CA IDMS - 19.0

Online Query (OLQ) Reports

Date: 10-Jan-2018
This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the "Documentation") is for your informational purposes only and is subject to change or withdrawal by CA at any time. This Documentation is proprietary information of CA and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION “AS IS” WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with “Restricted Rights.” Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2017 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.
Table of Contents

What are CA OLQ features? ................................................................................................. 11
  Menu facility ....................................................................................................................... 12
  CA OLQ command mode ................................................................................................. 13
  Two processing modes .................................................................................................... 13
  Two access modes ........................................................................................................... 13

Sign on and Sign off ........................................................................................................ 13
  Getting Help .................................................................................................................... 14
  Using Menu Facility ........................................................................................................ 15
  Using PF Keys ................................................................................................................ 17

How to Make a Report from a Table .................................................................................. 18
  Key Terms ........................................................................................................................ 19
  Creating Your Report From an SQL Table ....................................................................... 19
    Step 1 -- Select the type of table .................................................................................. 20
    Step 2 -- Select a table .................................................................................................. 21
    Step 3 -- Choose columns and specify selection criteria .............................................. 21
    Step 4 -- Specifying additional selection criteria ......................................................... 23
    Step 5 -- Retrieve the data ............................................................................................ 25
    Step 6 -- Display your report ......................................................................................... 26
  Creating Your Report From an ASF Table ...................................................................... 27
    Step 1 -- Select the type of table .................................................................................. 28
    Step 2 -- Select a table .................................................................................................. 28
    Step 3 -- Choose columns and specify selection criteria .............................................. 29
    Step 4 -- Specifying additional selection criteria ......................................................... 31
    Step 5 -- Retrieve the data ............................................................................................ 32
    Step 6 -- Display your report ......................................................................................... 34

How to Make a Report From More Than One Table ......................................................... 34
  How to Combine Data From More Than One Table ....................................................... 35
  Creating your report ......................................................................................................... 39
    Step 1 -- Select the type of table .................................................................................. 39
Step 2 -- Select tables ................................................................. 40
Step 3 -- Select, project, and join ...................................................... 41
Step 4 -- Retrieve the data ................................................................. 41
Step 5 -- Display your report .............................................................. 42

How to Format Your Report ................................................................. 42
Format Report Key Terms ................................................................. 43
Creating a Report .............................................................................. 44
  Step 1 -- Select a subschema ......................................................... 45
  Step 2 -- Select records ................................................................. 46
Step 3 -- Choose columns ................................................................. 46
Step 4 -- Retrieve the data ................................................................. 47
Step 5 -- Display your report .............................................................. 47
Step 1 - Create a Current Report ...................................................... 48
Step 2 - Modify Your Report Headers .......................................... 51
Step 3 - Change Your Report Column Pictures .............................. 52
Step 4 - Sort Your Report Rows and Specify Group Calculations .......... 53
Sorting Your Report Rows ................................................................. 54
  Step 1 - Specify your sort criteria ................................................... 54
  Step 2 - Display your report .......................................................... 55
  Step 1 - Sort your report ............................................................... 55
  Step 2 - Display your report .......................................................... 56
Editing Your Report Values .............................................................. 56
  Step 1 - Specify your edit criteria .................................................... 58
  Step 2 - Display your report .......................................................... 59
  Additional editing features ............................................................ 59
Changing Your Column Relative Positions ...................................... 62
Changing Your Column Pictures ...................................................... 63
  Step 1 - Modify a column picture ................................................... 65
  Step 2 - Display your report .......................................................... 66
Changing Column Headers .............................................................. 66
  Step 1 - Change your column headers .......................................... 68
  Step 2 - Display your report .......................................................... 68
Making Page Headers and Footers .................................................... 69
  Step 1 - Specify page headers and footers .................................... 73
  Step 2 - Display your report .......................................................... 73

How to Create Report Totals and Subtotals ................................. 73
Report Totals and Subtotals Key terms ........................................... 74
Creating a Qfile ............................................................. 107
Step 1 - Build the salary report ......................................................... 107
Step 2 - Add formatting enhancements .............................................. 110
Step 3 - Save the report as a Qfile .................................................... 112
Executing a Qfile ............................................................................ 113
Using One Qfile to Create Different Reports .................................... 115
Step 1 - Use execute with new criteria .............................................. 116
Step 2 - Change the Selection criteria ................................................ 117
Step 3 - Resume Qfile execution ........................................................ 117
Modifying Your Qfile Definition ......................................................... 119
Step 1 - Establish a current report ..................................................... 120
Step 2 - Modify the report definition ................................................. 122
Step 3 - Replace the Qfile definition ................................................. 123
Looking at Your Qfile Definition Syntax ........................................... 124
Step 1 - Switch to IDD ..................................................................... 126
Step 2 - View your Qfile in IDD ....................................................... 126
Step 3 - Leave IDD ......................................................................... 127
Modifying Your Qfile Definition Syntax ............................................. 127
Step 1 - Switch to IDD ..................................................................... 128
Step 2 - Modify the SALARY Qfile Definition .................................... 129
Step 3 - Replace the SALARY Qfile definition ................................... 130
Step 4 - Switch from IDD back to CA OLQ ....................................... 130
Executing a Qfile in Batch Mode ....................................................... 131
Step 1 - Select a Qfile ..................................................................... 132
Step 2 - Using the Batch Processing screen ....................................... 133
Deleting a Qfile ................................................................................. 133

How to Make a Report from Database Records ................................. 134
Key Terms 5 .................................................................................... 135
Create a Report ............................................................................... 137
Step 1 -- Choose a subschema. ......................................................... 137
Step 2 -- Select your records ........................................................... 138
Step 3 -- Choose columns and specify selection criteria ...................... 138
Step 4 -- Enter additional selection criteria ....................................... 143
Step 5 -- Retrieve the data and display your report .............................. 144

How to Save a Table From a Report ................................................ 145
Save a Table Key Terms .................................................................... 147
Creating a Report 3 .......................................................................... 148
Answers .......................................................................................................................... 209
Chapter 2 Answers ........................................................................................................ 209
Chapter 3 Answers ........................................................................................................ 211
Chapter 4 Answers ........................................................................................................ 212
Chapter 5 Answers ........................................................................................................ 215
Chapter 6 Answers ........................................................................................................ 216
Final Query ...................................................................................................................... 217
Glossary ............................................................................................................................ 217
Online Query (OLQ) Reports

In this section CA OLQ (Online Query) is a query tool and report writer that accesses data stored in a CA IDMS/DB database.

You can use CA OLQ to:

- Query a CA IDMS/DB database. For example, if you want to know the phone number of an employee, you can use CA OLQ to retrieve the phone number from the database.
- Create reports. For example, if you want to list the names, phone numbers, and sales quotas for a group of salespeople, you can use CA OLQ to create this report.

For more information, see the following topics:
- What are CA OLQ features? (see page 11)
- Sign on and Sign off (see page 13)
- How to Make a Report from a Table (see page 18)
- How to Make a Report From More Than One Table (see page 34)
- How to Format Your Report (see page 42)
- How to Create Report Totals and Subtotals (see page 73)
- How To Format Reports Containing Calculations (see page 92)
- How to Save a Set of Commands as a Qfile (see page 104)
- How to Make a Report from Database Records (see page 134)
- How to Save a Table From a Report (see page 145)
- How to Use CA OLQ in Batch Mode (see page 155)
- How to Print Your Report (see page 159)
- Introducing the OLQ SELECT statement (see page 161)
- Retrieving Information from a Table (see page 172)
- Summarizing Information (see page 187)
- Joining Tables or Database Records (see page 190)
- Sample Tables and Database (see page 205)
- Answers (see page 209)

What are CA OLQ features?

CA OLQ offers the following features:

- Menu facility (see page 12)
- CA OLQ command mode (see page 13)
- Two processing modes (see page 13)
- Two access modes (see page 13)
Menu facility

Using the CA OLQ menu facility you can learn how to create reports quickly. The menus are easy to use, and provide most of the power of syntax-driven CA OLQ.

Create reports

You can report on:

- SQL tables
- ASF tables
- Database records

Format your report

You can enhance the appearance of your report by:

- Sorting report rows
  - Modifying column and page headers
  - Specifying external pictures for columns
  - Changing the spacing between columns
- Using code tables to translate data stored in the database into an alternative external representation

Perform computations

You can create the following types of report calculations:

- Computed columns
  - Break processing and group subtotals
  - Report totals
- Calculations including aggregate and built-in functions

Save a report

Once you create a report, you can save it online for a specified period of time.

Create a table from a report

You can create a table by saving your CA OLQ report as a table. The rows and columns in the table correspond to those you have defined in the report.
Define predefined routines

You can save the set of CA OLQ commands used to create a report in a routine. This routine can be re-executed to create reports that reflect the changing nature of data in the database.

Print a report

You can route your CA OLQ report to output devices defined to your system.

Execute OLQ commands using the OLQ batch facility

You can use the CA OLQ batch facility to create reports that query very large amounts of data, or to execute during non-peak hours.

CA OLQ command mode

CA OLQ also offers syntax mode for very specific queries, or for people who prefer to use syntax.

For information on how to use CA OLQ command mode, refer to the CA OLQ Reference Section.

Two processing modes

You can create a report in either an online or a batch environment.

Two access modes

CA OLQ provides you with the ability to access both SQL and ASF tables, database and logical records, and sequential files (batch only). Setting the access switch to either idms or olq tells CA OLQ what kind of table you want to access.

Sign on and Sign off

Signing on

You sign on to CA OLQ from CA IDMS/DC or CA IDMS UCF. When CA IDMS/DC or CA IDMS UCF prompts you with ENTER NEXT TASK CODE, specify olq menu options.

To go directly to a specific screen, name the screen when you sign on. For example, to go straight to the Session Options screen when you sign on, type the following at the CA IDMS/DC or CA IDMS UCF system prompt:

V12 ENTER NEXT TASK CODE:
   olq menu options
CA IDMS - 19.0

Signing off

To sign off from CA OLQ:

- Press [PF3] from any menu facility screen.
- Type QUIT or BYE in the command line.

In this section, you will learn how to do the following in CA OLQ:

- Getting Help (see page 14)
- Using Menu Facility (see page 15)
- Using PF Keys (see page 17)

Getting Help

The CA OLQ help system provides quick answers about how to use CA OLQ.

For example, if you're working on the Column Select screen and have a question about how to specify selection criteria, you can press [PF1] to get more information.

The following help screen shows ways to specify retrieval selection criteria.

<table>
<thead>
<tr>
<th>CA OLQ. Release nn.n</th>
<th>*** Help ***</th>
<th>Page 10 of 20</th>
</tr>
</thead>
<tbody>
<tr>
<td>- &gt;</td>
<td></td>
<td>127000 Press the ENTER key to go to the next screen</td>
</tr>
<tr>
<td>127000 Press the ENTER key to go to the next screen</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HELP FOR *** COLUMN (FIELD) SELECT ***</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Retrieving all rows</td>
<td></td>
<td></td>
</tr>
<tr>
<td>If you don't enter any selection criteria on the screen, CA OLQ will retrieve all rows from the data tables or records.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>What to type under SELECTION CRITERIA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Each column has its own SELECTION CRITERIA entry. If you want to retrieve rows based on the value in a certain column, fill in the SELECTION CRITERIA entry of that column. For example:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X 02 COLUMN1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 COLUMN2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X 02 COLUMN3 eq 2000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X 02 COLUMN4 gt 5 * COLUMN1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2=GLOBAL HELP 3=QUIT 4=MESSAGE 6=MENU 7=BWD 8=FWD CLEAR=PREVIOUS SCREEN</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type of Help</th>
<th>PF key</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>How to use CA OLQ in general</td>
<td>[PF 2]</td>
<td>General information on how to use the CA OLQ menu facility. Topics include CA OLQ command-line commands, PF-key assignments, and overviews of how to report on tables and records.</td>
</tr>
<tr>
<td>How to use a specific menu facility screen</td>
<td>[PF 1]</td>
<td>How to use each screen, including how to use each of the fields on the screen.</td>
</tr>
</tbody>
</table>
Using Menu Facility

Screen components

Each CA OLQ screen has:

- **The screen name** listed in the top right corner.

- **A page field** under the screen name. The page field shows you which page of the current screen or report you are on. You can type over this field to jump to a specific page.

- **A command line** marked by the `->` prompt. You can enter commands to move you to other screens or to perform other CA OLQ functions.

- **A message line** right below the command line. CA OLQ messages either tell you what to do next or signal that you have tried to perform an invalid function.

- **A list of PF keys** along the bottom of the screen. The PF key assignments correspond to those established at system generation.

### CA OLQ Release nn.n

<table>
<thead>
<tr>
<th>PFkey</th>
<th>Option</th>
<th>Description</th>
<th>Command/Screen Name</th>
<th>Show/Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Choose tables</td>
<td>TABLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PF2</td>
<td>Choose subschema</td>
<td>SUBschema</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Choose records</td>
<td>REcord</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Choose columns</td>
<td>COLUMN</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Retrieve data</td>
<td>RETrieve</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Alter database</td>
<td>LINkage</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Processing Mode</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Execute</td>
<td>QFile</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Save</td>
<td>SAVE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Submit batch</td>
<td>BATch</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1=HELP 3=QUIT 4=MESSAGE 5=GLOBAL HELP 8=FWD

### How to use the screens

To select a screen option, type any character (except a blank, underscore, or d) on the line next to the option listing and press [Enter].
For example, to select the EMPLOYEE record from the Record Select screen, type a character on the line next to the EMPLOYEE option and press [Enter].

To get to another menu screen, type the command for that screen in the command line and press [Enter].

For example, to get to the Menu screen, type MENU in the command line of any menu facility screen and press [Enter].

To specify syntax on your screen, enter the syntax statement next to the prompt on the screen.

For example, to specify a COMPUTE statement, enter the statement next to the Compute prompt.
Using PF Keys

PF keys are used to:

- **Scroll through pages of a screen** ([PF7] and [PF8])
- **Display the current report** ([PF5])
- **Access a specific screen** (for example, [PF6] for the Menu screen)
- **Page right and left** through the default sequence of the Report Format screens ([PF10] and [PF11])
- **Access the help system** ([PF1], [PF2], [PF4])
- **Terminate a CA OLQ session** ([PF3])

For example, CA OLQ has a set of screens that you use to add formatting enhancements to your report. These screens are set up so that you can scroll through them by using [PF10] and [PF11].

After you have created your report, you can enhance its appearance by using the Report Format screens. Usually, you start with the Report Format - Sort screen and use [PF11] to scroll right to additional formatting screens.

**Default Paging Sequence of the Report Format screens:** You can use PF keys to page between the CA OLQ report formatting screens.
IDMSDB—Using PF Keys

PF key values are assigned at system generation. The values given in this discussion are the default system-generation values. If your site has other PF key assignments, they will be reflected in the help system.

How to Make a Report from a Table

In this section In this section, you create a report that retrieves data from a table. The following report displays data stored in the EMPLOYEE table.

<table>
<thead>
<tr>
<th>EMP-LAST-NAME</th>
<th>DEPT-ID</th>
<th>SALARY-AMOUNT</th>
<th>PROJECT</th>
</tr>
</thead>
<tbody>
<tr>
<td>BANK</td>
<td>4000</td>
<td>80000.00</td>
<td>TESTING</td>
</tr>
<tr>
<td>ANGELO</td>
<td>4000</td>
<td>18000.00</td>
<td>PLANNING</td>
</tr>
<tr>
<td>MCDUGALL</td>
<td>4000</td>
<td>18000.00</td>
<td>PLANNING</td>
</tr>
<tr>
<td>PENMAN</td>
<td>4000</td>
<td>39000.00</td>
<td>PLANNING</td>
</tr>
<tr>
<td>JACKSON</td>
<td>4000</td>
<td>34000.00</td>
<td>PLANNING</td>
</tr>
<tr>
<td>ZEDI</td>
<td>4000</td>
<td>37000.00</td>
<td>EVALUATE</td>
</tr>
</tbody>
</table>

What is a table?

A table is a systematic arrangement of data in rows and columns:

- Each row contains related information on a particular item. For example, data about an employee named Jackson.

- Each column represents a category of information. For example, each employee’s last name.

In other words, columns are attributes associated with each row. Each row contains one entry for each column.

Figure 3-1 on page 3-4 shows a table containing information on employees.

The EMPLOYEE table:
The EMP-ID column represents one category of information on the EMPLOYEE table. Each row represents one data occurrence, which includes all the information on one employee.

For more information on tables, refer to *CA IDMS SQL Self-Training Section*.

**Key Terms**

**Column**

A vertical division in a table. A column represents a category of information, for example, employee last name.

**Table**

A presentation of data as a series of rows and columns.

- *ASF tables* refers to tables associated with the IDMSR schema.
- *SQL tables* refers to tables associated with an SQL schema.

**Row**

A horizontal division in a table. A row represents one data occurrence, for example, information on each employee.

**Selection criteria**

An expression that specifies which rows of a table are to be selected for processing.

**Creating Your Report From an SQL Table**

**Contents**

- Step 1 -- Select the type of table (see page 20)
- Step 2 -- Select a table (see page 21)
Step 3 -- Choose columns and specify selection criteria (see page 21)
Step 4 -- Specifying additional selection criteria (see page 23)
Step 5 -- Retrieve the data (see page 25)
Step 6 -- Display your report (see page 26)

In this example, you create a report from an SQL table using the following steps:

1. Be certain the access switch is set to idms
2. Select a table
3. Indicate which columns you want to appear in your report and specify selection criteria that determine which rows are retrieved for your report
4. Specify any additional selection criteria for rows to be retrieved for your report
5. Retrieve the data for your report
6. Display your report

Step 1 -- Select the type of table

When you sign on to CA OLQ, select the Session Options screen:

V12 ENTER NEXT TASK CODE:
olq menu options

For this example, select Y to access SQL tables:

For more information on the access switch, refer to the CA OLQ Reference Section.
Step 2 -- Select a table

For this example, select Choose tables.

<table>
<thead>
<tr>
<th>Pfkey</th>
<th>Option</th>
<th>Description</th>
<th>Command/Screen Name</th>
<th>Show/Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Data Source for Report</td>
<td>Choose tables</td>
<td>Table</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Retrieval Activity</td>
<td>Choose records from selected subschema</td>
<td>Record</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Processing Mode</td>
<td>Execute or create a predefined routine</td>
<td>QFile</td>
<td></td>
</tr>
</tbody>
</table>

Specify SELECT and select the EMPLOYEE table.

Step 3 -- Choose columns and specify selection criteria

When you create a report that retrieves data from a table, you do not have to display the whole table. You can restrict the scope of your report in three ways:

- By choosing which columns you want to display
- By specifying selection criteria to restrict which rows are retrieved for your report

- By specifying that you want to display only unique rows on your report

Choosing columns

You don’t have to display all of the columns in your table. By using the Column Select screen, you can select only those columns that you want to include in your report.

For example, if you just wanted to list employees’ names and department numbers, you would only select those columns:

<table>
<thead>
<tr>
<th>Columns Currently Selected: 0</th>
<th>Selection Criteria</th>
<th>Distinct N/ Y/N</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPDEMO.EMPLOYEE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_ID 0415</td>
<td>s</td>
<td></td>
</tr>
<tr>
<td>EMP_LAST_NAME 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_FIRST_NAME 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_STREET 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_CITY 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_STATE 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_ZIP_FIRST_FIVE 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_ZIP_LAST_FOUR 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_PHONE 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_SALARY_AMOUNT 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_START_YEAR 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_TERMINATION_DATE 0415</td>
<td>s</td>
<td></td>
</tr>
<tr>
<td>EMP_DEPT_ID</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Specifying selection criteria

Selection criteria are logical expressions that you use to tell CA OLQ which rows of data to retrieve for your report. You specify your selection criteria in the Selection criteria field of the Column Select screen.

For example, if you only wanted to list the names of those employees in department 4000, you would specify the following:

<table>
<thead>
<tr>
<th>Columns Currently Selected: 0</th>
<th>Selection Criteria</th>
<th>Distinct Y/N</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPDEMO.EMPLOYEE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_ID 0415</td>
<td>s</td>
<td></td>
</tr>
<tr>
<td>EMP_LAST_NAME 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_FIRST_NAME 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_STREET 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_CITY 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_STATE 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_ZIP_FIRST_FIVE 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_ZIP_LAST_FOUR 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_PHONE 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_SALARY_AMOUNT 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_START_YEAR 0415</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP_TERMINATION_DATE 0415</td>
<td>s</td>
<td></td>
</tr>
<tr>
<td>EMP_DEPT_ID</td>
<td></td>
<td>eq 4000</td>
</tr>
</tbody>
</table>

Specifying only unique rows

You do not have to include duplicate rows (detail lines) on your report. By using the Distinct Y/N field on the Column Select screen, you can display a report that contains only unique rows.
Select lastname, state, phone, salary, start year, termination date, and department ID. Specify that you want to display only unique rows. Also, specify that you want to view only the list of employees in department 4000.

Step 4 -- Specifying additional selection criteria

Additional selection criteria

The Additional selection criteria field and the Selection Criteria screen, give you more room to add selection criteria for the rows of data you want in your report.

In this example, you want to report on those employees whose:

- Department ID is 4000 (already specified above)
- Phone exchange (first 3 digits) is '329'
- Last name begins with a 'C' or an 'S'
- State of residence is Massachusetts or New Hampshire
- Annual salary is greater than $40,000 and less than $100,000
- Start date with the company was after 1988
- Status is active (still an active employee)

Begin entering the additional selection criteria on this screen. Then specify Y to go to the Selection Criteria Screen for more room to enter selection criteria.
124000 Select columns, specify selection criteria and press the ENTER key.

Columns Currently Selected: 0

EMPDEMO.EMPLOYEE
  _ 02 EMP_ID 0415
  s 02 EMP_LAST_NAME 0415
  _ 02 EMP_FIRST_NAME 0415
  > 02 EMP_STREET 0415
  _ 02 EMP_CITY 0415
  s 02 EMP_STATE 0415
  _ 02 EMP_ZIP_FIRST_FIVE 0415
  s 02 EMP_ZIP_LAST_FOUR 0415
  s 02 EMP_PHONE 0415
  s 02 EMP_SALARY_AMOUNT 0415
  s 02 EMP_START_YEAR 0415
  s 02 EMP_TERMIMATION_DATE 0415
  s 02 EMP_DEPT_ID eq 4000

Additional Selection Criteria: EMP_PHONE_0415 = 329

Proceed to Selection Criteria Screen? Y Y/N
1=HELP  3=QUIT  4=MESSAGE  6=MENU  PA2=REFRESH

Finish entering the additional selection criteria.

146000 Type in selection criteria, and press the ENTER key.
Please Enter Additional Selection Criteria:
EMP_PHONE_0415 = 329 AND (EMP_LAST_NAME_0415 = 'C' OR EMP_LAST_NAME_0415 = 'S') AND (EMP_STATE_0415 = 'MA' OR EMP_STATE_0415 = 'NH') AND (EMP_SALARY_AMOUNT_0415 > 40000 AND EMP_SALARY_AMOUNT_0415 < 100000) AND (EMP_START_YEAR_0415 > 88) AND (EMPTERMINATION_DATE_0415 IS NOT NULL);

With the Selection Criteria screen you can also enter:

- Exists predicates
- Quantified predicates (ANY, SOME, ALL)
- Nested select predicates
- Multiple criteria for the same command
- Group by, order by, and so on
Step 5 -- Retrieve the data

When you are retrieving the data for your report, two CA OLQ screens help you monitor what's going on.

The Retrieval Interrupted screen

This screen indicates that your report will contain more records than the current interrupt count allows.

⚠️ Note: This screen is here just to show you an example. The numbers on the screen do not reflect the numbers you will encounter in this example.

Specify Yes to continue data retrieval.

<table>
<thead>
<tr>
<th>CA OLQ Release nn.n</th>
<th>*** Retrieval Interrupted ***</th>
</tr>
</thead>
<tbody>
<tr>
<td>131000 Select YES or NO and press the ENTER key</td>
<td></td>
</tr>
<tr>
<td>Retrieval interrupted due to excessive data base accesses.</td>
<td></td>
</tr>
<tr>
<td>Number of whole rows . . . . . . . . . . . . . . 90</td>
<td></td>
</tr>
<tr>
<td>Total number of records read . . . . . . . . . . 100</td>
<td></td>
</tr>
<tr>
<td>Total number of records selected . . . . . . . . 90</td>
<td></td>
</tr>
<tr>
<td>Number of data errors . . . . . . . . . . . . . . 0</td>
<td></td>
</tr>
<tr>
<td>Continue execution x Yes</td>
<td></td>
</tr>
<tr>
<td>X No</td>
<td></td>
</tr>
<tr>
<td>Current interrupt interval is 100 data base accesses.</td>
<td></td>
</tr>
</tbody>
</table>

What is an interrupt count?

The interrupt count indicates how many records are retrieved from the database at a time. It serves as a testing feature when you are designing your report, enabling you to create the report format that you want without retrieving more data than you need to.

Changing the interrupt count

The interrupt count is set at system generation. You can change this value by using the Interrupt Count option on the Retrieval Interrupted screen, or on the Session Options screen.

What to do
If the number of records that meet the criteria for your report exceeds the interrupt count, the Retrieval Interrupted screen asks you whether you want to continue data retrieval:

- **If you want to continue retrieval** select the Yes function on the screen. CA OLQ continues to retrieve more records until the interrupt count is met again, or until all of the records that meet the criteria for your report have been retrieved.

- **If you want to terminate retrieval** select the No function on the screen. CA OLQ create a report that contains only those records that have been retrieved up to this point.

The Retrieval Completed screen

This screen indicates that retrieval is completed for your report. From here you can either display your report or select one of the report formatting or print options from the menu on the screen.

In this example, only eight records are retrieved for your report. Since this is less than the current interrupt count, you bypass the Retrieval Interrupted screen and go straight to the Retrieval Completed screen.

**Step 6 -- Display your report**

Your report displays information on the employees in department 4000 with the additional selection criteria you specified.
Creating Your Report From an ASF Table

Contents

- Step 1 -- Select the type of table (see page 28)
- Step 2 -- Select a table (see page 28)
- Step 3 -- Choose columns and specify selection criteria (see page 29)
- Step 4 -- Specifying additional selection criteria (see page 31)
- Step 5 -- Retrieve the data (see page 32)
- Step 6 -- Display your report (see page 34)

In this example, you create a report from a table using the following steps:

1. Be certain that the access switch is set to olq.
2. Select a table.
3. Indicate which columns you want to appear in your report and specify selection criteria that determine which rows are retrieved for your report.

4. Retrieve the data for your report.

5. Display your report.

**Step 1 -- Select the type of table**

When you sign on to CA OLQ, select the Session Options screen:

```
V12 ENTER NEXT TASK CODE:
olq menu options
```

For this example, select N to access ASF tables:

```
CA OLQ Release nn.n
*** Session Options ***
Page 1 of 2
Access IDMS SQL tables: N (Y/N)
User options: Change
Help Option
-> Report Processing Options <-
_ _ NOFiller FILLer
_ _ FULL SPArse
_ _ HEader NOHeader
-> Column Processing Options <-
_ _ OLQheader NOOlqheader
_ _ PICTURE NOPIcture
_ _ CODetable NOCODetable
1=HELP 3=QUIT 4=MESSAGE 6=MENU 8=FWD
```

For more information on the access switch, refer to the CA OLQ Reference Section.

**Step 2 -- Select a table**

For this example, select **Choose tables**.

```
CA OLQ Release nn.n
*** Menu ***
Page 1 of 3
Choose before requesting a retrieval, display, or format activity
Select an option and press the ENTER key
Pfkey Option Description
Select Command/ Screen Name Help
--- Data Source for Report <---
TABle
```

10-Jan-2018  28/222
Specify **SELECT** and select the EMPLOYEE table.

---

**Step 3 -- Choose columns and specify selection criteria**

When you create a report that retrieves data from a table, you do not have to display the whole table. You can restrict the scope of your report in three ways:

- By choosing which columns you want to display
- By specifying selection criteria to restrict which rows are retrieved for your report
- By specifying that you want to display only unique rows on your report

**Choosing columns**

You don’t have to display all of the columns in your table. By using the Column Select screen, you can select only those columns that you want to include in your report.

For example, if you just wanted to list employees' names and department numbers, you would only select those columns:
### Specifying selection criteria

Selection criteria are logical expressions that you use to tell CA OLQ which rows of data to retrieve for your report. You specify your selection criteria in the **Selection criteria** field of the Column Select screen.

For example, if you only wanted to list the names of those employees in department 4000, you would specify the following:

<table>
<thead>
<tr>
<th>Columns Currently Selected: 0</th>
<th>Selection Criteria</th>
<th>Distinct N Y/N</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 EMP-ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x 02 EMP-LAST-NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 EMP-FIRST-NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 START-YEAR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 DEPT-HEAD-ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x 02 DEPT-ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 SALARY-AMOUNT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 PROJECT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 OFFICE-CODE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

More information on how to specify selection criteria is given in Section11, "How to Make a Report from Database Records".

### Specifying only unique rows

You do not have to include duplicate rows (detail lines) on your report. By using the **Distinct Y/N** field on the Column Select screen, you can display a report that contains only unique rows.

Select last name, start year, department head, department ID, salary amount, and project. Specify that you want to display only unique rows. Also, specify that you want to view only the list of employees in department 4000.

<table>
<thead>
<tr>
<th>Columns Currently Selected: 0</th>
<th>Selection Criteria</th>
<th>Distinct N Y/N</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 EMP-ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x 02 EMP-LAST-NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 EMP-FIRST-NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 START-YEAR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 DEPT-HEAD-ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x 02 DEPT-ID</td>
<td></td>
<td>eq 4000</td>
</tr>
<tr>
<td>02 SALARY-AMOUNT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 PROJECT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>02 OFFICE-CODE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Step 4 -- Specifying additional selection criteria

Additional selection criteria

The Additional selection criteria field and the Selection Criteria screen, give you more room to add selection criteria for the rows of data you want in your report.

In this example, you want to report on those employees whose:

- Department ID is 4000 (already specified above)
- Project is Evaluation
- Manager is Ms. Sanchez (DEPT-HEAD-ID = 0111)
- Last name begins with a 'C' or an 'S'
- Annual salary is greater than $40,000 and less than $100,000
- Start date with the company was after 1988

Begin entering the additional selection criteria on this screen. Then specify Y to go to the Selection Criteria Screen for more room to enter selection criteria.
Finish entering the additional selection criteria.

<table>
<thead>
<tr>
<th>CA OLQ Release nn.n</th>
<th><em><strong>Selection Criteria</strong></em></th>
</tr>
</thead>
<tbody>
<tr>
<td>-&gt;</td>
<td></td>
</tr>
</tbody>
</table>

146000 Type in selection criteria, and press the ENTER key.
Please Enter Additional Selection Criteria:

```
PROJECT EQ 'EVALUATION' AND (DEPT-ID EQ 0111) AND (EMP-LAST-NAME EQ 'C' OR EMP-LAST-NAME EQ 'S') AND (SALARY-AMOUNT GT 40000 AND SALARY-AMOUNT LT 100000) AND (START-YEAR GT 88)
```

With the Selection Criteria screen you can also enter:

- Logical record keywords
- Criteria expressions for subscripted fields

**Step 5 -- Retrieve the data**

When you are retrieving the data for your report, two CA OLQ screens help you monitor what's going on.

**The Retrieval Interrupted screen**

This screen indicates that your report will contain more records than the current interrupt count allows.

⚠️ Note: This screen is here just to show you an example. The numbers on the screen do not reflect the numbers you will encounter in this example.

Specify Yes to continue data retrieval.

<table>
<thead>
<tr>
<th>CA OLQ Release nn.n</th>
<th>*** Retrieval Interrupted ***</th>
</tr>
</thead>
<tbody>
<tr>
<td>-&gt;</td>
<td></td>
</tr>
<tr>
<td>131000</td>
<td>Select YES or NO and press the ENTER key</td>
</tr>
<tr>
<td></td>
<td>Retrieval interrupted due to excessive data base accesses.</td>
</tr>
<tr>
<td>Number of whole rows.</td>
<td>. . . . . . . . . . . . . . . . 46</td>
</tr>
</tbody>
</table>
What is an interrupt count?

The interrupt count indicates how many records are retrieved from the database at a time. It serves as a testing feature when you are designing your report, enabling you to create the report format that you want without retrieving more data than you need to.

Changing the interrupt count

The interrupt count is set at system generation. You can change this value by using the Interrupt Count option on the Retrieval Interrupted screen, or on the Session Options screen.

What to do

If the number of records that meet the criteria for your report exceeds the interrupt count, the Retrieval Interrupted screen asks you whether you want to continue data retrieval:

- **If you want to continue retrieval** select the Yes function on the screen. CA OLQ will continue to retrieve more records until the interrupt count is met again, or until all of the records that meet the criteria for your report have been retrieved.

- **If you want to terminate retrieval** select the No function on the screen. CA OLQ will create a report that contains only those records that have been retrieved up to this point.

The Retrieval Completed screen

This screen indicates that retrieval is completed for your report. From here you can either display your report or select one of the report formatting or print options from the menu on the screen.

In this example, only six records are retrieved for your report. Since this is less than the current interrupt count, you bypass the Retrieval Interrupted screen and go straight to the Retrieval Completed screen.
Step 6 -- Display your report

Your report displays information on the employees in department 4000 with the additional selection criteria you specified.

How to Make a Report From More Than One Table

In this section In this section, you create a report that combines information from two tables, the DEPARTMENT table and the EMPLOYEE table.
Join

A relational operation through which two or more tables are combined. Tables are joined based on columns that the tables have in common.

Join criteria

A logical expression that compares like columns in two or more tables.

Project

A relational operation through which only particular columns of a table are accessed.

Select

A relational operation through which only particular rows of a table are accessed.

Selection criteria

An expression that specifies which rows of a table are to be selected for processing.

How to Combine Data From More Than One Table

Relational operations

When you combine data from two or more tables, you need to specify information that relates them somehow. Three relational operations, select, project, and join, can be used to define and access tables:

- **Select** enables you to choose which **rows** you want to include in your report.
- **Project** enables you to choose which **columns** you want to display.
- **Join** enables you to combine **two or more tables** on the basis of common values.

In the preceding section, you applied two of these relational operations:

- You **selected** which rows you want displayed in your report by specifying selection criteria.

\[
\text{select} \ * \ \text{from} \ \text{department} \ \text{where} \ \text{dept-id} = 4000
\]

DEPARTMENT
DEPT-NAME         DEPT-ID         DEPT NAME/ID
Computer Operations  3200          DEPTNAME         DEPTID
Public Relations      4000          Public Relations  4000
Thermoregulation      5200

Selecting rows:

- You **projected** which columns you want displayed in your report.

**Project**

```
select emp-last-name, dept-id from employee
```

<table>
<thead>
<tr>
<th>EMP-ID</th>
<th>EMP-LAST-NAME</th>
<th>EMP-FIRST-NAME</th>
<th>START-YEAR</th>
<th>DEPT-ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0235</td>
<td>Alanza</td>
<td>Mariana</td>
<td>87</td>
<td>3200</td>
</tr>
<tr>
<td>1450</td>
<td>Enrique</td>
<td>Uriel</td>
<td>76</td>
<td>4000</td>
</tr>
<tr>
<td>9811</td>
<td>Roberts</td>
<td>Ellen</td>
<td>90</td>
<td>4000</td>
</tr>
<tr>
<td>6732</td>
<td>Ditka</td>
<td>Robert</td>
<td>81</td>
<td>4000</td>
</tr>
<tr>
<td>7648</td>
<td>Carr</td>
<td>Eustace</td>
<td>84</td>
<td>5200</td>
</tr>
<tr>
<td>5552</td>
<td>Chiu</td>
<td>Li</td>
<td>86</td>
<td>5200</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EMP/DEPT-ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMP-LAST-NAME</td>
</tr>
<tr>
<td>Alanza</td>
</tr>
<tr>
<td>Enrique</td>
</tr>
<tr>
<td>Roberts</td>
</tr>
<tr>
<td>Ditka</td>
</tr>
<tr>
<td>Carr</td>
</tr>
<tr>
<td>Chiu</td>
</tr>
</tbody>
</table>

**Projecting Columns:**

In this section you specify **join** criteria to combine data from the EMPLOYEE and DEPARTMENT tables into a report.

**What are join criteria?**

Join criteria are logical expressions that equate a column in one table with equivalent columns in additional tables. You **must** specify join criteria if you are reporting on more than one table at a time.

Joining tells CA OLQ which columns the tables have in common. By comparing the values in these columns, CA OLQ can match the rows in the tables and retrieve only those rows that the tables share.

```
select depart.dept-name, department.dept-id, employee.emp-last-name
from employee, department where department.dept-id = employee.dept-id
```
Joining the EMPLOYEE and DEPARTMENT tables:

How do you specify join criteria?

To join two tables, find a column that the two tables have in common. The columns do not have to have the same name, but they should:

- Have the same internal picture
- Have the same data type (for example, numeric)
- Contain some data that matches (in order to display data from both tables)

Where do you specify join criteria?

You specify join criteria in one of two places -- on the Column Select screen, or on the Selection Criteria screen:

- On the Column Select screen, in the Selection criteria field next to the join column: For example, these join criteria join TABLE1 and TABLE2 based on common values of the ID and NUMBER fields:

```
Columns Currently Selected: 0 Selection Criteria Distinct N Y/N
  TABLE1
    X 03 ID
    X 03 NAME
    X 03 PHONE
  TABLE2
    - 03 NUMBER            eq ID
    X 03 SALARY
Additional Selection Criteria:

Proceed to Selection Criteria Screen? N Y/N
```

(Note that you could also specify the join criteria, eq ID, next to the TABLE1 ID column.)
• On the **Column Select** screen, under **Additional Selection Criteria**:

![Screen shot showing column selections and selection criteria]

- **Columns Currently Selected:** 0
  - TABLE1
    - 03 ID
    - 03 NAME
    - 03 PHONE
  - TABLE2
    - 03 NUMBER
    - 03 SALARY
- **Additional Selection Criteria:**
  - name eq 'george'

**Proceed to Selection Criteria Screen? N Y/N**

• On the **Selection Criteria** screen, under **Additional Selection Criteria**:

![Screen shot showing additional selection criteria input]

**146000 Type in selection criteria, and press the ENTER key.**

**Please Enter Additional Selection Criteria:**

id eq number and (phone matches '617******') and (salary gt 30000 and salary lt 50000)

**Example**

You can join the **EMPLOYEE** and **JOB** tables because they both have columns that represent employee ID numbers (column ID in **EMPLOYEE** and column EMP-NUMBER in **JOB**). To join them, specify the following:

![Screen shot showing column selections and selection criteria with join criteria]

- **Columns Currently Selected:** 0
  - EMPLOYEE
    - 03 ID
    - 03 NAME
    - 03 PHONE
  - JOB
    - 03 EMP-NUMBER
    - 03 SALARY
- **Additional Selection Criteria:**
  - eq ID

**Proceed to Selection Criteria Screen? N Y/N**

Note that you do not have to select the EMP-NUMBER column from **JOB**. If you did, the report would list the employee ID twice, once for ID and once for EMP-NUMBER.

**What if columns have the same name?**

If your common columns have the same name, you must include the table name in the join criteria. The table names and the column names are separated by a period (.)

For example, if both **EMPLOYEE** and **JOB** contain the ID field, specify:

![Screen shot showing column selections and selection criteria with join criteria]

- **Columns Currently Selected:** 0
  - EMPLOYEE
    - 03 ID
    - 03 NAME
    - 03 PHONE
  - JOB
    - 03 ID
    - 03 SALARY
- **Additional Selection Criteria:**
  - eq employee.id

**Proceed to Selection Criteria Screen? N Y/N**
What happens if you don't specify join criteria?

When the access mode is set to olq, CA OLQ automatically prevents you from leaving out your join criteria. If you select two tables and don't specify how to join them, CA OLQ issues a message reminding you to do so.

Creating your report

Contents

- Step 1 -- Select the type of table (see page 39)
- Step 2 -- Select tables (see page 40)
- Step 3 -- Select, project, and join (see page 41)
- Step 4 -- Retrieve the data (see page 41)
- Step 5 -- Display your report (see page 42)

In this example, you create a report from two tables using the following steps:

1. Decide what kind of table you want to report on, a ASF table or an SQL table.
2. Choose two tables.
3. Specify selection criteria that select which rows of data are retrieved for your report.
4. Project which columns you want to appear in your report.
5. Specify join criteria that relate the two tables based on a common column.
6. Retrieve the data for your report.
7. Display your report.

Step 1 -- Select the type of table

When you sign on to CA OLQ, select the Session Options screen:

```
V12 ENTER NEXT TASK CODE: olq menu options
```

For this example, select Y to access SQL tables:

```
CA OLQ Release nn.n
->
107017 CA OLQ Release nn.n
107019 Copyright(C) 2003 CA, Inc.
Current interrupt count: 100
Access IDMS SQL tables: N (Y/N)
*** Session Options ***
Page 1 of 2
User options: Page Columns Spread: (L-Left,E-Even,M-Max,nn)
```

Current underline character: -
Current SQL NULL data value: .
Step 2 -- Select tables

When you sign on to CA OLQ, the first screen you see is the Menu screen. Section 2 describes how to sign on to CA OLQ.

