Datamart & Hyperion: Back to Basics

Version 11
Updated: 6/18/2012
About the Department of Administrative Services (DAS) – Datamart Team:

The Department of Administrative Services (DAS) Datamart team provides courses with a focus on helping our customers improve their financial and technological skills.

The training materials are for the students’ use both during the course and as a reference guide after the course is complete.

For more information and a list of other available products, please visit us at our website at [http://egov.oregon.gov/DAS](http://egov.oregon.gov/DAS).

Copyright & Trademarks

Copyright © 2011 by DAS Datamart Team. All rights reserved.

Trademarked names appear throughout this book. Rather than list the names and entities that own the trademarks or insert a trademark symbol with each mention of the trademark name, DAS Datamart team states that it is using the names for editorial purposes and to the benefit of the trademark owner with no intention of infringing upon the trademark.

Disclaimer

The DAS Datamart team has taken every effort to ensure the accuracy of this manual. If you should discover any discrepancies, please notify us immediately.

Department of Administrative Services
155 Cottage St., NE, Salem, OR 97310
SARS: (503)373-0269 – Fax: (503)378-3514
E-mail: Datamart.Support@state.or.us
Lesson 1

Datamart Basics

✓ What is a Datamart?
✓ How does information get into the Datamart?
✓ How can I get current information about the Datamart?
✓ History of Brio/Hyperion
✓ Am I ready to work in Hyperion?
✓ The Repository
What is a Datamart?

The Department of Administrative Services (DAS) financial Datamart is your one-stop warehouse for financial transactions, account balances, grant profiles, PCA information, Payroll data and more, which helps employees develop reports for business information and decision making.

Many years ago, the Department of Administrative Services (DAS) State Controller's Division (SCD) sponsored the creation of a Datamart. This initiative brought data from the primary transactional accounting system (SFMA) and the payroll system (OSPA) into a relational database for various agencies and users to easily query against for information using a reporting tool. Recently, Position Information Control System (PICS), PPDB and ORBITS (Budget) data have been added to this Datamart, and many other transactional applications have been created that capture relevant information that users can use to make better business decisions.

The Datamart provides a resource for easy reporting.

In 1997, the SCD developed a Datamart of accounting data from SFMA to meet the ad-hoc reporting needs of state agencies. This became known as the SCD Datamart and later known as the SFMA Datamart. In 1999, the SCD developed a Datamart of payroll data from OSPA, which became known as the OSPA Datamart. The number of active Datamart users has expanded over the past several years, from 145 users in 1999 to an average of 360 users per month in 2007 to over 800 users in 2012.

Many agencies are using these data sources on a regular basis to meet individual agency reporting needs. Some agencies use the Datamart to replace system-generated mainframe reports. This provides agencies with more meaningful, custom designed reports and saves time since data does not need to be re-keyed from hard-copy reports for various analyses.

The Datamart gives the user the ability to create custom reports.
How does information get into the Datamart?

*The DAS Datamart contains data from the following statewide systems:*

**SFMA (Statewide Financial Management Application)**

SFMA data is updated weekly with a download Friday evening and an upload on Saturday evening.

Mid-July through August (considered Period 13) the YE GL Detail, YE GL Summary, and Profile tables are updated three times per week from SFMA.

**OSPA (Oregon Statewide Payroll Application)**

The payroll or OSPA data is updated monthly after the Final Run 2, approximately the 10th of each month. The data is available the morning after Final Run 2 is posted to R*STARS (see the OSPS Processing Calendar for specific dates throughout the year).

**ORBITS (Oregon Relational Budget Information Tracking System)**

The Orbits or budget data is loaded weekly along with the normal SFMA load process.

**PPDB (Position & Personnel Database)**

The PPDB data is loaded the first of every month.

**PICS (Position Information Control System)**

The PICS information is loaded weekly along with the normal SFMA load process.
How can I get current information about the Datamart?

If you have any questions, suggestions, problems, or just plain comments please contact the Datamart Business Analyst listed below.

Datamart Business Analyst
155 Cottage Street NE, U50
Salem, OR 97301-3969
Phone: (503) 373-0269
Fax: (503) 378-3514
E-mail: Datamart.Support@state.or.us

Datamart Website

The SARS unit has a website dedicated to providing valuable information about the Datamart. Please visit http://egov.oregon.gov/DAS/SCD/SARS/datamart.shtml

Datamart News List

To join the Datamart News List, please visit the link above and click on Datamart-News. This brings you to the E-Communications site. Follow the instructions and soon you will be receiving important communication from the Datamart team as well as other Datamart users.

Datamart User Group

The SARS unit hosts the Datamart User Group. The goal of this group is to decide on the needs and direction of the Datamart. Please visit the web link above for meeting dates and times.

Newsletter

The SARS unit publishes a newsletter, on a varying basis, called the “Datamart Connection”. This publication is sent out via the Datamart News list and published to the Datamart website for easy access.
History of Brio/Hyperion?

Since the Datamart’s creation in 1997, the State of Oregon has come to rely on the Brio/Hyperion query tool as its primary means of retrieving accounting and other financial information from the Datamart.

The original Brio application was created by Brio Software, Inc. and was used extensively throughout the state to access the Datamart. However, in 2003, Brio was acquired by Hyperion Solutions Corporation. Hyperion continued to update the software until March 2007, when Hyperion was acquired by Oracle. Throughout these transition years, the State of Oregon continued to use the product. In May 2010, the State of Oregon purchased an enterprise license for ‘Oracle Enterprise Performance Management System 11’ also known as Hyperion Interactive Reporting Studio (aka IR Studio). This purchase allows the state to use the Brio/Hyperion/Oracle product for years to come.

Am I ready to work in Hyperion?

If you can answer “YES” to all the questions below, you are ready to work in Hyperion.

- Are you a state employee?
- Do you have a RACF ID?
- Do you have an IBM DB2 Open Database Connection (ODBC)?
- Do you have security access to the Datamart?
- Is Hyperion installed on your PC?
- Have you set up an Open Catalog Extension (OCE) for Hyperion?
- Have you completed Datamart training?
Exercises

1. What is the primary function of the Department of Administrative Services (DAS) Financial Datamart?

2. How often is the SFMA, OSPA, and PICS data updated with new information on the Datamart?

3. Name 3 resources for Datamart information:

4. What are 4 things I need before I can access the Datamart?

5. What is the name of the newest Brio/Hyperion version?
Lesson 2

OCE’s and Datamart Tables

✔ What is an Open Catalog Extension (OCE)?
✔ Setting up an SFMA Datamart OCE
✔ Setting up an OSPA Datamart OCE
✔ Modifying an OCE
✔ Tables on the Datamart
  o SFMA Financial Tables
  o General Ledger Detail Tables
  o General Ledger Summary Tables
  o Other Special Financial Tables
  o OSPA Datamart Tables
✔ The Repository
What is an Open Catalog Extension (OCE)?

In Hyperion, you use an OCE whenever you perform tasks that require you to connect to a database. The OCE is simply a definition of the connection required to view a particular database.

When you open Hyperion to begin a work session (as shown below), you must select the correct OCE for the targeted database. You can select from the list of Recent Database Connection Files or Browse your directory.

Your Recent Database Connection Files list will probably have different names compared to the above diagram because you are able to change the title of your OCE’s. Standard practice is to title the OCE whatever name helps you associate with the database you will access.
Setting up a SFMA Datamart OCE

Creating an Open Catalog Extension (OCE) in Hyperion Interactive Reporting Studio - Version 11

These are the detailed instructions to be used while setting up your Hyperion SFMA OCE. In these instructions, you will connect to the SFMA Datamart only.

1) Open Hyperion and you should see a “Welcome to Hyperion” dialog box. Select **A New Database Connection File**
Select **OK**
2) This takes you to the “Database Connection Wizard”
Select **ODBC** from the What connection software do you want to use?
Select **DB2** from the What type of database do you want to connect to?

Check the following two boxes:
1. **Show Meta Connection Wizard**
2. **Show advanced options**

Select **Next**
3) In the box labeled “User Name” type your Datamart User Name (this is your RACF ID).

In the box labeled “Password” type your Datamart Password (not your SFMA password).

From the dropdown list select SFMSP

Select Next
4) Leave this screen just as it is shown with “On the current connection” marked. Click ‘Next’, to continue.

5) On this “Meta Connection Wizard” screen select → Edit
6) This next screen is the “Meta Data Table Definition”
Select the following tab from the top→**Joins**

7) Select→**Custom**
The grayed out fields now become white for data entry.
8) Fill in the fields accordingly:

- **Primary Owner**: PKEY_OWNER
- **Primary Table**: PKEY_TABLE_NAME
- **Primary Column**: PKEY_COLUMN_NAME
- **Foreign Owner**: FKEY_OWNER
- **Foreign Table**: FKEY_TABLE_NAME
- **Foreign column**: FKEY_COLUMN_NAME
- **From**: SFMSPUSR.SFMS_WHSE_JOINS
- **Where**: PKEY_OWNER=:OWNER

Select **→ OK**
Then select **→ Next** to continue
9) Check the boxes for “Allow SQL-92 Advanced Set Operations” and “Exclude Hyperion Repository Tables”.

Next, select the middle **Define** button next to ‘Owner Name’.
10) This takes you to a screen where you can choose the tables you want in your OCE.

Select \textbf{Show Values}

Then, you can select the table names you want. You can select more than one by using \textbf{“CTRL+Click”}.

Select the following tables:
- SFMSPUSR \rightarrow SFMA Accounting Tables (no archived data)
- SFMSARCH \rightarrow SFMS Archive Tables
- YEAREND \rightarrow Year-End allows access to the Repository

Select \textbf{OK}

Then select \textbf{Next} to continue.

1. \textbf{Show Values, and get your tables!}

2. \textbf{OK}

3. \textbf{Next}
11) Update the next two screens as they appear below. Make changes as necessary.

Select **Next** to continue on both screens.
12) Select **Finish** starting the completion of this process

13) Select **Yes** to save your OCE
14) Give your OCE a name in the **File Name** field

Select → **Save** to finalize the saving of your OCE

You can save Open Catalog Extensions in any directory but best practice is to save to the Standard Hyperion directory. This allows for easy access to all OCE files.

The standard SFMA OCE is downloadable from the Datamart website. [http://dasapp.oregon.gov/datamart/](http://dasapp.oregon.gov/datamart/)
Setting up an OSPA Datamart OCE

Follow the same steps as the SFMA Datamart OCE process except replace step 10 above with the following step 10.

10) This takes you to a screen where a selection of tables must be decided for the OCE.

Select ➔ Show Values

Then you can select the table names you want. You can select more than one by using “CTRL+Click”.

Select the following tables:
OSPSPUSR
YEAREND

Select ➔ OK
Then select ➔ Next to continue

1. Show Values, and get your tables!

The standard OSPA OCE is downloadable from the Datamart website. http://dasapp.oregon.gov/datamart/
Modifying an OCE

When you create an OCE, you establish a working database connection for querying. You may need to modify an OCE to reflect changes in your network or hardware configuration, or to manage other connection information.

To modify an OCE:

1) Close any open Hyperion documents.

2) Select → Tools → Connection → Modify

The Modify Connection dialog box appears.

3) Select the connection file you want to modify and select → Open

The Database Connection Wizard appears and displays the information for the OCE you selected.

4) Make any desired changes and then save the OCE when prompted.

Have fun working in Hyperion! You are now ready to create queries and run reports.
Tables on the Datamart

There are two types of tables on the Datamart: Financial tables and Profile tables. The main difference between the two is the financial tables contain financial information, such as General Ledger balances, and transaction amounts. Using the profile tables provides titles for various data elements, or allows a roll-up of information to a higher level than the financial table can accomplish.

