**Example: Incorrect Use of GROUP BY**

**❌ Incorrect SQL Query:**

sql

CopyEdit

SELECT Author, COUNT(\*), Title FROM Books GROUP BY Author;

Assume the Books table has columns: BookID, Title, Author, PublishedYear.

**❗ Step 1: Run the Query**

When you execute the above, SQL Server gives this error:

pgsql

CopyEdit

Msg 8120, Level 16, State 1, Line 1

Column 'Books.Title' is invalid in the select list because it is not contained in either an aggregate function or the GROUP BY clause.

**✅ Step-by-Step Fix**

**🔸 Step 2: Understand the Error Message**

SQL Server is saying: You can't SELECT Title unless:

* It’s **in the GROUP BY clause**, or
* It’s used inside an **aggregate function** (like MAX, MIN, COUNT, etc.)

Because you're grouping by Author, SQL Server needs to know **how to combine multiple titles per author**.

**🔸 Step 3: Fix the Query Based on What You Want**

**✅ Option 1: Remove Title if it's not needed:**

sql

CopyEdit

SELECT Author, COUNT(\*) AS BookCount

FROM Books

GROUP BY Author;

**✅ Option 2: Use an aggregate function on Title, like MAX (or MIN, STRING\_AGG in newer versions):**

sql

CopyEdit

SELECT Author, COUNT(\*) AS BookCount, MAX(Title) AS ExampleTitle

FROM Books

GROUP BY Author;

**✅ Option 3: Get one row per book (don’t use GROUP BY):**

sql

CopyEdit

SELECT Author, Title FROM Books;

**🔸 Step 4: Run the Corrected Query**

For example:

sql

CopyEdit

SELECT Author, COUNT(\*) AS BookCount

FROM Books

GROUP BY Author;

This now works and returns the number of books per author.

**🧠 Summary**

| **Mistake** | **Fix** |
| --- | --- |
| Selecting non-aggregated, non-grouped column | Use an aggregate function or include it in GROUP BY |
| Error Msg 8120 | Add the column to GROUP BY or wrap with MAX, MIN, etc. |

Would you like to try fixing a broken GROUP BY query yourself? I can give you a sample for practice.