SQL Step by Step:

An advanced tutorial for business users

Judy Loren
Independent Consultant

Gregory S. Barnes Nelson
ASG, Inc.

SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

Abstract

Some excellent papers have been presented at SUGI and NESUG giving reasons to use SQL (Hermansen), overviews of SQL (Winn), selected features of SQL (Pass & Dickson, Peters), or behind the scene details of joins (Kent). This paper aims to solidify your understanding of the power of SQL by presenting an integrated series of examples that progress from the straightforward to the intricate. The paper will focus on coding but will pay some attention to performance issues and use of SQL via pass-thru to other databases.

Scope of this Paper

The purpose of this paper is to introduce advanced users of the SAS System to some of the more complicated tasks that can be accomplished using SQL. This is not intended to provide the reader with a blueprint of all the functionality found in SQL, but rather a focused exercise into the complexity of the language. Each of the examples is presented in order of common business questions that might have been asked.

The example we will be using throughout the paper will be data from the Northwinds Trading Company database that is distributed with Microsoft Access. The database is an order entry system for a company that sells a broad range of consumable/food products. We selected this database for building our sample warehouse application for three reasons: (1) it is available to anyone who has MS-Access, (2) MS-Access is ODBC compliant and it was easy to get this data into SAS via SAS/ACCESS to ODBC, and (3) it provides a classic example of an operational system used in order processing that most of us have some familiarity with.

Data Structure

Just as a programmer would have to know what data elements are available in their operational data store, we too must understand the type of data that we have in order to answer questions about that data. Figure 1 shows out main subject entity-relationship diagram (ERD) that tells us basically what tables we have in our database.

Figure 1. Main subject Area ERD

Note that this diagram shows the relationships between tables in our database. At the lowest level of detail we have information about specific orders that were placed at our fictitious company. For a detailed entity-relationship diagram, please refer to Appendix A.

From there, we can see the data in our system that support that level of detail, for example, we have information about our products, our customers, our employees in addition to having ancillary information about those things.

Now that we have at least a cursory understanding of our data, let's move forward and try to answer some basic business questions about this data. Let's turn our data into information.
From Data to Information

This paper is organized into 14 increasingly complex business questions. Each question will be presented along with a SQL solution. A brief description will follow each solution.

Q1 Manager asks, ingenuously, “What are sales for 1994?”

Concepts Introduced:
- PROC SQL
- Title statement
- Select (with as)
- Summary functions (down variables)
- Formatting values
- From (Multiple datasets)
- Where (and)
- YEAR function

Solution

libname db "C:\WINDOWS\Nesug97\Sasdata";
proc sql;
title "Sales for 1994";
SELECT SUM(qty*unitprc) as sales format = dollar15.
FROM db.orddtls dtl,
db.orders ord
WHERE dtl.ordid = ord.ordid
and year(orddate) = 1994;

Results

Sales for 1994

<table>
<thead>
<tr>
<th>SALES</th>
<th>$339,885</th>
</tr>
</thead>
</table>

Q2 Manager says, “How do I know whether this is good or bad? Show me how it’s changed since last year!”

Concepts Introduced:
- CASE here is used to create a BY or CLASS variable-type analysis
- Summary functions (down and across variables)
- Calculated keyword

Solution

libname db "C:\WINDOWS\Nesug97\Sasdata";
proc sql;
title "Sales Growth 1993 to 1994 by Quarter";
SELECT %calc(1993,n)
%calc(1994,y)
FROM db.orddtls dtl,
db.orders ord
WHERE dtl.ordid = ord.ordid
and year(orddate) = 1994;

Results

Sales Growth 1993 to 1994 by Quarter

<table>
<thead>
<tr>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
</tr>
</thead>
<tbody>
<tr>
<td>SALES94</td>
<td>$339,885</td>
<td>$339,885</td>
</tr>
<tr>
<td>SALES93</td>
<td>$716,569</td>
<td>$716,569</td>
</tr>
<tr>
<td>PCTCHG</td>
<td>(52.57%)</td>
<td>(52.57%)</td>
</tr>
</tbody>
</table>
Manager says, “Oh, that’s right, we’re only 1 quarter into 1994. Silly me. Show me the latest year vs the previous 12 months. And let’s look at whether we’ve been able to add customers to our base. And check whether we’ve lost any customers. In fact, show me whether customers are increasing their sales or decreasing their sales.”