For this example, select Choose tables.

```
CA OLQ Release nn.n
*** Menu ***
->
107017 CA OLQ Release nn.n
107019 Copyright(C) 2003 CA, Inc.
122000 Select an option and press the ENTER key
Select pfkey option description
---> Data Source for Report <---
 X Choose tables TABLE
   _ Choose subschema SUBschema _
   --> Retrieval Activity ----
   _ Choose records from selected subschema RECORD _
   _ Choose columns for report COLUMN _
   _ Retrieve data to build report RETrieve _
   _ Alter database access strategy LINKage _
   --> Processing Mode ----
   _ Execute or create a predefined routine QFILE _
   _ View existing or save current report SAVE _
   _ Submit batch report request BATCH _
1=HELP 2=GLOBAL HELP 3=QUIT 4=MESSAGE 8=FWD
```

Specify SELECT and select the DEPARTMENT and EMPLOYEE tables.
Step 3 -- Select, project, and join

In this step, you perform all three relational operations: select, project, and join:

- **Select** by restricting the rows of the report to those employees hired before 1990.
- **Project** by displaying only the DEPT-NAME, EMP-ID, EMP-LAST-NAME, and START-YEAR columns.
- **Join** by linking the DEPARTMENT and EMPLOYEE tables based on common values of the DEPT-ID field.

Select which columns you want to list in your report, specify that you want to include only those employees hired before 1990, and join the two tables based on common values in the DEPT-ID fields.

```
CA OLQ Release nn.n

124000 Select columns, specify selection criteria and press the ENTER key

Columns Currently Selected:   4 Selection Criteria    Distinct N Y/N
DEPARTMENT
  X 03 DEPT-NAME
  03 DEPT-ID
  03 DEPT-HEAD-ID
  EMPLOYEE
  X 03 EMP-ID
  X 03 EMP-LAST-NAME
  03 EMP-FIRST-NAME
  X 03 START-YEAR    lt 90
  03 DEPT-HEAD-ID
  03 DEPT-ID
  03 SALARY-AMOUNT
  03 PROJECT
  03 OFFICE-CODE    eq department.dept-id

Additional Selection Criteria:

Proceed to Selection Criteria Screen? N Y/N
```

Step 4 -- Retrieve the data

Depending on how high the interrupt count at your site is set, CA OLQ may bypass the Retrieval Interrupted screen and proceed directly to the Retrieval Completed screen.
Step 5 -- Display your report

Your report displays information from both the EMPLOYEE and DEPARTMENT tables.

How to Format Your Report

Using CA OLQ, you can format your report by:

- Sorting your report rows
Changing the external picture of your data columns

Modifying how data columns are displayed

Changing your column headers

Making page headers and footers

You can perform the following steps:

- Format Report Key Terms (see page 43)
- Creating a Report (see page 44)
- Sorting Your Report Rows (see page 54)
- Editing Your Report Values (see page 56)
- Changing Your Column Relative Positions (see page 62)
- Changing Your Column Pictures (see page 63)
- Changing Column Headers (see page 66)
- Making Page Headers and Footers (see page 69)

Format Report Key Terms

Code table

A table that is defined in the data dictionary and that contains corresponding pairs of values. For example:

<table>
<thead>
<tr>
<th>Internal Code</th>
<th>Displayed Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Alabama</td>
</tr>
<tr>
<td>02</td>
<td>Alaska</td>
</tr>
<tr>
<td>03</td>
<td>Arizona</td>
</tr>
</tbody>
</table>
Internal codes are used to efficiently store the data in the database. External values are used to display data in programs or reports.

**Column alignment**

A way to specify how report data columns align under the column headers. Options are left-justified, right-justified, and centered.

**Column header**

A header at the top of each column of report data.

**Display sequence**

Determines the order in which report columns are displayed.

**External picture**

A code that defines the way your column value is formatted in your report. The external picture is used to add punctuation (for example, commas) and special characters (for example, dollar signs) to your column data display.

**Page header**

A title at the top of each page of your report.

**Page footer**

A title at the bottom of each page of your report.

**Sort**

A way to order report rows. CA OLQ sorts the rows in your report based on the value of the sort fields that you specify. Rows can be sorted in ascending or descending order.

**Sparse/Full option**

A CA OLQ editing feature that determines how column values that repeat in consecutive rows are displayed:

- **Sparse** displays only the first of a repeating set of column values.
- **Full** displays all occurrences of the repeating value.

---

**Creating a Report**

**Contents**

- Step 1 -- Select a subschema (see page 45)
- Step 2 -- Select records (see page 46)
- Step 3 -- Choose columns (see page 46)
Before you can apply any report formatting features, you must have a current report. In this example you build a current report with records from a subschema using the following steps:

**Note:** Before you begin, make certain the access switch is set to `olq`.

For more information on setting the access switch, see 3.3.1, "Step 1 -- Select the type of table".

1. Select a subschema.
2. Select the records you want to include in your report.
3. Select which columns you want to display.
4. Retrieve the data from the database and display your report.

**Step 1 -- Select a subschema**

Select the EMPSS01 subschema.

```
CA OLQ Release nn.n           *** Signon Database View ***  
->                      Page  1  OF  1
121000 Select a subschema and press the ENTER key

| Dictionary name . : ASFDICT | Dictionary node name . : |
| Database name . . :         | Database node name . . : |
| Specify Subschema :         | of Schema . . :          | Version : |
| Select subschema:           | of Schema . . :          | Version : |
| EMPSS01 OF EMPSCHM VER 1   | EMPLOYEE DATA            |
| FINANC01 OF EMPSCHM VER 1  | 4Q87                      |
| SALES01 OF EMPSCHM VER 1   | SALES QUOTAS             |

1=HELP 3=QUIT 4=MESSAGE 6=MENU PA2=REFRESH
```
Step 2 -- Select records

Select the DEPARTMENT, EMPLOYEE, and EMPOSITION records.

```
123000 Select records and press the ENTER key

Enter records:

- and/or-
Select records:
  - COVERAGE
  - DEPARTMENT
  - EMPLOYEE
  - EMPOSITION
  - EXPERTISE
  - HOSPITAL-CLAIM
  - INSURANCE-PLAN
  - JOB
  - NON-HOSP-CLAIM
  - OFFICE
  - SKILL
  - STRUCTURE

1=HELP  3=QUIT  4=MESSAGE  6=MENU  PA2=REFRESH
```

Step 3 -- Choose columns


```
124000 Select columns, specify selection criteria and press the ENTER key

Columns Currently Selected: 0

DEPARTMENT
  02 DEPT-ID-0410 *
  02 DEPT-NAME-0410
  02 DEPT-HEAD-ID-0410
  EMPLOYEE
    02 EMP-ID-0415 *
    02 EMP-ID-0415
    02 EMP-ID-0415
    02 EMP-FIRST-NAME-0415 *
    03 EMP-LAST-NAME-0415 *

Enter additional criteria:

Proceed to Selection Criteria Screen? N Y/N
```

1=HELP  3=QUIT  4=MESSAGE  6=MENU  8=FWD  PA2=REFRESH
Step 4 -- Retrieve the data

Specify Yes to continue data retrieval.

```
CA OLQ Release nn.n *** Retrieval Interrupted ***
131000 Select YES or NO and press the ENTER key
Retrieval interrupted due to excessive data base accesses.
Number of whole rows. . . . . . . . 46
Total number of records read. . . . . 100
Total number of records selected. . . 90
Number of data errors . . . . . . . . 0

Continue execution  x Yes
      _ No

Current interrupt interval is 100 data base accesses.
```

Depending on how high the interrupt count at your site is set, CA OLQ may bypass the Retrieval Interrupted screen and proceed directly to the Retrieval Completed screen.

```
CA OLQ Release nn.n *** Retrieval Completed ***
130000 Select activity and press the ENTER key.
Number of whole rows. . . . . . . . 68  Total displayable cols . 20
Total number of records read. . . . . 143  Formatted line length . 372
Total number of records selected. . . 133
Number of data errors . . . . . . . . 0

Select Option ---> Display/Format Activity <--- Command/ Screen Name
  X  Display report   DISplay
  _  Save report     SAVE
  _  Choose the sort sequence of report SORT
  _  Change column headers HEAder
  _  Change page header and footer PAGE HEAder
  _  Change display format of data ($,commas) PICTURE
  _  Format columns (Alignment, sparse) EDIT
  _  Specify summary computations (Totals) GROUP BY
  _  Send the report to a printer PRINT

1=HELP  3=QUIT  4=MESSAGE  6=MENU
```

Step 5 -- Display your report

Your report displays the data as it is retrieved from the database.
Step 1 - Create a Current Report

When you save a report CA OLQ takes the set of commands you used to build your current report and saves them in the data dictionary. In this step, you build a current report.

This report uses the EMPLOYEE and EMPOSITION records from the sample employee database. These records reside in the EMPSS01 subschema.

Start on the Signon Database View screen. To get there, type `subschema` (or sub) in the command line of any screen.

Select the EMPSS01 subschema.
Select the EMPLOYEE and EMPOSITION records.


Select Yes to continue data retrieval.

Retrieval interrupted due to excessive data base accesses.

Number of whole rows: 54
Total number of records read: 100
Total number of records selected . . 99
Number of data errors . . . . 0

Continue execution  x Yes
X No

Current interrupt interval is 100 data base accesses.

Select Option Command/ Screen Name
X Display report DISplay
- Save report SAVE
- Choose the sort sequence of report SORT
- Change column headers HEAder
- Change page header and footer PAGE HEAder
- Change display format of data ($,commas) PICTURE
- Format columns (Alignment, sparse) EDIT
- Specify summary computations (Totals) GROup BY
- Send the report to a printer PRInt

This is how your report looks before you add any formatting features.

EMPLOYEE/EMPOSITION REPORT
mm/dd/yy

LINGER 092345812 33 38500.00
TERNER 045672222 11 13000.00
BROWN 019556712 44 42500.00
CHARLES 019556712 43 38000.00
PENMAN 014593186 33 39000.00
DUNCAN 010673343 72 85000.00
EVERETT 010673343 71 75000.00
LITERATA 023567831 43 37500.00
WILCO 111000023 72 80000.00
HEAROWITZ 031896154 42 33000.00
TYRO 019893456 21 20000.00
KAHALLY 029661234 21 20000.00

This is how your report looks before you add any formatting features.

1=HELP 3=QUIT 4=MESSAGE 6=MENU 8=FWD 10=LEFT 11=RIGHT
Step 2 - Modify Your Report Headers

In this step, you modify your report headers. Start on the Report Format - Header screen. To get there, type `header` in the command line.

Change your report headers to make them more legible.

```
CA OLQ Release nn.n
->
134000 Specify column headers and press the ENTER key

Underline character: -            Disp   Seq   Header
EMPLOYEE
X EMP-LAST-NAME-0415  1  name
X SS-NUMBER-0415   2 social security number
EMPOSITION
X SALARY-GRADE-0420  3 salary grade
X SALARY-AMOUNT-0420  4 salary

Compute:
1=HELP  3=QUIT  4=MESSAGE  5=DISPLAY  6=MENU  10=SORT  11=EDIT
```

CA OLQ displays your new report headers.

```
CA OLQ Release nn.n
->
125000 Press the ENTER key to go to the next page of the report.

EMPLOYEE/EMPOSITION REPORT
mm/dd/yy

<table>
<thead>
<tr>
<th>NAME</th>
<th>SOCIAL SECURITY NUMBER</th>
<th>SALARY GRADE</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINGER</td>
<td>092345812</td>
<td>33</td>
<td>38500.00</td>
</tr>
<tr>
<td>TERNER</td>
<td>045672222</td>
<td>11</td>
<td>13000.00</td>
</tr>
<tr>
<td>BROWN</td>
<td>019556712</td>
<td>44</td>
<td>42500.00</td>
</tr>
<tr>
<td>CHARLES</td>
<td>019556712</td>
<td>43</td>
<td>38000.00</td>
</tr>
<tr>
<td>PENMAN</td>
<td>014593186</td>
<td>33</td>
<td>39000.00</td>
</tr>
<tr>
<td>DUNCAN</td>
<td>018673343</td>
<td>72</td>
<td>85000.00</td>
</tr>
<tr>
<td>EVERETT</td>
<td>010673343</td>
<td>71</td>
<td>75000.00</td>
</tr>
<tr>
<td>LITERATA</td>
<td>023567831</td>
<td>43</td>
<td>37500.00</td>
</tr>
<tr>
<td>WILCO</td>
<td>111000023</td>
<td>72</td>
<td>80000.00</td>
</tr>
<tr>
<td>HEAROWITZ</td>
<td>031896154</td>
<td>42</td>
<td>33000.00</td>
</tr>
</tbody>
</table>

1=HELP  3=QUIT  4=MESSAGE  5=DISPLAY  6=MENU  8=FWD  10=LEFT  11=RIGHT
```
Step 3 - Change Your Report Column Pictures

In this step, you modify the display of the social security and salary columns to make them more legible. Start on the Report Format - Picture screen. To get there, type `picture` in the command line.

Change the picture for SS-NUMBER-0415 to include hyphens.

```
CA OLQ Release nn.n *** Report Format - Picture ***
Page 1 of 1
137000 Specify pictures and press the ENTER key

EMPLOYEE
X EMP-LAST-NAME-0415        1   _   _   _
X SS-NUMBER-0415            2   _   _   _ 99-999-9999

EMPOSITION
X SALARY-GRADE-0420         3   _   _   _ 99
X SALARY-AMOUNT-0420        4   _   _   _ -ZZZZZ9.99

Compute:
1=HELP   3=QUIT   4=MESSAGE   5=DISPLAY   6=MENU   10=EDIT   11=GROUP BY

CA OLQ displays your new picture.
```

```
CA OLQ Release nn.n *** Display Report ***
Page 1 Line 1
125000 Press the ENTER key to go to the next page of the report.

EMPLOYEE/EMPOSITION REPORT
mm/dd/yy

<table>
<thead>
<tr>
<th>NAME</th>
<th>SOCIAL SECURITY NUMBER</th>
<th>SALARY GRADE</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINGER</td>
<td>09-234-5812</td>
<td>33</td>
<td>38500.00</td>
</tr>
<tr>
<td>TERNER</td>
<td>04-567-2222</td>
<td>11</td>
<td>13000.00</td>
</tr>
<tr>
<td>BROWN</td>
<td>01-955-6712</td>
<td>44</td>
<td>42500.00</td>
</tr>
<tr>
<td>CHARLES</td>
<td>01-955-6712</td>
<td>43</td>
<td>38000.00</td>
</tr>
<tr>
<td>PENMAN</td>
<td>01-459-3186</td>
<td>33</td>
<td>39000.00</td>
</tr>
<tr>
<td>DUNCAN</td>
<td>01-067-3343</td>
<td>72</td>
<td>85000.00</td>
</tr>
<tr>
<td>EVERETT</td>
<td>01-067-3343</td>
<td>71</td>
<td>75000.00</td>
</tr>
<tr>
<td>LITERATA</td>
<td>02-356-7831</td>
<td>43</td>
<td>37500.00</td>
</tr>
<tr>
<td>WILCO</td>
<td>11-100-0023</td>
<td>72</td>
<td>80000.00</td>
</tr>
<tr>
<td>HEAROWITZ</td>
<td>03-189-6154</td>
<td>42</td>
<td>33000.00</td>
</tr>
</tbody>
</table>

- 1 -
1=HELP   3=QUIT   4=MESSAGE   6=MENU   8=FWD   10=LEFT   11=RIGHT
```
Step 4 - Sort Your Report Rows and Specify Group Calculations

In this step, you sort your report rows by salary grade, and compute the average salary in each salary grade. Start on the Report Format - Sort screen. To get there, type sort in the command line.

Sort your rows in ascending order by salary grade; group your report rows by salary grade. Sort within salary grade by salary amount.

For each salary grade group, calculate the average salary amount.

CA OLQ displays the average salary for each salary grade group.
Sorting Your Report Rows

Contents

- Step 1 - Specify your sort criteria (see page 54)
- Step 2 - Display your report (see page 55)
- Step 1 - Sort your report (see page 55)
- Step 2 - Display your report (see page 56)

In this example, you use the Report Format - Sort screen to sort your report rows. To get there, type sort in the command line.

Step 1 - Specify your sort criteria

To sort the rows in ascending order by department ID, specify 1 in the Sort Priority column and a in the Order (A/D) column next to DEPT-ID-0410. And to sort each department’s employees in ascending order by employee name, specify 2 in the Sort Priority column and an a in the Order (A/D) column next to EMP-LAST-NAME-0415.

---

Sorted report rows:

<table>
<thead>
<tr>
<th>NAME</th>
<th>SOCIAL SECURITY NUMBER</th>
<th>GRADE</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>FITZHUGH</td>
<td>11-234-5678</td>
<td>11</td>
<td>13000.00</td>
</tr>
<tr>
<td>TERNER</td>
<td>04-567-2222</td>
<td>11</td>
<td>13000.00</td>
</tr>
<tr>
<td>JOHNSON</td>
<td>01-134-7878</td>
<td>11</td>
<td>13500.00</td>
</tr>
<tr>
<td>NICEMAN</td>
<td>03-345-6110</td>
<td>12</td>
<td>14000.00</td>
</tr>
<tr>
<td>GARDNER</td>
<td>02-233-4444</td>
<td>12</td>
<td>14000.00</td>
</tr>
<tr>
<td>KRAAMER</td>
<td>02-378-6666</td>
<td>12</td>
<td>14000.00</td>
</tr>
<tr>
<td>KING</td>
<td>06-784-5516</td>
<td>12</td>
<td>14500.00</td>
</tr>
</tbody>
</table>

AVE FOR 11: 13166.66

---

User action prompts:

1=HELP 3=QUIT 4=MESSAGE 6=MENU 8=FWD 10=LEFT 11=RIGHT
Step 2 - Display your report

The rows are sorted in ascending order by department ID, and within each department by employee last name.

Step 1 - Sort your report

Start on the Report Format - Sort screen. To get there, type SORT in the command line.

Use **Disp Seq** to change the column order to reflect the sort priority. Use **Sort Priority** to specify that you want to sort the report rows by DEPT-ID (level 1), within each department by PROJECT (level 2), and within each project by SALARY-AMOUNT (level 3). Use **Order (A/D)** to specify that you want all of the sorts to be in ascending order. Press [PF5] to display your report.
### Step 2 - Display your report

The rows are sorted by DEPT-ID, within each department by PROJECT, and within each project by SALARY-AMOUNT.

<table>
<thead>
<tr>
<th>DEPT-ID</th>
<th>PROJECT</th>
<th>SALARY-AMOUNT</th>
<th>EMP-LAST-NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>4000</td>
<td>EVALUATE</td>
<td>37000.00</td>
<td>ZEDI</td>
</tr>
<tr>
<td></td>
<td>PLANNING</td>
<td>18000.00</td>
<td>MCDOUGALL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>25000.00</td>
<td>ANGELO</td>
</tr>
<tr>
<td></td>
<td></td>
<td>34000.00</td>
<td>JACKSON</td>
</tr>
<tr>
<td></td>
<td></td>
<td>39000.00</td>
<td>PENMAN</td>
</tr>
<tr>
<td>34000.00</td>
<td>TESTING</td>
<td>80000.00</td>
<td>BANK</td>
</tr>
<tr>
<td>5200</td>
<td>EVALUATE</td>
<td>31000.00</td>
<td>KASPAR</td>
</tr>
<tr>
<td></td>
<td>RESEARCH</td>
<td>33000.00</td>
<td>TIME</td>
</tr>
<tr>
<td></td>
<td></td>
<td>45000.00</td>
<td>FINN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>80000.00</td>
<td>WILCO</td>
</tr>
</tbody>
</table>

**Additional editing features**

In this example, you use the Report Format - Edit screen to:

- Suppress the display of repeating values in a given column
- Align the data under a column header (left-justified, center, right-justified)
• Change the order in which your columns are displayed

135000 Specify edit options and press the ENTER key.

<table>
<thead>
<tr>
<th></th>
<th>Disp</th>
<th>Sparse</th>
<th>Hex</th>
<th>Align</th>
<th>Code</th>
<th>Table</th>
<th>Ver</th>
<th>Len</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEPARTMENT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X DEPT-ID-0410</td>
<td>1</td>
<td>_</td>
<td>_</td>
<td>RIGHT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMPLOYEE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP-LAST-NAME-0415</td>
<td>2</td>
<td>_</td>
<td>_</td>
<td>LEFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMPOSITION</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SALARY-AMOUNT-0420</td>
<td>3</td>
<td>_</td>
<td>_</td>
<td>RIGHT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Compute:

1=HELP  3=QUIT  4=MESSAGE  5=DISPLAY  6=MENU  10=HEADER  11=PICTURE

Start on the Report Format - Edit screen. To get there, type EDIT in the command line of any screen and press [Enter].

Suppressing repeating values

When a column displays the same value for many consecutive rows, the report can become hard to read. To suppress repeating values in a column, use the Sparse option on the Report Format - Edit screen:

If you don't use Sparse: If you use Sparse:

DEPT EMPLOYEE PHONE DEPT EMPLOYEE PHONE

003 Sam 1250 003 Sam 1250
003 Elmer 3000 Elmer 3000
003 Leo 4110 Leo 4110
005 Jack 3092 005 Jack 3092
005 Gus 5555 Gus 5555

To restore a column to full display, type a space under Sparse next to that column.

Sparse/Full

Another way to suppress repeating values is to use the Sparse and Full options on the Session Options screen.

• Sparse displays only the first of a repeating set of column values.

• Full displays all occurrences of the repeating value.

To get to the Session Options screen, type OPTIONS in the command line and press [Enter].
When you specify the Sparse/Full option on the **Session Options screen**, the display of repeating values is suppressed throughout the entire report.

### Aligning columns

You can modify how the columns of data align under their headers by using the **Align** field. For example:

Using LEFT: Using CENTER: Using RIGHT:

```plaintext
BONUS-PERCENT BONUS-PERCENT BONUS-PERCENT
-------------- ────────────── ──────────────
03 03 03
02 02 02
06 06 06
```

To specify how you want data to align under the column headings, type LEFT, CENTER, or RIGHT under **Align** next to that column.

### Changing the column order

Use the **Display Sequence** field to specify the order in which columns are to appear. For each column, type a number to indicate the new position for that column. For example:

<table>
<thead>
<tr>
<th>Disp Seq</th>
<th>NAME</th>
<th>ID</th>
<th>PHONE</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>EMP-LAST-NAME-0415</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

---

### Step 1 - Specify your edit criteria

Select **Sparse** next to DEPT-ID-0410. And to center a column’s data under its header, specify **center** for that column under **Align**.

```plaintext
DEPARTMENT          Disp Seq Sparse Hex Align Code Table Ver Len
X DEPT-ID-0410      1 x _ center
X EMPLOYEE          2 _ _ LEFT
X EMP-LAST-NAME-0415 _ _ _
X SALARY-AMOUNT-0420 _ _ _ center
```
Step 2 - Display your report

Each department ID number is listed only once. Data in the DEPT-ID-0410 and SALARY-AMOUNT-0420 are centered under their headers.

<table>
<thead>
<tr>
<th>DEPT-ID-0410</th>
<th>EMP-LAST-NAME-0415</th>
<th>SALARY-AMOUNT-0420</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>FITZHUGH</td>
<td>13000.00</td>
</tr>
<tr>
<td></td>
<td>JOHNSON</td>
<td>13500.00</td>
</tr>
<tr>
<td></td>
<td>ORGRATZI</td>
<td>39000.00</td>
</tr>
<tr>
<td></td>
<td>PEOPLES</td>
<td>80000.00</td>
</tr>
<tr>
<td>2000</td>
<td>BLOOMER</td>
<td>15000.00</td>
</tr>
<tr>
<td></td>
<td>HUTTON</td>
<td>44000.00</td>
</tr>
<tr>
<td></td>
<td>JENSON</td>
<td>82000.00</td>
</tr>
<tr>
<td></td>
<td>KIMBALL</td>
<td>45000.00</td>
</tr>
<tr>
<td></td>
<td>KING</td>
<td>14500.00</td>
</tr>
<tr>
<td></td>
<td>NICEMAN</td>
<td>14000.00</td>
</tr>
<tr>
<td>3100</td>
<td>DOUGH</td>
<td>33000.00</td>
</tr>
<tr>
<td></td>
<td>GALLWAY</td>
<td>33000.00</td>
</tr>
</tbody>
</table>

Additional editing features

Using the Report Format - Edit screen, you can also:

- Display a column in hexadecimal notation
- Specify a code table used to edit a column
- Remove columns from the report (or bring back columns that you have removed)
- Assign fixed column positions in your report

Using hexadecimal notation

When you see the sign (@) in a column value, means that invalid data has been retrieved from the database. To see the actual value of the bad data, you can display the column in hexadecimal form. Once you have done this, report the value of the column to your support staff.

To display a column in hexadecimal form, type any character under **Hex** next to that column:

**Normal Display Hexadecimal Form JOE@ X'D1D6C5AC'**
To change the column from hexadecimal format back to normal display, type a space under **hex** next to that column.

**Editing with code tables**

Code tables are tables of values that have been defined in the data dictionary. CA OLQ uses these tables to translate internal codes from the database into a more meaningful external expression of the data.

Here's how the Credit Rating code table looks in the dictionary:

<table>
<thead>
<tr>
<th>Encoded Value</th>
<th>Decoded Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Credit O.K.</td>
</tr>
<tr>
<td>02</td>
<td>Rejected</td>
</tr>
<tr>
<td>03</td>
<td>Review</td>
</tr>
<tr>
<td>etc.</td>
<td>etc.</td>
</tr>
</tbody>
</table>

**What to do**

If your report contains a column of encoded values, you can tell CA OLQ to edit that column with a code table. Type the name of the appropriate code table under **Code Table** next to that column. You may also need to type a version number for that code table in the **Ver** entry. If you are unsure of the code table name (or version number) ask your support staff or database administrator (DBA).

**Built-in code tables for columns**

Some columns have code tables built into them. CA OLQ indicates these columns by displaying the following under **Code Table**:

**DICT**

If you want to use a different code table, you can type the name of the other code table over **DICT**.

If you don’t want to use any code table, type spaces over **DICT**.

**Removing columns from your report**

To remove a column from your report, type a space or the letter **d** (for delete) in front of that column. For example, to remove the PHONE and BONUS columns from your report, specify:

```
X NAME
X ID
D PHONE
X SALARY
D BONUS
```

**Bringing a column back**

To redisplay a column that you have removed, type any character (other than the letter **d**) in front of that column.

**Note:** You cannot redisplay computed columns. Once you have removed them from the report, they cannot be restored.
Fixing columns in place

You can also use the display sequence field to fix one or more columns in position on the Display Report screen. A fixed column always remains on the screen when you scroll right or left.

For example, you can fix the display of EMP-ID so that it always displays at the left margin of your report, no matter how many times you page right or left.

On your report, fixed columns always appear first (before the regular numbered columns). If you have more than one fixed column, you can switch their positions by typing F1, F2, and so on.
Changing Your Column Relative Positions

You can change how your columns are placed relative to one another by using the **Page columns spread** option on the Session Options screen.

```
CA OLQ Release nn.n
-> *** Session Options ***
141000 Select options to be changed and press the ENTER key

Current interrupt count: 100
Current underline character:

User options: Page columns spread: (L-Left,E-Even,M-Max,nn)

<table>
<thead>
<tr>
<th>Help</th>
<th>Change</th>
<th>Option</th>
<th>Current option</th>
<th>Alternate option</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

-> Report Processing Options <-

- _ _ NOFiller Filler
- _ _ FULL SPArse
- _ _ HEAder NOHeader

-> Column Processing Options <-

- _ _ OLQheader NOOlqheader
- _ _ PCtive NOPicture
- _ _ CODetable NOCODetable

1=HELP 3=QUIT 4=MESSAGE 6=MENU 8=FWD

Even

To evenly distribute the space between your report columns, specify **E** (for even):

```
Employee/Emposision Report

<table>
<thead>
<tr>
<th>First Name</th>
<th>Last Name</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOUGLAS</td>
<td>KAHALLY</td>
<td>20000</td>
</tr>
<tr>
<td>THEMIS</td>
<td>PAPAZEUS</td>
<td>100000</td>
</tr>
<tr>
<td>HARRY</td>
<td>ARM</td>
<td>46000</td>
</tr>
<tr>
<td>DORIS</td>
<td>KING</td>
<td>14500</td>
</tr>
<tr>
<td>BETH</td>
<td>CLOUD</td>
<td>52750</td>
</tr>
<tr>
<td>HENRIETTA</td>
<td>HENDON</td>
<td>240000</td>
</tr>
</tbody>
</table>
```

Left

To group your columns to the left of the page, specify **L** (for left):

```
Employee/Emposision Report

<table>
<thead>
<tr>
<th>First Name</th>
<th>Last Name</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOUGLAS</td>
<td>KAHALLY</td>
<td>20000</td>
</tr>
<tr>
<td>THEMIS</td>
<td>PAPAZEUS</td>
<td>100000</td>
</tr>
<tr>
<td>HARRY</td>
<td>ARM</td>
<td>46000</td>
</tr>
<tr>
<td>DORIS</td>
<td>KING</td>
<td>14500</td>
</tr>
<tr>
<td>BETH</td>
<td>CLOUD</td>
<td>52750</td>
</tr>
<tr>
<td>HENRIETTA</td>
<td>HENDON</td>
<td>240000</td>
</tr>
</tbody>
</table>
```

Max
To display the maximum amount of space between your report columns, specify **M** (for max):

```
First Name       Last Name       Salary
----------       ----------       -------
DOUGLAS         KAHALLY         20000.00
THEMIS          PAPAZEUS        100000.00
HARRY           ARM              46000.00
DORIS           KING             14500.00
BETH            CLOUD            52750.00
HENRIETTA       HENDON           240000.00
```

**Number of spaces**

To display a specific number of spaces between your report columns, specify that number. For example, if you specify a **4**, CA OLQ spaces the columns four spaces apart:

```
First Name       Last Name       Salary
----------       ----------       -------
DOUGLAS         KAHALLY         20000.00
THEMIS          PAPAZEUS        100000.00
HARRY           ARM              46000.00
DORIS           KING             14500.00
BETH            CLOUD            52750.00
HENRIETTA       HENDON           240000.00
```

---

## Changing Your Column Pictures

**Contents**
- Step 1 - Modify a column picture (see page 65)
- Step 2 - Display your report (see page 66)

In this example, you use the Report Format - Picture screen to change the format of data in your report columns.

```
CA OLQ Release nn.n           *** Report Format - Picture ***
->                       Page 1 of 1
137002 The command you have specified is invalid for this screen
```

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Select Options</th>
<th>Alter Picture</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEPARTMENT</td>
<td>X DEPT-10-0410</td>
<td>$</td>
</tr>
<tr>
<td>EMPLOYEE</td>
<td>X EMP-LAST-NAME-0415</td>
<td>,</td>
</tr>
<tr>
<td>EMPOSITION</td>
<td>X SALARY-AMOUNT-0420</td>
<td>0</td>
</tr>
</tbody>
</table>

```
What is a picture?

A picture is a code that represents how the data in a column is displayed. You can format your column data by defining a picture for that column.

For example, when numeric data appears on the report, you create different pictures to display it in different ways:

- With a dollar sign: $500  Or without: 500
- With commas: 65,000  Or without: 65000
- With leading zeros: 001229  Or without: 1229

You can also combine these formats:

$65,000 001,229

How do you specify a picture?

To specify a column picture, use either of the following fields (but not both):

- Select Options -- If you want a fast, easy way to format numeric data.
- Alter Picture -- If you require more complex formatting for your data.

Select Options

These three columns let you do basic formatting of numeric data. In most cases, these columns are all you need to define pictures for your report columns. You can choose one or more of these options:

$ displays a floating dollar sign
, displays a comma between every three digits (left of the decimal)
0 displays leading zeros

Alter picture

This option lets you format a column by changing its picture. To change a picture, type over the default value that appears in the Alter Picture entry. For example, suppose you have a column whose picture currently looks like this:

9999.99

To display a fixed dollar sign in this column, you could change the picture to this:

$9999.99

Picture symbols
### Symbol | What it Does
--- | ---
X | Stands for a single alphanumeric character
A | Stands for a single alphabetic character
9 | Stands for a single numeric character
Z | Stands for a numeric character and suppresses leading zeros
$ | Stands for a numeric character and displays a floating dollar sign
. | Displays the decimal point in that position
+ | Stands for a numeric character and displays + for positive values
- | Stands for a numeric character and displays - for negative values
B | Displays a blank character (a space) in that position
* | Requests check protection. Leading zeros are displayed as asterisks.

### Examples

| If you want your column to look like ... | and the data is stored in the database as ... | then specify this picture ...
--- | --- | ---
123400M | 123400M | X(7)
1 23400 M | 123400M | XBXXXXXBX
JOHN | JOHNSON | A(4)
TWO WORDS | TWOWORDS | A(3)BA(5)
2350000 | 2350000 | 9(7)
2350000.00 | 2350000 | 9(7).99
$2,350,000.00 | 2350000 | $$,$$$,$$9.99
23/50/000 | 2350000 | /999/999
120 | 00120 | ZZZZZZ
+9876 | 9876 | ++99

### Step 1 - Modify a column picture

To add a leading dollar sign, select the $ option for SALARY-AMOUNT-0420. To insert commas, select the , option for SALARY-AMOUNT-0420.

```
137000 Specify pictures and press the ENTER key
```

<table>
<thead>
<tr>
<th>DEPARTMENT</th>
<th>EMPLOYEE</th>
<th>EMP-NAME-0415</th>
<th>EMPOSITION</th>
<th>SALARY-AMOUNT-0420</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEPT-ID-0410</td>
<td>1</td>
<td>_</td>
<td>_</td>
<td>9999</td>
</tr>
<tr>
<td>EMLOYEE</td>
<td>2</td>
<td></td>
<td></td>
<td>X(15)</td>
</tr>
<tr>
<td>EMP-NAME-0415</td>
<td>3</td>
<td>x</td>
<td>x</td>
<td>-ZZZZZ9.99</td>
</tr>
</tbody>
</table>
Step 2 - Display your report

DEPARTMENT/EMPLOYEE/EMPLACEMENT REPORT

DEPT-ID-0410 EMP-LAST-NAME-0415 SALARY-AMOUNT-0420
1000 FITZHUGH $13,000.00
JOHNSON $13,500.00
ORGRATZI $39,000.00
PEOPLES $80,000.00
2000 BLOOMER $15,000.00
HUTTON $44,000.00
JENSON $82,000.00
KIMBALL $45,000.00
KING $14,500.00
NICEMAN $14,000.00
3100 DOUGH $33,000.00
GALLWAY $33,000.00

Changing Column Headers

Contents

- Step 1 - Change your column headers (see page 68)
- Step 2 - Display your report (see page 68)

In this step, you use the Report Format - Header screen to:

- Change the text of the column headers that appear on the report
- Underline column headers

To get to the Report Format - Header screen, type **header** in the command line of any screen.
Compute:

1=HELP    3=QUIT    4=MESSAGE    5=DISPLAY    6=MENU    10=SORT    11=EDIT

Default column headers

When you build a report, CA OLQ automatically creates column headers for you. These default headers come from the names of the columns in the database or from CA OLQ headers defined in the dictionary.

Changing the defaults

To change the default column headers, type over the text in the Header entry. You can change any or all of the headers in the report. Each header you type can be from 1 to 37 characters long.

Typing multiword headers

You can type headers that contain more than one word. To make a multiword header appear on a single line, type quotes around it. Otherwise, each word of the header appears on a separate line of the report. For example:

Input:

Last Name "Last Name"

Output:

Last Name Last Name

Jones Jones
Smith Smith
Whipple Whipple

Session Options That Affect Report Headers:

<table>
<thead>
<tr>
<th>Session Option</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>CA OLQ displays column headers on your report</td>
</tr>
<tr>
<td>Noheader</td>
<td>CA OLQ displays the report without headers (even if you’ve created new headers on the Report Format - Header screen)</td>
</tr>
<tr>
<td>Olqheader</td>
<td>CA OLQ uses any headers you’ve typed on the Report Format - Header screen (or that have been created in the data dictionary)</td>
</tr>
</tbody>
</table>
Session Option | Function
--- | ---
Noolqhea | CA OLQ ignores your headers and uses the column names from the database as the headers on the report

To get to the Session Options screen, type `options` on the command line and press [Enter].

**Underlining column headers**

Underlining column headers can make them easier to read. CA OLQ uses the dash (−) character to draw these lines.

To change the underline character, type the character you want to use next to `Underline Character`. If you don’t want underlining on your report, type a space here.

**Step 1 - Change your column headers**

Change your column headers to make them more legible.

134000 Specify column headers and press the ENTER key

Underline character: Disp Seq Header

DEPARTMENT
X DEPT-ID-0410 1 'Department Id'
EMPLOYEE
X EMP-LAST-NAME-0415 2 Name
EMPOSITION
X SALARY-AMOUNT-0420 3 Salary

Compute:

1=HELP 3=QUIT 4=MESSAGE 5=DISPLAY 6=MENU 10=SORT 11=EDIT

**Step 2 - Display your report**

125000 Press the ENTER key to go to the next page of the report.

DEPARTMENT/EMPLOYEE/EMPOSITION REPORT

<table>
<thead>
<tr>
<th>DEPARTMENT ID</th>
<th>NAME</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>FITZHUGH</td>
<td>$13,000.00</td>
</tr>
<tr>
<td></td>
<td>JOHNSON</td>
<td>$13,500.00</td>
</tr>
<tr>
<td></td>
<td>ORGRATZI</td>
<td>$39,000.00</td>
</tr>
<tr>
<td></td>
<td>PEOPLES</td>
<td>$80,000.00</td>
</tr>
</tbody>
</table>
## Making Page Headers and Footers

### Contents

- Step 1 - Specify page headers and footers (see page 73)
- Step 2 - Display your report (see page 73)

This example shows you how to use the Page Header/Footer screen to:

- Specify page header or footer text
- Include variable values (for example, the report date)
- Specify the alignment of page header or footer elements (right-justified, left-justified, or centered)
- Skip lines before and after the page header or footer
- Put more than one heading element on a line

To get to the Page Header/Footer screen, type `page` in the command line and press [Enter].

```
CA OLQ Release nn.n            *** Page Header/Footer ***
->                        Page 1  OF  1
152000 Specify page header(s), footer(s) and press the ENTER key
```

Format for $DATE: MM/DD/YY
Use variables: $DATE, $TIME, $PAGE, $LINE, $USER...
Skip lines before heading: 0        Skip lines after heading: 1

<table>
<thead>
<tr>
<th>Line</th>
<th>Page heading text</th>
<th>Align</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DEPARTMENT/EMPLOYEE/EMPOSITION REPORT</td>
<td>CENTER</td>
</tr>
<tr>
<td>2</td>
<td>$DATE</td>
<td>CENTER</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>CENTER</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>CENTER</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>CENTER</td>
</tr>
</tbody>
</table>

Skip lines before footing: 1        Skip lines after footing: 0

<table>
<thead>
<tr>
<th>Line</th>
<th>Page footing text</th>
<th>Align</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$PAGE</td>
<td>CENTER</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>CENTER</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>CENTER</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>CENTER</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>CENTER</td>
</tr>
</tbody>
</table>

### Default page headers and footers
When you build a new report, CA OLQ automatically adds a standard page header and page footer to each page of the report. The page header includes the names of the tables or records used and the current date:

EMPLOYEE REPORT

mm/dd/yy

The page footer displays the number of each report page:

- 1 -

**Note:** You specify page headers and footers in the same way. In this discussion, the examples will illustrate only page headers.

### Specifying text

To change the text in your header, type your text over the default text under **Page heading text**:

**Input:**

<table>
<thead>
<tr>
<th>Line</th>
<th>Page heading text</th>
<th>Align</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Widget Inventory</td>
<td>center</td>
</tr>
<tr>
<td>2</td>
<td>$date</td>
<td>center</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>center</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>center</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>center</td>
</tr>
</tbody>
</table>

**Output:**

Widget Inventory

mm/dd/yy

### Using variables

CA OLQ provides variables that you can include in your headers. These variables are placeholders for which CA OLQ substitutes real values (such as a page number).

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>$DATE</td>
<td>Displays the current date using the user-specified date format</td>
<td>10/24/99</td>
</tr>
<tr>
<td>$EDATE</td>
<td>Displays the current date (European format)</td>
<td>24/10/99</td>
</tr>
<tr>
<td>$JDATE</td>
<td>Displays the current date (Julian format)</td>
<td>99297</td>
</tr>
<tr>
<td>$TIME</td>
<td>Displays the current time</td>
<td>10:30:59</td>
</tr>
<tr>
<td>$PAGE</td>
<td>Displays the current page number</td>
<td>- 12 -</td>
</tr>
<tr>
<td>$USER</td>
<td>Displays the user ID (of the report’s creator)</td>
<td>JFK</td>
</tr>
<tr>
<td>$LINE</td>
<td>Displays the current line number</td>
<td>33</td>
</tr>
</tbody>
</table>

### Changing text alignment

To change how your header is aligned, specify LEFT, RIGHT, or CENTER under **Align**:

- **Left** left-justifies the heading text.
- **Right** right-justifies the heading text.
- **Center** centers the heading text.

