SQL Query Join Operation Problem # 13

select Dhruv.matchid, Dhruv.team1,Dhruv.goals,Joe.team2,Joe.goal, CASE

WHEN goals > goal THEN 'Team 1 wins'

WHEN goals < goal THEN 'Team 2 wins'

ELSE 'Tie'

END AS Result

from(

SELECT matchid,team1,team2,teamid,count(player) as "goals"

FROM game JOIN goal ON (id=matchid) where team1 = teamid

group by matchid,teamid) as Dhruv join (

SELECT matchid,team1,team2,teamid,count(player) as "goal"

FROM game JOIN goal ON (id=matchid) where team2 = teamid

group by matchid,teamid ) as Joe on Dhruv.matchid = Joe.matchid

**Module Feedback:**

#4

select \*

from

(

select count(INS\_SPR.SPR\_CODE) as "number" from INS\_SPR join CAM\_SMO on INS\_SPR.SPR\_CODE = CAM\_SMO.SPR\_CODE join INS\_MOD on CAM\_SMO.MOD\_CODE = INS\_MOD.MOD\_CODE join INS\_PRS on INS\_PRS.PRS\_CODE = INS\_MOD.PRS\_CODE join INS\_RES on CAM\_SMO.SPR\_CODE = INS\_RES.SPR\_CODE and CAM\_SMO.MOD\_CODE = INS\_RES.MOD\_CODE and CAM\_SMO.AYR\_CODE = INS\_RES.AYR\_CODE and CAM\_SMO.PSL\_CODE = INS\_RES.PSL\_CODE join INS\_QUE on INS\_QUE.QUE\_CODE = INS\_RES.QUE\_CODE join INS\_CAT on INS\_CAT.CAT\_CODE = INS\_QUE.CAT\_CODE where CAM\_SMO.MOD\_CODE = 'SET08108'AND INS\_QUE.CAT\_CODE = 4 or INS\_QUE.CAT\_CODE = 5 and CAM\_SMO.AYR\_CODE = '2016/7' and CAM\_SMO.PSL\_CODE = 'TR1'

) x

join

(

select count(INS\_SPR.SPR\_CODE) as "number" from INS\_SPR join CAM\_SMO on INS\_SPR.SPR\_CODE = CAM\_SMO.SPR\_CODE join INS\_MOD on CAM\_SMO.MOD\_CODE = INS\_MOD.MOD\_CODE join INS\_PRS on INS\_PRS.PRS\_CODE = INS\_MOD.PRS\_CODE join INS\_RES on CAM\_SMO.SPR\_CODE = INS\_RES.SPR\_CODE and CAM\_SMO.MOD\_CODE = INS\_RES.MOD\_CODE and CAM\_SMO.AYR\_CODE = INS\_RES.AYR\_CODE and CAM\_SMO.PSL\_CODE = INS\_RES.PSL\_CODE join INS\_QUE on INS\_QUE.QUE\_CODE = INS\_RES.QUE\_CODE join INS\_CAT on INS\_CAT.CAT\_CODE = INS\_QUE.CAT\_CODE where CAM\_SMO.MOD\_CODE = 'SET08108' and CAM\_SMO.AYR\_CODE = '2016/7' and CAM\_SMO.PSL\_CODE = 'TR1'

) y

#5

select \*

from

(

select count(INS\_SPR.SPR\_CODE) as "number5" from INS\_SPR join CAM\_SMO on INS\_SPR.SPR\_CODE = CAM\_SMO.SPR\_CODE join INS\_MOD on CAM\_SMO.MOD\_CODE = INS\_MOD.MOD\_CODE join INS\_PRS on INS\_PRS.PRS\_CODE = INS\_MOD.PRS\_CODE join INS\_RES on CAM\_SMO.SPR\_CODE = INS\_RES.SPR\_CODE and CAM\_SMO.MOD\_CODE = INS\_RES.MOD\_CODE and CAM\_SMO.AYR\_CODE = INS\_RES.AYR\_CODE and CAM\_SMO.PSL\_CODE = INS\_RES.PSL\_CODE join INS\_QUE on INS\_QUE.QUE\_CODE = INS\_RES.QUE\_CODE join INS\_CAT on INS\_CAT.CAT\_CODE = INS\_QUE.CAT\_CODE where CAM\_SMO.MOD\_CODE = 'SET08108'AND INS\_QUE.CAT\_CODE = 5 and CAM\_SMO.AYR\_CODE = '2016/7' and CAM\_SMO.PSL\_CODE = 'TR1'

