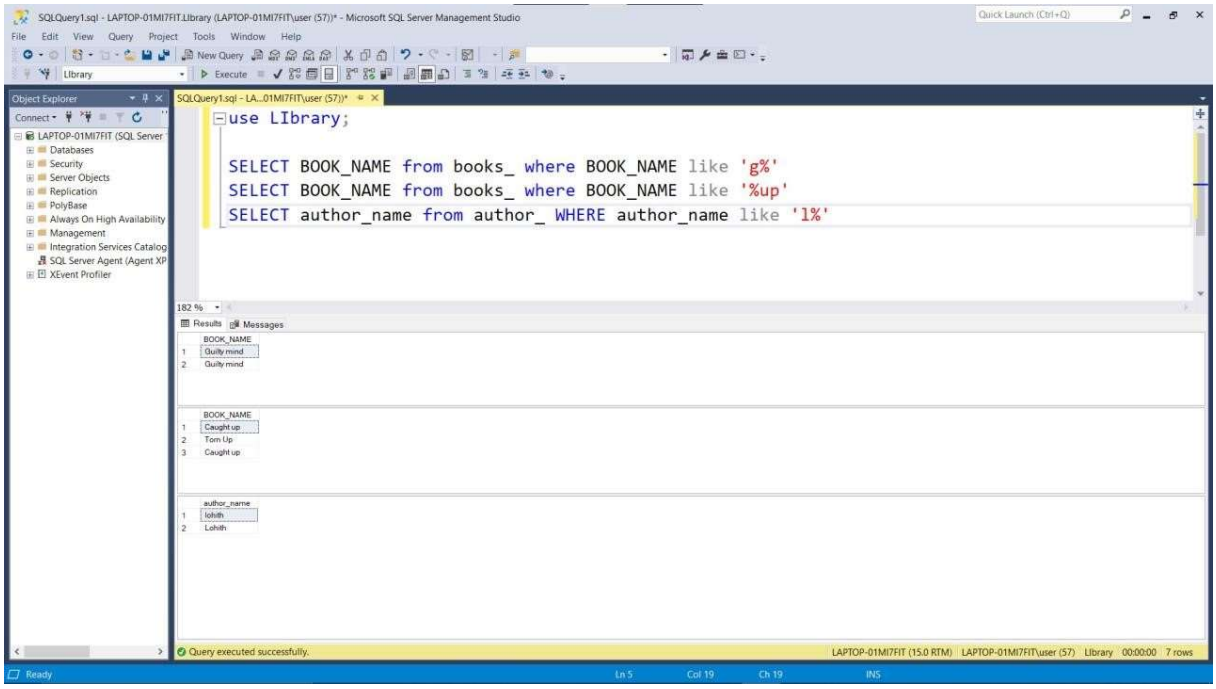
DBMS LAB ASSIGNMENT-5

Name:- MATHANGI SRAVAN

Roll no:-19BCS068

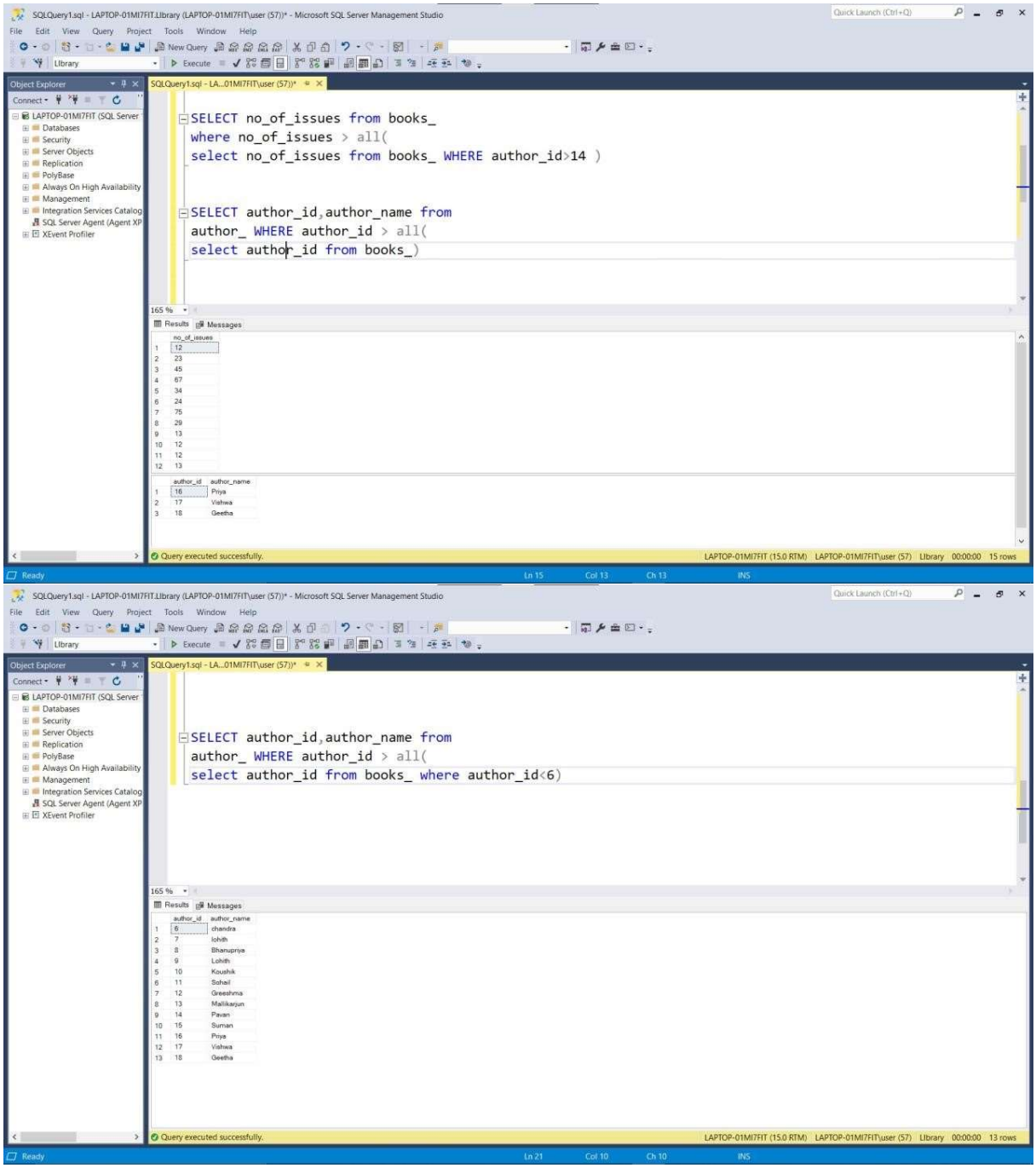
1)Illustrate logical ANY, ALL and LIKE operator- the queries should be relevant to your respective databases 3 queries for each operator. One query explaining the difference between ANY and ALL

