PeopleTools 8.12 PeopleSoft Query PeopleBook

PeopleTools 8.12 PeopleSoft Query PeopleBookPeopleTools 8.12 PeopleSoft Query PeopleBook

SKU MTPQr8SP1B 1200

PeopleBooks Contributors: Teams from PeopleSoft Product Documentation and Development.

Contents

About This PeopleBook

Chapter 1

Introduction to PeopleSoft Query

Chapter 2

Setting up Query Security

Chapter 3

Creating and Running Simple Queries

Chapter 4

Specifying Query Selection Criteria

Chapter 5

Advanced Query Options

Chapter 6

Running a Query on the Web

Chapter 6

Using Query Designer on the Web

Index

About This PeopleBook

This book discusses PeopleSoft Query, the PeopleTool you use to design database queries. Its chapters explain the basic concepts of selecting data, designing queries, and sending query results to other reporting tools.

This book is written for decision support team members who want to design database queries. To take full advantage of the information covered in this book, it is recommended that you have a basic understanding of how to use PeopleSoft applications. You should also be comfortable using Microsoft® Windows.

This book assumes that you are familiar with PeopleTools. It also assumes a basic familiarity with relational database concepts and SQL.

Introduction to PeopleSoft Query explains how to run a predefined query, and gives an overview of the query creation process.

Setting up Query Security explains the steps involved in giving users access to PeopleSoft Query functionality. It explains access groups, which are a means of controlling users' rights to query the tables in a database. This chapter is intended primarily for system administrators.

Creating and Running Simple Queries explains the basic features of PeopleSoft Query and how to use them to define a database query.

Specifying Query Selection Criteria explains how to add selection criteria to your queries, in order to limit your results.

Advanced Query Options explains how to create and use aggregate functions, joins, and subqueries.

Running a Query on the Web explains how you can run predefined queries from PeopleSoft Internet Architecture.

Using Query Designer on the Web describes how to use the new, web-based Query Designer to create queries from your browser.

Before You Begin

To benefit fully from the information covered in this book, you need to have a basic understanding of how to use PeopleSoft applications. We recommend that you complete at least one PeopleSoft introductory training course.

You should be familiar with navigating around the system and adding, updating, and deleting information using PeopleSoft windows, menus, and pages. You should also be comfortable using the World Wide Web and the Microsoft® Windows or Windows NT graphical user interface.

Related Documentation

To add to your knowledge of PeopleSoft applications and tools, you may want to refer to the documentation of the specific PeopleSoft applications your company uses. You can access additional documentation for this release from PeopleSoft Customer Connection (www.peoplesoft.com). We post updates and other items on Customer Connection, as well. In addition, documentation for this release is available on CD-ROM and in hard copy.

Important! Before upgrading, it is *imperative* that you check PeopleSoft Customer Connection for updates to the upgrade instructions. We continually post updates as we refine the upgrade process.

Documentation on the Internet

You can order printed, bound versions of the complete PeopleSoft documentation delivered on your PeopleBooks CD-ROM. You can order additional copies of the PeopleBooks CDs through the Documentation section of the PeopleSoft Customer Connection Web site: http://www.peoplesoft.com/

You'll also find updates to the documentation for this and previous releases on Customer Connection. Through the Documentation section of Customer Connection, you can download files to add to your PeopleBook library. You'll find a variety of useful and timely materials, including updates to the full PeopleSoft documentation delivered on your PeopleBooks CD.

Documentation on CD-ROM

Complete documentation for this PeopleTools release is provided in HTML format on the PeopleTools PeopleBooks CD-ROM. The documentation for the PeopleSoft applications you have purchased appears on a separate PeopleBooks CD for the product line.

Hardcopy Documentation

To order printed, bound volumes of the complete PeopleSoft documentation delivered on your PeopleBooks CD-ROM, visit the PeopleSoft Press Web site from the Documentation section of PeopleSoft Customer Connection. The PeopleSoft Press Web site is a joint venture between PeopleSoft and Consolidated Publications Incorporated (CPI), our book print vendor.

We make printed documentation for each major release available shortly after the software is first shipped. Customers and partners can order printed PeopleSoft documentation using any of the following methods:

Internet

From the main PeopleSoft Internet site, go to the Documentation section of Customer Connection. You can find order information under the Ordering PeopleBooks topic. Use a Customer Connection ID, credit card, or purchase order to place your order.

PeopleSoft Internet site: http://www.peoplesoft.com/.

Telephone

Contact Consolidated Publishing Incorporated (CPI) at **800 888 3559.**

Email

Email CPI at callcenter@conpub.com.

Typographical Conventions and Visual Cues

To help you locate and interpret information, we use a number of standard conventions in our online documentation.

Please take a moment to review the following typographical cues:

`monospace font`

Indicates PeopleCode.

Bold

Indicates field names and other page elements, such as buttons and group box labels, when these elements are documented below the page on which they appear. When we refer to these elements elsewhere in the documentation, we set them in Normal style (not in bold).

We also use boldface when we refer to navigational paths, menu names, or process actions (such as **Save** and **Run**).

Italics

Indicates a PeopleSoft or other book-length publication. We also use italics for *emphasis* and to indicate specific field values. When we cite a field value under the page on which it appears, we use this style: ***field value.***

We also use italics when we refer to words as words or letters as letters, as in the following: Enter the number *0*, not the letter *O*.

KEY+KEY

Indicates a key combination action. For example, a plus sign (+) between keys means that you must hold down the first key while you press the second key. For ALT+W, hold down the ALT key while you press W.

Jump links

Indicates a jump (also called a link, hyperlink, or hypertext link). Click a jump to move to the jump destination or referenced section.

Cross-references

The phrase For more information indicates where you can find additional documentation on the topic at hand. We include the navigational path to the referenced topic, separated by colons (:). Capitalized titles in *italics* indicate the title of a PeopleBook; capitalized titles in normal font refer to sections and specific topics within the PeopleBook. Cross-references typically begin with a jump link. Here's an example:

For more information**,** see Documentation on CD-ROM in *About These PeopleBooks:* Related Documentation.

- Topic list

Contains jump links to all the topics in the section. Note that these correspond to the heading levels you'll find in the Contents window.

Name of Page or Dialog Box

Opens a pop-up window that contains the named page or dialog box. Click the icon to display the image. Some screen shots may also appear inline (directly in the text).

Text in this bar indicates information that you should pay particular attention to as you work with your PeopleSoft system. If the note is preceded by **Important!,** the note is crucial and includes information that concerns what you need to do for the system to function properly.

Text in this bar indicates For more information cross-references to related or additional information.

Text within this bar indicates a crucial configuration consideration. Pay very close attention to these warning messages.

Comments and Suggestions

Your comments are important to us. We encourage you to tell us what you like, or what you would like changed about our documentation, PeopleBooks, and other PeopleSoft reference and training materials. Please send your suggestions to:

PeopleTools Product Documentation Manager
PeopleSoft, Inc.
4460 Hacienda Drive
Pleasanton, CA 94588

Or send comments by email to the authors of the PeopleSoft documentation at:

DOC@PEOPLESOFT.COM

While we cannot guarantee to answer every email message, we will pay careful attention to your comments and suggestions. We are always improving our product communications for you.

CHAPTER 1

Introduction to PeopleSoft Query

The main reason you store your business data in a database is so you can manipulate it to answer questions and solve business problems. However, getting just the information you're looking for often can be a difficult and time-consuming process.

With PeopleSoft Query, you can extract the precise data you want using visual representations of your PeopleSoft database, without having to write SQL statements. The queries can be as simple or as complex as necessary and they can be one-time ad-hoc queries or queries you'll use repeatedly.

In this chapter, you'll learn how to use PeopleSoft Query including the following:

- How Query fits into the overall system.
- How to run a predefined query.
- The process of creating queries.

Using Queries

You create queries to extract the data you need from your PeopleSoft database. Since there are lots of reasons you might need data, there are lots of ways you can use queries:

- **Display data in a grid (on the Results tab).** You can run queries within Query itself, displaying the result set in a grid for review. This option is useful as you refine your queries.
- **Provide input to a spreadsheet or Crystal report.** Query can seamlessly pass data to Microsoft Excel or Using Crystal Reports. From there, you can use the features of these products to create polished reports.
- **Search for records.** Many of the search dialog boxes in PeopleSoft applications enable you to open Query and create a query based on the search record definition. This option enables you to search for records based on sophisticated search criteria. This type of query is called a *search query*.
- **Create views.** Queries can be used in the Application Designer to generate views. Using queries instead of SQL has two big advantages: you can use the PeopleSoft Query to generate the SQL text for your view, and the SQL that is generated is database independent.
- **Check the database for conditions that trigger workflow events.** Using PeopleSoft Workflow, you can specify that certain conditions—say, the presence of overdue receivables—cause the system to send an email to someone or put an entry in someone's worklist. You write

queries to detect these conditions and schedule database agents to run them periodically.

- **Determine who to forward information to.** You can write queries that PeopleSoft Workflow uses to determine who to send emails, forms, or worklist entries to. This type of query is called a *role query*.
- Create the data structure for Peoplesoft nVision.
- **Create the basic structure of online analytical processing (OLAP) dimensions.** Queries can be used to modify tree structures, which you can then use as a basis for defining OLAP (On-line Analytical Processing) Cube data dimensions.
- **Run Query from any PeopleSoft application page**. Queries are integrated into both the PeopleSoft Windows client and Peoplesoft Internet Architecture. In the Windows client, you can select **Navigate, View, Query** to run a query. In the Peoplesoft Internet Architecture, you can run a query from within your browser.

Types of Queries

- **Ad hoc queries.** You can create and run queries to retrieve data from the database on the spot, when you need it. Additionally, queries written for use with Cube Manager must be ad hoc queries.
- **Reporting queries.** These queries retrieve data for reports and pass the data to Microsoft Excel, Crystal Reports Pro, or a PS/nVision report.
- **PS/nVision matrix queries.** You use these queries as part of a PS/nVision matrix layout.
- **View queries.** You can define queries in PeopleSoft Query that you'll use to create SQL views in the Application Designer.
- **Search queries.** Many of the search dialog boxes in PeopleSoft applications enable you to select a predefined query or create a new one rather than enter search criteria for the records you want to find.
- **Database agent queries.** If you use PeopleSoft Workflow, you can write queries to detect conditions that trigger business events, then schedule database agents to run them periodically.
- **Role queries.** You can write queries that PeopleSoft Workflow uses to determine who to send emails, forms, or worklist entries to.

Several of these query types have special requirements or considerations due to how they'll be used.

Ad Hoc Queries

Given the range of possible circumstances in which you might run an ad hoc query, there are no special considerations or requirements that apply to all of them.

Reporting Queries

When you define a custom report, you often include run-time variables that the user will specify when he or she runs the report. For example, you might want users to be able to say which business unit, location, or time period to report on. So, your reporting query may include one or more run-time prompt variables.

