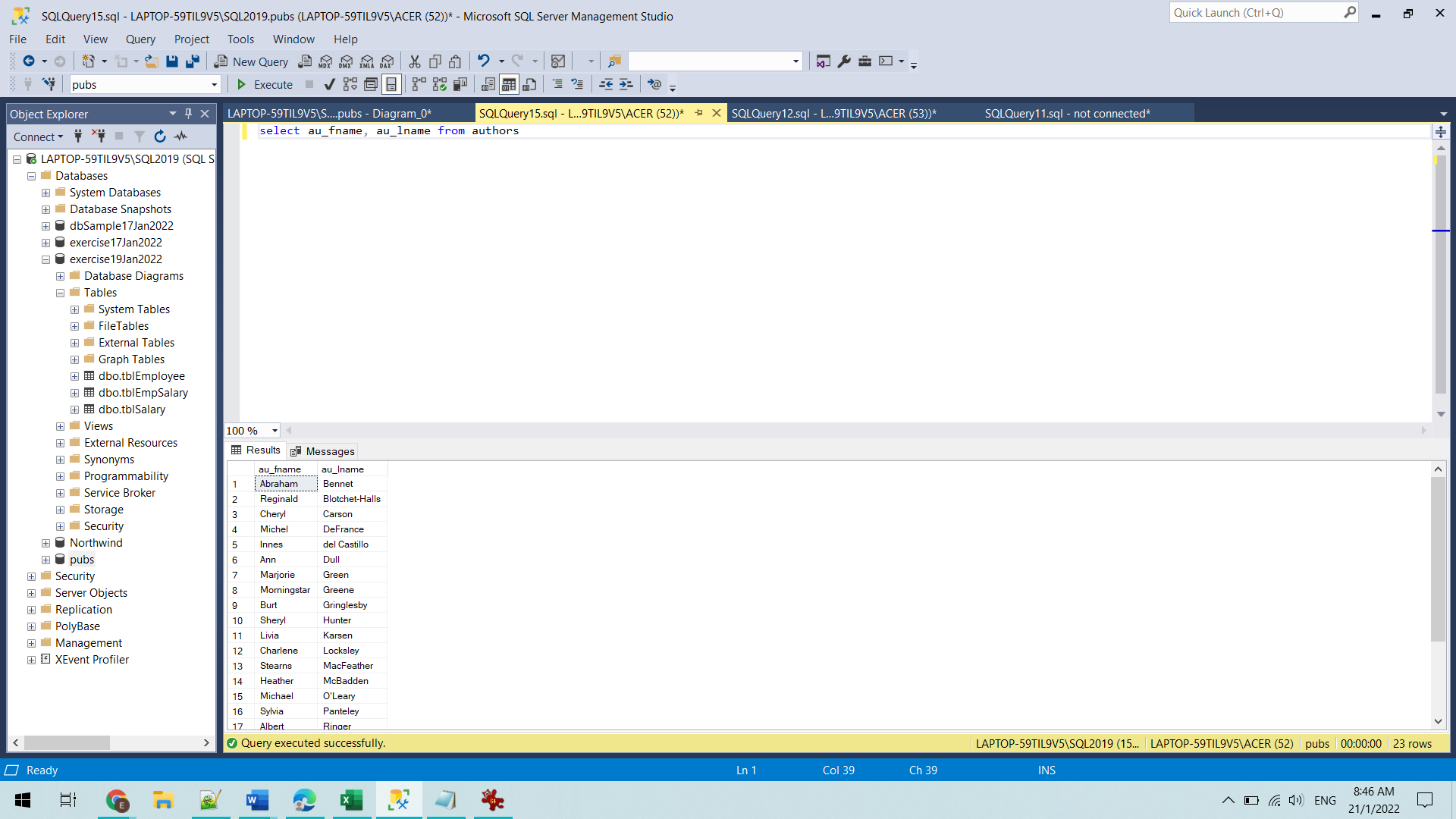
//Question16, Question17 are not completed yet.