Note that if you want detail finer than what is on any of the financial tables, you are not going to have much luck. The Datamart and the query tools can only slice and dice the data as deeply as the data goes. Say you want to get a subset of your comptroller objects. If you do not have any way to tie that subset together using data on the Datamart, you are going to be stuck (likely) having to go to the detailed records and the original documents. The following is an example of ‘Meals and Lodging’. If you use comptroller object only (no agency object), you will not be able to get a report out of SFMA giving you a breakdown of Meals costs separate from Lodging. You might be able to get something based on vendor, but you also might not.

SFMA Financial Tables

Accounting Event Tables

**Active Accounting Event** This table mirrors the data from the SFMA system. Data is reflective of the current 3 months of active transactions.

**All Accounting Event** This table mirrors the data from the SFMA system. Data is reflective of the current fiscal year and five prior years of transactions. This table may contain up to two years of addition information depending on the purge timetable.

**Archive Accounting Event** This table mirrors the data from the SFMA system. Data is reflective of the current fiscal year and seven prior years of transactions. This table may contain up to two years of addition information depending on the purge timetable.

The accounting event tables are the lowest level of detail in the Datamart. All other financial tables are summaries in some way of this information.
One thing to be aware of is that the date fields in the Datamart are actual date fields. Date fields in SFMA are text fields. While those fields are edited to assure proper dates are being used, this allows SFMA to contain dates not on the calendar. Period 13 identified through effective date is the best example of this. In SFMA period 13 is identified as 6/31/2010. In the Datamart, date fields are not allowed to have dates not on the calendar—data validation feature in DB2. In order for us to move data to the Datamart, we have to do some “data washing”. Dates like 6/31/2010 are changed to 6/30/2010, a new data element is introduced called Batch Date Ci. The download program picks up the last digit in the month and records it in this field. This way, any time an incorrect date is entered into SFMA, it is corrected in the Datamart, and an indicator is provided.

Note though, another data field in the Accounting Event tables will provide you the fiscal month. The Fiscal Month field identifies the appropriate fiscal month, including month 13.

**General Ledger Detail Tables**

**General Ledger Detail** This table mirrors the data from the SFMA System. This table summarizes General Ledger balances, and summarizes at the object and agency object level.

**YE General Ledger Detail** This table mirrors the data from the SFMA system. However, the data on this table is updated only during month 13 close. During the period of mid-July to close in August, this table is updated with the balances in the current fiscal year only. At other times of the year, this table contains zero data. This table summarizes General Ledger balances, and summarizes at the object and agency object level.

**Arch GL Detail** This table mirrors the data from the SFMA system. The period available is dependent on the date of agency implementation onto SFMA.
**General Ledger Summary Tables**

**General Ledger Summary** This table mirrors the data from the SFMA system. This table summarizes General Ledger balances, but does not contain any object level detail.

**YE General Ledger Summary** This table mirrors the data from the SFMA system. However, the data on this table updates only during month 13 close. During the period of mid-July to end of August, this table updates with the balances in the current fiscal year only. At other times of the year, this table does not contain data. This table summarizes General Ledger balances, but does not contain object level detail.

**Arch GL Detail** This table mirrors the data from the SFMA system. The period available is dependent on the date of agency implementation onto SFMA.

The General Ledger Tables are a higher level of summarization in the Datamart from the Accounting Event tables. Balances for the tables are by month, and are cumulative by month. This means that a balance in **Month 03 Balance** will contain the total of months one, two and three in that field. Monthly information for nominal GL accounts, except for month just closed, must be created using calculated fields.

**Other Special Financial Tables**

**Appropriation Financial Table**

The Appropriation Financial table mirrors the data from the SFMA system. This provides information similar to the “62” screen on SFMA.

**Contract Financial Table**

The Contract Financial table mirrors the data from the SFMA system. This provides information similar to the “68” screen on SFMA.
Grant Financial Table

The Grant Financial table mirrors the data from the SFMA system. This provides information similar to the “66” screen on SFMA.

Project Financial Table

The Project Financial table mirrors the data from the SFMA system. This provides information similar to the “80” screen on SFMA.

Receipt Extr Sum Table

The Receipt Extr Sum table mirrors the data from the SFMA system. This provides information similar to the “12/13” screen on SFMA.

Document Financial Tables

Doc Financial Table→ mirrors the data from the SFMA system. This provides information similar to the “64” screen on SFMA.

Arch Doc Financial Table→ mirrors the data from the SFMA system. This provides information similar to the “64” screen on SFMA.

Payment Control Table

The Payment Control table mirrors the data from the SFMA system. This provides information similar to the “47” screen on SFMA.
Profile Tables

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Screen Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agency</td>
<td>D02 Screen</td>
</tr>
<tr>
<td>Agency Code 1</td>
<td>D26 Screen</td>
</tr>
<tr>
<td>Agency Code 2</td>
<td>D27 Screen</td>
</tr>
<tr>
<td>Agency Code 3</td>
<td>D36 Screen</td>
</tr>
<tr>
<td>Agy GL</td>
<td>D32 Screen</td>
</tr>
<tr>
<td>Agy Obj</td>
<td>D11 Screen</td>
</tr>
<tr>
<td>Agy Obj Grp</td>
<td>D25 Screen</td>
</tr>
<tr>
<td>Appn No</td>
<td>20 Screen</td>
</tr>
<tr>
<td>Appr Fund</td>
<td>D22 Screen</td>
</tr>
<tr>
<td>Budget Obj</td>
<td>D01 Screen</td>
</tr>
<tr>
<td>Cash Fund</td>
<td>D73 Screen</td>
</tr>
<tr>
<td>Compt Obj</td>
<td>D10 Screen</td>
</tr>
<tr>
<td>Compt Srce Grp</td>
<td>D09 Screen  (this is now a tie to ORBITS data)</td>
</tr>
<tr>
<td>Contract</td>
<td>30 Screen</td>
</tr>
<tr>
<td>Fund D23</td>
<td>D23 Screen</td>
</tr>
<tr>
<td>GAAP Fund</td>
<td>D24 Screen</td>
</tr>
<tr>
<td>GL Acct</td>
<td>D31 Screen</td>
</tr>
<tr>
<td>Grant No</td>
<td>D47 Screen</td>
</tr>
<tr>
<td>Grant Obj</td>
<td>D48 Screen</td>
</tr>
<tr>
<td>Grant Phase</td>
<td>29 Screen</td>
</tr>
<tr>
<td>Index No</td>
<td>24 Screen</td>
</tr>
<tr>
<td>Operator ID</td>
<td>D96A/B Screen</td>
</tr>
<tr>
<td>Org Code</td>
<td>D03 Screen</td>
</tr>
<tr>
<td>PCA</td>
<td>26 Screen</td>
</tr>
<tr>
<td>PCA Index Rel</td>
<td>21 Screen</td>
</tr>
<tr>
<td>Program Code</td>
<td>D04 Screen  (this now contains the ORBITS Cross-Reference Number)</td>
</tr>
<tr>
<td>Project No</td>
<td>D42 Screen</td>
</tr>
<tr>
<td>Project Phase</td>
<td>27 Screen</td>
</tr>
<tr>
<td>State Fund Grp</td>
<td>D20 Screen</td>
</tr>
<tr>
<td>Sub Grantee</td>
<td>31 Screen</td>
</tr>
<tr>
<td>TDate</td>
<td>Generated – not on SFMA</td>
</tr>
</tbody>
</table>
**OSPA Datamart Tables**

The OSPA tables on the Datamart are not as complex as the SFMA tables. These tables are each a flat-file database made from report output. This does make querying a bit more difficult, as you really have to know exactly what you are looking for, and where to find it.

**OSPA DED Tables**
OSPA Deductions
OSPA Ded - Current Biennium
OSPA Ded B - Current and Prior Biennium
OSPA Ded PB - Prior Biennium
OSPA Ded PP - Prior-Prior Biennium (2 prior biennia)

**OSPA JS Lab ADJ Tables**
OSPA Job Status, Labor and Gross adjustments
OSPA JS Lab ADJ - Current Biennium
OSPA JS Lab ADJ B - Current and Prior Biennium
OSPA JS Lab ADJ PB - Prior Biennium
OSPA JS Lab ADJ PP - Prior-Prior Biennium (2 prior biennia)

**OSPA Labor Cost Tables**
Labor Costs
OSPA Labor Cost - Current Biennium
OSPA Labor Cost B - Current and Prior Biennium
OSPA Labor Cost PB - Prior Biennium
OSPA Labor Cost PP - Prior-Prior Biennium (2 prior biennia)

**OSPA LV Accrual Tables**
Leave Accruals
OSPA LV Accrual - Current Biennium
OSPA LV Accrual B - Current and Prior Biennium
OSPA LV Accrual PB - Prior Biennium
OSPA LV Accrual PP - Prior-Prior Biennium (2 prior biennia)

**OSPA Net Pay Tables**
Net Pay amounts
OSPA Net Pay - Current Biennium
OSPA Net Pay B - Current and Prior Biennium
OSPA Net Pay PB - Prior Biennium
OSPA Net Pay PP - Prior-Prior Biennium (2 prior biennia)
**OSPA Work Sched Table**

Work Schedule amounts

Note: When querying the OSPA tables, you must use a five-digit agency number in order to obtain the desired results. For example, DAS is agency 107 for SFMS, but 10700 for OSPA purposes. Another example is Parks & Recreation’s agency number on SFMS is 634, but on OSPA, it is 63400.

**The Repository**

The ‘Repository’ is a great tool to help share standard reports with users on a statewide level. The purpose of sharing these reports is to allow agencies to have an outlet to access reports created by different agencies. The Datamart team cannot guarantee these reports will function properly for each agency, due to specific agency structure setups; however, please feel free to make modifications to suit your agency.

The Repository contains many files, which access data from SFMA, ORBITS, OSPA, etc. Thanks to users throughout the state, we continually obtain more reports to help with our daily operations.

Below are instructions for accessing the Datamart Repository.

1. Open Oracle Hyperion Interactive Reporting Studio (Formerly known as Brio). Click **Cancel** if you get a ‘Welcome’ dialog box requesting to open files or open a database connection.

2. Select **File > Open from Repository > Select**. Select the Open Catalog Extension (OCE) that connects to the Repository.

3. Select the proper OCE and click **Open**. In this example, the OCE is called “Datamart”. Your OCE name may be different. You should see a screen similar to the following:

   There are special view tables that agencies request be created to help lock or unlock specific data within the Datamart.
4. You will be prompted for your user name and password. Input your information and click **OK**. You should see a screen similar to the following:

5. If no errors have occurred with your log in data, you should see a screen similar to the following:
If your OCE is set up to go directly to the “YEAREND” directory, instead of seeing the above info, you might see something similar to the following:

Click on the + next to “YEAREND”. Your screen should look like the following:

6. Click on the + next to “Standard Query with Reports”. Your screen should look like the following (Again this may be somewhat different depending on your OCE setup; you may not see anything but the three folders “Data Model”, “Standard Query” and “Standard Query with Reports”):
7. Scroll down until you find the report you want to use. Once selected a description field for the file will be shown on the right side. Click Open. Your screen should look like the following:

Once the Repository file is open, please review the Dashboard information contained in the file, prior to running the query.
Exercises

1. What is an OCE?

2. How many OCE’s can I set up on my machine?

3. Can I modify an existing OCE on my machine?

4. Name the 2 types of tables on the Datamart.

5. Name 3 financial tables found in the Datamart.

6. Name 3 profile tables found in the Datamart.

7. Name 3 OSPA tables found in the Datamart.

8. Name a resource, which provides a great tool to help share standard reports with users on a statewide level.
Lesson 3

Hyperion Queries

✓ Hyperion Spider Web
✓ BQY Documents, Sections, and Toolbars
✓ Online Help
✓ Opening and Saving BQY Documents
✓ Processing and Stopping Queries
✓ Adding, Duplicating, Renaming, and Deleting BQY Sections
✓ Hiding a BQY Section
✓ Sending BQY Documents as Email Attachments
✓ Printing BQY Sections
  ◦ Adjusting the Page Margins in Print Preview
  ◦ Adding Headers and Footers
  ◦ Printing Sections
Hyperion Spider Web

Let’s talk about the flow of data from the financial world to the Datamart. How does it get to your machine and out on a piece of paper for your boss or meeting?
BQY Documents, Sections, and Toolbars

BQY Documents

A BQY document is a reporting file created in Hyperion and has the file extension of .bqy. A .bqy can contain queries, results, tables, pivots, charts, reports, and dashboards.