If your query requires input parameters like these, you need to decide how the user will go about entering them. If they run the report from PeopleSoft Query, there's no problem: they can enter values into the dialog box that Query displays. But if you want users to be able to use Process Scheduler, you may need to create or modify a page to collect the necessary input parameters. Depending on what the parameters are, you might have to create or modify a record definition as well. You'll definitely have to add a process definition to the Process Scheduler.

For more information on run-time prompts, see Specifying Query Selection Criteria. For details on adding custom reports to the Process Scheduler, see Process Scheduler for the End User.

PS/nVision Queries

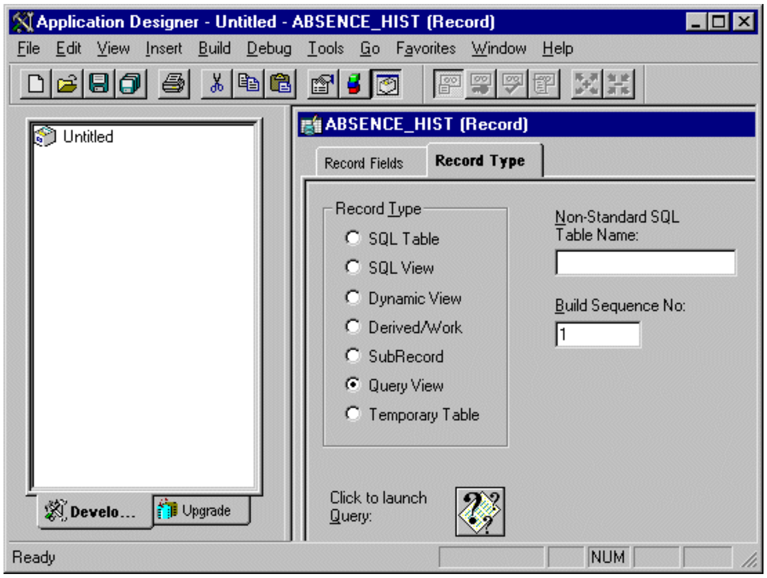
You can use a query created with PeopleSoft Query to specify the data source for a PS/nVision report layout. However, if you wish to use your query in a PS/nVision matrix layout, you must apply aggregate functions to at least one column.

For more information about aggregate functions, see Advanced Query Options. For more information about queries in PS/nVision reports, see Working with Layouts.

Queries used with PS/nVision tabular layouts don't have the same restrictions as matrix layout queries; they are like other reporting queries and do not require an aggregate column.

View Queries

To create a query that you'll use to define an SQL view, you start PeopleSoft Query from the **Record Type** tab in PeopleSoft Application Designer.


Application Designer - Untitled - ABSENCE_HIST (Record)
File Edit View Insert Build Debug Tools Go Favorites Window Help
Untitled
Develo...
Upgrade
ABSENCE_HIST (Record)
Record Fields
Record Type
Non-Standard SQL Table Name:
Click to launch Query:
Ready
NUM

Record Type
○ SQL Table
○ SQL View
○ Dynamic View
○ Derived/Work
○ SubRecord
● Query View
○ Temporary Table

Build Sequence No: 1

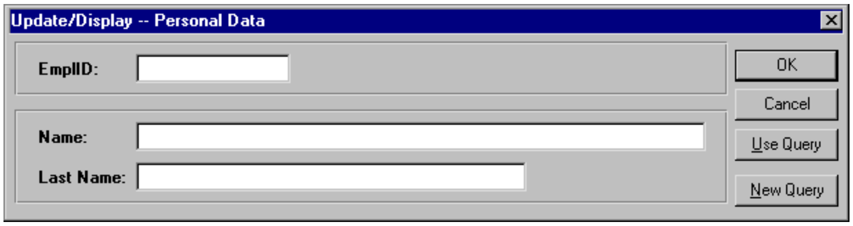
Application Designer – Untitle – ABSENCE_HIST (Record) page

When you start Query from Application Designer, the access group trees are not displayed, only the alphabetical list of record definitions. PeopleSoft Query doesn't display the access groups because it doesn't check which access groups you have rights to. It gives you access to all the record definitions that you have access to in Application Designer; that is, all the record definitions you have access to through Object Security.

For more information about using PeopleSoft Query queries to create SQL views, see Using Application Designer.

Search Dialog Box Queries

PeopleSoft applications display search dialog boxes whenever you select a page from a menu and need to specify which database record to display in the page. The search dialog box contains text boxes for each key field and other list box fields in the page's search record.



Search dialog box

To find the record you want, you usually enter search criteria into the text boxes. However, many search dialog boxes also enable you to use a query (defined in PeopleSoft Query) to specify the search values.

The **Use Query** and **New Query** buttons allow you to use Query to select your record and return the key values back to the search dialog box.

If you click the **New Query** button in a search dialog box, PeopleSoft Query starts and automatically selects the search record definition for the search dialog box. Query has also selected all the key fields and alternate search key fields as columns for output.

If you click the **Use Query** button, the system displays a list of the queries that are based on the search record definition. You can select one to run.

Note. The query you create or select must return values for all the text boxes in the search dialog box. In other words, key values for all list box fields in the search record definition must have values returned.

Database Agent Query

A *database agent* is a workflow program that runs a query and passes the results to PeopleSoft Message Agent. The database agent query checks the database for items that require further processing. For example, you could write a query that returns any overdue receivables, and schedule a database agent to run it once a week. When the database agent finds overdue receivables, it passes the data to Message Agent, which then enters the data into a page that in turn generates work list entries for the appropriate clerks.

A database agent can run any query that you can define with PeopleSoft Query. The query selection criteria define the business rule that's looking for exception conditions. Since database agents pass their query results to Message Agent, the query usually selects just the key fields for the page that Message Agent will fill out. Once the Message Agent enters the key field values, the page displays the rest of the data for that record.

All headings in the query must be text, rather than derived from the record field text (RFT) values.

Note. When you create a database agent query, it's very important that you use PeopleSoft Query's **Change, Query Type** option to identify it as a database agent query. PeopleSoft Workflow doesn't run the same security checks for database agent queries as it does for other queries, so if you forget to use change the query type to Database Agent, the query may not be successful at run time.

We recommend that you save your database agent queries with names that begin with [DBAG], so that you can easily identify them as database agent queries.

Lastly, in order for a database agent to transfer its results to Message Agent, you need to create a message definition for each query.

For more information about database agents, see the Workflow chapters in Understanding Workflow. That book also covers Message Agent and message definitions. For more information on PeopleSoft Query types, see Advanced Query Options

Role Queries

PeopleSoft Workflow uses role queries to determine who to send a work item to. A role query needs to return one or more role ID based on the data that's been saved on the page triggering the routing.

Since a role query returns a list of role users, the record definition you want is either PSROLEUSER (which lists role users and the role they are assigned to) or ROLEXLATOPR (which lists role users and their IDs).

The only field you'll select in your query is ROLEUSER. Of course, you'll use other fields and join to other record definitions to specify the criteria for which role users to select. But no matter how complex the query is—how many joins or selection criteria it has—it must return ROLEUSER and nothing more.

You define a role as a query because you want to route items differently based on the context of the transaction the user is performing. So, every role query will contain at least one bind variable whose value gets set at run time. The bind variable or variables correspond to the data you want to base the routing decision on. At run time, the system sets the values of the bind variables based on data from the page that triggers the event.

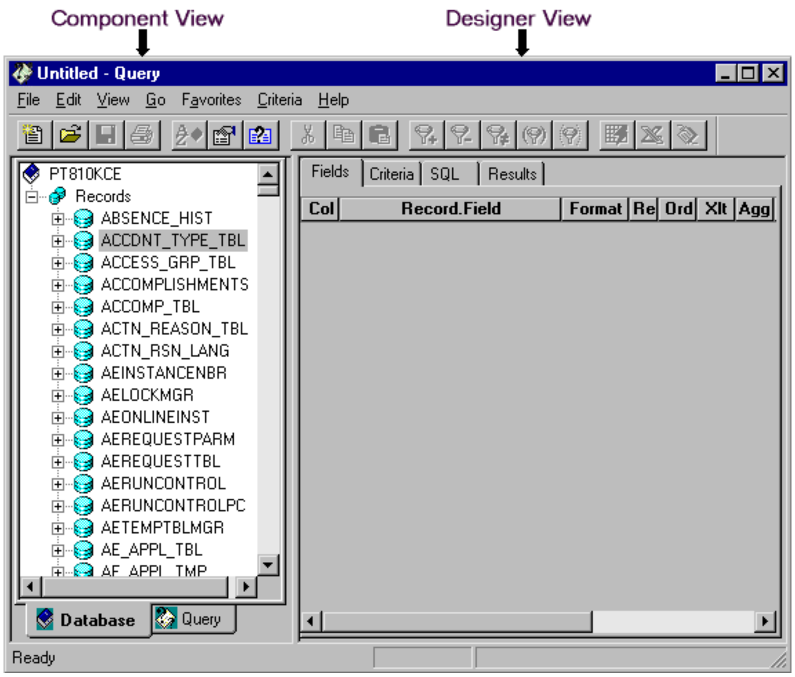
Note. Make sure you identify your query as a role query using PeopleSoft Query's Change, Query Type option. If you don't, the query may fail at run time.

Save your role queries using names that begin with [ROLE], so that you can identify them as role queries.

For more information about roles and role queries, see the Workflow chapters in Understanding Workflow.

Using PeopleSoft Query

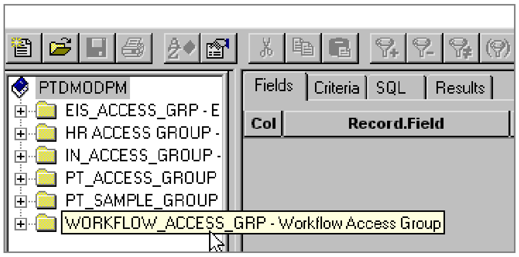
PeopleSoft Query is split into two views—the component view (on the left) and the designer view (on the right).



Component View and Designer View

You can easily adjust the relative size of either the component view or the designer view by dragging the divider bar that separates these views. The scroll bars at the bottom of each view are used to view the information on either side of the page that may not be visible.

Another convenient feature that you'll notice when using PeopleSoft Query is that you can view text in the component that is otherwise cut off by the divider bar if you simply place the mouse pointer over the item. You do not need to click.



View the full text of components by pointing at them.

Component View

The *component view*, on the left, has two tabs at the bottom, labeled **Database** and **Query.**

Database and Query Tabs

When you click the **Database** tab, you see a list of available record definitions (or record *components*) depending on your Query View Preferences. You can view these records as either a tree of access groups or as an alphabetical tree of records to which your user ID has access. When you click the **Query** tab, you see all the components used in the current query (such as record components, field components, prompts, expressions, and so on).

Component View Icons

The following icons are used in PeopleSoft Query displays:

Database name (displayed in the database component view only).

Query name and public/private status (displayed in the query component view only).

Record component.

Record hierarchy (displayed in record components) or related record hierarchy (displayed in field components).

Related record (displayed in record hierarchy).

Field component.

Key field.

Field having a related record.

Key field having a related record.

Expressions (the part of the query that contains individual expression components).

An expression component (a single expression that has been defined as part of the query).