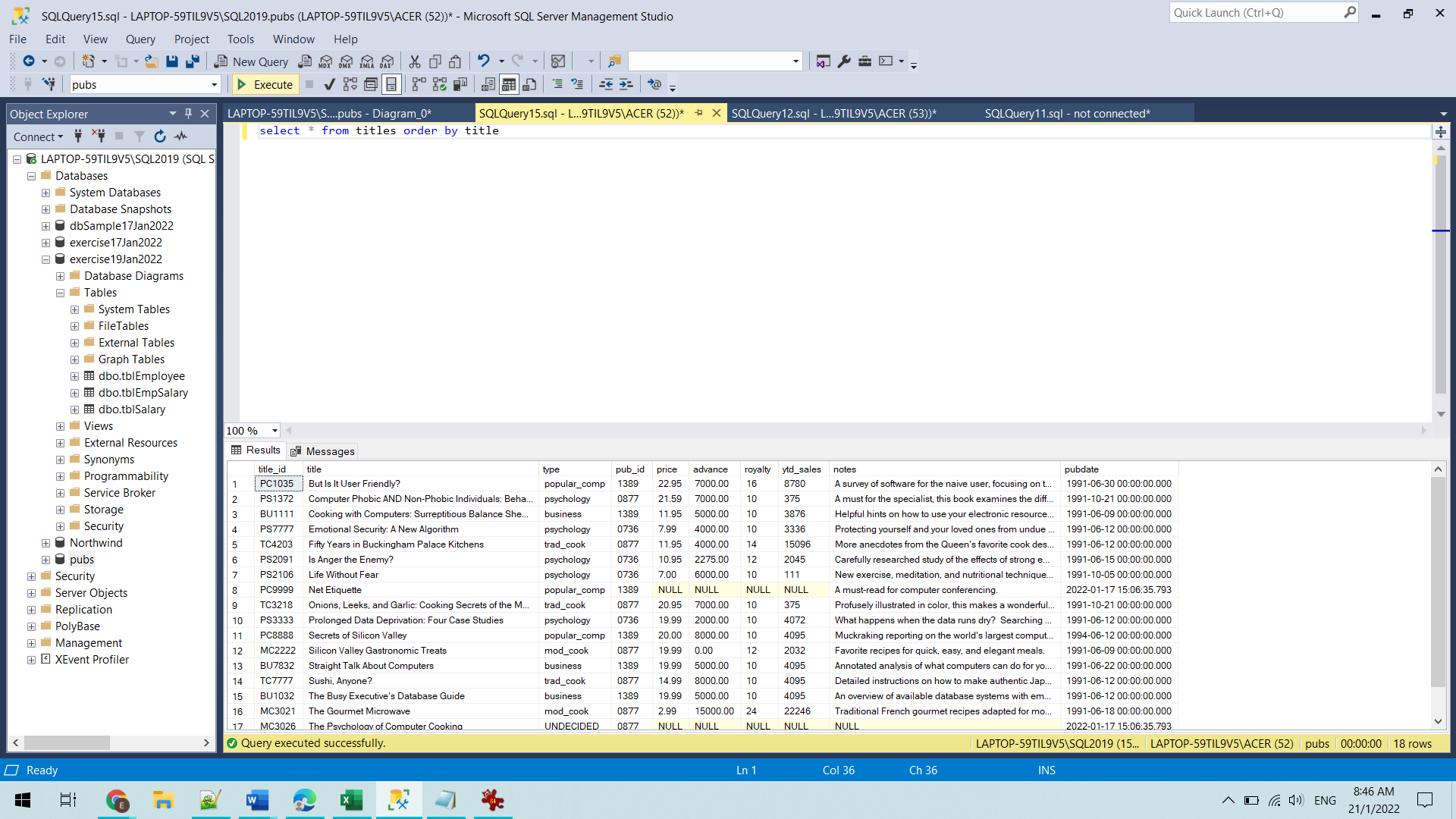
Question 1

select au\_fname, au\_lname from authors



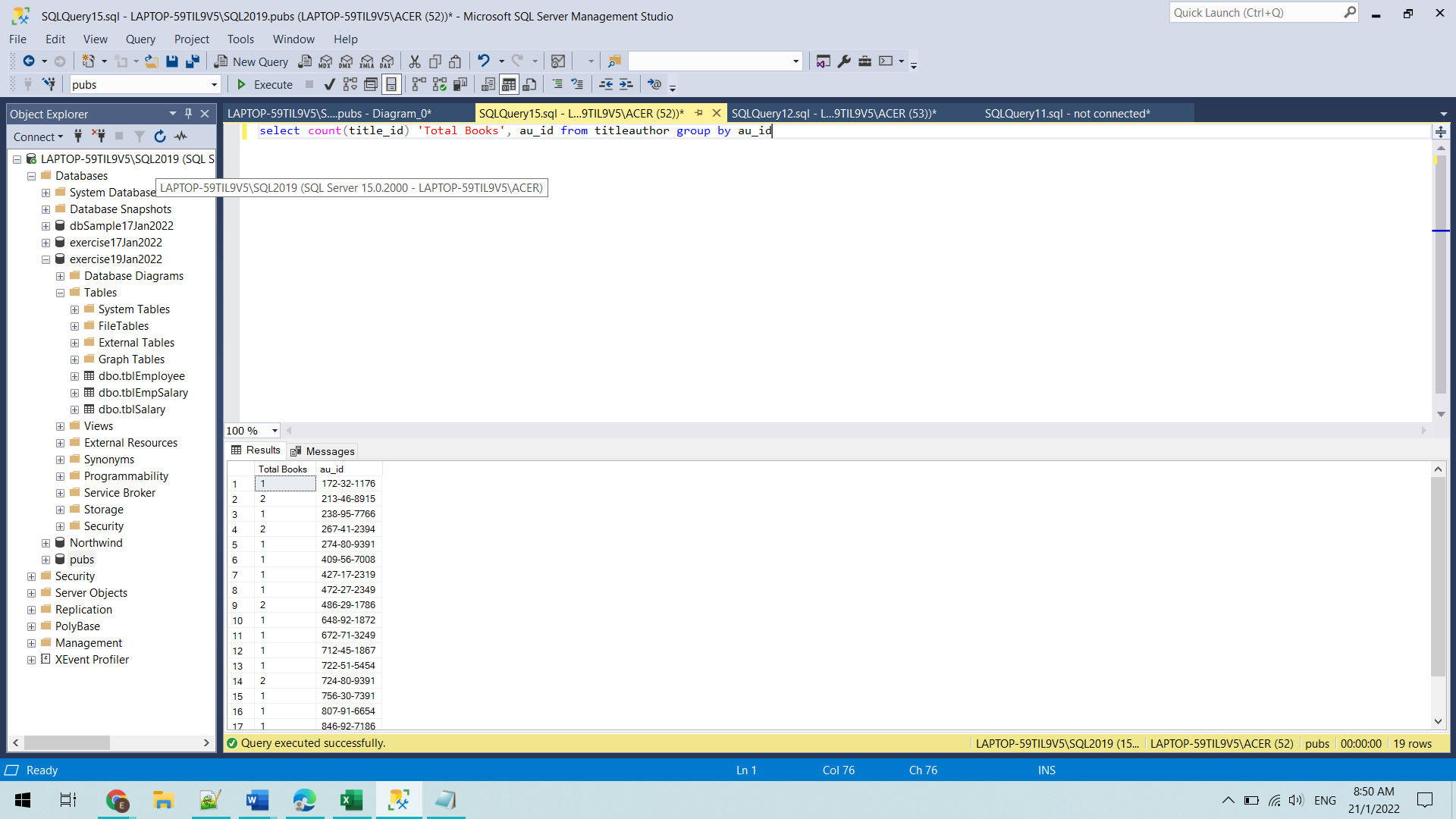
Question 2

select \* from titles order by title



Question 3

select count(title\_id) 'Total Books', au\_id from titleauthor group by au\_id



Question 4

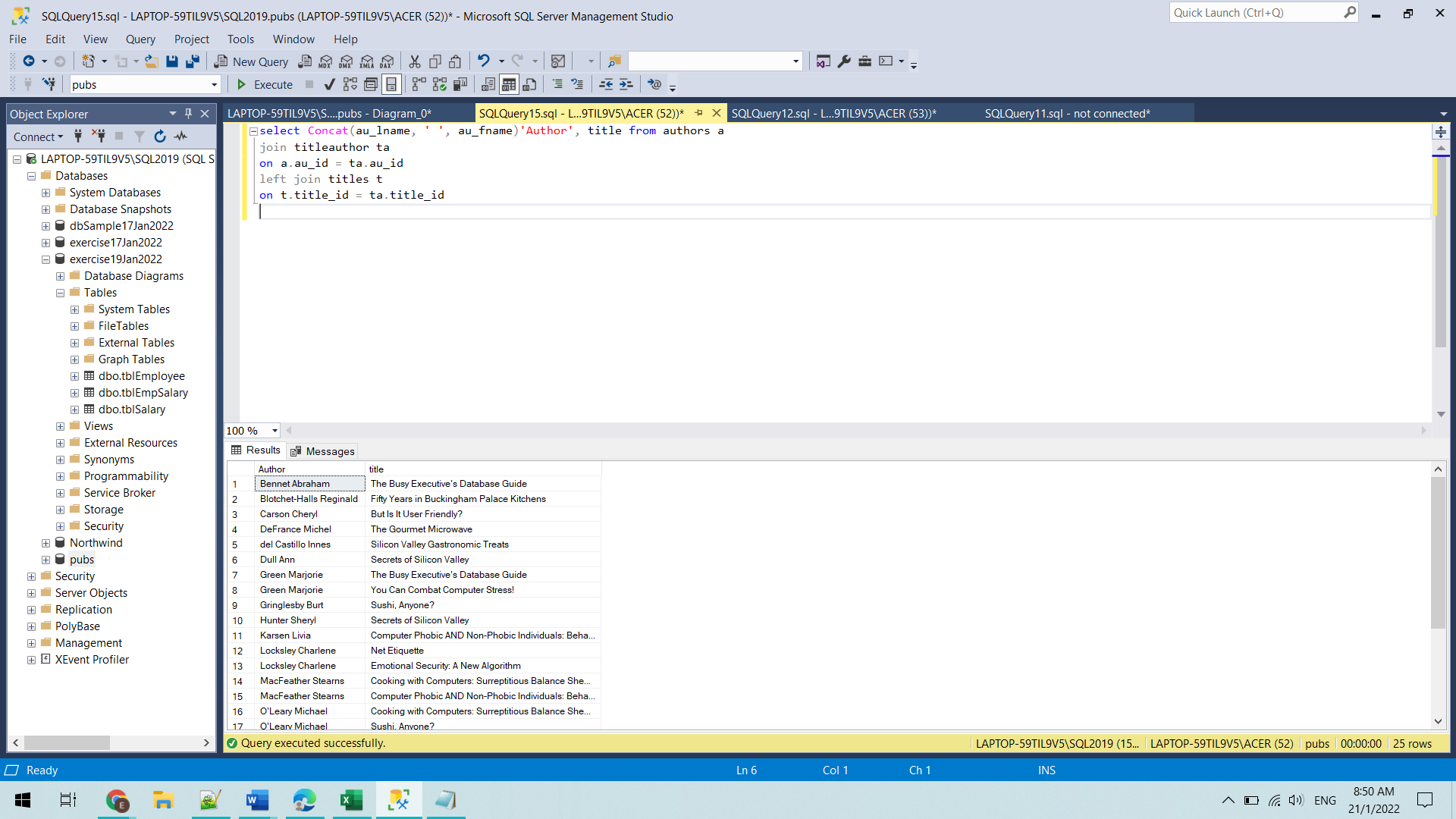
select Concat(au\_lname, ' ', au\_fname)'Author', title from authors a