Concepts Introduced:
- Coalesce
- Full Join
- Group by
- Order
- Subquery
- CASE logic to create a new variable (flag)

Solution

```
SELECT COALESCE(list93.custid, list94.custid) as custid,
  case when sales94 > sales93 and sales93 > 0 then "Growing"
       when sales94 > sales93 and sales93 <=0 then "New"
       when sales94 < sales93 and sales94 > 0 then "Shrinking"
       when sales94 <= 0 then "Lost"
       else "Not Known" end as type,
  FROM (SELECT custid, SUM(qty*unitprc) as sales93
       FROM   db.orders ord, db.orddtls dtl
       WHERE  ord.ordid=dtl.ordid
             and  ((year(orddate)=1993 and  qtr(orddate) =1)
               or (year(orddate)=1992 and qtr(orddate) gt 1))
       GROUP BY custid) as list93
FULL JOIN
  (SELECT custid, SUM(qty*unitprc) as sales94
   FROM   db.orders ord, db.orddtls dtl
   WHERE  ord.ordid=dtl.ordid
          and (year(orddate)=1994 and qtr(orddate) =1)
          or (year(orddate)=1993 and qtr(orddate) gt 1))
   GROUP BY custid) as list94
ON list93.custid = list94.custid
ORDER BY type, custid;
```

Results

```
Customer Growth Analysis

<table>
<thead>
<tr>
<th>CUSTID</th>
<th>Company Name</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANATR</td>
<td>Ana Trujillo Empanadas y helados</td>
<td>Growing</td>
</tr>
<tr>
<td>ANTON</td>
<td>Antonio Moreno Taqueria</td>
<td>Growing</td>
</tr>
<tr>
<td>ABOUT</td>
<td>Around the Horn</td>
<td>Growing</td>
</tr>
</tbody>
</table>
```

Manager says, “What are these ids? I need the company name.”

Concepts Introduced:
- Multi-table outer join (full join ON var=var, table)

Solution

```
SELECT COALESCE(list93.custid, list94.custid) as custid,
  company label='Company Name',
  case when sales94 > sales93 and sales93 > 0 then "Growing"
       when sales94 > sales93 and sales93 <=0 then "New"
       when sales94 < sales93 and sales94 > 0 then "Shrinking"
       when sales94 <= 0 then "Lost"
       else "Not Known" end as type,
  FROM (SELECT custid, SUM(qty*unitprc) as sales93
       FROM   db.orders ord, db.orddtls dtl
       WHERE  ord.ordid=dtl.ordid
             and  ((year(orddate)=1993 and  qtr(orddate) =1)
               or (year(orddate)=1992 and qtr(orddate) gt 1))
       GROUP BY custid) as list93
FULL JOIN
  (SELECT custid, SUM(qty*unitprc) as sales94
   FROM   db.orders ord, db.orddtls dtl
   WHERE  ord.ordid=dtl.ordid
          and (year(orddate)=1994 and qtr(orddate) =1)
          or (year(orddate)=1993 and qtr(orddate) gt 1))
   GROUP BY custid) as list94
ON list93.custid = list94.custid
ORDER BY type, custid;
```
Manager says, “I can’t tell anything from this list. Show me where the companies are located.”

Concepts Introduced:
- Reset flow

Solution

```
RESET FLOW:
SELECT COALESCE(list93.custid, list94.custid) as custid,
company label='Company Name',
country label='Company Location',
case when sales94 > sales93 and sales93 > 0 then "Growing"
when sales94 > sales93 and sales93 <=0 then "New"
when sales94 < sales93 and sales94 > 0 then "Shrinking"
when sales94 <= 0 then "Lost"
else "Not Known"
end as type label="Customer Growth Category"
FROM (SELECT custid, SUM(qty*unitprc) as sales93
FROM db.orders ord, db.orddtls dtl
WHERE  ord.ordid=dtl.ordid
and  ((year(orddate)=1993 and  qtr(orddate) =1)
or (year(orddate)=1992 and qtr(orddate) gt 1))
GROUP BY custid) as list93
FULL JOIN
(SELECT custid, SUM(qty*unitprc) as sales94
FROM db.orders ord, db.orddtls dtl
WHERE ord.ordid=dtl.ordid
and ((year(orddate)=1994 and qtr(orddate) =1)
or (year(orddate)=1993 and qtr(orddate) gt 1))
GROUP BY custid) as list94
ON list93.custid = list94.custid,
db.custs cust
WHERE CALCULATED custid = cust.custid
ORDER BY type, custid
```