) x

join

(

select count(INS\_SPR.SPR\_CODE) as "number4" from INS\_SPR join CAM\_SMO on INS\_SPR.SPR\_CODE = CAM\_SMO.SPR\_CODE join INS\_MOD on CAM\_SMO.MOD\_CODE = INS\_MOD.MOD\_CODE join INS\_PRS on INS\_PRS.PRS\_CODE = INS\_MOD.PRS\_CODE join INS\_RES on CAM\_SMO.SPR\_CODE = INS\_RES.SPR\_CODE and CAM\_SMO.MOD\_CODE = INS\_RES.MOD\_CODE and CAM\_SMO.AYR\_CODE = INS\_RES.AYR\_CODE and CAM\_SMO.PSL\_CODE = INS\_RES.PSL\_CODE join INS\_QUE on INS\_QUE.QUE\_CODE = INS\_RES.QUE\_CODE join INS\_CAT on INS\_CAT.CAT\_CODE = INS\_QUE.CAT\_CODE where CAM\_SMO.MOD\_CODE = 'SET08108' and CAM\_SMO.AYR\_CODE = '2016/7' and CAM\_SMO.PSL\_CODE = 'TR1' and INS\_QUE.CAT\_CODE = 4

) y

on 1=1

join

(

select count(INS\_SPR.SPR\_CODE) as "number3" from INS\_SPR join CAM\_SMO on INS\_SPR.SPR\_CODE = CAM\_SMO.SPR\_CODE join INS\_MOD on CAM\_SMO.MOD\_CODE = INS\_MOD.MOD\_CODE join INS\_PRS on INS\_PRS.PRS\_CODE = INS\_MOD.PRS\_CODE join INS\_RES on CAM\_SMO.SPR\_CODE = INS\_RES.SPR\_CODE and CAM\_SMO.MOD\_CODE = INS\_RES.MOD\_CODE and CAM\_SMO.AYR\_CODE = INS\_RES.AYR\_CODE and CAM\_SMO.PSL\_CODE = INS\_RES.PSL\_CODE join INS\_QUE on INS\_QUE.QUE\_CODE = INS\_RES.QUE\_CODE join INS\_CAT on INS\_CAT.CAT\_CODE = INS\_QUE.CAT\_CODE where CAM\_SMO.MOD\_CODE = 'SET08108'AND INS\_QUE.CAT\_CODE = 3 and CAM\_SMO.AYR\_CODE = '2016/7' and CAM\_SMO.PSL\_CODE = 'TR1'

) z

on 1=1

join

(

select count(INS\_SPR.SPR\_CODE) as "number2" from INS\_SPR join CAM\_SMO on INS\_SPR.SPR\_CODE = CAM\_SMO.SPR\_CODE join INS\_MOD on CAM\_SMO.MOD\_CODE = INS\_MOD.MOD\_CODE join INS\_PRS on INS\_PRS.PRS\_CODE = INS\_MOD.PRS\_CODE join INS\_RES on CAM\_SMO.SPR\_CODE = INS\_RES.SPR\_CODE and CAM\_SMO.MOD\_CODE = INS\_RES.MOD\_CODE and CAM\_SMO.AYR\_CODE = INS\_RES.AYR\_CODE and CAM\_SMO.PSL\_CODE = INS\_RES.PSL\_CODE join INS\_QUE on INS\_QUE.QUE\_CODE = INS\_RES.QUE\_CODE join INS\_CAT on INS\_CAT.CAT\_CODE = INS\_QUE.CAT\_CODE where CAM\_SMO.MOD\_CODE = 'SET08108'AND INS\_QUE.CAT\_CODE = 2 and CAM\_SMO.AYR\_CODE = '2016/7' and CAM\_SMO.PSL\_CODE = 'TR1'

