**M2L2Q1**

**Code for Last Part**

proc sql;

title "Customers Whose Average Retail Price Exceeds the Average Retail Price for All Customers";

select Customer\_ID, avg(Total\_Retail\_Price) as MeanPrice /\*This is meansales in the output in the doc\*/

from ST522.order\_fact

where Order\_Type=1

group by Customer\_ID

having avg(Total\_Retail\_Price) >

(select avg(Total\_Retail\_Price)

from ST522.order\_fact

where Order\_Type=1

);

quit;

**Log for Last Part**

1 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

72

73 proc sql;

74 title "Customers Whose Average Retail Price Exceeds the Average Retail Price for All

74 ! Customers";

75 select Customer\_ID, avg(Total\_Retail\_Price) as MeanPrice /\*This is meansales in the output

75 ! in the doc\*/

76 from ST522.order\_fact

77 where Order\_Type=1

78 group by Customer\_ID

79 having avg(Total\_Retail\_Price) >

80 (select avg(Total\_Retail\_Price)

81 from ST522.order\_fact

82 where Order\_Type=1

83 );

84

85 quit;

NOTE: PROCEDURE SQL used (Total process time):

real time 0.06 seconds

cpu time 0.06 seconds

86

87 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

99

**Output for Last Part**

Table

Description automatically generated

**M2L2Q2**

**Code for Last Part**

/\*Q2\*/

%let my\_date\_formatted = %sysfunc(today(), MONTH.);

%PUT &my\_date\_formatted;

\* B ;

proc sql;

title "Employees with Current Month Anniversaries";

select Employee\_ID,

scan(Employee\_Name, 2, ",") LENGTH=15 LABEL="First Name" as First,

scan(Employee\_Name, 1, ",") LENGTH=15 LABEL="Last Name" as Last

from ST522.employee\_addresses

where Employee\_ID in

(select Employee\_ID

from ST522.employee\_payroll

where MONTH(Employee\_Hire\_Date) = &my\_date\_formatted

)

order by Last;

quit;

**Log for Last Part**

1 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

72

73 /\*Q2\*/

74

75 %let my\_date\_formatted = %sysfunc(today(), MONTH.);

76 %PUT &my\_date\_formatted;

1

77

78 \* B ;

79

80 proc sql;

81 title "Employees with Current Month Anniversaries";

82 select Employee\_ID,

83 scan(Employee\_Name, 2, ",") LENGTH=15 LABEL="First Name" as First,

84 scan(Employee\_Name, 1, ",") LENGTH=15 LABEL="Last Name" as Last

85 from ST522.employee\_addresses

86 where Employee\_ID in

87 (select Employee\_ID

88 from ST522.employee\_payroll

89 where MONTH(Employee\_Hire\_Date) = &my\_date\_formatted

90 )

91 order by Last;

92 quit;

NOTE: PROCEDURE SQL used (Total process time):

real time 0.14 seconds

cpu time 0.14 seconds

93

94 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

106

**Output for Last Part**

These should differ from the pdf, because I actually did it based on the current day not February. Last I checked it was the same for Feb though.

Table

Description automatically generated

**M2L2Q3**

**Code for Last Part**

/\*Q3\*/

%let date = 02FEB2013;

%PUT &date;

proc sql;

title "Level I or II Purchasing Agents Who are older than ALL Purchasing Agent IIIs ";

select Employee\_ID,

Job\_Title LABEL="Employee Job Title ",

Birth\_Date LABEL="Employee Birth Date",

floor(yrdif(Birth\_Date, '02FEB2013'd)) as Age

from ST522.staff

where Job\_Title in ('Purchasing Agent I', 'Purchasing Agent II')

and yrdif(Birth\_Date, '02FEB2013'd) > all(

select yrdif(Birth\_Date, '02FEB2013'd) as Age

from ST522.staff

where Job\_Title = 'Purchasing Agent III'

);

quit;