For example:

**Input:**

<table>
<thead>
<tr>
<th>Line</th>
<th>Page footing text</th>
<th>Align</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Agricultural Budget Report</td>
<td>right</td>
</tr>
<tr>
<td>2</td>
<td>Central Region</td>
<td>right</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>CENTER</td>
</tr>
<tr>
<td>4</td>
<td>Page $page</td>
<td>right</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>CENTER</td>
</tr>
</tbody>
</table>

**Output:**

Agricultural Budget Report
Central Region
Page 22

**Leaving blank lines before your text**

To add to the top margin of your report, enter a number next to **Skip lines before heading**. If you type 0, CA OLQ will leave no blank lines before it prints the page header. To leave one or more blank lines, type a number from 1 through 9.

**Leaving blank lines after your text**

To provide space after your page header, enter a number next to **Skip Lines After Heading**. If you type 0, CA OLQ will leave no blank lines after it prints the page header. To leave one or more blank lines, type a number from 1 through 9.

**Putting more than one text element on a line**

To list more than one text element on a line, specify the same number in the **Line** entry for both elements. For example:

**Input:**

<table>
<thead>
<tr>
<th>Line</th>
<th>Page heading text</th>
<th>Align</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Industrial Gizmos Inventory</td>
<td>CENTER</td>
</tr>
<tr>
<td>1</td>
<td>$date</td>
<td>right</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>CENTER</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>CENTER</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>CENTER</td>
</tr>
</tbody>
</table>

**Output:**

Industrial Gizmos Inventory
10/17/99

**Session Options That Affect Page Headers and Footers:**

<table>
<thead>
<tr>
<th>Session Option</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header</td>
<td>CA OLQ displays page headers on your report.</td>
</tr>
<tr>
<td>Noheader</td>
<td></td>
</tr>
</tbody>
</table>
CA OLQ displays the report without page headers (even if you've created new headers on the Report Format - Page Header/Footer screen).

Olqhea CA OLQ uses any headers you've typed on the Report Format - Header screen (or that have been created in the data dictionary).

Noolqhea CA OLQ ignores your headers and uses the column names from the database as the headers on the report.

To get to the Session Options screen, type **options** on the command line of any screen and press [Enter].

**Specifying the date**

The **Format for $DATE**: field displays the report date format. Valid date formats are listed in the following table.

### Specifying a Date Format on the Page Header/Footer Screen:

<table>
<thead>
<tr>
<th>Format</th>
<th>Explanation</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>MONT H</td>
<td>Displays the full month name in uppercase.</td>
<td>JANUARY</td>
</tr>
<tr>
<td>Month</td>
<td>Displays the first letter of the month name in uppercase and the rest of the month name in lowercase.</td>
<td>January</td>
</tr>
<tr>
<td>month</td>
<td>Displays the full month name in lowercase.</td>
<td>january</td>
</tr>
<tr>
<td>MON</td>
<td>Displays the first three letters of the month name in uppercase.</td>
<td>JAN</td>
</tr>
<tr>
<td>Mon</td>
<td>Displays the first three letters of the month name. The first letter is uppercase; the last two letters are lowercase.</td>
<td>Jan</td>
</tr>
<tr>
<td>mon</td>
<td>Displays the first three letters of the month name in lowercase.</td>
<td>jan</td>
</tr>
<tr>
<td>MM</td>
<td>Displays a zero-significant numeric representation of the month.</td>
<td>January is represented as 01.</td>
</tr>
<tr>
<td>ZM</td>
<td>Displays a zero-suppressed numeric representation of the month.</td>
<td>January is represented as 1.</td>
</tr>
<tr>
<td>DD</td>
<td>Displays a zero-significant numeric representation of the day.</td>
<td>The 3rd is represented as 03.</td>
</tr>
<tr>
<td>ZD</td>
<td>Displays a zero-suppressed numeric representation of the month.</td>
<td>The 3rd is represented as 3.</td>
</tr>
<tr>
<td>YY</td>
<td>Displays a two-digit representation of the year.</td>
<td>1999 is represented as 99.</td>
</tr>
<tr>
<td>YYYY</td>
<td>Displays a four-digit representation of the year.</td>
<td>1999 is represented as 1999.</td>
</tr>
<tr>
<td>CC</td>
<td>Displays a two-digit representation of the century.</td>
<td>20</td>
</tr>
</tbody>
</table>
Step 1 - Specify page headers and footers

Specify the text you want displayed at the top of each page of your report and the alignment that you prefer. Add Created by as part of your report footer. Include the $user variable to display the user ID of the person who created the report.

```
CA OLQ Release nn.n
->  *** Page Header/Footer ***
152000 Specify page header(s), footer(s) and press the ENTER key

Format for $DATE: MM/DD/YY
Use variables: $DATE, $TIME, $PAGE, $LINE, $USER...
Skip lines before heading: 0    Skip lines after heading: 1

Line   Page heading text    Align
1     Salary Report For Central Division  CENTER
2     $DATE  right
3
4
5

Skip lines before footing: 1    Skip lines after footing: 0

Line   Page footing text    Align
1     - $PAGE -  CENTER
2     Created by:$user  left
3
4
5
```

1=HELP  3=QUIT  4=MESSAGE  5=DISPLAY  6=MENU

Step 2 - Display your report

```
CA OLQ Release nn.n
->  *** Display Report ***
125000 Press the ENTER key to go to the next page of the report.

SALARY REPORT FOR CENTRAL DIVISION 07/29/99

<table>
<thead>
<tr>
<th>DEPARTMENT ID</th>
<th>NAME</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>FITZHUGH</td>
<td>$13,000.00</td>
</tr>
<tr>
<td></td>
<td>JOHNSON</td>
<td>$13,500.00</td>
</tr>
<tr>
<td></td>
<td>ORGRATZI</td>
<td>$39,000.00</td>
</tr>
<tr>
<td></td>
<td>PEOPLES</td>
<td>$80,000.00</td>
</tr>
<tr>
<td>2000</td>
<td>BLOOMER</td>
<td>$15,000.00</td>
</tr>
<tr>
<td></td>
<td>HUTTON</td>
<td>$44,000.00</td>
</tr>
<tr>
<td></td>
<td>JENSON</td>
<td>$82,000.00</td>
</tr>
<tr>
<td></td>
<td>KIMBALL</td>
<td>$45,000.00</td>
</tr>
<tr>
<td></td>
<td>KING</td>
<td>$14,500.00</td>
</tr>
<tr>
<td></td>
<td>NICEMAN</td>
<td>$14,000.00</td>
</tr>
<tr>
<td>3100</td>
<td>DOUGH</td>
<td>$33,000.00</td>
</tr>
</tbody>
</table>

CREATED BY:RMG
1=HELP  3=QUIT  4=MESSAGE  6=MENU  8=FWD  10=LEFT  11=RIGHT
```

How to Create Report Totals and Subtotals

This section shows you how to create report subtotals. The discussion is broken up into the following parts:
- **A list of key terms** used to describe report groups and subtotals
- **A quick example** of how to use CA OLQ to create subtotals
- **A step-by-step example** that walks you through the process of creating a report containing subtotals

For more information, see the following topics:
- Report Totals and Subtotals Key terms (see page 74)
- A Quick Example Creating a Subtotal (see page 75)
- A Step-By-Step Example Creating a Report With Subtotals (see page 82)
- Create Your Report (see page 83)
- Sorting Your Report Rows 1 (see page 85)
- Creating Groups and Specifying Subtotals (see page 85)
- Creating Report Totals (see page 88)
- Creating Levels of Nested Subtotals (see page 90)

---

**Report Totals and Subtotals Key terms**

Here are a few terms used to discuss report subtotals and totals:

**Aggregate function**

A function that performs a predefined operation on a group of report rows. You can apply aggregate functions to report groups. Examples of aggregate functions are: average, high value, low value, count, and total.

**Built-in function**

A function that performs a predefined string, arithmetic, date/time, or trigonometric calculation on your report rows. You can apply built-in functions to report groups by including them in a **COMPUTE** statement. Examples of built-in functions are: gregorian date, cosine, and square root.

**COMPUTE statement**

A CA OLQ syntax statement used to perform calculations in the menu facility. You can apply the **COMPUTE** statement to report groups. Any time you specify a built-in or aggregate function, CA OLQ creates a **COMPUTE** statement. You can also specify your own **COMPUTE** statements.

**Group by all**

A report total including all rows in your report. Group by all means the same thing as report total.

**Group field**

A report column whose value is used to divide your report rows into groups. For example, you could list all of the company's employees grouped according to which department they work in. In this case, DEPARTMENT-NAME is the group field.
Report group

A set of report rows where each row contains the same value of the group field. For example, the personnel department is a report group with DEPARTMENT NAME as the group field. Each row in this group contains PERSONNEL in the DEPARTMENT NAME field.

Report subtotal

A computation applied to a report group. For example, if you grouped your employees by department, you can create report subtotals that apply to that group, computing the average salary in each department.

Report total

A computation that includes all of the rows in your report. For example, you can compute the total sales revenue earned by all of your employees. Note that a report total does not have to be a sum. It can also be an average, a counter, a high value, or a low value.

A Quick Example Creating a Subtotal

Contents

- The Report Format - Sort screen (see page 77)
  - Step 1 - Sort your report rows (see page 77)
  - Step 2 - Create report groups (see page 79)
- The Report Format - Group By screen (see page 80)
  - Step 3 - Specify your group calculations (see page 80)

Creating a Report Subtotal

Step 1

Sort Report Rows

Step 2

Specify Report Groups

<table>
<thead>
<tr>
<th>Step 3</th>
<th>Aggregate functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report Format - Group By screen</td>
<td>Specify Group Calculations</td>
</tr>
<tr>
<td></td>
<td>- COUNT</td>
</tr>
<tr>
<td></td>
<td>- SUM</td>
</tr>
<tr>
<td></td>
<td>- AVERAGE</td>
</tr>
<tr>
<td></td>
<td>- MAXIMUM</td>
</tr>
</tbody>
</table>
Creating a Report Containing Subtotals:

The Boston Marathon

Suppose you were put in charge of keeping statistics for the Boston Marathon. The marathon database lists information on each of the 7,000 runners, including their name, country, time, sex, etc..

This is what your report looks like when you start out:

<table>
<thead>
<tr>
<th>LAST NAME</th>
<th>SEX</th>
<th>COUNTRY</th>
<th>FINISH TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>GUERRIERI</td>
<td>M</td>
<td>ITALY</td>
<td>2:19:30</td>
</tr>
<tr>
<td>PARSONS</td>
<td>M</td>
<td>USA</td>
<td>3:00:04</td>
</tr>
<tr>
<td>VANDER WYK</td>
<td>F</td>
<td>BELGIUM</td>
<td>2:54:12</td>
</tr>
<tr>
<td>BJORNSEN</td>
<td>F</td>
<td>SWEDEN</td>
<td>2:59:12</td>
</tr>
<tr>
<td>BARBADO</td>
<td>M</td>
<td>ITALY</td>
<td>2:40:12</td>
</tr>
<tr>
<td>KEEFE</td>
<td>M</td>
<td>USA</td>
<td>2:40:19</td>
</tr>
<tr>
<td>KLEIN</td>
<td>F</td>
<td>USA</td>
<td>3:09:23</td>
</tr>
<tr>
<td>YAMADA</td>
<td>M</td>
<td>JAPAN</td>
<td>2:30:41</td>
</tr>
<tr>
<td>MARTIN</td>
<td>M</td>
<td>FRANCE</td>
<td>3:10:29</td>
</tr>
<tr>
<td>FRECETTE</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:10:01</td>
</tr>
<tr>
<td>FUCCI</td>
<td>F</td>
<td>ITALY</td>
<td>3:40:37</td>
</tr>
<tr>
<td>BENOTTI</td>
<td>F</td>
<td>ITALY</td>
<td>3:12:42</td>
</tr>
</tbody>
</table>

What you're looking for

You want to create a subtotal that lists the time of the fastest female runner from each country. A portion of your report (one subtotal) would look something like this:

<table>
<thead>
<tr>
<th>LAST NAME</th>
<th>SEX</th>
<th>COUNTRY</th>
<th>FINISH TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>VANDER WYK</td>
<td>F</td>
<td>BELGIUM</td>
<td>2:54:12</td>
</tr>
<tr>
<td>FRECETTE</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:10:01</td>
</tr>
<tr>
<td>HELFGOTT</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:24:31</td>
</tr>
</tbody>
</table>

FASTEST IN BELGIUM: 2:54:12

What To Do

You need to perform three steps to create this subtotal:

- On the Report Format - Sort screen:
  - **Step 1** - Sort your report rows.

- On the Report Format - Group By screen:
  - **Step 3** - Specify your group calculations.
# The Report Format - Sort screen

## Step 1 - Sort your report rows

You specify how you want to sort your report rows on the Report Format - Sort screen. Your Report Format - Sort screen looks like this:

```
CA OLQ Release nn.n

->
```

Specify sort or group by request and press the ENTER key.

<table>
<thead>
<tr>
<th>OLQ-EXAMPLE</th>
<th>Disp</th>
<th>Sort Priority (A/D)</th>
<th>Order</th>
<th>Seq</th>
<th>Group By Level #</th>
</tr>
</thead>
<tbody>
<tr>
<td>X LAST-NAME</td>
<td>1</td>
<td>2</td>
<td>a</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>X COUNTRY</td>
<td>6</td>
<td>3</td>
<td>a</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>X FINISH-TIME</td>
<td>5</td>
<td>3</td>
<td>a</td>
<td>5</td>
<td>-</td>
</tr>
<tr>
<td>X SEX</td>
<td>4</td>
<td>1</td>
<td>a</td>
<td>4</td>
<td>-</td>
</tr>
</tbody>
</table>

Display lines: Detail X and/or Summary X  Group by all

Compute:

1=HELP  3=QUIT  4=MESSAGE  5=DISPLAY  6=MENU  11=HEADER

**Sort Priority** is used to specify the sort level. **Order (A/D)** is used to specify the sort order (ascending or descending).

### BOSTON MARATHON

<table>
<thead>
<tr>
<th>LAST NAME</th>
<th>SEX</th>
<th>COUNTRY</th>
<th>FINISH TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>VANDER WYK</td>
<td>F</td>
<td>BELGIUM</td>
<td>2:54:12</td>
</tr>
<tr>
<td>FRECETTE</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:10:01</td>
</tr>
<tr>
<td>HELFGOTT</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:24:31</td>
</tr>
</tbody>
</table>

**Sort Level 3**

Sort Field = FINISH-TIME  
Sort Priority = 3  
Sort Order = Ascending

<table>
<thead>
<tr>
<th>LAST NAME</th>
<th>SEX</th>
<th>COUNTRY</th>
<th>FINISH TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>DECRE</td>
<td>F</td>
<td>FRANCE</td>
<td>3:11:09</td>
</tr>
<tr>
<td>BENOTTI</td>
<td>F</td>
<td>ITALY</td>
<td>3:12:42</td>
</tr>
<tr>
<td>FUCI</td>
<td>F</td>
<td>ITALY</td>
<td>3:40:37</td>
</tr>
<tr>
<td>BJORNSEN</td>
<td>F</td>
<td>SWEDEN</td>
<td>2:59:12</td>
</tr>
<tr>
<td>KLEIN</td>
<td>F</td>
<td>USA</td>
<td>3:09:24</td>
</tr>
</tbody>
</table>

**Sort Level 2**

Sort Field = COUNTRY  
Sort Priority = 2  
Sort Order = Ascending

<table>
<thead>
<tr>
<th>LAST NAME</th>
<th>SEX</th>
<th>COUNTRY</th>
<th>FINISH TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>VANPRAAG</td>
<td>M</td>
<td>BELGIUM</td>
<td>2:45:60</td>
</tr>
<tr>
<td>MARTIN</td>
<td>M</td>
<td>FRANCE</td>
<td>3:10:29</td>
</tr>
<tr>
<td>GUERRIERI</td>
<td>M</td>
<td>ITALY</td>
<td>2:19:30</td>
</tr>
<tr>
<td>BARBADO</td>
<td>M</td>
<td>ITALY</td>
<td>2:40:12</td>
</tr>
<tr>
<td>YAMADA</td>
<td>M</td>
<td>JAPAN</td>
<td>2:30:41</td>
</tr>
<tr>
<td>KEEFE</td>
<td>M</td>
<td>USA</td>
<td>2:40:19</td>
</tr>
<tr>
<td>SIMON</td>
<td>M</td>
<td>USA</td>
<td>2:52:48</td>
</tr>
<tr>
<td>PARSONS</td>
<td>M</td>
<td>USA</td>
<td>3:00:04</td>
</tr>
</tbody>
</table>
Sort Level 1
Sort Field = SEX
Sort Priority = 1
Sort Order = Ascending

Sort Levels and Sort Order:
In this report, you want to sort your report rows by sex (to group all of the women together), by country, and within each country by race time. You must sort before you can specify any report groups.

If You Do Sort First
Your report accurately reflects the fastest time for each country.

<table>
<thead>
<tr>
<th>LAST NAME</th>
<th>SEX</th>
<th>COUNTRY</th>
<th>FINISH TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>VANDER WYK</td>
<td>F</td>
<td>BELGIUM</td>
<td>2:54:12</td>
</tr>
<tr>
<td>FRECETTE</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:10:01</td>
</tr>
<tr>
<td>HELFGOTT</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:24:31</td>
</tr>
<tr>
<td>DECRE</td>
<td>F</td>
<td>FRANCE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>FASTEST FROM BELGIUM :2:54:12</td>
<td>3:11:09</td>
</tr>
<tr>
<td>BENOTTI</td>
<td>F</td>
<td>ITALY</td>
<td>3:12:42</td>
</tr>
<tr>
<td>Fucci</td>
<td>F</td>
<td>ITALY</td>
<td>3:40:37</td>
</tr>
<tr>
<td>BJORNSEN</td>
<td>F</td>
<td>SWEDEN</td>
<td>2:59:12</td>
</tr>
<tr>
<td>KLEIN</td>
<td>F</td>
<td>USA</td>
<td>3:09:24</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FASTEST FROM SWEDEN :2:59:12</td>
<td>3:09:24</td>
</tr>
<tr>
<td>FRECETTE</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:10:01</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FASTEST FROM BELGIUM :2:54:12</td>
<td>3:11:09</td>
</tr>
</tbody>
</table>

If You Don't Sort First
If you sort by SEX, but forget to sort by COUNTRY, CA OLQ creates a group whenever it encounters a change in the COUNTRY column. This causes inaccurate results when you try to calculate the fastest time for each country.

<table>
<thead>
<tr>
<th>LAST NAME</th>
<th>SEX</th>
<th>COUNTRY</th>
<th>FINISH TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>BENOTTI</td>
<td>F</td>
<td>ITALY</td>
<td>3:12:42</td>
</tr>
<tr>
<td>Fucci</td>
<td>F</td>
<td>ITALY</td>
<td>3:40:37</td>
</tr>
<tr>
<td>VANDER WYK</td>
<td>F</td>
<td>BELGIUM</td>
<td>2:54:12</td>
</tr>
<tr>
<td>BJORNSEN</td>
<td>F</td>
<td>SWEDEN</td>
<td>2:59:12</td>
</tr>
<tr>
<td>FRECETTE</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:10:01</td>
</tr>
<tr>
<td>KLEIN</td>
<td>F</td>
<td>USA</td>
<td>3:09:24</td>
</tr>
<tr>
<td>DECRE</td>
<td>F</td>
<td>FRANCE</td>
<td>3:11:09</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FASTEST FROM FRANCE :3:11:09</td>
<td>3:11:09</td>
</tr>
</tbody>
</table>
Why You Have to Sort First: CA OLQ groups data by looking for a change in the value of the group field (COUNTRY). If the data is not sorted in order by the group field, the changes can occur at random and produce subtotals other than the ones you intended.

Step 2 - Create report groups

You also specify your group levels on the Report Format - Sort screen. Your Report Format - Sort screen looks like this:

```
CA OLQ Example
->
133000 Specify sort or group by request and press the ENTER key
Disp Seq Sort Priority Disp Seq Sort Priority
Group By Level #
CA OLQ-EXAMPLE
X LAST-NAME 1
X COUNTRY 6 2 a 2
X FINISH-TIME 5 3 a
X SEX 4 1 a 1

Display lines: Detail X and/or Summary X  Group by all _
Compute: 1=HELP 3=QUIT 4=MESSAGE 5=DISPLAY 6=MENU 11=HEADER
```

Group By Level # is used to specify group levels.

BOSTON MARATHON

mm/dd/yy

<table>
<thead>
<tr>
<th>LAST NAME</th>
<th>SEX</th>
<th>COUNTRY</th>
<th>FINISH TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP LEVEL 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GROUP FIELD = COUNTRY HELFGOTT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VANDER WYK</td>
<td>F</td>
<td>BELGIUM</td>
<td>2:54:12</td>
</tr>
<tr>
<td>FRECETTE</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:10:01</td>
</tr>
<tr>
<td>DECRE</td>
<td>F</td>
<td>FRANCE</td>
<td>3:11:09</td>
</tr>
<tr>
<td>BENOTTI</td>
<td>F</td>
<td>ITALY</td>
<td>3:12:42</td>
</tr>
<tr>
<td>FUCCI</td>
<td>F</td>
<td>ITALY</td>
<td>3:40:37</td>
</tr>
<tr>
<td>BJORNSEN</td>
<td>F</td>
<td>SWEDEN</td>
<td>2:59:12</td>
</tr>
<tr>
<td>GROUP LEVEL 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GROUP FIELD = SEX</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KLEIN</td>
<td>F</td>
<td>USA</td>
<td>3:09:24</td>
</tr>
<tr>
<td>VANPRAAG</td>
<td>M</td>
<td>BELGIUM</td>
<td>2:45:60</td>
</tr>
<tr>
<td>MARTIN</td>
<td>M</td>
<td>FRANCE</td>
<td>3:10:29</td>
</tr>
<tr>
<td>GUERRIERI</td>
<td>M</td>
<td>ITALY</td>
<td>2:19:30</td>
</tr>
<tr>
<td>BARBADO</td>
<td>M</td>
<td>ITALY</td>
<td>2:40:12</td>
</tr>
<tr>
<td>YAMADA</td>
<td>M</td>
<td>JAPAN</td>
<td>2:30:41</td>
</tr>
<tr>
<td>KEEFE</td>
<td>M</td>
<td>USA</td>
<td>2:40:19</td>
</tr>
<tr>
<td>SIMON</td>
<td>M</td>
<td>USA</td>
<td>2:52:48</td>
</tr>
<tr>
<td>PARSONS</td>
<td>M</td>
<td>USA</td>
<td>3:00:04</td>
</tr>
</tbody>
</table>
**Group Levels**: This report is grouped at two levels: by SEX (level 1) and by COUNTRY (level 2).

### The Report Format - Group By screen

You specify your group calculations on the Report Format - Group By screen. The Report Format - Group By screen looks like this:

```
CA OLQ Release nn.n   *** Report Format - Group By ***
                 ->             Page   1  OF  1
OLQ-EXAMPLE.COUNTRY
136000 Specify summary computations and press the ENTER key
```

**Group by** displays the current group field. **Level #** displays the current group level. **Sum, Avg, Max, Min, and Count** are mathematical functions that you can apply to any report column. These functions apply to the extent of the current group level.

### Step 3 - Specify your group calculations

To find the fastest time for women from each country, you need to compute the lowest race score in each country's group. You specify this calculation by selecting the **Min** (minimum) function in the row across from FINISH TIME. CA OLQ automatically calculates which woman in each country has the fastest time, and lists each group's top time.

**BOSTON MARATHON**

<table>
<thead>
<tr>
<th>LAST NAME</th>
<th>SEX</th>
<th>COUNTRY</th>
<th>FINISH TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>VANDER WYK</td>
<td>F</td>
<td>BELGIUM</td>
<td>2:54:12</td>
</tr>
<tr>
<td>FRECETTE</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:10:01</td>
</tr>
<tr>
<td>HELFGOTT</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:24:31</td>
</tr>
</tbody>
</table>

**Group Calculation**

(subgroup)

Group Level = 2 FASTEST FROM BELGIUM  :2:54:12
Current Group Field = COUNTRY
Aggregate Function = MIN

<table>
<thead>
<tr>
<th>NCE</th>
<th>:3:11:09</th>
</tr>
</thead>
<tbody>
<tr>
<td>BENOTTI</td>
<td>F</td>
</tr>
<tr>
<td>FUCCI</td>
<td>F</td>
</tr>
</tbody>
</table>

FASTEST FROM ITA
Group Calculations: This report uses the MIN (minimum) aggregate function to calculate the lowest time in each country’s group.

When you’re done, your report looks like this:

BOSTON MARATHON
05/19/99

<table>
<thead>
<tr>
<th>LAST NAME</th>
<th>SEX</th>
<th>COUNTRY</th>
<th>FINISH TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>VANDER WYK</td>
<td>F</td>
<td>BELGIUM</td>
<td>2:54:12</td>
</tr>
<tr>
<td>FRECETTE</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:10:01</td>
</tr>
<tr>
<td>HELFGOTT</td>
<td>F</td>
<td>BELGIUM</td>
<td>3:24:31</td>
</tr>
</tbody>
</table>

FASTEST IN BELGIUM :2:54:12

| DECRE        | F   | FRANCE      | 3:11:09     |

FASTEST IN FRANCE :3:11:09

| BENOTTI      | F   | ITALY       | 3:12:42     |
| FUCCHI       | F   | ITALY       | 3:40:37     |

FASTEST IN ITALY :3:12:42

| BJORNSEN     | F   | SWEDEN      | 2:59:12     |

FASTEST IN SWEDEN :2:59:12

| KLEIN        | F   | USA         | 3:09:23     |

FASTEST IN USA :3:09:23

Aggregate functions

CA OLQ provides five predefined calculations that you can apply to your report groups. These calculations are called aggregate functions. They are:

- **Count** counts the number of rows in the report group.
- **Total** adds up occurrences of numeric columns in the report group.
- **Average** finds the average of the specified column in the report group.
- **Maximum** finds the highest value of the specified column in the report group.
- **Minimum** finds the lowest value of the specified column in the report group.

How to use aggregate functions

Aggregate functions are listed on the Report Format - Group By screen. To use them:
1. Make sure that the current group field is the one you want. The current group field is listed in the **Group by** field.

2. Select an aggregate function corresponding to the appropriate report column.

---

**A Step-By-Step Example Creating a Report With Subtotals**

In the rest of this section, you will create a series of reports that contain subtotals. Each report uses an additional CA OLQ grouping feature to enhance the original report.

You will first:

1. **Create the original report**, using the EMPLOYEE data table.
   Your enhancements will then:

2. **Sort** the rows of the report.

3. Specify **report groups**, arranging the employees according to which department they work in.

4. Compute a **report subtotal**, finding the average salary of each department.

5. Compute a **report total**, finding the total amount of money paid in employee salaries.

6. Create **nested subtotals**, listing the number of employees working on each project within each department.

This is how your final report looks after you have added all of the grouping enhancements:

```
CA OLQ Release nn.n
-> 125004 Press the ENTER key for DISPLAY/FORMAT ACTIVITY selections

*** Display Report ***

EMPLOYEE REPORT

<table>
<thead>
<tr>
<th>PROJECT</th>
<th>SALARY-AMOUNT</th>
<th>EMP-LAST-NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESEARCH</td>
<td>45000.00</td>
<td>FINN</td>
</tr>
<tr>
<td></td>
<td>80000.00</td>
<td>WILCO</td>
</tr>
<tr>
<td>-----------------</td>
<td>---------------</td>
<td>--------------</td>
</tr>
<tr>
<td>COUNT FOR RESEARCH:</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td>---------------</td>
<td>--------------</td>
</tr>
<tr>
<td>AVE FOR 5200 : 45400.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>453000.00</td>
<td></td>
</tr>
</tbody>
</table>

END OF REPORT
```

---

1=HELP  3=QUIT  4=MESSAGE  6=MENU  7=BWD  10=LEFT  11=RIGHT
Report rows are sorted by DEPT-ID, PROJECT, and SALARY-AMOUNT. Within each department, employees are grouped by which project they work on. The average salary for each department is listed. The total amount paid in employee salaries is listed at the bottom of the report.

Create Your Report

Contents

- Step 1 - Set your session options (see page 83)
- Step 2 - Select your table (see page 83)
- Step 3 - Select columns (see page 84)
- Step 4 - Retrieve your data (see page 84)
- Step 5 - Display your report (see page 85)

Step 1 - Set your session options

Specify N. In this example, you will use ASF tables. Change the Full/Sparse option to Sparse. This option makes reports containing groups easier to read.

Step 2 - Select your table

Specify SELECT. Select the EMPLOYEE data table.

---

10-Jan-2018 83/222
Function: s Select
          _ Create
          _ Delete
          _ Add
          _ Replace
          _

Enter table:
-or-
Select table
- ACCOUNTING
- BUDGET
- DEPARTMENT
- EMP-TABLE
s EMPLOYEE
- EMPLOYEE TABLE
- EMPLOYEE-DATA
- JOB SALARIES
- MELROSE EMPLOYEES
- OLQ EXAMPLE
- PERSONNEL
1=HELP  3=QUIT  4=MESSAGE  6=MENU  8=FWD

Step 3 - Select columns

Select EMP-LAST-NAME, DEPT-ID, SALARY-AMOUNT, and PROJECT.

124000 Select columns, specify selection criteria and press the ENTER key

Columns Currently Selected: 0 Selection Criteria Distinct N Y/N
- EMPLOYEE
  03 EMP-ID
s 03 EMP-LAST-NAME
  03 EMP-FIRST-NAME
  03 START-YEAR
  03 DEPT-HEAD-ID
s 03 DEPT-ID
s 03 SALARY-AMOUNT
s 03 PROJECT
  03 OFFICE-CODE

Additional selection criteria:

1=HELP  3=QUIT  4=MESSAGE  6=MENU  PA2=REFRESH

Step 4 - Retrieve your data

Depending on how high the interrupt count at your site is set, CA OLQ may bypass the Retrieval Interrupted screen and proceed directly to the Retrieval Completed screen.

130000 Select activity and press the ENTER key

Number of whole rows . . . . . . . 11 Total displayable cols . 20
Total number of records read . . . 11 Formatted line length . 372
Total number of records selected . 11
Number of data errors . . . . . . . 0
Select Command/
Step 5 - Display your report

At this point, the data in your report is listed in the order in which it has been retrieved from the database.

CA OLQ Release nn.n

EMPLOYEE REPORT

mm/dd/yy

EMP-LAST-NAME DEPT-ID SALARY-AMOUNT PROJECT

BANK 4000 80000.00 TESTING
ANGELO 4000 18000.00 PLANNING
MCDUGALL 4000 18000.00 PLANNING
PENMAN 4000 39000.00 PLANNING
JACKSON 4000 34000.00 PLANNING
FINN 5200 45000.00 RESEARCH
WILCO 5200 80000.00 RESEARCH
TIME 5200 33000.00 RESEARCH
KASPAR 5200 31000.00 EVALUATE
ZEDI 4000 37000.00 EVALUATE
CLOTH 5200 38000.00 EVALUATE

END OF REPORT

1=HELP 3=QUIT 4=MESSAGE 6=MENU

Sorting Your Report Rows 1

In this step, you sort the rows of the report on three different levels:

- By DEPT-ID
- Within each department group, by PROJECT
- Within each project group, by EMP-LAST-NAME

Creating Groups and Specifying Subtotals
CA IDMS - 19.0

- Step 1 - Specify your group level (see page 86)
- Step 2 - Specify your subtotal (see page 86)
- Step 3 - Display your report (see page 87)

In this step, you find the average salary of each department by specifying DEPT-ID as a group field and applying a calculation to that group. To do this:

1. Sort the rows of your report by department. You just did this in the last step of this example.
2. Group the employees according to which department they work in.
3. Calculate the average salary amount for that group.

Step 1 - Specify your group level

In this step, you calculate the average salary in each department.

Start on the Report Format - Sort screen. To get there, type SORT in the command line.

Specify that you want to group your employees by department by entering a 1 next to Group By Level#. Specifying 1 means that DEPT-ID is the highest level group field. You may later create subgroups within each department.

```
CA OLQ Release nn.n
*** Report Format - Sort ***
Disp Seq Sort Priority Order (A/D) Level #
EMPLOYEE EMP-LAST-NAME 4
X  DEPT-ID 1 1 A 1
X  SALARY-AMOUNT 3 3 A
X  PROJECT 2 2 A
```

Display lines: Detail X and/or Summary X Group by all

Compute:
1=HELP 3=QUIT 4=MESSAGE 5=DISPLAY 6=MENU 11=HEADER

Step 2 - Specify your subtotal

Group by displays EMPLOYEE.DEPT-ID as the current group field. Level# displays 1 as the current group level. Use Avg to specify that you want CA OLQ to compute the average salary for each department.

```
CA OLQ Release nn.n
*** Report Format - Group By ***
Disp Seq Order (A/D) Group By
EMPLOYEE EMP-LAST-NAME 4
```

Specify summary computations and press the ENTER key.
### CA OLQ Report

**Step 3 - Display your report**

CA OLQ displays the average salary for each department.

```
---
```

#### Display Report

- **Report Format**: 
  - **Sort**: Group by Level# 1
  - **Disp**, **Sort**, **Order (A/D)**, **Group By**

#### Fields
- **EMPLOYEE.DEPT-ID**
- **EMPLOYEE.EMP-LAST-NAME**
- **EMPLOYEE.DEPT-ID**
- **EMPLOYEE.SALARY-AMOUNT**
- **EMPLOYEE.PROJECT**

#### Compute Statement

CA OLQ generates this COMPUTE statement to calculate the average salary for each department.

```
COMPUTE FIELDS:
  X 'EMPLOYEE.SALARY-AMOUNT-AVE-2'=AVE(EMPLOYEE.SALARY-AMOUNT) GROUP BY EMPLOYEE.DEPT-ID LEVEL 1
```

#### Display

- **Detail X** and/or **Summary X**
- **Group by all**

#### Compute

- **1=HELP**
- **3=QUIT**
- **4=MESSAGE**
- **5=DISPLAY**
- **6=MENU**
- **11=HEADER**

### EMPLOYEE REPORT

```
DEPT-ID    PROJECT        SALARY-AMOUNT   EMP-LAST-NAME
--------    --------        -----------    ------------
4000       EVALUATE       37000.00      ZEDI
           PLANNING       18000.00      MCDUGALL
           34000.00      ANGELO
           39000.00      JACKSON
```

---

10-Jan-2018
Creating Report Totals

Contents
- Step 1 - Specify your group level (see page 88)
- Step 2 - Specify your report total (see page 89)
- Step 3 - Display your report (see page 89)

In this step, you find the total amount of money paid in employee salaries. Because this sum includes all of the rows in the report, you use the Group by all field.

Step 1 - Specify your group level

Start on the Report Format - Sort screen. To get there, type SORT in the command line.

Select Group by all to include all of the rows in your report in a group computation.

Display lines: Detail X and/or Summary X Group by all x

Compute:

1=HELP 3=QUIT 4=MESSAGE 5=DISPLAY 6=MENU 11=HEADER
Step 2 - Specify your report total

**Group by** shows you that you are including all of the rows in the report in your computation. Specify that you want to know the sum of all the employee salaries in the report. Use **Seq** to specify the column under which you want the computed field displayed.

```
CA OLQ Release nn.n
*** Report Format - Group By ***
->
136000 Specify summary computations and press the ENTER key

Group by: ALL

<table>
<thead>
<tr>
<th>Seq</th>
<th>Sum</th>
<th>Avg</th>
<th>Max</th>
<th>Min</th>
<th>Count</th>
<th>Level#</th>
</tr>
</thead>
</table>
| EMPLOYEE
| X EMP-LAST-NAME | 4 |   |   |   |       |       |
| X DEPT-ID | 1 |   |   |   |       |       |
| X SALARY-AMOUNT | 3 | x |   |   |       |       |
| X PROJECT | 2 |   |   |   |       |       |

COMPUTE FIELDS:
X ‘EMPLOYEE.SALARY-AMOUNT-AVE-2’=AVE(EMPLOYEE.SALARY-AMOUNT) GROUP BY EMPLOYEE.DEPT-ID LEVEL 1
```

Skip lines after group 1
Separator character =

Compute:

```
1=HELP 3=QUIT 4=MESSAGE 5=DISPLAY 6=MENU 10=PICTURE
```

CA OLQ generates this COMPUTE statement to calculate the report total.

```
CA OLQ Release nn.n
*** Report Format - Sort ***
->
133000 Specify sort or group by request and press the ENTER key

Disp      Sort    Order    Group By
Seq      Priority (A/D)    Level #
EMPLOYEE
X EMP-LAST-NAME | 4 |   |   | 1
X DEPT-ID | 1 | 1 | A | 1
X SALARY-AMOUNT | 3 | 3 | A | 1
X PROJECT | 2 | 2 | A | 1

COMPUTE FIELDS:
X ‘EMPLOYEE.SALARY-AMOUNT-AVE-2’=AVE(EMPLOYEE.SALARY-AMOUNT) GROUP BY EMPLOYEE.DEPT-ID LEVEL 1
X TOTAL-3-ALL=TOTAL(EMPLOYEE.SALARY-AMOUNT) GROUP BY ALL LEVEL 1
```

Display lines: Detail X and/or Summary X Group by all X

Compute:

```
1=HELP 3=QUIT 4=MESSAGE 5=DISPLAY 6=MENU 11=HEADER
```

Step 3 - Display your report

CA OLQ displays the total amount spent in employee salaries.
Creating Levels of Nested Subtotals

Contents
- Step 1 - Specify your group level (see page 90)
- Step 2 - Specify your subtotal (see page 91)
- Step 3 - Display your report (see page 92)

In this step, you list the number of employees in each project group. For example, you can find out how many people are currently on your department's planning committee.

Because each department has a planning committee, this kind of calculation requires two group levels:

- **Group level 1** groups the employees according to which department they work in. DEPT-ID is the group field.

- **Group level 2** groups the employees within each department according to which project they are currently assigned to. PROJECT is the group field.

**Step 1 - Specify your group level**

Start on the Report Format - Sort screen. To get there, type `Sort` in the command line.

Specify that you want to group by PROJECT within each department group.

---

EMPLOYEE

---

1=HELP 3=QUIT 4=MESSAGE 6=MENU 10=LEFT 11=RIGHT
Display lines: Detail X and/or Summary X   Group by all X

Compute:

1=HELP   3=QUIT   4=MESSAGE   5=DISPLAY   6=MENU   11=HEADER

Step 2 - Specify your subtotal

**Group by** displays EMPLOYEE.PROJECT as the current group level. **Level#** displays 2 as the current group level. Specify that you want to count the number of employees within each project.

CA OLQ Release nn.n  *** Report Format - Group By ***
->  Page 1 OF 1
136000 Specify summary computations and press the ENTER key

Group by: EMPLOYEE.PROJECT

<table>
<thead>
<tr>
<th>Seq</th>
<th>Sum</th>
<th>Avg</th>
<th>Max</th>
<th>Min</th>
<th>Count</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP-LAST-NAME</td>
<td>4</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
<tr>
<td>DEPT-ID</td>
<td>1</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
<tr>
<td>SALARY-AMOUNT</td>
<td>3</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
<tr>
<td>PROJECT</td>
<td>2</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
</tbody>
</table>

**COMPUTE FIELDS:**

```
X 'EMPLOYEE.PROJECT-COUNT-2'=COUNT GROUP BY EMPLOYEE.PROJECT LEVEL 2
```

Skip lines after group 1   Separator character -

CA OLQ generates this COMPUTE statement to count the total number of employees working on each project.
Step 3 - Display your report

CA OLQ lists the number of employees assigned to each project.

CA OLQ Release nn.n

*** Display Report ***

125004 Press the ENTER key for DISPLAY/FORMAT ACTIVITY selections

EMPLOYEE REPORT

PROJECT | SALARY-AMOUNT | EMP-LAST-NAME
---------|---------------|---------------
RESEARCH | 45000.00      | FINN
80000.00 |               | WILCO

-----------
COUNT FOR RESEARCH: 3

-----------
AVE FOR 5200 : 45400.00

-----------
453000.00

END OF REPORT

- 3 -

1=HELP  3=QUIT  4=MESSAGE  6=MENU  7=BWD  10=LEFT  11=RIGHT

How To Format Reports Containing Calculations

In this section This section shows you how to enhance the appearance of groups and group calculations in your CA OLQ report. Using CA OLQ, you can:

- Specify how many lines you want to skip after a group
- Specify which character you want to use to set apart computations
- Tailor your computation headings
- Display only lines containing computations
- Display all lines except those containing computations
- Skip pages after a group
The report shown below illustrates some of the ways you can format your computations using CA OLQ.

```
CA OLQ Release nn.n  *** Display Report ***
->  Page  2  Line  13
125004  Press the ENTER key for DISPLAY/FORMAT ACTIVITY selections

EMPLOYEE REPORT

mm/dd/yy

EMP-LAST-NAME           SALARY-AMOUNT
CLOTH                   38000.00
KASPAR                  31000.00

FINN                    45000.00
TIME                    33000.00
WILCO                   80000.00

AVERAGE SALARY: 45400.00

TOTAL SPENT IN SALARIES: 453000.00
```

Key Terms

Here are a few terms used to discuss formatting report groups and calculations:

**Detail line**

A report line that displays a row of data retrieved from the database. A detail line can contain a computed column, but does not contain a group computation.

