How to Code Summary Queries Assignment

NOTE: You must delete the db1 database, download the db1 create script.sql from Blackboard and run the script. I have added the HAFAMORE schema to the file.

NAME: T.S.

1. Determine the highest gross pay by the employee in the table [db1].[costco].[employee]. Use the tables [db1].[costco].[job], [db1].[costco].[hours] & [db1].[costco].[employee]. Format your column with the correct column heading (see solution example) and format the data returned by your query with a leading dollar sign ($), comas every three digits and two decimal places.

Query goes here

select format(max([hours].hours \* job.BASE\_PAY), 'C') as 'Highest Weekly Gross Pay'

from Costco.employee

join Costco.hours on employee.EMP\_CODE = hours.emp\_code

join costco.job on employee.JOB\_CODE = job.JOB\_CODE

Graphical user interface, text, application, email

Description automatically generated

1. Determine the lowest weekly sales from the WEST region. Use the tables, [db1].[costco].[store\_sales], [db1].[costco].[region]. Format your column with the correct column heading (see solution example) and format the data returned by your query with a leading dollar sign ($), comas every three digits and two decimal places.

Query goes here

select format(min(weekly\_sales), 'C') as 'West region lowest weekly sales'

from Costco.store\_sales

join Costco.Region on Store\_sales.region\_code =region.REGION\_CODE

where costco.region.REGION\_CODE =2

Screen shot goes here

Graphical user interface, text, application, email

Description automatically generated

1. Determine the lowest starting bid for any item the [db1].[costco].[item] table where the opening bid is in the month of January 2008. Format your column with the correct column heading (see solution example) and format the data returned by your query with a leading dollar sign ($), comas every three digits and two decimal places.

Query goes here

select format(min(startingBid),'c')as 'minimum starting Bid'

from costco.item

where openDateTime BETWEEN '2008-01-01' and '2008-01-31'

Screen shot goes here

Graphical user interface, text, application, email

Description automatically generated

1. List the just the store name once and total weekly sales of the stores and the total weekly sales for the all the stores. Use the [db1].[costco].[store\_sales] & [db1].[costco].[store] tables. Format your column with the correct column heading (see solution example) and format the data returned by your query with a leading dollar sign ($), comas every three digits and two decimal places.

Hint: Use grouping sets.

Query goes here

select STORE\_NAME,format(sum(Weekly\_sales),'c')as 'weekly sale'

from costco.store

join Costco.store\_sales on store.store\_code =store\_sales.STORE\_CODE

group by STORE\_NAME

UNION ALL

select '',format(sum(Weekly\_sales),'c')

FROM costco.store\_sales

Screen shot goes here

Table

Description automatically generated with medium confidence

1. List the total weekly sales by store name, region and the grand total. Use the [db1].[costco].[store\_sales], [db1].[costco].[region] & [db1].[costco].[store] tables. Format your column with the correct column heading (see solution example) and format the data returned by your query with a leading dollar sign ($), comas every three digits and two decimal places.

Hint: Use grouping sets.

Query goes here

select st.STORE\_NAME, r.DESCRIPT, format(sum(s.Weekly\_Sales), 'C') as 'Weekly Sales'

from db1.Costco.store st

join db1.Costco.store\_sales s on st.STORE\_CODE = s.store\_Code

join db1.Costco.region r on s.region\_code = r.REGION\_CODE

and r.DESCRIPT = 'East'

group by st.STORE\_NAME, r.DESCRIPT

union all

select 'NULL', 'East', format(sum(s.Weekly\_Sales), 'C')

from db1.Costco.store st

join db1.Costco.store\_sales s on st.STORE\_CODE = s.store\_Code

join db1.Costco.region r on s.region\_code = r.REGION\_CODE

and r.DESCRIPT = 'East'

union all

select st.STORE\_NAME, r.DESCRIPT, format(sum(s.Weekly\_Sales), 'C')

from db1.Costco.store st

join db1.Costco.store\_sales s on st.STORE\_CODE = s.store\_Code

join db1.Costco.region r on s.region\_code = r.REGION\_CODE

and r.DESCRIPT = 'North'

group by st.STORE\_NAME, r.DESCRIPT

union all

select 'NULL', 'North', format(sum(s.Weekly\_Sales), 'C')