**Log for Last Part**

1 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

72

73 /\*Q3\*/

74

75 %let date = 02FEB2013;

76 %PUT &date;

02FEB2013

77

78 proc sql;

79 title "Level I or II Purchasing Agents Who are older than ALL Purchasing Agent IIIs ";

80 select Employee\_ID,

81 Job\_Title LABEL="Employee Job Title ",

82 Birth\_Date LABEL="Employee Birth Date",

83 floor(yrdif(Birth\_Date, '02FEB2013'd)) as Age

84 from ST522.staff

85 where Job\_Title in ('Purchasing Agent I', 'Purchasing Agent II')

86 and yrdif(Birth\_Date, '02FEB2013'd) > all(

87 select yrdif(Birth\_Date, '02FEB2013'd) as Age

88 from ST522.staff

89 where Job\_Title = 'Purchasing Agent III'

90 );

91 quit;

NOTE: PROCEDURE SQL used (Total process time):

real time 0.05 seconds

cpu time 0.04 seconds

92

93 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

105

**Output for Last Part**

You may notice that the Age is 4 more on this one than the PDF because apparently the birthdate is also 4 years sooner, only took me 30 minutes to figure out why the result looked right but was off by 4 years from the PDF

Graphical user interface

Description automatically generated with medium confidence

**M2L2Q4**

**Code for Last Part**

/\*Q4\*/

proc sql;

title "Latest Order Date for Orion Club Low Activity Members";

select Customer\_ID,

Last LABEL="Order Date" FORMAT=date9. as Order

from

/\* This gets the last date per customer in the low activity subquery\*/

(

select Customer\_ID,

max(Order\_Date) as Last

from ST522.order\_fact as of

where of.Customer\_ID in

/\* This subquery just gets the unique customer id's that are low

activity, also the column names are apparently different in the

different tables\*/

(

select unique CustomerID

from ST522.customer\_type as ct,

ST522.customer as c

where ct.Customer\_Type = 'Orion Club members low activity' and

c.CustomerTypeID = ct.Customer\_Type\_ID

)

group by of.Customer\_ID

) as MaxDate

/\* This is the condition for if the last order date for the

virtual table is old enough\*/

where '01JAN2012'd > MaxDate.Last;

quit;

**Log for Last Part**

1 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

72

73 /\*Q4\*/

74

75 proc sql;

76 title "Latest Order Date for Orion Club Low Activity Members";

77 select Customer\_ID,

78 Last LABEL="Order Date" FORMAT=date9. as Order

79 from

80 /\* This gets the last date per customer in the low activity subquery\*/

81 (

82 select Customer\_ID,

83 max(Order\_Date) as Last

84 from ST522.order\_fact as of

85 where of.Customer\_ID in

86 /\* This subquery just gets the unique customer id's that are low

87 activity, also the column names are apparently different in the

88 different tables\*/

89 (

90 select unique CustomerID

91 from ST522.customer\_type as ct,

92 ST522.customer as c

93 where ct.Customer\_Type = 'Orion Club members low activity' and

94 c.CustomerTypeID = ct.Customer\_Type\_ID

95 )

96 group by of.Customer\_ID

97 ) as MaxDate

98 /\* This is the condition for if the last order date for the

99 virtual table is old enough\*/

100 where '01JAN2012'd > MaxDate.Last;

101 quit;

NOTE: PROCEDURE SQL used (Total process time):

real time 0.06 seconds

cpu time 0.07 seconds

102

103 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

115

**Output for Last Part**

Table

Description automatically generated

**M2L2Q5**

**Code for Last Part**

\*B;

proc sql;

title "2011 Sales Summary By Country";

select Country,

max(Value\_Sold) FORMAT=7.2 as MaxVal,

max(Orders) FORMAT=4.2 as MaxOrd,

max(Avg\_Order) FORMAT=7.2 as MaxAvg,

min(Avg\_Order) FORMAT=7.2 as MinAvg

from (

select Country,

First\_Name,

Last\_Name,

sum(Total\_Retail\_Price) FORMAT=7.2 as Value\_Sold,

count(distinct Order\_ID) as Orders,

(calculated Value\_Sold/ calculated Orders) FORMAT=7.2 as Avg\_Order

from ST522.order\_fact as of,

ST522.sales as s

where s.Employee\_ID = of.Employee\_ID and

year(Order\_Date) = 2011

group by Country, First\_Name, Last\_Name

) as Combined

where Value\_Sold ge 200

group by Country

order by Country;

quit;