Prompts

When using PeopleSoft Query, you can execute all operations on a component from the **Edit** menu. First you select a component in the tree, and then you select some edit operation to be applied to that component through the **Edit** menu. You can also right-click on a component to call up a context-sensitive pop-up menu. The menu item text in the **Edit** menu and in pop-up menus will vary depending on the type of component that is selected.

Designer View

The designer view is where you can view the fields, criteria, and other details associated with the current query. The first two tabs along the top of the designer view (**Fields** and **Criteria**) provide a simple and efficient way to create a query; each part of the SQL statement being built has its own space on the appropriate tab, neatly isolating the main pieces of the query. Clicking the **SQL** tab shows the SQL code generated by the SELECT statement, and the **Results** tab shows the result of the query, after it has been run.

For more information on controlling what you see in Query's views, see Designer View Preferences.

Running Predefined Queries

Sometimes all you need PeopleSoft Query for is to run a query that you or someone else has previously defined. You run a predefined query from within Query or from your browser using the PeopleSoft Internet Architecture.

For more information about running a query from your browser, see Running a Query on the Web.

To run a predefined query from Windows:

1. Select **File, Open** and open the desired query.

   You can select the Type of query you want to open using the Type drop-down list.

2. Select **File, Run** or click to run the query.

   This will send query results to the grid control (on the **Results** tab) for immediate viewing. You can also set your view preferences to run the current query when you click on the **Result** tab.

For more information about running queries, including other output options, see Creating and Running Simple Queries.

Creating Queries

To give you an idea of how Query works, let's create an example query to find all employees who have taken vacations of more than three weeks (15 working days). In this example, you won't see every feature in Query or get in-depth explanations for each step, but this will give you a good overview of using Query. Subsequent chapters provide the details of what you can do with this powerful yet easy-to-use tool.

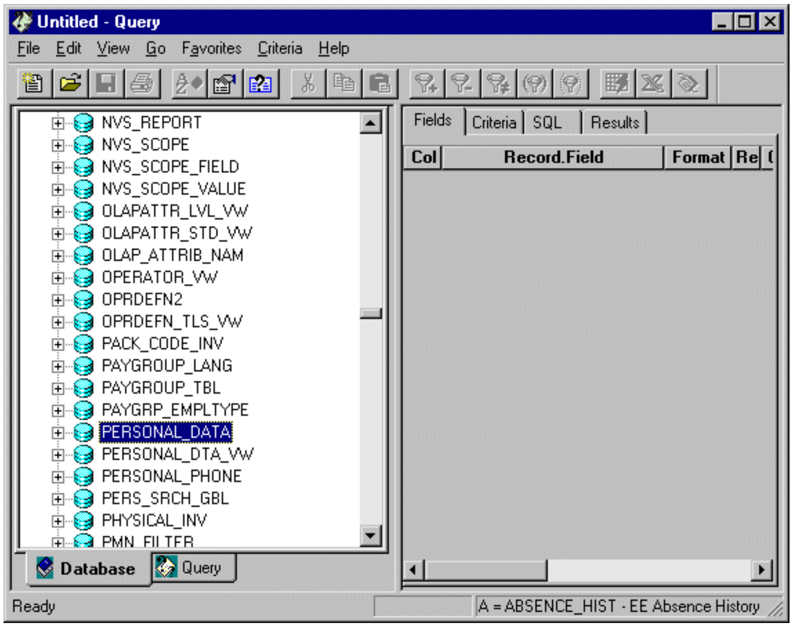
Selecting Record Components

In the database component view, you'll see either an access group structure or an alphabetical list of records to which you have access. You can switch between these two views by selecting **View, Preferences, Component View** and turning the **Show Access Groups** option on or off. For now, leave this option off.

For more information on view preferences, see Creating and Running Simple Queries.

To select record components:

3. Select Go, PeopleTools, Query.



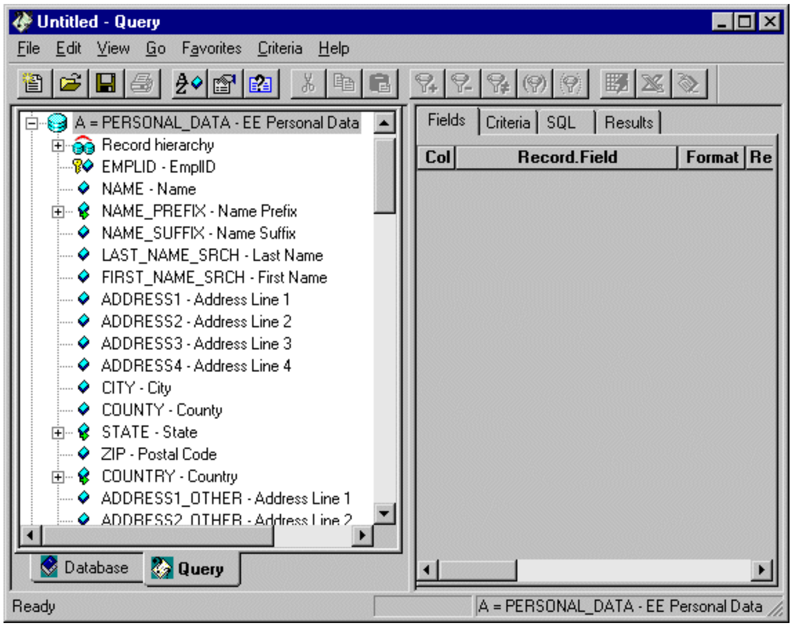
Untitled – Query page

4. Select Employee Personal Data record by scrolling down the database component view and selecting **PERSONAL_DATA.** You can do this several ways:

- Double-click the record *name*. (Try to avoid single-clicking the + sign at this point, as it expands the record component to show its structure, but does not add it to the query.)
- Drag the record component from the component view (on the left) to the designer view (on the right).
- Right-click the record component and select **Add** from the pop-up menu.
- Highlight the record component and select **Edit, Add** from the main menu. When adding a subsequent record, select **Edit, New Join.**

Note. You can also type the first few characters of the record name to scroll automatically to the record component.

Once you've selected a record component, Query automatically switches the component view from the **Database** tab to the **Query** tab to show the new record component that was added to the query. The designer view may or may not show fields, depending on how you've specified your view preferences. The default is to only show selected fields on the **Fields** tab. So far you've just added a record component, but no fields have yet been selected, so the **Fields** tab is empty.



A = PERSONAL_DATA - EE Personal Data
Record hierarchy
EMPLID - EmplID
NAME - Name
NAME_PREFIX - Name Prefix
NAME_SUFFIX - Name Suffix
LAST_NAME_SRCH - Last Name
FIRST_NAME_SRCH - First Name
ADDRESS1 - Address Line 1
ADDRESS2 - Address Line 2
ADDRESS3 - Address Line 3
ADDRESS4 - Address Line 4
CITY - City
COUNTY - County
STATE - State
ZIP - Postal Code
COUNTRY - Country
ADDRESS1_OTHER - Address Line 1
ADDRESS2_OTHER - Address Line 2

A = PERSONAL_DATA - EE Personal Data

Untitled - Query

Joining Records

The top of the query component view displays the name of the record component you selected, along with the letter *A*, followed by the fields comprising the record. Query uses *A* as an alias for the record component name. If your view preferences enable you to see fields in the designer view, you'll notice that each field name is prefixed with *A*. This prefix is a shorthand way of indicating the record component name.

For the example query, you're also going to need data from the Employee Absence History record. You'll need to *join* these two record components. Joining record components means finding a matching field in the two records and combining the rows that share a value in that field. In the Employee Personal Data and Employee Absence History record components, the matching field is EMPLID.

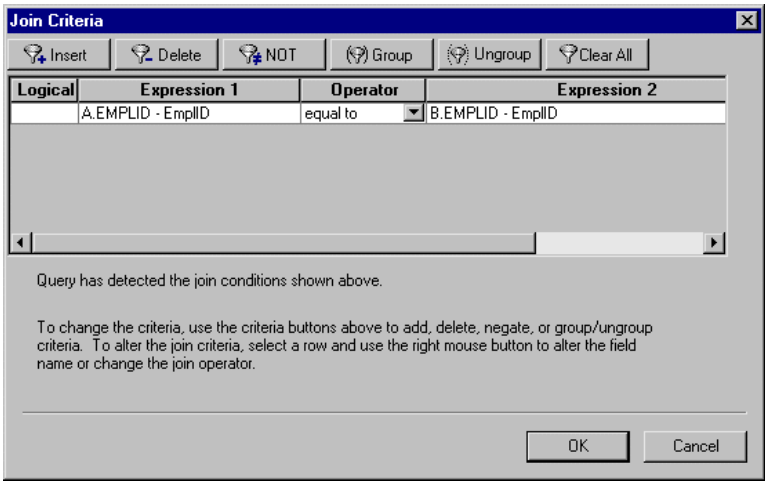
For more information on joins, see Advanced Query Options.

To join two records:

5. Click the **Database** tab in the component view to return to the view of records.

6. Scroll to the **ABSENCE_HIST** record in the database component view, and either double-click this record name or drag it into the designer view.

   Query will detect the join and display the following dialog box.



Logical	Expression 1	Operator	Expression 2
	A.EMPLID - EmplID	equal to	B.EMPLID - EmplID

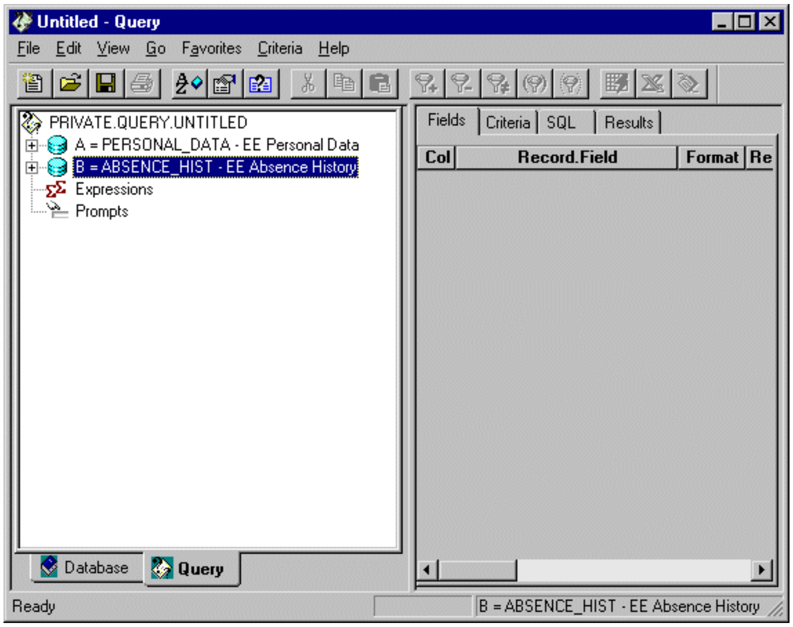
Query has detected the join conditions shown above.

To change the criteria, use the criteria buttons above to add, delete, negate, or group/ungroup criteria. To alter the join criteria, select a row and use the right mouse button to alter the field name or change the join operator.