i) LIKE :-



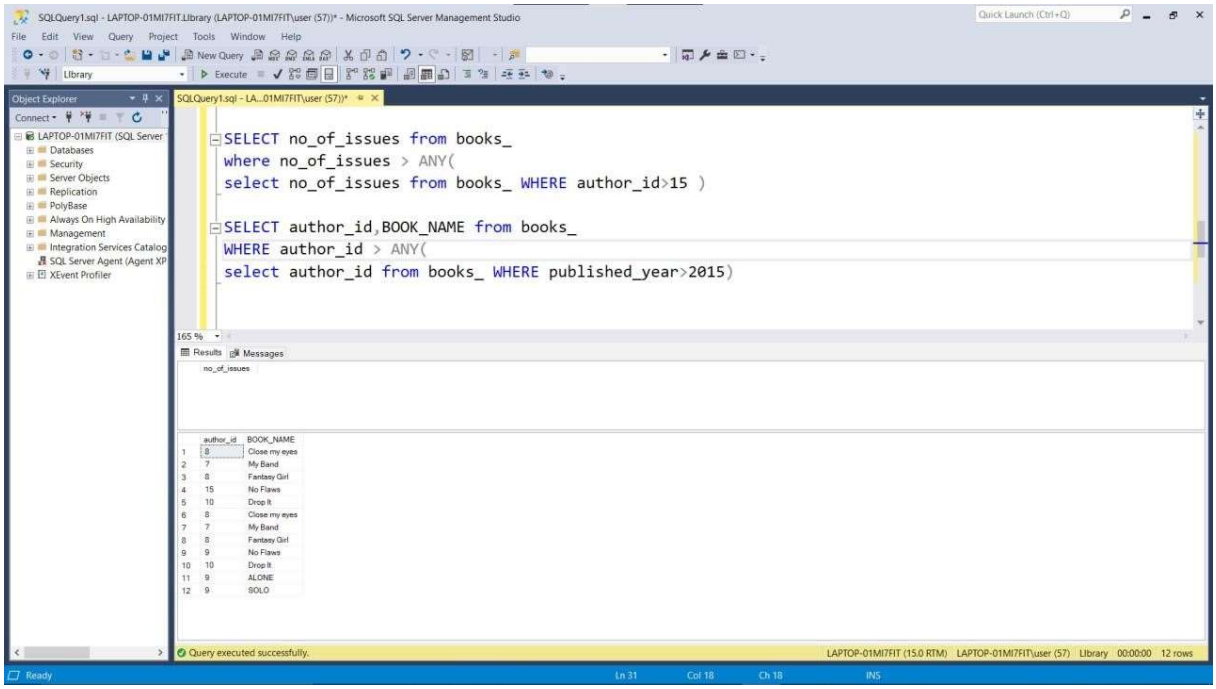
~ ii) ALL

:-

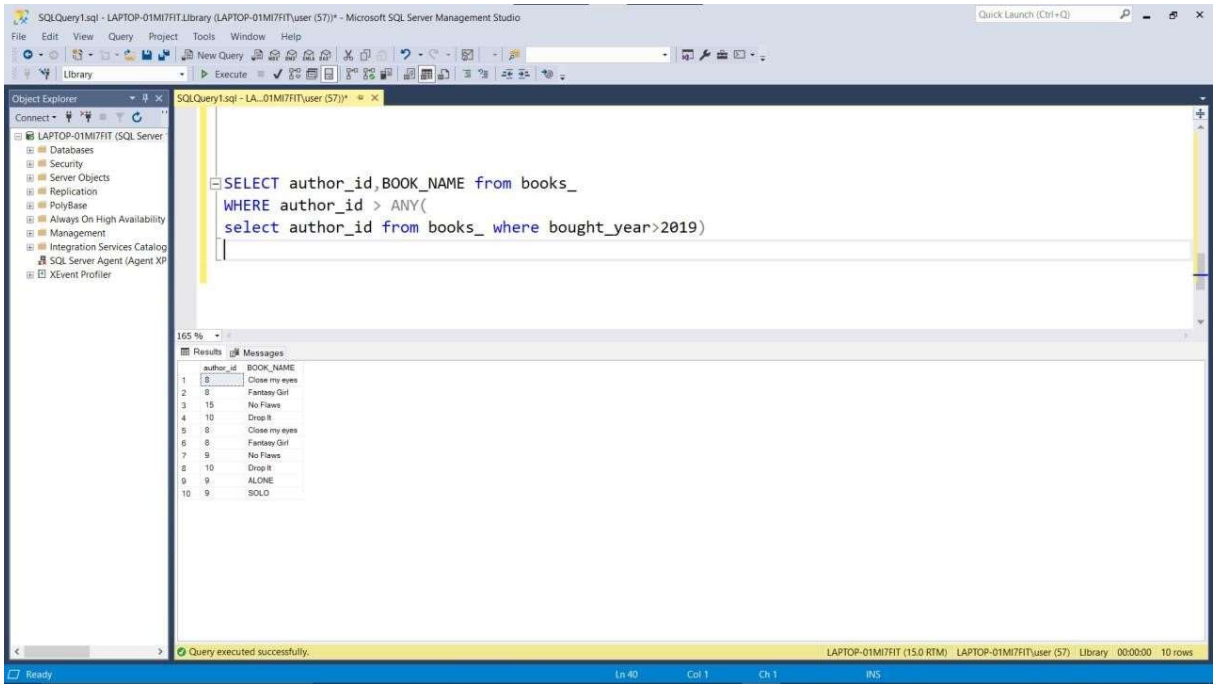

SQLQuery1.sql - LAPTOP-01MI7FIT.Library (LAPTOP-01MI7FIT\user (57))* - Microsoft SQL Server Management Studio
SELECT no_of_issues from books_
where no_of_issues > all(
select no_of_issues from books_ WHERE author_id>14)
SELECT author_id,author_name from
author_ WHERE author_id > all(
select author_id from books_)
Query executed successfully.
SQLQuery1.sql - LAPTOP-01MI7FIT.Library (LAPTOP-01MI7FIT\user (57))* - Microsoft SQL Server Management Studio
SELECT author_id,author_name from
author_ WHERE author_id > all(
select author_id from books_ where author_id<6)
Query executed successfully.

