SACRAMENTO STATE UNIVERSITY
College of Business Administration

MIS 155 – 4GL Applications

P5 - Graphics and Forms

Points: 40 Due Date: Wednesday, April 19

The objective of this assignment is to introduce you to Oracle Graphics and its integration with Forms. This assignment involves building two graphs, one showing proportion to a whole or relative frequency and the other growth or tend, and integrating them with a Form.

In contrast to tabular, master-detail and matrix reports, graphs offer a better means to convey *information* about summarized data, particularly in identifying trend and proportion. Generally, most business analyses rely on three types: line graphs, pie charts and bar (horizontal and vertical) charts. Each presents information differently. Although they are created from the same sources of data, bar charts depict frequency or relative frequency (through the height or length of the bars) while pie charts a proportion of a relative frequency to the whole. In the case of bar charts, the independent variable placed on the axis usually represents a category (i.e., nominal scale data). Measurement in terms of the quantitative units assigned to the complementary axis determines the bar's length. In contrast, the entire pie of a pie chart represents the sum of all components and the slices their relative proportions or frequencies to one another. As in the case of bar charts, categories or entities (i.e., nominal scale data) are assigned to the independent variable, the slices. The quantitative measurement (at least of interval scale), either in its raw form or as a percentage, determines the size of the slice

Line graphs allow the viewer to identify trend as in the case of time series data. Unlike bar and pie charts, the independent variable assigned to the horizontal or *x* axis must of at least ordinal scale (i.e., distinct non-interval values placed in ascending or descending sequence with the notion that one is of *greater* or *lesser* value than its predecessor). Thus, when used in time series analysis, the *x* axis would show the time periods (years, quarters, months, etc.) sequenced in their correct order. The dependent variable assigned to the vertical axis represents a measurement of at least interval scale (see your statistics textbook for further discussion).

The results of several IS behavioral studies support the use of graphical presentations to aid decision-makers during computer-aid problem solving tasks. However, different types of graphs convey different types of information.

Graph Design

Design and create two graphs in Graphics Builder following one of the three design options in Table 1. The graphs will retrieve their data from the cd_sales_00_05 table and names/titles from the music_categories <u>or</u> stores table. When selecting a graph type and subtype, follow the guidelines presented in the week09 class notes.

Table 1. Graph options

Option	Subject	Graph 1 displays trend using…	Graph 2 displays relative frequency or proportion using…
1	Stores	Annual sales revenue (in thousands) for '01 through '05 by stores[1]	Annual sales revenue (in thousands) for '01, '02, '03, '04 or '05 by (all) stores
2	Music categories	Annual sales revenue (in thousands) for '01 through '05 by music categories[2]	Annual sales revenue (in thousands) for '01, '02, '03, '04 or '05 by (all) music categories[3]

[1] Only stores with 5-year revenues greater than 14 percent of the total revenues for all 5 years.
[2] Only music categories with 5-year revenues greater than 5 percent of the total revenues for all 5 years.
[3] For pie charts place music categories or record labels that contribute less than 3 percent to the total into an "other" group.

The denormalized design of the cd_sales_00_05 table will not support a trend graph as specified in Table 1. To overcome this *problem*, you will need to create two views: the first to consolidate the five monthly revenue columns (i.e., sales_revenue_00, sales_revenue_01, etc.) into a single annual column, and the second to build a criterion data set. The data model in Graphics Builder will retrieve its data through these views using a subquery with a condition (i.e., WHERE) that incorporates the ALL option. The first view consolidates the 5 monthly revenue columns for years 2001 through 2005 into a single summarized annual column for a specific category (i.e., store, music category, record label) with a series of (SQL) unions (Figure 1). The resulting view will contain three columns: the category, year and annual sales revenue. Thus, each category will retain five rows corresponding to 2001, 2002, 2003, 2004 and 2005.

The second view builds a criterion data set (Figure 2). The data will be used to select the categories that meet the criterion specified in Table 1. The resulting view will contain 5 rows, each corresponding to 2001, 2002, 2003, 2004 and 2005 (i.e., groups). Each row refers to a percent of the total revenues for its year. The example in Figure 2 shows a calculation that establishes a 25 percent threshold (i.e., minimum value that must be met) for each year.

The data model will draw its data through a subquery that incorporates the first view in its outer query and the second for its inner query (Figure 3). Essentially, the inner query will determine which categories will be selected for the graph. It establishes a set of threshold values that a category must comply to. Because a single column and multiple rows will be returned, the subquery is referred to as a non-scalar, table type (subquery). The > ALL states that a category must exceed all values returned by the inner query to qualify (this can also be interpreted as all categories must exceed the maximum 25 percent value of the five years). (Please keep in mind this is only one approach and others may exist.) Your graph will appear similar to that displayed in Figure 4.

```
create or replace view view_name1 as
select category, to_char(sales_month_00,'yyyy') as Year,
   sum(sales_revenue_97) as Sales
   from cd_sales_00_05
      group by category, to_char(sales_month_00,'yyyy')
union
   ...

union
select category, to_char(sales_month_05,'yyyy') as Year,
   sum(sales_revenue_05) as Sales
   from cd_sales_00_05
      group by category, to_char(sales_month_05,'yyyy');
```

Figure 1. First view consolidates the 5 annual columns into a single column

```
create or replace view view_name2 as
select year, sum(sales)*.25 as threshold  ← Establishes minimum values
   from view_name1
      group by year;
```

Figure 2. Second views builds a criterion set

```
select category, year, sales/1000 ←  Reduces revenues to thousands
from view1 alias1
where exists
  (select category, min(sales)
  from view1 alias2
  where alias1.category = alias2.category
  group by category
  having min(sales) > ALL    ← ALL ensures group must exceed 25 percent set
      (select threshold from view2))   ← Subquery returns 25 percent of total sales
order by year, category;
```

Figure 3. Subquery

Specifications

Develop your report to meet the following specifications.