) a

on 1=1

join

(

select count(INS\_SPR.SPR\_CODE) as "number1" from INS\_SPR join CAM\_SMO on INS\_SPR.SPR\_CODE = CAM\_SMO.SPR\_CODE join INS\_MOD on CAM\_SMO.MOD\_CODE = INS\_MOD.MOD\_CODE join INS\_PRS on INS\_PRS.PRS\_CODE = INS\_MOD.PRS\_CODE join INS\_RES on CAM\_SMO.SPR\_CODE = INS\_RES.SPR\_CODE and CAM\_SMO.MOD\_CODE = INS\_RES.MOD\_CODE and CAM\_SMO.AYR\_CODE = INS\_RES.AYR\_CODE and CAM\_SMO.PSL\_CODE = INS\_RES.PSL\_CODE join INS\_QUE on INS\_QUE.QUE\_CODE = INS\_RES.QUE\_CODE join INS\_CAT on INS\_CAT.CAT\_CODE = INS\_QUE.CAT\_CODE where CAM\_SMO.MOD\_CODE = 'SET08108'AND INS\_QUE.CAT\_CODE = 1 and CAM\_SMO.AYR\_CODE = '2016/7' and CAM\_SMO.PSL\_CODE = 'TR1'

) b

on 1=1

**HELP DESK:**

**Help Desk #13**

Select \* from(

select \*, DATE\_FORMAT(Call\_date,'%Y-%m-%d %H')Date\_hour,

DATE\_FORMAT(Call\_date,'%Y-%m-%d') Date,

DATE\_FORMAT(Call\_date,'%H') Hours,

DATE\_FORMAT(Call\_date,'%i') Time

from Issue join Staff on

Issue.Taken\_by = Staff.Staff\_code) as Dhruv

join

(select Shift.Shift\_type,DATE\_FORMAT(Shift\_date, '%Y-%m-%d %H') date\_hour,

DATE\_FORMAT(Shift\_date, '%Y-%m-%d') date,

DATE\_FORMAT(Shift\_date, '%H') hour,

EXTRACT(Hour FROM Start\_time)start,

EXTRACT(Hour FROM End\_time)end

from Shift join Shift\_type on Shift.Shift\_type = Shift\_type.Shift\_type) as David

on Dhruv.Date = David.date where Time >= 55 and Hours = end -1

#14

select \* from(

select count(distinct Name) as Total\_Cust,Company\_name from (

select Call\_ref,Issue.Caller\_id,Company\_name, DATE\_FORMAT(Call\_date,'%Y-%m-%d %H')Date\_hour,

DATE\_FORMAT(Call\_date,'%Y-%m-%d') Date,

DATE\_FORMAT(Call\_date,'%H') Hours,

DATE\_FORMAT(Call\_date,'%i') Time,

CONCAT(First\_name, ' ', Last\_name) as Name

from Issue join Caller on Issue.Caller\_id = Caller.Caller\_id join

Customer on Customer.Company\_ref = Caller.Company\_ref) as Dhruv group by Company\_name) as Joey

Join

(select count(distinct Name) as Total\_Custs,Company\_name from (

select Call\_ref,Issue.Caller\_id,Company\_name, DATE\_FORMAT(Call\_date,'%Y-%m-%d %H')Date\_hour,

DATE\_FORMAT(Call\_date,'%Y-%m-%d') Date,

DATE\_FORMAT(Call\_date,'%H') Hours,

DATE\_FORMAT(Call\_date,'%i') Time,

CONCAT(First\_name, ' ', Last\_name) as Name

from Issue join Caller on Issue.Caller\_id = Caller.Caller\_id join

Customer on Customer.Company\_ref = Caller.Company\_ref) as Dhruv where Date = '2017-08-13' group by Company\_name) as Jake

on Joey.Company\_name = Jake.Company\_name where Total\_cust = Total\_Custs

#15

Select \* from(

Select \*,RANK() OVER (

PARTITION BY Taken,Date

ORDER BY diff ASC

) rank

from(

Select \*,TIMESTAMPDIFF(MINUTE,Date,Calldate) as "diff" from(

Select Taken\_by as "Taken", Call\_date as "Date",First\_call from (

Select \*,

CASE WHEN Moretime <=10 and LessTime >10 or LessTime IS NULL THEN

1

ELSE 0

END AS "First\_call"