**Group computation**

A calculation that CA OLQ performs on a report group.

**Separator character**

A character used to separate group computations from the rest of the report.

**Summary computation**

Another term for group computation.

**Summary line**

A report line that displays a group computation.

---

Create Your Reports

In this sample, you format the report you created in here (see page 73). To build your current report, follow the steps outlined in Section 6.
EMPLOYEE REPORT

DEPT-ID | PROJECT | EMP-LAST-NAME
--- | --- | ---
4000 | EVALUATE | ZEDI

COUNT FOR EVALUATE: 1

PLANNING

COUNT FOR PLANNING: 4

TESTING

1=HELP  3=QUIT  4=MESSAGE  6=MENU  8=FWD  10=LEFT  11=RIGHT

Skipping Lines After Groups

In this step, you specify how many blank lines you want CA OLQ to display between report groups.

You can indicate a different spacing for each group level. If your report contains more than one group level, you must establish the current group field and group level before you can indicate how many lines you want to skip.

<table>
<thead>
<tr>
<th>Step</th>
<th>What To Do</th>
<th>How To Do It</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Proceed to the Report Format - Group By screen.</td>
<td>Type GROUP BY in the command line of any screen.</td>
</tr>
<tr>
<td>2</td>
<td>Establish the current group field and level.</td>
<td>Type the group field next to Group by: Type the corresponding level number in the Level# field. Press [Enter].</td>
</tr>
<tr>
<td>3</td>
<td>Specify how many lines you want displayed after the current group.</td>
<td>Enter the number next to Skip lines after group.</td>
</tr>
</tbody>
</table>

If you’re not sure how to specify your group field and level, they are listed in the COMPUTE statement on the Report Format - Group By screen. For example:

**If your COMPUTE statement reads:**

```
'EMPLOYEE.PROJECT-COUNT'=COUNT
GROUP BY EMPLOYEE.PROJECT LEVEL 2'
```

Your **group field is:**

EMPLOYEE.PROJECT

Your **group level is:**

2

Specify employee.project as the current group field. Specify 2 as the current group level.
CA OLQ Release nn.n                          *** Report Format - Group By ***
->                                          Page 1 OF 2
136000 Specify summary computations and press the ENTER key

Group by: **employee.project**

<table>
<thead>
<tr>
<th>Seq</th>
<th>Sum</th>
<th>Avg</th>
<th>Max</th>
<th>Min</th>
<th>Count</th>
<th>_</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X EMP-LAST-NAME</td>
<td>3</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
<tr>
<td>X DEPT-ID</td>
<td>1</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
<tr>
<td>X SALARY-AMOUNT</td>
<td>4</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
<tr>
<td>X PROJECT</td>
<td>2</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
</tbody>
</table>

**COMPUTE FIELDS:**

X TOTAL-3-ALL=TOTAL(EMPLOYEE.SALARY-AMOUNT) GROUP BY ALL LEVEL 1

X 'EMPLOYEE.SALARY-AMOUNT-AVE-2'=AVE(EMPLOYEE.SALARY-AMOUNT) GROUP BY EMPLOYEE.DEPT-ID LEVEL 1

Skip lines after group 1  Separator character -

Compute:

1=HELP  3=QUIT  4=MESSAGE  5=DISPLAY  6=MENU  8=FWD  10=PICTURE

CA OLQ echoes the screen, listing the current group field and level. Specify that you want to skip 0 lines after each project group.

CA OLQ Release nn.n                          *** Display Report ***
->                                          Page 1 Line 1
104009 DISPLAY RIGHT to see more report columns
125000 Press the ENTER key to go to the next page of the report.

DEPT-ID       PROJECT       EMP-LAST-NAME
4000          EVALUATE      ZEDI

**COUNT FOR EVALUATE:** 1

PLANNING       ANGELO
JACKSON        JACKSON
MCDougall
Specifying a SCparator character

In this step, you specify which character you want to use to set your group calculations apart from the rest of your report.

CA OLQ provides lines separating your group calculations from the detail lines in your report. The default separator characters that make up these lines are:

- A hyphen (-) for subtotals
- An equal sign (=) for report totals

You can change these default separator characters to modify your report’s appearance.

Each group level's separator character is modified individually. Before you can specify a separation character, you must establish a current group field and level.

<table>
<thead>
<tr>
<th>Step</th>
<th>What To Do</th>
<th>How To Do It</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Go to the Report Format - Group By screen.</td>
<td>Type group by in the command line of any screen.</td>
</tr>
<tr>
<td>2</td>
<td>Establish the current group field and level.</td>
<td>Type the group field in the Group by entry. Type the corresponding level number in the Level # field. Press [Enter].</td>
</tr>
<tr>
<td>3</td>
<td>Specify your separator character.</td>
<td>Type the character you want in the Separator character field.</td>
</tr>
</tbody>
</table>

Specify all as your current group field. Specify 1 as the current group level.

```
136000 Specify summary computations and press the ENTER key
```

```
Group by: all

EMPLOYEE
X EMP-LAST-NAME 3
X DEPT-ID 1
X SALARY-AMOUNT 4
X PROJECT 2

COMPUTE FIELDS:
X TOTAL-3-ALL=TOTAL(EMPLOYEE.SALARY-AMOUNT) GROUP BY ALL LEVEL 1
X 'EMPLOYEE.SALARY-AMOUNT-AVE-2'=AVE(EMPLOYEE.SALARY-AMOUNT) GROUP BY
```
Compute:

1=HELP    3=QUIT    4=MESSAGE    5=DISPLAY    6=MENU    8=FWD    10=PICTURE

CA OLQ echoes the screen, listing the current group field and level. Specify that you want the underscore (_) to set apart the report total.

CA OLQ Release nn.n

*** Report Format - Group By ***

136000 Specify summary computations and press the ENTER key

Group by: **ALL**

<table>
<thead>
<tr>
<th>Seq</th>
<th>Sum</th>
<th>Avg</th>
<th>Max</th>
<th>Min</th>
<th>Count</th>
<th>Level#1</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE</td>
<td>X EMP-LAST-NAME</td>
<td>3</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
</tr>
<tr>
<td>X DEPT-ID</td>
<td>1</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
<tr>
<td>X SALARY-AMOUNT</td>
<td>4</td>
<td>X</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
<tr>
<td>X PROJECT</td>
<td>2</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td></td>
</tr>
</tbody>
</table>

COMPUTE FIELDS:

X 'TOTAL-3-ALL=TOTAL(EMPLOYEE.SALARY-AMOUNT) GROUP BY ALL LEVEL 1'

X 'EMPLOYEE.SALARY-AMOUNT-AVE-2'=AVE(EMPLOYEE.SALARY-AMOUNT) GROUP BY EMPLOYEE.DEPT-ID LEVEL 1

You may have to scroll forward (PF8) to view this computation. Underscores separate the report total computation from the preceding detail lines.

CA OLQ Release nn.n

*** Display Report ***

125004 Press the ENTER key for DISPLAY/FORMAT ACTIVITY selections

 EMPLOYEE REPORT

<table>
<thead>
<tr>
<th>PROJECT</th>
<th>EMP-LAST-NAME</th>
<th>SALARY-AMOUNT</th>
</tr>
</thead>
<tbody>
<tr>
<td>EVALUATE</td>
<td>CLOTH</td>
<td>38000.00</td>
</tr>
<tr>
<td>KASPAR</td>
<td>31000.00</td>
<td></td>
</tr>
</tbody>
</table>

COUNT FOR EVALUATE: 2

| RESEARCH | 2 | FINN | 45000.00 |
| TIME | 33000.00 |
| WILCO | 80000.00 |

COUNT FOR RESEARCH: 3

AVE FOR 5200 : 45400.00

453000.00

1=HELP    3=QUIT    4=MESSAGE    6=MENU    7=BWD    10=LEFT    11=RIGHT
Giving Your Computation a Heading

In this step, you modify the default headers for your group computations.

Computations with ALL as the group field are not assigned a default header. All other group computations are assigned default headers. These headers include:

- A literal that labels the report group, consisting of a dollar sign ($) plus the group field name
- Any built-in or aggregate functions included in that group computation

For example, the default header for the average salary for each department is:

'AVERAGE FOR $EMPLOYEE.DEPT-ID'

This header would look like this for department 5200:

'AVERAGE FOR 5200 :'

The default headers are listed on the Report Format - Header screen. To get there, type `header` in the command line of any screen.

```
CA OLQ Release nn.n   *** Report Format - Header ***
->                   Page 1 of  2
134000 Specify column headers and press the ENTER key
Underline character: - Disp Seq   Header
   EMPLOYEE
   X EMP-LAST-NAME   3 EMP-LAST-NAME
   X DEPT-ID        1 DEPT-ID
   X SALARY-AMOUNT  4 SALARY-AMOUNT
   X PROJECT        2 PROJECT

COMPUTE FIELDS:
   X TOTAL-3-ALL=TOTAL(EMPLOYEE.SALARY-AMOUNT) GROUP BY ALL LEVEL 1
   X 'EMPLOYEE.SALARY-AMOUNT-AVE-2'=AVE(EMPLOYEE.SALARY-AMOUNT) GROUP BY EMPLOYEE.DEPT-ID LEVEL 1

Compute:
1=HELP 3=QUIT 4=MESSAGE 5=DISPLAY 6=MENU 8=FWD 10=SORT 11=EDIT
CA OLQ Release nn.n   *** Report Format - Header ***
->                   Page 2 of  2
134000 Specify column headers and press the ENTER key
Underline character: - Disp Seq   Header
   EMPLOYEE
   X EMP-LAST-NAME   3 EMP-LAST-NAME
   X DEPT-ID        1 DEPT-ID
   X PROJECT        2 PROJECT

COMPUTE FIELDS:
   X 'EMPLOYEE.PROJECT-COUNT'=COUNT(EMPLOYEE.PROJECT) GROUP BY EMPLOYEE.PROJECT LEVEL 2
```
Compute:
1=HELP  3=QUIT  4=MESSAGE  5=DISPLAY  6=MENU  7=BWD  10=SORT  11=EDIT

Specify more legible labels for your report total and department computations.

CA OLQ Release nn.n
*** Report Format - Header ***
Page 1 of 2
34000 Specify column headers and press the ENTER key
Underline character: -
Seq          Header
EMPLOYEE
X EMP-LAST-NAME  3 EMP-LAST-NAME
X DEPT-ID        1 DEPT-ID
X SALARY-AMOUNT  4 SALARY-AMOUNT
X PROJECT        2 PROJECT

COMPUTE FIELDS:
X TOTAL-3-ALL=TOTAL(EMPLOYEE.SALARY-AMOUNT) GROUP BY ALL LEVEL 1
X 'EMPLOYEE.SALARY-AMOUNT-AVE-2'=AVE(EMPLOYEE.SALARY-AMOUNT) GROUP BY EMPLOYEE.DEPT-ID LEVEL 1

Compute:
1=HELP  3=QUIT  4=MESSAGE  5=DISPLAY  6=MENU  8=FWD  10=SORT  11=EDIT

Specify a more legible label for your project computation. In this example, the label contains a variable field ($EMPLOYEE.PROJECT) for the project name.

CA OLQ Release nn.n
*** Report Format - Header ***
Page 2 of 2
34000 Specify column headers and press the ENTER key
Underline character: -
Seq          Header
COMPUTE FIELDS:
X 'EMPLOYEE.PROJECT-COUNT'=COUNT
GROUP BY EMPLOYEE.PROJECT LEVEL 2

'WORKING ON $EMPLOYEE.PROJECT:'

Compute:
1=HELP  3=QUIT  4=MESSAGE  5=DISPLAY  6=MENU  7=BWD  10=SORT  11=EDIT

CA OLQ displays a new label for your project groups.

CA OLQ Release nn.n
*** Display Report ***
Page 1 Line 1
34000 Press the ENTER key to go to the next page of the report.
Displaying Only Summary Computations

In this step, you modify your report definition so that it displays only those report rows that contain summary computations.

Start on the Report Format - Sort screen. To get there, type `sort` on the command line of any screen.

Type a blank next to `Detail` to suppress the display of detail lines.
### Displaying Only Detail Lines

In this step, you modify your report definition so that it displays only detail lines (those lines that do not contain any summary computations).

Start on the Report Format - Sort screen. To get there, type `sort` on the command line of any screen. Type a character next to `Detail` to restore the display of detail lines. Type a blank next to `Summary` to suppress the display of rows containing summary computations.

```sql
CA OLQ displays only rows containing summary computations. Page right (PF11) to see the entire report.
```

```sql
Display lines: Detail and/or Summary X Group by all X
```
Skipping To a New Page After Computations

In this step, you request that CA OLQ skip to a new page after each change in the group field.

Before you specify this option, you must establish a current group field and group level.

<table>
<thead>
<tr>
<th>Step</th>
<th>What To Do</th>
<th>How To Do It</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Go to the Report Format - Group By screen.</td>
<td>Type group by in the command line of any screen.</td>
</tr>
<tr>
<td>2</td>
<td>Establish the current group field and level.</td>
<td>Type the group field in the Group by field. Type the corresponding level number in the Level# field. Press [Enter].</td>
</tr>
<tr>
<td>3</td>
<td>Ask CA OLQ to skip to a new page after a computation.</td>
<td>Enter pg in the Skip lines after group field.</td>
</tr>
</tbody>
</table>
Specify `employee.dept-id` as your current group field. Specify 1 as the current group level.

```
CA OLQ Release nn.n

*** Report Format - Group By ***
Page  1 OF  2
136000 Specify summary computations and press the ENTER key

Group by: `employee.dept-id`          Level# 1

<table>
<thead>
<tr>
<th>Seq</th>
<th>Sum</th>
<th>Avg</th>
<th>Max</th>
<th>Min</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE</td>
<td>EMP-LAST-NAME</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
</tr>
<tr>
<td></td>
<td>DEPT-ID</td>
<td>1</td>
<td>_</td>
<td>_</td>
<td>_</td>
</tr>
<tr>
<td></td>
<td>SALARY-AMOUNT</td>
<td>4</td>
<td>_</td>
<td>_</td>
<td>_</td>
</tr>
<tr>
<td></td>
<td>PROJECT</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
</tr>
</tbody>
</table>

Compute fields:
- `TOTAL-3-ALL=TOTAL(EMPLOYEE.SALARY-AMOUNT) GROUP BY ALL LEVEL 1`
- `EMPLOYEE.SALARY-AMOUNT-AVE-2'=AVE(EMPLOYEE.SALARY-AMOUNT) GROUP BY EMPLOYEE.DEPT-ID LEVEL 1`

Skip lines after group 1
Separator character -

Compute:
Type `sort` in the command line. CA OLQ echoes the screen, listing the current group field and level.
Type `pg` next to Skip lines after group to specify that you want CA OLQ to skip a page after each change in the current group field (DEPT-ID).

```
CA OLQ Release nn.n

*** Report Format - Group By ***
Page  1 OF  2
136000 Specify summary computations and press the ENTER key

Group by: `EMPLOYEE.DEPT-ID`          Level# 1

<table>
<thead>
<tr>
<th>Seq</th>
<th>Sum</th>
<th>Avg</th>
<th>Max</th>
<th>Min</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE</td>
<td>EMP-LAST-NAME</td>
<td>3</td>
<td>A</td>
<td>_</td>
<td>_</td>
</tr>
<tr>
<td></td>
<td>DEPT-ID</td>
<td>1</td>
<td>_</td>
<td>_</td>
<td>_</td>
</tr>
<tr>
<td></td>
<td>SALARY-AMOUNT</td>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
</tr>
</tbody>
</table>

Skip lines after group `pg`
Separator character -

Compute:
Type a character next to Summary to restore the display of lines containing summary computations.

```
CA OLQ Release nn.n

*** Report Format - Sort ***
Page  1 of  2
133000 Specify sort or group by request and press the ENTER key

<table>
<thead>
<tr>
<th>Disp</th>
<th>Sort</th>
<th>Order</th>
<th>Group By</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE</td>
<td>EMP-LAST-NAME</td>
<td>3</td>
<td>A</td>
</tr>
<tr>
<td></td>
<td>DEPT-ID</td>
<td>1</td>
<td>_</td>
</tr>
<tr>
<td></td>
<td>SALARY-AMOUNT</td>
<td>4</td>
<td>_</td>
</tr>
</tbody>
</table>
X PROJECT

COMPUTE FIELDS:
X TOTAL-3-ALL=TOTAL(EMPLOYEE.SALARY-AMOUNT) GROUP BY ALL LEVEL 1

X 'EMPLOYEE.SALARY-AMOUNT-AVE-2'=AVE(EMPLOYEE.SALARY-AMOUNT) GROUP BY EMPLOYEE.DEPT-ID LEVEL 1

Display lines: Detail X and/or Summary x Group by all X

Compute:
1=HELP 3=QUIT 4=MESSAGE 5=DISPLAY 6=MENU 8=FWD 11=HEADER

CA OLQ displays each department’s statistics on a separate page. Press [PF8] to page through the whole report.

CA OLQ Release nn.n *** Display Report ***
-> Page 1 Line 1
104009 DISPLAY RIGHT to see more report columns
125000 Press the ENTER key to go to the next page of the report.

EMPLOYEE REPORT

DEPT-ID SALARY-AMOUNT EMP-LAST-NAME
4000 37000.00 ZEDI

18000.00 ANGELO
34000.00 JACKSON
18000.00 MCDUGALL
39000.00 PENMAN

80000.00 BANK

AVERAGE SALARY: 37666.66

1=HELP 3=QUIT 4=MESSAGE 6=MENU 8=FWD

CA OLQ Release nn.n *** Display Report ***
-> Page 2 Line 13
125000 Press the ENTER key to go to the next page of the report.

EMPLOYEE REPORT

DEPT-ID SALARY-AMOUNT EMP-LAST-NAME
5200 38000.00 CLOTH
31000.00 KASPAR

45000.00 FINN
33000.00 TIME
80000.00 WILCO

AVERAGE SALARY: 45400.00

1=HELP 3=QUIT 4=MESSAGE 6=MENU 7=BWD 8=FWD

How to Save a Set of Commands as a Qfile
**What is a qfile?**

A qfile is a way to save a set of CA OLQ commands in a file.

**What do you do with qfiles?**

You can execute a qfile to retrieve data from the database and create a report. The qfile formats the report the same way each time it is executed, but the data in the report changes to reflect the current data in the database.

**Example**

You could create a qfile that displays quarterly financial information. When you execute the routine at the end of each fiscal quarter, the report looks the same each time, but the financial information changes.

**Creating and Executing the QUARTER Qfile:**

<table>
<thead>
<tr>
<th>What You Want to Do</th>
<th>Which CA OLQ Function to Use</th>
<th>Formatting Retained?</th>
<th>Calculations Reexecuted?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quarterly report</td>
<td>Create: Use OLQ to create a quarterly qfile. Use the qfile processing screen to save those commands in IDD as the QUARTER qfile.</td>
<td>Retained</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Execute: Each fiscal quarter execute the QUARTER qfile to create a new report.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Qfiles differ from other CA OLQ functions in that they:**

- Reflect the changing nature of data in the database
- Retain any formatting enhancements you have made to your report
- Re-execute any report calculations each time the qfile is executed
### What You Want to Do

<table>
<thead>
<tr>
<th>What You Want to Do</th>
<th>Which CA OLQ Function to Use</th>
<th>Formatting Retained?</th>
<th>Calculations Reexecuted?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save the set of commands used to build a report</td>
<td>Qfile</td>
<td></td>
<td>Reexecuted each time the report is built</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Save a copy of a report for later use</td>
<td>Saved report</td>
<td>Retained</td>
<td>Executed at report definition</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create a table from your report</td>
<td>Saved table</td>
<td>Not retained</td>
<td>Retained as they are executed when the table is saved</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For more information, see the following topics:

- Qfiles and Reports Key Terms (see page 106)
- Creating a Qfile (see page 107)
- Executing a Qfile (see page 113)
- Using One Qfile to Create Different Reports (see page 115)
- Modifying Your Qfile Definition (see page 119)
- Looking at Your Qfile Definition Syntax (see page 124)
- Modifying Your Qfile Definition Syntax (see page 127)
- Executing a Qfile in Batch Mode (see page 131)
- Deleting a Qfile (see page 133)

---

### Qfiles and Reports Key Terms

Here are some terms used to discuss qfiles and reports:

**Current report**

The report you’re working on in an active CA OLQ session. If you retrieve a saved report, CA OLQ clears out the current report.

**Data dictionary**

The storage facility used by CA products as a central source for data definitions, modules, and runtime information. Qfile definitions are stored in the data dictionary.

**Integrated Data Dictionary (IDD)**

The CA product used to access definitions stored in the dictionary.

**Qfile definition**

The CA OLQ syntax statements stored in the data dictionary when you create your qfile.
Creating a Qfile

Contents
- Step 1 - Build the salary report (see page 107)
- Step 2 - Add formatting enhancements (see page 110)
- Step 3 - Save the report as a Qfile (see page 112)

In this example, you create a qfile using the following steps:

1. Build a SALARY report containing employee salary information.
2. Add formatting enhancements you want kept in the qfile.
3. Save the report definition as a qfile.

Creating a Qfile:
1. Use OLQ to create the SALARY report.
2. Save these comments as the SALARY qfile.

Step 1 - Build the salary report
When you create a qfile, CA OLQ takes the set of commands you used to build your current report and saves them as the qfile definition. In this step, you build your current Salary report.
This report uses the EMPLOYEE and EMPOSITION records from the sample database. These records reside in the EMPSS01 subschema.

Start on the Signon Database View screen. To get there, type sub in the command line of any screen.

Select the EMPSS01 subschema.

Select the EMPLOYEE and EMPOSITION records.

Select EMP-NAME-0415, START-YEAR-0415, START-MONTH-0415, START-DAY-0415, SALARY GRADE-0420, SALARY AMOUNT-0420, and BONUS PERCENT-0420. Page forward (PF8) to view all of the columns you need.
Enter additional criteria:

Proceed to Selection Criteria Screen? N Y/N
1=HELP  3=QUIT  4=MESSAGE  6=MENU  8=FWD  PA2=REFRESH

Select Yes to continue data retrieval.

CA OLQ Release nn.n
*** Retrieval Interrupted ***
->
131000 Select YES or NO and press the ENTER key
Retrieval interrupted due to excessive data base accesses.

Number of whole rows . . . . . . . . 54
Total number of records read . . . . 100
Total number of records selected . . 99
Number of data errors . . . . . . . . 0

Continue execution x Yes
X No

Current interrupt interval is 100 data base accesses.

1=HELP  3=QUIT  4=MESSAGE

Depending on how high the interrupt count at your site is set, CA OLQ may bypass the Retrieval Interrupted screen and proceed directly to the Retrieval Completed screen. Specify that you want to proceed to the Report Format - Sort screen.

CA OLQ Release nn.n
*** Retrieval Completed ***
->
130000 Select activity and press the ENTER key

Number of whole rows . . . . . . . . 68  Total displayable cols . 20
Total number of records read . . . . 125  Formatted line length . . 372
Total number of records selected . . 124
Number of data errors . . . . . . . . 0

Select Option ---> Display/Format Activity <--- Command/
X     Display report   DISplay
     Save report     SAVE
x     Choose the sort sequence of report SORT
     Change column headers HEAder
     Change page header and footer PAGE HEAder
Step 2 - Add formatting enhancements

Your qfile definition saves all of the formatting enhancements you have included in your report. Enhancements include sorts, break processing, external pictures, headings, and report titles.

See Section 5, "How to Format Your Report" for more information on how to format your report.

In this step, you enhance your qfile definition by:

- Sorting the report rows by salary grade
- Grouping the report rows according to salary grade
- Changing the report headers

Start on the Report Format - Sort screen. To get there, type `sort` on the command line of any screen.

To sort the rows by salary grade, specify 1 in the Sort Priority column next to SALARY-GRADE-0420. Specify a (for ascending) in the Order (A/D) column next to SALARY-GRADE-0420.

To group your report rows by salary grade, specify 1 in the Group By Level # column next to SALARY-GRADE-0420.

Display lines: Detail X and/or Summary X Group by all _ Compute:

Type `header` on the command line.
Group by COLUMN: SALARY GRADE-0420

<table>
<thead>
<tr>
<th>Seq</th>
<th>Total</th>
<th>Avg</th>
<th>Hival</th>
<th>Loval</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP FIRST NAME-0415</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMP LAST NAME-0415</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>START YEAR-0415</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>START MONTH-0415</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>START DAY-0415</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMPOSITION</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SALARY GRADE-0420</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SALARY AMOUNT-0420</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BONUS PERCENT-0420</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Skip lines after group 1  Separator character -

Compute:
1=HELP 3=QUIT 4=MESSAGE 5=DISPLAY 6=MENU 10=PICTURE

Change the display of the date columns to list them in order of month, day, and year. Change your headers to make them more legible.

Underline character: -

<table>
<thead>
<tr>
<th>Seq</th>
<th>Header</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEE</td>
<td></td>
</tr>
<tr>
<td>EMP FIRST NAME-0415</td>
<td>1 First Name</td>
</tr>
<tr>
<td>EMP LAST NAME-0415</td>
<td>2 Last Name</td>
</tr>
<tr>
<td>START YEAR-0415</td>
<td>3 Start</td>
</tr>
<tr>
<td>START MONTH-0415</td>
<td>4</td>
</tr>
<tr>
<td>START DAY-0415</td>
<td></td>
</tr>
<tr>
<td>EMPOSITION</td>
<td></td>
</tr>
<tr>
<td>SALARY GRADE-0420</td>
<td>6 Grade</td>
</tr>
<tr>
<td>SALARY AMOUNT-0420</td>
<td>7 Salary Amount</td>
</tr>
<tr>
<td>BONUS PERCENT-0420</td>
<td>8 Bonus</td>
</tr>
</tbody>
</table>

Compute:
1=HELP 3=QUIT 4=MESSAGE 5=DISPLAY 6=MENU 8=FWD 10=SORT 11=EDIT

Sort successfully completed. 68 records in. 68 records out.

Press the ENTER key to go to the next page of the report.

<table>
<thead>
<tr>
<th>FIRST NAME</th>
<th>LAST NAME</th>
<th>START</th>
<th>GRADE</th>
<th>SALARY AMOUNT</th>
<th>BONUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOM</td>
<td>FITZHUGH</td>
<td>09 19</td>
<td>81</td>
<td>13000.00</td>
<td>.004</td>
</tr>
<tr>
<td>NANCY</td>
<td>TERNER</td>
<td>05 26</td>
<td>82</td>
<td>13000.00</td>
<td>.004</td>
</tr>
<tr>
<td>CYNTHIA</td>
<td>JOHNSON</td>
<td>03 23</td>
<td>77</td>
<td>13500.00</td>
<td>.004</td>
</tr>
<tr>
<td>ROBIN</td>
<td>GARDNER</td>
<td>06 15</td>
<td>81</td>
<td>14000.00</td>
<td>.004</td>
</tr>
<tr>
<td>BRIAN</td>
<td>NICEMAN</td>
<td>05 06</td>
<td>80</td>
<td>14000.00</td>
<td>.004</td>
</tr>
<tr>
<td>DORIS</td>
<td>KING</td>
<td>08 16</td>
<td>80</td>
<td>14500.00</td>
<td>.004</td>
</tr>
<tr>
<td>SANDY</td>
<td>KRAAMER</td>
<td>04 04</td>
<td>81</td>
<td>14000.00</td>
<td>.004</td>
</tr>
</tbody>
</table>

AVE FOR 11: 13166.66
Step 3 - Save the report as a Qfile

You now save your report as a qfile. CA OLQ takes all of the commands used to build the current report and saves them in a qfile definition in the data dictionary.

Start on the Qfile Processing screen. To get there, type qfile in the command line.

Select Create. Specify salary next to Routine name. Add any descriptive comments you would like to be shown on the screen.

CA OLQ returns a message indicating that your current report definition has been saved as a qfile. And CA OLQ lists the SALARY qfile.
Executing a Qfile

In this example, you execute a qfile to create a report.

**Executing a Qfile:**

1. **On the qfile processing screen**, select the **EXECUTE activity** and the **SALARY qfile**.

2. **OLQ** retrieves the **SALARY qfile definition** from the data dictionary and executes it, creating the salary report.

![Diagram showing the flow of executing a Qfile]

Start on the Qfile Processing screen. To get there, type `qfile` in the top line of any screen.

Select **Execute**. Name the **SALARY qfile**.

```
140000 Select function and press the ENTER key
User: SYB
Dictionary name: TSTDICT
```

<table>
<thead>
<tr>
<th>Function:</th>
<th>x Execute</th>
<th>Create</th>
<th>Replace</th>
<th>Delete</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Execute with new criteria</td>
<td>List</td>
<td>Edit</td>
<td></td>
</tr>
</tbody>
</table>
Execute batch

Routine name:                             Version:
Comments:

<table>
<thead>
<tr>
<th>SELECT ROUTINE</th>
<th>VERSION</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEPARTMENT</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>MANAGERS</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>NEW EMPLOYEES</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>SALARY Grade</td>
<td>1 SALARY SUMMARY COMPUTATIONS</td>
<td></td>
</tr>
<tr>
<td>SALARY</td>
<td>1 EXAMPLE</td>
<td></td>
</tr>
</tbody>
</table>

1=HELP            3=QUIT              4=MESSAGE              6=MENU

Specify Yes to continue data retrieval.

CA OLQ Release nn.n

*** Retrieval Interrupted ***

092018 The default DICTNAME value has been modified.
092027 Underline character has been modified to -
Retrieval interrupted due to excessive data base accesses.

Number of whole rows: . . . . . . . . . 54
Total number of records read: .. 100
Total number of records selected: .. 99
Number of data errors: . . . . . . . . . 0

Continue execution x Yes
X No

Current interrupt interval is 100 data base accesses.

1=HELP            3=QUIT              4=MESSAGE

Depending on how high the interrupt count at your site is set, CA OLQ may bypass the Retrieval Interrupted screen and proceed directly to the Retrieval Completed screen.

CA OLQ Release nn.n

*** Retrieval Completed ***

105022 Sort successfully completed. 68 records in. 68 records out.
092021 Execution has completed for q-file: SALARY

Number of whole rows: . . . . . . . . . . 68
Total number of records read: .. 125
Total number of records selected: .. 124
Number of data errors: . . . . . . . . . 0

Select Option ---> Display/Format Activity <--- Command/

| X    | Display report | DISplay |
| ---  | Save report    | SAVe    |
| _    | Choose the sort sequence of report | SORT |
| _    | Change column headers | HEAder |
| _    | Change page header and footer | PAGe HEAder |
| _    | Change display format of data ($,commas) | PICTure |
| _    | Format columns (Alignment, sparse) | EDIt |
| _    | Specify summary computations (Totals) | GROup BY |
| _    | Send the report to a printer | PRInt |

1=HELP            3=QUIT              4=MESSAGE              6=MENU
Using One Qfile to Create Different Reports

Contents

- Step 1 - Use execute with new criteria (see page 116)
- Step 2 - Change the Selection criteria (see page 117)
- Step 3 - Resume Qfile execution (see page 117)

In this example, you use the same qfile to create two or more reports that are nearly the same. Using **Execute with new criteria**, you can suspend qfile execution and change the retrieval selection criteria used to build the report.

**Example**

By executing the SALARY qfile and modifying the retrieval criteria, you can create two reports:

- One displays all salary grades above $30,000.
- One displays all salary grades below $30,000.

**Using One Qfile to Make More Than One Report:**

---

EMPLOYEE/EMPOSITION REPORT

FIRST NAME                LAST NAME    START      GRADE    SAL.     BONUS

TOM        FITZHUGH     09 19 81    11        13000.00 .004
NANCY      TERNER       05 26 82    11        13000.00 .004
CYNTHIA    JOHNSON      03 23 77    11        13500.00 .004

AVE FOR 11:   13166.66

ROBIN      GARDNER      06 15 81    12        14000.00 .004
BRIAN      NICEMAN      05 06 80    12        14000.00 .004
DORIS      KING         08 16 80    12        14500.00 .004
SANDY      KRAAEMER     04 04 81    12        14000.00 .004

- 1 -

1=HELP   3=QUIT   4=MESSAGE   6=MENU   8=FWD   10=LEFT  11=RIGHT
IDMSDB--Using One Qfile to Create Different Reports

In this example, you execute the SALARY qfile using the following steps:

1. Execute the SALARY qfile using **Execute with new criteria**.

2. CA OLQ starts to execute the qfile. Execution continues up to the first statement that would retrieve data from the database.

3. CA OLQ suspends execution of the qfile and displays the Column Select screen.

4. You can change the retrieval criteria to display only those salary grades larger than 16.

5. When you press [Enter], CA OLQ resumes execution of the qfile.

**Step 1 - Use execute with new criteria**

In this step, you use the Execute with new criteria option to suspend execution of the SALARY qfile before it retrieves any data from the database.

Start on the Qfile Processing screen. To get there, type `qfile` in the command line of any screen.

Select **Execute with new criteria** and select the SALARY qfile.

CA OLQ Release nn.n

140000 Select function and press the ENTER key

*** QFILE Processing ***
Step 2 - Change the Selection criteria

In this step, you change the data retrieval selection criteria to retrieve only those employees in salary grades greater than 16.

Specify **greater than 16** in the **Selection criteria** column for the SALARY-GRADE-0420 field.

Additional selection criteria:

```
1=HELP 3=QUIT 4=MESSAGE 6=MENU
```

Step 3 - Resume Qfile execution

In this step, you resume execution of the qfile to apply new data retrieval selection criteria.

The final report displays only those employees in salary grades of 16 or higher.

Select **Yes** to continue data retrieval.
CA OLQ Release nn.n

*** Retrieval Interrupted ***

131000 Select YES or NO and press the ENTER key

Retrieval interrupted due to excessive data base accesses.

Number of whole rows . . . . . . . 48
Total number of records read . . . . 100
Total number of records selected . . 87
Number of data errors . . . . . . . 0

Continue execution x Yes
X No

Current interrupt interval is 100 data base accesses.

1=HELP 3=QUIT 4=MESSAGE

Depending on how high the interrupt count at your site is set, CA OLQ may bypass the Retrieval Interrupted screen and proceed directly to the Retrieval Completed screen.

CA OLQ Release nn.n

*** Retrieval Completed ***

130000 Select activity and press the ENTER key

Number of whole rows . . . . . . . 59 Total displayable cols . 20
Total number of records read . . . . 125 Formatted line length . 372
Total number of records selected . . 106
Number of data errors . . . . . . . 0

Select Option ---> Display/Format Activity <--- Screen Name
X Display report DISplay
_ Save report SAVe
_ Choose the sort sequence of report SORt
_ Change column headers HEAder
_ Change page header and footer PAGE HEAder
_ Change display format of data ($, commas) PICTure
_ Format columns (Alignment, sparse) EDIt
_ Specify summary computations (Totals) GROup BY
_ Send the report to a printer PRInt
1=HELP 3=QUIT 4=MESSAGE 6=MENU

CA OLQ Release nn.n

*** Display Report ***

125000 Press the ENTER key to go to the next page of the report.

EMPLOYEE/EMPOSITION REPORT

mm/dd/yy

FIRST NAME LAST NAME START GRADE SAL. AMOUNT BONUS

HERBERT LIPSICH 04 29 81 21 18500.00 .004
MICHAEL ANGELO 09 08 79 21 18000.00 .004
RALPH TYRO 12 21 80 21 20000.00 .004
MICHAEL ANGELO 09 08 79 21 17000.00 .004
DOUGLAS KAHALLY 09 29 79 21 20000.00 .004
CAROL MCDougALL 06 07 80 21 18000.00 .004

AVE FOR 21: 18583.33
Modifying Your Qfile Definition

Contents
- Step 1 - Establish a current report (see page 120)
- Step 2 - Modify the report definition (see page 122)
- Step 3 - Replace the Qfile definition (see page 123)

When you modify a qfile, CA OLQ overlays your IDD qfile definition with the commands used to build your current report. Using this feature, you can:

- Execute an existing qfile, change the report, and replace the qfile
- Execute a qfile, change the report, and save it as a different qfile
- Create a new report from scratch and save it under the name of an existing qfile

In this example, you modify the SALARY qfile using the following steps:

1. Establish a current report by executing the SALARY qfile
2. Modify the report definition to suppress the display of detail lines (those not containing any summary calculations)
3. Replace the qfile, overriding the current SALARY qfile definition with the commands used to build the summary report

Modifying Your Qfile Definition:
Step 1 - Establish a current report

In this step, you execute the SALARY qfile to create a current report. Start on the Qfile Processing screen. To get there type `qfile` in the command line of any screen.

Select **Execute** and select the **SALARY qfile**.
Select function and press the ENTER key

User: SYB
Dictionary name: TSTDICT
Function: x Execute _ Create _ Replace _ Delete
 _ Execute with new criteria _ List _ Edit
 _ Execute batch

Routine name: Version:
Comments:

SELECT ROUTINE VERSION COMMENTS
 _ DEPARTMENT 1
 _ MANAGERS 1
 _ NEW EMPLOYEES 1
 _ SALARY GRADE 1 SALARY SUMMARY COMPUTATIONS
x SALARY 1 EXAMPLE

Specify Yes to continue data retrieval.

Specify that you want to proceed to the Report Format - Sort screen.

Current interrupt interval is 100 data base accesses.
Step 2 - Modify the report definition

In this step, you modify the SALARY qfile to suppress the display of all detail lines (those not containing any summary calculations) in the report. Thus the report displays only the average salary for each salary grade.

Type a space next to Detail to suppress the display of detail lines.

Display lines: Detail and/or Summary X Group by all _

Compute:

Type qfile in the command line to proceed to the Qfile Processing screen.
Step 3 - Replace the Qfile definition

In this step you override the SALARY command file with the commands used to make the current report.

Specify Replace. Select the SALARY qfile.

CA OLQ responds with a message that the qfile has been replaced.

CA OLQ responds with a message that the qfile has been replaced.
Looking at Your Qfile Definition Syntax

Contents
- Step 1 - Switch to IDD (see page 126)
- Step 2 - View your Qfile in IDD (see page 126)
- Step 3 - Leave IDD (see page 127)

Where are qfile definitions stored?
When you create a qfile, CA OLQ takes all the commands used in your current session and saves them as CA OLQ syntax statements in the data dictionary. Definitions stored in the data dictionary are accessed using a CA product called the Integrated Data Dictionary (IDD).

How to look at your qfile definition
Using the List option on the Qfile Processing screen, you can switch from CA OLQ to IDD to see how your qfile commands are stored in the data dictionary.

For more information on IDD refer to the CA IDMS IDD DDDL Reference Section.

⚠️ Note: Depending on the security level that has been assigned to you, you may or may not be able to access your qfile definition in IDD. If your specifying List or Edit on the Qfile Processing screen does not switch you out of CA OLQ, you probably do not have authority to access IDD.

Looking at Qfile Syntax:
IDMSDB—Looking at Your Qfile Definition Syntax

In this example, you see how the SALARY qfile is stored in the data dictionary by using the following steps:

1. Switch to IDD using the List option on the Qfile Processing screen.
2. Look at the SALARY qfile definition in IDD.
3. Transfer from IDD back to CA OLQ.
Step 1 - Switch to IDD

Select List and the SALARY qfile.

```
> CA OLQ Release nn.n *** QFILE Processing ***
109017 Requested operation for SALARY (1) has been successfully completed
140000 Select function and press the ENTER key
User: SYB
Dictionary name: TSTDICT Dictionary node:
Function: _ Execute _ Create _ Replace _ Delete
_ Execute with new criteria x List _ Edit
_ Execute batch
Routine name: Version:
SELECT ROUTINE VERSION COMMENTS
  - DEPARTMENT 1
  _ MANAGERS 1
  _ NEW EMPLOYEES 1
  _ SALARY GRADE 1 SALARY SUMMARY COMPUTATIONS
  x SALARY 1 EXAMPLE

1=HELP 3=QUIT 4=MESSAGE 6=MENU
```

Step 2 - View your Qfile in IDD

In this step, you look at your qfile definition:

- **To page forward** through the qfile definition, press [PF8].
- **To page backward**, press [PF7].

Note that whenever CA OLQ retrieves data from the database, it creates a Structured Query Language (SQL) SELECT statement. The data retrieval selection criteria that you specified on the Column Select screen are specified in the WHERE clause of the SELECT statement.

For a more detailed explanation of this and other CA OLQ syntax statements, refer to the CA OLQ Reference Section.

Your qfile syntax includes a SELECT (OLQ access mode) statement. The WHERE clause specifies data retrieval selection criteria.