join titleauthor ta

on a.au\_id = ta.au\_id

left join titles t

on t.title\_id = ta.title\_id



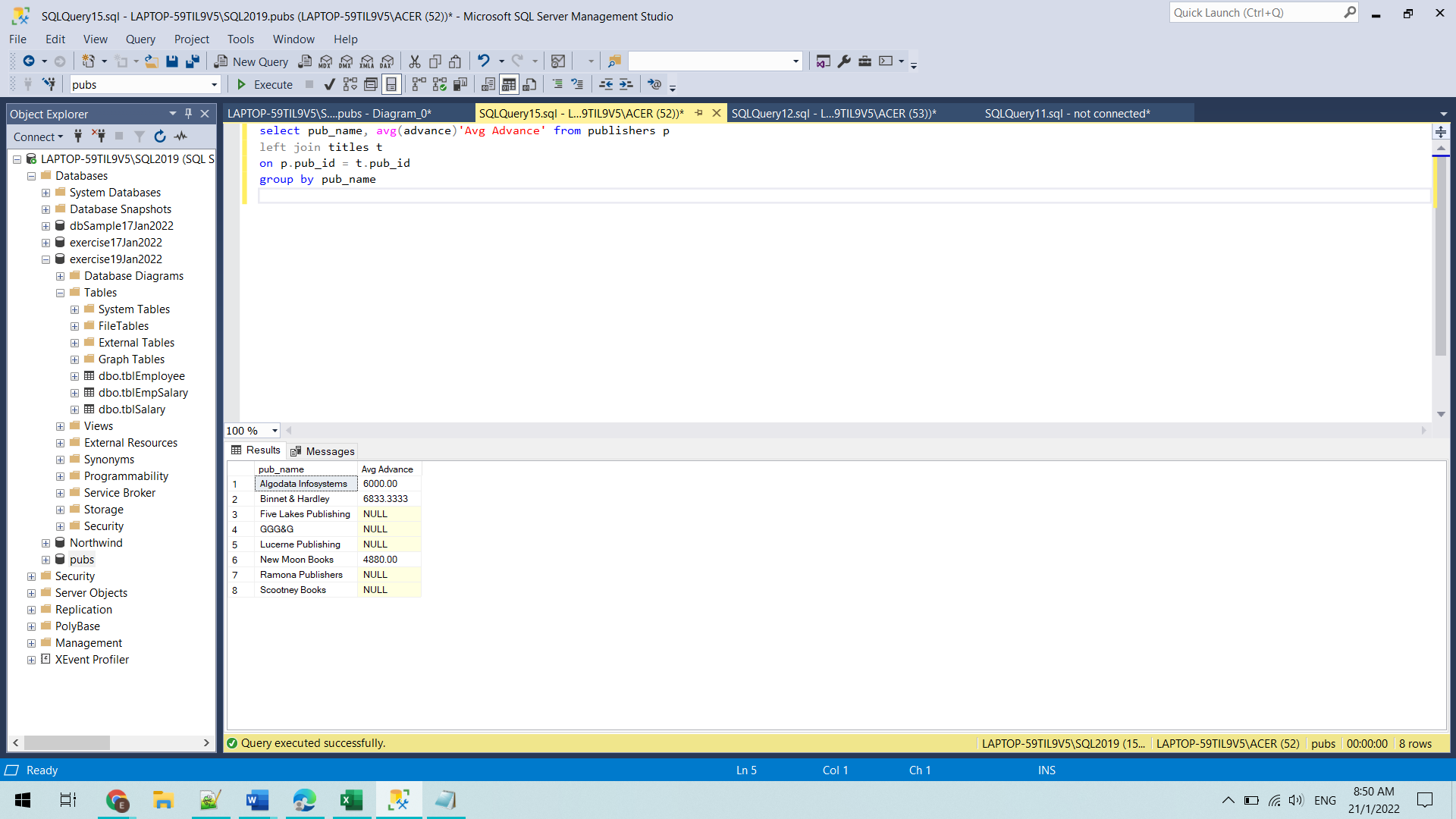
Question 5

select pub\_name, avg(advance)'Avg Advance' from publishers p

left join titles t

on p.pub\_id = t.pub\_id

group by pub\_name



Question 6

select pub\_name, Concat(au\_lname, ' ', au\_fname)'Author', qty \* price 'Sale Amount',