BQY Sections

Queries→ Used to create questions against the Datamart.

Results→ Used to display the rows returned/retrieved from a query built in the query section. You can take the results of the query and develop filters, sorts, computed items, and format.

Tables→ Used to create organized subsets of your query results. These are tabular-styled simple reports. You can develop filters, sorts, computed items, and format this data.

Pivots→ Used to create cross-tabular style reports in which you can analyze data. You can focus or drill-down on data items in this area. Data facts are aggregated (combined) in this area.

Charts→ Used to create horizontal and vertical bar charts, line, pie, cluster, scatter, and ribbon. You can format areas of your charts.

Reports→ Used to develop free-form reports where multiple result sets are displayed in a combined area. This is where you can integrate foreign results that have been imported into your .bqy with current information from the Datamarts.

Dashboards→ Used to create an Executive Information System (EIS), which offer a user-friendly approach to querying and manipulating results, pivots, charts, and reports. Users are able to check boxes, access drop-down lists, select radio buttons, and filter results based on their specific needs and criteria.
BQY Toolbars

The following figure shows the sections and toolbars of a Hyperion .bqy.

Your Section and Catalog (or Elements) Pane will vary depending upon the .bqy you are developing or using. The Section Toolbar, which is not shown, displays when you are working in Charts, Reports, and Dashboards.
If you can't visually see a toolbar or section, you can select **View** from the **Menu Bar** and choose what you would like to see.

You can view or hide sections and toolbars by selecting:

**View** ➔ **Toolbars**

Selecting a section from the View menu list will display or hide that section.
Online Help

Online help is available through Hyperion.

To access help:

From the Menu Bar (Note: Must have bqy started)

Help → Help on Content

Or

Select the ‘F1’ key

This will take you to Hyperion online help. You can search for help by three methods:

- **Contents** → Allows the user to search for help on a pre-defined topic
- **Index** → Search for help on predefined topics alphabetically
- **Search** → Type your help issue in the search box for results
Opening and Saving BQY Documents

A .bqy document is just like any other document you would create, edit, save, close, or open; only it lives in Hyperion. The .bqy is the file extension for a Hyperion document.

Opening a BQY Document

You can double-click on an existing .bqy to open it or you can go through Hyperion. Below is an example of the opening dialog box you will see when you open Hyperion.

You can choose a Recent Document from the bottom half of this screen or open an OCE from the upper half of this screen.
Starting a new .bqy using the SFMS.oce:

Open Hyperion from your Start Menu

Select → Browse from Recent Database Connection File

Select → SFMS.oce → OK

Type → User Name and Password (this is your assigned Datamart user name and password)

You should now see the following screen:

You may have to click on the + next to Tables to see all the topics available to use.
You can also right-click on top of the tables to see their full names. This shows you what view or tables these topics are linked.
Saving a BQY Document

Just like any other application, you have several choices for saving your .bqy.

You can save a .bqy by selecting:

**File ➔ Save**
This saves the .bqy under the existing name and updates any changes you have made to the opened document. This does not create a new .bqy document unless it is newly created.

**File ➔ Save As**
This takes an existing .bqy document and saves it as a new document with the changes you have made. Use this option when saving a .bqy for the first time or when you desire to keep the existing .bqy document and a new .bqy document.

**File ➔ Save Options ➔ Save Query Results With Document**
Use this if you want to perform off-line data analysis of your .bqy.

**File ➔ Save Options ➔ Compress Document**
This features results in smaller file sizes and enables you to send your file quickly.

**File ➔ Save Options ➔ Password Protect Document**
Users will need a password to access this document. Be careful! You need to remember your password.

**File ➔ Save Options ➔ Password Protect Design Mode**
Password protection used to guard Dashboard designs.
Processing and Killing Queries

Processing Queries

This process refreshes the data in the .bqy document based on your requested items and filters specified. When a query is processed Hyperion goes back to the Datamart and retrieves this information for your query. The process time depends on the data you are asking the Datamart to retrieve and the power of the Datamart server and connection to the server.

Because your .bqy document could have more than one query, there are three process options to choose from:

- **Process Current** → Processes the query that is currently active.
- **Process All** → Processes all queries that exist in the .bqy document.
- **Process Custom** → Opens a Process Custom dialog box so you can choose which queries to process. Simply put an X in the box next to the query.

Process Order of Queries

It is important to remember how you develop your queries. If you create a .bqy with multiple queries and some of those queries depend upon others in that .bqy, you will want to make sure you order them to process in a certain order.
Tools → Process Query → Process Order

Prior to processing a query, you must complete the following steps:

*Add a ‘Table’ to your query section:*

Bring a Table over from the Elements Pane to the Content Pane. You can drag and drop or double-click.
Datamart & Hyperion: Back to Basics  
Lesson 3  
Hyperion Queries

Add fields in a Table to the Request line:

**All Fields:** Right-click ➔ Table Name ➔ Left-click ➔ Add Selected Items  
**Individual Fields:** Right-click ➔ Field Name ➔ Left-click ➔ Add Selected Items

Create a Filter for your query:

Double-click ➔ Field Name. A ‘Filter’ dialog box pops up.

Manually input a value in the ‘Custom Values’ section and Click ➔ OK or; Click ➔ Show Values ➔ Select an option ➔ Click ➔ OK

Select ➔ from the Standard Toolbar

You will now see your results under the Results section of the Section Pane.
Stopping Queries

Sometimes when you process a query or queries in a .bqy document you may find the data is not being returned from the Datamart in a timely manner. Users want to be conscious of their query/process time to keep up top performance. It is important that you use the process listed below to stop the query. Exiting out of Hyperion does not stop the query process to the Datamart and it will continue to use resources to process the query.

To stop a query:

Open the Internet
   Go to ➔ http://dasdm1.iservices.state.or.us/
Log in using your Datamart user name and password
   Select ➔ List or Kill Queries
   Select ➔ Kill Queries

Sometimes after you stop a query, you may notice a ‘Results’ set of data is returned. This is called ‘partial data’ and you must be careful because these results are not complete.

You can also stop a query once it starts providing results.
   Hold down the ‘alt’ key along with the ‘End’ key
Adding, Duplicating, Renaming, and Deleting BQY Sections

You can add, duplicate, rename, or delete a .bqy section at any time.

To add a new section:

Insert ➔ New ➔ Select the Section you want to add

Your choices are:

To duplicate a new section:

Edit ➔ Duplicate Section ➔ Right-click on the Section you want to duplicate
To rename a section:

**Edit ➔ Rename Section**

Right-click on the Section you want to rename

To delete a section:

**Edit ➔ Delete Section**

Right-click on the Section you want to delete
In Version 11 of Hyperion you cannot re-arrange the sections of your .bqy.
Hiding a BQY Section

Often it is helpful to hide a section on the Section Pane for security reasons or making your .bqy more user friendly for the end user.

To hide a section:

Select View→Hide Section

To show your hidden section:

Select View→Unhide Section  
Select→Query→OK
Sending BQY Documents as Email Attachments

Hyperion Interactive Reporting Studio has the capability of sending .bqy as email attachments.

To send a .bqy as an email attachment:

Select **File ➔ Send**

![Mail Document window]

- Use To, CC, Bcc, and Subject the same way you do in your regular mail program.
- Notice you can see the name of your .bqy in the middle of the screen.
- Select Send Results to email the results section with your email.
- Select Send Compressed to zip your file prior to email.
- Notice you can also type an Additional Message.

Send it!
## Printing BQY Sections

### Adjusting the Page Margins in Print Preview

When your .bqy is complete, you may want to view your finished product and adjust the margins, add page numbers, and headers and footers before to printing or emailing. Prior to Print Preview, you must select the section you want to preview in the Section Pane. **Please note:** The Report & Dashboard Sections do not have a Print Preview mode.

**To Print Preview a section:**

**File ➔ Print Preview**

**To adjust the page margins:**

Just put your mouse pointer over the dashed margin lines and click and drag to get your desired margins. You can do this both vertically and horizontally.

<table>
<thead>
<tr>
<th>Fund</th>
<th>Gl Acct</th>
<th>M13 Balance (O)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>2</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>3</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>4</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>5</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>6</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>7</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>8</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>9</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>10</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>11</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>12</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>13</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>14</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>15</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>16</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>17</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>18</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>19</td>
<td>250</td>
<td>0065</td>
</tr>
<tr>
<td>20</td>
<td>250</td>
<td>0065</td>
</tr>
</tbody>
</table>
Adding Headers and Footers

Adding a Header or Footer to your .bqy section gives you the opportunity to add page numbers, file names, date and time, and query filters.

To add a Header or Footer:

Select Insert ➔ Page Header or Page Footer

Printing Sections

You can print any section of your .bqy.

To print a Section:

Select File ➔ Print

You can select the correct printer, printer properties, number of copies, and range you would like to print.
Exercises

Name the 6 sections you can create in a .bqy.

a. 

b. 

c. 

d. 

e. 

f. 

1. Start a new .bqy using the SFMS.oce.

2. Expand the Table list on the Element Pane.

3. Bring the ‘GL Detail’ Table over to the Content Pane.

4. Sort the table fields alphabetically. (Hint: Right-click on the title of the table and select properties).

5. Add the following Items to the Request Line:
   a. Agency
   b. GL Acct
   c. Fund
   d. Fiscal Year
   e. M13 Balance GL

6. Filter the following Items:
   a. Agency – 585
   b. Fiscal Year – 2010
   c. ‘M13 Balance GL’ not equal to zero
7. Process the query. Does your query look similar to the examples below? How many rows were returned? (Answer: 81,773)

8. Rename your **Query** section as “Query - GL Detail 10” and rename your **Results** section as “Results - GL Detail 10”.

9. Duplicate your “Query - GL Detail 10” section. Notice it automatically alters the name of the new sections.

10. Add a page header to your **Results** section called “Results - GL Detail 10”.

11. Alert the instructor after correctly completing the exercise.
Lesson 4

Hyperion Query Building Processes

✓ BQY Process
✓ Managing your Table
  o What can I do to the Table Properties?
✓ Understanding Joins
✓ Building the Request Line
✓ Maximizing the Query Building Process
  o Setting Restrictions on your Query Properties
  o Estimating the Size of your Query
**BQY Process**

BQY documents are files you create and use to retrieve information from a database, analyze the information, and build reports. Because Hyperion is an integrated query, analysis, and reporting tool, documents have multiple sections, each of which governs one part of the query and reporting process. You create sections progressively: first, you query a database, then retrieve results, and last generate reports.

Documents can contain data from any number of relational databases queries, multidimensional database queries, and/or from imported data. Documents usually include one or more of the following items:

- A *data model*, is a visual representation of actual database tables
- A *query* or multiple queries for retrieving a subset of data from the database
- *Join* options, including local joins between different data sets within a single document, local join filters, and optional join path generation
- A *results* set displayed in a table-style format
- *Reports* presenting customized hierarchical views of your data
- Multidimensional *pivot* tables that permit drill-down analysis of data results
- *Charts* that graphically display your query results and allow different angles of vision on the data.

Most Hyperion documents have at least one Query section and one Results section. From the Results section, you can create multiple Pivot, Chart, Table, and Report sections to analyze and present data.
Managing your Table

Shown below is a Table also known as a Topic. By double-clicking on the Topic Title Bar you will open the Properties of this particular Table. You can also right-click on the Title Bar to open the Properties.