Results

Customer Growth Analysis

<table>
<thead>
<tr>
<th>CustID</th>
<th>Company Name</th>
<th>Company Location</th>
<th>Growth Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANATR</td>
<td>Ana Trujillo</td>
<td>Mexico</td>
<td>Growing</td>
</tr>
<tr>
<td>ANTON</td>
<td>Antonio Moreno</td>
<td>Mexico</td>
<td>Growing</td>
</tr>
<tr>
<td>ABOUT</td>
<td>Around the Horn</td>
<td>UK</td>
<td>Growing</td>
</tr>
<tr>
<td>WOLZA</td>
<td>Wolski  Zajazd</td>
<td>Poland</td>
<td>Growing</td>
</tr>
<tr>
<td>CENTC</td>
<td>Centro comercial</td>
<td>Mexico</td>
<td>Lost</td>
</tr>
<tr>
<td>WITC</td>
<td>White Clover Markets</td>
<td>USA</td>
<td>Growing</td>
</tr>
<tr>
<td>WOLSA</td>
<td>Wolski  Zajazd</td>
<td>Poland</td>
<td>Growing</td>
</tr>
<tr>
<td>ALFKI</td>
<td>Alfreds Futterkiste</td>
<td>Germany</td>
<td>New</td>
</tr>
<tr>
<td>FRANR</td>
<td>France restauration</td>
<td>France</td>
<td>New</td>
</tr>
<tr>
<td>LACOR</td>
<td>La corne d'abondance</td>
<td>France</td>
<td>New</td>
</tr>
<tr>
<td>LETSS</td>
<td>Let's Stop N Shop</td>
<td>USA</td>
<td>New</td>
</tr>
<tr>
<td>THECR</td>
<td>The Cracker Box</td>
<td>USA</td>
<td>New</td>
</tr>
<tr>
<td>TRAIH</td>
<td>Trail's Head Gourmet Provisioners</td>
<td>USA</td>
<td>New</td>
</tr>
<tr>
<td>WILMK</td>
<td>Wilman Kala</td>
<td>Finland</td>
<td>New</td>
</tr>
<tr>
<td>BLONP</td>
<td>Blondel père et fils</td>
<td>France</td>
<td>Shrinking</td>
</tr>
<tr>
<td>COMMI</td>
<td>Comércio Mineiro</td>
<td>Brazil</td>
<td>Shrinking</td>
</tr>
<tr>
<td>COMINH</td>
<td>Consolidated Holdings</td>
<td>UK</td>
<td>Shrinking</td>
</tr>
</tbody>
</table>

Manager says, “Well this list is too long to really tell what’s going on. Just show me how many are in each growth category by country.”

Concepts Introduced:
- Create view
- Subqueries (instead of view?)

Solution

```
RESET FLOW :
create view growth as
SELECT COALESCE(list93.custid, list94.custid) as custid,
company label='Company Name',
country label='Company Location',
case when sales94 > sales93 and sales93 > 0 then "Growing"
when sales94 > sales93 and sales93 <=0 then "New"
when sales94 < sales93 and sales94 > 0 then "Shrinking"
when sales94 <= 0 then "Lost"
else "Not Known"
end as type label="Customer Growth Category"
FROM (SELECT custid, SUM(qty*unitprc) as sales93
FROM db.orders ord, db.orddtls dtl
WHERE  ord.ordid=dtl.ordid
and  ((year(orddate)=1993 and  qtr(orddate) =1)
or (year(orddate)=1992 and qtr(orddate) gt 1))
GROUP BY custid) as list93
FULL JOIN
(SELECT custid, SUM(qty*unitprc) as sales94
FROM db.orders ord, db.orddtls dtl
WHERE ord.ordid=dtl.ordid
and ((year(orddate)=1994 and qtr(orddate) =1)
or (year(orddate)=1993 and qtr(orddate) gt 1))
GROUP BY custid) as list94
ON list93.custid = list94.custid,
db.custs cust
WHERE CALCULATED custid = cust.custid
ORDER BY type, country
;
select count(*) as cos label='Companies', country, type
from growth
where type = "Growing"
GROUP BY country, type
;
```