title from titles t

right join publishers p

on t.pub\_id = p.pub\_id

left join titleauthor ta

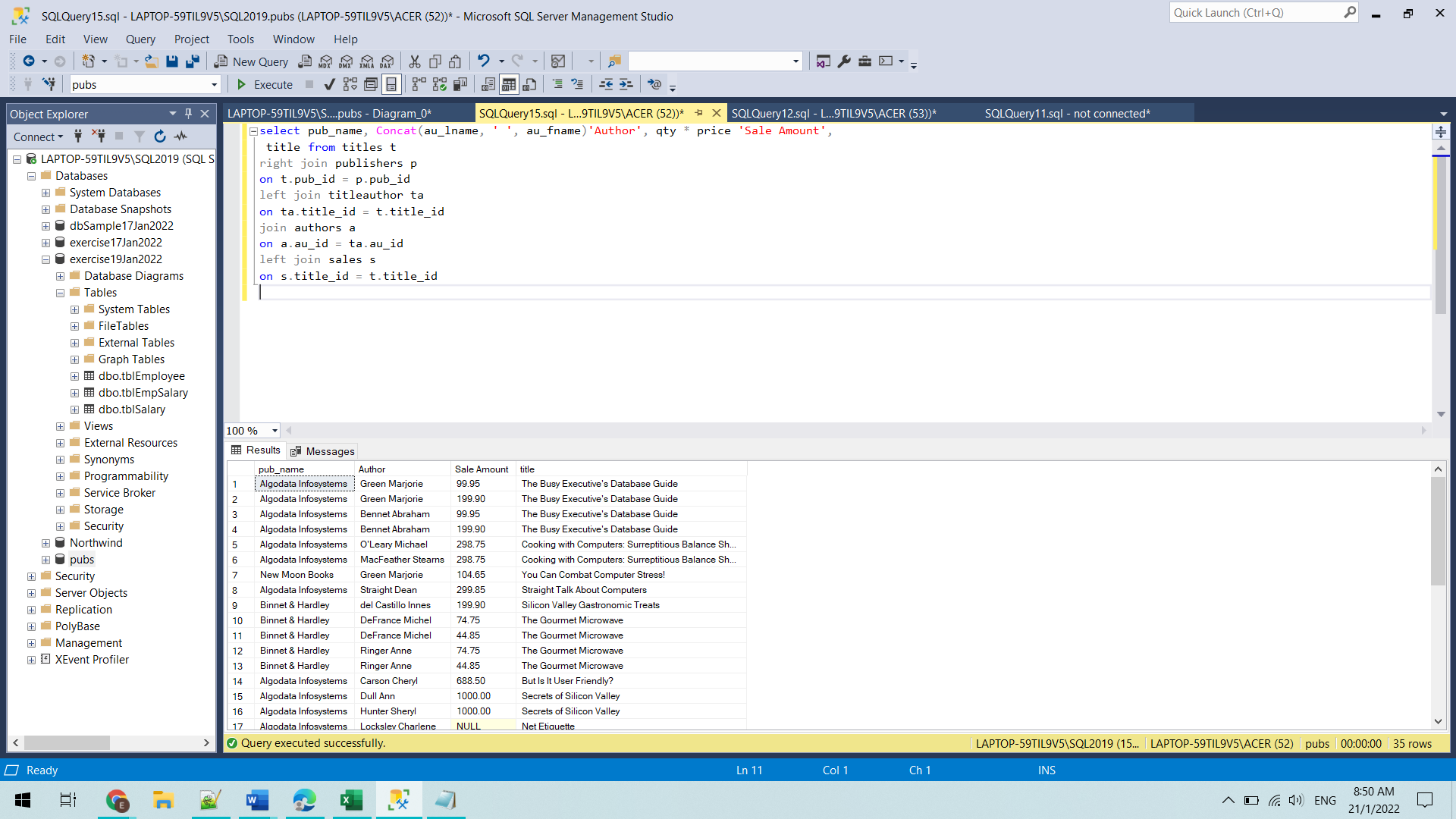
on ta.title\_id = t.title\_id

join authors a

on a.au\_id = ta.au\_id

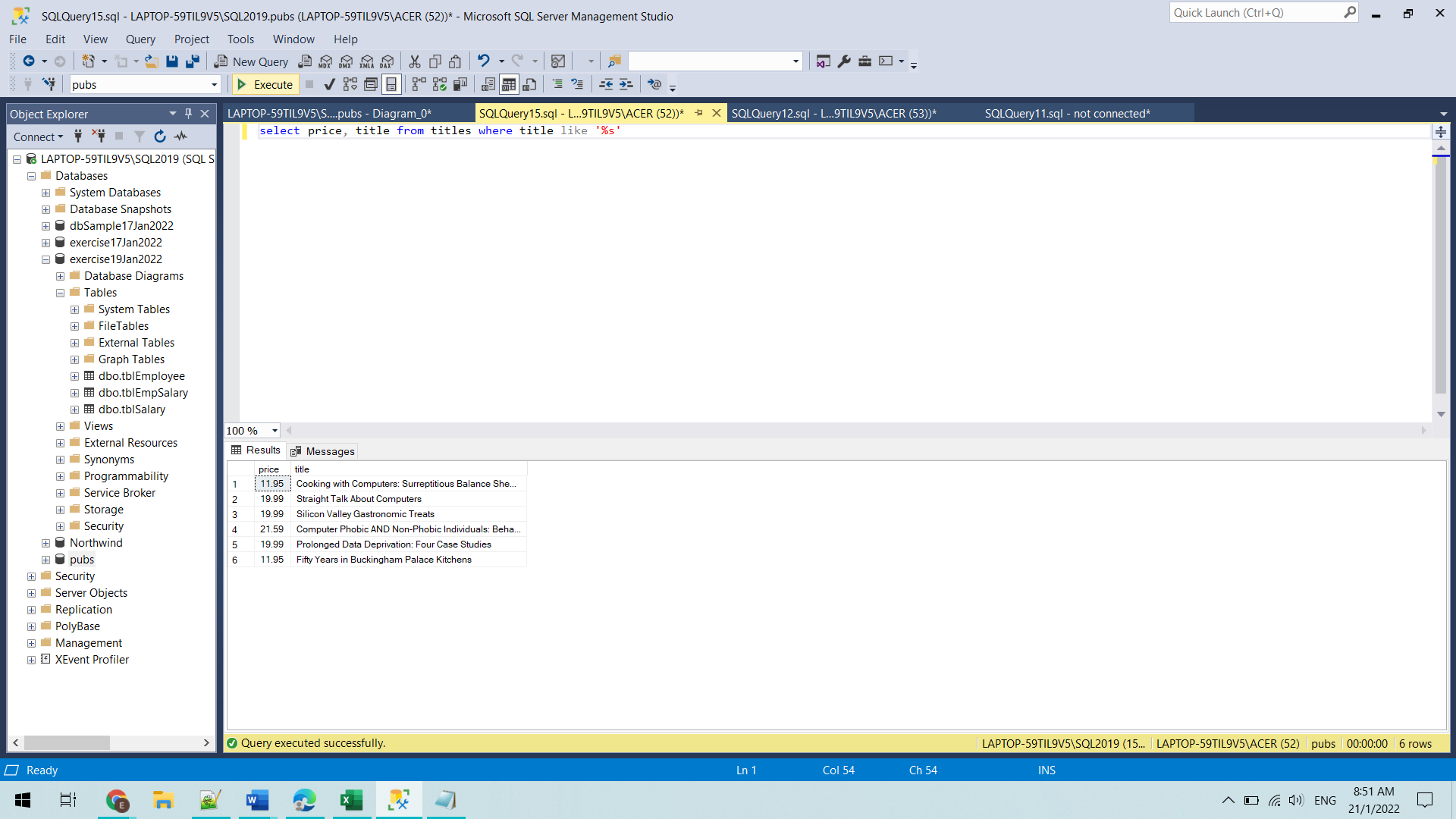
left join sales s

on s.title\_id = t.title\_id



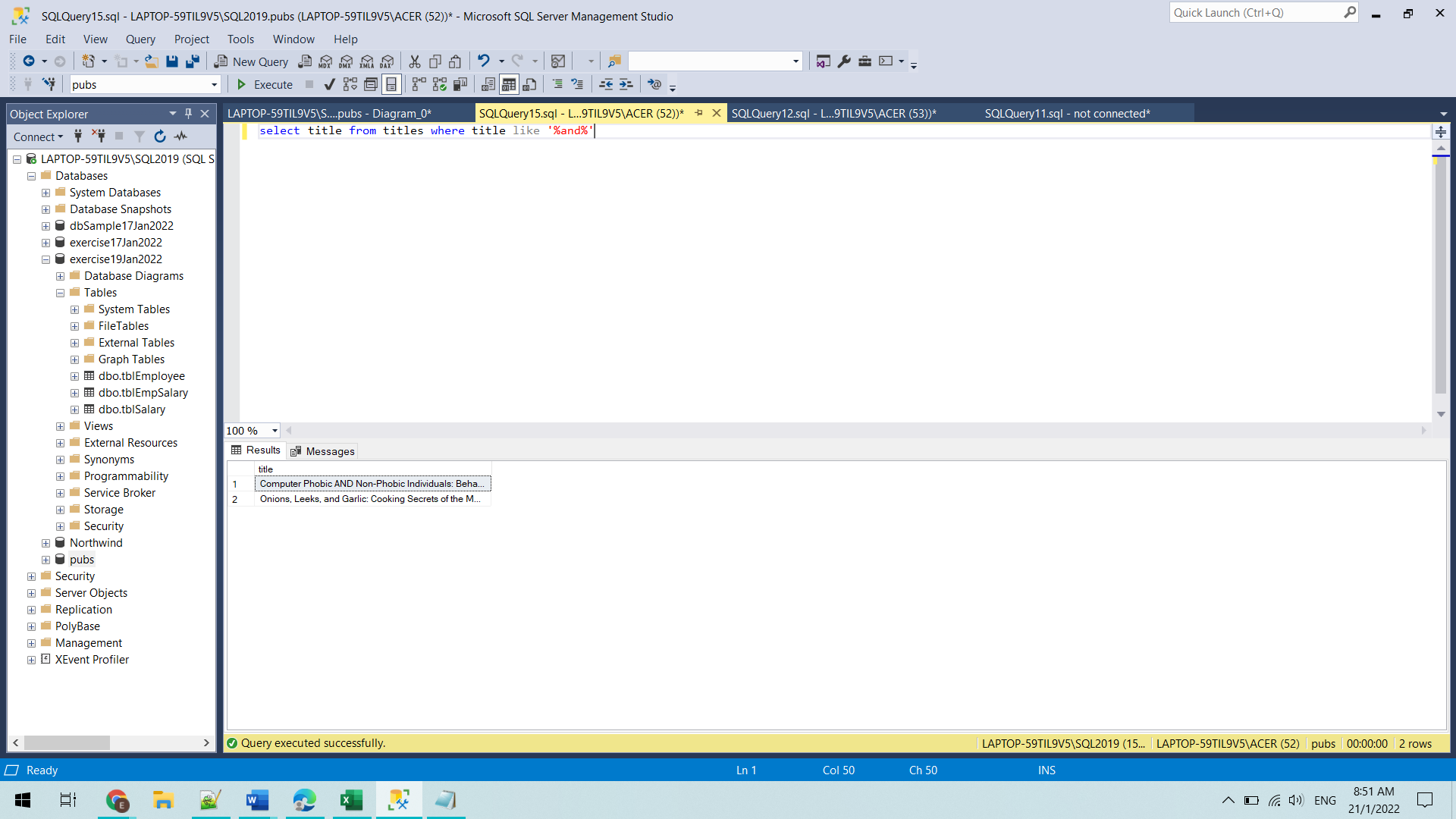
Question 7

select price, title from titles where title like '%s'



Question 8

select title from titles where title like '%and%'