Below you can see the Properties of the Table **Grant No**. Notice the Table Name is more user-friendly than the Physical Name. This is a transition change that happens when the data is uploaded from SFMA to the Datamart.
What can I do to the Table Properties?

**Table Name** ➔ It is possible to change this name to something that is more understandable.

**Physical Name** ➔ Normally, you would not alter this name. This is the full name of the underlying database table.

**Hide/Show All** ➔ Hides or actively show all table items.

**Up/Down** ➔ Moves selected item up or down one space in the topic display.

**Sort** ➔ Alphabetically sorts listed items.

**Set As Dimension** ➔ Defines the drill-down path or hierarchy for dimensional analysis as shown in the data model.

**Allow Icon View** ➔ Enables the icon view option for the table.

**Allow Detail View** ➔ Enables the detail view option for the table.

**Cause Reload** ➔ Specifies automatic reloading of server values the next time Detail View is activated.

**Rows to Load** ➔ Specifies the number of rows to be loaded and displayed in Detail View.

Icon View - Be Careful! Icon View deactivates a table and reduces it to an icon in the Content frame. This feature will break joins.

Detail View, Cause Reload & Rows to Load: If you choose Detail View you will see the amount of rows identified in Rows to Load. You will get an updated value if Cause Reload is checked or Rows to Load is altered.
Understanding Joins

Tables in relational databases share information through a conceptual link, or join, between related columns in different tables. These relationships are displayed in the data model through visual join lines between topic items.

Joins enable you to connect or link records in two tables by way of a shared data field. Once a data field is shared, other data contained in the joined tables can be accessed. In this way, each record can share data with another record, but does not store and duplicate the same kind of information.

Joins can be automatically created for you, or you can manually join topics.

Join types determine how data is retrieved from a database.

To specify a join type:
1. Select a join line and select View, then Properties or click the Properties icon. The Join Properties dialog box is displayed.
2. Select a join type and click OK.

Four types of joins are supported:
- Simple join (=, >, <, >=, <=, >=): A simple (linear) join retrieves the records in both tables that have an identical data in the joined columns. You can change the default join setting for simple joins by choosing an operator from the drop-down box. The default setting, Equal, is preferred in most situations.
- Left join (+=): A left join retrieves all rows from the topic on the left and any rows from the topic on the right that have matching values in the join column.
- Right join (=+): A right join retrieves all rows from the topic on the right and any rows from the topic on the left that have matching values in the join column.
- Outer join (+ = +): An outer join combines the impact of a left and right join. An outer join retrieves all rows from both tables matching joined column values, if found, or retrieves nulls for non-matching values. Every row represented in both topics is displayed at least once.

Note: Not all database servers support all join types.
Building the Request Line

As you build your query, you can reorder, remove, or hide items on the Request line. This allows you to change the way in which the query processes and displays.

Reordering Request Items
You can move Request items to reorder them for viewing results.

➢ To reorder items on the Request line, select the item to be moved and drag it to a new location on the Request line.

Removing Request Items
You can remove items from the Request line to exclude the data from your query or Results set.

➢ To remove an item from the Request line, select the desired item and complete one of the following actions:
  • Click the Delete button on the standard toolbar.
  • Click Remove on the shortcut menu.
  • Press the Delete key.

If you have not yet processed the query, Interactive Reporting removes the item from the Request line.

If you have previously processed the query, the Report Refresh dialog is displayed with this message: “The section XXXX (section) references the following removed column(s): XXXX (column name). Do you want to keep references to these columns or turn off auto-refresh or remove them with the query is next processed?” You can select to keep the references, or remove them.

Caution! Remove items with caution as a computed item or report may draw data from the item you delete.

Hiding Request Items
You can hide items that are displayed on the Request line. This allows you to incorporate data in the results set without displaying it. Hidden request items cannot be referenced for computations.

➢ To hide a request item, complete one of the following actions:
  • Select the item and click Hide on the shortcut menu.
  • Select the item and choose View, then Hide Request Items.
➢ To show a hidden request item:

1. Complete one of the following actions:
   • Click in the Request line and click Unhide on the shortcut menu.
   • Select View, then Unhide Request Items.

   The Unhide Columns dialog box is displayed.

2. Select the items to view and click OK.

When you have identified the items to include in the query, you can perform a number of other operations before processing the query. You can add filters or computed items to the Request line, or you can use a Request line item to specify a sort order.
Maximizing the Query Building Process

It is good practice to routinely manage your query process. Often, you may be processing a query for the first time and will be unsure of your results and want to prevent a runaway query. There are properties you can control in order to view smaller datasets of your query.

Setting Restrictions on Your Query Properties

This process allows you to put restrictions on your query process prior to running the query. You can set a temporary constraint on the query to return a specific number of rows or to process for a specific number of minutes and then stop. You can also set the query to return only unique rows.

To access Query Properties:

Select **Query** → **Query Options**
or
Double-click **Request** on the **Request Line**

Return Unique Rows** Eliminates duplicate rows from the dataset retrieved by the query. Only unique rows are returned.

Return First** Filters the number of database rows retrieved to the number entered. You must enter a row filter and check mark the box.

Time Limit** Limits the amount of time the query is allowed to run to the number entered. Seconds are entered as a decimal number. Time limits work for asynchronous (having only each operation started only after the preceding operation is completed) database connections and cancel at the earliest opportunity for non-asynchronous connections.
Auto-Process ➔ Specifies the current query as a Standard Query to be processed automatically on download from the Repository.

Custom Group By ➔ Check the box to customize the Group By criteria used to compute aggregate Request Items, with selected items not factored into the grouping. This feature is available only when a data function is placed on the Request Item. For example, you have specified the Sum function on the SFMS Trans Amt item. You might want to exclude other items in your Request Line from the aggregation – that’s when you select this option – so you can specify which items to exclude.

Estimating the Size of Your Query

Queries that sift through and retrieve enormous amounts of data can take a long time to process, and may consume unnecessary system and server resources. If you suspect these factors exist, you may want to limit the size of your query before you hit the process button.

The Estimate Size feature queries the database to see how many records your query will retrieve. You can use this feature to test a questionable query or to decide whether to prevent or postpone processing a large dataset during peak times.

To Estimate the Size of Your Query:

Select Query ➔ Estimate Query Size
**Exercises**

*4A)*
1. Create a new .bqy using the SFMS.oce.

2. Bring the GL Detail table over to the Content Pane.

3. Open the Properties dialog box for the ‘GL Detail’ Table and sort the items.

4. Hide any Items starting with a letter “M”. (Hint – Individually click on the * in front of each item name.)

5. Enter 15 as the value of ‘Rows to Load’ and check the ‘Cause Reload’ box.

6. Select OK.

7. Display the Table in ‘Detail View’. You should see 15 total rows of data.

8. Show all the Items. (Hint: Table properties & ‘show all’ button).

9. Display the Table in ‘Structure View’.

10. Add all the Items to the Request Line.

11. Estimate the size of this query. (Discuss answer with instructor)

12. Remove the Table from the Content Pane, also removing the Items from the Request Line.

13. Once complete, close the bqy document and do not save.

*4B)*
1. Open “GL Detail 0910.bqy” (Obtained from instructor.)
2. Process the query and write down the number of rows and the total of ‘M13 Balance GL’. (Answer: 10,800 rows)

3. Go to the query section.

4. Expand the Tables within the Elements section and add ‘Agency’ and ‘Fund D23’ to the Content Pane.

5. Join the profile tables to the ‘GL Detail’ table. Create ‘simple joins’ on Agency (contained in both tables), Fund and Fiscal Year. (Hint: Drag and Drop).

6. Add the following Items to the Request Line:
   a. Agency table
      i. Agency Title
   b. Fund D23 table
      i. Fund Title
      ii. GAAP Fund
      iii. GAAP Fund Title

7. Process the query.

8. How many rows were returned? (Answer: 10,190 rows)

9. Compare step 2 data with the new totals from step 8. Discuss the difference with the instructor.
Lesson 5

Filters and Sorts

✓ Filtering Queries
  o Setting Filters
  o Creating Custom Values List
✓ Modifying Filters
  o Ignoring Filters
  o Removing Filters
  o Filter Line Logic and Order of Operations
✓ Variable Filters
  o Customizing Variable Filters Dialog Box
✓ Sorting Data
✓ Data Functions on the Requested Items
Filtering Queries

For most queries it is necessary to set a filter on the amount of data to be returned. The Datamart contains massive amounts of data, and you want to avoid requesting unnecessary information. Before processing a query, set the appropriate filters in the Query Section.

In the Query Section, you can:
- Manually specify filter values
- Query the Datamart for filter values
- Create custom lists of filter values
- Define compound filter formulas on the Filter Line
- Set filters as variables to prompt the user to select certain values

Setting Filters

To set a filter:

1) To show the Filter Line: Select \(\text{Filters(0)}\) from the Section Title Bar

2) Drag and drop the Table Item you want to filter to the Filter Line or Double-click the Table Item

The Filter dialog box will now be displayed on your screen if your OCE is active. Otherwise, you must log into your OCE prior to moving forward.
3) From the drop-down list ➔ Select a **Comparison Operator**

4) In the Edit text box ➔ **Type Values** – Separated by commas

5) Select ➔ **Green ✓** to add the values in the Edit text box to the Custom Values List

6) Select ➔ One or more **values**

7) Select ➔ **OK**

**Name** ➔ Displays the name of the filter – it defaults to the item name but you can modify it to say whatever you like.

**Include Nulls** ➔ Check mark this box to include null values.

**Operator** ➔ Select an operator from a drop-down list.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equal =</td>
<td>Retrieves records where the filtered item equals the specified value(s).</td>
</tr>
<tr>
<td>Not Equal (=)</td>
<td>Retrieves records where the filtered item does not equal the specified values(s).</td>
</tr>
<tr>
<td>Less Than (&lt;)</td>
<td>Retrieves records where the filtered item is less than the specified values(s).</td>
</tr>
<tr>
<td>Less or Equal (&lt;=)</td>
<td>Retrieves records where the filtered item is equal to or less than the specified value(s).</td>
</tr>
<tr>
<td>Greater Than (&gt;)</td>
<td>Retrieves records where the filtered item is greater than the specified value(s).</td>
</tr>
<tr>
<td>Greater or Equal (&gt;=)</td>
<td>Retrieves records where the filtered item is equal to or greater than the specified value(s).</td>
</tr>
<tr>
<td>Begins With</td>
<td>Retrieves records where the filtered item begins with the specified value(s) up to and including the end value.</td>
</tr>
<tr>
<td>Contains</td>
<td>Retrieves records where the filtered item contains the specified value(s) regardless of location.</td>
</tr>
<tr>
<td>Ends With</td>
<td>Retrieves records where the filtered item ends with the specified value(s).</td>
</tr>
<tr>
<td>Like (with wildcards)</td>
<td>Retrieves records where a text string appears and reflects the placement of the specified values(s).</td>
</tr>
</tbody>
</table>
For example: Names Like %LI_ would retrieve records for all employees whose names have the letters II followed by a single character at the end. So, “%” represents multiple characters and “_” represents a single character.

<table>
<thead>
<tr>
<th>Filter Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is Null</td>
<td>Retrieves records where the filtered item has no value; for example, a field in which no data has been entered.</td>
</tr>
<tr>
<td>Between</td>
<td>Retrieves records where the value of the filtered item lies between (and including) the specified values.</td>
</tr>
</tbody>
</table>

**Not (with operator)** Negates the operator it precedes, reversing the results of the equation.

**Not** Reverses (inverts) the operator selected from the drop-down list. For example, if you select Equal from the operator list and then choose Not, the operator is effectively changed to Not Equal.