Results

Customer Growth Analysis

<table>
<thead>
<tr>
<th>CustID</th>
<th>Company Name</th>
<th>Company Location</th>
<th>Growth Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANATR</td>
<td>Ana Trujillo</td>
<td>Mexico</td>
<td>Growing</td>
</tr>
<tr>
<td>ANTON</td>
<td>Antonio Moreno</td>
<td>Mexico</td>
<td>Growing</td>
</tr>
<tr>
<td>ABOUT</td>
<td>Around the Horn</td>
<td>UK</td>
<td>Growing</td>
</tr>
<tr>
<td>WOLZA</td>
<td>Wolski  Zajazd</td>
<td>Poland</td>
<td>Growing</td>
</tr>
<tr>
<td>CENTC</td>
<td>Centro commercial</td>
<td>Mexico</td>
<td>Lost</td>
</tr>
<tr>
<td>WITC</td>
<td>White Clover Markets</td>
<td>USA</td>
<td>Growing</td>
</tr>
<tr>
<td>WOLSA</td>
<td>Wolski  Zajazd</td>
<td>Poland</td>
<td>Growing</td>
</tr>
<tr>
<td>ALFKI</td>
<td>Alfreds Futterkiste</td>
<td>Germany</td>
<td>New</td>
</tr>
<tr>
<td>FRANR</td>
<td>France restauration</td>
<td>France</td>
<td>New</td>
</tr>
<tr>
<td>LACOR</td>
<td>La corne d'abondance</td>
<td>France</td>
<td>New</td>
</tr>
<tr>
<td>LETSS</td>
<td>Let's Stop N Shop</td>
<td>USA</td>
<td>New</td>
</tr>
<tr>
<td>THECR</td>
<td>The Cracker Box</td>
<td>USA</td>
<td>New</td>
</tr>
<tr>
<td>TRAIH</td>
<td>Trail's Head Gourmet Provisioners</td>
<td>USA</td>
<td>New</td>
</tr>
<tr>
<td>WILMK</td>
<td>Wilman Kala</td>
<td>Finland</td>
<td>New</td>
</tr>
<tr>
<td>BLONP</td>
<td>Blondel père et fils</td>
<td>France</td>
<td>Shrinking</td>
</tr>
<tr>
<td>COMMI</td>
<td>Comércio Mineiro</td>
<td>Brazil</td>
<td>Shrinking</td>
</tr>
<tr>
<td>COMINH</td>
<td>Consolidated Holdings</td>
<td>UK</td>
<td>Shrinking</td>
</tr>
</tbody>
</table>

Results

Customer Growth Analysis

<table>
<thead>
<tr>
<th>CustID</th>
<th>Company Name</th>
<th>Company Location</th>
<th>Growth Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANATR</td>
<td>Ana Trujillo</td>
<td>Mexico</td>
<td>Growing</td>
</tr>
<tr>
<td>ANTON</td>
<td>Antonio Moreno</td>
<td>Mexico</td>
<td>Growing</td>
</tr>
<tr>
<td>ABOUT</td>
<td>Around the Horn</td>
<td>UK</td>
<td>Growing</td>
</tr>
<tr>
<td>WOLZA</td>
<td>Wolski  Zajazd</td>
<td>Poland</td>
<td>Growing</td>
</tr>
<tr>
<td>CENTC</td>
<td>Centro comercial</td>
<td>Mexico</td>
<td>Lost</td>
</tr>
<tr>
<td>WITC</td>
<td>White Clover Markets</td>
<td>USA</td>
<td>Growing</td>
</tr>
<tr>
<td>WOLSA</td>
<td>Wolski  Zajazd</td>
<td>Poland</td>
<td>Growing</td>
</tr>
<tr>
<td>ALFKI</td>
<td>Alfreds Futterkiste</td>
<td>Germany</td>
<td>New</td>
</tr>
<tr>
<td>FRANR</td>
<td>France restauration</td>
<td>France</td>
<td>New</td>
</tr>
<tr>
<td>LACOR</td>
<td>La corne d'abondance</td>
<td>France</td>
<td>New</td>
</tr>
<tr>
<td>LETSS</td>
<td>Let's Stop N Shop</td>
<td>USA</td>
<td>New</td>
</tr>
<tr>
<td>THECR</td>
<td>The Cracker Box</td>
<td>USA</td>
<td>New</td>
</tr>
<tr>
<td>TRAIH</td>
<td>Trail's Head Gourmet Provisioners</td>
<td>USA</td>
<td>New</td>
</tr>
<tr>
<td>WILMK</td>
<td>Wilman Kala</td>
<td>Finland</td>
<td>New</td>
</tr>
<tr>
<td>BLONP</td>
<td>Blondel père et fils</td>
<td>France</td>
<td>Shrinking</td>
</tr>
<tr>
<td>COMMI</td>
<td>Comércio Mineiro</td>
<td>Brazil</td>
<td>Shrinking</td>
</tr>
<tr>
<td>COMINH</td>
<td>Consolidated Holdings</td>
<td>UK</td>
<td>Shrinking</td>
</tr>
</tbody>
</table>
Manager says, “I can’t take any action on this! I want to know which countries have increased sales and which have lost sales in the last 12 months! And show it to me in some kind of order so I know which countries are the best and which are the worst!”