Join Criteria dialog box

7. Click **OK** to accept the join conditions.

   Again, the component view will switch automatically from the **Database** tab to the **Query** tab, and the ABSENCE_HIST record appears in the query component view with a *B* prefix. If your view preferences enable you to see fields in the designer view, you'll notice that each field name is prefixed with *B*.


Untitled - Query
File Edit View Go Favorites Criteria Help
PRIVATE.QUERY.UNTITLED
A = PERSONAL_DATA - EE Personal Data
B = ABSENCE_HIST - EE Absence History
Expressions
Prompts
Fields
Criteria
SQL
Results
Col
Record.Field
Format
Re
Database
Query
Ready
B = ABSENCE_HIST - EE Absence History

B = ABSENCE_HIST selected

For more information on selecting record components, see Creating and Running Simple Queries .

Selecting Field Components

Once you've selected the record components that hold the data you want, you need to specify which field components to select from the record components. This procedure is very similar to the way in which you selected you record components.

To select field components:

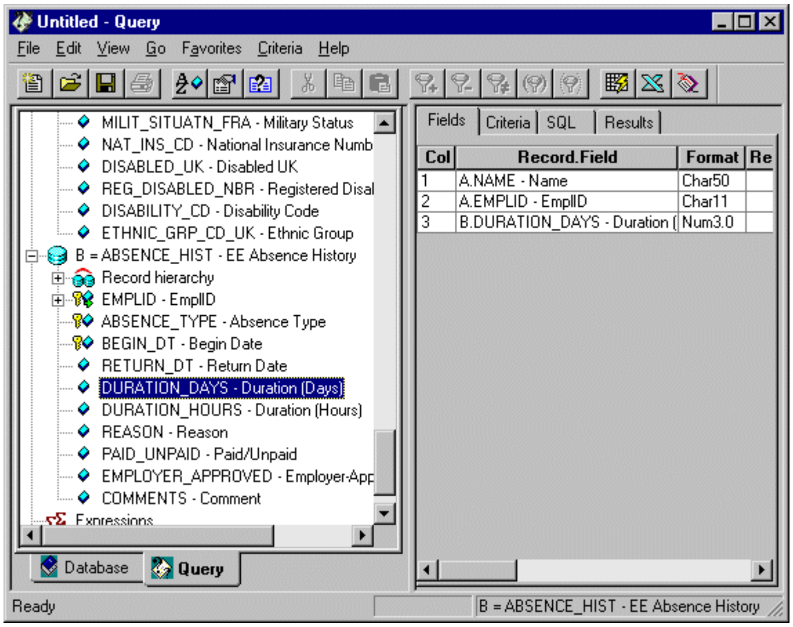
8. Select field components by clicking the Query tab at the bottom of the component view to ensure that it is active.

9. Next, select PERSONAL_DATA in the component view. If necessary, you'll click the + sign to show the fields in this record.

10. Then, select the fields NAME and EMPLID, in that order, by dragging them onto the Fields tab in the designer view. You can also add fields by double-clicking them.

11. Select ABSENCE_HIST in the query component view and choose the field DURATION_DAYS.

The numbers 1, 2, and 3 should appear in the **Col** column for NAME, EMPLID, and DURATION_DAYS, respectively.


Untitled - Query
File Edit View Go Favorites Criteria Help
MILIT_SITUATN_FRA - Military Status
NAT_INS_CD - National Insurance Numb
DISABLED_UK - Disabled UK
REG_DISABLED_NBR - Registered Disal
DISABILITY_CD - Disability Code
ETHNIC_GRP_CD_UK - Ethnic Group
B = ABSENCE_HIST - EE Absence History
Record hierarchy
EMPLID - EmplID
ABSENCE_TYPE - Absence Type
BEGIN_DT - Begin Date
RETURN_DT - Return Date
DURATION_DAYS - Duration (Days)
DURATION_HOURS - Duration (Hours)
REASON - Reason
PAID_UNPAID - Paid/Unpaid
EMPLOYER_APPROVED - Employer-App
COMMENTS - Comment
Expressions
Database
Query
Fields
Criteria
SQL
Results
Ready
B = ABSENCE_HIST - EE Absence History

Col	Record.Field	Format	Re
1	A.NAME - Name	Char50	
2	A.EMPLID - EmplID	Char11	
3	B.DURATION_DAYS - Duration (	Num3.0	

Untitled - Query

Note that the three fields you've chosen are prefixed automatically by *A* or *B* to indicate the record from which they were selected.

For more information on selecting field components, see Creating and Running Simple Queries.

Formatting Query Output

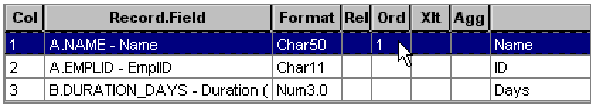
Query offers a number of different options for formatting query output. You can do the following:

- Change the column headings.
- Specify a sort order for the result rows.
- Display Translate Table values in place of codes.

In this example, you will format the query output by ordering the rows by employee name.

To format query output, double-click the Name row in the Ord column of the PERSONAL_DATA record to sort the order of the result rows.

A "1" appears to indicate that this field will determine the order of the result rows.



Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
1	A.NAME - Name	Char50		1			Name
2	A.EMPLID - EmplID	Char11					ID
3	B.DURATION_DAYS - Duration (	Num3.0					Days

Selecting the order of the results row

For more information on formatting query output, seeCreating and Running Simple Queries.

Choosing Selection Criteria

In most cases, you don’t want all the rows of data from the table, just the rows that meet certain conditions. In this example, you don’t want a list of all employee absences, only those that were for vacations of more than three weeks (15 working days). To limit the returned rows, you define *selection criteria*. If you know some SQL, you’ll notice that the criteria correspond to the WHERE clause of the SELECT statement.

To view or add selection criteria, click the **Criteria** tab in the designer view. This page displays one row for each query criterion.

For this example, you will add two selection criteria. The first one says that you only want records where the ABSENCE_TYPE is VAC (for vacation), and the second limits the results to absences of over 15 days. Notice that you’re able to use ABSENCE_TYPE in the selection criteria even though you haven’t included it as one of the output columns.

You’ll follow the steps below to add your criteria to the example. Use the horizontal scroll bar at the bottom of the designer view to scroll to the view as necessary, so that desired columns show up.

To specify selection criteria:

12. Click the Criteria tab active in the designer view.

13. Click the Query tab at the bottom of the component view, and then scroll to the ABSENCE_HIST record.

14. Drag the ABSENCE_TYPE field on to Criteria tab in the designer view.

15. In the Operator column, select ***equal to*** from the drop-down box.

(The ***equal to*** operator is the default, so you could bypass this step for ABSENCE_TYPE. However, it’s a good idea to verify that you’ve selected the correct operator.)

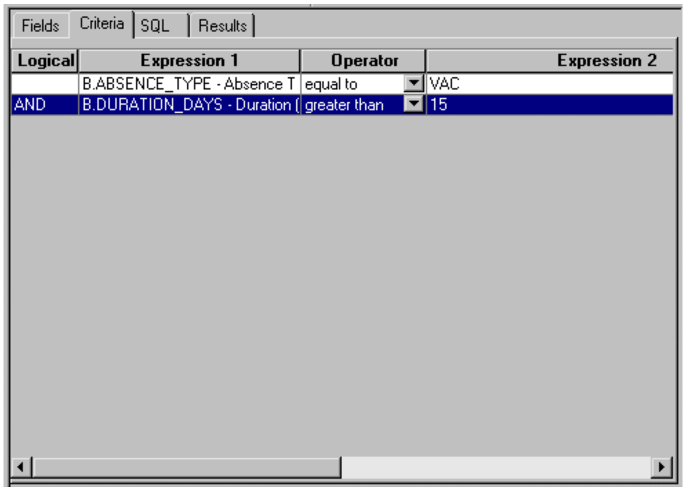
16. Right-click in the *right-hand* Expression 2 column.

There are two Expression columns, one to the left of the Operator column, and one to the right. You’ll typically have no need to change the *left-hand* Expression 1 column—it contains the field name (or, optionally, another expression).

17. If the Expression 2 column isn't visible, you can use the horizontal scroll bar at the bottom of the designer view to scroll to it, or simply resize the window so that both the left and right hand expression columns are visible.

18. After you right-click in the Expression 2 column, select Constant from the pop-up menu, and enter ***VAC*** in the dialog box that appears.

When you're done setting up the criteria for the ABSENCE_TYPE field, you'll set up the DURATION_DAYS criteria. Follow steps 1 through 6 above, making some minor changes to the selection criteria as you go. Select ***greater than*** in the Operator column, and enter a constant of ***15*** in the Expression 2 column. Note that the Logical column defaults to a value of ***AND***. You accept this, because you want to use both criteria together (absence type of VAC *and* a duration of 15 days), as opposed to either criterion (absence type of VAC *or* a duration of 15 days).



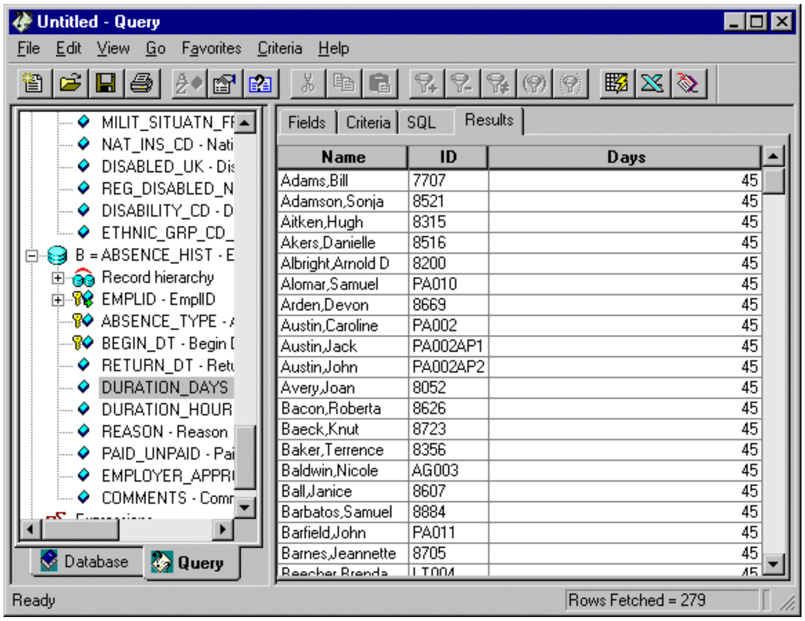
Logical	Expression 1	Operator	Expression 2
	B.ABSENCE_TYPE - Absence T	equal to	VAC
AND	B.DURATION_DAYS - Duration (	greater than	15

Criteria tab

For more information **on defining selection criteria,** see Specifying Query Selection Criteria.

Running a Query

Select File, Run **(Current Query)** or click the button. Query retrieves the requested data from the database and displays the results in a grid control in the designer view's **Results** tab. Note that the results are ordered by name, as was specified in step 3, "Format the query output."