**Edit Line** Enter a value or multiple values (separated by a comma, no spaces) and select the green ✓ to add the item(s) to the custom values list to complete the filter information. Select the red X to erase the contents of the Edit field.

**Show Values** Retrieves a list of values actually in the Datamart (or in the dataset in Results) and allows you to choose a value based on the data available. Show values cannot be used for filtering computed or aggregated items.

**BE CAREFUL!** Because ‘Show Values’ goes directly to the Datamart and searches for the specified values it may take a while to return information. This means it searches all the data for exact values. If you think what you are asking for might be large, don’t select ‘Show Values’; manually type your request in the Edit Line.

If you are setting a filter while in the Result Section don’t hesitate to use Show Values. Using Show Values in the Results Section only searches the information on your Query.

**Transfer** Displays after you’ve selected Show Values – adds values your specified from the list to the Custom List window.
Custom Values ➔ This is the default value when you set a filter. One reason to use custom lists with distributed .bqy documents is many data items change rarely, if at all. For example, a Gender item has three consistent values (male, female, and unknown). An Agency item has many more values, but may only change every year or so. Data that changes rarely is said to have low cardinality. Under such circumstances, it makes sense for users to select from a custom values list, rather than continuously querying to show Datamart values.

Custom SQL ➔ Displays the custom SQL dialog box for coding Filters directly in SQL. The Custom SQL button appears only if you access the Filter dialog box from the Query Section.

Select All ➔ Selects all values displayed in the displayed value area.

Remove ➔ Removes highlighted values from the custom list or a filter.

Ignore ➔ Temporarily suspends a filter without deleting it.

Advanced ➔ Displays Loaded Values settings and Subquery Options.

Loaded Values Settings ➔ Toggles a custom values list to be read from a file or from the Datamart. Change File allows you to specify the file name. If you read values from a text file, vertical tabs or paragraph markers must delimit each value. Use Show Values to display the file contents.

Create Subquery ➔ Creates a Subquery where the current query is the parent. If you access the Filter dialog box from the Results Section, the Advanced button appears as Options and the Create Subquery option does not appear.
Creating Custom Values List

You don’t always have to know your Filters prior to setting them up. You can directly query the Datamart for specific values to filter. **Be Careful! Only use this on items that rarely change or are not sizeable.**

Once filter values are retrieved from the Datamart, you can transfer them to a Custom Values List so they are stored locally with the .bqy. You should do this if the filter values rarely change. If those values change occasionally, you will need to renew the Custom Values List to keep it current with the values in the Datamart.

*To create a Custom Values List:*

1) Drag and drop the **Table Item** you want to filter to the **Filter Line** or Double-click the **Table Item**

The Filter dialog box will now be displayed on your screen if your OCE is active. Otherwise, you must log into your OCE prior to moving forward.

2) From the drop-down list → **Select a Comparison Operator**

3) **Select** → **Show Values**

4) **Select** → **Select All** or Select the specific values you want

5) **Select** → **Transfer**

6) **Select** → **Values you want for the Filter**

7) **Select** → **OK**
Modifying Filters

A Filter can always be modified. You can change the values and settings permanently, temporarily ignore it, or delete it.

To modify a Filter Item:

1) In the Query Section ➔ Double-click the Filter Item

2) Make your changes to the Filter dialog box
   Operator
   Type and select new values
   Show Values and select from a new values list
   Remove values from the Custom Values List

3) Select ➔ OK

Ignoring Filters

You can temporarily ignore a filter without deleting it.

To ignore a Filter Item:

1) In the Query Section ➔ Double-click the Filter Item

2) Select ➔ Ignore

Removing Filters

You can permanently remove a Filter Item.

To remove a Filter Item:

Select the Filter ➔ Right-click ➔ Select Remove
**Filter Line Logic and Order of Operations**

The Filter Line in the Query Section provides AND and OR operators as well as parentheses to control the logic and order of operations of the filters. These features are available only in the Query Section so keep this in mind when creating a query.

By using AND, OR, and parentheses, you can create complex filters and use an item on the Filter Line as many times as needed. When you add duplicate items on the Filter Line they are numbered consecutively.

The following rules apply to all Filter Line expressions:

→ By default, all equations are solved from left to right, when enclosed sub-operations evaluated first.

→ **AND** is evaluated before **OR**.

→ The **AND** operator retrieves data that meets **both** conditions. Both conditions have to be true to get information returned. For example, if you query Agencies, and filter Agency to “100” AND “107”, the dataset returned will include both Agencies.

→ The **OR** operator retrieves data that satisfies **either** of two condition. For example, if you filter Agency to “100” OR Grant No to “515000”, the dataset returned would include Agency 100 and information about Grant No 515000.

→ Suboperations override the default evaluation order and may be required for certain operations that involve both **AND** and **OR** operators.
To set AND, OR, or parentheses on the Filter Line in the Query Section:

1) Drag two or more table items to the Filter Line. Define individual filters in the Filters dialog boxes. (The AND operator is displayed by default.)

2) Select on the left side of the Filter Line.

3) Select from the filter controls to complete the equation.
   To toggle from AND and OR, click the operator.
   To enclose Suboperations, select items to be enclosed and then click the parentheses button.
   To remove parentheses, select them and click the parentheses button.
Variable Filters

Once a filter is created in the Query Section, you can construct it using the variable filter feature to prompt for values when the query is processed. Using variable filters eliminates the need for multiple queries using different filter values. You can use one query to return multiple datasets for different users.

Variable filters work great with Custom Values Lists. If a Custom Values List is created, you can answer the prompt with a choice from the Custom Values List. Each time a user processes the query, the filters are selected from the Custom Values List rather than being recreated.

To create a Variable Filter:

1) In the Query Section make sure to display the Filter Line

2) Select \[ \text{Filter} \]

3) Select \[ \text{Filter Item} \rightarrow \text{Var} \] on the Filter Line
   or
   Right-click the Filter Item \[ \rightarrow \text{Select Variable Filter} \]

   \[
   \begin{array}{|c|c|c|c|c|c|c|}
   \hline
   \text{Request} & \text{Agency} & \text{Gl Acct} & \text{Fund} & \text{Grant No} & \text{Fiscal Year} & \text{M13 Balance Gl} \\
   \hline
   \text{Filter} & \text{Var} & \text{Grant No} & \text{AND} & \text{Fiscal Year} & \text{OR} & \text{Fiscal Year 2 V(1)} \\
   \hline
   \end{array}
   \]

   The Filter Item is displayed with a V(1) next to it. If you add other Variable Filters they are numbered V(2), V(3), etc.
Customizing Variable Filters Dialog Box

The Custom Filter dialog box allows you to control access to the features available in the Filter dialog box. This is useful when you distribute the .bqy to end users. It may be preferable to disable or even remove such features as Include Null or the Custom Values buttons. However, please remember, customizing only affects a single filter.

To customize a variable filter:

1) Display the Filter Line in the Query Section

2) Select Variable Filter Item to be customized ➔ Right-click ➔ Select Customize Filter

3) From the Customize Filter dialog box shown above you can do the following:

Prompt ➔ Add a user-friendly message for display when prompted
Values ➔ Select Values you are giving end users access to use
Options ➔ Again, if these are check marked, you are giving the end user access to do these Options
**Sorting Data**

Use the Sort Line in the Query Section to sort data before it is returned from the Datamart. You can only include items on the Sort Line that also exist on the Request Line.

The appearance and functionality of the Sort Line is nearly identical in the Query, Results, and Table Sections. In each section, the Sort Line uses a drag-and-drop method similar to the Request and Filter Lines. Data sorted in the Results and Table Sections are sorted on your desktop and not directly from the Datamart.

<table>
<thead>
<tr>
<th>Request</th>
<th>Agency</th>
<th>GL Acct</th>
<th>Fund</th>
<th>Grant No</th>
<th>Fiscal Year</th>
<th>M13 Balance GI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sort</td>
<td>Fund</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Filter</td>
<td>Agency AND</td>
<td>Grant No AND</td>
<td>Fiscal Year</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**To use the Sort Line in the Query, Results, or Table Sections:**

1) Make sure to display the **Sort Line** if needed by selecting Sort on the Section Title Bar.

2) Select Table Item on the Request Line you wanted sorted, Drag-and-drop this Table Item onto the Sort Line.

3) Select or **Double-click the Table Item on the Sort Line to toggle sort ascending or sort descending**

4) **Review your Sorted Data**
   - If you are sorting in the Query section, the data is sorted by the Datamart server when you process the query.
   - If the Sort Line is displayed in the Query Section, the item appears on the Sort Line. If you sort in ascending order, an up arrow appears to the right of the item name on the Sort Line; if you sort in descending order, a down arrow appears.
   - If you are sorting in the Results or Table Section-Select **Sort Now**
Data Functions on the Requested Items

You can apply a data function to an Item on the Request Line in the Query Section to compute an aggregated value. An aggregated value is a summarized value. You can summarize items to show averages, sum, minimum, maximum, and count functions to name a few. By setting these aggregated data functions in the request area you are creating a smaller result group and demanding less of the Datamart.

If you need both summary data and increasing levels of detail breakdown in your reporting or analysis, do not aggregate data in the Query Section.

Listed below are the data functions you can choose from using Hyperion. Although most data functions are applied in the Query Section, you can use many of them in the Pivot, Chart, or Results Sections. You cannot apply data functions to the Table Section. In Report Sections, you can apply data functions only if you select a single fact column. You can also change the way data functions are computed.

<table>
<thead>
<tr>
<th>Data Function</th>
<th>Description</th>
<th>Where Can I Use This?</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>Returns unaggregated values as stored in the database. This is the default option in the Query Section.</td>
<td>Query</td>
</tr>
<tr>
<td>Sum</td>
<td>Returns sum of underlying values.</td>
<td>Query, Pivot, Chart, Results, Report</td>
</tr>
<tr>
<td>Average</td>
<td>Return average of underlying values.</td>
<td>Query, Pivot, Chart, Results, Report</td>
</tr>
<tr>
<td>Minimum</td>
<td>Returns lowest of underlying values.</td>
<td>Query, Pivot, Chart, Results, Report</td>
</tr>
<tr>
<td>Maximum</td>
<td>Returns highest of underlying values.</td>
<td>Query, Pivot, Chart, Results, Report</td>
</tr>
<tr>
<td>Count</td>
<td>Returns number of underlying values.</td>
<td>Query, Pivot, Chart, Results, Report</td>
</tr>
<tr>
<td>Count Distinct</td>
<td>Returns the number of distinct values in a column. This function is not supported by all database servers.</td>
<td>Query, Pivot, Report</td>
</tr>
<tr>
<td>Weight</td>
<td>Use for computing weighted items in pivot tables.</td>
<td>Query</td>
</tr>
<tr>
<td>Null Count</td>
<td>Returns number of nulls among underlying values</td>
<td>Pivot, Chart, Report</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>Non-Null Count</td>
<td>Returns number of underlying values; null values excluded.</td>
<td>Pivot, Chart, Report</td>
</tr>
<tr>
<td>Non-Null Average</td>
<td>Returns average of underlying values; null values excluded.</td>
<td>Pivot, Chart, Report</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>Returns standard deviation of values. This function is not supported by all database servers.</td>
<td>Query</td>
</tr>
<tr>
<td>Variance</td>
<td>Returns variance of values. This function is available through Oracle servers only.</td>
<td>Query</td>
</tr>
<tr>
<td>% of Column</td>
<td>Returns sum of all underlying values as a percentage of their respective surface column.</td>
<td>Pivot</td>
</tr>
<tr>
<td>% of Row</td>
<td>Returns sum of underlying values as a percentage of their respective surface row.</td>
<td>Pivot</td>
</tr>
<tr>
<td>% of Grand</td>
<td>Returns sum of underlying values as a percentage of all surface values in the report.</td>
<td>Pivot, Chart</td>
</tr>
<tr>
<td>% of Category</td>
<td>Returns group total percentage of the selected value.</td>
<td>Report</td>
</tr>
<tr>
<td>Increase</td>
<td>Calculates the increase between the previous two rows or columns.</td>
<td>Pivot</td>
</tr>
<tr>
<td>% Increase</td>
<td>Calculates the percentage increase between the previous two rows of columns.</td>
<td>Pivot</td>
</tr>
<tr>
<td>Title</td>
<td>Returns column names.</td>
<td>Report</td>
</tr>
</tbody>
</table>

To use apply a data function to a Request Line Item in the Query Section:

1) Right-click on the Item on the Request Line ➔ Select Data Function ➔ Select desired Data Function
Here are some examples of non-aggregated data vs. aggregated data:

**Example 1**

This query shows non-aggregated data using the GL Detail table.