**Log for Last Part**

101

102 \*B;

103

104 proc sql;

105 title "2011 Sales Summary By Country";

106 select Country,

107 max(Value\_Sold) FORMAT=7.2 as MaxVal,

108 max(Orders) FORMAT=4.2 as MaxOrd,

109 max(Avg\_Order) FORMAT=7.2 as MaxAvg,

110 min(Avg\_Order) FORMAT=7.2 as MinAvg

111 from (

112 select Country,

113 First\_Name,

114 Last\_Name,

115 sum(Total\_Retail\_Price) FORMAT=7.2 as Value\_Sold,

116 count(distinct Order\_ID) as Orders,

117 (calculated Value\_Sold/ calculated Orders) FORMAT=7.2 as Avg\_Order

118 from ST522.order\_fact as of,

119 ST522.sales as s

120 where s.Employee\_ID = of.Employee\_ID and

121 year(Order\_Date) = 2011

122 group by Country, First\_Name, Last\_Name

123 ) as Combined

124 where Value\_Sold ge 200

125 group by Country

126 order by Country;

127 quit;

NOTE: PROCEDURE SQL used (Total process time):

real time 0.04 seconds

cpu time 0.04 seconds

128

129 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

141

**Output for Last Part**

Table

Description automatically generated

**M2L2Q6**

**Code for Last Part**

\*C;

proc sql;

title "Employee Salaries as a percent of Department Total";

select C.Department,

C.Employee\_Name,

ep.Salary,

(ep.Salary / C.Dept\_Salary\_Total) as Percent

from (

select A.Department as Department,

A.Dept\_Salary\_Total as Dept\_Salary\_Total,

B.Employee\_ID as Employee\_ID,

B.Employee\_Name as Employee\_Name

from (

select Department,

sum(Salary) as Dept\_Salary\_Total

from ST522.employee\_payroll as ep,

ST522.employee\_organization as eo

where ep.Employee\_ID = eo.Employee\_ID

group by eo.Department

) as A

inner join (

select ea.Employee\_ID as Employee\_ID,

eo.Department as Department,

ea.Employee\_Name as Employee\_Name

from ST522.employee\_addresses as ea

inner join ST522.employee\_organization as eo

on ea.Employee\_ID = eo.Employee\_ID

) as B

on A.Department = B.Department

) as C

inner join ST522.employee\_payroll as ep

on ep.Employee\_ID = C.Employee\_ID

order by Department, Percent desc;

quit;

**Log for Last Part**

107

108 \*C;

109

110 proc sql;

111 title "Employee Salaries as a percent of Department Total";

112 select C.Department,

113 C.Employee\_Name,

114 ep.Salary,

115 (ep.Salary / C.Dept\_Salary\_Total) as Percent

116 from (

117 select A.Department as Department,

118 A.Dept\_Salary\_Total as Dept\_Salary\_Total,

119 B.Employee\_ID as Employee\_ID,

120 B.Employee\_Name as Employee\_Name

121 from (

122 select Department,

123 sum(Salary) as Dept\_Salary\_Total

124 from ST522.employee\_payroll as ep,

125 ST522.employee\_organization as eo

126 where ep.Employee\_ID = eo.Employee\_ID

127 group by eo.Department

128 ) as A

129 inner join (

130 select ea.Employee\_ID as Employee\_ID,

131 eo.Department as Department,

132 ea.Employee\_Name as Employee\_Name

133 from ST522.employee\_addresses as ea

134 inner join ST522.employee\_organization as eo

135 on ea.Employee\_ID = eo.Employee\_ID

136 ) as B

137 on A.Department = B.Department

138 ) as C

139 inner join ST522.employee\_payroll as ep

140 on ep.Employee\_ID = C.Employee\_ID

141 order by Department, Percent desc;

