কী filter করে      Individual rows      Groups

Aggregate function ব্যবহার করা যায় না      ব্যবহার করা যায়

উদাহরণ                  WHERE T\_salary > 1000 HAVING SUM(T\_salary) > 2000

---

## SQL Query Execution Order (কোনটা আগে চলে):

1. FROM      → table select করা
2. WHERE      → individual rows filter করা
3. GROUP BY      → rows group করা
4. HAVING      → groups filter করা
5. SELECT      → columns select করা

6. ORDER BY → results সাজানো

7. LIMIT → নির্দিষ্ট সংখ্যক rows নেওয়া

মনে রাখার সহজ উপায়:

"ফাঁসি We Get Happy Sometimes Often Late"

- FROM
  - WHERE
  - GROUP BY
  - HAVING
  - SELECT
  - ORDER BY
  - LIMIT
- 

GROUP BY ব্যবহারের নিয়ম:

1.  SELECT-এ যে columns আছে, সেগুলো হয় GROUP BY-তে থাকতে হবে, না হয় aggregate function-এর ভিতরে
2.  Aggregate functions: COUNT, SUM, AVG, MIN, MAX
3.  WHERE = rows filter (aggregate function ছাড়া)
4.  HAVING = groups filter (aggregate function সহ)
5.  একাধিক column দিয়ে group করা যায়
6.  ORDER BY সবার শেষে ব্যবহার করতে হয়

ভুল উদাহরণ:

--  ভুল: T\_name GROUP BY-তে নেই এবং aggregate function-এও নেই

```
SELECT T_dept, T_name, COUNT(*)
```

```
FROM teacher2
```

```
GROUP BY T_dept;
```

## সঠিক উদাহরণ:

-- সঠিক: T\_name GROUP BY-তে আছে

```
SELECT T_dept, T_name, COUNT(*)
```

```
FROM teacher2
```

```
GROUP BY T_dept, T_name;
```

---

## বাস্তব জীবনে ব্যবহার:

- **Sales Report:** প্রতি মাসে কত টাকা বিক্রয় হয়েছে
- **Employee Analysis:** প্রতি department-এ কতজন employee
- **Student Grades:** প্রতি class-এর গড় নম্বর
- **Product Sales:** কোন product সবচেয়ে বেশি বিক্রি হয়েছে
- **Financial Reports:** মাসিক আয়-ব্যয়ের হিসাব