```
** SET DICTNAME TSTDICT
** SET UNDERLINE '. '
** SET ACCESS OLQ** SIGNON SS EMPSS01 SCHEMA EMPSCHM ( 100)
** OPTIONS ALL HEADER ECHO NOFILLER FULL WHOLE INTERRUPT OLQHEADER
** NOPATHSTAT NOSTAT COMMENT VERBOSE NDBKEY PICTURE CODETAB NOSYN
** SELECT EMPLOYEE.EMP-FIRST-NAME-0415 EMPLOYEE.EMP-LAST-NAME-0415 EMPLOYEE.ST
** - YEAR-0415 EMPLOYEE.START-MONTH-0415 EMPLOYEE.START-DAY-0415 EMPOSITION.SAL
** - GRADE-0420 EMPOSITION.SALARY-AMOUNT-0420 EMPOSITION.BONUS-PERCENT-0420 FRO
** EMPLOYEE, EMPOSITION WHERE (EMP-EMPOSITION.EMPLOYEE.EMPOSITION)**+ PAGE HEADER BLANK LINES AFTER 1 -
```
Step 3 - Leave IDD

In this step, you switch from IDD back to CA OLQ.

Type **end** in the command line.

```
end
```

```
*+ SET DICTNAME TSTDICT
*+ SET UNDERLINE '.
*+ SET ACCESS OLQ
*+ SIGNON SS EMPS01 SCHEMA EMPSCMH ( 100)
*+ OPTIONS ALL HEADER ECHO NOFILLER FULL WHOLE INTERRUPT CA OLQHEADER
*+ NOPATHSTAT NOSTAT COMMENT VERBOSE NODBKEY PICTURE CODETAB NOSYN
*+ SELECT EMPLOYEE.EMP-FIRST-NAME-0415 EMPLOYEE.EMP-LAST-NAME-0415 EMPLOYEE.START-YEAR-0415 EMPLOYEE.START-MONTH-0415 EMPLOYEE.START-DAY-0415 EMPOSITION.SALGRADE-0420 EMPOSITION.SALARY-AMOUNT-0420 EMPOSITION.BONUS-PERCENT-0420 FRO
*+ EMPLOYEE, EMPOSITION WHERE (EMP-EMPOSITION.EMPLOYEE.EMPOSITION)
*+ PAGE HEADER BLANK LINES AFTER 1
*+ LINE 1 'EMPLOYEE/EMPOSITION REPORT' CENTER -
*+ LINE 2 '}$DATE' CENTER
*+ PAGE FOOTER BLANK LINES BEFORE 1
*+ EDIT EMP-FIRST-NAME-0415 -
  *+ ALIGN LEFT -
  *+ CA OLQHEADER 'FIRST'-
  *+ 'NAME'-
  *+ PICTURE 'X(10)'
*+ EDIT EMP-LAST-NAME-0415 -
  *+ ALIGN LEFT -
  *+ CA OLQHEADER 'LAST'-
```
3. Replace the SALARY qfile definition.

4. Switch from IDD back to CA OLQ.

**Step 1 - Switch to IDD**

In this step, you suspend your current CA OLQ session and transfer to the Integrated Data Dictionary (IDD).

Select **Edit** and the **SALARY** qfile.

```plaintext
CA OLQ Release nn.n
->

140000 Select function and press the ENTER key

User: SYB
Dictionary name: TSTDICT

Function: Execute Create Replace Delete
- Execute with new criteria
- Execute batch

Routine name: Version:

SELECT ROUTINE VERSION COMMENTS
- DEPARTMENT 1
- MANAGERS 1
- NEW EMPLOYEES 1
- SALARY GRADE 1 SALARY SUMMARY COMPUTATIONS
- SALARY 1 EXAMPLE

1=HELP 3=QUIT 4=MESSAGE 6=MENU
```

⚠️ **Note:** When you first see the qfile definition, erase the first three lines (those that sign you on to IDD, specify the dictionary name, and display the qfile). After you have erased these lines, your first line should begin with the verb MODIFY. If you don't do this, IDD will see DISPLAY as the first verb when you press [Enter], and will redisplay the qfile (without the changes you have made).

In this example, VERB MODIFY means that the qfile definition can be updated. Press [PF8] to scroll to the next page.

```plaintext
IDD nn.n ONLINE NO ERRORS DICT=TSTDICT 1/86
SIGNON DICTNAME= TSTDICT NODENAME= ' ' USA UPD.
DISPLAY QFILE 'SALARY

MODIFY
QFILE NAME IS SALARY VERSION IS 1
LANGUAGE IS OLQ
TEXT IS OLQ
++ DATE CREATED IS 02/20/91
++ DATE LAST UPDATED IS 02/20/91
++ TIME LAST UPDATED IS 11160166
++ PREPARED BY SYB
++ REVISED BY SYB
```
Step 2 - Modify the SALARY Qfile Definition

In this step, you change the SALARY qfile definition using the following steps:

1. Press [PF4] to supply space to type in your changes.

2. Add a restriction to the SELECT statement WHERE clause that limits retrieval to salary grades higher than 18.

3. Press [PF5] to apply the change.

For more information on how to change dictionary definitions in IDD, refer to the CA IDMS Dictionary Module Editor User Section.

In this example, your qfile syntax includes a SELECT (OLQ access mode) statement. Position the cursor on the line that contains the WHERE clause. Press [PF4] to insert additional criteria.

IDD nn.n ONLINE PAGE 2 LINE 1 DICT=TSTDICT 24/86
QFILE SOURCE FOLLOWS
SET DICTNAME TSTDICT
SET UNDERLINE '-'
SET ACCESS OLQ SIGNON SS EMPSS01 SCHEMA EMPSCHM (100)
OPTIONS ALL HEADER ECHO NOPATHSTAT NOSTAT COMMENT VERBOSE NODBKEY PICT URE CODETAB NOSYN

Type a blank over the closing parenthesis on the WHERE clause. Add AND SALARY-GRADE-0420 GT 18] to the WHERE clause. Don’t forget the closing parenthesis. Then press [PF5] to apply the change.

IDD nn.n ONLINE PAGE 2 LINE 11 DICT=TSTDICT 34/86
Step 3 - Replace the SALARY Qfile definition

In this step, you replace the qfile definition in the data dictionary.

Press [PF6] to replace the qfile definition in the data dictionary.

IDD nn.n ONLINE PAGE 2 LINE 11 DICT=TSTDICT 34/91
EMPLOYEE, EMPOSITION WHERE (EMP-EMPOSITION.EMPLOYEE.EMPOSITION AND SALARY-GRADE -0420 GT 18)
PAGE HEADER BLANK LINES AFTER 1
LINE 1 'EMPLOYEE/EMPOSITION REPORT' CENTER
LINE 2 'SDATE' CENTER
PAGE FOOTER BLANK LINES BEFORE 1
LINE 1 '. $PAGE . ' CENTER
EDIT EMP-FIRST-NAME-0415 -
  ALIGN LEFT
  OLQHEADER 'FIRST'
  'NAME'.
  PICTURE 'X(10)'
EDIT EMP-LAST-NAME-0415 -
  ALIGN LEFT
  OLQHEADER 'LAST'.
  'NAME'.
  PICTURE 'X(15)'
EDIT START-YEAR-0415 -
  ALIGN RIGHT
  OLQHEADER ' '.
  PICTURE '99'
EDIT START-MONTH-0415 -
  ALIGN RIGHT

IDD responds with the message NO ERRORS on the top line of the screen.

SIGNON DICTNAME= TSTDICT NODENAME= ' ' USA UPD.
DISPLAY QFILE 'SALARY' VERSION 1 AS SYN VERB MODIF

*+ DATE CREATED IS 02/20/91
*+ DATE LAST UPDATED IS 02/20/91
*+ TIME LAST UPDATED IS 11160166
*+ PREPARED BY SYB
*+ REVISED BY SYB
DESCRIPTION IS OLQ
USER IS SYB
REGISTERED FOR ALL RESPONSIBLE FOR CREATION AND UPDATE AND DELETION
TEXT IS OLQ
PUBLIC ACCESS IS ALLOWED FOR ALL
*+ ACCESSED BY USER SYB
COMMENTS 'EXAMPLE'

Step 4 - Switch from IDD back to CA OLQ

In this step, you terminate your IDD session and switch back to the Qfile Processing screen.

Type end in the command line to get back to CA OLQ.
Executing a Qfile in Batch Mode

Contents
- Step 1 - Select a Qfile (see page 132)
- Step 2 - Using the Batch Processing screen (see page 133)

When do you use batch?
If you have a qfile that you know is going to access a large amount of data, or if you want to execute your qfile overnight, the batch option is the most efficient operating method. Executing a job in batch mode frees computer resources for other users. Your DBA can help you decide when to execute your qfile in batch.

Because using batch requires advance preparation by your DBA, you will not actually execute a qfile in this example. This example shows you the CA OLQ tools you need to know about to execute a qfile in batch mode.

You will:

1. Select a qfile.
2. Transfer to the Batch Processing screen.

**Step 1 - Select a Qfile**

Select **Execute batch** and select the **SALARY** qfile.

```
User: SYB
Dictionary name: TSTDICT
Function: _ Execute _ Create _ Replace _ Delete
          _ Execute with new criteria _ List _ Edit
          x Execute batch
```

**Routine name:** Version:

```
SELECT ROUTINE VERSION COMMENTS
  _ DEPARTMENT 1
  _ MANAGERS 1
  _ NEW EMPLOYEES 1
  _ SALARY GRADE 1 SALARY SUMMARY COMPUTATIONS
  x SALARY 1 EXAMPLE
```

The default values for executing a qfile in batch are displayed when you access this screen from the Qfile Processing screen.
Step 2 - Using the Batch Processing screen

To execute a qfile in batch mode:

1. Enter your user password.

2. Make sure that the proper Job Control Module is displayed. Ask your DBA to make sure you have the right name.

3. Select an output destination, or accept the default destination that has been listed for you. In addition, the output of the batch job is always displayed at your terminal.

4. Check to see if the correct qfile name is listed in the Enter OLQ Syntax/Comments: field.

5. Press [Enter] to initiate your batch job.

For more information on how to use batch processing in CA OLQ, refer to Section12, "How to Use CA OLQ in Batch Mode".

Deleting a Qfile

When you delete a qfile, CA OLQ uses the following:

- Deletes the qfile definition in the data dictionary
- Removes the name of the definition from the list of qfiles on the Qfile screen

In this example, you delete a qfile using the Qfile Processing screen.

Start on the Qfile Processing screen. To get there, type **qfile** in the command line of any screen.

Select **Delete** and select the **SALARY** qfile.
CA OLQ responds with a message indicating that the qfile definition has been deleted. The name of the qfile is deleted from the list of routine names.

How to Make a Report from Database Records

In this section Using CA OLQ, you can report on records stored in a CA IDMS/DB.

Access mode: This section only applies when the access switch is set to OLQ.

For information on setting the access switch, see Step 1? Select the type of table.

For example, you can make an employee report based on data retrieved from the EMPLOYEE record.

Creating a Report:
Key Terms 5

Here are some of the terms used to discuss making a report from database records:

Database view

A more descriptive term for subschema. The two terms are used interchangeably.

Element
The smallest significant unit of data in a CA IDMS/DB database. Record elements correspond to columns in a table. For example, the record element DEPT-ID-0410 corresponds to the DEPARTMENT ID column in a table.

**Record**

A group of related elements. For example, the DEPT-NAME-0410, DEPT-ID-0410, and DEPT-HEAD-ID-0410 elements are all grouped into the DEPARTMENT record. Records correspond to rows in a table. For example, the record element DEPT-NAME-0410 corresponds to the DEPARTMENT NAME column in a table.

**Selection criteria**

A logical expression that you use to tell CA OLQ which rows of data to retrieve for your report.

**Additional selection criteria**

A logical expression that you use to tell CA OLQ which rows of data to retrieve for your report. Unlike selection criteria, you can also use:

- Logical record keywords
- Criteria expressions for subscripted fields

**Subschema**

A view of the database that contains a subset of the records, elements, sets, and areas that make up the entire database. A subschema usually views functionally related data.

For example, the personnel department uses a subschema that views employee information such as salary, date of hire, and personal information. All of the information used by the corporation (sales, accounting, and personnel) are held in the same database, but the personnel department views only the information that it needs.

**Using Subschemas:**

...
Create a Report

In this sample, you create a report using the following steps:

- **Step 1 -- Choose a subschema.** (see page 137)
- **Step 2 -- Select your records** (see page 138)
- **Step 3 -- Choose columns and specify selection criteria** (see page 138)
- **Step 4 -- Enter additional selection criteria** (see page 143)
- **Step 5 -- Retrieve the data and display your report** (see page 144)

**Step 1 -- Choose a subschema.**

In this step, you select a subschema that views employee information.

Select the EMPSS01 subschema.

CA OLQ Release nn.n

*** Signon Database View ***

Page 3 OF 4

10-Jan-2018

137/222
CA IDMS - 19.0

121000 Select a subschema and press the ENTER key

- Specify Subschema:
  - x EMPSS01 OF EMPSCHM VER 100 DEPARTMENT AND EMPLOYEE INFORMATION
  - FINAN01 OF EMPSCHM VER 100 3091
  - SALES01 OF EMPSCHM VER 100 SALES QUOTAS

1=HELP 3=QUIT 4=MESSAGE 6=MENU 7=BWD 8=FWD PA2=REFRESH

Step 2 -- Select your records

In this step, you specify that you want to report on the EMPLOYEE and OFFICE records.

123000 Select records and press the ENTER key

Enter records:
- Select records:
  - EMPLOYEE
  - EMPOSITION
  - EXPERTISE
  - HOSPITAL-CLAIM
  - JOB
  - NON-HOSP-CLAIM
  - OFFICE
  - SKILL
  - STRUCTURE

1=HELP 3=QUIT 4=MESSAGE 6=MENU PA2=REFRESH

Step 3 -- Choose columns and specify selection criteria

In this step, you:

1. Choose which columns you want to display in the report.

2. Specify retrieval selection criteria that restrict what data is retrieved from the database. In this sample, your selection criteria limit the report to those employees from Boston.
124000 Select columns, specify selection criteria and press the ENTER key

Columns currently selected: 0 Selection criteria

EMPLOYEE
x 02 EMP-ID-0415 *
 02 EMP-NAME-0415
- 03 EMP-FIRST-NAME-0415 *
 03 EMP-LAST-NAME-0415 *
- 02 EMP-ADDRESS-0415
- 03 EMP-STREET-0415
x 03 EMP-CITY-0415 eq boston
- 03 EMP-STATE-0415
- 04 EMP-ZIP-FIRST-FIVE-0415
- 04 EMP-ZIP-LAST-FOUR-0415
x 02 EMP-PHONE-0415
- 02 EMP-PHONE-0415
- 02 STATUS-0415
Additional Selection Criteria:

Proceed to Selection Criteria Screen? N Y/N
1=HELP 3=QUIT 4=MESSAGE 6=MENU 8=FWD PA2=REFRESH

More about selection criteria

Selection criteria are logical expressions you use to tell CA OLQ which rows of data you want to include in your report. You don't have to specify any selection criteria. If you don't, CA OLQ retrieves all rows from the record.

If you specify... Your report looks like...

EMPLOYEE Report

Emp ID Last Name City State

EMP-ID-0415 EQ '0075' -> 0075
Lanzarotta Lowell MA

EMPLOYEE Report

Emp ID Last Name City State

EMP-CITY-0415 EQ 'SHELBURNE FALLS' -> 3302
Elopoulos Shelburne Falls MA
4230 Ho Shelburne Falls MA
3871 Mahoney Shelburne Falls MA

EMPLOYEE Report

Emp ID Last Name City State

EMP-STATE-0415 NE 'MA' -> 2789
Vangelis Buckatunna MS
5558 Runningbrook Casper WY

EMPLOYEE Report

Emp ID Last Name City State
How to specify selection criteria

Each column has its own Selection criteria field. If you want to retrieve rows based on the value in a certain column, fill in the Selection criteria field next to that column. For example:

```
X 02 COLUMN1
02 COLUMN2
X 02 COLUMN3 = 2000
X 02 COLUMN4 > 5 * COLUMN1
```

The following table gives examples of the types of operators you can include in your selection criteria.

<table>
<thead>
<tr>
<th>Type of Operator</th>
<th>Operator Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical comparison operators</td>
<td>= Equal to</td>
<td>= 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>= 100 to 500 (exclusive)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>= 100 thru 500 (inclusive)</td>
</tr>
<tr>
<td></td>
<td>!= Not equal to</td>
<td>!= COLUMN2</td>
</tr>
<tr>
<td></td>
<td>&gt; Greater than</td>
<td>&gt; COLUMN1 * COLUMN2</td>
</tr>
<tr>
<td></td>
<td>&gt;= Greater than or equal to</td>
<td>&gt;= COLUMN4 + 100</td>
</tr>
<tr>
<td></td>
<td>&lt; Less than</td>
<td>&lt; COLUMN4 - COLUMN1</td>
</tr>
<tr>
<td></td>
<td>&lt;= Less than or equal to</td>
<td>&lt;= 750 / COLUMN4</td>
</tr>
<tr>
<td>Arithmetic operators</td>
<td>+ Addition</td>
<td>COLUMN1 + COLUMN2</td>
</tr>
<tr>
<td></td>
<td>- Subtraction</td>
<td>COLUMN1 - 5</td>
</tr>
<tr>
<td></td>
<td>* Multiplication</td>
<td>COLUMN1 * 1/2</td>
</tr>
<tr>
<td></td>
<td>/ Division</td>
<td>COLUMN1/COLUMN2</td>
</tr>
<tr>
<td>MATCHES and CONTAINS characters</td>
<td>* Stands for any character</td>
<td>contains &quot;Z**05&quot;</td>
</tr>
<tr>
<td></td>
<td>@ Stands for any alphabetic character</td>
<td>matches &quot;NEW@@@@@@@@@@&quot;</td>
</tr>
<tr>
<td></td>
<td># Stands for any numeric character</td>
<td>matches &quot;669#####&quot;</td>
</tr>
</tbody>
</table>

The IN clause

You can include the IN clause in your selection criteria. The IN clause is a subset of the SQL SELECT statement that compares your column value to a list of data values.

For complete syntax and syntax rules, refer to the CA OLQ Reference Section.

IN IN yields a true comparison if the column value matches one or more of the data values. For example, you can compare the value of the EMP-LAST-NAME field to the following IN clause:
IN (‘JONES’, ‘TANAKA’, ‘ANDERSON’)

NOT IN NOT IN yields a true comparison if the column value does not match or more of the data values. For example, you can compare the value of the EMP-LAST-NAME field to the following IN clause:

NOT IN (‘JONES’, ‘TANAKA’, ‘ANDERSON’)

The LIKE clause

You can also include the LIKE clause in your selection criteria. The LIKE clause is a subset of the SQL SELECT statement that searches for a pattern string.

For complete syntax and syntax rules, refer to the CA OLQ Reference Section.

How to use the LIKE clause

Your LIKE clause contains two parts:

- The LIKE/NOT LIKE keywords
- A pattern string

LIKE

LIKE determines whether a column expression contains a pattern string. LIKE yields a true comparison if the column contains the pattern. For example, you can compare the EMP-LAST-NAME field to the following LIKE clause:

LIKE ‘MAC%’

This comparison is true if the last name is any number of characters long and begins with the string MAC.

NOT LIKE

NOT LIKE yields a true comparison if the column does not contain the pattern. For example, you can compare the OFFICE-CODE field to the following NOT LIKE clause:

NOT LIKE ‘002’

This comparison is true if the office code is anything except for 002.

Pattern string

The pattern string is what is compared to the column value. The pattern can contain:

- **Alphanumeric characters** for an exact match
- **Special characters** used as wild cards
- **An escape character** to exactly match the special characters

The LIKE clause is summarized in the table below.
Object String | Pattern String | Example of Syntax | Example of True comparison
---|---|---|---
Underscore (_ ) | Any single character | NAME LIKE 'S_' | True if NAME is exactly 3 characters long and the first character is S
Percent sign (%) | Any sequence of zero or more characters | NAME LIKE '%C_' | True if NAME is 3 or more characters long and the third from last character is C
Single alphanumeric character | Exact match to that alphanumeric character | NAME LIKE 'MAC' | True if NAME is MAC
Escape character + underscore (_) | Exact match to the underscore (_) | PARTNUM LIKE '*_115' ESCAPE '*1' | True if PARTNUM is __115
Escape character + percent sign (%) | Exact match to the percent sign (%) | PARTNUM LIKE '*%15' ESCAPE '*1' | True if PARTNUM is *%15
Escape character alone | Exact match to the escape character | PARTNUM LIKE '***' | True if PARTNUM is ***

The escape character can be any single alphanumeric character and is set by specifying ESCAPE 'escape-character' in your SELECT statement.

**The AND operator**

If you enter selection criteria for more than one column, CA OLQ combines the expressions by placing the AND operator between them. AND means that a row must meet both conditions in order to be retrieved.

For example, when you specify:

```
X 02 COLUMN1
02 COLUMN2
X 02 COLUMN3  eq 2000
X 02 COLUMN4  gt 5 * COLUMN1
```

CA OLQ produces:

```
COLUMN3  eq 2000  and  COLUMN4  gt 5 * COLUMN1
```

**The OR operator**

You may not want to combine your expressions with AND operators. As an alternative, you can type the OR operator at the beginning of any expression. OR means that a row need only meet one of the conditions in order to be retrieved.

For example, when you specify:

```
X 02 COLUMN1
_ 02 COLUMN2
```

Selection criteria

```
``
CA IDMS - 19.0

X 02 COLUMN3  eq 2000
X 02 COLUMN4  or gt 5 * COLUMN1

CA OLQ produces:
COLUMN3 eq 2000 or COLUMN4 gt 5 * COLUMN1

Selection criteria for unchosen columns

You can specify criteria for columns that do not appear on your report. CA OLQ tests the values in such columns when it chooses which rows to retrieve, but does not display those values on the report.

Built-in functions

CA OLQ provides built-in functions that you can use in your selection criteria. Built-in functions are a shorthand way of performing common calculations (such as square root) and string manipulations (such as concatenation).

For more information on CA OLQ built-in functions, refer to the CA OLQ Reference Section.

Step 4 -- Enter additional selection criteria

In this step, you:

Specify Y to proceed to the Additional selection criteria screen.

CA OLQ produces:

Proceed to Selection Criteria Screen? Y Y/N

At the Selection criteria screen, specify additional retrieval selection criteria that restrict what data is retrieved from the database. In this sample, your additional selection criteria limit the employees whose:

- Area code (first 3 digits) is '508'
- City of residence is not one beginning with 'New'
- Zip code begins with a '01'

```
146000 Type in selection criteria, and press the ENTER key.
```

Please Enter Additional Selection Criteria:

```
(EMP-PHONE-0415 MATCHES '508########') AND (EMP-CITY-0415 NOT LIKE 'NEW%' 
OR EMP-CITY-0415 NOT LIKE 'BOSTON') AND (EMP-ZIP-FIRST-FIVE LIKE '01###')
```

More about additional selection criteria

The **Additional selection criteria** field works like the **Selection criteria** field, enabling you to enter simple and compound expressions. In addition, you can enter:

- **Logical record keywords**
- **Criteria expressions for subscripted fields**

<table>
<thead>
<tr>
<th>Type of Expression</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple Expressions</td>
<td>COLUMN1 eq 2000 thru 5000</td>
</tr>
<tr>
<td></td>
<td>not COLUMN1 eq 2000 thru 5000</td>
</tr>
<tr>
<td>Compound Expressions</td>
<td>(COLUMN3 eq 2000) and (COLUMN4 gt 5 * COLUMN1)</td>
</tr>
<tr>
<td></td>
<td>(COLUMN3 eq 2000) or (COLUMN4 gt 5 * COLUMN1)</td>
</tr>
<tr>
<td>Logical Record Keywords</td>
<td>ACCOUNTANT and EMPLOYEE-ON-LEAVE</td>
</tr>
<tr>
<td>Subscripted Fields</td>
<td>COLUMN5(3) matches &quot;329****&quot;</td>
</tr>
<tr>
<td></td>
<td>(COLUMN1 eq (2 * COLUMN6(1))) or (COLUMN7 ne 1)</td>
</tr>
</tbody>
</table>

**Step 5 -- Retrieve the data and display your report**

In this step, you retrieve your report data from the database and display your report.
How to Save a Table From a Report

**In this section** This section shows you how to save your current report as a data table. In this example, you will:

1. Create an EMPLOYEE/OFFICE report, retrieving data from the EMPLOYEE and OFFICE records

2. Save your current report as an ASF table
3. Save your current report as an SQL table

4. View the tables you just created

In this sample, you retrieve data from database records and store it in a table format. Each column of your table corresponds to an element in one of the records you retrieved. For example, the EMP-ID-0415 column corresponds to the EMP-ID-0415 element in the EMPLOYEE record.

**Saving a Table from Your Current Report:**

Each row in the table you create corresponds to a single occurrence of a record. For example, the table row that contains data for employee ANGELO corresponds to one occurrence each of the EMPLOYEE and OFFICE records.

**Record Type:** EMPLOYEE
Save a Table Key Terms

Some terms used to discuss saving a table from a report are:

**Automatic System Facility (ASF)**

A tool in CA IDMS/DB used to create and manage tables. You can use ASF to modify the table definition, once you have created a table using CA OLQ.

**ASF Dictionary**

An application dictionary used by ASF. You must be using the ASF dictionary when you are creating data tables with CA OLQ. The application dictionary contains schema definitions, subschema definitions, user information, and application definitions (such as records, maps, dialogs, and programs).

**Current report**

The report you're working on in an active CA OLQ session. If you retrieve a saved report, CA OLQ clears out the current report.

**Table**

A presentation of data as a series of rows and columns.

- *ASF tables* refers to tables associated with the IDMSR schema.
- *SQL tables* refers to tables associated with an SQL schema.
Creating a Report 3

Contents
- Step 1 -- Select a subschema (see page 148)
- Step 2 -- Select what records you want (see page 148)
- Step 3 -- Select your columns (see page 149)
- Step 4 -- Retrieve your data (see page 149)
- Step 5 -- Display your report (see page 150)

In this step, you create an OFFICE/EMPLOYEE report. This report retrieves information from the EMPLOYEE and OFFICE records.

Step 1 -- Select a subschema

The EMPLOYEE and OFFICE records are defined in the EMPSS01 subschema for the sample employee database.

Start on the Signon Database View screen. To get there, type sub (for subschema) in the command line of any screen.

Select the EMPSS01 subschema.

```
CA OLQ Release nn.n *** Signon Database View ***
-> 121000 Select a subschema and press the ENTER key

Dictionary name . : EMPLOYEE  Database name . . :
       Dictionary node name . :
       Database node name . . :

Specify Subschema : of Schema . . . : Version :
-OR-
    Select subschema: Description:
       EMPSS01 OF EMPSCHM  VER 1
       FINAN01 OF EMPSCHM  VER 1
       SALES01 OF EMPSCHM  VER 1

1=HELP
3=QUIT
4=MESSAGE
6=MENU
PA2=REFRESH
```

Step 2 -- Select what records you want

Select the EMPLOYEE and OFFICE records.

```
CA OLQ Release nn.n *** Record Select ***
->
```

10-Jan-2018
Step 3 -- Select your columns


Step 4 -- Retrieve your data

Press [Enter] to view your report.
Step 5 -- Display your report

Your report lists data from the EMPLOYEE and OFFICE records.

```
CA OLQ Release nn.n
*** Display Report ***
->
->
125000 Press the ENTER key to go to the next page of the report.
```

OFFICE/EMPLOYEE REPORT

```
EMP-ID-0415  EMP-LAST-NAME-0415  OFFICE-CODE-0450  OFFICE-CITY-0450
-----------  ------------------  ----------------  ----------------
0120        ANGELO              002              BOSTON
0007        BANK                002              BOSTON
0069        BLOOMER             002              BOSTON
0081        FITZHUGH            002              BOSTON
0045        FONRAD              002              BOSTON
0053        GARDNER             002              BOSTON
0030        HENDON              002              BOSTON
0100        HUTTON              002              BOSTON
0158        JACKSON             002              BOSTON
0011        JENSON              002              BOSTON
0051        JOHNSON              002              BOSTON
0049        KAHALLY             002              BOSTON
```

Saving Your Report As an SQL Table

**Note:** Before beginning this step, make sure that the access switch has been set to `idms`.

For information on setting the access mode switch, see Step 1? Select the type of table.

In this step, you save your report as a SQL table. Type **table** in the command line to proceed to the Table Processing screen.

Select **Create**. Name your table `emp_office`.
CA OLQ Release nn.n

**Table Processing**

138000 Select function, table(s) and press the ENTER key

Dictname: EMPLOYEE Schema: EMPLOYEE
Function: Select Create
Add Replace

Enter table: emp_office
-or-
Select table
  EMPLOYEE.ACCOUNTING
  EMPLOYEE.BUDGET
  EMPLOYEE.DEPARTMENT
  EMPLOYEE.EMP_OFFICE

1=HELP 3=QUIT 4=MESSAGE 6=MENU 8=FWD

CA OLQ responds with a message that your table has been defined. The EMP-OFFICE table is listed.

102020 The appending, replacing, or creating of an SQL table has been successful.
102021 A total of 100 rows have been inserted into the named SQL table.

Dictname: EMPLOYEE Schema: EMPLOYEE
Function: Select Create
Add Replace

Enter table:
-or-
Select table
  EMPLOYEE.ACCOUNTING
  EMPLOYEE.BUDGET
  EMPLOYEE.DEPARTMENT
  EMPLOYEE.EMP_OFFICE

1=HELP 3=QUIT 4=MESSAGE 6=MENU 8=FWD

**Saving Your Report As an ASF Table**

⚠️ **Note:** Before beginning this step, make sure that the access switch has been set to olq.

For information on setting the access mode switch, see Step 1? Select the type of table.
Before you begin, make sure that you are signed on to the correct ASF dictionary. Ask your DBA or support staff to make sure you are signed on to the right dictionary.

In this step, you save your report as an ASF table. Type `table` in the command line to proceed to the Table Processing screen.

Select `Create`. Name your table `emp-office`.

```
CA OLQ Release nn.n
*** Table Processing ***
->
```

138000 Select function, table(s) and press the ENTER key

```
Owner: DOC1
Catalog: ASFDICT Location:
Function: _ Select X Create Delete
         _ Add        _ Replace

Enter table: emp-office
-or-
Select table
_ ACCOUNTING
_ BUDGET
_ DEPARTMENT
_ EMP-OFFICE
```

CA OLQ responds with a message that your table has been defined. The EMP-OFFICE table is listed.

```
CA OLQ Release nn.n
*** Table Processing ***
->
102017 Table processing has been successfully completed
```

```
Owner: DOC1
Catalog: ASFDICT Location:
Function: X Select _ Create _ Delete
         _ Add        _ Replace

Enter table: -or-
Select table
_ ACCOUNTING
_ BUDGET
_ DEPARTMENT
_ EMP-OFFICE
```

```
1=HELP  3=QUIT  4=MESSAGE  6=MENU  8=FWD
```
Viewing Your SQL Table

Note: Before beginning this step, make sure that the access switch has been set to idms.

For information on setting the access mode switch, see Step 1? Select the type of table.

In this step, you look at the table you just created.

Choose Select and select the EMP_OFFICE table.

Choose Select and select the EMP_OFFICE table.

Select all columns for viewing. When you select the table name, you automatically select all of the columns within that table.

Additional selection criteria:
Viewing Your ASF Table

⚠️ **Note:** Before beginning this step, make sure that the access switch has been set to olq.

For information on setting the access mode switch, see Step 1? Select the type of table.

In this step, you look at the table you just created.

Choose **Select** and select the EMP-OFFICE table.

```
CA OLQ Release nn.n
-> *** Table Processing ***
130000 Select function, table(s) and press the ENTER key

Owner: DOC1
Catalog: ASFDICT       Location:          
Function: X Select  
  _ Create
  _ Delete
  _ Add
  _ Replace

Enter table:
- or-
Select table
  _ ACCOUNTING
  _ BUDGET
  _ DEPARTMENT
  x EMP-OFFICE
```

Select all columns for viewing. When you select the table name, you automatically select all of the columns within that table.

```
CA OLQ Release nn.n
-> *** Column Select ***
124000 Select columns, specify selection criteria and press ENTER key

Columns Currently Selected: 0    Selection Criteria    Distinct N Y/N
x EMP-OFFICE
  _ 03 EMP-ID-0415
  _ 03 EMP-LAST-NAME-0415
  _ 03 OFFICE-CODE-0450
  _ 03 OFFICE-CITY-0450
```
How to Use CA OLQ in Batch Mode

What is batch processing? Batch processing is a way to create a report without requiring any interaction from the user. Once you have set up the requirements of the batch job, you can start it running and it will run until it has completed, without needing any input from you.

When do you use batch? Batch processing is used to process large amounts of data, or to perform an operation in off hours.

If you are creating a report that contains a very large number of records, the amount of time that the system spends retrieving your report's data restricts with the system's availability to other users. Thus, if you are creating a very large report, you probably want to perform the data retrieval at off hours, when the system's resources aren't at high demand.

Example

Suppose you were put in charge of creating a personnel report for the phone company. Your report retrieves 30,000 records, one record for each employee. Since you are creating this report for the first time, you will probably want to change it once you see how it looks.

Because your report contains so many records, you don't want to retrieve all of them until you are sure that you have the report exactly the way you want it. You can create a report prototype to make sure that you are satisfied with the report before you retrieve the data.

Using batch processing

The step-by-step solution for the personnel report would be to:

1. Set your interrupt count to a very low level (for example, 20 records)

2. Create the report, but limit it to 20 records

3. Save the report definition as a qfile

4. Execute the qfile in off hours using batch processing

Processing a Qfile in Batch:
Set the interrupt count to 20 records

Create a current report

Save it as a qfile

Execute the qfile using batch processing
OLQ Batch Mode Key Terms

Some terms used to discuss batch processing are:

**Batch**

Batch processing means that the user doesn’t have to interact with the computer system in order to perform a function. Usually, a batch job is set up in advance (such as when you fill out your Batch Processing screen). Once the job has started running, you cannot intervene except to cancel execution.

**Input file**

A file that contains input for a batch job.

**Interactive**

A way of performing a function in which the computer system requires the user to provide input, and then responds to that input. An example would be CA OLQ menu mode. Another term to describe interactive processing is *online*.

**Interrupt count**

The maximum number of records CA OLQ will retrieve when building a report. If the number of records that meet the selection criteria for that report exceeds the interrupt count, CA OLQ suspends data retrieval and issues a message asking you if you want to continue to retrieve records.

**Job Control Language**

A language used to define the special requirements of your batch program to the system. Job Control Language (JCL) statements name input and output files, the name of your program, and your output destination.

**Output destination**

Any type of device that receives the report that you have created through your batch job. Output destinations can be a printer, a terminal, or a log.

**Output file**

A file that contains the results of your batch job.

Batch Process to Create a Report

To create a report using batch, you do the following:

**Steps to Create a Report using Batch:**
## What To Do

<table>
<thead>
<tr>
<th>Step</th>
<th>What To Do</th>
<th>How To Do It</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Specify which report you want to execute batch.</td>
<td>Build it from a table or records.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Execute a qfile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Retrieve a saved report.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Specify your processing commands on the Batch Processing screen.</td>
</tr>
<tr>
<td>2</td>
<td>Move to the Batch Processing screen.</td>
<td>Type BATCH in the command line.</td>
</tr>
<tr>
<td>3</td>
<td>Specify your batch processing options.</td>
<td>See Table 12-2</td>
</tr>
<tr>
<td>4</td>
<td>Initiate batch processing.</td>
<td>Press [Enter].</td>
</tr>
</tbody>
</table>

### The Batch Processing screen

```
CA OLQ Release nn.n
->  *** Batch Processing ***
132000 Select activity, output selection and press the ENTER key

Password: Job control module: OLQBATCH-JCL-RMG
Select activity:
  _ Submit current report request and comments
  X Submit CA OLQ syntax
To automatically generate output syntax:
  X Display (Output to SYSLST)
  _ Print (DC printer) Class: Dest:
  _ Save report Name :

Enter CA OLQ Syntax/Comments:
> QFIL 'SALARY '(1) DICT TSTDICT DICTNO ' '
> >
> >
> >
> >
> >
> 1=HELP 3=QUIT 4=MESSAGE 6=MENU
```

You can accept the default values on the Batch Processing screen or type over them. You’ll probably need to ask your DBA how to fill out these fields.

When you press [Enter], your batch job begins processing.

Batch Processing screen fields are explained in the following.

### Batch Processing Screen Fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password:</td>
<td>Provides security for saved reports and qfile definitions. Usually, you want to enter your user password (it will not show on the screen).</td>
</tr>
<tr>
<td>Job control module:</td>
<td>Names a module that contains Job control language (JCL) statements that tell the system how to run your batch job. Your DBA defines Job Control Language modules for your site.</td>
</tr>
<tr>
<td>Select activity:</td>
<td>Tells CA OLQ what is to be processed in batch mode. You can: Execute the current report in batch mode (Submit current report request). If you access the Batch Processing screen from the Saved Reports screen, this field is the</td>
</tr>
</tbody>
</table>
How to Print Your Report

Some terms used to discuss printing reports are:

**Current report**

The report you are working on in an active CA OLQ session. If you retrieve a saved report or execute a qfile, CA OLQ clears out the current report.

**Destination**

When you print a report, you specify an output destination where the report is to be printed. Usually, the destination is a file associated with a printer.

For more information, see the following topics:
- Printing a Report (see page 159)

Printing a Report

To print a report, do the following:

**Steps for Printing a Report:**

<table>
<thead>
<tr>
<th>Step</th>
<th>What To Do</th>
<th>How To Do It</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Create a current report.</td>
<td>Build it from a table or records.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Execute a qfile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Retrieve a saved report.</td>
</tr>
<tr>
<td>2</td>
<td>Move to the Print Processing screen.</td>
<td>Type PRINT in the command line.</td>
</tr>
<tr>
<td>3</td>
<td>Specify your print options.</td>
<td>See Table 13-2</td>
</tr>
<tr>
<td>4</td>
<td>Initiate print processing.</td>
<td>Press [Enter].</td>
</tr>
</tbody>
</table>
When you get to the Print Processing screen, the fields are filled out with default values. You can accept these default values or type over them. When you press [Enter], your job is routed to a printer.

Print current report: Y (Y-Yes/N-No)
Line size . . . : 80
Line count . . . : 60
Number of copies: 1
From page: 1 for: ALL pages

Enter destination: or class:
- or-
Select destination
  DEST01
  DEST02
  DEST03

Print Processing screen fields are explained in table in the section "Batch Processing".

**Print Processing Screen Fields:**

<table>
<thead>
<tr>
<th>What to Specify</th>
<th>Which Field to Use</th>
<th>Restrictions</th>
<th>Defaults</th>
</tr>
</thead>
<tbody>
<tr>
<td>Whether you want to print the report</td>
<td>Print current report:</td>
<td>Specify Yes or No</td>
<td>No</td>
</tr>
<tr>
<td>How many characters per print line</td>
<td>Line size:</td>
<td>1 through 9999</td>
<td>80</td>
</tr>
<tr>
<td>The number of lines per page</td>
<td>Line count:</td>
<td>1 through 9999</td>
<td>60</td>
</tr>
<tr>
<td>How many copies of the report you want to print</td>
<td>Number of copies:</td>
<td>1 to 256</td>
<td>1</td>
</tr>
<tr>
<td>The number of the first page you want to print</td>
<td>From page:</td>
<td>Any page number in the report</td>
<td>Page 1</td>
</tr>
<tr>
<td>How many report pages you want to print</td>
<td>for:</td>
<td>The number of pages in the report</td>
<td>All of the report</td>
</tr>
<tr>
<td>The printer used to print the report</td>
<td>Enter destination:</td>
<td>A printer destination defined by your DBA</td>
<td>Site specific. Ask your support staff which printer is set up as the default printer.</td>
</tr>
</tbody>
</table>
Introducing the OLQ SELECT statement

What this section presents
This section introduces the OLQ SELECT statement, which you can use to retrieve information in a CA IDMS/DB database. It also contains some tips for using the command mode and reporting features of CA OLQ.

When to use OLQ SELECT

Use the OLQ SELECT statement, to access:

- ASF tables
- Database records
- Logical records
- Sequential files (batch only)

Important! The examples of the OLQ SELECT statement in this section are valid when the access switch is set to olq.

For more information, see the following topics:
- The OLQ SELECT Statement (see page 161)
- Defining Your Data (see page 163)
- Some Tips On Using CA OLQ (see page 163)

The CA IDMS/DB SQL SELECT statement is used when the access switch is set to sql. For more information, see the CA IDMS SQL Reference Section.

The OLQ SELECT Statement

When to use the SELECT statement

CA OLQ uses the OLQ SELECT statement to access information from ASF tables, logical records, and database records.

Generate queries

By using the SELECT statement, you can formulate both simple and complex queries for information. CA OLQ interprets the SELECT statement and produces a report.