Concepts Introduced:
- summary statistics (across variables)
- DESC on GROUP BY

Solution

```sql
TITLE "Geographic Growth Analysis";
SELECT COALESCE(l1.list93.country, l1.list94.country) as country,
       l1.country, l1.sales93 as sales93,
       l1.sales94 as sales94,
       ((l1.sales94/l1.sales93 - 1) * 100) as pctchg
FROM (SELECT country, SUM(qty*unitprc) as sales93
      FROM db.orders ord, db.orddtls dtl, db.custs cust
      WHERE ord.ordid=dtl.ordid
      AND (year(orddate)=1992 and qtr(orddate) > 1)
      GROUP BY country) as l1,
      (SELECT country, SUM(qty*unitprc) as sales94
      FROM db.orders ord, db.orddtls dtl, db.custs cust
      WHERE ord.ordid=dtl.ordid
      AND (year(orddate)=1993 and qtr(orddate) > 1)
      GROUP BY country) as l2
FULL JOIN
GROUP BY country) as list93
ON l1.country = list93.country
ORDER BY pctchg desc;
```

Manager says, “Well how can I tell what’s going on if I don’t know what’s selling? Maybe the countries doing more business like some of our product categories better than others. Show me the growth in business by product category, and index the percent change to the total company growth rate so I can see which ones are driving our growth!

Concepts Introduced:
- Reset noprint
- How to fill a macro variable from SQL
- Putting grand total on subtotal lines

Solution

```sql
libname db "C:\WINNT\Mesu99\7\qsa\data";
proc sql;
reset noprint;
reset flow;
TITLE "Growth in Product Categories Compared to Overall Growth";
SELECT SUM(CASE WHEN year(orddate)=1994
               and (year(orddate)=1993 and qtr(orddate) = 1)
               or (year(orddate)=1993 and qtr(orddate) gt 1)
               and cust.custid = ord.custid
GROUP BY country) as list94
ON list93.country = list94.country
```

```sql
WHERE dtl.ordid=ord.ordid
and year(orddate) between 1993 and 1994;
END into :tot94
FROM db.orddtls dtl,
    db.orders ord
WHERE dtl.ordid=ord.ordid
and year(orddate) between 1993 and 1994;
SELECT SUM(CASE WHEN (year(orddate)=1993 and qtr(orddate)=1)
               or (year(orddate)=1992 and qtr(orddate) gt 1)
               and cust.custid = ord.custid
GROUP BY country) as list94
ON list93.country = list94.country
ORDER BY pctchg desc;
```
then dtl.qty*dtl.unitprc
ELSE 0
END) into :tot93
FROM db.orddtls dtl,
db.orders ord
WHERE dtl.ordid=ord.ordid
and year(orddate) between 1992 and 1993;
reset print;
SELECT cat.catname format=$15. label='Product Category',
SUM(CASE WHEN year(orddate)=1994
or (year(orddate)=1993 and qtr(orddate) gt 1)
then dtl.qty*dtl.unitprc
ELSE 0
END) as sales94 format dollar10. label='Sales Q2 93-Q1 94',
SUM(CASE WHEN (year(orddate)=1993 and qtr(orddate)=1) or
(year(orddate)=1992 and qtr(orddate) gt 1)
then dtl.qty*dtl.unitprc
ELSE 0
END) as sales93 format dollar10. label='Sales Q2 92-Q1 93',
CALCULATED sales94/CALCULATED sales93 -1 as catchg
format=percent8.1 label='Category Percent Change',
CALCULATED catches/CALCULATED petchg * 100 as index
format=8.1 label='Category Change Index to Company'
FROM db.orddtls dtl,
db.orders ord,
db.cats cat,
db.products prod
WHERE dtl.ordid=ord.ordid
and dtl.prodid=prod.prodid
and prod.catid=cat.catid
and year(orddate) between 1992 and 1994
GROUP BY cat.catname