~

iii) ANY :-

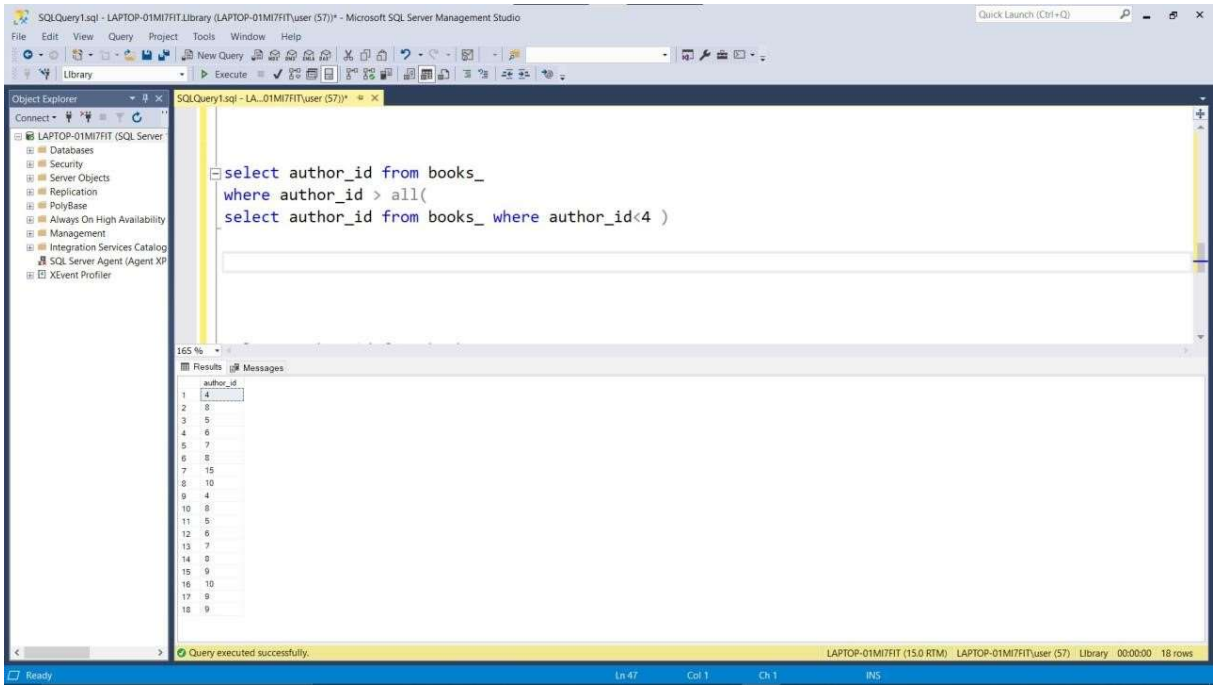


~

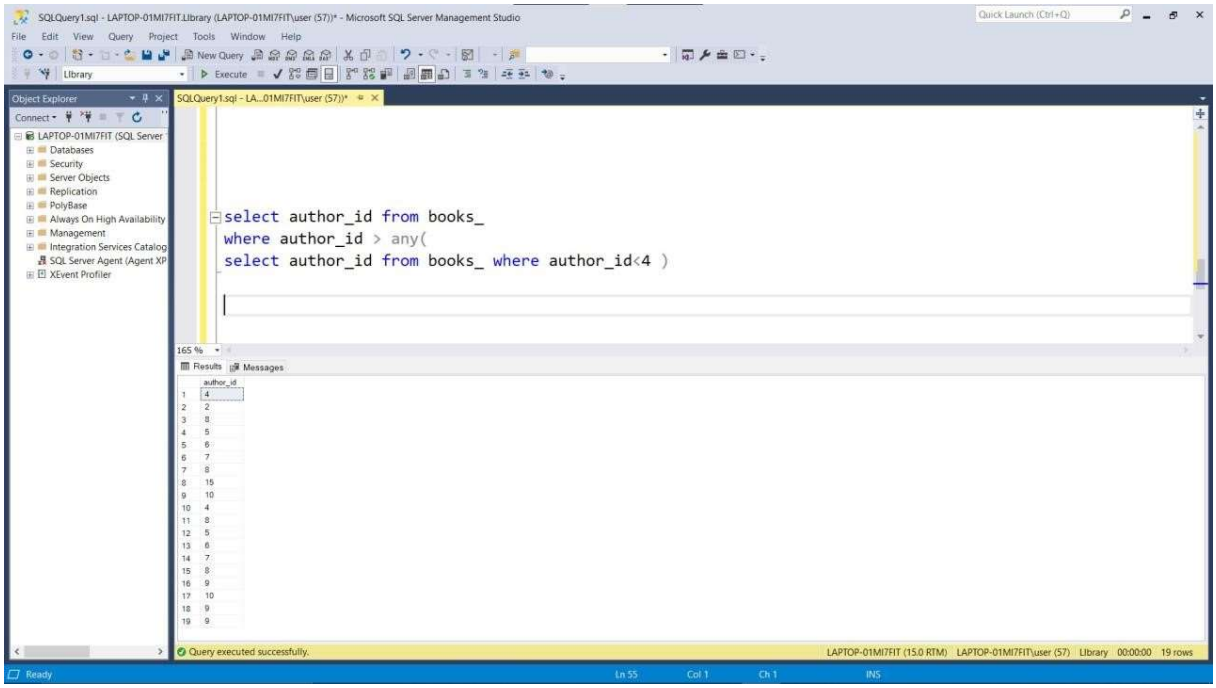


~ iv) ANY and ALL comparison

:-

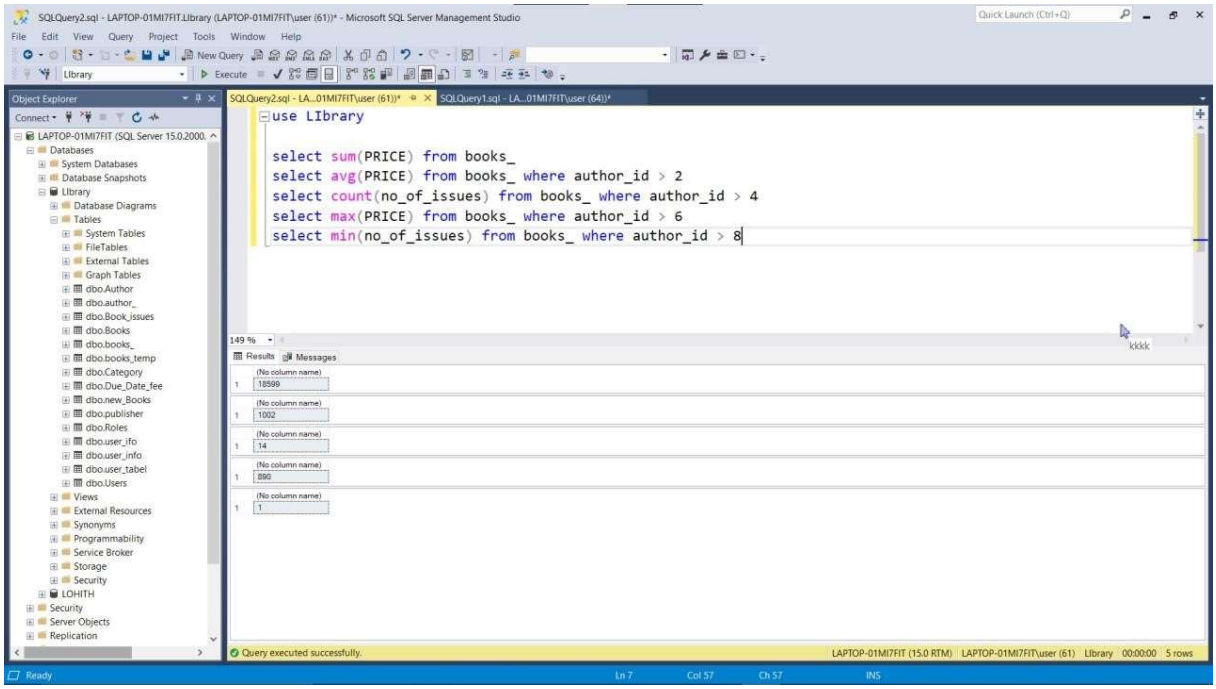


~



~

2). One query for each Aggregate function.

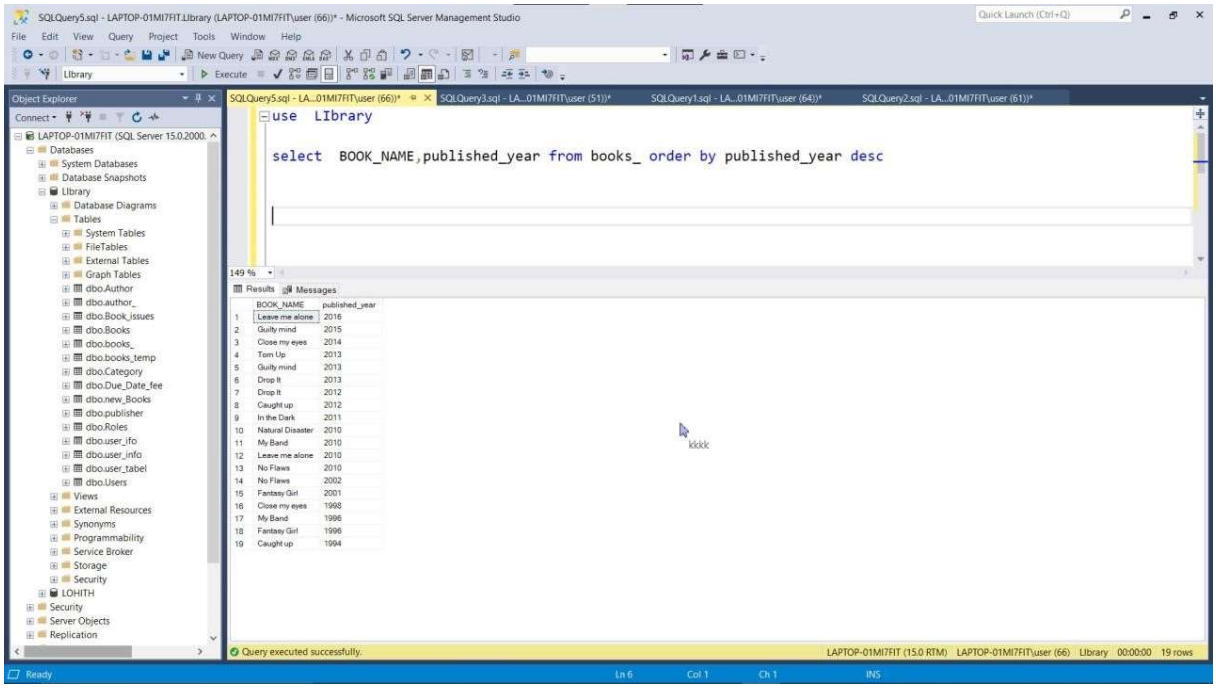


~

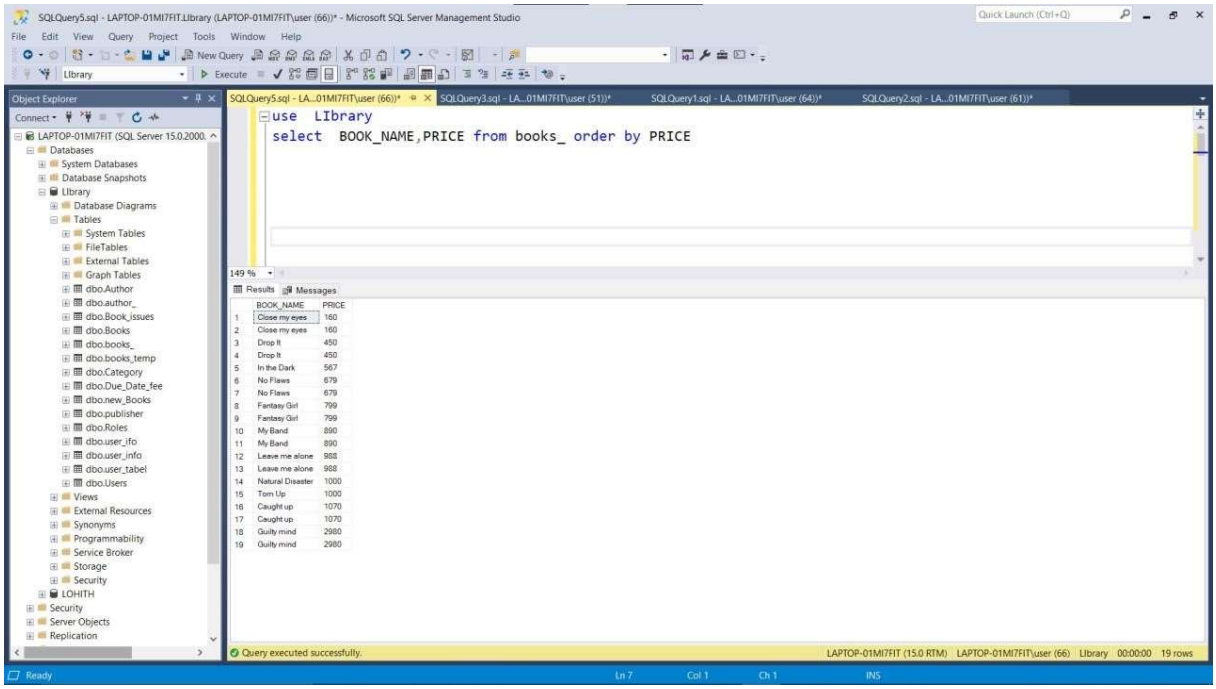
3)

Illustrate the usage of order by, group by and having clause (2 queries for each case)