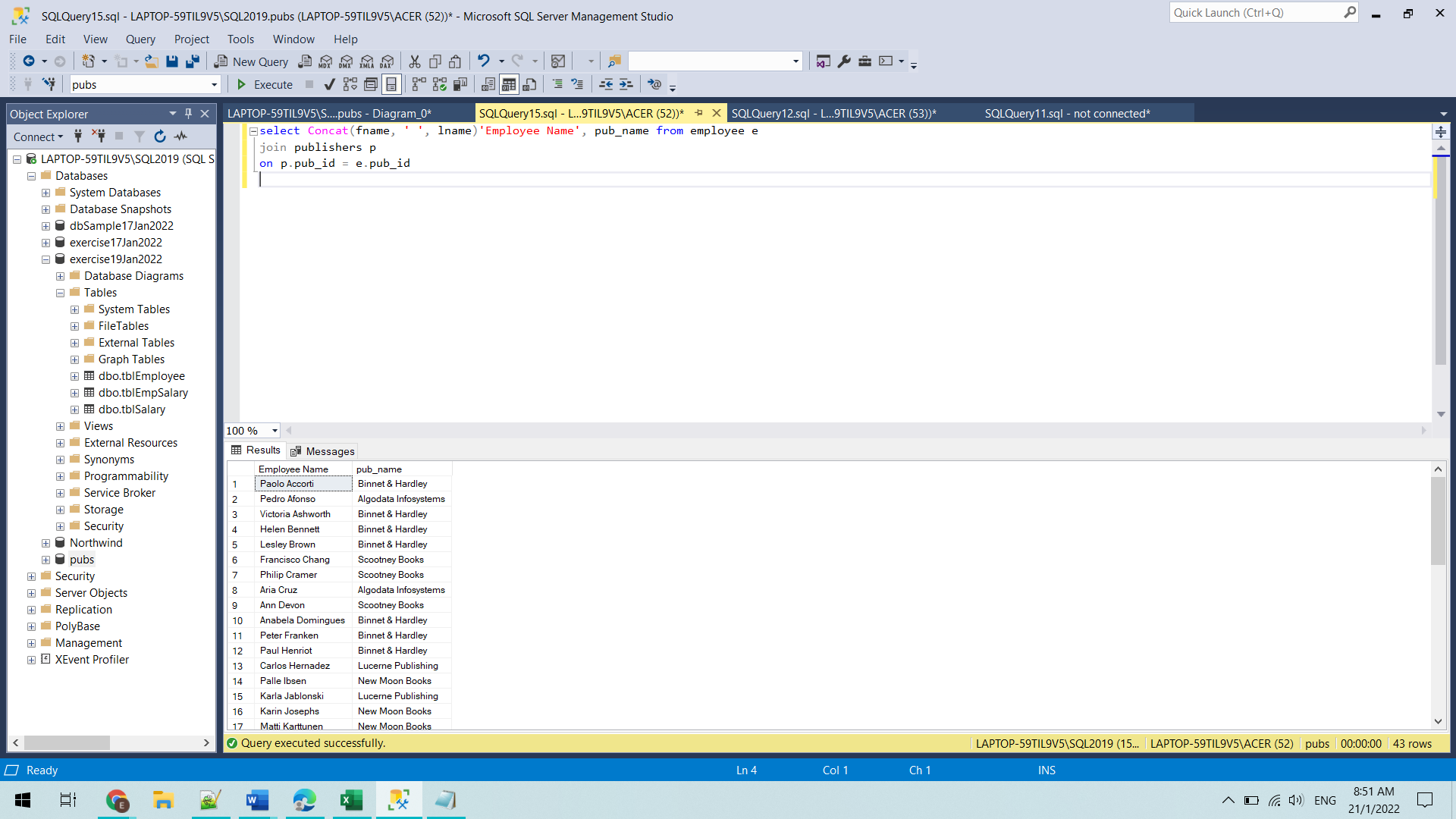


Question 9

select Concat(fname, ' ', lname)'Employee Name', pub\_name from employee e

join publishers p

on p.pub\_id = e.pub\_id



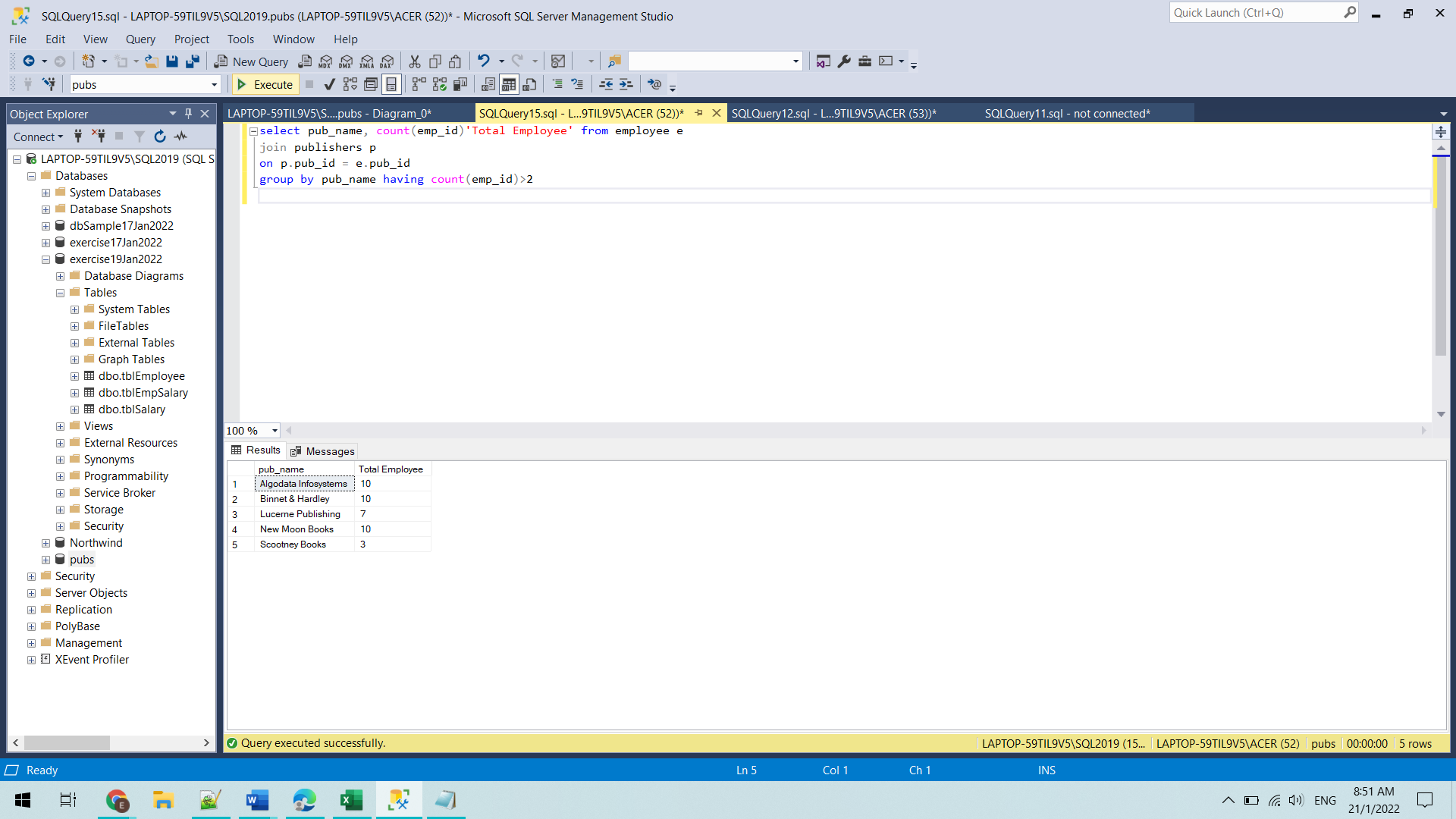
Question 10

select pub\_name, count(emp\_id)'Total Employee' from employee e

join publishers p

on p.pub\_id = e.pub\_id