Results

Growth in Product Categories Compared to Overall Growth

<table>
<thead>
<tr>
<th>Product Category</th>
<th>Sales Q2</th>
<th>Sales Q2 93-Q1</th>
<th>Category Percent Change</th>
<th>Company Percent Change</th>
<th>Category Change Index to Company</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beverages</td>
<td>$184,024</td>
<td>$113,172</td>
<td>62.6%</td>
<td>84.8%</td>
<td>73.8</td>
</tr>
<tr>
<td>Condiments</td>
<td>$73,010</td>
<td>$44,892</td>
<td>62.9%</td>
<td>84.8%</td>
<td>74.1</td>
</tr>
<tr>
<td>Confections</td>
<td>$112,087</td>
<td>$70,615</td>
<td>58.7%</td>
<td>84.8%</td>
<td>69.3</td>
</tr>
<tr>
<td>Dairy Products</td>
<td>$173,979</td>
<td>$86,711</td>
<td>106.0%</td>
<td>84.8%</td>
<td>125.0</td>
</tr>
<tr>
<td>Grains/Cereals</td>
<td>$70,250</td>
<td>$34,110</td>
<td>106.4%</td>
<td>84.8%</td>
<td>123.6</td>
</tr>
<tr>
<td>Meat/Poultry</td>
<td>$120,137</td>
<td>$72,652</td>
<td>65.4%</td>
<td>84.8%</td>
<td>77.1</td>
</tr>
<tr>
<td>Produce</td>
<td>$74,727</td>
<td>$31,212</td>
<td>139.4%</td>
<td>84.8%</td>
<td>164.4</td>
</tr>
<tr>
<td>Seafood</td>
<td>$105,255</td>
<td>$41,256</td>
<td>156.4%</td>
<td>84.8%</td>
<td>184.4</td>
</tr>
</tbody>
</table>

Manager says, “We have to market our product categories more aggressively. Let’s hire a consultant to do some real analysis. You provide the consultant with a table showing which products are ordered together. Then we can come up with some upelling guidelines.”

Concepts Introduced:
- Reset outobs=50
- create table
- reflexive join (table with itself)

Q10 Manager says, “I can’t wait for that consultant to finish his analysis. Just show me the things that are ordered together most often.

Concepts Introduced:
- summary functions
- embedded use of summary statistics to select records
- subquery to put grand total stats on subtotal records

Solution

libname db "C:\WINNT\Nesug97\Sasdata";
proc sql;
CREATE table db.pairs as
SELECT a.prodid as prod1,
b.prodid as prod2,
c.custid
FROM   db.orddtls a,
       db.orddtls b,
       db.orders c
WHERE  a.ordid=b.ordid
and  a.ordid=c.ordid
and  a.prodid < b.prodid
ORDER BY custid
reset outobs=50;
select * from db.pairs;
run;

Results

OBS  PROD1  PROD2  CUSTID
1    51     55     ALFKI
2    5      51     ALFKI
3    5      55     ALFKI
4    28     39     ALFKI
5    28     46     ALFKI
6    59     77     ALFKI
7    59     77     ALFKI
8    3      76     ALFKI
9    58     71     ALFKI
10   6      28     ALFKI
11   11     13     ANATR
12   11     19     ANATR
13   11     72     ANATR
14   13     72     ANATR
15   19     72     ANATR
16   13     19     ANATR
17   69     70     ANATR
18   14     42     ANATR
19   42     60     ANATR
20   14     60     ANATR
21   11     40     ANATR
22   17     53     ANATR
23   11     59     ANATR
24   43     48     ANATR
25   57     59     ANATR
26   40     59     ANATR
27   17     34     ANATR
28   34     53     ANATR
29   46     57     ANATR
30   11     57     ANATR
31   33     66     ANATR
**Concepts Introduced:**