From(

Select \*,

TIMESTAMPDIFF(MINUTE,Call\_date,NextDate) as "Moretime",

TIMESTAMPDIFF(MINUTE,PrevDate,Call\_date)as "LessTime"

from(

select \*,

Lag(Call\_date, 1) OVER(

ORDER BY Call\_date,Taken\_by ASC) AS PrevDate,

Lead(Call\_date, 1) OVER(

ORDER BY Call\_date,Taken\_by ASC) AS NextDate

from(

select \*, DATE\_FORMAT(Call\_date,'%Y-%m-%d %H:%i')Date\_hour,

DATE\_FORMAT(Call\_date,'%Y-%m-%d') Date,

DATE\_FORMAT(Call\_date,'%H') Hours,

DATE\_FORMAT(Call\_date,'%i') Time

from Issue join Staff on

Issue.Taken\_by = Staff.Staff\_code order by Date,Taken\_by,Hours,Time) as Dhruv) as John) as Dennis having First\_call = 1) as Johnny) as Jesuses

Join

(Select Taken\_by as "Takenby", Call\_date as "CallDate",Last\_call as "LastCall" from (

Select \*,

CASE WHEN Moretime >10 and LessTime <=10 THEN

1

ELSE 0

END AS "Last\_call"

From(

Select \*,

TIMESTAMPDIFF(MINUTE,Call\_date,NextDate) as "Moretime",

TIMESTAMPDIFF(MINUTE,PrevDate,Call\_date)as "LessTime"

from(

select \*,

Lag(Call\_date, 1) OVER(

ORDER BY Call\_date,Taken\_by ASC) AS PrevDate,

Lead(Call\_date, 1) OVER(

ORDER BY Call\_date,Taken\_by ASC) AS NextDate

from(

select \*, DATE\_FORMAT(Call\_date,'%Y-%m-%d %H:%i')Date\_hour,

DATE\_FORMAT(Call\_date,'%Y-%m-%d') Date,

DATE\_FORMAT(Call\_date,'%H') Hours,

DATE\_FORMAT(Call\_date,'%i') Time

from Issue join Staff on

Issue.Taken\_by = Staff.Staff\_code order by Date,Taken\_by,Hours,Time) as Dhruv) as John) as Dennis having Last\_call = 1) as Johnny) as Jesus on Taken = Takenby having TIMESTAMPDIFF(MINUTE,Date,Calldate) >=0 order by Date,CallDate) as Jenny) as Donkey where rank = 1 and diff = (

Select max(diff) from(

Select \*,RANK() OVER (

PARTITION BY Taken,Date

ORDER BY diff ASC

) rank

from(

Select \*,TIMESTAMPDIFF(MINUTE,Date,Calldate) as "diff" from(

Select Taken\_by as "Taken", Call\_date as "Date",First\_call from (

Select \*,

CASE WHEN Moretime <=10 and LessTime >10 or LessTime IS NULL THEN

1

ELSE 0

END AS "First\_call"

From(

Select \*,

TIMESTAMPDIFF(MINUTE,Call\_date,NextDate) as "Moretime",

TIMESTAMPDIFF(MINUTE,PrevDate,Call\_date)as "LessTime"

from(

select \*,

Lag(Call\_date, 1) OVER(

ORDER BY Call\_date,Taken\_by ASC) AS PrevDate,

Lead(Call\_date, 1) OVER(

ORDER BY Call\_date,Taken\_by ASC) AS NextDate

from(

select \*, DATE\_FORMAT(Call\_date,'%Y-%m-%d %H:%i')Date\_hour,

DATE\_FORMAT(Call\_date,'%Y-%m-%d') Date,

DATE\_FORMAT(Call\_date,'%H') Hours,

DATE\_FORMAT(Call\_date,'%i') Time

from Issue join Staff on

Issue.Taken\_by = Staff.Staff\_code order by Date,Taken\_by,Hours,Time) as Dhruv) as John) as Dennis having First\_call = 1) as Johnny) as Jesuses