i)ORDER BY :-

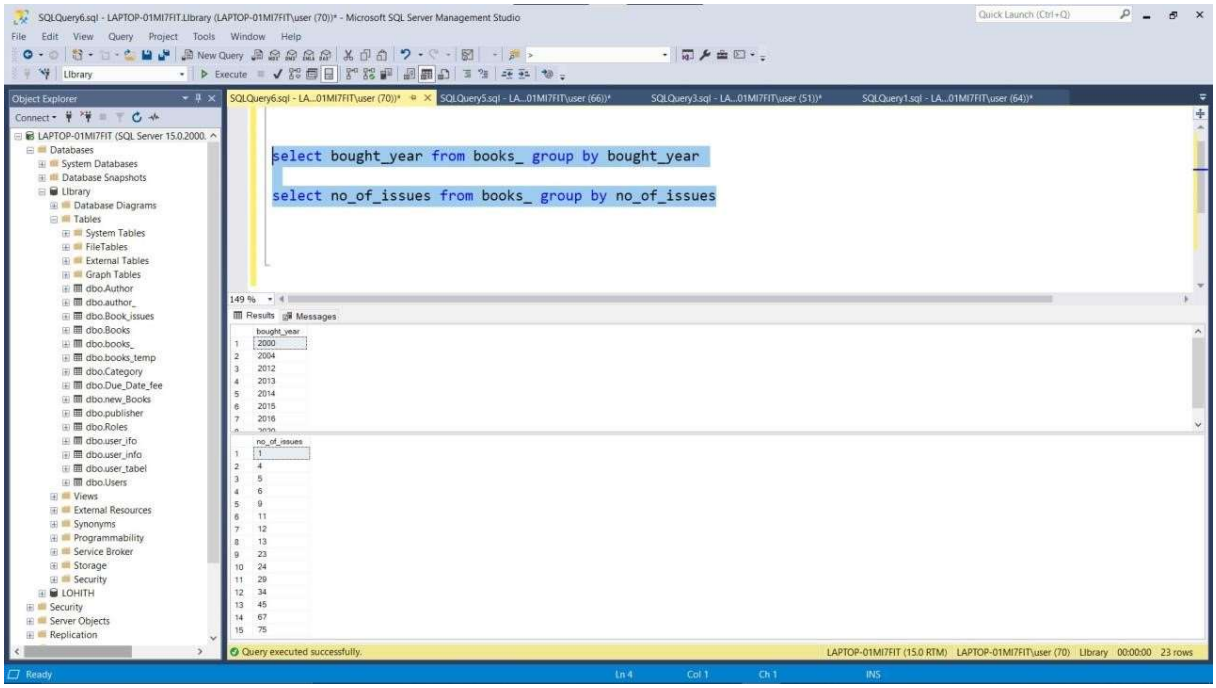


~



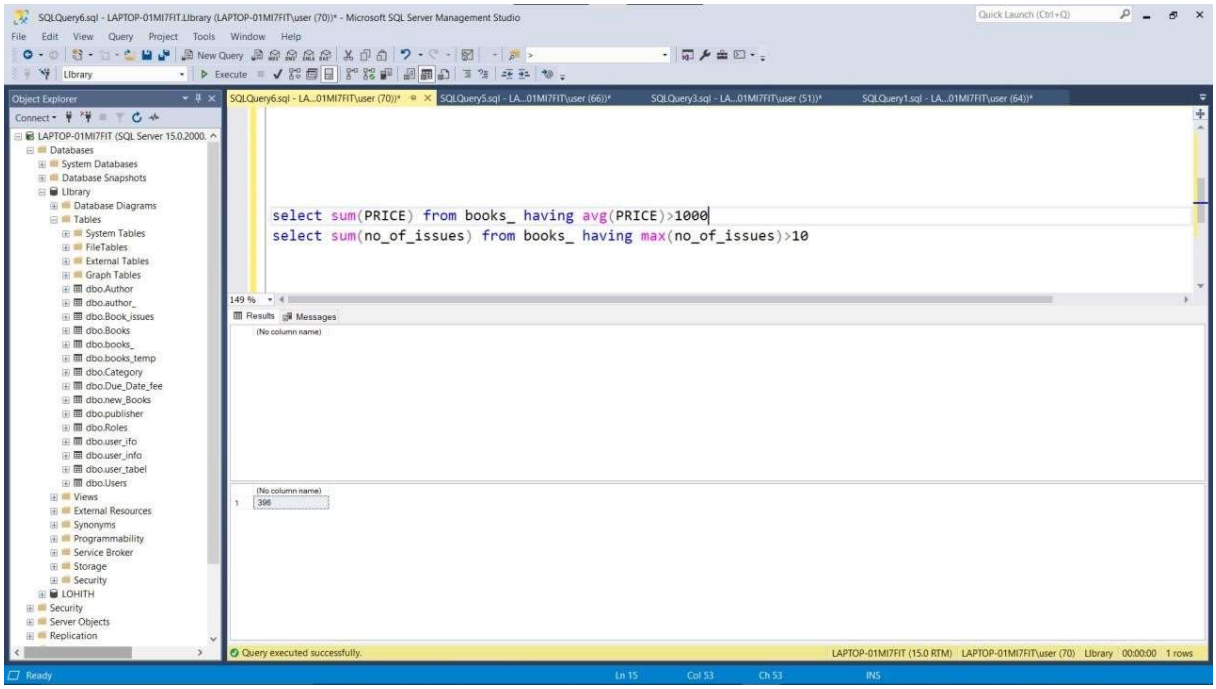
~

ii)GROUP BY :-



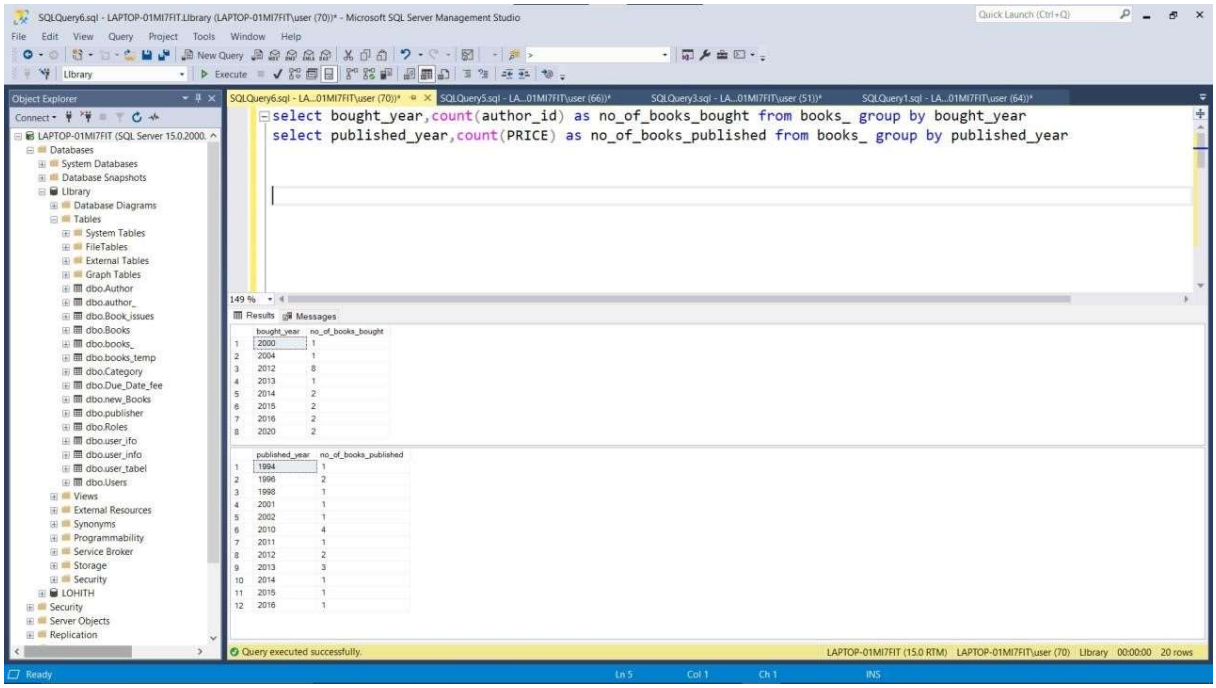
~

iii) HAVING :-

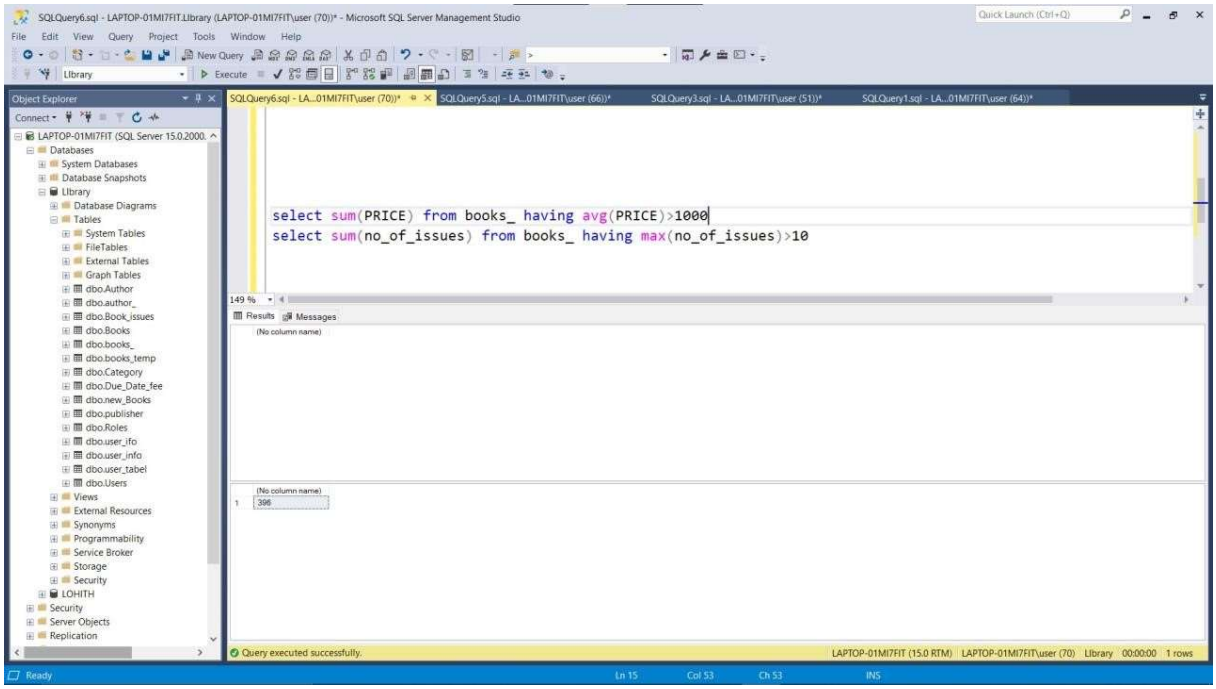


~

4)Use Aggregate function with group by and having