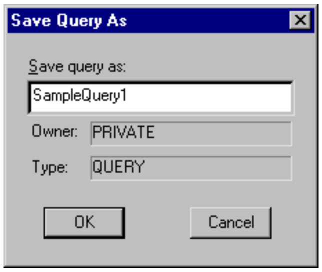
Name	ID	Days
Adams,Bill	7707	45
Adamson,Sonja	8521	45
Aitken,Hugh	8315	45
Akers,Danielle	8516	45
Albright,Arnold D	8200	45
Alomar,Samuel	PA010	45
Arden,Devon	8669	45
Austin,Caroline	PA002	45
Austin,Jack	PA002AP1	45
Austin,John	PA002AP2	45
Avery,Joan	8052	45
Bacon,Roberta	8626	45
Baeck,Knut	8723	45
Baker,Terrence	8356	45
Baldwin,Nicole	AG003	45
Ball,Janice	8607	45
Barbatos,Samuel	8884	45
Barfield,John	PA011	45
Barnes,Jeannette	8705	45
Beecher,Brenda	LT004	45

Rows Fetched = 279

Untitled Query – Results page

For more information on options for running your queries, see Creating and Running Simple Queries.

To save your query, simply select File, Save (or Save As) or click the button. Enter a name for your query, and then click OK. Here the query is named *SampleQuery1*.



Save query as: SampleQuery1

Owner: PRIVATE

Type: QUERY

Save Query As dialog box
You can use this dialog to save your query under another name

CHAPTER 2

Setting up Query Security

Security is critical for your business data. Typically, you don't want everyone in your company to have access to all your data. With this need in mind, we've integrated our standard security features into PeopleSoft Query. Query takes advantage of user profiles and row-level security. You can control what query operations a user can perform and what data they can access using Query.

In this chapter, we'll show you how to establish a security scheme for PeopleSoft Query. You'll learn how to do the following:

- Build query trees, which organize your record components into logical groups for security access.
- Provide users with rights to all or part of the query trees.
- Set up your record definitions so that Query enforces row-level security.

You perform these setup tasks using PeopleTools other than Query, **including Maintain Security, Tree Manager, and Application Designer.** This chapter describes the steps for setting up Query security.

Defining Query Profiles

Query takes advantage of user profiles and row-level security. Using Maintain Security, you can control what query operations a user can perform and what data they can access while they are using Query.

For more information on creating Query Profiles in Maintain Security, see Query/Table-Level Security.

Building Query Trees

Trees are a graphical way of presenting hierarchical information. PeopleSoft Query uses *query trees* to control security of the tables in your PeopleSoft database. You define a hierarchy of record components, based on logical or functional groupings, and then give users access to one or more "branches" of the tree. Users can use Query to retrieve information only from those tables whose record definitions they have access to.

You create and update query trees using PeopleSoftTree Manager. To open Tree Manager, select **Go, PeopleTools, Tree Manager.** Using Tree Manager, you can view and update existing query trees or create new ones.

To get you started, we've included some sample query trees with your PeopleSoft applications. Which trees you have depends on which PeopleSoft applications you've installed. Each tree contains access groups and record components categorized by function. *Access groups* mark and define a functional group of records or other access groups—in other words, they are descriptive placeholders used to categorize actual record components in a logical, hierarchical format. When you define users' security rights to a tree, you specify which access groups they have access to.

This section explains how to create query trees. It assumes that you're familiar with the concept and terminology of trees.

For more information**,** see Introduction to Tree Manager.

Query Tree Considerations

You should create your own query trees based on your organization's needs and on any customizations you've made. Remember that the sample trees we provide may be replaced when you upgrade to a subsequent PeopleSoft release, so if you modify the samples rather than create your own trees, you may lose your customizations.

Every record component that you want users to be able to query must be in a query tree. However, they don't all have to be in the same query tree. One strategy is to use the sample query trees to provide access to the standard PeopleSoft record definitions, but create separate query trees for record definitions that you add in the course of customizing the system. This way, you take advantage of the sample trees but avoid overwriting your changes during future upgrades.

How you organize the contents of your query tree depends on the needs of your organization and your users. For example, you might want to create small trees that are not intimidating to non-technical or casual users. The sample query trees provided in your PeopleSoft application are divided by functions, but to simplify the trees, you may want to create separate trees that contain subcategories of each function. For example, you could create separate trees for U.S., Canadian, and international record components in order to grant users in each region security access to only those record components they should use.

Note. You should consider adding record components to your query trees in a hierarchy that matches the parent/child relationship of records in your database. Though you don't have to organize records this way—Application Designer actually controls the parent/child hierarchy in your database—you'll probably find it helpful to keep your query trees consistent with your database structure.

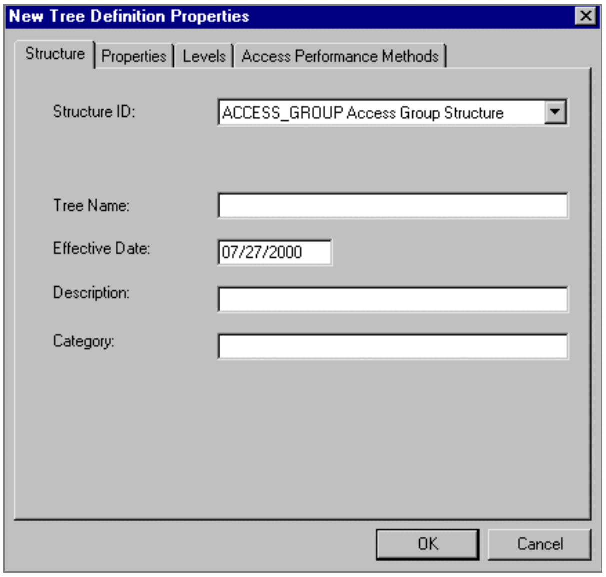
It's a good idea when creating your tree to have an access group that includes all components of the tree makes. This makes it easier to give users access to all tables—you only need to add one row on the Access Groups tab in Query Security.

Defining Your Query Tree

Before you can insert nodes for access groups and record components, you must first define a number of important characteristics for your tree.

To define a query tree:

1. In Tree Manager, select **File, New** to open to the New Tree Definition Properties dialog box, where you specify general information about the tree.



Structure ID: ACCESS_GROUP Access Group Structure

Tree Name:

Effective Date: 07/27/2000

Description:

Category:

New Tree Definition Properties dialog box – Structure tab

2. Enter a Structure ID for the query tree.

   The Structure ID for a query tree must always be ACCESS_GROUP. You can either type ACCESS_GROUP in the text box or press **F4** and select it from the prompt list.

   When you move the cursor out of this field, **Use of Levels** defaults to ***Not Used*** because query trees don't use levels.

3. Enter a name for the tree.

   For the tree name, we recommend that you start the name with ***QRY_*** so that you can easily identify the tree as a custom query tree. (The standard query trees we deliver with the system start with QUERY_.)

4. Enter an Effective Date for the tree.

   The status defaults to ***Active***. Query trees are available immediately if the effective date is active; you don't need to run an SQR utility like you do for organizational security trees.

5. In the Description field, specify a name or caption for the tree.

This description will appear along with the name and effective date in the list box whenever you select from a list of trees.

Note. You can change a tree description at any time by selecting **File, Definition** from within Tree Manager.

6. You can also specify a Category**,** if desired.

Note. Currently, **Detail Values** are not necessary for query trees.

7. Finish your new Tree definition by completing the Properties, Levels, and Access Performance Methods dialogs.

For more information about completing these dialogs and creating new Trees, see Introduction to Tree Manager.

8. Once you've completed your tree definition, click OK**.**

   Tree Manager displays the name and effective date of the query tree you are creating. Together, these fields form a key that identifies each unique version of a tree.

Inserting Nodes

Once you've defined your tree, it's time to build the hierarchy of record components that you want users to have access to.

The hierarchy in a query tree consists of access group nodes and record component nodes. Access group nodes are important for security—to provide users with access to record components, you specify which access groups they have rights to. So, you group all the record components that "go together" from a security perspective under a single access group node. Similarly, if you frequently give rights to Access Group A when you give rights to Access Group B, you can create a higher-level access group that contains both of them. To see an example, open one of the sample query trees in Tree Manager.

Note. To make the relationship between related record components visually clear, you can make one component definition the child of another. However, this organization has no functional effect. The Application Designer manages the relationship between parent and child record definitions.

The first node in the tree is known as the *root node*, and it is the highest level in the hierarchy of your tree. You can only insert the root node using the **Edit** menu. Once the root has been created, there are two ways to enter additional nodes:

- Use the Insert commands on the Edit menu.
- Use the Insert Node button on the toolbar.

For more information about trees, see Introduction to Tree Manager..

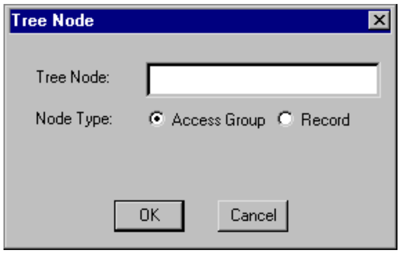
To insert the root node and additional nodes:

9. To enter the root node, select Edit, Insert Node.

A dialog box appears, prompting you to enter the name of the tree node and to indicate whether the node type is an access group or a record.

10. Assign a name to the root node.

The name you assign to the root node is the name that will appear as the high-level access group name in Query.



Tree Node dialog box

If you're adding an access group node and you want to include more than one word in the name, the words need not be connected with an underline (_). In fact, you'll find it easier to distinguish between access group and record component nodes if you don't use underlines.

When you add a record component node, you must enter the exact name of the record definition you're adding.

11. Once you've entered a name and selected the type of node, click **OK.**

If you selected an access group node, you'll be prompted to enter a **Description** and a **Definition.** These are optional.

Note. If you use an access group name that's been used in the past, Tree Manager adds the node to the tree using the existing description and definition. To review them, double-click on the node.

If you selected a record node, the system automatically enters the description for the record definition.

12. To add nodes off the root, select **Edit, Insert Child.** The process is otherwise the same as for the root node.

    Every node is defined according to its relation to other nodes.

13. To insert nodes, highlight an existing node and specify whether the new node should share the same parent (**Edit, Insert Sibling**) or be inserted at a lower level (**Edit, Insert Child**).

Note. The **Edit, Insert** command is context-sensitive, in that it changes according to the location of the cursor. When you highlight the root node, the **Edit** menu only displays **Insert Child**. When you highlight any other node in the tree, it displays both **Insert Sibling** and **Insert Child**.

14. When you're done defining the tree, select **File, Save** to save your work.

Row-Level Security and Query Security Record Definitions

By default, when you give Query users access to a record component, they have access to all the rows of data in the table built using the associated record definition. In some cases, though, you want to restrict users from seeing some of those data rows. For example, you might not want your human resources staff to have access to compensation data for vice presidents or above. In other words, you want to enforce *row-level security*, which is offered by many PeopleSoft applications.

With row-level security, users can have access to a table without having access to all rows on that table. This type of security is typically applied to tables that hold sensitive data. For example, you might want users to be able to review personal data for employees in their own department, but not for people in other departments. You would give everyone access to the PERSONAL_DATA table, but would enforce row-level security so that they could only see rows where the DEPTID matches their own.