- between
- in
- putting subtotals on detail records

**Solution**

```sql
libname db "C:\\\WINDOWS\\Nesug97\\Sasdata";
proc sql;
title "Customer Groups by Preferred Category of Product";
create view custgrps as
select gtot.custid, custtot,
sum(case when catid between 1 and 3 then qty*dtl.unitprc else 0 end) as misc,
sum(case when catid in (4,7) then qty*dtl.unitprc else 0 end) as farm,
sum(case when catid in (5,7) then qty*dtl.unitprc else 0 end) as vegan,
sum(case when catid in (6,8) then qty*dtl.unitprc else 0 end) as protein
from (select custid, sum(qty*unitprc) as custtot
from db.orders ord,
db.orddtls dtl
where ord.ordid=dtl.ordid
and dtl.prodid=prod.prodid
and catid between 1 and 3
and catid in (4,7)
and catid in (5,7)
and catid in (6,8)
group by custid) as gtot,
db.orders ord,
db.orddtls dtl,
db.products prod
where gtot.custid=ord.custid
and ord.ordid=dtl.ordid
and dtl.prodid=prod.prodid
and catid between 1 and 3
and catid in (4,7)
and catid in (5,7)
and catid in (6,8)
group by gtot.custid;
select count(*) as compns,
custgrp
from (select custid,
case when misc/custtot > .5 then "Misc" when farm/custtot > .5 then "Farm" when vegan/custtot > .5 then "Vegan" when protein/custtot > .5 then "Protein" else "Omni" end as custgrp
from custgrps)
group by custgrp;
```

**Results**

<table>
<thead>
<tr>
<th>COMPNS</th>
<th>CUSTGRP</th>
</tr>
</thead>
<tbody>
<tr>
<td>47</td>
<td>Omni</td>
</tr>
<tr>
<td>30</td>
<td>Misc</td>
</tr>
<tr>
<td>7</td>
<td>Protein</td>
</tr>
<tr>
<td>2</td>
<td>Vegan</td>
</tr>
</tbody>
</table>

Manager says, “I went to a Marketing conference and got this great idea. Let’s categorize Customers based on the products they order. Customers who buy mostly Beverages, Condiments, and Confections are junk food addicts. But we’ll call them miscellaneous. Customers who order mostly Dairy Products and Produce are farmers at heart. Customers who like Grains/Cereals and Produce are vegan. And customers who order mostly Meat/Poultry and Seafood are protein heads. How many customers do we have who fit these definitions?”
product, send them an apology and all that. First tell me what kinds of pies we sell.”

Concepts Introduced:
• like %
• UPCASE

Solution 1

```
select prod.proddesc
from db.products prod
where upcase(prod.proddesc) like "%PIE%"
```

Results 1

<table>
<thead>
<tr>
<th>PRODDESC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perth Meat Pies</td>
</tr>
<tr>
<td>Pork Pie</td>
</tr>
<tr>
<td>Shepard’s Pie</td>
</tr>
<tr>
<td>Pierrot Camembert</td>
</tr>
<tr>
<td>Sugar Pie</td>
</tr>
</tbody>
</table>

Note: The Pierrot Camembert shouldn’t be selected. Have to use the leading space on “% PIE%”

Solution 2

```
select prod.proddesc
from db.products prod
where upcase(prod.proddesc) like "% PIE%"
```

Results 2

<table>
<thead>
<tr>
<th>PRODDESC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perth Meat Pies</td>
</tr>
<tr>
<td>Pork Pie</td>
</tr>
<tr>
<td>Shepard’s Pie</td>
</tr>
<tr>
<td>Sugar Pie</td>
</tr>
</tbody>
</table>

Manager say, “Now give me a nice list of all the customers who got any of these pies since January 1. Make sure you give me addresses, phone numbers, contact names, the works.”