SELECT statement syntax
Depending on your request, use some or all of these clauses in the sequence listed:

<table>
<thead>
<tr>
<th>Clause</th>
<th>Information you supply</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT</td>
<td>Columns or fields to display.</td>
</tr>
<tr>
<td>FROM</td>
<td>The source of information (that is, table or record names).</td>
</tr>
<tr>
<td>WHERE</td>
<td>Selection criteria applied to rows or record occurrences.</td>
</tr>
<tr>
<td>GROUP</td>
<td>Column or field names for grouping information with the same value (for example, all rows with the same department ID).</td>
</tr>
<tr>
<td>HAVING</td>
<td>Selection criteria applied to grouped information (for example, a summary row where the average salary is less than $35,000).</td>
</tr>
<tr>
<td>UNION</td>
<td>Multiple SELECT statements. UNION combines the rows retrieved from each SELECT statement.</td>
</tr>
<tr>
<td>ORDER</td>
<td>Column or field names on which to sort retrieved information.</td>
</tr>
</tbody>
</table>

Reading a SELECT statement

Read a SELECT statement like an English sentence. The order of the clauses falls into place. For example, to get all employee IDs less than 100 displayed in ascending order, you would:

- **SELECT** the employee ID
- **FROM** the EMP table
- **WHERE** the employee ID is less than 100
- **ORDER** the display **BY** employee ID

Here is the complete SELECT statement:

```
select empid from emp where empid < 100 order by empid
```

Some examples

These examples show how the SELECT statement is used to retrieve information from a database:

- List all information on corporate departments:
  ```
  select * from dept
  ```

- List the names and phone numbers of employees in department 4000 sorted by the employee's last name:
  ```
  select lastname, firstname, phone from emp where deptid = 4000 order by lastname
  ```

- Display the date Themis Papazeus was hired:
  ```
  select startdate from emp where firstname='&xq.themis' and lastname='&xq.papazeus'
  ```

- List the number of employees in department 4000:
  ```
  select count(*) from emp where deptid = 4000
  ```
How many employees in each department were hired before January 1, 1975?

```
select deptid, count(*) from emp where startdate < 750101 group by deptid
```

Defining Your Data

Creating sample tables

If you want to use the SELECT statement as you proceed through this section, you need two or more ASF-generated tables.

To create an ASF-generated table, see the CA IDMS ASF User Section.

Sample tables used in this section

The sample tables used in this section are ASF-generated tables created in part from data stored in the Employee database your site receives at installation. The sample tables, which appear in Appendix A, "Sample Tables and Database", are:

<table>
<thead>
<tr>
<th>The table</th>
<th>Contains information on</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMP</td>
<td>Employees in departments 3100, 4000, and 6666</td>
</tr>
<tr>
<td>DEPT</td>
<td>All corporate departments</td>
</tr>
<tr>
<td>JOBLIST</td>
<td>Position held by each corporate employee</td>
</tr>
<tr>
<td>JOBCLASS</td>
<td>Class and associated salary range assigned to a position</td>
</tr>
<tr>
<td>BOSTON</td>
<td>Employees in the Boston office</td>
</tr>
<tr>
<td>WESTON</td>
<td>Employees in the Weston office</td>
</tr>
<tr>
<td>SPRINGFIELD</td>
<td>Employees in the Springfield office</td>
</tr>
</tbody>
</table>

Sample database

For the most part, examples in this section are based on the sample tables listed above. Examples that retrieve information from network records use the Employee database. The data structure diagram for the Employee database appears in Appendix A.

Some Tips On Using CA OLQ

Contents

- Signing on (see page 164)
- Signing off (see page 165)
- Entering commands (see page 165)
- Setting the access to IDMS (see page 167)
- Tailoring reports (see page 167)

This section gives you some tips on using CA OLQ:
CA IDMS - 19.0

- Signing on
- Signing off
- Entering statements
- Setting the access to IDMS
- Tailoring your reports

Signing on

To sign on to CA OLQ follow these steps:

Step 1 -- Sign on to a CA IDMS/DC or CA IDMS UCF system

Access your CA IDMS/DC or CA IDMS UCF system and enter your signon id and password at the ENTER NEXT TASK CODE prompt:

V85  REL nn.n ENTER NEXT TASK CODE:
    signon cub scout

Step 2 -- Specify a dictionary

Specify the data dictionary that contains the table or subschema definitions describing the data you want. You may be assigned to a dictionary when you sign on depending on the contents of your signon profile. You can change your signon dictionary or specify a dictionary by issuing a DCUF command to set the dictionary name:

IDMS DC258003 V85 USER CUB SIGNED ON LTERM VL85008 AT 11:57:35.75 88.040
V85  REL nn.n ENTER NEXT TASK CODE:
    dcuf set dictname asfdict

Note: You can also specify a dictionary in the SIGNON statement.

Step 3 -- Enter CA OLQ's task code

Issue your site's task code for CA OLQ at the ENTER NEXT TASK CODE command. The task code provided at installation is OLQ:

SET DICTNAME ASFDICT
IDMS DC402009 V85 DICTNAME ASFDICT HAS BEEN SET
V85  REL nn.n ENTER NEXT TASK CODE:
    olq

You are now signed on to CA OLQ.

Step 4 -- Set your default access to OLQ

The examples of the SELECT statement in this section are only valid when your default access is set to OLQ. You can do this for your CA OLQ session by specifying:
set access olqOLQ 092033 00  Processing mode changed to OLQ.

Signing off

To end your DC/UCF session, follow these steps:

**Step 1 -- Sign off CA OLQ**

Sign off CA OLQ by entering BYE on the command line:

```
bye
```

OLQ 091057 00  Please enter next command.

CA OLQ terminates the CA OLQ session and returns you to DC/UCF.

**Step 2 -- End your session**

End your session by entering BYE at the ENTER NEXT TASK CODE prompt:

```
OLQ 100029 00  Signoff accepted - OLQ session terminated
```

V401  ENTER NEXT TASK CODE:

```
bye
```

Entering commands

**Switching between menu/command mode**

If you have signed on to CA OLQ and you see a screen of menu options, you are in CA OLQ's menu facility. Press [PF9] to swap to CA OLQ's command mode screen, which looks like this:

```
OLQ 107017 00  CA OLQ Release nn,n
OLQ 017019 Copyright(C) 2003 CA,Inc.
OLQ 091057 00  Please enter next command.
```

**Command area of the screen**

Type CA OLQ commands in the space above the CA OLQ messages you see on the screen above. You may notice that your terminal shows more or less space in the command area than you see in the above example. The size of the command portion is established at system generation by the INPUT LINE SIZE clause of the OLQ system generation statement. If the command portion of your screen is too small, ask your system administrator to adjust the value assigned to the INPUT LINE SIZE clause.

**How to enter commands**

Enter CA OLQ commands starting at the upper left corner of the screen. Depending on your preference, enter commands in lower case letters, upper case letters, or both. When CA OLQ interprets a command, it changes the display to upper case letters.
Note: By issuing a DCUF SET UPLow command before signing on to CA OLQ, CA OLQ recognizes case in character strings and displays processed commands and headings as you entered them.

Separate each word in a command by at least one blank. At a 3270-type terminal, press &retsym. to continue a command on the next line. Press [Enter] to process the command. The processed command appears on the screen. You can edit the command and resubmit it to CA OLQ.

Some useful commands

The SELECT statement is the focus of this section. However, you'll need some of the commands listed below to facilitate your CA OLQ session:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIGNON</td>
<td>Initiates a CA OLQ session by defining a view of the database</td>
<td>signon ss empss01</td>
</tr>
<tr>
<td>HELP</td>
<td>Explains how to use CA OLQ commands and displays information on your tables and subschema records</td>
<td>help table</td>
</tr>
<tr>
<td>SET</td>
<td>Sets session parameters in a signon profile, such as ACCESS</td>
<td>set access olq</td>
</tr>
<tr>
<td>OPTIONS</td>
<td>Sets processing and display options for the session</td>
<td>options = olqheader</td>
</tr>
<tr>
<td>DISPLAY</td>
<td>Displays a report</td>
<td>display</td>
</tr>
<tr>
<td>BYE</td>
<td>Ends a CA OLQ session</td>
<td>bye</td>
</tr>
</tbody>
</table>

Some of these commands are described below. For a complete description, see the CA OLQ Reference Section.

Command delimiters

CA OLQ provides a separator character, which lets you string many commands together. The system default is an exclamation mark (!). All the examples in this section show an exclamation mark to separate the SELECT statement from a DISPLAY command, which displays the report: select * from dept ! display.

CA OLQ also provides a comment character, which tells CA OLQ treat all text following the delimiter as a comment. The system default is a semi-colon (;). By using a comment character, you can save portions of a command that you want to edit and resubmit. This example shows a DISPLAY command typed over a SELECT statement CA OLQ has already processed:

```
SELECT * FROM DEPT
display; FROM DEPT
```

To change the separator and comment characters for your session, use the SET command:

```
SET SEPARATOR CHARACTER '\$'
```

```
OLQ 092014 00 The SEPARATOR CHARACTER has been modified.
```

Signing on to a table or subschema
Issue a SIGNON command to tell CA OLQ what subschema to access for data. A SIGNON command is required for CA IDMS/DB subschemas. It is optional for tables.

To sign on to a specific table or subschema enter a SIGNON command: `signon table emp`.

CA OLQ responds that it is ready to retrieve data:

```
SIGNON TABLE EMP
```

```
CA OLQ 100021 00 Ready to retrieve data from subschema RU000371
CA OLQ 100022 00 Sc
hema: CA-IDMSDB Version: 1
CA OLQ 100023 00 Database name: ASFDICT
CA OLQ 100025 00 Dictionary name: ASFDICT
```

**Using the SELECT statement**

You can use CA OLQ command mode to create your own SELECT statements, or you can let CA OLQ's menu facility build the statements for you when you select data. All the examples in this section use command mode to enter the SELECT statement.

**Setting the access to IDMS**

When you want to access SQL-defined tables, your default access setting must be set to IDMS. You can do this for your CA OLQ session by specifying:

```
set access idms
```

You are now able to use the SELECT statement to access SQL-defined tables.

⚠️ **Note:** The examples of the SELECT statement in this section are not necessarily valid for SQL-defined tables.

For more information on using the SELECT statement with SQL-defined tables, refer to the *CA IDMS SQL Reference Section*.

**Tailoring reports**

**Tailoring your report**

The screen displays in this section show the direct results of the SELECT statement. The displays reflect default formatting options. For example, reports have the name of the source table as a title, fields aren't edited, and groups of data are not separated by blank lines.

You can easily tailor your reports by using options available in CA OLQ's menu facility.

The following series of screens show a few of these options:

**Display the report**
In command mode, issue a SELECT statement to CA OLQ:

```
select deptid, lastname, salary, phone from emp order by deptid, lastname ! display
```

```
SELECT DEPTID, LASTNAME, SALARY, PHONE FROM EMP 
ORDER BY DEPTID, LASTNAME ! DISPLAY
```

### EMP REPORT

\[\text{mm/dd/yy}\]

<table>
<thead>
<tr>
<th>DEPTID</th>
<th>LASTNAME</th>
<th>SALARY</th>
<th>PHONE</th>
</tr>
</thead>
<tbody>
<tr>
<td>3100</td>
<td>DOUGH</td>
<td>33000.00</td>
<td>6174458155</td>
</tr>
<tr>
<td>3100</td>
<td>GALLWAY</td>
<td>33000.00</td>
<td>6173349155</td>
</tr>
<tr>
<td>3100</td>
<td>GARFIELD</td>
<td>65000.00</td>
<td>6173321967</td>
</tr>
<tr>
<td>3100</td>
<td>GRANGER</td>
<td>34500.00</td>
<td>6173341212</td>
</tr>
<tr>
<td>3100</td>
<td>HEAROWITZ</td>
<td>33000.00</td>
<td>6173349634</td>
</tr>
<tr>
<td>3100</td>
<td>JACOBI</td>
<td>55000.00</td>
<td>6173348912</td>
</tr>
<tr>
<td>3100</td>
<td>JENSEN</td>
<td>37000.00</td>
<td>6172241955</td>
</tr>
<tr>
<td>3100</td>
<td>LITERATA</td>
<td>37500.00</td>
<td>6175912323</td>
</tr>
<tr>
<td>3100</td>
<td>TYRO</td>
<td>20000.00</td>
<td>6174459191</td>
</tr>
<tr>
<td>4000</td>
<td>ANGELO</td>
<td>18000.00</td>
<td>6178870235</td>
</tr>
<tr>
<td>4000</td>
<td>BANK</td>
<td>80000.00</td>
<td>6173321933</td>
</tr>
<tr>
<td>4000</td>
<td>JACKSON</td>
<td>34000.00</td>
<td>6175346767</td>
</tr>
<tr>
<td>4000</td>
<td>MCD OUGALL</td>
<td>18000.00</td>
<td>6178871324</td>
</tr>
<tr>
<td>4000</td>
<td>PENMAN</td>
<td>39000.00</td>
<td>6175341199</td>
</tr>
</tbody>
</table>

- 1 -

Go to the menu facility

Swap to CA OLQ's menu facility by pressing [PF9].

⚠️ **Note:** You can also move directly to a screen by entering the name of the screen on the command line.

CA OLQ displays the Menu screen:

```
CA OLQ Release nn.n

*** Menu ***

Select an option and press the ENTER key.

--- Data Source for Report ---

Choose data tables
Choose subschema

--- Retrieval Activity ---

Choose records from selected subschema
Choose columns for report
Retrieve data to build report
Alter database access strategy

--- Processing Mode ---

Execute or create a predefined routine
View existing or save current report
Submit batch report request

1=HELP 2=GLOBAL HELP 3=QUIT 4=MESSAGE 8=FWD
```

Change the column headings
On page 2 of the Menu screen, select the option to change column headers. CA OLQ brings you to the Report Format - Header screen. You can assign new column headings for your report and specify a character to underline the column headings. The default character is a hyphen (-).

```
> CA OLQ Release nn.n  *** Report Format - Header ***
134000 Specify column headers and press the ENTER key.
```

Underline character: -

<table>
<thead>
<tr>
<th>EMP</th>
<th>DEPTID</th>
<th>1 department id</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LASTNAME</td>
<td>employee name</td>
</tr>
<tr>
<td></td>
<td>PHONE</td>
<td>PHONE</td>
</tr>
<tr>
<td></td>
<td>SALARY</td>
<td>SALARY</td>
</tr>
</tbody>
</table>

When you press [Enter], CA OLQ displays a revised report:

```
> CA OLQ Release nn.n  *** Display Report ***
125000 Press the ENTER key to go to the next page of the report.
```

```
DEPARTMENT EMPLOYEE
ID NAME SALARY PHONE
3100 DOUGH 33000.00 6174458155
3100 GALLWAY 33000.00 6173349155
3100 GARFIELD 65000.00 6173321967
3100 GRANGER 34500.00 6173341212
```

Change the report title

Press [PF6] to return to the menu options. Choose the option to change page headers and footers. CA OLQ brings you to the Page Header/Footer screen, where you can give the report a meaningful title:

```
> CA OLQ Release nn.n  *** Page Header/Footer ***
152000 Specify page header(s), footer(s) and press the ENTER key.
```

Format for $DATE: MM/DD/YY
Use variables: $DATE, $TIME, $PAGE, $LINE, $USER...
Skip lines before heading: 0  Skip lines after heading: 1

<table>
<thead>
<tr>
<th>Line</th>
<th>Page heading text</th>
<th>Align</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>employee report by department</td>
<td>ENTER</td>
</tr>
<tr>
<td>2</td>
<td>$DATE</td>
<td>CENTER</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>CENTER</td>
</tr>
</tbody>
</table>

Once you have entered the information, press [Enter]. CA OLQ displays this report:

```
> CA OLQ Release nn.n  *** Display Report ***
125000 Press the ENTER key to go to the next page of the report.
```

```
DEPARTMENT ID EMPLOYEE NAME SALARY PHONE
3100 DOUGH 33000.00 6174458155
3100 GALLWAY 33000.00 6173349155
3100 GARFIELD 65000.00 6173321967
3100 GRANGER 34500.00 6173341212
```
Format the phone and salary data

Return to the Menu screen by pressing [PF6] and choose the option to change the display format of data. CA OLQ brings you to the Report Format - Picture screen where you can:

- Edit the phone number display
- Select options to display the SALARY field with a dollar sign and commas

```
CA OLQ Release nn.n *** Report Format - Picture ***
137000 Specify pictures and press the ENTER key.
```

CA OLQ displays this report once it processes the information entered on the Report Format - Picture screen:

```
CA OLQ Release nn.n *** Display Report ***
125000 Press the ENTER key to go to the next page of the report.
```

Group the rows by department

Press [PF6] to return to the Menu screen and choose the option to sort the sequence of the report. CA OLQ brings you to the Report Format - Sort screen. On this screen, you can indicate:

- The sort priority for the fields (which isn't really necessary because the ORDER BY clause of the SELECT statement sorted the report's contents)
- The field on which to group data for summary calculations. In this case, the DEPTID field is assigned Group Level # 1.
- Whether to display detail lines, summary lines (subtotals), or both. In this screen, both types of lines are chosen.

```
CA OLQ Release nn.n *** Report Format - Sort ***
133000 Specify sort or group by request and press the ENTER key.
```
Once CA OLQ processes the information provided above, it brings you to the Report Format - Group By screen where you can enter summary information for the report. In this example, the report will display the total salaries for each department. The SKIP LINES and SEPARATOR CHARACTER fields indicate that CA OLQ will:

- Insert one line between the detail and subtotal lines
- Write a separator line (composed of hyphens) between the detail and subtotal lines

When you press [Enter], CA OLQ displays this report:

---

**CA OLQ Release nn.n***  Display Report  ***

**Page 1**

105022 Sort successfully completed. 19 records in. 19 records out.

125000 Press the ENTER key to go to the next page of the report.

**EMPLOYEE** REPORT **BY** DEPARTMENT

<table>
<thead>
<tr>
<th>mm/dd/yy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Department</th>
<th>Employee</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

10-Jan-2018   171/222
The end result

The report shown above contains the same data as the original report. However, this report is easier to read and contains summary information for each department.

Retrieving Information from a Table

Retrieving All Columns

Use an asterisk (*) To retrieve all of the columns defined for a table, you can use an asterisk (*) in the SELECT statement. An asterisk is shorthand for all of the table's columns. By using an asterisk, you don't need to know the table's column names in order to retrieve data.

Example

To display the DEPT table, enter:

\[
\text{select * from dept ! display}
\]

If all of the columns or all of the rows of the table don't fit on one screen, use these PF keys to scroll:

<table>
<thead>
<tr>
<th>PF7</th>
<th>Scroll backward</th>
</tr>
</thead>
<tbody>
<tr>
<td>PF8</td>
<td>Scroll forward</td>
</tr>
<tr>
<td>PF10</td>
<td>Scroll left</td>
</tr>
</tbody>
</table>
Retrieving Selected Columns

Name the columns

To retrieve some of the columns from your table, list the columns you want in the SELECT statement, using a comma to separate each name. CA OLQ displays the columns in the order you enter them. Each column name becomes the column header in the report.

**Note:** To determine the column names used in the sample tables, look in Appendix A or retrieve all of the table's columns by using an asterisk in the SELECT statement.

Example

List each department's name and manager ID:

```
select deptname, mgrid from dept ! display
```

<table>
<thead>
<tr>
<th>DEPTNAME</th>
<th>MGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERSONNEL</td>
<td>0013</td>
</tr>
<tr>
<td>ACCOUNTING AND PAYROLL</td>
<td>0011</td>
</tr>
<tr>
<td>INTERNAL SOFTWARE</td>
<td>0003</td>
</tr>
<tr>
<td>COMPUTER OPERATIONS</td>
<td>0004</td>
</tr>
<tr>
<td>PUBLIC RELATIONS</td>
<td>0007</td>
</tr>
<tr>
<td>BRAINSTORMING</td>
<td>0015</td>
</tr>
<tr>
<td>THERMOREGULATION</td>
<td>0349</td>
</tr>
<tr>
<td>BLUE SKIES</td>
<td>0321</td>
</tr>
<tr>
<td>EXECUTIVE ADMINISTRATION</td>
<td>0030</td>
</tr>
</tbody>
</table>

Eliminating Duplicate Data

Use DISTINCT

A table can have one or more columns with duplicate data entries. For example, the JOBLIST table lists COMPUTER OPERATOR three times because three employees have that title. To retrieve unique, rather than duplicate, values in a column, use the keyword DISTINCT in the SELECT statement.

Example

List unique job titles in the company:
select distinct title from joblist ! display

JOBLIST REPORT
   mm/dd/yy
   TITLE
   ACCOUNTANT
   AP CLERK
   AR CLERK
   COMPUTER OPERATOR
   CUMULUS CARETAKER
   DATABASE ADMIN.
   DIR CORP CONFUSION
   DIR OPERATIONS
   DIR WEATHER
   DOCUMENTATION SPEC
   FINANCIAL ANALYST
   HUMIDITY CONTROL CLK
   ILLUSTRATOR

Displaying Calculations in Columns

Compute new values

The SELECT statement displays calculated values, in addition to values stored in the table. For example, you might want to calculate a 6% bonus for each employee.

To display a calculated column, include an arithmetic expression in the column list following the SELECT statement. An arithmetic expression uses these operators:

<table>
<thead>
<tr>
<th>Operator</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>Addition</td>
</tr>
<tr>
<td>-</td>
<td>Subtraction</td>
</tr>
<tr>
<td>*</td>
<td>Multiplication</td>
</tr>
<tr>
<td>/</td>
<td>Division</td>
</tr>
</tbody>
</table>

You can also use a built-in function to display a calculated column. Built-in functions are described later in this section.

Some examples

A few examples of arithmetic expressions appear below. These examples show you can:

- Leave spaces, or omit spaces, before and after these arithmetic operators: *, +, /. You must include a blank space before and after a minus (-) sign.
- Use parentheses to show how the arithmetic expression should be evaluated and to improve readability.
Provide a heading for a calculation

To give each calculated column a heading, use the keyword AS and a heading name following the arithmetic expression. Enclose headings with two or more words in single quotation marks.

Example

List each employee's salary and end-of-year bonus, based on 6% of salary:

```sql
select empid, salary as 'annual salary', salary * 0.06 as bonus
from emp
display
```

<table>
<thead>
<tr>
<th>EMPID</th>
<th>ANNUAL SALARY</th>
<th>BONUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>76000.00</td>
<td>4560.0000</td>
</tr>
<tr>
<td>0003</td>
<td>65000.00</td>
<td>3900.0000</td>
</tr>
<tr>
<td>0007</td>
<td>80000.00</td>
<td>4800.0000</td>
</tr>
<tr>
<td>0019</td>
<td>37000.00</td>
<td>2220.0000</td>
</tr>
<tr>
<td>0020</td>
<td>55000.00</td>
<td>3300.0000</td>
</tr>
<tr>
<td>0021</td>
<td>20000.00</td>
<td>1200.0000</td>
</tr>
<tr>
<td>0024</td>
<td>33000.00</td>
<td>1980.0000</td>
</tr>
<tr>
<td>0027</td>
<td>33000.00</td>
<td>1980.0000</td>
</tr>
<tr>
<td>0028</td>
<td>34500.00</td>
<td>2070.0000</td>
</tr>
<tr>
<td>0029</td>
<td>33000.00</td>
<td>1980.0000</td>
</tr>
<tr>
<td>0030</td>
<td>240000.00</td>
<td>14400.0000</td>
</tr>
<tr>
<td>0035</td>
<td>37500.00</td>
<td>2250.0000</td>
</tr>
<tr>
<td>0120</td>
<td>18000.00</td>
<td>1080.0000</td>
</tr>
<tr>
<td>0127</td>
<td>18000.00</td>
<td>1080.0000</td>
</tr>
</tbody>
</table>

Putting Rows in Order

Use ORDER BY

To sort selected rows by the values in a column, use the ORDER BY clause. CA OLQ assumes you want rows in ascending order. If you want rows in descending order, specify DESCENDING as Example 2 shows below.

CA OLQ sorts selected rows by the first column named in the ORDER BY clause. It then sorts each group of rows sharing a common value in order of the second column named in the ORDER BY clause, and so on. For example, you might want to display a table of bank transactions in order of branch number, and within each branch number, in order of transaction date.

Example 1 -- Sorting on 1 column

For each employee in department 4000, list the ID, name, and hire date starting with the first person hired to the last person hired:
```sql
select empid, lastname, firstname, startdate as 'hire date' from emp where deptid = 4000 order by startdate ! display
```

### EMP REPORT

<table>
<thead>
<tr>
<th>EMPID</th>
<th>LASTNAME</th>
<th>FIRSTNAME</th>
<th>HIRE DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0476</td>
<td>ZEDI</td>
<td>BETSY</td>
<td>760223</td>
</tr>
<tr>
<td>0158</td>
<td>JACKSON</td>
<td>JOCK</td>
<td>770707</td>
</tr>
<tr>
<td>0149</td>
<td>PENMAN</td>
<td>LAURA</td>
<td>770908</td>
</tr>
<tr>
<td>0007</td>
<td>BANK</td>
<td>MONTE</td>
<td>780430</td>
</tr>
<tr>
<td>0120</td>
<td>ANGELO</td>
<td>MICHAEL</td>
<td>790908</td>
</tr>
<tr>
<td>0127</td>
<td>MCDougall</td>
<td>CAROL</td>
<td>800607</td>
</tr>
</tbody>
</table>

Example 2 -- Sorting on 2 columns

List the department ID, employee name, and hire date of all employees sorted by department. Within each department, list the employees in alphabetic descending order:

```sql
select deptid, lastname, startdate as 'hire date' from emp order by deptid, lastname descending ! display
```

### EMP REPORT

<table>
<thead>
<tr>
<th>DEPTID</th>
<th>LASTNAME</th>
<th>HIRE DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>3100</td>
<td>TYRO</td>
<td>801221</td>
</tr>
<tr>
<td>3100</td>
<td>LITERATA</td>
<td>800909</td>
</tr>
<tr>
<td>3100</td>
<td>JENSEN</td>
<td>820929</td>
</tr>
<tr>
<td>3100</td>
<td>JACOBI</td>
<td>811111</td>
</tr>
<tr>
<td>3100</td>
<td>HEAROWITZ</td>
<td>810909</td>
</tr>
<tr>
<td>3100</td>
<td>GRANGER</td>
<td>800527</td>
</tr>
<tr>
<td>3100</td>
<td>GALLWAY</td>
<td>811010</td>
</tr>
<tr>
<td>3100</td>
<td>DOUGH</td>
<td>760808</td>
</tr>
<tr>
<td>4000</td>
<td>ZEDI</td>
<td>760223</td>
</tr>
<tr>
<td>4000</td>
<td>PENMAN</td>
<td>770908</td>
</tr>
<tr>
<td>4000</td>
<td>MCDougall</td>
<td>800607</td>
</tr>
<tr>
<td>4000</td>
<td>JACKSON</td>
<td>770707</td>
</tr>
<tr>
<td>4000</td>
<td>BANK</td>
<td>780430</td>
</tr>
</tbody>
</table>

- 1 -

### Sorting on calculated columns

If you want to sort the contents of your report based on values in a calculated column, specify the column number in the ORDER BY clause. Count the columns from left to right, beginning with 1.

Example

List salaries and anticipated 5% year-end bonus for employees in department 4000. List rows in order of smallest to largest bonus:

```sql
select lastname, salary, &xq.year end bonus=', salary*0.005 from emp where deptid=4000 order by 4 ! display
```

### EMP REPORT

<table>
<thead>
<tr>
<th>LASTNAME</th>
<th>SALARY</th>
<th>YEAR END BONUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANGELO</td>
<td>18000.00</td>
<td>90.000000</td>
</tr>
<tr>
<td>MCDougall</td>
<td>18000.00</td>
<td>90.000000</td>
</tr>
</tbody>
</table>
Retrieving Selected Rows

Contents
- A simple comparison (see page 178)
- Complex comparisons (see page 179)
- Comparisons to a list of values (see page 181)
- Exclusive comparisons (see page 181)
- Range comparisons (see page 182)
- Character string comparisons (see page 183)

Use WHERE

You can retrieve selected rows from a table by specifying selection criteria in the WHERE clause. A WHERE clause contains one or more comparison expressions. A comparison expression compares one value to another value. The simplest comparison expression compares column values to a constant. If the comparison expression is true, CA OLQ selects the row for the report.

Within the WHERE clause, you can select rows by using these symbols and keywords:

<table>
<thead>
<tr>
<th>Expression</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison operators (for example, = and &lt;=)</td>
<td>Compares each value in a column to another value.</td>
</tr>
<tr>
<td>AND and OR</td>
<td>Connects comparison expressions. AND is true if all the comparisons are true. OR is true if any of the comparisons is true.</td>
</tr>
<tr>
<td>IN</td>
<td>Compares a column's values to a list of values. The expression is true if the value in the column equals one of the listed values.</td>
</tr>
<tr>
<td>NOT</td>
<td>Negates the comparison expression. That is, if the comparison expression is true, NOT returns a value of false so the row is not selected.</td>
</tr>
<tr>
<td>BETWEEN</td>
<td>Compares each value in a column to a specified range, including the starting and ending range values.</td>
</tr>
<tr>
<td>LIKE</td>
<td>Compares a character string to a mask (that is, pattern).</td>
</tr>
<tr>
<td>Arithmetic expression</td>
<td>Compares a value in a column to the result of an arithmetic expression.</td>
</tr>
<tr>
<td>Built-in function</td>
<td>Compares each value in a column to a value calculated by a predefined function.</td>
</tr>
</tbody>
</table>

CA OLQ does not support NULL, an ANSI-standard keyword, for this release.
A simple comparison

Defining a comparison expression

As stated above, the simplest comparison expression compares each value in a column to a constant. The constant can be either:

- Numeric (that is, decimal, integer, real, hexadecimal, binary, or multibit binary).
- Character. A character constant is enclosed in single quotation marks.

The symbols that compare one value to the other are:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>=</td>
<td>Equal to</td>
</tr>
<tr>
<td>&lt;&gt; or ¬=</td>
<td>Not equal to</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater than</td>
</tr>
<tr>
<td>&lt;</td>
<td>Less than</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Less than or equal to</td>
</tr>
</tbody>
</table>

If you prefer, you can also use these symbols: EQ, NE, GT, LT, GE, and LE.

Example 1 -- Comparing a number

List all the employees that work in department 4000:

```
select deptid, firstname, lastname
from emp
where deptid = 4000 ! display
```

```
EMP REPORT

DEPTID  FIRSTNAME  LASTNAME
4000    MONTE      BANK
4000    MICHAEL    ANGELO
4000    CAROL      MCDUGALL
4000    LAURA      PENMAN
4000    JOCK       JACKSON
4000    BETSY      ZEDI

END OF REPORT
```

Example 2 -- Comparing a character

List the department and names of all female employees:

```
select deptid, firstname, lastname
from emp
where sex = &xq.f' ! display
```

```
EMP REPORT

DEPTID  FIRSTNAME  LASTNAME
3100    JENNIFER   GARFIELD
3100    JULIE       JENSEN
3100    JANE        DOUGH

END OF REPORT
```
Example 3 -- Selecting lower values

List the hire date, in ascending order, and names of all employees employed before January 1, 1978:

```
select startdate as 'hire date', firstname, lastname
from emp
where startdate < 780101
order by startdate
```

```
EMP REPORT

HIRE DATE           FIRSTNAME     LASTNAME
731121              HENRIETTA     HENDON
750223              JOHN          RUPEE
760223              BETSY         ZEDI
760808              JANE          DOUGH
770121              JENNIFER      GARFIELD
770707              JOCK          JACKSON
770908              LAURA         PENMAN

END OF REPORT
```

Example 4 -- Using a calculation

List all job classes where the salary range is less than $3,000:

```
select class, minsalary as 'minimum salary', maxsalary as 'maximum salary'
from jobclass
where (maxsalary - minsalary) < 3000
```

```
JOBCLASS REPORT

CLASS         MINIMUM SALARY     MAXIMUM SALARY
21            18000.00           20000.00
33            37000.00           39000.00
42            33000.00           35000.00

END OF REPORT
```

Complex comparisons

Using AND and OR

You can specify more than one comparison expression to select rows from a table by connecting each expression with an AND or an OR:

**AND**  Retrieves the row if each comparison expression is true

**OR**   Retrieves the row if any comparison expression is true

If a WHERE clause contains both AND and OR, CA OLQ evaluates the OR expressions first. For example, to process the example below, CA OLQ determines whether the employee's department is 4000 or whether the employee was hired before January 1, 1980. If either of these conditions is true, CA OLQ determines if the employee's job class is 21:

```
where class=21 and deptid=4000 or startdate<800101
```
Note: You can use parentheses to indicate the order in which CA OLQ evaluates the expressions. You can also use parentheses to improve the readability and accuracy of complex expressions.

Example 1 -- Using AND

List information on all employees who were hired before January 1, 1980 and whose salary exceeds $50,000:

```sql
select empid, lastname, startdate as 'hire date', salary from emp where (startdate < 800101) and (salary > 50000) ! display
```

<table>
<thead>
<tr>
<th>EMPID</th>
<th>LASTNAME</th>
<th>HIRE DATE</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>RUPEE</td>
<td>750223</td>
<td>76000.00</td>
</tr>
<tr>
<td>0003</td>
<td>GARFIELD</td>
<td>770121</td>
<td>65000.00</td>
</tr>
<tr>
<td>0007</td>
<td>BANK</td>
<td>780430</td>
<td>80000.00</td>
</tr>
<tr>
<td>0030</td>
<td>HENDON</td>
<td>731121</td>
<td>240000.00</td>
</tr>
<tr>
<td>0471</td>
<td>PAPAZEUS</td>
<td>780907</td>
<td>90000.00</td>
</tr>
<tr>
<td>0472</td>
<td>WILDER</td>
<td>790716</td>
<td>90000.00</td>
</tr>
</tbody>
</table>

END OF REPORT

Example 2 -- Using parentheses with AND and OR

Parentheses determine how CA OLQ evaluates complex comparisons. Each of the SELECT statements shown below have the same three comparison expressions. However, the first SELECT statement uses parentheses to group the expressions connected by AND. The second groups the expressions connected by OR.

**SELECT Statement 1**

List the names, birthdays, and salaries of all employees who are either:

- Female, born before January 1, 1947 or
- Making less than $25,000

```sql
select lastname, firstname, birthdate, salary from emp where (sex = 'f' and birthdate < 470101) or (salary < 25000) ! display
```

<table>
<thead>
<tr>
<th>LASTNAME</th>
<th>FIRSTNAME</th>
<th>BIRTHDATE</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>HENDON</td>
<td>HENRIETTA</td>
<td>331006</td>
<td>240000.0</td>
</tr>
<tr>
<td>ZEDI</td>
<td>BETSY</td>
<td>401229</td>
<td>37000.00</td>
</tr>
<tr>
<td>PENMAN</td>
<td>LAURA</td>
<td>440504</td>
<td>39000.00</td>
</tr>
<tr>
<td>GARFIELD</td>
<td>JENNIFER</td>
<td>450818</td>
<td>65000.00</td>
</tr>
<tr>
<td>TYRO</td>
<td>RALPH</td>
<td>551225</td>
<td>20000.00</td>
</tr>
<tr>
<td>ANGELO</td>
<td>MICHAEL</td>
<td>576405</td>
<td>18000.00</td>
</tr>
<tr>
<td>MCDougall</td>
<td>CAROL</td>
<td>590304</td>
<td>18000.00</td>
</tr>
</tbody>
</table>

END OF REPORT

**SELECT Statement 2**
In comparison, list the same information on employees who:

- Are female and
- Earn less than $25,000 or were born before January 1, 1947

```
select lastname, firstname, birthdate, salary from emp where (sex = 'f') and (birthdate < 470101 or salary < 25000) order by birthdate
```

```
EMP REPORT

LASTNAME | FIRSTNAME | BIRTHDATE   | SALARY   
----------|-----------|-------------|----------
HENDON    | HENRIETTA | 331006      | 240000.00 |
ZEDI      | BETSY     | 401229      | 37000.00  |
PENMAN    | LAURA     | 440504      | 39000.00  |
GARFIELD  | JENNIFER  | 450818      | 65000.00  |
MCDougall | CAROL     | 590304      | 18000.00  |

END OF REPORT
```

Comparisons to a list of values

Using **IN**

To compare a value to one of several values in a list, use the IN keyword. The IN keyword is a short way of coding two or more comparison expressions connected by an OR. For example, both of these SELECT statements yield the same result:

```
select empid from emp where deptid in (3100, 4000)
select empid from emp where (deptid = 3100) or (deptid = 4000)
```

Separate each value in the list by a comma. A blank following the comma is optional.

Example

List all employees whose job falls into one of these classes: 11, 21, 43, or 71 (the report shown below indicates that there are no employees that have a job in class 11):

```
select firstname, lastname, class from emp where class in (11, 21, 43, 71) order by class
display
```

```
EMP REPORT

FIRSTNAME | LASTNAME | CLASS  
-----------|----------|--------
RALPH      | TYRO     | 21     |
MICHAEL    | ANGELO   | 21     |
CAROL      | MCDougall| 21     |
JULIE      | JENSEN   | 43     |
LARRY      | LITERATA | 43     |
JENNIFER   | GARFIELD | 71     |

END OF REPORT
```

Exclusive comparisons

Using **NOT**
You can retrieve all rows that are exceptions to the comparison expression by using the keyword NOT. For example, you can retrieve information on all employees except those that work in departments 6666 and 3100.

A few examples appear below:

<table>
<thead>
<tr>
<th>Example</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>where not (deptid &lt; 4000)</td>
<td>All rows where the department id is greater than or equal to 4000</td>
</tr>
<tr>
<td>where deptid = 4000 and not (empid = 0007)</td>
<td>All rows where the department id is 4000 except the row with employee id 0007</td>
</tr>
</tbody>
</table>

Example

List all employees except those with manager id 0007:

```sql
select firstname, lastname, mgrid from emp where not (mgrid = 0007) order by mgrid
```

display

<table>
<thead>
<tr>
<th>FIRSTNAME</th>
<th>LASTNAME</th>
<th>MGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>LARRY</td>
<td>LITERATA</td>
<td>0003</td>
</tr>
<tr>
<td>RALPH</td>
<td>TYRO</td>
<td>0003</td>
</tr>
<tr>
<td>JULIE</td>
<td>JENSEN</td>
<td>0003</td>
</tr>
<tr>
<td>JAMES</td>
<td>JACOBI</td>
<td>0003</td>
</tr>
<tr>
<td>VLADIMIR</td>
<td>HEARWITZ</td>
<td>0003</td>
</tr>
<tr>
<td>JANE</td>
<td>DOUGH</td>
<td>0003</td>
</tr>
<tr>
<td>PERCY</td>
<td>GRANGER</td>
<td>0003</td>
</tr>
<tr>
<td>JAMES</td>
<td>GALLWAY</td>
<td>0003</td>
</tr>
<tr>
<td>MONTE</td>
<td>BANK</td>
<td>0030</td>
</tr>
<tr>
<td>JENNIFER</td>
<td>GARFIELD</td>
<td>0030</td>
</tr>
<tr>
<td>HENRIETTA</td>
<td>HENDON</td>
<td>0030</td>
</tr>
<tr>
<td>JOHN</td>
<td>RUPEE</td>
<td>0030</td>
</tr>
<tr>
<td>THEMIS</td>
<td>PAPAZEUS</td>
<td>0030</td>
</tr>
<tr>
<td>ROBBY</td>
<td>WILDER</td>
<td>0030</td>
</tr>
</tbody>
</table>

- 1 -

Range comparisons

Using BETWEEN

To retrieve rows from a table where the values of one column fall in a range of values, use BETWEEN. BETWEEN selects all rows that have values in between and equal to the starting and ending values of the specified range.