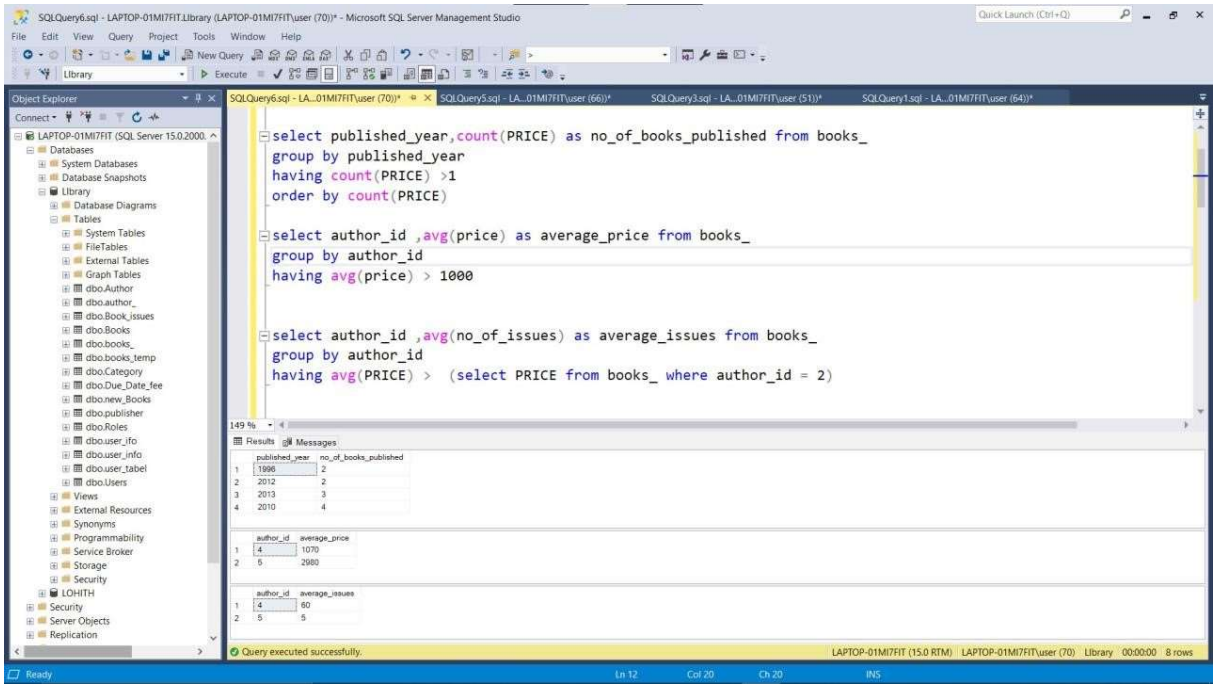


~



~

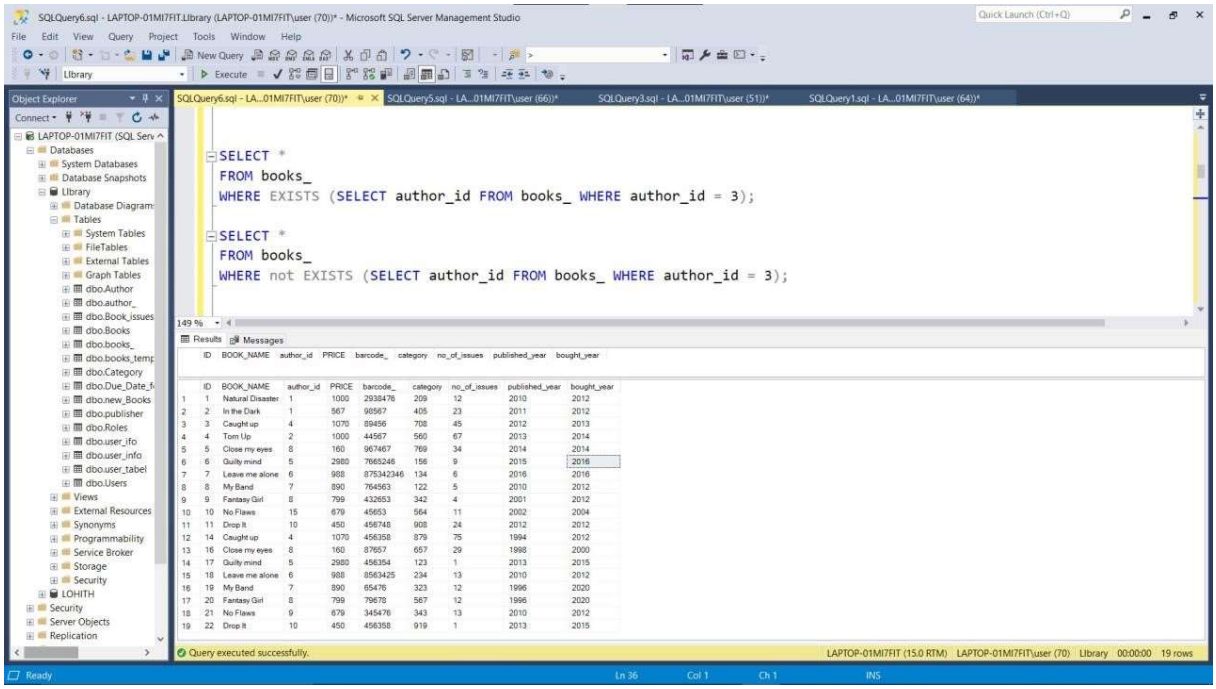
5) Write at least 3 nested queries using order by, group by and having clause.



~

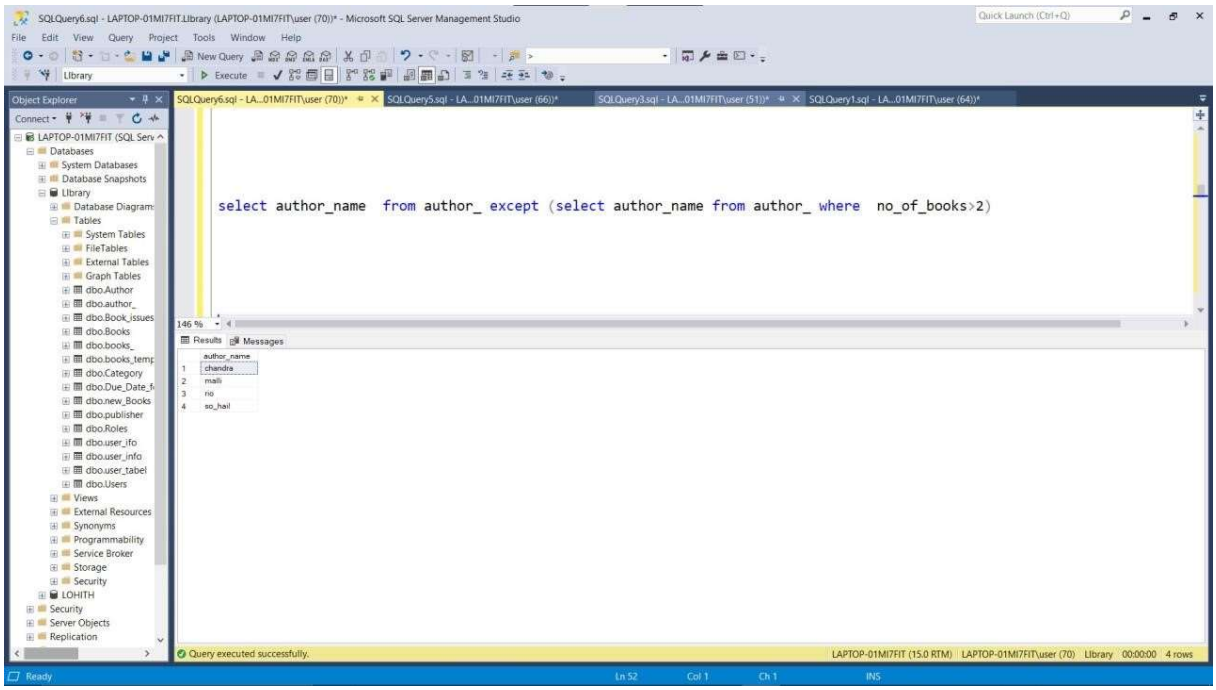
6)Illustrate the Usage of Except, Exists, Not Exists, Union, Intersection