from db1.Costco.store st

join db1.Costco.store\_sales s on st.STORE\_CODE = s.store\_Code

join db1.Costco.region r on s.region\_code = r.REGION\_CODE

and r.DESCRIPT = 'North'

union all

select st.STORE\_NAME, r.DESCRIPT, format(sum(s.Weekly\_Sales), 'C')

from db1.Costco.store st

join db1.Costco.store\_sales s on st.STORE\_CODE = s.store\_Code

join db1.Costco.region r on s.region\_code = r.REGION\_CODE

and r.DESCRIPT = 'South'

group by st.STORE\_NAME, r.DESCRIPT

union all

select 'NULL', 'South', format(sum(s.Weekly\_Sales), 'C')

from db1.Costco.store st

join db1.Costco.store\_sales s on st.STORE\_CODE = s.store\_Code

join db1.Costco.region r on s.region\_code = r.REGION\_CODE

and r.DESCRIPT = 'South'

union all

select st.STORE\_NAME, r.DESCRIPT, format(sum(s.Weekly\_Sales), 'C')

from db1.Costco.store st

join db1.Costco.store\_sales s on st.STORE\_CODE = s.store\_Code

join db1.Costco.region r on s.region\_code = r.REGION\_CODE

and r.DESCRIPT = 'West'

group by st.STORE\_NAME, r.DESCRIPT

union all

select 'NULL', 'West', format(sum(s.Weekly\_Sales), 'C')

from db1.Costco.store st

join db1.Costco.store\_sales s on st.STORE\_CODE = s.store\_Code

join db1.Costco.region r on s.region\_code = r.REGION\_CODE

and r.DESCRIPT = 'West'

union all

--all regions

select 'NULL', 'NULL', format(sum(s.Weekly\_Sales), 'C')

from db1.Costco.store st

join db1.Costco.store\_sales s on st.STORE\_CODE = s.store\_Code

join db1.Costco.region r on s.region\_code = r.REGION\_CODE

union all

--all stores

select st.STORE\_NAME, 'NULL', format(sum(s.Weekly\_Sales), 'C')

from db1.Costco.store st

join db1.Costco.store\_sales s on st.STORE\_CODE = s.store\_Code

group by st.STORE\_NAME

Screen shot goes here

Table

Description automatically generated

1. Display the ManagerID, MFName, MLName, and number of buildings managed, for all managers that manage more than one building. Use the HAFHMORE schema.

Query goes here

select ManagerID, MFname,MLname, COUNT(\*)

FROM HAFHMORE.manager

join HAFHMORE.building on manager.managerid=building.bmanagerid

group BY managerID, MFname,MLname HAVING COUNT(\*)>1

Table

Description automatically generated

1. Display the ManagerID, MFName, MLName, and the number of buildings managed, for all managers. Use the HAFHMORE schema.

Query goes here

SELECT managerid, mfname, mlname, count(\*)

from HAFHMORE.manager m

JOIN HAFHMORE.building b on m.managerid = b.bmanagerid

group by managerid, mfname, mlname

Screen shot goes here

Table

Description automatically generated

1. Display the SMemberID, SMemberName, and the number of apartments that the staff members cleans, for all staff members. Use the HAFHMORE schema.

Query goes here

SELECT s.smemberid, smembername, COUNT(aptno)

from HAFHMORE.cleaning c

JOIN HAFHMORE.staffmember s on s.smemberid = c.smemberid

group by s.smemberid, smembername

Screen shot goes here

Table

Description automatically generated

1. Display the total amount HAFH spends on manager salaries (as TotalSalary) and the total amount HAFH spends on manager bonuses (as TotalBonus). Use the HAFHMORE schema. Format the data returned by your query with a leading dollar sign ($), comas every three digits and two decimal places.

Query goes here

SELECT

format(SUM(msalary), '$##,##0.00') as TotalSalary ,

format(SUM(mbonus), '$##,##0.00') as TotalBonus

from

HAFHMORE.manager

Graphical user interface, text, application, email

Description automatically generated

1. Display the answer to the following question: How many HAFH buildings have exactly 4 floors? Use the HAFHMORE schema.

Query goes here

SELECT

COUNT(bnooffloors) as COUNT

from

HAFHMORE.building

WHERE bnooffloors = 4

Graphical user interface, application

Description automatically generated