PeopleSoft applications implement row-level security by using a SQL view that joins the data table with an authorization table. When a user searches for data in the data table, the system performs a related record join between the view and the base table rather than searching the table directly. The view adds a security check to the search, based on the criteria you've set up for row-level security. For example, to restrict users to seeing data from their own department, the view would select from the underlying table just those rows where the DEPTID matches the user's DEPTID.

Query Security Record Definitions

You implement row-level security by having Query search for data using a *query security record definition*. The query security record definition adds a security check to the search.

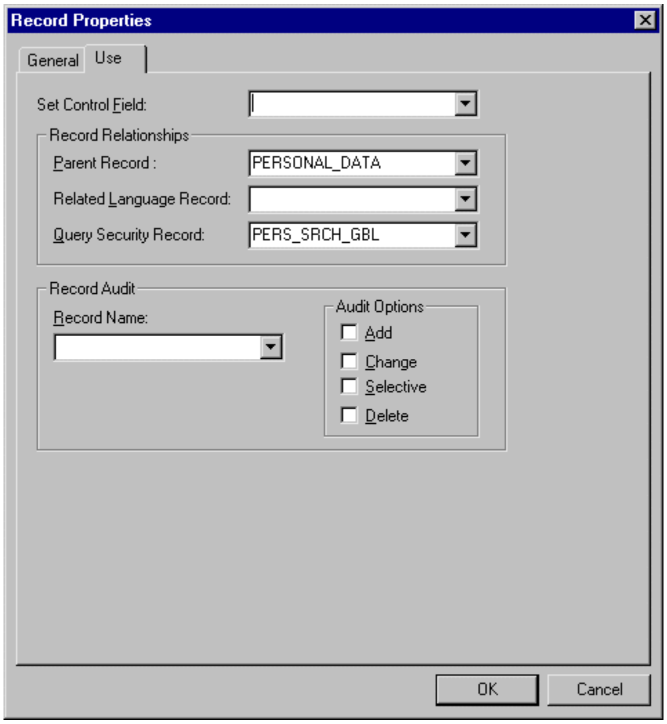
Query security record definitions serve the same purpose as search record definitions do for panels. Just as a panel's search record definition determines what data the user can display in the panel, the query security record definition determines what data the user can display with Query.

To get Query to retrieve data by joining a security record definition to the base table, you specify the appropriate **Query Security Record** when you create the base table's record definition.

To apply row level security

15. Select **Go, PeopleTools, Application Designer** to open the Application Designer, and open the record on which you want to apply row-level security.

16. With the record definition open in the Application Designer, click the **Properties** button , and select the **Use** tab from the **Record Properties** dialog box.



Set Control Field:

Record Relationships

Parent Record : PERSONAL_DATA

Related Language Record:

Query Security Record: PERS_SRCH_GBL

Record Audit

Record Name:

Audit Options

□ Add

□ Change

□ Selective

□ Delete

Record Properties dialog box – Use tab

Note. You use this dialog box to set a number of different aspects of the record definition. For details, see Using Application Designer. The only item we're concerned with here is the **Query Security Record** list box.

17. Select the security record definition (usually a view) in the **Query Security Record** list box.

    Each PeopleSoft product line comes with a set of views for implementing its standard row-level security options. See the product documentation for details.

Note. The **Parent Record** list box is also relevant to Query. It identifies a record definition that is the current definition's *parent*, meaning that it holds related data and that its keys are a subset of the current record definition's keys. If you designate a parent record, Query automatically knows what fields to use when you join these two tables for a query.

For more information**,** see Advanced Query Options.

In most cases, the Query Security Record definition you'll want to select is the same one you use as the search record definition for the panel that manages this table. If you're enforcing one of the standard row-level security options from a PeopleSoft application, select the PeopleSoft-supplied security view for that option. See the application documentation for a list of the available views. If you've designed your own security scheme, select a record definition that appropriately restricts the rows a query will return.

18. Once you've set the query security record definition, click **OK** to close the Record Properties dialog box, then save the record definition.

If you've already used SQL Create to build a table from this record definition, you don't need to rebuild it.

Note. PeopleSoft's row-level security views restrict users from seeing certain rows of data. If you specify a query security record for a given base record definition, PeopleSoft Query adds a qualifier to the WHERE clause of each query, instructing the system to retrieve only rows in organizational entities you've been granted access to. If you perform a "historical" query—for example, a query asking for the employees in your department as of last year—you may not get the results you expect. Because the system is enforcing row-level security, PeopleSoft Query will only return those employees who were in the department last year *and are currently in a department you have access to*.

CHAPTER 3

Creating and Running Simple Queries

This chapter describes the basic operation of PeopleSoft Query. It discusses the basic functions you need to know in order to create, run, and save simple queries.

Creating a Query

The basic steps involved in creating a query are as follows:

1. Select record components.
2. Select field components.
3. Format the query output.
4. Specify selection criteria.
5. Run the query.
6. Save the query.

This chapter provides details on each of these steps, except for the selection of criteria.

For more information**,** see Specifying Query Selection Criteria.

Selecting Record Components

The first step in creating a query is to select the database tables that hold the data you want. In your PeopleSoft database, tables are represented as record definitions. In PeopleSoft Query, we refer to the record definitions as record *components*. We do this to indicate that they are one component of your query. As you'll see, queries can have different kinds of components.

You can execute all operations on a component from the Edit menu. First you select a component in the tree, and then you select some edit operation to be applied to that component through the Edit menu. You can also right-click on a component to call up a context-sensitive pop-up menu. The menu item text in the Edit menu and in pop-up menus will vary depending on the type of component that is selected.

To select a record component:

1. Click the Database tab in the component view.

The component view (on the left) has two tabs at the bottom, Database and Query. With the database tab active, you'll see either an access group structure or an alphabetical list of records to which you have access. You can switch between these two views by changing your View Preferences.

2. Select the desired record.

You can do this several ways:

- Double-click on the record *name* (not the + sign).
- Drag the record component from the component view (on the left) to the designer view (on the right).
- Right-click on the record component and select **Add** from the pop-up menu.
- Select the record component and select **Edit, Add.** If this is not the first record component in the query, the menu option will read **Edit, New Join** to indicate a join operation.

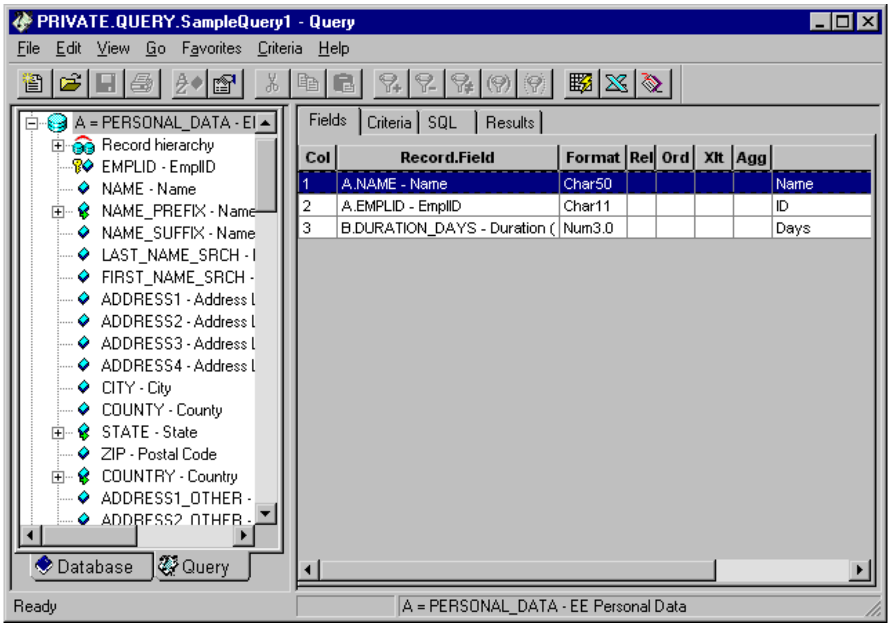
Note. To use the **Edit** menu, you must first select a component in the component view.

Once you've selected a record component, Query automatically switches the component view from the **Database** tab to the **Query** tab to show the new record component that was added to the query. The designer view may or may not show fields, depending on how you've specified your view preferences.

If you've selected the record component for an effective-dated table, Query displays a dialog box for you to choose how you want to handle effective dates.

For more information about choosing effective date options, see Specifying Effective Date Criteria.

At the top of the query component tree is the record component you selected, along with a letter, followed by the fields comprising the record. Query uses the letter as an "alias" for the record component name. If your view preferences enable you to see fields in the designer view, you'll notice that each field name is prefixed with a letter. This prefix is a shorthand way of indicating the record component name. The first record component you select will appear as record A, the second as record B, and so on.



Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
1	A.NAME - Name	Char50					Name
2	A.EMPLID - EmplID	Char11					ID
3	B.DURATION_DAYS - Duration (	Num3.0					Days

Viewing Selected Records and Fields

Selecting Field Components

Once you've selected the record components that hold the data you want, you need to specify which fields to select from the record components. The way you accomplish this is similar to the way in which you selected the record components you wanted to work with.

To select a field component:

3. Click the Query tab in the component view.

   The component view (on the left) has two tabs at the bottom, Database and Query. Click the Query tab to ensure that it is the active tab.

4. Either double-click the field you want to select or drag and drop the field from the component view to the designer view.

Note. You can also right-click the field you want and choose Select from the pop-up menu, or simply add all fields in the record definition to the query by choosing Select all fields from the pop-up menu. With the field component selected, you can also choose Edit, Select from the main menu.

Notice what happens in the designer view. With the **Fields** tab active, a number appears in the first column (**Col**), indicating the order of the fields in the query results. PeopleSoft Query adds columns in the order you select the fields—the first field you select will appear as column one, the second as column two, and so on. You can select fields in any order; you don't need to use the order from the record component.

Note. If you're not happy with where a field is in the order, you can change it by right-clicking the row that contains the field you'd like to move.

5. Select **Column Number** from the pop-up menu, enter the new column number in the dialog box that appears, and click OK**.**

   Query renumbers the remaining fields to reflect the new column order.

6. To deselect a field that you've decided not to include, double-click it.

Query removes the field from the query and renumbers the remaining fields.

The designer view can show the fields from one or all of the record components you've selected. Again, this depends on your view preferences. However, you always can tell which record component a field comes from by looking at its prefix, which matches the component's alias.

In most cases, you don't want all the rows of data from the record components specified in your query, just the rows that meet certain conditions.

For more information**,** see Specifying Query Selection Criteria.

View Preferences

What you see in the component view and designer view often depends on your view preferences. You can change your view preferences by selecting **View, Preferences.** The Preferences dialog box includes three tabs labeled General, Component View, and Designer View.

The default view preferences are as follows.