i) EXISTS and NOT EXISTS :-



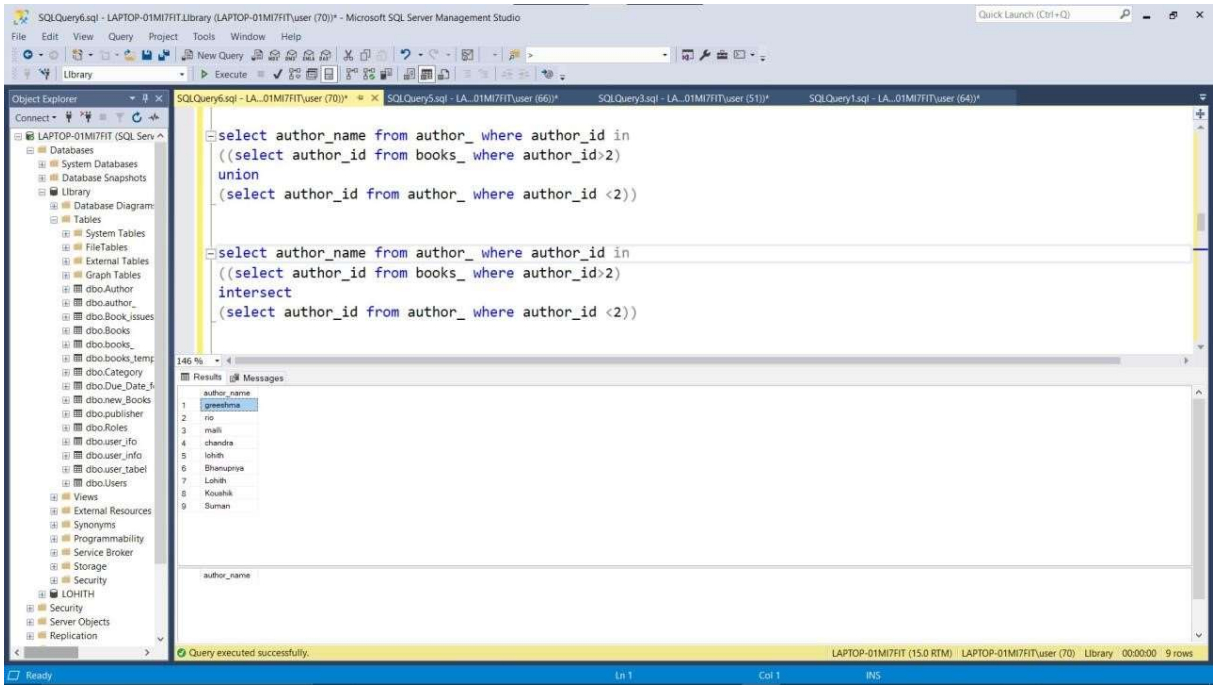
~ ii) EXCEPT

:-



~ iii) UNION and INTERSECT

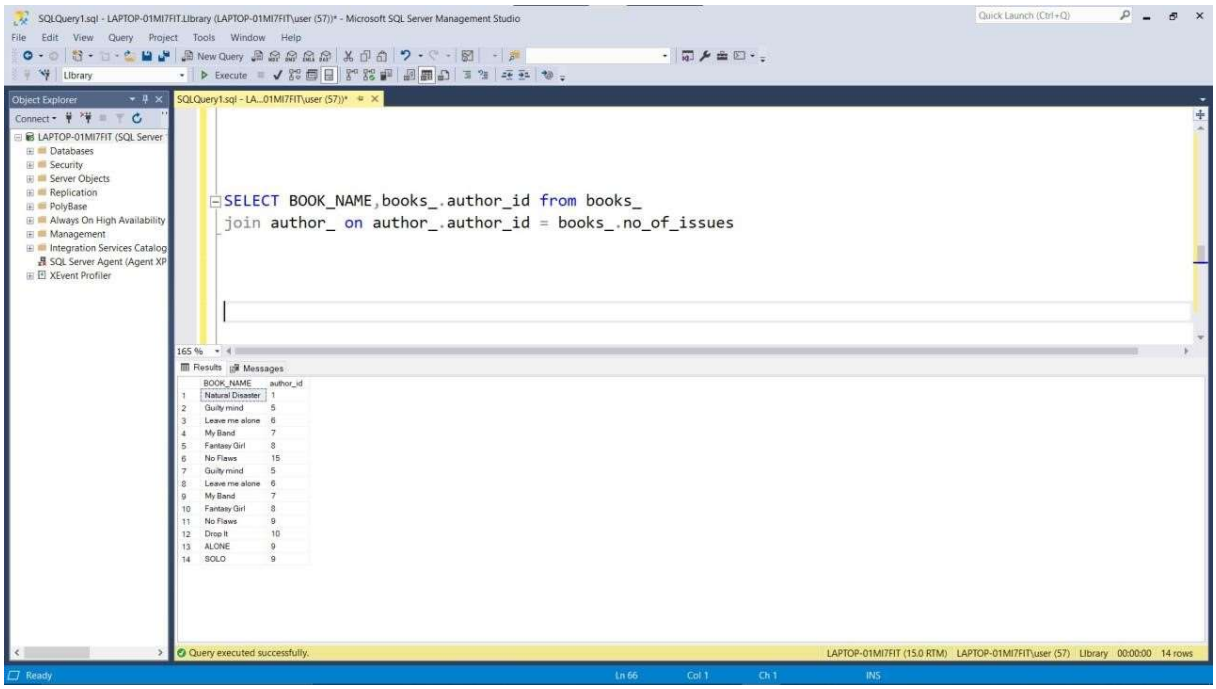
:-



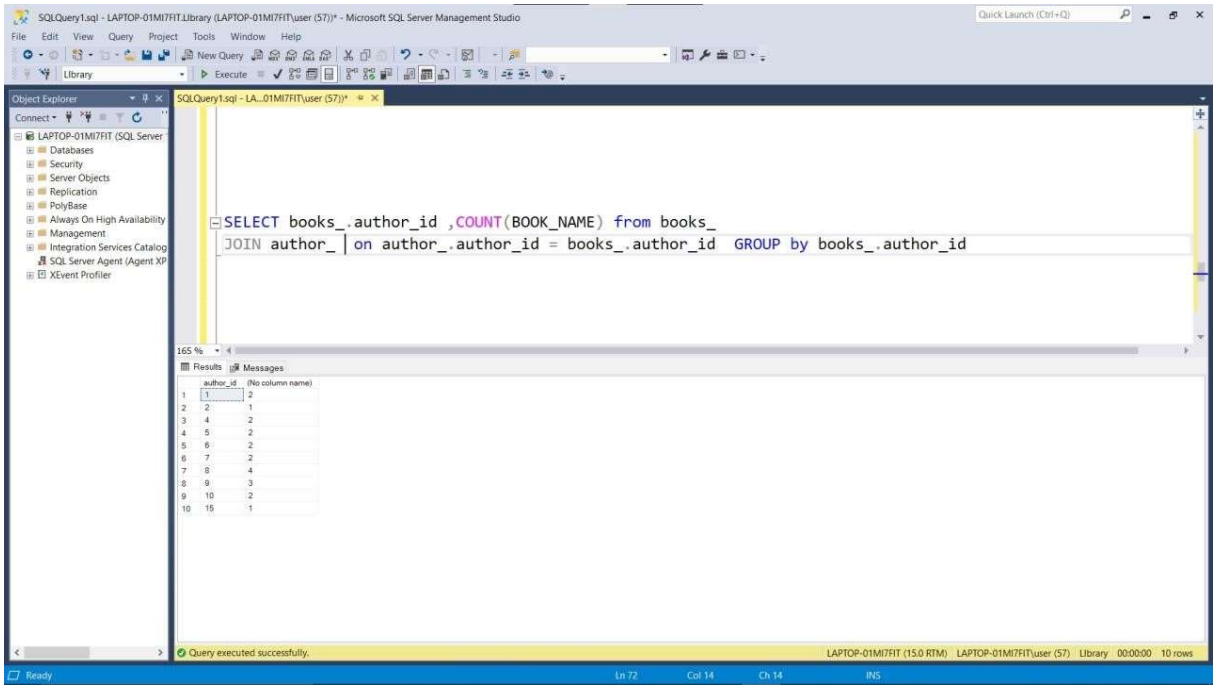
~

7) INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN- 3 queries for each instance I)