group by pub\_name having count(emp\_id)>2



Question 11

select Concat(au\_lname, ' ', au\_fname)'Author Name' from authors a

join titleauthor ta

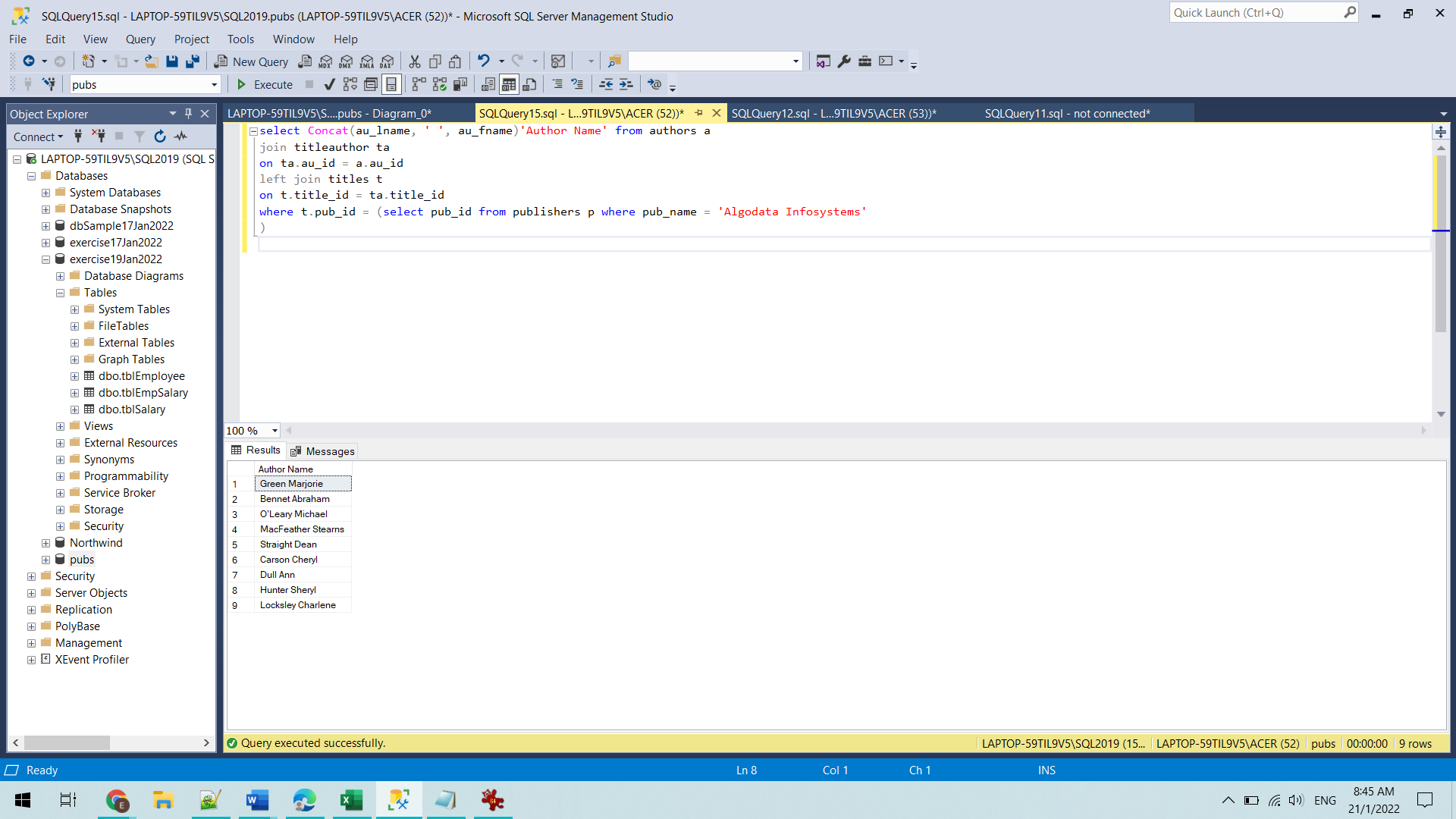
on ta.au\_id = a.au\_id

left join titles t

on t.title\_id = ta.title\_id

where t.pub\_id = (select pub\_id from publishers p where pub\_name = 'Algodata Infosystems'

)