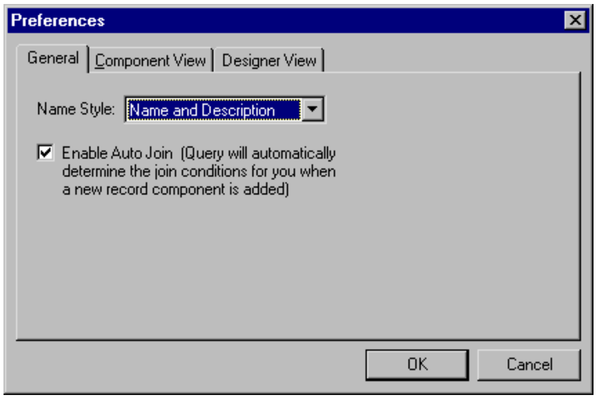
View Preferences Tab	***Default Value(s)***
General	Show User Queries (Ad Hoc)
Component View	Show Access Groups (Checked On)
Designer View	Show Fields In Select List; Sort Fields By Col Order
	Don't run query when Result tab is clicked.
	Don't show row headers in Result tab.

Whenever you change your view preferences, the new settings are saved in the Windows registry and are retained from one Query session to another.

Note. Unless otherwise stated, the examples in this document use the default views.

General View Preferences

The General view preferences tab enables you to select a name style and choose whether or not you want Query to determine the conditions of an Any Record join by enabling the Auto Join option. When you select **View**, **Preferences**, the Enable Auto Join option is automatically checked.



Name Style: Name and Description

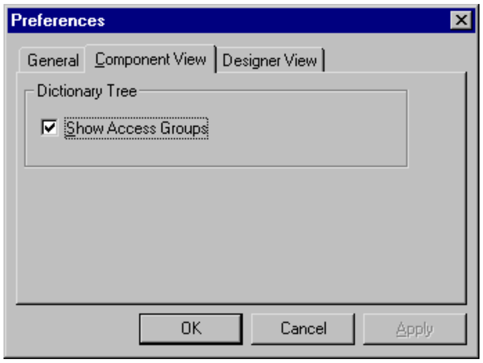
☑ Enable Auto Join (Query will automatically determine the join conditions for you when a new record component is added)

Preferences dialog box – General tab

The **Name Style** field determines whether field components appear with both their field name and description or with the description only.

Component View Preferences

The Component View preferences tab enables you to turn the Show Access Groups option on or off.



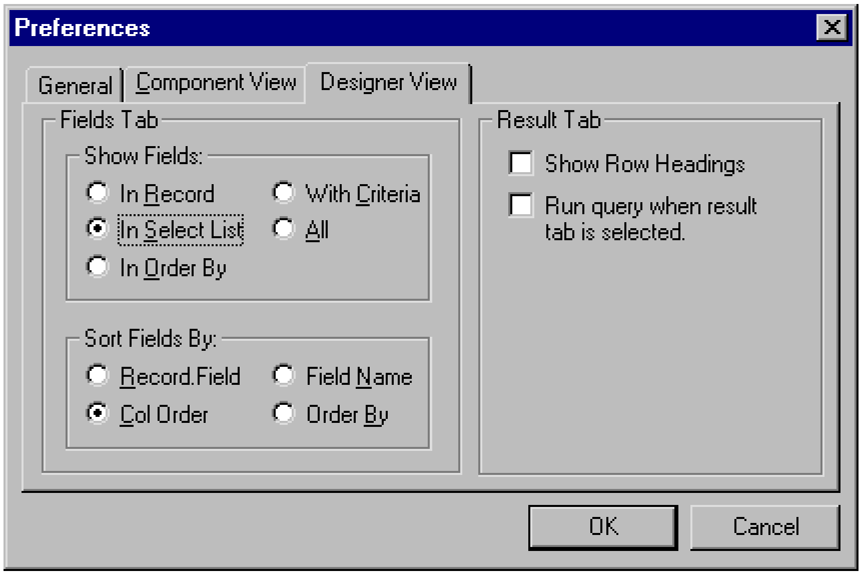
☑ Show Access Groups

Preferences dialog box – Component View tab

In the database component view, you'll see either an access group structure or an alphabetical list of records to which you have access. You can switch between these two views by selecting or clearing the Show Access Groups check box.

Designer View Preferences

The **Designer View** tab controls which fields you see when building your query and in what order they appear. You can use a combination of the **Designer View** options to display the fields in the way that best suits your query needs.



Preferences

General | Component View | Designer View

Fields Tab

Show Fields:

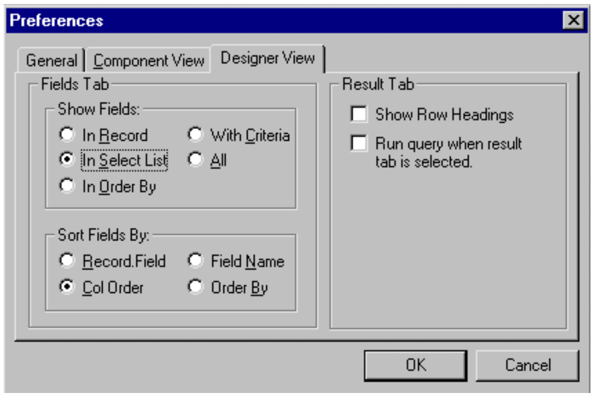
○ In Record ○ With Criteria
● In Select List ○ All
○ In Order By

Sort Fields By:

○ Record.Field ○ Field Name
● Col Order ○ Order By

Result Tab

□ Show Row Headings
□ Run query when result tab is selected.



Preferences

General | Component View | Designer View

Fields Tab

Show Fields:

○ In Record ○ With Criteria
● In Select List ○ All
○ In Order By

Sort Fields By:

○ Record.Field ○ Field Name
● Col Order ○ Order By

Result Tab

□ Show Row Headings
□ Run query when result tab is selected.

Setting Preferences for Viewing and Sorting Fields

Show Fields Options

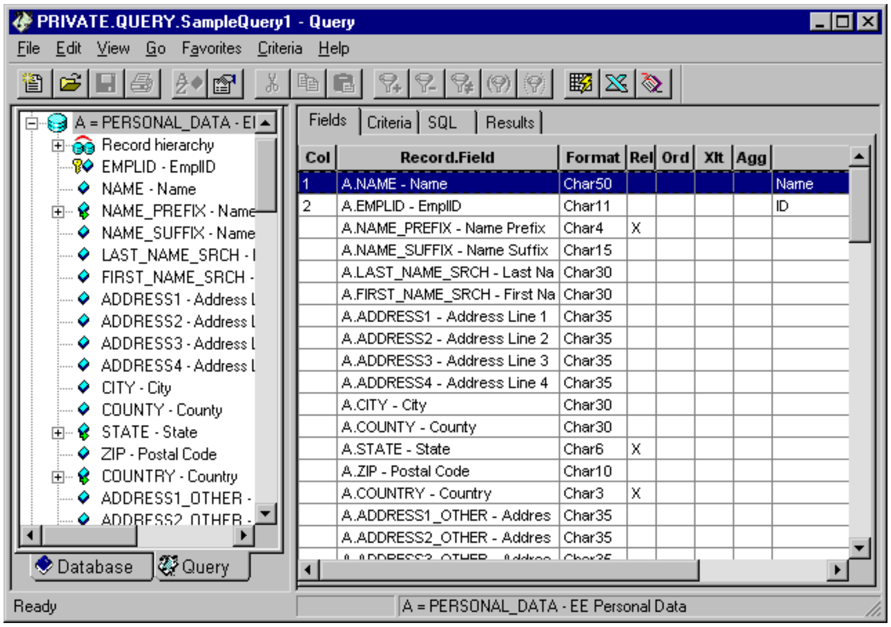
The radio buttons in the **Show Fields** group box specify what fields appear in the designer view.

In Record Option

In this view, you see all the fields in the record that is highlighted in the component view. Even if you have selected more than one record in your query, you only see the fields in the highlighted record. For example, since PERSONAL_DATA is highlighted, you see the fields in that record.

For more information on joining records, see Advanced Query Options.

Note. The last record selected in the query tab of the component view is known as the active record. The active record may not always be highlighted in the component view, but it is always displayed in the status bar. The record alias preceding the field name also serves to identify the active record. It is not necessary to select a record component to change the active record. The active record will change whenever you select any child component of another record component (a field component, for example).



A = PERSONAL_DATA - EI
Record hierarchy
EMPLID - EmplID
NAME - Name
NAME_PREFIX - Name
NAME_SUFFIX - Name
LAST_NAME_SRCH - l
FIRST_NAME_SRCH -
ADDRESS1 - Address l
ADDRESS2 - Address l
ADDRESS3 - Address l
ADDRESS4 - Address l
CITY - City
COUNTY - County
STATE - State
ZIP - Postal Code
COUNTRY - Country
ADDRESS1_OTHER -
ADDRESS2_OTHER -

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
1	A.NAME - Name	Char50					Name
2	A.EMPLID - EmplID	Char11					ID
	A.NAME_PREFIX - Name Prefix	Char4	X				
	A.NAME_SUFFIX - Name Suffix	Char15					
	A.LAST_NAME_SRCH - Last Na	Char30					
	A.FIRST_NAME_SRCH - First Na	Char30					
	A.ADDRESS1 - Address Line 1	Char35					
	A.ADDRESS2 - Address Line 2	Char35					
	A.ADDRESS3 - Address Line 3	Char35					
	A.ADDRESS4 - Address Line 4	Char35					
	A.CITY - City	Char30					
	A.COUNTY - County	Char30					
	A.STATE - State	Char6	X				
	A.ZIP - Postal Code	Char10					
	A.COUNTRY - Country	Char3	X				
	A.ADDRESS1_OTHER - Addres	Char35					
	A.ADDRESS2_OTHER - Addres	Char35					
	A.ADDRESS3_OTHER - Addres	Char35					

Ready

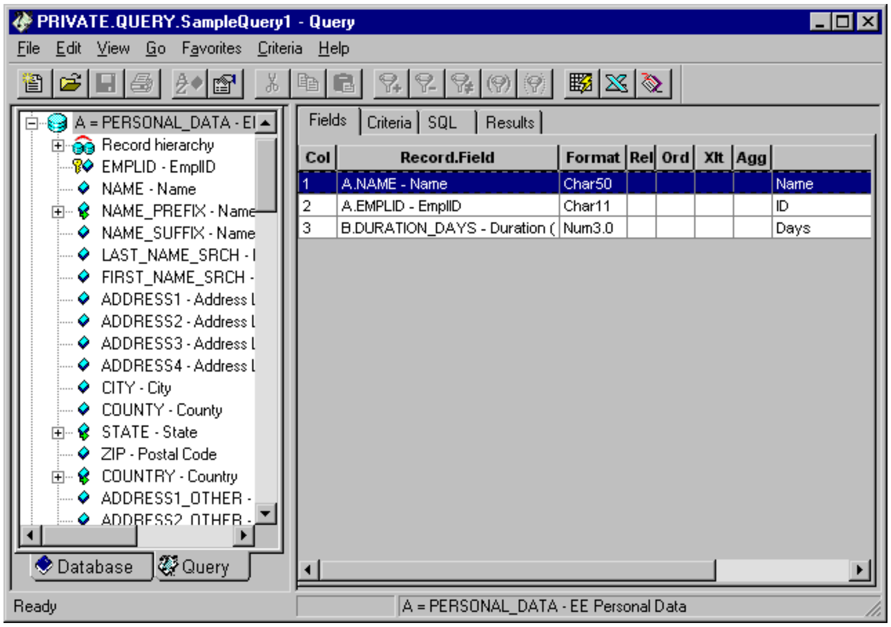
A = PERSONAL_DATA - EE Personal Data

In Record view

Note. When the **In Record** preference is set, new record components are not automatically expanded by Query. It is assumed that the user will operate on field components from the designer view, rather than the component view.