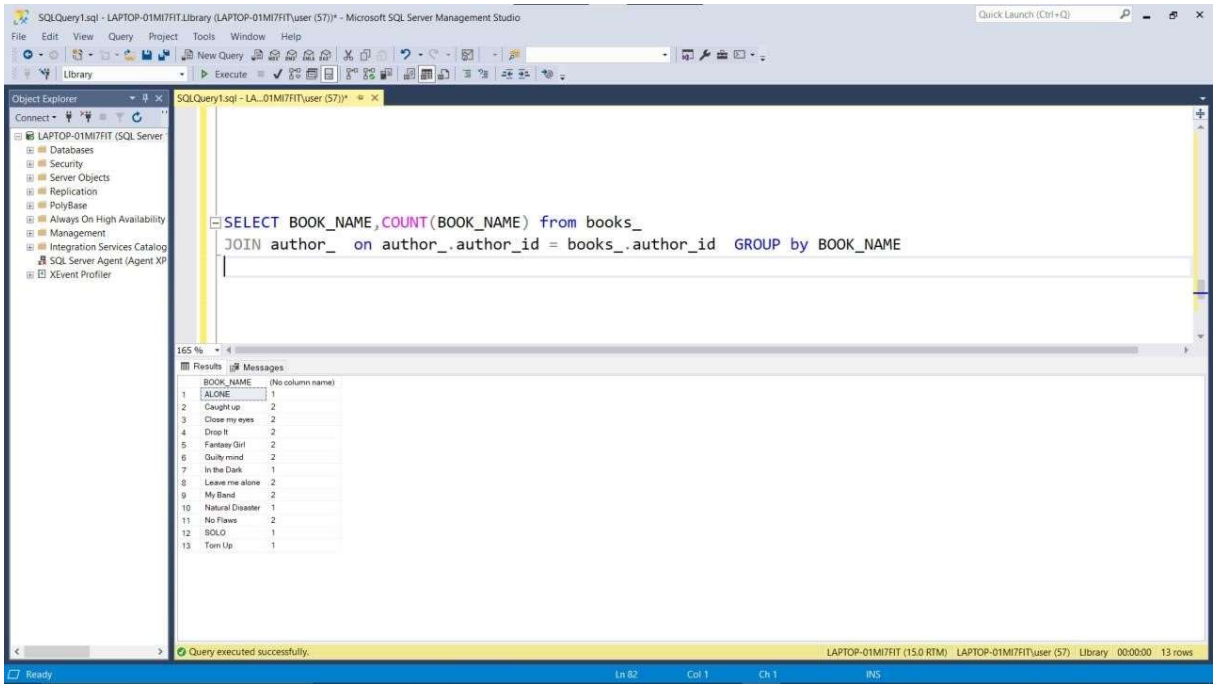
JOIN :-



~

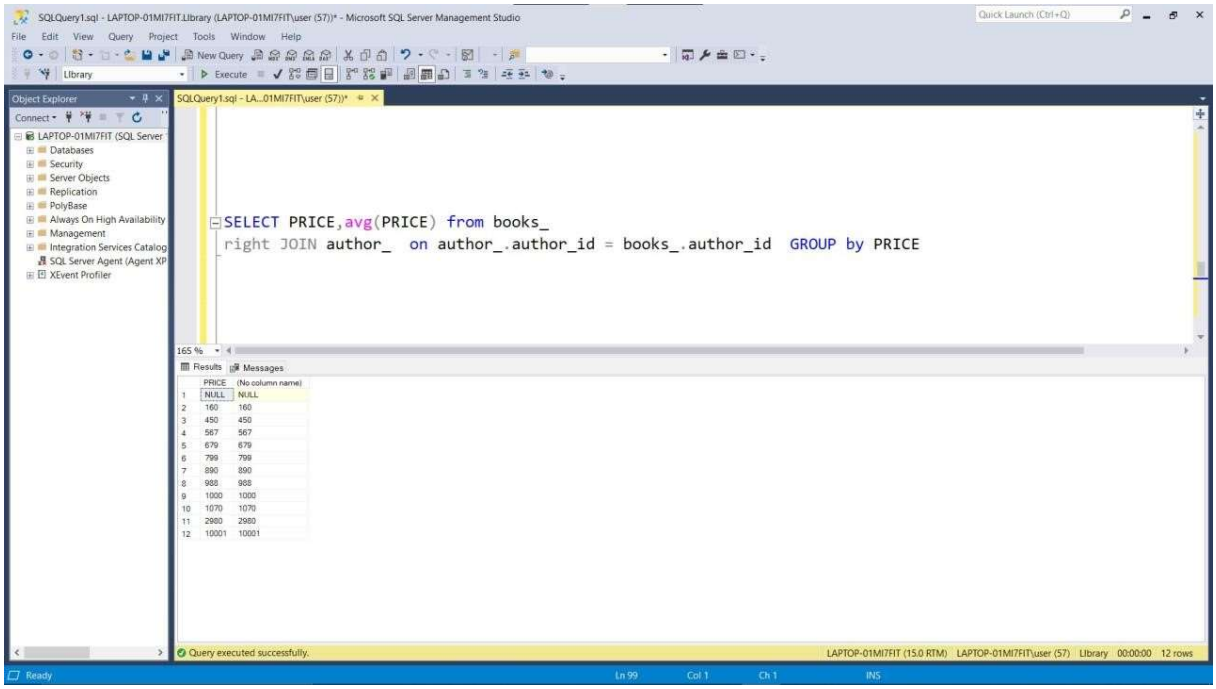


~

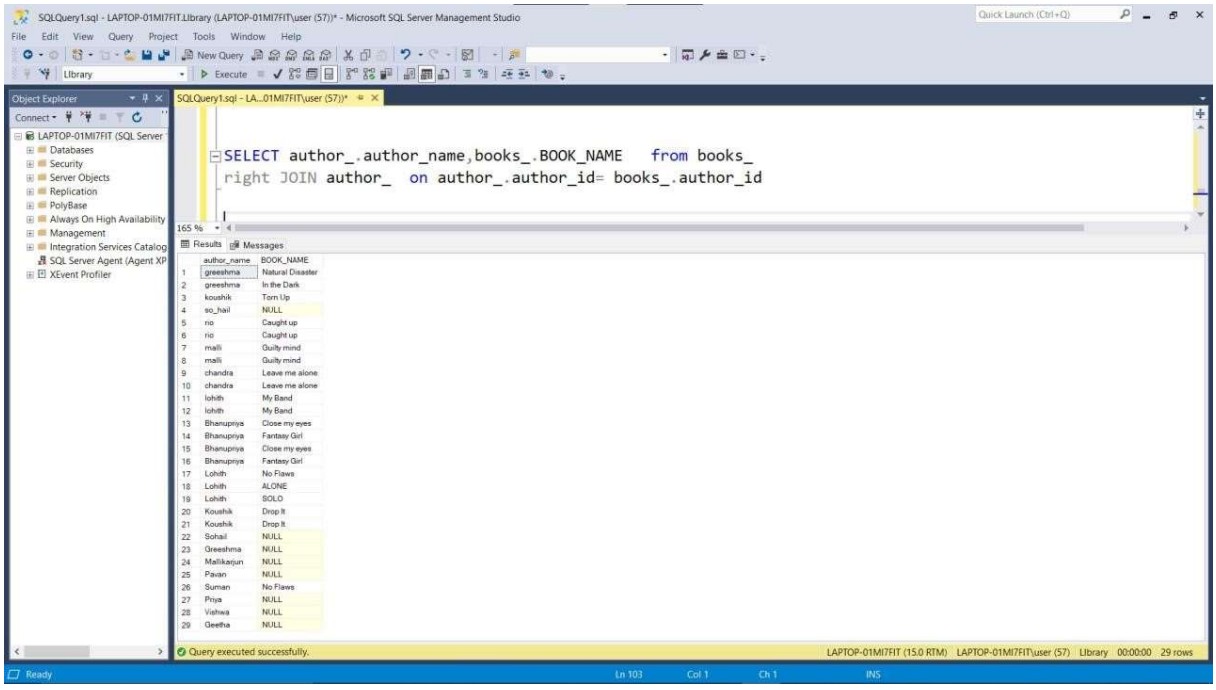


~

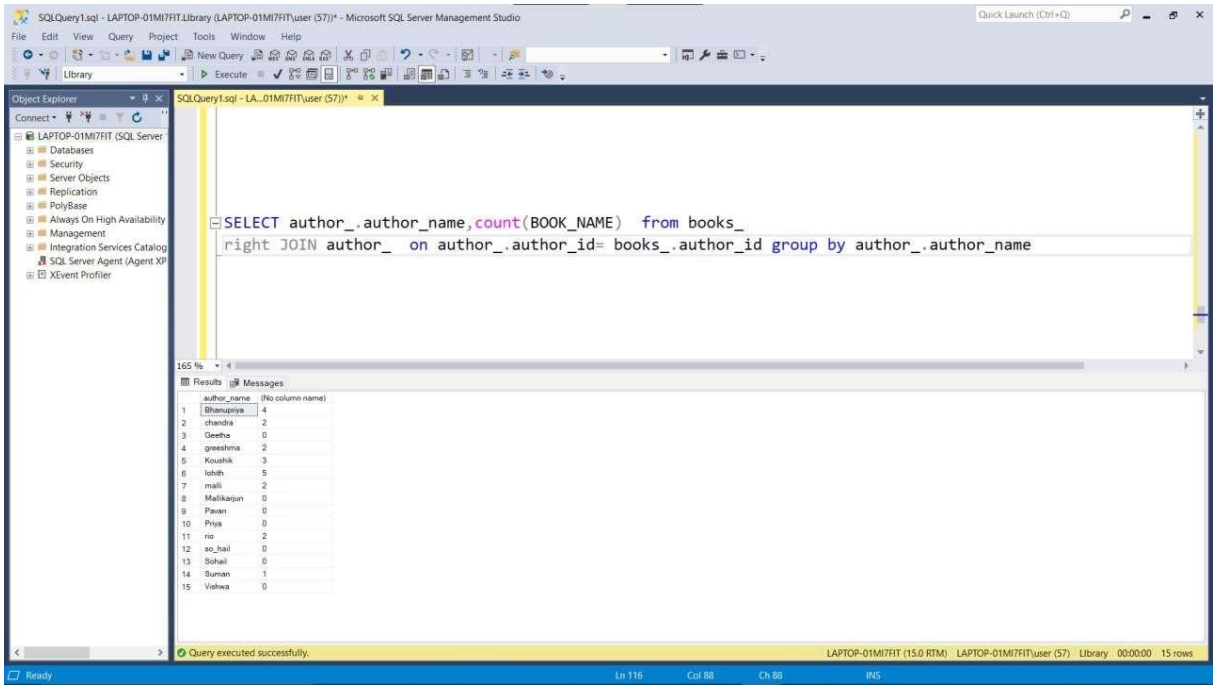
ii) RIGHT JOIN :-


SQLQuery1.sql - LAPTOP-01MI7FIT.Library (LAPTOP-01MI7FIT\user (57))* - Microsoft SQL Server Management Studio
SELECT PRICE,avg(PRICE) from books_
right JOIN author_ on author_.author_id = books_.author_id GROUP by PRICE
Query executed successfully.

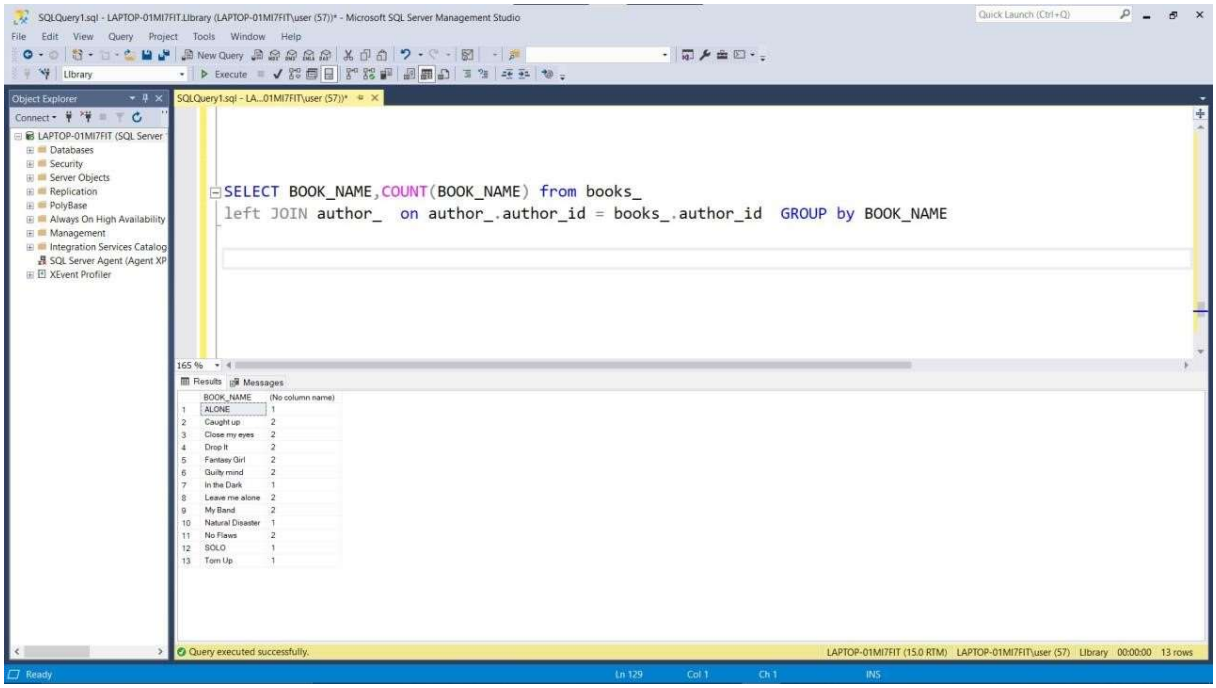
~


SQLQuery1.sql - LAPTOP-01MI7FIT.Library (LAPTOP-01MI7FIT\user (57))* - Microsoft SQL Server Management Studio
SELECT author_.author_name,books_.BOOK_NAME from books_
right JOIN author_ on author_.author_id= books_.author_id
Query executed successfully.