<table>
<thead>
<tr>
<th>Request</th>
<th>Agency</th>
<th>Gl Acct</th>
<th>Fund</th>
<th>Project No</th>
<th>Project Phase</th>
<th>Pca</th>
<th>M13 Balance Gl</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Agency</th>
<th>Gl Acct</th>
<th>Fund</th>
<th>Project No</th>
<th>Project Phase</th>
<th>Pca</th>
<th>M13 Balance Gl</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td></td>
<td>(30,000.00)</td>
</tr>
<tr>
<td>2</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td>41599</td>
<td>30,000.00</td>
</tr>
<tr>
<td>3</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td></td>
<td>(30,000.00)</td>
</tr>
<tr>
<td>4</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td>41599</td>
<td>30,000.00</td>
</tr>
<tr>
<td>5</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td></td>
<td>(30,000.00)</td>
</tr>
<tr>
<td>6</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td>41599</td>
<td>30,000.00</td>
</tr>
<tr>
<td>7</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td></td>
<td>(30,000.00)</td>
</tr>
<tr>
<td>8</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td>41599</td>
<td>30,000.00</td>
</tr>
<tr>
<td>9</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td></td>
<td>(30,000.00)</td>
</tr>
<tr>
<td>10</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td>41599</td>
<td>30,000.00</td>
</tr>
</tbody>
</table>

**Example 2**

This query shows a ‘Sum’ Data Function on the M13 Balance Gl.

<table>
<thead>
<tr>
<th>Request</th>
<th>Agency</th>
<th>Gl Acct</th>
<th>Fund</th>
<th>Project No</th>
<th>Project Phase</th>
<th>Pca</th>
<th>SUM(M13 Balance Gl)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Agency</th>
<th>Gl Acct</th>
<th>Fund</th>
<th>Project No</th>
<th>Project Phase</th>
<th>Pca</th>
<th>M13 Balance Gl</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td></td>
<td>(150,000.00)</td>
</tr>
<tr>
<td>2</td>
<td>405</td>
<td>0065</td>
<td>4250</td>
<td>415000</td>
<td>00</td>
<td>41599</td>
<td>150,000.00</td>
</tr>
</tbody>
</table>
Exercises

5A)

1. Start a new .bqy using the SFMS.oce.

2. Expand the Table Elements and add the ‘All Acct Event’ Table to the Content Pane.

3. Add the following Items to the Request Line:
   a. Batch Agy
   b. Batch Date
   c. GL Acct
   d. SFMS Trans Amt


5. Filter the Rows Returned to 100 (Hint: Lesson 4 – Return First; p.4-8).

6. Process the query.

7. Sort the Results Section by ‘Batch Date’ and ‘GL Acct’.

8. Add another Table Element to the Content Pane called ‘GL Acct’.

9. Add ‘GL Acct Title’ from the above Table to the ‘Request Line’.

10. Remove the Return First filter of 100 Rows.

11. Add a filter by ‘Batch Date’ using the ‘Between’ data function. Showing data between ‘06/01/10’ and ‘06/14/10’. (Hint: You need to manually type in the dates, separated by a comma because these dates do not show under the ‘Show Values’ button.)
12. Add another Filter and make this one a ‘Custom Filter’ on ‘GL Acct’. (Hint: p.5-6)
   a. Add ‘GL Acct’ from ‘All Acct Event’ to the Filter Line
   b. Show Values
   c. Select any GL Acct’s between 2700-3600
   d. Select the Transfer Button
   e. Set the 2700-3600 accounts as the default for a Custom Value

13. Process the query.

14. In the ‘Result’ Section move the ‘GL Acct Title’ to the right of the ‘GL Acct’.

15. How many rows are returned? (Answer: 2,854)


17. Process the query.

18. How many rows are returned? (Answer: 34)

19. Alert the instructor after correctly completing the exercise.

5B)
1. Start a new .bqy using the SFMS.oce.

2. Expand the ‘Table Elements’ pane and bring the following tables over to the Content Pane:
   a. All Acct Event
   b. PCA
   c. Program Code
3. Sort the fields in each Table.

4. Add the following fields to the Request Line:
   a. All Acct Event
      i. Batch Agy
      ii. Batch Date
      iii. GL Acct
      iv. SFMS Trans Amt (Sum - using the Data Function)
      v. Program Code
      vi. PCA
   b. PCA
      i. PCA Title
   c. Program Code
      i. Program Code Title

5. Filter on ‘Batch Agency’ equal to ‘585’ and ‘Fiscal Year’ equal to ‘2010’.

6. Process the query.

7. How many rows are returned? (Answer: 19,914)

8. Go back to the ‘Query’ Section and Filter on ‘Program Code’ equal to 5160 and equal to 7100.


10. In the Results Section
    a. Sort Ascending: Batch Date, GL Acct and Program Code.

11. How many rows are returned? (Answer: 573)

12. Go back to the ‘Query’ Section and ignore the ‘Program Code’ field from the ‘Filter Line’.
   a. Drag ‘GL Acct’ to the Filter line. You will be prompted by a pop up window.
   b. Set the ‘Custom Value’ to Begins with 31 and click ‘ok’.
   c. Right-click on the ‘GL Acct’ Filter and select ‘Variable Filter’.

14. Process the query. (The Variable filter for ‘GL Acct’ will pop up. Verify selection and click ‘ok’)

15. How many rows are returned? (Answer: 738)

16. Alert the instructor after correctly completing the exercise.
Lesson 6

Tables and Formatting

✓ What’s the difference between Results and Table Sections?
✓ Formatting Data in Results and Table Sections
✓ Sorting Data in a Table Section
✓ Suppressing Duplicate Values
✓ Setting and Removing Local Filters in Results and Table Sections
✓ Creating Date Groups
✓ Creating Grouping Columns
✓ Removing and Hiding Data
✓ Creating Grand Totals and Break Totals
What’s the difference between Results and Table Sections?

The Results and Table Sections are identical in functionality and appearance. So, what’s the difference? The Results Section is the gas tank for your .bqy. This is where all your queried data lives. The Results Sections allows you to review your Query and confirm the Results. You can build different tables, which include smaller sections of your Results. Think of a Table Section as a filing cabinet with different files. It is a way to organize your data into smaller sub-sections and then develop Pivots, Charts, and Report from these sub-sets. Frankly, a Table keeps you better organized.

Sections Structure:

Dashboard
- This section does not indent and can contain information from multiple Queries, Results, Tables, Pivots, Charts, and Reports. It appears at the top of the Section Catalog.

Report
- This section does not indent and can contain information from multiple Results, Tables, Pivots, and Charts. It appears directly below Dashboard Sections and above all other sections.

Query
- This section does not indent and is a stand-alone section in your .bqy. This section comes straight from the Datamart and is the building block for all the other sections.

Results
- This section does not indent but resides directly below the Query Section it is dependant upon. This section is directly related to the Query Section.

Table
- This section does not indent and resides directly below the Results Section it is associated with. You can have more than one Table Section related to a Results Section. Used primarily for creating tabular-style reports.

Notice the indenting or non-indenting for the different Sections. It is good practice to rename your sections for easier identification.
Pivot

This section does indent and resides below the Table Section it is associated with. You can have more than one Pivot Section related to a Results Section. The Pivot Section is used to create cross-tabular style reports called Pivot Reports.

Chart

This section does indent and resides below the Table and/or Pivot Section it is associated with. You can have more than one Chart Section related to a Results Section. Chart Reports are fully interactive, three-dimensional views of data.

Below shows a Table Section before any data is added to the Content Pane. Notice the Data Layout Section.

To create a Table Section (based on the Results section data):

1) From the Results section, select **Insert**→**New Table**

2) If the table Data Layout is not already displayed→Click **Data Layout** on the Section Title Bar

3) Drag Result Items from the Catalog Pane to the table Data Layout to build your Table. (Hyperion automatically populates the table columns.)
Formatting Data in Results and Table Sections

Use the following techniques to format data in both a Results and a Table Section.

<table>
<thead>
<tr>
<th>Format</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resize a column</td>
<td>✓ To resize a column manually, drag the right margin of the column.</td>
</tr>
<tr>
<td></td>
<td>✓ To autosize a column, double-click the column margin or select <strong>Format</strong>→<strong>Column</strong>→<strong>Auto-Size Width</strong> (or <strong>Standard Width</strong>).</td>
</tr>
<tr>
<td>Resize a row</td>
<td>✓ To resize a row manually, select a row (click in the left margin), position the cursor on the bottom of the row, and then drag the margin up or down.</td>
</tr>
<tr>
<td></td>
<td>✓ To resize a row to a standard size, select a row and then select <strong>Format</strong>→<strong>Row</strong>→<strong>Standard Height</strong>.</td>
</tr>
<tr>
<td>Modify the font</td>
<td>✓ Select a column, select <strong>Format</strong>→<strong>Font</strong>, and then select a font type, style, size, and effect. Columns can have different fonts.</td>
</tr>
<tr>
<td>Modify a number, date, or time format</td>
<td>✓ For numeric formatting only, select a column and then on the Formatting toolbar, select numeric formatting options.</td>
</tr>
<tr>
<td></td>
<td>✓ For a complete set of formatting options, select a column, select <strong>Format</strong>→<strong>Number</strong>, and then in the <strong>Category</strong> group box, select a format.</td>
</tr>
<tr>
<td>Justify data</td>
<td>✓ Highlight columns, and then on the Formatting toolbar, click a justification button.</td>
</tr>
<tr>
<td>Display row numbers</td>
<td>✓ To toggle row numbers on and off, select <strong>Format</strong>→<strong>Row Numbers</strong>.</td>
</tr>
<tr>
<td>Format grid lines</td>
<td>✓ Select <strong>Format</strong>→<strong>Grid Lines</strong>, and then select properties for horizontal and vertical grid lines.</td>
</tr>
<tr>
<td>Display or hide column titles</td>
<td>✓ Select <strong>Format</strong>→<strong>Column Titles</strong>.</td>
</tr>
<tr>
<td>Wrap text in a column</td>
<td>✓ Select a column, and then select <strong>Format</strong>→<strong>Text Wrap</strong>.</td>
</tr>
<tr>
<td>Modify the border around a table</td>
<td>✓ Select <strong>Format</strong>→<strong>Border and Background</strong>, select properties for the border, and then click <strong>OK</strong>.</td>
</tr>
</tbody>
</table>
Modify the table fill

✓ Select a column, multiple columns, or the entire table, select **Format ➔ Border and Background**, select background properties, and then click **OK**.

Modify the text color of a column

✓ Select a column, and then on the **Formatting** toolbar, select a text color.

The following table provides a listing of commands available on the Format Menu.