Join

(Select Taken\_by as "Takenby", Call\_date as "CallDate",Last\_call as "LastCall" from (

Select \*,

CASE WHEN Moretime >10 and LessTime <=10 THEN

1

ELSE 0

END AS "Last\_call"

From(

Select \*,

TIMESTAMPDIFF(MINUTE,Call\_date,NextDate) as "Moretime",

TIMESTAMPDIFF(MINUTE,PrevDate,Call\_date)as "LessTime"

from(

select \*,

Lag(Call\_date, 1) OVER(

ORDER BY Call\_date,Taken\_by ASC) AS PrevDate,

Lead(Call\_date, 1) OVER(

ORDER BY Call\_date,Taken\_by ASC) AS NextDate

from(

select \*, DATE\_FORMAT(Call\_date,'%Y-%m-%d %H:%i')Date\_hour,

DATE\_FORMAT(Call\_date,'%Y-%m-%d') Date,

DATE\_FORMAT(Call\_date,'%H') Hours,

DATE\_FORMAT(Call\_date,'%i') Time

from Issue join Staff on

Issue.Taken\_by = Staff.Staff\_code order by Date,Taken\_by,Hours,Time) as Dhruv) as John) as Dennis having Last\_call = 1) as Johnny) as Jesus on Taken = Takenby having TIMESTAMPDIFF(MINUTE,Date,Calldate) >=0 order by Date,CallDate) as Jenny) as Donkey where rank = 1)

**Hotel Guest**

11 -

select \*, CASE

WHEN checkin = booking\_date then 1

WHEN checkin = checkout then 2

WHEN check\_out = booking\_date then 3

WHEN check\_out = checkout then 4

WHEN checkin < booking\_date and check\_out > booking\_date then 5

WHEN checkin >booking\_date and checkin < checkout then 6

WHEN check\_out > booking\_date and check\_out < checkout then 7

ELSE 0

END as "Logic"

from(

select \* from (

select booking\_id,booking\_date,first\_name,last\_name, DATE\_ADD(booking\_date,INTERVAL nights DAY) as "checkout" from booking join guest on booking.guest\_id = id)as Joel

Join

(select booking\_id as "booked",booking\_date as "checkin",first\_name as "first",last\_name as "last", DATE\_ADD(booking\_date,INTERVAL nights DAY) as "check\_out" from booking join guest on booking.guest\_id = id) as Jen

where last\_name = last and first\_name!=first) as Dan having Logic <> 0

# 12.

select count(booking\_id),check\_out,Room from(

select \*,CASE

WHEN room\_no BETWEEN 100 and 199 then 'Room1'

WHEN room\_no BETWEEN 200 and 299 then 'Room2'

WHEN room\_no BETWEEN 300 and 399 then 'Room3'

ElSE 4

END AS "Room",

DATE\_ADD(booking\_date,INTERVAL nights DAY) as "check\_out" from booking join room on room\_no = room.id) as Don

group by room,check\_out order by check\_out

#13

select distinct(room\_no) from booking where room\_no not in (

select DISTINCT (room\_no) from(

select \*,

CASE

WHEN "2016-11-25" between checkin and checkout AND checkout != "2016-11-25" then 1

ELSE 0

END As "OCCUPIED"

from(

Select \*,DATE\_FORMAT(check\_out,'%Y-%m-%d') as checkout from(

Select \*,DATE\_ADD(booking\_date,INTERVAL nights DAY) as "check\_out",DATE\_FORMAT(booking\_date,'%Y-%m-%d') as checkin from booking join room on room\_no = room.id) as Joe) as Jon) as Jake where OCCUPIED in (1))

#14

SELECT room\_no,max(checkout) as MaxTime from (

select \*,DATE\_FORMAT(check\_out,'%Y-%m-%d') as checkout from(

select \*,DATE\_FORMAT(booking\_date,'%Y-%m-%d') as checkin ,DATE\_ADD(booking\_date,INTERVAL nights DAY) as check\_out from booking join rate on room\_type\_requested = room\_type) as Dhruv where room\_type = 'single' and checkin >= '2016-11-25' order by room\_no asc,checkin asc) as Joel group by room\_no