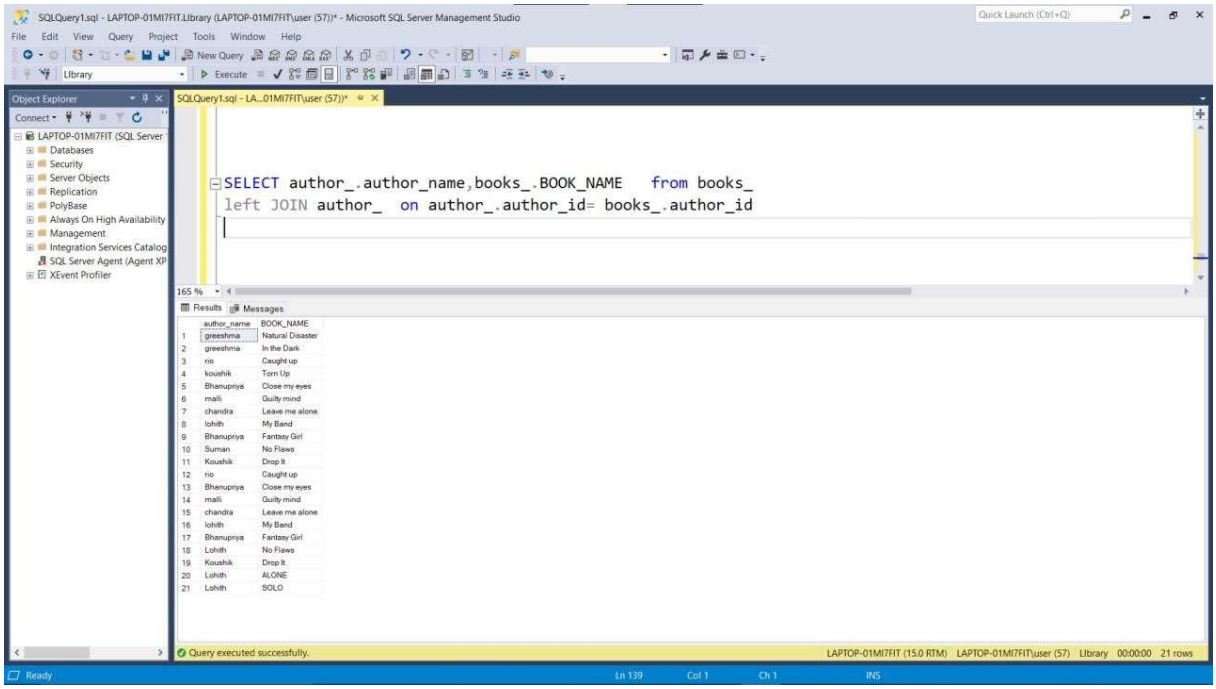
~



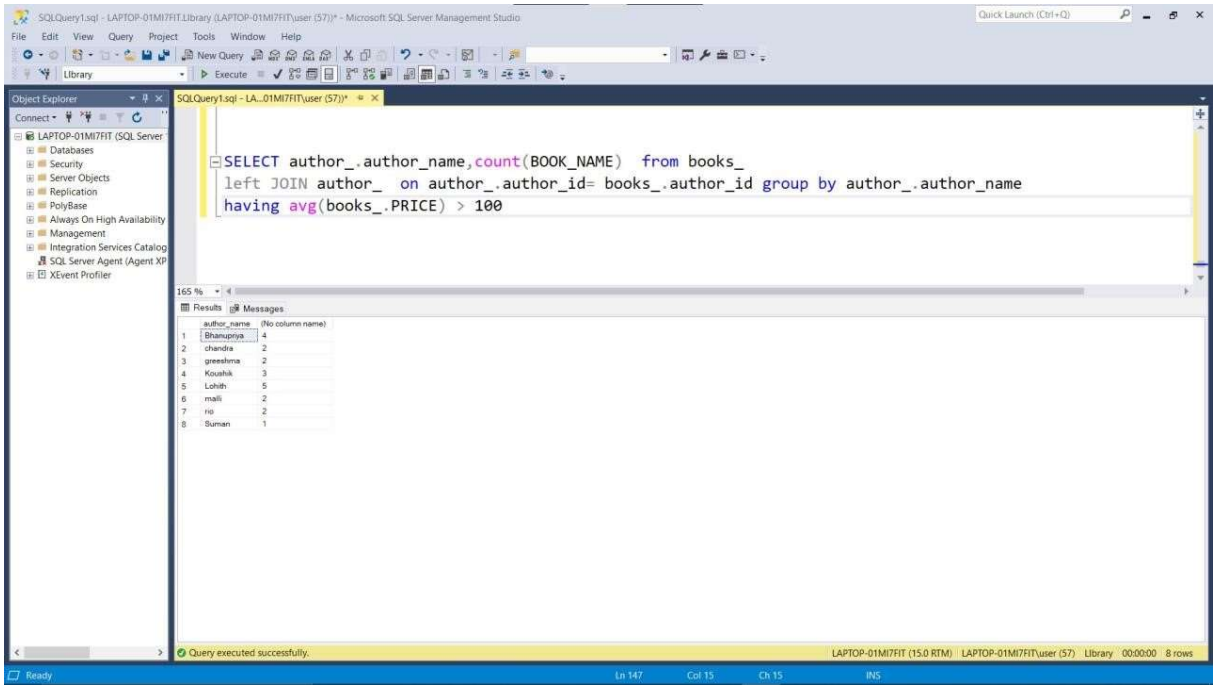
~ iii) LEFT JOIN :-



~


SQLQuery1.sql - LAPTOP-01MI7FIT.Library (LAPTOP-01MI7FIT\user (57))* - Microsoft SQL Server Management Studio
File Edit View Query Project Tools Window Help
New Query
Execute
Library
Object Explorer
Connect
LAPTOP-01MI7FIT (SQL Server
Databases
Security
Server Objects
Replication
PolyBase
Always On High Availability
Management
Integration Services Catalog
SQL Server Agent (Agent XP
XEvent Profiler
SQLQuery1.sql - LA...01MI7FIT\user (57))*
SELECT author_.author_name,books_.BOOK_NAME from books_
left JOIN author_ on author_.author_id= books_.author_id
165 %
Results
Messages
author_name BOOK_NAME
1 greeshma Natural Disaster
2 greeshma In the Dark
3 rio Caught up
4 koushik Torn Up
5 Bhanupriya Close my eyes
6 malli Guilty mind
7 chandra Leave me alone
8 lohith My Band
9 Bhanupriya Fantasy Girl
10 Suman No Flaws
11 Koushik Drop It
12 rio Caught up
13 Bhanupriya Close my eyes
14 malli Guilty mind
15 chandra Leave me alone
16 lohith My Band
17 Bhanupriya Fantasy Girl
18 Lohith No Flaws
19 Koushik Drop It
20 Lohith ALONE
21 Lohith SOLO
Query executed successfully.
LAPTOP-01MI7FIT (15.0 RTM) LAPTOP-01MI7FIT\user (57) Library 00:00:00 21 rows
Ready
Ln 139 Col 1 Ch 1 INS

~


SQLQuery1.sql - LAPTOP-01MI7FIT.Library (LAPTOP-01MI7FIT\user (57))* - Microsoft SQL Server Management Studio
File Edit View Query Project Tools Window Help
New Query
Execute
Library
Object Explorer
Connect
LAPTOP-01MI7FIT (SQL Server
Databases
Security
Server Objects
Replication
PolyBase
Always On High Availability
Management
Integration Services Catalog
SQL Server Agent (Agent XP
XEvent Profiler
SQLQuery1.sql - LA...01MI7FIT\user (57))*
SELECT author_.author_name,count(BOOK_NAME) from books_
left JOIN author_ on author_.author_id= books_.author_id group by author_.author_name
having avg(books_.PRICE) > 100
165 %
Results
Messages
author_name (No column name)
1 Bhanupriya 4
2 chandra 2
3 greeshma 2
4 Koushik 3
5 Lohith 5
6 malli 2
7 rio 2
8 Suman 1
Query executed successfully.
LAPTOP-01MI7FIT (15.0 RTM) LAPTOP-01MI7FIT\user (57) Library 00:00:00 8 rows
Ready
Ln 147 Col 15 Ch 15 INS