- Each graph should have an appropriate title, and display descriptive labels (i.e., concatenated city and store number, and category code and title) and axis titles. To reduced the clutter, remove the minor grid lines and increase the interval size for revenues (major grid lines). Be sure to state that the revenues are stated in thousands.

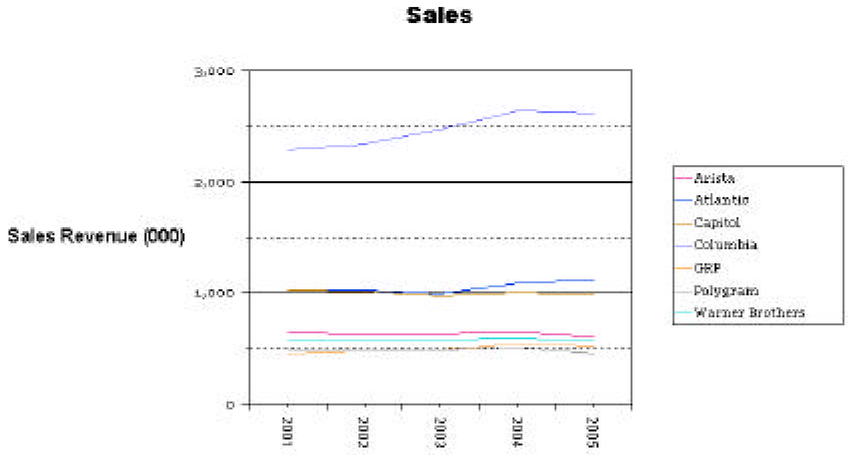


Figure 4. Line graph

- Include the percent in the slice label for pie charts.

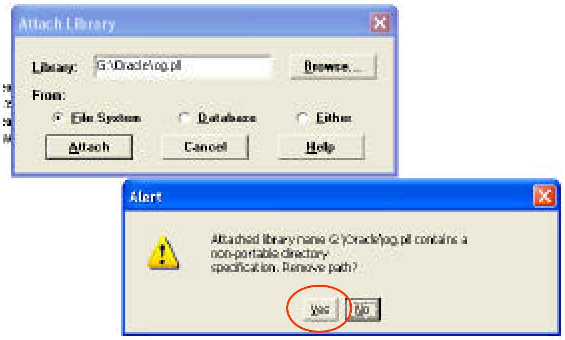
- Apply a format mask to all numeric fields.

- Apply a date mask to all date fields (when applicable). Do NOT use the default mask.

- Use a font other than the default.

- Aesthetically arrange the chart items on the form module and the graphs within the chart items.

- Do NOT include the directory path to the OG library file (og.pll) when attaching it to your form module. Select ***yes*** to removing the path after clicking on *Attach*. Saving the path makes your form non-portable since your form will follow the path and search the specified directory for the library file when it is executed. If no such path exists, the form will not be able to open the graph. Thus, to overcome this problem, copy the og.pll file to your \orawin95\bin subdirectory or working (*start in...*) directory; these directories have defined paths.



Library: G:\Oracle\og.pll

From: ● File System ○ Database ○ Either

Attached library name G:\Oracle\og.pll contains a non-portable directory specification. Remove path?

- Use your view graphs form module from P1. Be sure its canvas' properties and menu module are inherited per the specifications of P1. Change the background color of the chart item to match the canvas', thereby giving it an illusion of transparency.

- Use push buttons to launch the graphs in a single chart item on your form module via a *Smart trigger* (i.e., one button launches the trend graph and another the proportion/relative frequency graph). In your PL/SQL code, be sure to follow the *og.open* with an *og.close* (refer to the week10.ppt for the syntax).

Note. When designing your report, use your ***common sense***. If ***you*** have difficulty understanding your graphs, the user certainly isn't going to fare any better! Remember, the purpose of the reports is to convey information, *half* of which is the way it is presented.

Warning! Following the ScreenCam presentations and the examples used in the class notes too closely will be a detriment to your grade. *Be creative!*

Tangibles

Place the following in a 9 × 12-inch manila envelope:

- Printout of your screen capture showing the executed form with the two graphs. **Specify the option** (in Table 1) you are following.
- 3½-inch diskette, CD or USB flash memory with your menu, form, graph and og.pll files.
- A text file (txt) containing the SQL CREATE VIEW commands for the two views referenced in your graph's data model (this requirement is similar to that in P2). They will be used to recreate your views and test your assignment. Please do NOT include the SQL Plus prompts. Otherwise, you assignment cannot be graded.

Please do ***NOT***:

- Include files other than those necessary to run your assignment (form modules for the greeting screen, main menu and graph, menu module, graphic modules).
- Create subdirectories on your diskette. All files should be in its root directory. Otherwise, your assignment cannot be tested.

Write your name on all material submitted in your envelope.

As stated in the syllabus, assignments are due at the beginning of class. No late assignments will be accepted.

Be sure to make a backup copy of your assignment. The copy you submit should NOT be your only copy!

Warning! Check your diskette for viruses. Any diskette, CD or flash memory with a virus will not be graded.