#15 You get the rate of every room by joining. Then you identify the closest Friday to checkout date as long as it’s not the next day and then you sum and group by all the Fridays and you will get the money

select \*, booking\_id,room\_no,nights,pay,booking\_date,checkout,bkdwy,chkdwy,BKBegFri,BKEndThurs,COBegFri,COEndThurs from(

select \*, CASE

WHEN chkdwy = 7 then DATE\_ADD(checkout,INTERVAL 5 DAY)

WHEN chkdwy = 6 then DATE\_ADD(checkout,INTERVAL 6 DAY)

WHEN chkdwy = 5 then checkout

WHEN chkdwy = 4 then DATE\_ADD(checkout,INTERVAL 1 DAY)

WHEN chkdwy = 3 then DATE\_ADD(checkout,INTERVAL 2 DAY)

WHEN chkdwy = 2 then DATE\_ADD(checkout,INTERVAL 3 DAY)

WHEN chkdwy = 1 then DATE\_ADD(checkout,INTERVAL 4 DAY)

Else 0

END AS "COEndThurs",

CASE

WHEN chkdwy = 7 then DATE\_ADD(checkout,INTERVAL -1 DAY)

WHEN chkdwy = 6 then checkout

WHEN chkdwy = 5 then DATE\_ADD(checkout,INTERVAL -6 DAY)

WHEN chkdwy = 4 then DATE\_ADD(checkout,INTERVAL -5 DAY)

WHEN chkdwy = 3 then DATE\_ADD(checkout,INTERVAL -4 DAY)

WHEN chkdwy = 2 then DATE\_ADD(checkout,INTERVAL -3 DAY)

WHEN chkdwy = 1 then DATE\_ADD(checkout,INTERVAL -2 DAY)

ELSE 0

END AS "COBegFri",

CASE

WHEN bkdwy = 7 then DATE\_ADD(checkout,INTERVAL 5 DAY)

WHEN bkdwy = 6 then DATE\_ADD(checkout,INTERVAL 6 DAY)

WHEN bkdwy = 5 then booking\_date

WHEN bkdwy = 4 then DATE\_ADD(checkout,INTERVAL 1 DAY)

WHEN bkdwy = 3 then DATE\_ADD(checkout,INTERVAL 2 DAY)

WHEN bkdwy = 2 then DATE\_ADD(checkout,INTERVAL 3 DAY)

WHEN bkdwy = 1 then DATE\_ADD(checkout,INTERVAL 4 DAY)

Else 0

END AS "BKEndThurs",

CASE

WHEN Bkdwy = 7 then DATE\_ADD(checkout,INTERVAL -1 DAY)

WHEN bkdwy = 6 then checkout

WHEN bkdwy = 5 then DATE\_ADD(checkout,INTERVAL -6 DAY)

WHEN bkdwy = 4 then DATE\_ADD(checkout,INTERVAL -5 DAY)

WHEN bkdwy = 3 then DATE\_ADD(checkout,INTERVAL -4 DAY)

WHEN bkdwy = 2 then DATE\_ADD(checkout,INTERVAL -3 DAY)

WHEN bkdwy = 1 then DATE\_ADD(checkout,INTERVAL -2 DAY)

ELSE 0

END AS "BKBegFri"

from(

select \*,DAYOFWEEK(booking\_date)as "bkdwy", DAYOFWEEK(checkout) as "chkdwy" from (

select booking\_id,room\_no,nights,(amount \* nights) as pay,booking\_date, DATE\_ADD(booking\_date,INTERVAL nights DAY)as checkout from booking join rate on room\_type\_requested = room\_type) as Dhruv) as Joel) as Jan

**Adventure Works**

Question 11 –

select \* from(

Select Customer.CustomerID,AddressType,AddressLine1,City,Address.AddressID,CompanyName from Customer join CustomerAddress on Customer.CustomerID = CustomerAddress.AddressID join Address on CustomerAddress.AddressID = Address.AddressID where Customer.CustomerID IN (

Select Customer.CustomerID from Customer join CustomerAddress on Customer.CustomerID = CustomerAddress.AddressID join Address on CustomerAddress.AddressID = Address.AddressID and City = 'Dallas')) as Joe