<table>
<thead>
<tr>
<th>Format Menu Command</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Font</td>
<td>✓ Opens the Font page of the Properties dialog box.</td>
</tr>
<tr>
<td>Style</td>
<td>✓ Choose between Plain, Bold, Italics, Underline, Overline, Double Underline.</td>
</tr>
<tr>
<td>Number</td>
<td>✓ Opens the Number page of the Properties dialog box.</td>
</tr>
<tr>
<td>Justify</td>
<td>✓ Choose between Left, Center, Right and Top, Middle, Bottom.</td>
</tr>
<tr>
<td>Feature</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Column</td>
<td>By default, Results columns are evenly sized without regard to the length of data. Numeric data that does not fit is replaced with pound signs (#). To manually resize a column, drag the right edge of the column to a new position. To automatically size all columns so that the column width fits its contents, press [Ctrl+A] [Ctrl+E].</td>
</tr>
<tr>
<td>Auto-Size Width</td>
<td>Resizes the selected column to the width of the contents.</td>
</tr>
<tr>
<td>Standard Width</td>
<td>Resizes the selected column to the standard column width.</td>
</tr>
<tr>
<td>Row</td>
<td>Resizes all rows to the standard row height. To resize all rows in a table, drag the bottom edge of a row to a new position. All rows in the table are resized.</td>
</tr>
<tr>
<td>Column Titles</td>
<td>Toggles the display of column titles.</td>
</tr>
<tr>
<td>Row Numbers</td>
<td>Toggles the display of row numbers. Row numbers are printed on reports, but are not copied to the clipboard or exported to a file.</td>
</tr>
<tr>
<td>Text Wrap</td>
<td>Wraps text in a column.</td>
</tr>
<tr>
<td>Suppress Duplicates</td>
<td>Suppresses duplicate values in a column. Use this feature if you want to display only the first instance of a duplicate value when individual database records include redundant information.</td>
</tr>
<tr>
<td>Grid Lines</td>
<td>Opens the Gridlines page of the Properties dialog box.</td>
</tr>
<tr>
<td>Border and Background</td>
<td>Opens the Border and Background page of the Properties dialog box.</td>
</tr>
<tr>
<td>Spotlighter</td>
<td>Opens the Spotlight dialog box. Use to spotlight important values.</td>
</tr>
<tr>
<td>Export Properties</td>
<td>Opens the Export Properties dialog box. Use to set the number of rows that should be included on an HTML page before the data breaks to another page, and to export data that does not contain any quote to a tab-delimited text file.</td>
</tr>
</tbody>
</table>
Sorting Data in a Table Section

You can sort data in the Table Section using the sort options on the Standard Toolbar or the Sort Line.

To sort data using the Standard Toolbar:

1) Select Column you want sorted

2) Select Sort Ascending or Sort Descending

To sort data using the Sort Line:

1) Make sure the Sort Line is visible; if it isn’t select Sort on the Section Title Bar

2) Drag and drop Items from the Content Pane to the Sort Line. By default, Items will be sorted in ascending order.

3) Double-click Table Item on the Sort Line to toggle sort ascending or sort descending

4) Select Sort Now on the Sort Line
Suppressing Duplicate Values

You use Suppress Duplicates to suppress duplicate values in a column. Use this feature if you want to display only the first instance of a duplicate value when individual database records include redundant information. In order to use this feature correctly, you must first sort your data.

<table>
<thead>
<tr>
<th>Batch Key</th>
<th>Batch Date</th>
<th>GLAcct</th>
<th>$SmTrans Amt</th>
<th>Pca</th>
<th>Pca Title</th>
<th>Program Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>301</td>
<td>07/13/09</td>
<td>3100</td>
<td>12,865.52</td>
<td>00090</td>
<td>LEGAL SERVICES REVENUE</td>
<td>0090</td>
</tr>
<tr>
<td>3100</td>
<td>(12,005.52)</td>
<td>3100</td>
<td>6,942.00</td>
<td>99990</td>
<td>UNAPPLIED AGENCY TRANSFERS &amp; PAYMENTS</td>
<td>1055</td>
</tr>
<tr>
<td>3100</td>
<td>787.66</td>
<td>84585</td>
<td>OF LIMITED-OPERATING-OAH</td>
<td>1060</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3100</td>
<td>(757.00)</td>
<td>3100</td>
<td>21,783.54</td>
<td>80901</td>
<td>SSD- FLEET ADMIN</td>
<td>8410</td>
</tr>
<tr>
<td>3100</td>
<td>(21,736.54)</td>
<td>3100</td>
<td>1.08</td>
<td>98300</td>
<td>MASS TRANSIT</td>
<td>2013</td>
</tr>
<tr>
<td>3100</td>
<td>(1.08)</td>
<td>3100</td>
<td>5,550.00</td>
<td>41002</td>
<td>ADMINISTRATION - OF</td>
<td>4100</td>
</tr>
<tr>
<td>3100</td>
<td>167.89</td>
<td>71002</td>
<td>SET ASIDE FUND</td>
<td>7100</td>
<td></td>
<td></td>
</tr>
<tr>
<td>07/14/09</td>
<td>3100</td>
<td>10.90</td>
<td>98300</td>
<td>MASS TRANSIT</td>
<td>2013</td>
<td></td>
</tr>
<tr>
<td>3100</td>
<td>(10.80)</td>
<td>3100</td>
<td>152.00</td>
<td>22534</td>
<td>ARCHIVES - ADMINISTRATIVE RULES</td>
<td>2200</td>
</tr>
<tr>
<td>3100</td>
<td>(152.00)</td>
<td>3100</td>
<td>21,042.36</td>
<td>11380</td>
<td>FAC - O&amp;M-REVENUE</td>
<td>6310</td>
</tr>
<tr>
<td>3100</td>
<td>(21,042.36)</td>
<td>3101</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To suppress duplicate values:

1) Sort the data in the Results or Table Section

2) Select the columns you want to suppress values on Right-click
   Select Suppress Duplicates
Setting and Removing Local Filters in Results and Table Sections

When you set a filter in the Results or Tables Section, you can locally filter the dataset returned from a query. You are not accessing the Datamart. This is different from setting a filter in a Query Section. When you set a filter in a query, you are sending this filter directly to the Datamart. Because local filters only hide data, it is a good way to test hypothetical situations without accessing the Datamart. You can always reverse the effects of your filter and bring your query back to its original state.

To set a filter in a Results or Table Section:

1) In a Results or Table section, double-click a column. A Filter dialog box is displayed
   or
   Drag and drop the Table Item you want to filter to the Filter Line

   The Filter dialog box will now be displayed on your screen.

   ![Filter Dialog Box]

2) From the drop-down list ➔ Select a Comparison Operator

3) In the Edit text box ➔ Type Values – Separated by commas

4) Select ➔ Green ✓ to add the values in the Edit text box to the Custom Values List

5) Select ➔ OK
To remove a filter in a Results or Table Section:

1) Right-click the Filtered Item ➔ Select Remove

To remove all filters in a Results or Table Section:

1) Right-click the Filter on the Filter Line ➔ Select Remove
Creating Date Groups

This feature separates regular date type columns into Month, Year, and Quarter columns. This feature only works on columns of a date type. Please notice the before and after diagrams shown below.

### Before Date Groups

<table>
<thead>
<tr>
<th>Batch Agt</th>
<th>Batch Date</th>
<th>$fms Trans Amt</th>
</tr>
</thead>
<tbody>
<tr>
<td>585</td>
<td>04/01/10</td>
<td>60.74</td>
</tr>
<tr>
<td>585</td>
<td>08/25/09</td>
<td>6,163.92</td>
</tr>
<tr>
<td>585</td>
<td>12/16/09</td>
<td>72.69</td>
</tr>
<tr>
<td>585</td>
<td>01/05/10</td>
<td>4,743.20</td>
</tr>
<tr>
<td>585</td>
<td>08/17/09</td>
<td>81.80</td>
</tr>
<tr>
<td>585</td>
<td>08/27/09</td>
<td>3,502.42</td>
</tr>
<tr>
<td>505</td>
<td>03/31/10</td>
<td>3,403.54</td>
</tr>
<tr>
<td>585</td>
<td>01/11/10</td>
<td>64.17</td>
</tr>
<tr>
<td>585</td>
<td>03/26/10</td>
<td>5,265.35</td>
</tr>
<tr>
<td>585</td>
<td>06/16/10</td>
<td>4,055.84</td>
</tr>
</tbody>
</table>

The first diagram is without Date Grouping. The second diagram now has a Month, Year, and Quarter column in addition to the regular date column.

### Date Groups Applied

<table>
<thead>
<tr>
<th>Batch Agt</th>
<th>Batch Date</th>
<th>$fms Trans Amt</th>
<th>Batch Date Month</th>
<th>Batch Date Year</th>
<th>Batch Date Quarter</th>
</tr>
</thead>
<tbody>
<tr>
<td>585</td>
<td>04/01/10</td>
<td>60.74</td>
<td>Apr</td>
<td>2010</td>
<td>Q2</td>
</tr>
<tr>
<td>585</td>
<td>08/25/09</td>
<td>6,163.92</td>
<td>Aug</td>
<td>2009</td>
<td>Q3</td>
</tr>
<tr>
<td>585</td>
<td>12/16/09</td>
<td>72.69</td>
<td>Dec</td>
<td>2009</td>
<td>Q4</td>
</tr>
<tr>
<td>585</td>
<td>01/05/10</td>
<td>4,743.20</td>
<td>Jan</td>
<td>2010</td>
<td>Q1</td>
</tr>
<tr>
<td>585</td>
<td>08/17/09</td>
<td>81.80</td>
<td>Aug</td>
<td>2009</td>
<td>Q3</td>
</tr>
<tr>
<td>585</td>
<td>08/27/09</td>
<td>3,502.42</td>
<td>Aug</td>
<td>2009</td>
<td>Q3</td>
</tr>
<tr>
<td>585</td>
<td>03/31/10</td>
<td>3,403.54</td>
<td>Mar</td>
<td>2010</td>
<td>Q1</td>
</tr>
<tr>
<td>585</td>
<td>01/11/10</td>
<td>64.17</td>
<td>Jan</td>
<td>2010</td>
<td>Q1</td>
</tr>
<tr>
<td>585</td>
<td>03/26/10</td>
<td>5,265.35</td>
<td>Mar</td>
<td>2010</td>
<td>Q1</td>
</tr>
<tr>
<td>585</td>
<td>06/16/10</td>
<td>4,055.84</td>
<td>Jun</td>
<td>2010</td>
<td>Q2</td>
</tr>
</tbody>
</table>

To create a date group:

1) Select a regular Date column in the Results or Table Section

2) Right-click ➔ Add Date Groups
Creating Grouping Columns

Grouping columns is a way of creating new data in your results set by grouping data from a column. You can use grouping columns to consolidate non-numeric data values into more general group values and map the group values to a new column in the data set. Grouped columns are new items added to the Results Section and are available for use in Report Sections.

Steps to create a Grouping Column:

1) Notice the Table Section below. The goal is to create an additional new column called Quarter.

2) Select the Column you want to group ➔ Right-click ➔ Select Add Grouping Column

3) The Grouping Column dialog box opens
**Column Name** ➔ Names the new grouping column in the Results/Table window. This is the column header.

**New Group** ➔ Creates a custom group to be displayed as a value in the new grouping column.

**Groups** ➔ Select a custom group to define by adding or removing items.

**Items in Group** ➔ Removes an item from a selected custom group.

**Available Values** ➔ Add items to a selected custom group.

**Options** ➔ Indicates how to signify unassigned values within the grouping column. Options include:

- **Null** ➔ Leaves the values ungrouped and unaggregated.
- **Default** ➔ Allows you to specify a default name to assign to all ungrouped values.
- **Individual Group** ➔ Assigns each ungrouped value the name originally assigned to it.

4) In the Column Name field ➔ **Name your Grouping Column**

5) Select **New Group** ➔ **Enter New Group Name** ➔ **OK**
   Do this process until you have all the new groups set up.