Example

List the ID, job class, and salary of all employees who earn from $33,000 to $39,000:

```sql
select empid, class, salary from emp where salary between 33000 and 39000 order by salary
```

display

```
EMP REPORT 
mm/dd/yy

<table>
<thead>
<tr>
<th>EMPID</th>
<th>CLASS</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

- 1 -
Character string comparisons

Using LIKE

You can compare an alphanumeric field to a mask (pattern) that contains alphanumeric characters and wild card symbols. For example, you might want to retrieve information on jobs that have CLERK in the job title.

To code the pattern, use the wild card symbols described below. If the pattern contains embedded blanks, enclose it in single quotation marks.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Percent (%)</td>
<td>Specifies from 0 to any number of unknown characters</td>
</tr>
<tr>
<td>Underscore (_)</td>
<td>Specifies a single unknown character</td>
</tr>
</tbody>
</table>

Some examples using these symbols appear below:

<table>
<thead>
<tr>
<th>Example</th>
<th>Selects the row if the value contains</th>
</tr>
</thead>
<tbody>
<tr>
<td>'%m%'</td>
<td>An M</td>
</tr>
<tr>
<td>'<em>m</em>'</td>
<td>3 characters with an M in the middle</td>
</tr>
</tbody>
</table>

Note: To improve CA OLQ's performance, use conditional operators, rather than LIKE, to perform character string comparisons. For example, use **where firstname = 'b'** to retrieve all employees whose first name begins with B.

Example 1 -- Using a % sign

List the names of all employees with initials JG:

```
select firstname, lastname from emp where firstname like &xq.j%' and lastname like &xq.g%' ! display
```

EMP REPORT
mm/dd/yy

FIRSTNAME         LASTNAME
JENNIFER          GARFIELD
Using Built-In Functions

Definition

CA OLQ provides many predefined functions that evaluate expressions and return results. These functions, called built-in functions, can be used anywhere you would normally specify arithmetic or comparison expressions. The built-in functions fall into these categories:

<table>
<thead>
<tr>
<th>Category</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>Perform operations on character strings, such as concatenating BUSY and BEE</td>
</tr>
<tr>
<td>Arithmetic</td>
<td>Perform arithmetic operations, such as rounding 5.77 to 6</td>
</tr>
<tr>
<td>Trigonometric</td>
<td>Perform mathematical calculations, such as evaluating the cosine of 30 degrees</td>
</tr>
<tr>
<td>Date</td>
<td>Perform calculations on dates, such as evaluating the number of days between January 14, 1956 and June 26, 1987</td>
</tr>
</tbody>
</table>

The examples shown below are only a few of the built-in functions you can use. For information on all the built-in functions CA OLQ provides, see the CA OLQ Reference Section.

Example 1 -- Retrieving a substring
Create a 3-letter department code for each department by using the first three letters of the department name:

```sql
select substring(deptname,1,3) as dept code from dept ! display
```

<table>
<thead>
<tr>
<th>DEPT REPORT</th>
<th>mm/dd/yy</th>
<th>DEPT CODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACC</td>
<td></td>
<td>ACC</td>
</tr>
<tr>
<td>BLU</td>
<td></td>
<td>BLU</td>
</tr>
<tr>
<td>BRA</td>
<td></td>
<td>BRA</td>
</tr>
<tr>
<td>COM</td>
<td></td>
<td>COM</td>
</tr>
<tr>
<td>EXE</td>
<td></td>
<td>EXE</td>
</tr>
<tr>
<td>INT</td>
<td></td>
<td>INT</td>
</tr>
<tr>
<td>PER</td>
<td></td>
<td>PER</td>
</tr>
<tr>
<td>PUB</td>
<td></td>
<td>PUB</td>
</tr>
<tr>
<td>THE</td>
<td></td>
<td>THE</td>
</tr>
</tbody>
</table>

END OF REPORT

Example 2 -- Concatenating names

List the last name of each employee, followed by a comma and a blank, and the employee's first name. The EXTRACT function deletes all trailing blanks from the employee's last name. The CONCATENATE function strings together:

- The character string returned by the EXTRACT function

- The literal containing the comma and the blank

- The employee's first name

```sql
select concatenate(extract(lastname), ', ',firstname) from emp ! display
```

<table>
<thead>
<tr>
<th>EMP REPORT</th>
<th>mm/dd/yy</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUPEE, JOHN</td>
<td></td>
</tr>
<tr>
<td>GARFIELD, JENNIFER</td>
<td></td>
</tr>
<tr>
<td>BANK, MONTE</td>
<td></td>
</tr>
<tr>
<td>JENSEN, JULIE</td>
<td></td>
</tr>
<tr>
<td>JACOBI, JAMES</td>
<td></td>
</tr>
<tr>
<td>TYRO, RALPH</td>
<td></td>
</tr>
<tr>
<td>DOUGH, JANE</td>
<td></td>
</tr>
<tr>
<td>HEAROWITZ, VLADIMIR</td>
<td></td>
</tr>
<tr>
<td>GRANGER, PERCY</td>
<td></td>
</tr>
<tr>
<td>GALLWAY, JAMES</td>
<td></td>
</tr>
<tr>
<td>HENDON, HENRIETTA</td>
<td></td>
</tr>
<tr>
<td>LITERATA, LARRY</td>
<td></td>
</tr>
<tr>
<td>ANGELO, MICHAEL</td>
<td></td>
</tr>
<tr>
<td>MCDouGALL, CAROL</td>
<td></td>
</tr>
</tbody>
</table>

- 1 -

Example 3 -- Determining a date's weekday

List the day of the week each employee was born. The GWEEKDAY function accepts a Gregorian date and returns the day of the week on which that date falls:

```sql
select lastname, birthdate, gweekday(birthdate) as weekday from emp ! display
```
Example 4 -- Calculating employee age

List each employee's name and age. The DATEDIF function determines the number of days between today's date (January 13, 1999) and the employee's birthday. The NEXT-INT-EQLO function rounds the number of years to the next lowest integer:

```sql
select firstname, lastname,
       next-int-eqlo(datedif(990113, birthdate)/365)
from emp
```
Manager with ID 0007 or 0003

Job that begins with the number 3

3. Identify all employees whose sex code was entered incorrectly (that is, is not M or F)

4. List the number of years employees have from January 1, 1988 until they reach retirement age (65)

Summarizing Information

Summary lines This section tells you how to create reports containing summary lines, rather than detail lines. A detail line contains information on individual rows in a table. Summary lines contain summary information on detail lines. For example, a line containing salary information on Henrietta Hendon is a detail line. A line containing total salaries for all employees is a summary line.

Aggregate functions

To summarize information, CA OLQ provides the aggregate functions shown below. An aggregate function is a type of built-in function that evaluates all the values in a column and returns a single value.

<table>
<thead>
<tr>
<th>Function</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUNT</td>
<td>Counts the number of rows</td>
</tr>
<tr>
<td>SUM</td>
<td>Supplies a total value for the named column</td>
</tr>
<tr>
<td>AVG</td>
<td>Supplies an average value for the named column</td>
</tr>
<tr>
<td>MIN</td>
<td>Supplies the lowest value in the named column</td>
</tr>
<tr>
<td>MAX</td>
<td>Supplies the highest value in the named column</td>
</tr>
<tr>
<td>STD</td>
<td>Standard deviation</td>
</tr>
<tr>
<td>VAR</td>
<td>Variance</td>
</tr>
</tbody>
</table>

How to use aggregate functions

You use aggregate functions in the list of column names or expressions following the SELECT statement. The function is calculated by giving the function (for example, SUM) followed by a column name or value in parentheses.

You can use all aggregate functions, except COUNT, with:

- Values in a column: sum(salary)
- An arithmetic expression: sum(salary * 0.06)
- Any combination of the items listed above

To count the number of selected rows, use COUNT followed by an asterisk in parentheses: count(*). CA OLQ displays the value in the report.
Example -- Aggregate functions in column lists

List the number of employees within the company, and the company's total salary payment and average salary payment:

\[
\text{select count(*) as \#number of employees', sum(salary) as \#total salary', avg(salary) as \#average salary' from emp ! display}
\]

<table>
<thead>
<tr>
<th>NUMBER OF EMPLOYEES</th>
<th>TOTAL SALARY</th>
<th>AVERAGE SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>1070000.00</td>
<td>56315.78</td>
</tr>
</tbody>
</table>

END OF REPORT

For more information, see the following topics:
- Summarizing Data In Groups (see page 188)
- Specifying a Condition For a Group (see page 189)
- Summarizing Information From Another Table (see page 189)
- Testing Your Knowledge 1 (see page 190)

Summarizing Data In Groups

Groups

You can display summary information on groups. A group is a collection of detail lines that share a common value in one or more columns. For example, you can display summary salary information for each department. Summarizing information on groups is similar to break processing for those familiar with that reporting terminology.

Use GROUP BY

To summarize information for groups of values, use the GROUP BY clause. The GROUP BY clause indicates which columns to group. For example, this clause groups all rows that share the same department ID: `GROUP BY deptid`.

You can specify up to 31 columns in the GROUP BY clause. For example, you can group rows by department, and within the department, by job ID: `GROUP BY deptid, jobid`.

When you group rows, each column listed in the SELECT statement, except those named in the GROUP BY clause, must be acted upon by an aggregate function, such as AVG or MIN. For example, you might group rows by department ID to return average and minimum salaries for each department. CA OLQ displays one row for each group it evaluates.

Example 1 -- Grouping based on 1 column

List the number of employees in each department and the department's total and average salaries:

\[
\text{select deptid, count(*) as \#number of employees', sum(salary) as \#total salary', avg(salary) as \#average salary' from emp group by deptid ! display}
\]

EMP REPORT

<table>
<thead>
<tr>
<th>DEPTID</th>
<th>NUMBER</th>
<th>TOTAL</th>
<th>AVERAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>SALARY</td>
<td>SALARY</td>
</tr>
<tr>
<td>19</td>
<td></td>
<td>1070000.00</td>
<td>56315.78</td>
</tr>
</tbody>
</table>
Example 2 -- Grouping based on 2 columns

For each department, list the number of employees, total salary, and average salary by gender. Notice that a sex code was entered incorrectly for an employee in department 6666:

```
select deptid, sex, count(*) as number of employees', sum(salary) as total salary', avg(salary) as average salary'
from emp
group by deptid, sex
```

```
EMP REPORT
dd/mm/yy
DEPTID  SEX  NUMBER OF EMPLOYEES  TOTAL SALARY  AVERAGE SALARY
3100   F    3                135000.00  45000.00
3100   M    6                213000.00  35500.00
4000   F    3                 94000.00  31333.33
4000   M    3                132000.00  44000.00
6666   F    1                240000.00 240000.00
6666   M    2                166000.00  83000.00
6666   N    1                 90000.00  90000.00
END OF REPORT
```

Specifying a Condition For a Group

**Use HAVING**

To apply selection criteria to the result of a GROUP BY clause, use a HAVING clause after the GROUP BY clause. The HAVING clause is similar to a WHERE clause, except that it applies to summary rows only. As in the WHERE clause, you can connect multiple conditional expressions by using AND and OR.

**Example**

List the departments where the total salary exceeds $300,000:

```
select deptid, sum(salary) as total salary', avg(salary) as average salary'
from emp
group by deptid
having sum(salary) > 300000
```

```
EMP REPORT
dd/mm/yy
DEPTID  TOTAL SALARY  AVERAGE SALARY
3100    348000.00  38666.66
6666    496000.00  124000.00
END OF REPORT
```

Summarizing Information From Another Table

**Using a nested SELECT statement**
A SELECT statement within a WHERE clause is called a nested SELECT. You can use a nested SELECT statement to retrieve information from one table based on summary information in another table.

**Example -- COUNT in a nested SELECT**

List the jobs held by more than one employee. The SELECT statement contains a nested SELECT that returns the number of employees assigned to each job ID. The primary SELECT statement determines if the number is greater than 1:

```sql
select distinct jobid, title from joblist where 1 < (select count(*) from emp where joblist.jobid = emp.jobid) ! display
```

JOBLIST REPORT
02/10/99

<table>
<thead>
<tr>
<th>JOBID</th>
<th>TITLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>3001</td>
<td>MGR INTERNL SOFTWARE</td>
</tr>
<tr>
<td>3025</td>
<td>PROGRAMMER/ANALYST</td>
</tr>
<tr>
<td>4025</td>
<td>PR WRITER</td>
</tr>
</tbody>
</table>

END OF REPORT

**Testing Your Knowledge 1**

Using the sample tables in num=A.Sample Tables and Database, code a SELECT statement for each of the queries listed below. num=B.Answers, contains one possible answer. Remember, there can be more than one way to achieve the same result when you use the SELECT statement.

1. For each manager in table EMP, determine the number of staff reporting to the manager and the average salary of the staff members.

2. List the number of jobs greater than 3 that are assigned to classes 10 and 50. Display the report in order by the number of jobs.

3. For departments 6666, 4000, and 3100, list the minimum and maximum salaries within the department, provided the average departmental salary is greater than $37,800.

4. List information on employees earning less than the average salary.

**Joining Tables or Database Records**

**Tables**

In many cases, you want to retrieve information from two or more tables. For example, to retrieve the name of each department manager, you must get the manager’s ID from the DEPT table and the name that corresponds to the ID from the EMP table. With CA OLQ, you can obtain information from two tables by joining the tables based on a column both tables have in common.

**Database records**
Additionally, if you are using CA IDMS/DB database records, you can join the records based on a common value or you can join the records by specifying a record-to-record set relationship. For example, to retrieve information on employees and where they work, you would use the OFFICE-EMPLOYEE set of the Employee Database.

**Tables and records**

In some cases, you will want to retrieve information that exists in a table and in a record; for example, a CA IDMS/DB record (DEPARTMENT) and a table (EMP).

This section explains how to retrieve data from each of these data structures.

- Joining Tables (see page 191)
- Retrieving Information From CA IDMS/DB Records (see page 197)
- Retrieving Data From Tables and Records (see page 203)
- Test Your Knowledge (see page 204)

## Joining Tables

### Contents

- Joining different tables (see page 191)
- Joining a table to itself (see page 194)
- Comparing a column to more than one value (see page 197)

You can join tables that share a column of common values. For example, you can join:

- Two or more different tables.
- A table to itself when one column contains similar values to another column. For example, table EMP contains a column of manager IDs. You would join table EMP to itself to determine the name of an employee's manager.

Both of these topics are described below. This section also describes a method to translate a nested SELECT that retrieves more than one value into a SELECT statement that is a join operation.

### Joining different tables

**Associate one column with another**

To join tables together, each table must have at least one column that corresponds to a column in another table. You join tables together by equating these columns in the WHERE clause of the SELECT statement. The WHERE clause defines the join condition. This figure joins the EMP and DEPT tables by equating the department ID values in both tables:

<table>
<thead>
<tr>
<th>EMP table</th>
<th>DEPT table</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPNAME</td>
<td>DEPTID</td>
</tr>
<tr>
<td>Jennifer Lanzarotta</td>
<td>3100</td>
</tr>
<tr>
<td>Bart Elopoulos</td>
<td>3200</td>
</tr>
<tr>
<td></td>
<td>DEPTID</td>
</tr>
<tr>
<td></td>
<td>3100</td>
</tr>
<tr>
<td></td>
<td>3200</td>
</tr>
<tr>
<td></td>
<td>DEPTNAME</td>
</tr>
<tr>
<td></td>
<td>Internal Software</td>
</tr>
<tr>
<td></td>
<td>Computer Operations</td>
</tr>
</tbody>
</table>
Join rows in EMP table and DEPT table that have DEPTID equal to 5300 to list information about department 5300 and its employees.

<table>
<thead>
<tr>
<th>EMPNAME</th>
<th>DEPTID</th>
<th>DEPTNAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ankur Sahu</td>
<td>5300</td>
<td>Blue Skies</td>
</tr>
</tbody>
</table>

**Which columns can you use?**

The columns should contain comparable data. For example, you could compare the EMPID column in table EMP with the MGRID column in table DEPT. Both columns contain employee ID values.

**Choosing a column**

In some cases, the tables you want to join will have more than one corresponding column. For example, you can join the EMP and DEPT tables by comparing:

- The DEPTID column defined in both tables, or
- The EMPID column defined in the EMP table and the MGRID column defined in the DEPT table

If you compare the department ID values, CA OLQ retrieves information on all employees and their departments. If you compare employee and manager ID values, CA OLQ retrieves information on all employees who are department managers and the departments they manage.

**Qualifying column names**

When the tables you want to join have the same names for some or all of the columns (like the DEPTID column in the EMP and DEPT tables), qualify the column names by specifying the table name, followed by a period and the column name: `dept.deptid`. In fact, it's a good idea to qualify all column names in join operations to make the SELECT statement easier to read.

**Coding the SELECT statement**

To join tables in a SELECT statement:

1. Name selected columns from any or all of the tables in the column list following the SELECT keyword: `dept.deptid, emp.lastname`. 
2. Name the tables, separated by a comma, in the FROM clause of the SELECT statement: *dept*, *emp*. The order of the tables is not important.

3. Compare the values of the associated columns in the WHERE clause: *dept.deptid = emp.deptid*. The WHERE clause can contain more than one comparison expression, as shown in Example 2 below.

Example 1 -- Name the department managers

List information on each department manager. The SELECT statement joins the DEPT and EMP tables by getting rows from both tables where the manager ID in the department table is the same as the employee ID in the EMP table:

```
select dept.deptid, dept.deptname, emp.firstname, emp.lastname
from dept, emp
where dept.mgrid = emp.empid
```

```
DEPT/EMP REPORT
mm/dd/yy

DEPTID    DEPTNAME            FIRSTNAME  LASTNAME
--------  --------------------  ----------  ----------
   3100    INTERNAL SOFTWARE  JENNIFER   GARFIELD
   4000    PUBLIC RELATIONS   MONTE       BANK
  6666    EXECUTIVE ADMINISTRATION  HENRIETTA   HENDON

END OF REPORT
```

Example 2 -- Name employees hired before their manager

Assuming that employee IDs are assigned sequentially, list all employees who have worked at the company longer than their manager; that is, those employees who have a lower ID than that of the department's manager. The SELECT statement joins the EMP and DEPT tables by retrieving all rows where:

- The employee's department is the same as the manager's department, and
- The employee's ID is less than the manager's ID

```
select emp.empid, emp.lastname, emp.firstname, dept.mgrid
from emp, dept
where (dept.deptid = emp.deptid) and (emp.empid < dept.mgrid)
```

```
EMP/DEPT REPORT
mm/dd/yy

EMPID    LASTNAME   FIRSTNAME  MGRID
------    ----------  ---------  ------
  0001    RUPEE      JOHN      0030

END OF REPORT
```

Joining more than two tables

If you need to join more than two tables, specify a join condition for each pair of tables. That is, to join three tables, you'll need at least two join conditions. For example, to join the EMP, DEPT, and JOBCLASS tables, you could join the tables this way:

<table>
<thead>
<tr>
<th>Tables</th>
<th>Join condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMP and DEPT</td>
<td>dept.deptid = emp.deptid</td>
</tr>
<tr>
<td>EMP and JOBCLASS</td>
<td>emp.class = jobclass.class</td>
</tr>
</tbody>
</table>
The resulting report would contain information on each employee's department and job class.

**Example -- Join three tables**

List job information on all employees who earn the minimum salary for their job class. The SELECT statement joins three tables: EMP, JOBLIST, and JOBCLASS. CA OLQ retrieves all rows where:

- The employee ID in the EMP table matches the employee ID assigned to a job in the JOBLIST table, and
- The employee's job class matches the class assigned in the JOBCLASS table, and
- The employee's salary equals the minimum salary in the JOBCLASS table

```sql
select emp.empid, emp.lastname, joblist.title, jobclass.class, emp.salary
from emp, joblist, jobclass
where joblist.empid=emp.empid and
      jobclass.class = emp.class and
      emp.salary = jobclass.minsalary
```

**EMP/JOBLIST/JOBCLASS REPORT**

<table>
<thead>
<tr>
<th>EMPID</th>
<th>LASTNAME</th>
<th>TITLE</th>
<th>CLASS</th>
<th>SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>RUPEE</td>
<td>DIR OPERATIONS</td>
<td>72</td>
<td>76000.00</td>
</tr>
<tr>
<td>0024</td>
<td>DOUGH</td>
<td>PROGRAMMER/ANALYST</td>
<td>42</td>
<td>33000.00</td>
</tr>
<tr>
<td>0027</td>
<td>HEAROWITZ</td>
<td>PROGRAMMER/ANALYST</td>
<td>42</td>
<td>33000.00</td>
</tr>
<tr>
<td>0029</td>
<td>GALLWAY</td>
<td>PROGRAMMER/ANALYST</td>
<td>42</td>
<td>33000.00</td>
</tr>
<tr>
<td>0120</td>
<td>ANGELO</td>
<td>ILLUSTRATOR</td>
<td>21</td>
<td>18000.00</td>
</tr>
<tr>
<td>0129</td>
<td>ANGELO</td>
<td>PASTE-UP ARTIST</td>
<td>21</td>
<td>18000.00</td>
</tr>
<tr>
<td>0127</td>
<td>MCDougall</td>
<td>PASTE-UP ARTIST</td>
<td>21</td>
<td>18000.00</td>
</tr>
<tr>
<td>0476</td>
<td>ZEDI</td>
<td>PR WRITER</td>
<td>33</td>
<td>37000.00</td>
</tr>
</tbody>
</table>

END OF REPORT

**Joining a table to itself**

**Why join a table to itself?**

You join a table to itself when one column in a table requires the table itself to supply additional information. For example, table EMP has a column of manager IDs. To find the name of Michael Angelo's manager, you find the manager's ID in the MGRID column and then find the same ID in the EMPID column. The manager's name is associated with the employee ID:

**EMP table**

<table>
<thead>
<tr>
<th>EMPID</th>
<th>EMPNAME</th>
<th>MGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0075</td>
<td>Jennifer Lanzarotta</td>
<td>0003</td>
</tr>
<tr>
<td>3302</td>
<td>Bart Elpoulos</td>
<td>0004</td>
</tr>
<tr>
<td>3871</td>
<td>Reginald Mahoney</td>
<td>0007</td>
</tr>
<tr>
<td>4230</td>
<td>Duc Ho</td>
<td>0011</td>
</tr>
<tr>
<td>6264</td>
<td>Daniel Ortega</td>
<td>0013</td>
</tr>
<tr>
<td>6348</td>
<td>Edith Jones</td>
<td>0015</td>
</tr>
<tr>
<td>7170</td>
<td>Anthea Poznanski</td>
<td>0075</td>
</tr>
<tr>
<td>8939</td>
<td>Ankur Sahu</td>
<td>0321</td>
</tr>
<tr>
<td>8957</td>
<td>William Sternbach</td>
<td>0349</td>
</tr>
</tbody>
</table>
Join table EMP to itself by equating Anthea Poznanski's manager ID to an employee ID

<table>
<thead>
<tr>
<th>EMPID</th>
<th>EMPNAME</th>
<th>MGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>7170</td>
<td>Anthea Poznanski</td>
<td>0075</td>
</tr>
</tbody>
</table>

This type of join is called a **reflexive join** and is used to implement a **nested structure**, which is also called a **bill-of-materials** structure. A nested structure is one where there is a relationship between columns in the same table. For example, nested relationships exist:

- In an industrial environment where a part in a component of another part and can contain component parts itself. For example, a door is a component of a car and contains these component parts: handle, lock, and window.

- In the corporate environment used for examples in this section where an employee manages other employees. Likewise, an employee can report to more than one supervisor.

**How to join a table to itself**

To join a table to itself, you simply treat the table as two tables by assigning **aliases**, or alternative names, to the table in the SELECT statement. Thereafter, the SELECT statement coding requirements listed below are the same as if you were joining two different tables. This figure illustrates how to join table EMP to itself by assigning two alias table names -- MANAGER and WORKER:

**EMP table**

<table>
<thead>
<tr>
<th>EMPID</th>
<th>EMPNAME</th>
<th>MGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0075</td>
<td>Jennifer Lanzarotta</td>
<td>0003</td>
</tr>
<tr>
<td>3302</td>
<td>Bart Elopoulos</td>
<td>0004</td>
</tr>
<tr>
<td>3871</td>
<td>Reginald Mahoney</td>
<td>0007</td>
</tr>
<tr>
<td>4242</td>
<td>Chinua Achebe</td>
<td>0075</td>
</tr>
<tr>
<td>7170</td>
<td>Anthea Poznanski</td>
<td>0075</td>
</tr>
</tbody>
</table>

Assign alias table names

**MANAGER table**

<table>
<thead>
<tr>
<th>EMPID</th>
<th>EMPNAME</th>
<th>MGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0075</td>
<td>Jennifer Lanzarotta</td>
<td>0003</td>
</tr>
<tr>
<td>3302</td>
<td>Bart Elopoulos</td>
<td>0004</td>
</tr>
<tr>
<td>3871</td>
<td>Reginald Mahoney</td>
<td>0007</td>
</tr>
<tr>
<td>4242</td>
<td>Chinua Achebe</td>
<td>0075</td>
</tr>
<tr>
<td>7170</td>
<td>Anthea Poznanski</td>
<td>0075</td>
</tr>
</tbody>
</table>

**WORKER table**

Join the **MANAGER** and **WORKER** tables by equating the ID of the worker's manager to the employee ID of the manager

<table>
<thead>
<tr>
<th>EMPID</th>
<th>EMPNAME</th>
<th>MGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0075</td>
<td>Jennifer Lanzarotta</td>
<td>0003</td>
</tr>
<tr>
<td>3302</td>
<td>Bart Elopoulos</td>
<td>0004</td>
</tr>
<tr>
<td>3871</td>
<td>Reginald Mahoney</td>
<td>0007</td>
</tr>
<tr>
<td>4242</td>
<td>Chinua Achebe</td>
<td>0075</td>
</tr>
<tr>
<td>7170</td>
<td>Anthea Poznanski</td>
<td>0075</td>
</tr>
</tbody>
</table>
**Coding the SELECT statement**

To join a table to itself, follow these steps:

1. Qualify each column listed after the SELECT keyword with an alias table name: `manager.lastname`
2. For each reflexive join (that is, for each time you join a table to itself), assign an alias by coding:
   a. The table name
   b. A blank
   c. The alias

Separate each table and its alias from another with a comma: `emp manager, emp worker`

3. In the WHERE clause, compare two columns that share the same type of information: `manager.empid = worker.mgrid`

**Example**

List each manager and associated staff. To retrieve this information, join the EMP table to itself, equating a manager's employee ID to the ID of a staff member's manager.

The SELECT statement assigns these aliases to EMP table: MANAGER and WORKER. The WHERE clause selects rows where the employee ID in the MANAGER table equals the manager's ID in the WORKER table. The columns display the manager's name retrieved from the MANAGER table and the worker's name retrieved from the WORKER table:

```sql
SELECT manager.lastname as supervisor, worker.lastname as staff
FROM emp manager, emp worker
WHERE manager.empid = worker.mgrid;
```

<table>
<thead>
<tr>
<th>SUPERVISOR</th>
<th>STAFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>GARFIELD</td>
<td>JENSEN</td>
</tr>
<tr>
<td>GARFIELD</td>
<td>JACOBI</td>
</tr>
<tr>
<td>GARFIELD</td>
<td>TYRO</td>
</tr>
<tr>
<td>GARFIELD</td>
<td>DOUGH</td>
</tr>
<tr>
<td>GARFIELD</td>
<td>HEAROWITZ</td>
</tr>
<tr>
<td>GARFIELD</td>
<td>GRANGER</td>
</tr>
<tr>
<td>GARFIELD</td>
<td>GALLWAY</td>
</tr>
<tr>
<td>GARFIELD</td>
<td>LITERATA</td>
</tr>
<tr>
<td>BANK</td>
<td>ANGELO</td>
</tr>
</tbody>
</table>
Comparing a column to more than one value

Why you compare more than one value

You may want to compare values in one table to a list of values in another table. For example, you might want to obtain information on all employees whose ID matches the manager IDs in table DEPT. If you know the manager IDs in table DEPT, you could code a SELECT statement that compares employee IDs to a list of manager IDs:

```
select empid, lastname from emp where empid in (0013, 0011, 0003, 0004, 0007, 0015, 0349, 0321, 0030)
```

However, this type of query isn't practical in some cases, especially for large tables. CA OLQ provides a method to retrieve this information.

Coding it as a join operation

You can obtain the same information by joining the EMP and DEPT tables where the manager ID in table DEPT is the same as the employee ID in table EMP:

```
select emp.lastname from dept, emp where emp.empid = dept.mgrid
```

Retrieving Information From CA IDMS/DB Records

Contents

- Retrieving data from a single record (see page 198)
- Retrieving data from two or more records (see page 199)
- Retrieving data from a record joined to itself (see page 201)

Examples to this point are all based on ASF-generated tables. However, you can also use the SELECT statement to retrieve data from CA IDMS/DB database records. This portion of the section tells you how to retrieve data from:

- One CA IDMS/DB record
- Two or more CA IDMS/DB records by using sets
- A CA IDMS/DB bill-of-materials data structure
Retrieving data from a single record

Comparing a record to a table

Retrieving information from a CA IDMS/DB database record is comparable to retrieving rows from a table. The CA IDMS/DB record type (for example, the EMPLOYEE record) is like a table (for example, the EMP table). The EMPLOYEE record occurrences are like the rows of EMP table.

Signing on to a subschema

To access CA IDMS/DB database records, you have to first sign on to a subschema. A subschema is a view of the database. That is, it describes a subset of the database records that a CA IDMS/DB database contains.

To signon to a subschema, issue a SIGNON command:

```
SIGNON ss=empss01
```

Coding the SELECT statement

To retrieve information from a single CA IDMS/DB database record, code the SELECT statement like this:

1. Enter record field names instead of column names following the SELECT keyword: `emp-last-name-0415`
2. Enter the record name instead of the table name following the FROM keyword: `employee`

Finding record names

If you do not know the record or the record field names that you need, issue these commands:

```
HELP RECORDS
HELP RECORD=record-name
```

Example

List each office's code number, in ascending order, and city:

```
SELECT office-code-0450, office-city-0450 FROM office ORDER BY office-code-0450 !
DISPLAY
```

```
OFFICE REPORT
01/27/99

<table>
<thead>
<tr>
<th>OFFICE-CODE-0450</th>
<th>OFFICE-CITY-0450</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>005</td>
<td>GLASSTER</td>
</tr>
<tr>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>012</td>
<td>CAMBRIDGE</td>
</tr>
</tbody>
</table>

END OF REPORT```
Retrieving data from two or more records

How CA OLQ interprets a set relationship

CA IDMS/DB database records relate to each other through set&$. RB. relationships or through data values. A CA IDMS/DB set links occurrences of one record type with associated occurrences of another record type. One record type is the owner of the set. The other record type is a member of the set. For example, the OFFICE-EMPLOYEE set associates each employee with a particular office. The OFFICE record is the owner and the EMPLOYEE record is the member.

When you retrieve data from two or more tables, you join the tables on a common value. For example, you join the DEPT and EMP tables by equating department IDs: `where dept.deptid=emp.deptid`. Similarly, when you retrieve data from two or more records, you join the records by using a set relationship. For example, you join the OFFICE and EMPLOYEE records with the OFFICE-EMPLOYEE set relationship: `where office-employee`. The figure below illustrates how CA OLQ interprets a set relationship between the OFFICE and EMPLOYEE database records:

![Diagram illustrating set relationship]

The table below shows the data associated with the office and employee records:

<table>
<thead>
<tr>
<th>OFFICE-CODE</th>
<th>EMPID</th>
<th>EMPNAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>002</td>
<td>0075</td>
<td>Jennifer Lanzarotta</td>
</tr>
<tr>
<td>002</td>
<td>3302</td>
<td>Bart Elopolos</td>
</tr>
<tr>
<td>002</td>
<td>3871</td>
<td>Reginald Mahoney</td>
</tr>
<tr>
<td>008</td>
<td>4230</td>
<td>Duc Ho</td>
</tr>
<tr>
<td>008</td>
<td>6264</td>
<td>Daniel Ortega</td>
</tr>
<tr>
<td>008</td>
<td>6348</td>
<td>Edith Jones</td>
</tr>
</tbody>
</table>

Coding the SELECT statement

To retrieve data from two or more CA IDMS/DB records, code the SELECT statement like this:
1. Enter record field names instead of column names following the SELECT keyword. If the same field name appears in more than one record, qualify the fields with the record name: `employee.emp-id-0415`.

2. Enter the record names instead of the table names following the FROM keyword: `office`, `employee`.

3. Enter the set names following the WHERE keyword. Separate set names by AND. You can also include other WHERE criteria. Separate additional WHERE criteria from set names by using AND, also: `where (office-employee and dept-employee) and (dept-id-0410 = 4000)`.

**Example 1 -- Retrieving data from 2 records**

List all employees who work in the Boston office. The SELECT statement shown below selects EMPLOYEE and OFFICE records in the OFFICE-EMPLOYEE set having an office code of BOSTON:

```
select emp-last-name-0415 as &xq.employee name', office-city-0450 as &xq.office' from 
employee, office 
where office-employee and office-city-0450 = &xq.boston' ! display 
```

<table>
<thead>
<tr>
<th>EMPLOYEE NAME</th>
<th>OFFICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANGELO</td>
<td>BOSTON</td>
</tr>
<tr>
<td>BANK</td>
<td>BOSTON</td>
</tr>
<tr>
<td>BLOOMER</td>
<td>BOSTON</td>
</tr>
<tr>
<td>FITZHugh</td>
<td>BOSTON</td>
</tr>
<tr>
<td>FONRAD</td>
<td>BOSTON</td>
</tr>
<tr>
<td>GARDNER</td>
<td>BOSTON</td>
</tr>
<tr>
<td>HENDON</td>
<td>BOSTON</td>
</tr>
<tr>
<td>HUTTON</td>
<td>BOSTON</td>
</tr>
<tr>
<td>JACKSON</td>
<td>BOSTON</td>
</tr>
<tr>
<td>JENSON</td>
<td>BOSTON</td>
</tr>
<tr>
<td>JOHNSON</td>
<td>BOSTON</td>
</tr>
<tr>
<td>KAHALLY</td>
<td>BOSTON</td>
</tr>
<tr>
<td>KIMBALL</td>
<td>BOSTON</td>
</tr>
<tr>
<td>KING</td>
<td>BOSTON</td>
</tr>
</tbody>
</table>

**Example 2 -- Retrieving data from 3 records**

List the department and office location of each employee. The SELECT statement shown below:

```
select dept-id-0410 as department, emp-last-name-0415 as employee, office-city-0450 as office 
from department, employee, office 
where dept-employee and office-employee 
order by dept-id-0410, emp-last-name-0415 ! display 
```

<table>
<thead>
<tr>
<th>DEPARTMENT</th>
<th>EMPLOYEE</th>
<th>OFFICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEPARTMENT</td>
<td>EMPLOYEE</td>
<td>OFFICE</td>
</tr>
</tbody>
</table>
Retrieving data from a record joined to itself

By using a nested structure

Like tables, records can participate in nested structures. For example, employees who are supervisors have employees who are staff members. Likewise, employees who are staff can report to more than one supervisor.

This type of set relationship is called a bill-of-materials structure. The data structure diagram in Sample Tables and Database shows a bill-of-materials structure between the EMPLOYEE and STRUCTURE records:

- One set is MANAGES. It associates supervisors with staff.
- The other set is REPORTS-TO. It associates each employee with one or more supervisors.

The STRUCTURE record exists only to facilitate these set relationships.

Assign alias record names

This figure shows how CA OLQ interprets a bill-of-materials structure relationally, by using alias names for the EMPLOYEE record:

- The SUPERVISOR alias contains occurrences of supervisors. The MANAGES set relates each supervisor to employees who are staff.
- The WORKER alias contains occurrences of staff. The REPORTS-TO set relates each staff member to employees who are supervisors.

CA OLQ uses the concept illustrated in the figure shown under Retrieving data from two or more records (see page 199) to interpret these set relationships:
CA OLQ relates the supervisor IDs to worker record occurrences and relates worker worker IDs to manager record occurrences in two tables.

<table>
<thead>
<tr>
<th>MANAGES</th>
<th>REPORTS TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>MGRID</td>
<td>WORKER</td>
</tr>
<tr>
<td></td>
<td>WORKERID</td>
</tr>
<tr>
<td></td>
<td>MANAGER</td>
</tr>
<tr>
<td>0075</td>
<td>Poznanski</td>
</tr>
<tr>
<td>0075</td>
<td>Achebe</td>
</tr>
</tbody>
</table>

Employee 0075 manages Poznanski and Achebe. Employee 0075 works for Sarem and Romans.

**Coding the SELECT statement**

As with a reflexive table join, the SELECT statement is unique in that you assign alias names to the same record. To code the SELECT statement, follow these steps:

1. Qualify each record field name with an alias record name: `supervisor.emp-last-name-0415`.

2. For each bill-of-materials navigation, assign a unique alias to the record by coding:
   a. The record name
   b. A blank
   c. The alias

   Separate each record and its alias from another with a comma: `employee supervisor, employee worker, structure`.

3. Name the sets that participate in the bill-of-materials following the WHERE clause by coding:
   a. A qualified set name. A qualified set name is the set name followed by a period and the alias record name: `where reports-to.worker`.
   b. An AND logical operator.
c. A second qualified set name.

**Example -- Listing managers and their staff**

Retrieve each project leader and the staff working on the project. The SELECT statement assigns these aliases to the EMPLOYEE record: SUPERVISOR and WORKER. STRUCTURE is the name of the CA IDMS/DB record that facilitates this bill-of-materials data structure.

The alias table names qualify record field names that appear following the SELECT keyword and set names that appear in the WHERE clause:

```sql
select supervisor.emp-last-name-0415 as 'project leader',
       worker.emp-last-name-0415 as 'staff'
from employee supervisor, employee worker, structure
where manages.supervisor and reports-to.worker
order by supervisor.emp-last-name-0415
```

**EMPLOYEE/EMPLOYEE REPORT**

<table>
<thead>
<tr>
<th>PROJECT LEADER</th>
<th>STAFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>BANK</td>
<td>ZEDI</td>
</tr>
<tr>
<td>BANK</td>
<td>JACKSON</td>
</tr>
<tr>
<td>BANK</td>
<td>PENMAN</td>
</tr>
<tr>
<td>BANK</td>
<td>MCDUGALL</td>
</tr>
<tr>
<td>BANK</td>
<td>ANGELO</td>
</tr>
<tr>
<td>CRANE</td>
<td>GARDNER</td>
</tr>
<tr>
<td>CRANE</td>
<td>KAHALLY</td>
</tr>
<tr>
<td>CRANE</td>
<td>KLWELLEN</td>
</tr>
<tr>
<td>CRANE</td>
<td>LIPSICH</td>
</tr>
<tr>
<td>CRANE</td>
<td>KRAAMER</td>
</tr>
<tr>
<td>CRANE</td>
<td>TERNER</td>
</tr>
<tr>
<td>CRANE</td>
<td>FONRAD</td>
</tr>
<tr>
<td>CRANE</td>
<td>FERNDALE</td>
</tr>
</tbody>
</table>

- 1 -

**Retrieving Data From Tables and Records**

**Signing on**

To join information from a table and a database record, sign on to the subschema that contains the database record and sign on to the table. Use a view id in the SIGNON statement to keep each signon active and to qualify records and tables that share the same name:

```sql
signon table emp view emptab
    !signon ss=empss01 view empssc
```

**Comparing view IDs to alias names**

A view ID applies when you sign on to more than one subschema. It qualifies records or tables that have the same name in different subschemas.

An alias applies to records and tables in the SELECT statement. It qualifies fields (columns) and sets that have the same name in different records or tables.

**Coding the SELECT statement**

Once you have signed on to the subschemas, code the SELECT statement like this:
1. Enter column names and/or record field names following the SELECT keyword: **lastname,** **dept-name-0410**.

2. Enter table and/or record names following the FROM keyword. If the record and table share the same name, qualify them with the view id you assigned to the subschema at signon: **emptab.emp, empssc.department**.

3. Equate values in the WHERE clause. If you are retrieving information from a table and a database record, compare a column to a field: **deptid = dept-id-0410**

**Example -- Joining a table and a record**

Join table EMP and database record DEPARTMENT to list all employees and their associated departments by:

1. Signing on to the EMPSS01 subschema in dictionary TSTDICT.
2. Signing on to the EMP table in dictionary ASFDICT.
3. Select the employee's name from table EMP and the department's name from record DEPARTMENT where a table row and record occurrence have the same department ID value.
   The ORDER BY clause instructs CA OLQ to display the rows alphabetically by employees names.

```
signon ss empss01 dictname tstdict view empssc !
signon table emp dictname asfdict view emptab !
select lastname as employee, dept-name-0410 as department
from emp, department where deptid = dept-id-0410 order by lastname ! display
```

<table>
<thead>
<tr>
<th>EMPLOYEE</th>
<th>DEPARTMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANGELO</td>
<td>PUBLIC RELATIONS</td>
</tr>
<tr>
<td>BANK</td>
<td>PUBLIC RELATIONS</td>
</tr>
<tr>
<td>DOUGH</td>
<td>INTERNAL SOFTWARE</td>
</tr>
<tr>
<td>GALLWAY</td>
<td>INTERNAL SOFTWARE</td>
</tr>
<tr>
<td>GARFIELD</td>
<td>INTERNAL SOFTWARE</td>
</tr>
<tr>
<td>GRANGER</td>
<td>INTERNAL SOFTWARE</td>
</tr>
<tr>
<td>HEAROWITZ</td>
<td>INTERNAL SOFTWARE</td>
</tr>
<tr>
<td>HENDON</td>
<td>EXECUTIVE ADMINISTRATION</td>
</tr>
<tr>
<td>JACKSON</td>
<td>PUBLIC RELATIONS</td>
</tr>
<tr>
<td>JACOBI</td>
<td>INTERNAL SOFTWARE</td>
</tr>
<tr>
<td>JENSEN</td>
<td>INTERNAL SOFTWARE</td>
</tr>
<tr>
<td>LITERATA</td>
<td>INTERNAL SOFTWARE</td>
</tr>
<tr>
<td>MCDougall</td>
<td>PUBLIC RELATIONS</td>
</tr>
<tr>
<td>PAPAZEUS</td>
<td>EXECUTIVE ADMINISTRATION</td>
</tr>
</tbody>
</table>

**Test Your Knowledge**

Using the sample tables in Appendix A, code a SELECT statement for each of the queries listed below. num=B.Answers, contains one possible answer. Remember, there can be more than one way to achieve the same result when you use the SELECT statement.