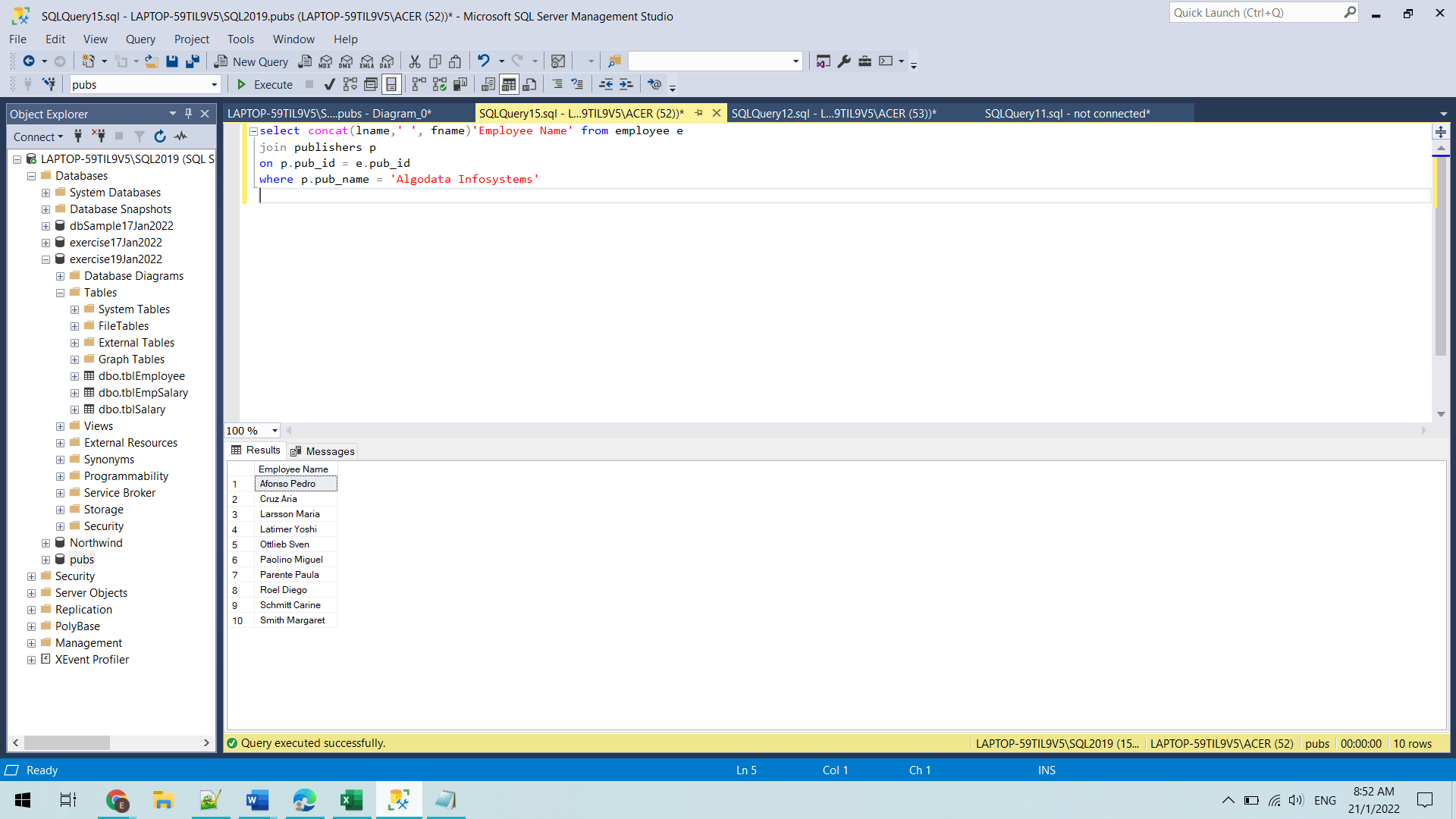
Question 12

select concat(lname,' ', fname)'Employee Name' from employee e

join publishers p

on p.pub\_id = e.pub\_id

where p.pub\_name = 'Algodata Infosystems'



Question 13

13.1

create table tblEmployee(

empID int identity(100,1) primary key,

empName varchar(100),

empAge int,

empPhone varchar(15) not null,

empGender varchar(10)

)

13.2

create table tblSalary(

salID int identity(1,100) primary key,

salBasic float,

salHRA float,

salDA float,

salDeduction float

)

13.3

create table tblEmpSalary(

transNo int primary key,

empID int constraint fk\_empSal references tblEmployee(empID),

salID int constraint fl\_empSal\_1 references tblSalary(salID),

transDate datetime

)

13.4

alter table tblEmployee

add email varchar(100)

13.5

insert into tblEmployee values ('Ester Ding','30','9352008','female','esterding@yahoo.com')

insert into tblEmployee values ('Koh Meng Hu','25','9865889','male','menghu@hotmail.com')

insert into tblEmployee values ('Loqman Abdul','22','9877587','male','abdul@yahoo.com')

insert into tblEmployee values ('Dywah','20','9111452','female','dyway@gmail.com') insert into tblEmployee values ('David Frankton','28','9543210','male','franktond@gmail.com')

insert into tblSalary values(2000,1000,1000,200)

insert into tblSalary values(5000,1500,800,500)

insert into tblSalary values(10000,800,2000,100)

insert into tblSalary values(3500,200,600,500)

insert into tblEmpSalary values(1,101,1,'2021-06-20')

insert into tblEmpSalary values(2,102,101,'2021-06-18')

insert into tblEmpSalary values(3,103,201,'2021-06-15')

insert into tblEmpSalary values(4,100,301,'2021-06-21')

insert into tblEmpSalary values(5,100,201,'2021-07-21')

13.6

create proc proc\_printTotalSal(@empId int, @date datetime)

as

begin

print 'employee id : ' + cast(@empId as varchar(10))

declare

@count int

set @count = (select count(transNo) from tblSalary s

inner join tblEmpSalary es on es.salID = s.salID

where es.empID = @empId and es.transDate = @date)

if(@count >0)

begin

declare

@total float,

@basic float,

@hra float,

@da float,

@deduction float

set @basic = (select salBasic from tblSalary s

inner join tblEmpSalary es on es.salID = s.salID

where es.empID = @empId and es.transDate = @date)

set @hra = (select salHRA from tblSalary s

inner join tblEmpSalary es on es.salID = s.salID

where es.empID = @empId and es.transDate = @date)

set @da = (select salDA from tblSalary s

inner join tblEmpSalary es on es.salID = s.salID

where es.empID = @empId and es.transDate = @date)

set @deduction = (select salDeduction from tblSalary s

inner join tblEmpSalary es on es.salID = s.salID

where es.empID = @empId and es.transDate = @date)

set @total = @basic + @hra + @da - @deduction

print ' salary : ' + cast(@total as varchar(10))

end

else

print ' salary : -'

end

13.7

create proc proc\_calAvgSalary(@empID int)

as

begin

declare

@avgSalary float,

@numOfMonths int

set @numOfMonths = (select count(transNo) from tblEmpSalary

where empID = @empID)

if(@numOfMonths>0)

begin

declare

@totalBasic float,

@totalDeduction float,

@totalhra float,

@totalda float

set @totalBasic = (select sum(s.salBasic) from tblSalary s

inner join tblEmpSalary es

on s.salID = es.salID

and empID = @empID)

set @totalDeduction = (select sum(saldeduction) from tblEmpSalary es

join tblSalary s

on s.salID = es.salID

where empID = @empID)

set @totalhra = (select sum(salHRA) from tblEmpSalary es

join tblSalary s

on s.salID = es.salID

where empID = @empID)

set @totalda = (select sum(salDA) from tblEmpSalary es

join tblSalary s

on s.salID = es.salID

where empID = @empID)

set @avgSalary = (@totalBasic - @totalDeduction + @totalhra + @totalda)/@numOfMonths

print 'Average salary per month is : ' + cast(@avgSalary as varchar(10))

end

else

print 'Average salary per month is : -'

end

13.8

create proc proc\_calTax(@empID int)

as

begin

declare

@count int

set @count = (select count(transNo) from tblEmpSalary

where empID = @empID)

if(@count > 0)

begin

declare

@totalSalary float,

@totalBasic float,

@totalDeduction float,

@totalhra float,

@totalda float

set @totalBasic = (select sum(s.salBasic) from tblSalary s

inner join tblEmpSalary es

on s.salID = es.salID

and empID = @empID)

set @totalDeduction = (select sum(saldeduction) from tblEmpSalary es

join tblSalary s

on s.salID = es.salID

where empID = @empID)

set @totalhra = (select sum(salHRA) from tblEmpSalary es

join tblSalary s

on s.salID = es.salID

where empID = @empID)

set @totalda = (select sum(salDA) from tblEmpSalary es

join tblSalary s

on s.salID = es.salID

where empID = @empID)

set @totalSalary = (@totalBasic - @totalDeduction + @totalhra + @totalda)

print cast(@totalSalary as varchar(10))

if(@totalSalary <= 100000)

print 'Payable tax is : 0'

else if(@totalSalary > 100000 and @totalSalary <= 200000)

print 'Payable tax is : ' + cast(@totalSalary \* 0.05 as varchar(10))

else if(@totalSalary > 200000 and @totalSalary <= 350000 )

print 'Payable tax is : ' + cast(@totalSalary \* 0.06 as varchar(10))

else if(@totalSalary > 350000 )

print 'Payable tax is : ' + cast(@totalSalary \* 0.075 as varchar(10))

end

else

print 'Payable tax is : 0'

end

Question 14

create function fn\_newSum(@empID int, @date datetime)

returns float

as

begin

declare

@sum float,

@basic float,

@hra float,

@da float

set @basic = (select salbasic from tblSalary s

join tblEmpSalary es

on es.salID = s.salID

where es.transDate =@date and empID = @empID)

set @hra = (select salHRA from tblSalary s

join tblEmpSalary es

on es.salID = s.salID

where es.transDate = @date and empID = @empID)

set @da = (select salDA from tblSalary s

join tblEmpSalary es

on es.salID = s.salID

where es.transDate = @date and empID = @empID)

set @sum = @basic + @hra + @da

return @sum

end

Question 15

DECLARE @employeeId int, @employeeName varchar(50), @employeeAge int,

@employeePhone varchar(15), @employeeGender varchar(10), @employeeEmail varchar(100)