6) Select **Group** in Group Column ➔ **Move Available Values** by selecting them and click on ⬅️ to move them over to the **Items in Group Column**.
   Continue this process until you have all your groups set up.
7) Select → OK

<table>
<thead>
<tr>
<th>Batch Agy</th>
<th>Batch Date</th>
<th>Gl Acct</th>
<th>Sfms Trans Amt</th>
<th>Pca</th>
<th>Fiscal Quarter</th>
</tr>
</thead>
<tbody>
<tr>
<td>585</td>
<td>07/27/09</td>
<td>1111</td>
<td>(70.00)</td>
<td>71002</td>
<td>Quarter 1</td>
</tr>
<tr>
<td>585</td>
<td>07/30/09</td>
<td>1011</td>
<td>70.00</td>
<td>71002</td>
<td>Quarter 1</td>
</tr>
<tr>
<td>585</td>
<td>07/27/09</td>
<td>3101</td>
<td>(2,072.25)</td>
<td>71002</td>
<td>Quarter 1</td>
</tr>
<tr>
<td>585</td>
<td>07/30/09</td>
<td>0070</td>
<td>(140.00)</td>
<td>71002</td>
<td>Quarter 1</td>
</tr>
<tr>
<td>585</td>
<td>07/30/09</td>
<td>3500</td>
<td>140.00</td>
<td>71002</td>
<td>Quarter 1</td>
</tr>
<tr>
<td>585</td>
<td>07/30/09</td>
<td>1111</td>
<td>70.00</td>
<td>71002</td>
<td>Quarter 1</td>
</tr>
<tr>
<td>585</td>
<td>12/08/09</td>
<td>1111</td>
<td>(70.00)</td>
<td>71002</td>
<td>Quarter 2</td>
</tr>
<tr>
<td>585</td>
<td>12/08/09</td>
<td>1011</td>
<td>(70.00)</td>
<td>71002</td>
<td>Quarter 2</td>
</tr>
<tr>
<td>585</td>
<td>12/07/09</td>
<td>3500</td>
<td>140.00</td>
<td>71002</td>
<td>Quarter 2</td>
</tr>
<tr>
<td>585</td>
<td>12/07/09</td>
<td>3501</td>
<td>(140.00)</td>
<td>71002</td>
<td>Quarter 2</td>
</tr>
<tr>
<td>585</td>
<td>12/11/09</td>
<td>0501</td>
<td>(405.56)</td>
<td>71002</td>
<td>Quarter 2</td>
</tr>
</tbody>
</table>

*To modify a Grouping Column:*

1) Right-click the Grouping Column → Modify Column
Removing and Hiding Data

You can either Remove or Hide items from your Query, Results, and/or Table Sections. Use caution when removing items from any section! Once an item is removed, you can’t use it in a Pivot, Chart, or Report. If you hide an item it is still usable in all these sections.

To remove an item in the Query or Table Section:

Remove the Table Item from the Request Line from the Query Section or the Data Layout Section in the Table Section.

To hide an item in the Results or Table Section:

Select the Column ➔ Right-click ➔ Hide Column
Creating Grand Totals and Break Totals

Results and Table Sections can display grand totals and break totals. You can display multiple totals each utilizing a different data function. While a grand total would appear at the bottom of a column, a break total will appear at a specified break in the dataset.

To display a grand total:

1) Select column you want to total\rightarrow Right-click\rightarrow Select Grand Total

The Insert Grand Total dialog box appears.

<table>
<thead>
<tr>
<th>Batch Agy</th>
<th>Batch Date</th>
<th>Gl Acct</th>
<th>Smtn Trans Amt</th>
<th>Fiscal Year</th>
<th>Pca</th>
<th>Pca Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>525</td>
<td>07/01/09</td>
<td>101</td>
<td>$500.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/01/09</td>
<td>111</td>
<td>$500.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/02/09</td>
<td>101</td>
<td>$70.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/02/09</td>
<td>111</td>
<td>$70.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/06/09</td>
<td>101</td>
<td>$70.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/06/09</td>
<td>111</td>
<td>$70.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/06/09</td>
<td>111</td>
<td>$70.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/06/09</td>
<td>101</td>
<td>$70.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/06/09</td>
<td>111</td>
<td>$70.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/14/09</td>
<td>0070</td>
<td>$70.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/14/09</td>
<td>0070</td>
<td>$70.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>525</td>
<td>07/14/09</td>
<td>101</td>
<td>$70.00</td>
<td>2010</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

You have a choice of Grand and Break Total functions:

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum</td>
<td>Returns sum of underlying values.</td>
</tr>
<tr>
<td>Average</td>
<td>Returns average of underlying values.</td>
</tr>
<tr>
<td>Minimum</td>
<td>Returns lowest of underlying values.</td>
</tr>
<tr>
<td>Maximum</td>
<td>Returns highest of underlying values.</td>
</tr>
<tr>
<td>Count</td>
<td>Returns number of underlying values.</td>
</tr>
<tr>
<td>Other</td>
<td>Allows you to create a custom function using JavaScript.</td>
</tr>
</tbody>
</table>

2) Select\rightarrow Grand total function you desire

3) Select\rightarrow Where to Add grand total to

4) Select\rightarrow OK
You can also create a SUM grand total using an icon on the Standard Toolbar:

1) Select → Column you want to total

2) Select → on the Standard Toolbar

To remove a grand total row:

1) Select → Row containing the grand total

2) Right-click → Remove Row

To add a custom title in the grand total row, not the grand total cell:

1) Double-click on a cell in the same row as the grand total

2) In the Custom Function dialog box enter the title you desire surrounding by quotes “” as shown below:
3) Select → OK

4) From the Modify Total Function dialog box select → Other

5) Select → OK
To add a custom title in the grand total cell:

1) Double-click on the grand total cell

2) From the Modify Total Function dialog box→Select Other

3) Enter your desired title surrounded by quotes “”, for example “Total: ”. Make sure to insert a space after your title!

4) Concatenate text and functions with a plus sign (+)

   “Total ”+Round(Sum(Sfms_Trans_Amt),2)

5) Select→OK

6) In the Modify Total Function dialog box→Select OK
To display break totals:

1) Select column you want to total → Right-click → Select Break Total
   The Insert Break Total dialog box appears.

2) At every break in drop-down → Select your break column criteria

3) Break total function → Select function

4) Select one or more columns → Add Break total to

5) Select → OK
Exercises

1. Create a new .bqy using the SFMS.oce.

2. Include the following tables and fields in the query:
   a. All Acct Event - table:
      i. Batch Agy
      ii. Batch Date
      iii. GL Acct
      iv. PCA
      v. Fund
      vi. SFMS Trans Amt (Use Data Function: Sum)
   b. Agency - table:
      i. Agency Title
   c. Fund D23 - table:
      i. Fund Title
      ii. GAAP Fund
      iii. GAAP Fund Title
   d. GL Acct - table:
      i. GL Acct Title

3. Arrange the fields shown below on the ‘Request Line’ to look like the following:

<table>
<thead>
<tr>
<th>Request</th>
<th>Batch Agy</th>
<th>Agency Title</th>
<th>Batch Date</th>
<th>SFMS Trans Amt</th>
<th>GL Acct</th>
<th>GL Acct Title</th>
<th>PCA</th>
<th>Fund</th>
<th>Fund Title</th>
<th>GAAP Fund</th>
<th>GAAP Fund Title</th>
</tr>
</thead>
</table>

4. Filter the query to ‘Batch Agy’ equal to 585, ‘Batch Date’ between 07/01/09 and 06/30/10 and ‘SFMS Trans Amt’ not equal to zero.

5. Process the query. (Answer: 26,870 rows)

6. Filter the ‘Results’ Section to ‘SFMS Trans Amt’ not equal to zero. (Answer: 24,110 of 26,870 rows)

7. Insert a New Table Section.
8. Bring the following fields over to the ‘Data Layout’ section to allow them to show on the Content Pane of the ‘Table’ Section:
   a. Batch Date
   b. PCA
   c. GL Acct
   d. GL Acct Title
   e. Fund
   f. Fund Title
   g. SFMS Trans Amt

9. Within the Table section, sort by ‘Batch Date’, ‘PCA’, ‘GL Acct’, and ‘Fund’ fields in Ascending Order and create a Filter of ‘PCA’ beginning with 2 or 3.

10. Change the font of every column on your Table to Arial - size 9 and auto size the width of your columns. (Hint: To auto size use ‘Ctrl A’ & ‘Ctrl E’).


12. Add a ‘Date Group’ from the ‘Batch Date’ column. Then hide the ‘Batch Month’ and ‘Batch Year’ columns. Move the ‘Batch Date Quarter’ column to the right of ‘Batch Date’.

13. Next, create the following grouped column based on the ‘GL Acct’ column. The new columns’ name refers to the CAFR Acct; however, you only need to input the number of the group (For example, ‘CAFR Acct 999-Default Oregon’ will only be input as ‘999’):
   a. CAFR Acct: 010-Cash and Cash Equivalents
   b. CAFR Acct: 050-Accounts & Interest Receivable
   c. CAFR Acct: 060-Inventories
   d. CAFR Acct: 090-Accumulated Depreciation & Amortization
   e. CAFR Acct: 120-Accounts & Interest Payable
f. CAFR Acct: 160-Trust Funds Payable  
g. CAFR Acct: 162-Compensated Abs Pay-Current  
h. CAFR Acct: 175-Compensated Abs Pay-Noncurrent  
i. CAFR Acct: 270-Budgetary Fund Balance  
j. CAFR Acct: 271-Encumbrance Control  
k. CAFR Acct: 301-Reserved for Encumbrances  
l. CAFR Acct: 307-GAAP Expenditure Offset  
m. CAFR Acct: 310-Unreserved, Undesignated Fund Balance  
n. CAFR Acct: 360-Reserved for Inventories  
o. CAFR Acct: 371-Invested in Capital Assets  
p. CAFR Acct: 375-Net Assets  
q. CAFR Acct: 999-Default Oregon  

14. Next, assign the ‘Available Values’ to the New Groups as follows:  
   a. CAFR Acct: 010  
      i. 0065  
      ii. 0070  
      iii. 0075  
   b. CAFR Acct: 050  
      i. 0501  
      ii. 0507  
      iii. 0573  
   c. CAFR Acct: 060  
      i. 0601  
   d. CAFR Acct: 090  
      i. 0870  
      ii. 0873  
   e. CAFR Acct: 120  
      i. 1211  
      ii. 1215  
      iii. 1216  
   f. CAFR Acct: 160  
      i. 1551  
   g. CAFR Acct: 162  
      i. 1605
h. CAFR Acct: 175
   i. 1718
i. CAFR Acct: 270
   i. 2700
   ii. 2900
j. CAFR Acct: 271
   i. 2735
k. CAFR Acct: 301
   i. 3011
l. CAFR Acct: 307
   i. 3600
m. CAFR Acct: 310
   i. 3075
   ii. 3100
   iii. 3101
   iv. 3150
   v. 3500
   vi. 3501
   vii. 3505
   viii. 3550
n. CAFR Acct: 360
   i. 3005
o. CAFR Acct: 371
   i. 3018
p. CAFR Acct: 375
   i. 3074
q. CAFR Acct: 999
   i. 1011
   ii. 1111
   iii. 2701
   iv. 2901
   v. 2904
   vi. 2954
   vii. 9010
   viii. 9011
15. Select ‘Ok’ when finished, and format the new column as necessary.

16. Modify the newly created ‘CAFR Acct’ grouped column to show all unassigned values as Default → ‘Error’.
   
   **Hint:** Within the pop up window for creating new groups, select ‘options’.
   
   **Note:** This will allow any new unassigned values to show within an ‘Error’ category, so that you do not forget to assign the new value.

17. Within the ‘Table’ section, under the ‘SFMS Trans Amt’ column, input a summed ‘Grand Total’.
   Your Grand Total should equal 1,618,748.24.

18. Alert the instructor after correctly completing the exercise.