Concepts Introduced:

- PROC FORMS
- SAS programmers shouldn’t forget their roots in other PROCs

Solution

```
proc sql;
title 'Customers Who Received Pies after Jan. 1, 1994';
create view piecusts as
select distinct cust.*
from db.orders ord,
     db.orddtls dtl,
     db.products prod,
     db.custs cust
where ord.ordid=dtl.ordid
  and dtl.prodid=prod.prodid
  and ord.custid=cust.custid
  and upcase(prod.proddesc) like "% PIE%"
  and year(ord.shipdate) = 1994
/;
quit;
proc forms data=piecusts lines=6 nacross=2 width=40;
line 1 company;
line 2 contact;
line 3 conttitl;
line 4 address;
line 5 city region zip country / pack;
line 6 phone;
run;
```

Results

Customers Who Received Pies after Jan. 1, 1994

<table>
<thead>
<tr>
<th>Bon app’</th>
<th>Bottom-Dollar Markets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laurence Lebihan</td>
<td>Elizabeth Lincoln</td>
</tr>
<tr>
<td>Owner</td>
<td>Accounting Manager</td>
</tr>
<tr>
<td>12, rue des Bouchers</td>
<td>23 Tsawassen Blvd.</td>
</tr>
<tr>
<td>Marseille 13008 France</td>
<td>Tsawassen BC T2F 8M4 Canada</td>
</tr>
<tr>
<td>91.24.45.40</td>
<td>(604) 555-4729</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chop-suey Chinese</th>
<th>Eastern Connection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yang Wang</td>
<td>35 King George</td>
</tr>
<tr>
<td>Owner</td>
<td>London W36 6FW UK</td>
</tr>
<tr>
<td>Hauptstr. 29</td>
<td>(71) 555-0297</td>
</tr>
<tr>
<td>Bern 3012 Switzerland</td>
<td></td>
</tr>
<tr>
<td>0452-076545</td>
<td></td>
</tr>
</tbody>
</table>

| Ernst Handel       | Folk och få HB     |
| Roland Mendel      | Maria Larsson      |
| Sales Manager      | Åkergatan 24       |
| Kirchgasse 6       | Bräcke S-844 67 Sweden |
| Graz 8010 Austria  | 0699-34 67 21      |
| 7675-3625          |                    |

| Godos Cocina Tipica | Island Trading     |
| José Pedro Frayre   | Helen Bennett      |
| Sales Manager       | Marketing Manager  |
| C/ Romero, 33       | Garden House Crowther Way |
| Sevilla 41101 Spain  | Hedge End Lancashire LAX P&X UK |
| (95) 555 82 82      | (24) 555-8888      |

| Königlich Essen     | Lonesome Pine Restaurant |
| Philip Cramer       | Fran Wilson             |
| Sales Associate     | Sales Manager           |
| Maubelstr. 90       | 89                     |
| Chiaroscuro Rd.     | Portland               |
| Brandenbourg 14776 Germany | OR 97219 USA |
| 0555-09876          | 555-9573               |

| Q14                 | QUICK-Stop            |
|                     | Rattlesnake Canyon Grocery |
| Horst Kloss         | Paula Wilson          |
| Accounting Manager  | Assistant Sales Rep   |
| Taucherstrasse 10   | 2817 Milton Dr.       |
| Cunewalde 01307 Germany | Albuquerque NM 87110 USA |
| 0372-035188         | (505) 555-5939        |

| Richter Supermarkt  | Save-a-lot Markets   |
| Michael Holz        | Jose Pavarotti       |
Conclusion

The goal of this paper was to provide the reader with a practical guide to using SQL for answering some common business questions. The advanced nature of this paper allowed us to show how SQL can be used to perform some fairly complex analysis with fairly straightforward code. Our hope is that when you read this paper, it will provoke some thought on your part on how you might use SQL for more than just joins and simple queries.

Acknowledgments

We would like to give special thanks to Paul Kent and Sigurd Hermansen for their assistance in reviewing this manuscript.

Bibliography

"Fun with the SQL Procedure--An Advanced Tutorial", Lewis Church and Paul Kent, NESUG '93 p. 29

"Ten Good Reasons to Learn SAS Software's SQL Procedure", Sig Hermansen, SUGI 22, p. 180

"Introduction to Using PROC SQL” Tom Winn SUGI 22 p. 383

"SELECT ITEMS FROM PROC.SQL WHERE ITEMS > BASICS” Alan Dickson and Ray Pass SUGI 21 p. 227


Address

Questions pertaining to this article should be addressed to:
Judith F. Loren
P.O. Box 306
Cumberland Center, Maine 04021
(207) 829-6246

Gregory S. Barnes Nelson
2000 Regency Parkway, Suite 355
Cary, NC 27511
919-467-0505
919-467-2469 (Fax)
gbn@asg-inc.com
Appendix

Entity-Relationship Diagram for the Northwinds Trading Company Database.