142 quit;

NOTE: PROCEDURE SQL used (Total process time):

real time 0.40 seconds

cpu time 0.39 seconds

143

144 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

156

**Output for Last Part**

Table

Description automatically generated

**M2L2Q7**

**Code for Last Part**

/\*Q7\*/

proc sql;

title "2011 Total Sales Figures";

select cat(Manager\_First, ' ', Manager\_Last) as Manager,

cat(Employee\_First, ' ', Employee\_Last) as Employee,

Total\_Sales

from (

select scan(Names.EName, 1, ', ') as Employee\_Last,

scan(Names.EName, 2, ', ') as Employee\_First,

scan(Names.MName, 1, ', ')as Manager\_Last,

scan(Names.MName, 2, ', ') as Manager\_First,

Sales.Total\_Sales as Total\_Sales,

Names.Country as Country

from (select e.Employee\_ID as Employee\_ID,

e.Employee\_Name as EName,

e.Country as Country,

m.Employee\_ID as Manager\_ID,

m.Employee\_Name as MName

from ST522.employee\_organization as o,

ST522.employee\_addresses as m,

ST522.employee\_addresses as e

where e.Employee\_ID = o.Employee\_ID and

o.Manager\_ID = m.Employee\_ID

) as Names,

(

select Employee\_ID,

sum(Total\_Retail\_Price) as Total\_Sales

from ST522.order\_fact

where year(Delivery\_Date) = 2011

group by Employee\_ID

) as Sales

where Names.Employee\_ID = Sales.Employee\_ID

) as Combined

order by Country, Manager\_Last, Manager\_First, Total\_Sales desc;

quit;

**Log for Last Part**

1 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

72

73 /\*Q7\*/

74

75 proc sql;

76 title "2011 Total Sales Figures";

77 select cat(Manager\_First, ' ', Manager\_Last) as Manager,

78 cat(Employee\_First, ' ', Employee\_Last) as Employee,

79 Total\_Sales

80 from (

81 select scan(Names.EName, 1, ', ') as Employee\_Last,

82 scan(Names.EName, 2, ', ') as Employee\_First,

83 scan(Names.MName, 1, ', ')as Manager\_Last,

84 scan(Names.MName, 2, ', ') as Manager\_First,

85 Sales.Total\_Sales as Total\_Sales,

86 Names.Country as Country

87 from (select e.Employee\_ID as Employee\_ID,

88 e.Employee\_Name as EName,

89 e.Country as Country,

90 m.Employee\_ID as Manager\_ID,

91 m.Employee\_Name as MName

92 from ST522.employee\_organization as o,

93 ST522.employee\_addresses as m,

94 ST522.employee\_addresses as e

95 where e.Employee\_ID = o.Employee\_ID and

96 o.Manager\_ID = m.Employee\_ID

97 ) as Names,

98 (

99 select Employee\_ID,

100 sum(Total\_Retail\_Price) as Total\_Sales

101 from ST522.order\_fact

102 where year(Delivery\_Date) = 2011

103 group by Employee\_ID

104 ) as Sales

105 where Names.Employee\_ID = Sales.Employee\_ID

106 ) as Combined

107

108 order by Country, Manager\_Last, Manager\_First, Total\_Sales desc;

NOTE: The query as specified involves ordering by an item that doesn't appear in its SELECT clause.

109 quit;

NOTE: PROCEDURE SQL used (Total process time):

real time 0.11 seconds

cpu time 0.12 seconds

110

111 OPTIONS NONOTES NOSTIMER NOSOURCE NOSYNTAXCHECK;

123

**Output for Last Part**

Table

Description automatically generated