#12

select SalesOrderHeader.SalesOrderID, OrderQty \* UnitPrice, OrderQty\*ListPrice from SalesOrderHeader join SalesOrderDetail on SalesOrderHeader.SalesOrderID = SalesOrderDetail.SalesOrderID join Product on Product.ProductID = SalesOrderDetail.ProductID

#13

select ProductID,sum(Sum1) ,sum(Sum2) from (

select SalesOrderDetail.SalesOrderID,Product.ProductID, OrderQty \* UnitPrice as "Sum1", OrderQty\*ListPrice as "Sum2" from SalesOrderHeader join SalesOrderDetail on SalesOrderHeader.SalesOrderID = SalesOrderDetail.SalesOrderID join Product on Product.ProductID = SalesOrderDetail.ProductID) as Joe group by ProductID order by sum(Sum1)desc

#14-

Select sum(John.Order) as 'Orders',sum(John.Value) as 'Value',John.Range from(

Select \*, CASE

WHEN VALUE BETWEEN 0 AND 99 THEN 'RANGE1'

WHEN VALUE BETWEEN 100 AND 999 THEN 'RANGE2'

WHEN VALUE BETWEEN 1000 and 9999 THEN 'RANGE3'

ELSE 'RANGE4'

END AS "RANGE"

from(

select sum(UnitPrice) as "Value", Sum(OrderQty) as "Order",SalesOrderID from(

select SalesOrderHeader.SalesOrderID,OrderQty,UnitPrice

from SalesOrderHeader join SalesOrderDetail on SalesOrderHeader.SalesOrderID = SalesOrderDetail.SalesOrderID join Product on Product.ProductID = SalesOrderDetail.ProductID) as Joe group by SalesOrderID order by Sum(OrderQty) desc) as Joe) as John

group by John.Range

#15

Select \* from(

Select \*,RANK()OVER(

PARTITION BY Jen.City

ORDER BY Jen.total DESC)price\_rank from(

Select sum(John.total) as "Total",John.Name,John.City from(

Select Customer.CustomerID,City,OrderQty,UnitPrice,OrderQty\*UnitPrice as "total",ProductCategory.Name from Customer join CustomerAddress on Customer.CustomerID = CustomerAddress.CustomerID join Address on CustomerAddress.AddressID = Address.AddressID join SalesOrderHeader on SalesOrderHeader.CustomerID = Customer.CustomerID join SalesOrderDetail on SalesOrderHeader.SalesOrderID = SalesOrderDetail.SalesOrderID join Product on Product.ProductID = SalesOrderDetail.ProductID join ProductCategory on ProductCategory.ProductCategoryID = Product.ProductCategoryID where City IN (select City from (

Select sum(total),City from(

Select Customer.CustomerID,City,OrderQty,UnitPrice,OrderQty\*UnitPrice as "total",ProductCategory.Name from Customer join CustomerAddress on Customer.CustomerID = CustomerAddress.CustomerID join Address on CustomerAddress.AddressID = Address.AddressID join SalesOrderHeader on SalesOrderHeader.CustomerID = Customer.CustomerID join SalesOrderDetail on SalesOrderHeader.SalesOrderID = SalesOrderDetail.SalesOrderID join Product on Product.ProductID = SalesOrderDetail.ProductID join ProductCategory on ProductCategory.ProductCategoryID = Product.ProductCategoryID) as Joe group by City order by sum(total) desc limit 3) as Jan)) as John group by John.Name,John.City order by City,Total desc) as Jen) as John where price\_rank <4

**University**

#11-

Have to add some case whens to the statement so that I ensure when they are in between

select \* from(

select staff.id as "ID",dow,tod,tod + duration as "end" from staff join teaches on staff.id = teaches.staff join event on event.id = teaches.event where staff.id = 'co.CHt') as John join (

select staff.id as "iD",dow as "DOW",tod as "TOD",tod + duration as "End" from staff join teaches on staff.id = teaches.staff join event on event.id = teaches.event where staff.id = 'co.ACg') as Jen on John.dow = Jen.DOW