DECLARE cur\_emp CURSOR FOR select \* from tblEmployee

OPEN cur\_emp

FETCH NEXT from cur\_emp into @employeeId, @employeeName, @employeeAge, @employeePhone, @employeeGender, @employeeEmail

while (@@FETCH\_STATUS = 0)

BEGIN

print 'Employee Id : '+ CAST(@employeeId as varchar(10))

print 'Employee Name : '+ @employeeName

print 'Employee Age : '+ CAST(@employeeAge as varchar(10))

print 'Employee Phone : '+ @employeePhone

print 'Employee Gender : '+ @employeeGender

print 'Employee Email : '+ @employeeEmail

print '-----------------------------------'

DECLARE @transactionNo int, @salaryId int, @transactionDate datetime

DECLARE cur\_trans CURSOR FOR select transNo, salID, transDate from tblEmpSalary

where empID = @employeeId

OPEN cur\_trans

FETCH NEXT from cur\_trans into @transactionNo, @salaryId, @transactionDate

while(@@FETCH\_STATUS = 0)

BEGIN

print 'Transaction No : '+ CAST(@transactionNo as varchar(10))

print 'Salary ID : '+ CAST(@salaryId as varchar(10))

print 'Transaction Date : '+ CAST(@transactionDate as varchar(20))

print '-----------------------------------'

DECLARE @basicSalary float, @hra float, @ida float, @deduction float

DECLARE cur\_Sal CURSOR FOR select salBasic, salHRA, salDA, salDeduction from tblSalary

where salID = @salaryId

OPEN cur\_Sal

FETCH NEXT from cur\_Sal into @basicSalary, @hra, @ida, @deduction

WHILE(@@FETCH\_STATUS = 0 )

BEGIN

print 'Basic Salary : '+ CAST(@basicSalary as varchar(10))

print 'HRA : '+ CAST(@hra as varchar(10))

print 'IDA : '+ CAST(@ida as varchar(20))

print 'DEDUCTION : '+ CAST(@deduction as varchar(20))

print '-----------------------------------'

FETCH NEXT from cur\_Sal into @basicSalary, @hra, @ida, @deduction

END

CLOSE cur\_Sal

DEALLOCATE cur\_sal

FETCH NEXT from cur\_trans into @transactionNo, @salaryId, @transactionDate

END

CLOSE cur\_trans

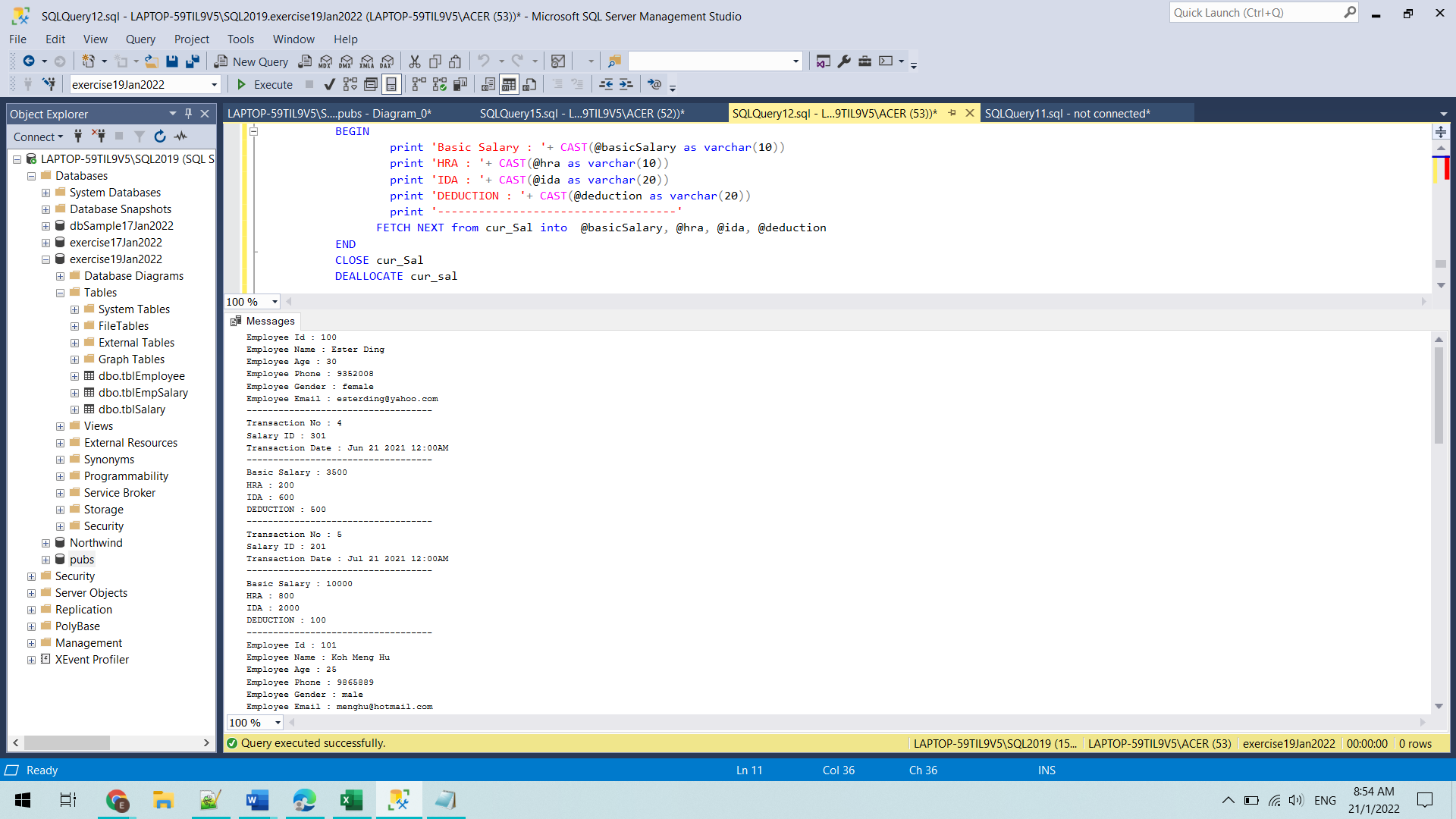
DEALLOCATE cur\_trans

FETCH NEXT from cur\_emp into @employeeId, @employeeName, @employeeAge, @employeePhone, @employeeGender, @employeeEmail

END

CLOSE cur\_emp

DEALLOCATE cur\_emp



17) https://www.hackerrank.com/challenges/maximum-element/problem

18) https://www.geeksforgeeks.org/find-if-there-is-a-subarray-with-0-sum/