1. List the average salary for employees in the Boston office
2. List all employees in the Springfield office who are programmer/analysts

3. List the average salary of the managers in table DEPT

4. List all employees who are either programmer/analysts, paste-up artists, or a brainstorming manager

Using the data structure diagram in code a SELECT statement for these database record queries:

1. List each employee's job title and salary

2. For each job assigned to more than one employee, list the number of employees assigned to the job and their average salaries

3. List each employee's manager

Sample Tables and Database

**BOSTON Table**

<table>
<thead>
<tr>
<th>EMPID</th>
<th>LASTNAME</th>
<th>HIREDATE</th>
<th>OFFICECODE</th>
<th>TOWN</th>
</tr>
</thead>
<tbody>
<tr>
<td>0120</td>
<td>ANGELO</td>
<td>090879</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0007</td>
<td>BANK</td>
<td>043078</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0069</td>
<td>BLOOMER</td>
<td>050580</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0119</td>
<td>BOWER</td>
<td>121477</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0081</td>
<td>FITZHUGH</td>
<td>091981</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0045</td>
<td>FONRAD</td>
<td>041480</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0053</td>
<td>GARDNER</td>
<td>061581</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0030</td>
<td>HENDON</td>
<td>112173</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0100</td>
<td>HUTTON</td>
<td>090777</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0158</td>
<td>JACKSON</td>
<td>070777</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0011</td>
<td>JENSON</td>
<td>092980</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0051</td>
<td>JOHNSON</td>
<td>032377</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0049</td>
<td>KAHALLY</td>
<td>092979</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0067</td>
<td>KIMBALL</td>
<td>091978</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0106</td>
<td>KING</td>
<td>081680</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0074</td>
<td>KRAAMER</td>
<td>040481</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0127</td>
<td>MCDOUGHALL</td>
<td>060780</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0101</td>
<td>NICEMAN</td>
<td>050680</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0011</td>
<td>ORGRATZI</td>
<td>101980</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0149</td>
<td>PENMAN</td>
<td>090877</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0013</td>
<td>PEOPLES</td>
<td>010281</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0048</td>
<td>TERNER</td>
<td>052682</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0466</td>
<td>ANDALE</td>
<td>061582</td>
<td>002</td>
<td>BOSTON</td>
</tr>
<tr>
<td>0457</td>
<td>ARM</td>
<td>012365</td>
<td>002</td>
<td>BOSTON</td>
</tr>
</tbody>
</table>

**DEPT Table**

<table>
<thead>
<tr>
<th>DEPTID</th>
<th>DEPTNAME</th>
<th>MGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>PERSONNEL</td>
<td>0013</td>
</tr>
<tr>
<td>2000</td>
<td>ACCOUNTING AND PAYROLL</td>
<td>0011</td>
</tr>
<tr>
<td>3100</td>
<td>INTERNAL SOFTWARE</td>
<td>0003</td>
</tr>
</tbody>
</table>
EMP Table

<table>
<thead>
<tr>
<th>EMPID</th>
<th>FIRSTNAME</th>
<th>LASTNAME</th>
<th>DEPTID</th>
<th>MGRID</th>
<th>SALARY</th>
<th>JOBID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>JOHN</td>
<td>RUPEE</td>
<td>6666</td>
<td>0030</td>
<td>76000.00</td>
<td>3001</td>
</tr>
<tr>
<td>0003</td>
<td>JENNIFER</td>
<td>GARFIELD</td>
<td>3100</td>
<td>0030</td>
<td>65000.00</td>
<td>3001</td>
</tr>
<tr>
<td>0007</td>
<td>MONTE</td>
<td>BANK</td>
<td>4000</td>
<td>0030</td>
<td>80000.00</td>
<td>4001</td>
</tr>
<tr>
<td>0019</td>
<td>JULIE</td>
<td>JENSEN</td>
<td>3100</td>
<td>0003</td>
<td>37000.00</td>
<td>3025</td>
</tr>
<tr>
<td>0020</td>
<td>JAMES</td>
<td>JACOBI</td>
<td>3100</td>
<td>0003</td>
<td>55000.00</td>
<td>3011</td>
</tr>
<tr>
<td>0028</td>
<td>PERCY</td>
<td>GRANGER</td>
<td>3100</td>
<td>0003</td>
<td>34500.00</td>
<td>4025</td>
</tr>
<tr>
<td>0029</td>
<td>JAMES</td>
<td>GALLWAY</td>
<td>3100</td>
<td>0003</td>
<td>33000.00</td>
<td>3025</td>
</tr>
<tr>
<td>0120</td>
<td>MICHAEL</td>
<td>ANGELO</td>
<td>4000</td>
<td>0007</td>
<td>18000.00</td>
<td>4051</td>
</tr>
<tr>
<td>0149</td>
<td>LAURA</td>
<td>PENMAN</td>
<td>4000</td>
<td>0007</td>
<td>39000.00</td>
<td>4025</td>
</tr>
<tr>
<td>0158</td>
<td>Jock</td>
<td>JACKSON</td>
<td>4000</td>
<td>0007</td>
<td>34000.00</td>
<td>4025</td>
</tr>
<tr>
<td>0471</td>
<td>THEMIS</td>
<td>PAPAZEUS</td>
<td>6666</td>
<td>0030</td>
<td>90000.00</td>
<td>5001</td>
</tr>
<tr>
<td>0476</td>
<td>Betsy</td>
<td>ZEDI</td>
<td>4000</td>
<td>0007</td>
<td>37000.00</td>
<td>4025</td>
</tr>
</tbody>
</table>

EMPID | PHONE | SSENAME | STARTDATE | BIRTHDATE | CLASS | SEX |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>6173421515</td>
<td>013445656</td>
<td>750223</td>
<td>330219</td>
<td>72</td>
<td>M</td>
</tr>
<tr>
<td>0003</td>
<td>6173321967</td>
<td>021994516</td>
<td>778121</td>
<td>450818</td>
<td>71</td>
<td>F</td>
</tr>
<tr>
<td>0007</td>
<td>6173321933</td>
<td>022446676</td>
<td>780430</td>
<td>500101</td>
<td>72</td>
<td>M</td>
</tr>
<tr>
<td>0019</td>
<td>6172241955</td>
<td>033456718</td>
<td>820929</td>
<td>480730</td>
<td>43</td>
<td>F</td>
</tr>
<tr>
<td>0020</td>
<td>6173348912</td>
<td>018813465</td>
<td>811111</td>
<td>401101</td>
<td>63</td>
<td>M</td>
</tr>
<tr>
<td>0024</td>
<td>6174459191</td>
<td>019809345</td>
<td>801211</td>
<td>551225</td>
<td>21</td>
<td>M</td>
</tr>
<tr>
<td>0027</td>
<td>6173349634</td>
<td>031896154</td>
<td>760808</td>
<td>510329</td>
<td>42</td>
<td>F</td>
</tr>
<tr>
<td>0028</td>
<td>6173341212</td>
<td>011234545</td>
<td>800927</td>
<td>580222</td>
<td>42</td>
<td>M</td>
</tr>
<tr>
<td>0029</td>
<td>6173349155</td>
<td>014567777</td>
<td>811010</td>
<td>471004</td>
<td>42</td>
<td>M</td>
</tr>
<tr>
<td>0030</td>
<td>6175123232</td>
<td>023567831</td>
<td>800909</td>
<td>550430</td>
<td>42</td>
<td>M</td>
</tr>
<tr>
<td>0120</td>
<td>6178871234</td>
<td>153879779</td>
<td>800607</td>
<td>590394</td>
<td>21</td>
<td>F</td>
</tr>
<tr>
<td>0149</td>
<td>6175341199</td>
<td>014593186</td>
<td>779098</td>
<td>440504</td>
<td>33</td>
<td>F</td>
</tr>
<tr>
<td>0474</td>
<td>6174612777</td>
<td>022887770</td>
<td>780907</td>
<td>350304</td>
<td>72</td>
<td>N</td>
</tr>
<tr>
<td>0476</td>
<td>6174319909</td>
<td>010004560</td>
<td>760223</td>
<td>401229</td>
<td>33</td>
<td>F</td>
</tr>
</tbody>
</table>

JOBCLASS Table

<table>
<thead>
<tr>
<th>CLASS</th>
<th>MINSALARY</th>
<th>MAXSALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>12000.00</td>
<td>18000.00</td>
</tr>
<tr>
<td>21</td>
<td>18000.00</td>
<td>20000.00</td>
</tr>
<tr>
<td>33</td>
<td>37000.00</td>
<td>39000.00</td>
</tr>
<tr>
<td>42</td>
<td>33000.00</td>
<td>35000.00</td>
</tr>
<tr>
<td>43</td>
<td>35500.00</td>
<td>40000.00</td>
</tr>
</tbody>
</table>
## JOBLIST Table

<table>
<thead>
<tr>
<th>JOBID</th>
<th>TITLE</th>
<th>EMPID</th>
<th>CLASS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1001</td>
<td>MGR PERSONNEL</td>
<td>0013</td>
<td>72</td>
</tr>
<tr>
<td>1023</td>
<td>RECRUITER/INTERVWR</td>
<td>0001</td>
<td>43</td>
</tr>
<tr>
<td>1051</td>
<td>PERSONNEL CLERK</td>
<td>0051</td>
<td>11</td>
</tr>
<tr>
<td>1051</td>
<td>PERSONNEL CLERK</td>
<td>0081</td>
<td>11</td>
</tr>
<tr>
<td>2001</td>
<td>MGR ACCTNG/PAYROLL</td>
<td>0011</td>
<td>72</td>
</tr>
<tr>
<td>2023</td>
<td>ACCOUNTANT</td>
<td>0067</td>
<td>62</td>
</tr>
<tr>
<td>2025</td>
<td>FINANCIAL ANALYST</td>
<td>0100</td>
<td>62</td>
</tr>
<tr>
<td>2051</td>
<td>AP CLERK</td>
<td>0101</td>
<td>12</td>
</tr>
<tr>
<td>2053</td>
<td>AR CLERK</td>
<td>0106</td>
<td>12</td>
</tr>
<tr>
<td>2055</td>
<td>PAYROLL CLERK</td>
<td>0069</td>
<td>13</td>
</tr>
<tr>
<td>3001</td>
<td>MGR INTERNL SOFTWARE</td>
<td>0003</td>
<td>71</td>
</tr>
<tr>
<td>3003</td>
<td>MGR COMPUTER OPS</td>
<td>0004</td>
<td>71</td>
</tr>
<tr>
<td>3011</td>
<td>DATABASE ADMIN.</td>
<td>0020</td>
<td>63</td>
</tr>
<tr>
<td>3023</td>
<td>SYSTEMS PROGRAMMER</td>
<td>0016</td>
<td>52</td>
</tr>
<tr>
<td>3025</td>
<td>PROGRAMMER/ANALYST</td>
<td>0019</td>
<td>43</td>
</tr>
<tr>
<td>3025</td>
<td>PROGRAMMER/ANALYST</td>
<td>0023</td>
<td>44</td>
</tr>
<tr>
<td>3025</td>
<td>PROGRAMMER/ANALIST</td>
<td>0024</td>
<td>42</td>
</tr>
<tr>
<td>3025</td>
<td>PROGRAMMER/ANALYST</td>
<td>0027</td>
<td>42</td>
</tr>
<tr>
<td>3025</td>
<td>PROGRAMMER/ANALYST</td>
<td>0028</td>
<td>42</td>
</tr>
<tr>
<td>3025</td>
<td>PROGRAMMER/ANALYST</td>
<td>0029</td>
<td>42</td>
</tr>
<tr>
<td>3027</td>
<td>PROGRAMMER TRAINEE</td>
<td>0021</td>
<td>21</td>
</tr>
<tr>
<td>3027</td>
<td>PROGRAMMER TRAINEE</td>
<td>0023</td>
<td>43</td>
</tr>
<tr>
<td>3029</td>
<td>COMPUTER OPERATOR</td>
<td>0031</td>
<td>21</td>
</tr>
<tr>
<td>3029</td>
<td>COMPUTER OPERATOR</td>
<td>0032</td>
<td>22</td>
</tr>
<tr>
<td>3029</td>
<td>COMPUTER OPERATOR</td>
<td>0049</td>
<td>21</td>
</tr>
<tr>
<td>3031</td>
<td>DOCUMENTATION SPEC</td>
<td>0035</td>
<td>43</td>
</tr>
<tr>
<td>3051</td>
<td>DATA ENTRY CLERK</td>
<td>0045</td>
<td>13</td>
</tr>
<tr>
<td>3051</td>
<td>DATA ENTRY CLERK</td>
<td>0048</td>
<td>11</td>
</tr>
<tr>
<td>3051</td>
<td>DATA ENTRY CLERK</td>
<td>0053</td>
<td>12</td>
</tr>
<tr>
<td>3051</td>
<td>DATA ENTRY CLERK</td>
<td>0074</td>
<td>12</td>
</tr>
<tr>
<td>4001</td>
<td>MGR PUBLIC RELATIONS</td>
<td>0007</td>
<td>72</td>
</tr>
<tr>
<td>4021</td>
<td>SPORTS CONSULTANT</td>
<td>0158</td>
<td>42</td>
</tr>
<tr>
<td>4023</td>
<td>PHOTOGRAPHER</td>
<td>0999</td>
<td>33</td>
</tr>
<tr>
<td>4025</td>
<td>PR WRITER</td>
<td>0149</td>
<td>33</td>
</tr>
<tr>
<td>4025</td>
<td>PR WRITER</td>
<td>0476</td>
<td>33</td>
</tr>
<tr>
<td>4051</td>
<td>ILLUSTRATOR</td>
<td>0120</td>
<td>21</td>
</tr>
<tr>
<td>4053</td>
<td>PASTE-UP ARTIST</td>
<td>0120</td>
<td>21</td>
</tr>
<tr>
<td>4053</td>
<td>PASTE-UP ARTIST</td>
<td>0127</td>
<td>21</td>
</tr>
<tr>
<td>5001</td>
<td>MGR BRAINSTORMING</td>
<td>0015</td>
<td>72</td>
</tr>
<tr>
<td>5001</td>
<td>MGR BRAINSTORMING</td>
<td>0471</td>
<td>72</td>
</tr>
<tr>
<td>5003</td>
<td>MGR THERMOREGULATION</td>
<td>0349</td>
<td>72</td>
</tr>
<tr>
<td>5005</td>
<td>MGR BLUE SKIES</td>
<td>0321</td>
<td>71</td>
</tr>
<tr>
<td>5021</td>
<td>RAINMAKER</td>
<td>0301</td>
<td>53</td>
</tr>
<tr>
<td>5023</td>
<td>RAINDANCE CONSULTANT</td>
<td>0334</td>
<td>43</td>
</tr>
<tr>
<td>5025</td>
<td>SNOWBLOWER</td>
<td>0341</td>
<td>51</td>
</tr>
<tr>
<td>5025</td>
<td>SNOWBLOWER</td>
<td>0466</td>
<td>51</td>
</tr>
<tr>
<td>5027</td>
<td>KEEPER OF THE WINDS</td>
<td>0467</td>
<td>51</td>
</tr>
<tr>
<td>5029</td>
<td>STURM/DRANG ADMIN</td>
<td>0457</td>
<td>53</td>
</tr>
<tr>
<td>5029</td>
<td>STURM/DRANG ADMIN</td>
<td>0458</td>
<td>53</td>
</tr>
<tr>
<td>5031</td>
<td>KEEPER OF BALLOONS</td>
<td>0329</td>
<td>52</td>
</tr>
<tr>
<td>5033</td>
<td>WINTERIZER</td>
<td>0355</td>
<td>51</td>
</tr>
<tr>
<td>5033</td>
<td>WINTERIZER</td>
<td>0469</td>
<td>51</td>
</tr>
<tr>
<td>5035</td>
<td>HUMIDITY CONTROL CLK</td>
<td>0479</td>
<td>43</td>
</tr>
<tr>
<td>5037</td>
<td>SUNSHINE SUPERVISOR</td>
<td>0321</td>
<td>53</td>
</tr>
<tr>
<td>5037</td>
<td>SUNSHINE SUPERVISOR</td>
<td>0366</td>
<td>51</td>
</tr>
<tr>
<td>5039</td>
<td>CUMULUS CARETAKER</td>
<td>0371</td>
<td>53</td>
</tr>
</tbody>
</table>
### SPRINGFIELD Table

<table>
<thead>
<tr>
<th>EMPID</th>
<th>LASTNAME</th>
<th>STARTYEAR</th>
<th>OFFICECODE</th>
<th>CITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>0004</td>
<td>CRANE</td>
<td>051477</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0024</td>
<td>DOUGH</td>
<td>080876</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0032</td>
<td>FERNDALE</td>
<td>090979</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0329</td>
<td>FINN</td>
<td>061679</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0829</td>
<td>GALLWAY</td>
<td>101081</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0003</td>
<td>GARFIELD</td>
<td>012177</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0028</td>
<td>GRANGER</td>
<td>052780</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0027</td>
<td>HEAROWITZ</td>
<td>090981</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0020</td>
<td>JACOBI</td>
<td>111181</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0019</td>
<td>JENSEN</td>
<td>092982</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0016</td>
<td>KLWELLEN</td>
<td>018678</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0031</td>
<td>LIPSICH</td>
<td>042981</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0035</td>
<td>LITERATA</td>
<td>090980</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0023</td>
<td>O’HEARN</td>
<td>022375</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0001</td>
<td>RUPEE</td>
<td>022375</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0021</td>
<td>TYRO</td>
<td>122180</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0472</td>
<td>WILDER</td>
<td>071679</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0476</td>
<td>ZEDI</td>
<td>022376</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0007</td>
<td>BANK</td>
<td>043078</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0004</td>
<td>CRANE</td>
<td>051477</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0024</td>
<td>DOUGH</td>
<td>080876</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0032</td>
<td>FERNDALE</td>
<td>090979</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0329</td>
<td>FINN</td>
<td>061679</td>
<td>008</td>
<td>WESTON</td>
</tr>
</tbody>
</table>

### WESTON Table

<table>
<thead>
<tr>
<th>EMPID</th>
<th>LASTNAME</th>
<th>STARTDATE</th>
<th>OFFICECODE</th>
<th>CITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>0466</td>
<td>ANDALE</td>
<td>061582</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0457</td>
<td>ARM</td>
<td>012365</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0467</td>
<td>BREEZE</td>
<td>022980</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0469</td>
<td>KASPAR</td>
<td>091476</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0341</td>
<td>MUNYON</td>
<td>031482</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0471</td>
<td>PAPAZEUS</td>
<td>090778</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0355</td>
<td>TIME</td>
<td>050675</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0458</td>
<td>WAGNER</td>
<td>070883</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0349</td>
<td>WILCO</td>
<td>111179</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0119</td>
<td>BOWER</td>
<td>121477</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0081</td>
<td>FITZHUGH</td>
<td>091981</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0004</td>
<td>CRANE</td>
<td>051477</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0024</td>
<td>DOUGH</td>
<td>080876</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0032</td>
<td>FERNDALE</td>
<td>090979</td>
<td>008</td>
<td>WESTON</td>
</tr>
<tr>
<td>0329</td>
<td>FINN</td>
<td>061679</td>
<td>008</td>
<td>WESTON</td>
</tr>
</tbody>
</table>
EMPLYEO Database Data Structure Diagram

Answers

Chapter 2 Answers

Query 1

List the salary range for each job class:

```
select class, maxsalary - minsalary as salary_range from jobclass
```

CLASS REPORT

mm/dd/yy

<table>
<thead>
<tr>
<th>CLASS</th>
<th>SALARY RANGE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

10-Jan-2018
Query 2

List all employees who have both:

- Manager with ID 0007 or 0003
- Job that begins with the number 3

```
select lastname, mgrid, jobid from emp where mgrid in (0003, 0007) and jobid like &q.3___'
```

```
EMP REPORT
mm/dd/yy

LASTNAME    MGRID    JOBID
JENSEN       0003    3025
JACOBI       0003    3011
TYRO         0003    3027
DOUGH        0003    3025
HEAROWITZ    0003    3025
GRANGER      0003    3025
GALLWAY      0003    3025
LITERATA     0003    3031

END OF REPORT
```

Query 3

Identify all employees whose sex code was entered incorrectly (that is, is not M or F):

```
select firstname, lastname, sex from emp where sex not in (&q.m', &q.f')
```

```
EMP REPORT
mm/dd/yy

FIRSTNAME    LASTNAME    SEX
THEMIS       PAPAZEUS    N

END OF REPORT
```

Query 4

List the number of years employees have from January 1, 1988 until they reach retirement age (65):

```
select firstname, lastname, birthdate, (65 - next-int-eqlo(datedif(880101,birthdate)/365)) as &q.years until retirement' from emp
```

```
EMP REPORT
mm/dd/yy

FIRSTNAME    LASTNAME    BIRTHDATE    YEARS UNTIL RETIREMENT
JOHN         RUPEE       330219       11
JENNIFER     GARFIELD    450818       23

END OF REPORT
```
CA IDMS - 19.0

Chapter 3 Answers

Query 1

For each manager in table EMP, determine the number of staff reporting to the manager and the average salary of the staff members:

```
select mgrid, count(*) as number of staff, avg(salary) as average salary
from emp
group by mgrid
```

EMP REPORT

<table>
<thead>
<tr>
<th>MGRID</th>
<th>NUMBER OF STAFF</th>
<th>AVERAGE SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>0003</td>
<td>8</td>
<td>35375.00</td>
</tr>
<tr>
<td>0007</td>
<td>5</td>
<td>29200.00</td>
</tr>
<tr>
<td>0030</td>
<td>6</td>
<td>106833.33</td>
</tr>
</tbody>
</table>

END OF REPORT

Query 2

List the number of jobs greater than or equal to 3 that are assigned to classes 10 and 50. Display the report in order by the number of jobs:

```
select count(*) as number of jobs, class
from joblist
where class between 10 and 50
group by class
having count(*) >= 3
order by 1
```

JOBLIST REPORT

<table>
<thead>
<tr>
<th>NUMBER OF JOBS</th>
<th>CLASS</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>33</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>5</td>
<td>42</td>
</tr>
<tr>
<td>6</td>
<td>21</td>
</tr>
<tr>
<td>6</td>
<td>43</td>
</tr>
</tbody>
</table>

END OF REPORT

Query 3

For departments 6666, 4000, and 3100, list the minimum and maximum salaries within the department, provided the average departmental salary in greater than $37,800:

```
select deptid, min(salary) as minimum salary, max(salary) as maximum salary
from emp
where deptid in (3100, 4000, 6666)
group by deptid
having avg(salary) > 37800
```
Query 4

List information on employees earning less than the average salary:

select lastname, salary from emp where salary < (select avg(salary) from emp) !
display

EMP REPORT
mm/dd/yy
LASTNAME                SALARY
JENSEN                   37000.00
JACOBI                  55000.00
TYRO                     20000.00
DOUGH                   33000.00
HEAROWITZ               33000.00
GRANGER                 34500.00
GALLWAY                 33000.00
LITERATA                37500.00
ANGELO                  18000.00
MCDOUGALL              18000.00
PENMAN                 39000.00
JACKSON                 34000.00
ZEDI                   37000.00

Chapter 4 Answers

Table Query 1

List the average salary for employees in the Boston office:

select count(*) as &xq.number of boston employees', avg(salary) as &xq.average salary'
from emp, boston where emp.empid = boston.empid

EMPLOYEE/BOSTON REPORT
mm/dd/yy
NUMBER OF BOSTON EMPLOYEES   AVERAGE SALARY
7                             74142.85

Table Query 2

List all employees in the Springfield office who are programmer/analysts:

select emp.lastname, joblist.title, springfield.city from emp, joblist, springfield
where emp.empid = springfield.empid and emp.empid = joblist.empid and joblist.title =
&xq.programmer/anaylsts'

JOBLIST/EMPLOYEE/SPRINGFIELD REPORT
mm/dd/yy
LASTNAME                 TITLE             CITY
Table Query 3

List the average salary of the managers in table DEPT:

```
select avg(salary) as 'average manager salary' from emp, dept where emp.empid = dept.mgrid
```

EMP/DEPARTMENT REPORT

<table>
<thead>
<tr>
<th>AVERAGE MANAGER SALARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>128333.33</td>
</tr>
</tbody>
</table>

END OF REPORT

Table Query 4

List all employees who are either programmer/analysts, paste-up artists, or a brainstorming manager:

```
select emp.lastname, joblist.title from emp, joblist where joblist.title in ('programmer/analyst', 'paste-up artist', 'mgr brainstorming') and joblist.empid = emp.empid
```

JOBLIST/EMPLOYEE REPORT

<table>
<thead>
<tr>
<th>LASTNAME</th>
<th>TITLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOUGH</td>
<td>PROGRAMMER/ANALYST</td>
</tr>
<tr>
<td>HEAROWITZ</td>
<td>PROGRAMMER/ANALYST</td>
</tr>
<tr>
<td>GRANGER</td>
<td>PROGRAMMER/ANALYST</td>
</tr>
<tr>
<td>GALLWAY</td>
<td>PROGRAMMER/ANALYST</td>
</tr>
<tr>
<td>ANGELO</td>
<td>PASTE-UP ARTIST</td>
</tr>
<tr>
<td>MCDOUGALL</td>
<td>PASTE-UP ARTIST</td>
</tr>
<tr>
<td>PAPAZEU</td>
<td>MGR BRAINSTORMING</td>
</tr>
</tbody>
</table>

END OF REPORT

Record Query 1

List each employee's job title and salary:

```
select employee.emp-last-name-0415, job.job-title-0440, emposition.salary-amount-0420 from employee, emposition, job where emp-emposition and job-emposition
```

JOB/EMPOSITION/EMPLOYEE REPORT

<table>
<thead>
<tr>
<th>EMP-LAST-NAME-0415</th>
<th>TITLE-0440</th>
<th>SALARY-AMOUNT-0420</th>
</tr>
</thead>
<tbody>
<tr>
<td>O'HEARN</td>
<td>PROGRAMMER TRAINEE</td>
<td>38000.00</td>
</tr>
<tr>
<td>TYRO</td>
<td>PROGRAMMER TRAINEE</td>
<td>20000.00</td>
</tr>
<tr>
<td>WILCO</td>
<td>MGR THERMOREGULATION</td>
<td>80000.00</td>
</tr>
<tr>
<td>GARFIELD</td>
<td>MGR INTERNAL SOFTWARE</td>
<td>65000.00</td>
</tr>
<tr>
<td>RUPEE</td>
<td>MGR INTERNAL SOFTWARE</td>
<td>76000.00</td>
</tr>
<tr>
<td>JOHNSON</td>
<td>PERSONNEL CLERK</td>
<td>135000.00</td>
</tr>
<tr>
<td>FITZHUGH</td>
<td>PERSONNEL CLERK</td>
<td>130000.00</td>
</tr>
<tr>
<td>BLOOMER</td>
<td>PAYROLL CLERK</td>
<td>150000.00</td>
</tr>
<tr>
<td>ARM</td>
<td>STURM/DRANG ADMIN</td>
<td>46000.00</td>
</tr>
<tr>
<td>WAGNER</td>
<td>STURM/DRANG ADMIN</td>
<td>47000.00</td>
</tr>
<tr>
<td>ORGRATZI</td>
<td>RECRUITER/INTERVWR</td>
<td>390000.00</td>
</tr>
<tr>
<td>BANK</td>
<td>MGR PUBLIC RELATIONS</td>
<td>800000.00</td>
</tr>
</tbody>
</table>
Record Query 2

For each job assigned to more than one employee, list the number of employees assigned to the job and their average salaries:

```sql
select job.title-0440, count(*) as &xq.number of jobs', avg(emposition.salary-amount-0420) as &xq.average salary' from employee, emposition, job
where emp-emposition and
job-emposition
group by job.title-0440
having count(*) > 1
```

<table>
<thead>
<tr>
<th>JOB/EMPOSITION REPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>mm/dd/yy</td>
</tr>
<tr>
<td>TITLE-0440</td>
</tr>
<tr>
<td>-----------------------</td>
</tr>
<tr>
<td>COMPUTER OPERATOR</td>
</tr>
<tr>
<td>DATA ENTRY CLERK</td>
</tr>
<tr>
<td>DATABASE ADMIN.</td>
</tr>
<tr>
<td>DOCUMENTATION SPEC</td>
</tr>
<tr>
<td>MGR BRAINSTORMING</td>
</tr>
<tr>
<td>MGR COMPUTER OPS</td>
</tr>
<tr>
<td>MGR INTERNL SOFTWARE</td>
</tr>
<tr>
<td>PASTE-UP ARTIST</td>
</tr>
<tr>
<td>PERSONNEL CLERK</td>
</tr>
<tr>
<td>PR WRITER</td>
</tr>
<tr>
<td>PROGRAMMER TRAINEE</td>
</tr>
<tr>
<td>PROGRAMMER/ANALYST</td>
</tr>
<tr>
<td>RAINMAKER</td>
</tr>
</tbody>
</table>

Record Query 3

List each employee's manager:

```sql
select worker.last-name-0415 as &xq.staff', supervisor.last-name-0415 as &xq.project leader' from employee worker, employee supervisor, structure
where reports-to.worker and manages.supervisor
order by worker.emp-last-name-0415
```

<table>
<thead>
<tr>
<th>EMPLOYEE/EMPLOYEE REPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>mm/dd/yy</td>
</tr>
<tr>
<td>STAFF</td>
</tr>
<tr>
<td>ANDALE</td>
</tr>
<tr>
<td>ANGELO</td>
</tr>
<tr>
<td>ARM</td>
</tr>
<tr>
<td>BANK</td>
</tr>
<tr>
<td>BLOOMER</td>
</tr>
<tr>
<td>BOWER</td>
</tr>
<tr>
<td>BREEZE</td>
</tr>
<tr>
<td>CLOTH</td>
</tr>
<tr>
<td>CLOUD</td>
</tr>
<tr>
<td>CLOUD</td>
</tr>
<tr>
<td>CRANE</td>
</tr>
<tr>
<td>CROW</td>
</tr>
</tbody>
</table>
Chapter 5 Answers

Query 1

List the jobs for which employees earn more than $65,000:

```
select * from joblist where exists (select * from emp where emp.jobid=joblist.jobid and salary > 65000)
```

<table>
<thead>
<tr>
<th>EMPID</th>
<th>CLASS</th>
<th>JOBID</th>
<th>TITLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0003</td>
<td>71</td>
<td>3001</td>
<td>MGR INTERNL SOFTWARE</td>
</tr>
<tr>
<td>0007</td>
<td>72</td>
<td>4001</td>
<td>MGR PUBLIC RELATIONS</td>
</tr>
<tr>
<td>0015</td>
<td>72</td>
<td>5001</td>
<td>MGR BRAINSTORMING</td>
</tr>
<tr>
<td>0471</td>
<td>72</td>
<td>5001</td>
<td>MGR BRAINSTORMING</td>
</tr>
<tr>
<td>0030</td>
<td>93</td>
<td>9001</td>
<td>PRESIDENT</td>
</tr>
<tr>
<td>0472</td>
<td>81</td>
<td>9005</td>
<td>DIR CORP CONFUSION</td>
</tr>
</tbody>
</table>

Query 2

List openings for jobs that command salaries in the range $35,000 to $40,000:

```
select class, jobid, title from joblist where not exists (select * from emp where emp.jobid=joblist.jobid) and exists (select * from jobclass where jobclass.class=joblist.class and minsalary >= 35000 and maxsalary <= 40000)
```

<table>
<thead>
<tr>
<th>CLASS</th>
<th>JOBID</th>
<th>TITLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>43</td>
<td>1023</td>
<td>RECRUITER/INTERVWR</td>
</tr>
<tr>
<td>33</td>
<td>4023</td>
<td>PHOTOGRAPHER</td>
</tr>
<tr>
<td>43</td>
<td>5023</td>
<td>RAINDANCE CONSULTANT</td>
</tr>
<tr>
<td>43</td>
<td>5035</td>
<td>HUMIDITY CONTROL CLK</td>
</tr>
</tbody>
</table>

Query 3

List all employees in the Springfield office who have this job title: 'Programmer/Analyst':

```
select * from springfield where exists (select * from employee where employee.empid = springfield.empid and exists (select * from joblist where employee.empid = joblist.empid and joblist.title = 'Programmer/Analyst'))
```

<table>
<thead>
<tr>
<th>EMPID</th>
<th>LASTNAME</th>
<th>STARTYEAR</th>
<th>OFFICECODE</th>
<th>CITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>0024</td>
<td>DOUGH</td>
<td>080876</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0029</td>
<td>GALLWAY</td>
<td>101081</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0028</td>
<td>GRANGER</td>
<td>052780</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
<tr>
<td>0027</td>
<td>HEAROWITZ</td>
<td>090981</td>
<td>001</td>
<td>SPRINGFIELD</td>
</tr>
</tbody>
</table>

Query 4
List all departments that have no associated employee:

```
select * from dept
where not exists (select * from emp where emp.deptid = dept.deptid)
```

```
DEPT REPORT

DEPTID      DEPTNAME          MGRID
1000        PERSONNEL        0013
2000        ACCOUNTING AND PAYROLL 0011
3200        COMPUTER OPERATIONS 0004
5100        BRAINSTORMING    0015
5200        THERMOREGULATION 0349
5300        BLUE SKIES       0321

END OF REPORT
```

Chapter 6 Answers

**Query 1**

List employees in the Boston and Springfield offices, including commuters. Display the report in order of last name:

```
select * from boston
union
select * from springfield
order by 2
```

```
BOSTON REPORT

EMPID      LASTNAME     HIREDATE    OFFICECODE  TOWN
0466      ANDALE       061582  002   BOSTON
0120      ANGELO       090879  002   BOSTON
0457      ARM          012365  002   BOSTON
0007      BANK         043078  001   SPRINGFIELD
0007      BANK         043078  002   BOSTON
0069      BLOOMER      050580  001   SPRINGFIELD
0069      BLOOMER      050580  002   BOSTON
0119      BOWER        121477  002   BOSTON
0467      BREEZE       022980  001   SPRINGFIELD
0004      CRANE        051477  001   SPRINGFIELD
0024      DOUGH        080876  001   SPRINGFIELD
0032      FERNDALE     090979  001   SPRINGFIELD
0329      FINN         061679  001   SPRINGFIELD

END OF REPORT
```

**Query 2**

List the average salaries of employees in the Boston and Springfield offices:

```
select avg(salary) as average salary', city as city
from boston, emp
where boston.empid = emp.empid
group by city
union
select avg(salary), city
from springfield, emp
where springfield.empid = emp.empid
group by city
```

```
BOSTON/EMP REPORT

AVERAGE SALARY   CITY
49500.00          SPRINGFIELD
74142.85          BOSTON

END OF REPORT
```
Query 3

List all the employees in the Springfield and Weston offices whose jobid is 3001 or 5001:

```
select lastname, city from weston, emp
where weston.empid=emp.empid and jobid in (3001, 5001) union
select lastname, city from springfield, emp
where springfield.empid=emp.empid and jobid in (3001, 5001)
```

```
EMP/WESTON REPORT
LASTNAME | JOBID | CITY
GARFIELD | 3001  | SPRINGFIELD
PAPAZEUS | 5001  | WESTON
RUPEE   | 3001  | SPRINGFIELD
END OF REPORT
```

Final Query

Contents
- Glossary (see page 217)

For each job, list the number of employees greater than 1 who each the minimum salary for the job's class. Display the report in order of job title:

```
select joblist.title, count(*) as 'number of employees', avg(salary) as salary
from joblist, emp
where exists (select * from jobclass
where emp.class=jobclass.class and emp.salary = jobclass.minsalary) and joblist.empid=emp.empid
group by title
having count(*) > 1 order by title
```

```
JOBLIST/EMP REPORT
TITLE | NUMBER OF EMPLOYEES | SALARY
PASTE-UP ARTIST | 2 | 18000.00
PROGRAMMER/ANALYST | 3 | 33000.00
END OF REPORT
```

Glossary

Additional selection criteria

Logical expressions, logical record keywords, and criteria expressions for subscripted fields that you use to tell CA OLQ which rows of data to retrieve for your report. Additional selection criteria are specified in the Additional selection criteria field of the Selection criteria screen.

Aggregate function

A function that performs a predefined operation on a group of report rows. Examples of aggregate functions are: average, high value, low value, count, and total.

ASF (Automatic System Facility)
A tool in CA-IDMS/DB used to create and manage tables. Once you have created a table using CA OLQ, you can use ASF to modify the table definition.

ASF dictionary

An alternate data dictionary used by ASF. You must be using the ASF dictionary when you are creating data tables.

ASF table

A presentation of data as a series of rows and columns from a table associated with the IDMSR schema.

Batch

Batch processing means that the user doesn’t have to interact with the computer system in order to perform a function. Usually, a batch job is set up in advance (such as when you fill out your Batch Processing screen). Once the job has started running, you cannot intervene except to suspend execution.

Built-in function

A function that performs a predefined string, arithmetic, date/time, or trigonometric calculation on your report rows. Examples of built-in functions are: substring, Gregorian date, cosine, and square root.

Code table

A table defined in the data dictionary that contains corresponding pairs of values. One column in the table lists internal code values that are used to efficiently store the data in the database. The other column in the table lists external values that are used in programs or reports.

For example, a STATE code table could represent the STATE column as 01 in the internal (encoded) expression and Alabama in the external (decoded) expression.

Column

A vertical division in a table. A column represents a category of information. For example, employee last name.

Column alignment

An editing feature that determines how report data columns align under the column headers. Options are left, right, and center.

Column header

A header at the top of each column of report data.

COMPUTE statement
A CA OLQ syntax statement used to perform calculations in the menu facility. Any time you specify a built-in or aggregate function, CA OLQ creates a COMPUTE statement. You can also provide your own COMPUTE statement.

Current report

The report you’re working on in an active CA OLQ session. If you retrieve a saved report, CA OLQ clears out the current report.

Data dictionary

The storage facility used by CA products as a central source for data definitions, modules, and run-time information. Qfile definitions are stored in the data dictionary.

Database view

Another term for subschema.

Destination

When you print a report, you specify an output destination where the report is to be printed. Usually, the destination is a file associated with a printer.

Display sequence

A numeric sequence listed on CA OLQ report formatting screens. The numbers in the display sequence correspond to the order in which report columns are displayed.

Element

The smallest significant unit of data in a CA-IDMS/DB database. Record elements correspond to columns in a table. For example, DEPT-ID-0410 is an element.

External picture

A code that defines the way your column value is formatted in your report. The external picture is used to add punctuation (for example, commas) and special characters (for example, dollar signs) to your column data display.

Group by all

A report total including all rows in your report. Group by all means the same thing as report total.

Group computation

A calculation that CA OLQ performs on a report group.

Group field

A report column whose value is used to divide your report rows into groups. For example, you could list all of the company's employees grouped according to which department they work in. In this case, DEPARTMENT NAME is the group field.
Input file
A file that contains input into a batch program.

Integrated Data Dictionary (IDD)
The CA product used to access definitions stored in the data dictionary.

Interactive
A way of performing a function in which the computer system requires the user to provide input and then responds to that input. An example would be CA OLQ menu mode. Another term to describe interactive processing is online.

Interrupt count
The maximum number of records CA OLQ will retrieve when building a report. If the number of records that meet the selection criteria for that report exceeds the interrupt count, CA OLQ suspends data retrieval and issues a message asking you if you want to continue to retrieve records.

Job control language
A language used to define the special requirements of your batch program to the system. Job control language (JCL) statements name input and output files, the name of your program, and your output destination.

Join
A relational operation through which two or more tables are combined. Tables are joined based on columns that the tables have in common.

Join criteria
A logical expression that compares like columns in two or more tables.

Output destination
Any type of device that receives the report that you have created as a result of your batch job. Output destinations can be a printer, a terminal, or a log.

Output file
A file that contains the results of your batch program.

Page header
A title at the top of each page of your report.

Page footer
A title at the bottom of each page of your report.

Project
A relational operation through which only particular columns of a table are accessed.

Qfile definition

The CA OLQ syntax statements stored in the data dictionary when you create your qfile.

Record

A group of related elements. Records correspond to rows in a table. For example, DEPT-NAME-0410, DEPT-ID-0410, and DEPT-HEAD-ID-0410 are all grouped into the DEPARTMENT record.

Report group

A set of report rows such that each row contains the same value of the group field. For example, the personnel department is a report group with DEPARTMENT NAME as the group field. Each row in this group contains Personnel in the DEPARTMENT NAME field.

Report subtotal

A computation applied to a report group. For example, if you grouped your employees by department, you could create report subtotals that compute the average salary in each department.

Report total

A computation that includes all of the rows in your report. For example, you could compute the total sales revenue earned by all of your employees. Note that a report total does not have to be a sum. It can also be an average, a counter, a high value, or a low value.

Row

A horizontal row in a table. A row represents one data occurrence. For example, information on each employee.

Retention period

The number of days your saved report file is kept in your directory. After the retention period has expired, the report file is deleted.

Saved report

A report file maintained in your user directory that contains a copy of a current CA OLQ report.

Select

A relational operation through which only particular rows of a table are accessed.

Selection criteria

Logical expressions that you use to tell CA OLQ which rows of data to retrieve for your report. Selection criteria are specified in the Selection criteria field of the Column Select screen.

Separator character
A character used to separate group computations from the rest of the report.

Sort

A way to order report rows. CA OLQ sorts the rows in your report based on the value of the sort field that you specify. Rows can be sorted in ascending or descending order.

Sparse/Full option

A CA OLQ editing feature that determines how column values that repeat in consecutive rows are displayed:

- **Sparse** displays only the first of a repeating set of column values.
- **Full** displays all occurrences of the repeating value.

SQL table

A presentation of data as a series of rows and columns from a table associated with an SQL schema.

Subschema

A view of the database that contains a subset of the records, elements, sets, and areas that make up the entire database. A subschema usually views data that is functionally related.

For example, the personnel department uses a subschema that views employee information such as salary, date of hire, and personal information. All of the information is held in the same database, but the personnel department views only the information that it needs.

Summary computation

Another term for group computation.

Summary line

A report line that displays a group computation.