In Select List Option

In this view, you see all the fields selected for use in the entire query. For example, this query uses fields from two records, PERSONAL_DATA and ABSENCE_HIST, and we see the fields from both records, regardless of which record is selected in the component view. In Selected List is the default Show Field option.


PRIVATE.QUERY.SampleQuery1 - Query
File Edit View Go Favorites Criteria Help
Fields
Criteria
SQL
Results
Database
Query

A = PERSONAL_DATA - EI
Record hierarchy
EMPLID - EmplID
NAME - Name
NAME_PREFIX - Name
NAME_SUFFIX - Name
LAST_NAME_SRCH - I
FIRST_NAME_SRCH -
ADDRESS1 - Address L
ADDRESS2 - Address L
ADDRESS3 - Address L
ADDRESS4 - Address L
CITY - City
COUNTY - County
STATE - State
ZIP - Postal Code
COUNTRY - Country
ADDRESS1_OTHER -
ADDRESS2_OTHER -

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
1	A.NAME - Name	Char50					Name
2	A.EMPLID - EmplID	Char11					ID
3	B.DURATION_DAYS - Duration (	Num3.0					Days

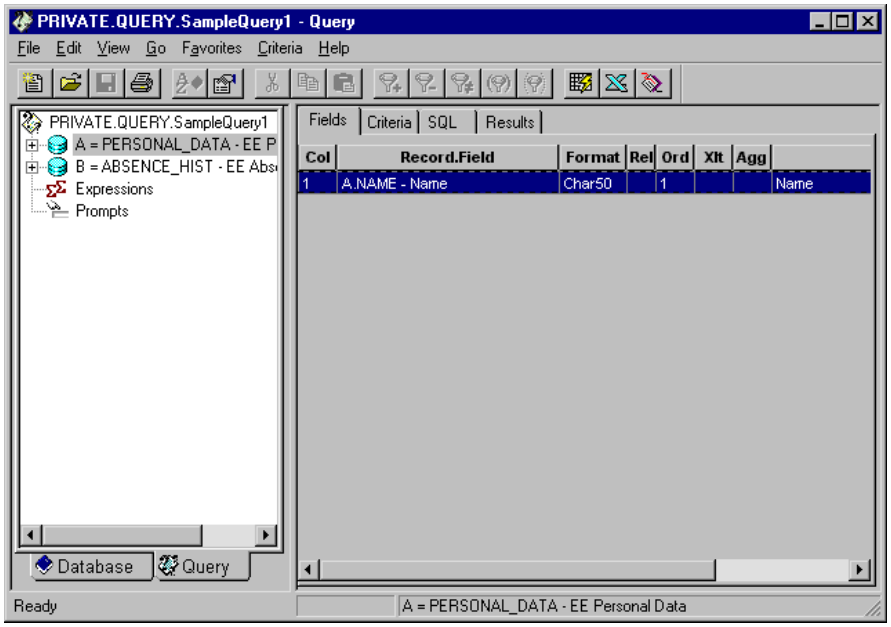
Ready
A = PERSONAL_DATA - EE Personal Data

In Selected List view

In the SQL statement that PeopleSoft Query builds, the fields in this view follow the SELECT command.

In Order by Option

In this view, you see all the fields in the entire query that are being used to control the display order for the output rows. For example, this query uses the NAME field to order the output, and we see this field displayed in the designer view, regardless of which record is selected in the component view.


PRIVATE.QUERY.SampleQuery1 - Query
File Edit View Go Favorites Criteria Help
Fields
Criteria
SQL
Results
Database
Query

PRIVATE.QUERY.SampleQuery1
A = PERSONAL_DATA - EE P
B = ABSENCE_HIST - EE Abs
Expressions
Prompts

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
1	A.NAME - Name	Char50		1			Name

Ready
A = PERSONAL_DATA - EE Personal Data

In Order By view

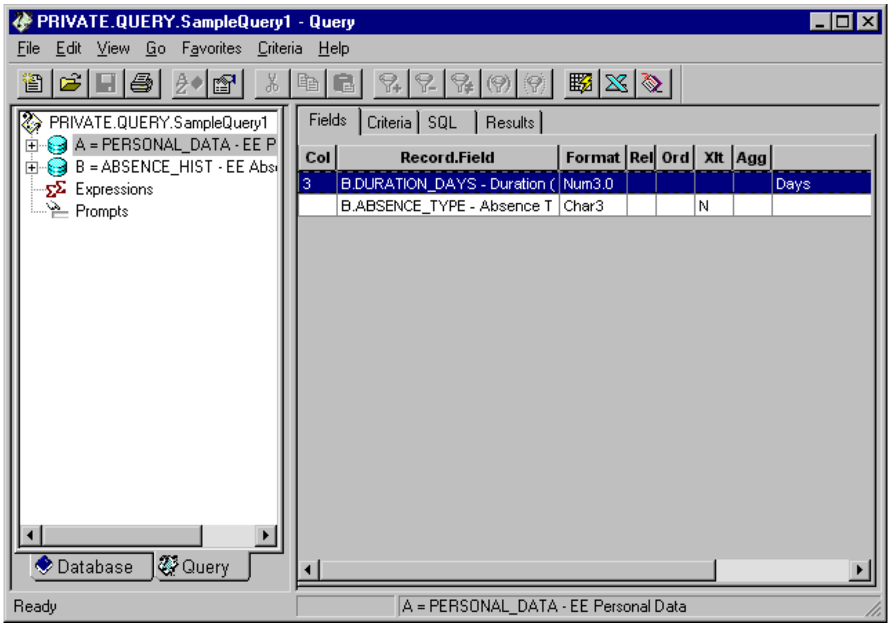
In the SQL statement that PeopleSoft Query builds, the fields in this view make up the ORDER BY clause.

For more information about specifying "order by" fields, see Formatting Query Output.

With Criteria Option

In this view, you see all the fields in the entire query that have selection criteria associated with them. In our example, the query has two fields with criteria, ABSENCE_TYPE and DURATION_DAYS, and we see them displayed in the designer view on the Criteria tab. The display of fields with criteria is shown, regardless of which record is selected in the component view.

For more information, see Specifying Query Selection Criteria.


PRIVATE.QUERY.SampleQuery1 - Query
File Edit View Go Favorites Criteria Help
PRIVATE.QUERY.SampleQuery1
A = PERSONAL_DATA - EE P
B = ABSENCE_HIST - EE Abs
Expressions
Prompts
Database
Query
Fields
Criteria
SQL
Results

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
3	B.DURATION_DAYS - Duration (	Num3.0					Days
	B.ABSENCE_TYPE - Absence T	Char3			N		

Ready

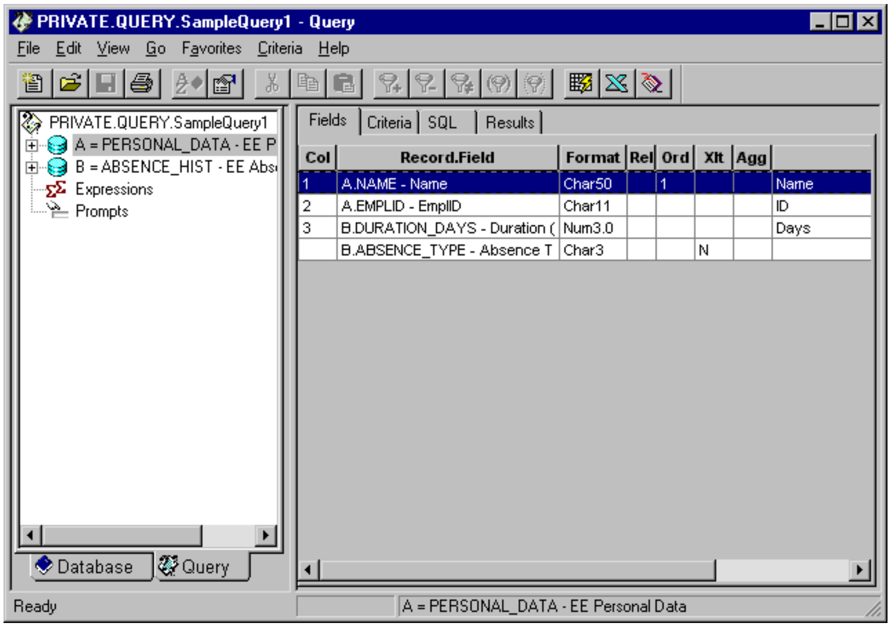
A = PERSONAL_DATA - EE Personal Data

With Criteria view. No fields are shown because none have criteria.

In the SQL statement that PeopleSoft Query builds, the fields in this view make up the WHERE clause.

View by All Option

In this view, you see a combination of the previous views, that is, all the fields involved in the entire query—those that have been selected, have ordering, or have criteria associated with them. In our example, there are a total of four fields involved in this query, and we see all fields, regardless of which record is selected in the component view.


PRIVATE.QUERY.SampleQuery1 - Query
File Edit View Go Favorites Criteria Help
PRIVATE.QUERY.SampleQuery1
A = PERSONAL_DATA - EE P
B = ABSENCE_HIST - EE Abs
Expressions
Prompts
Database
Query
Fields
Criteria
SQL
Results
Ready

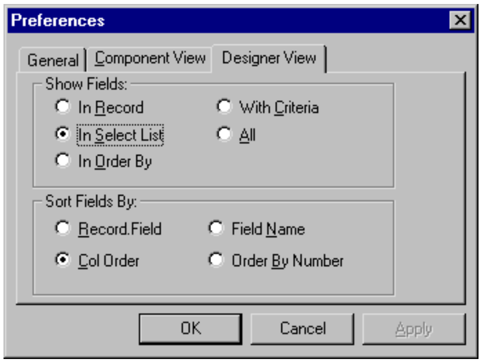
Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
1	A.NAME - Name	Char50		1			Name
2	A.EMPLID - EmplID	Char11					ID
3	B.DURATION_DAYS - Duration (	Num3.0					Days
	B.ABSENCE_TYPE - Absence T	Char3			N		

A = PERSONAL_DATA - EE Personal Data

All Fields view

Sort Field Options

The options in the Sort Fields By group box enable you to determine the order of the fields in the designer view. You can use any Sort Fields By option with any Show Fields option, although some sort options work particularly well with certain field options; for example, the Order By Number option works well with the In Order By view.



Show Fields:

○ In Record ○ With Criteria
● In Select List ○ All
○ In Order By

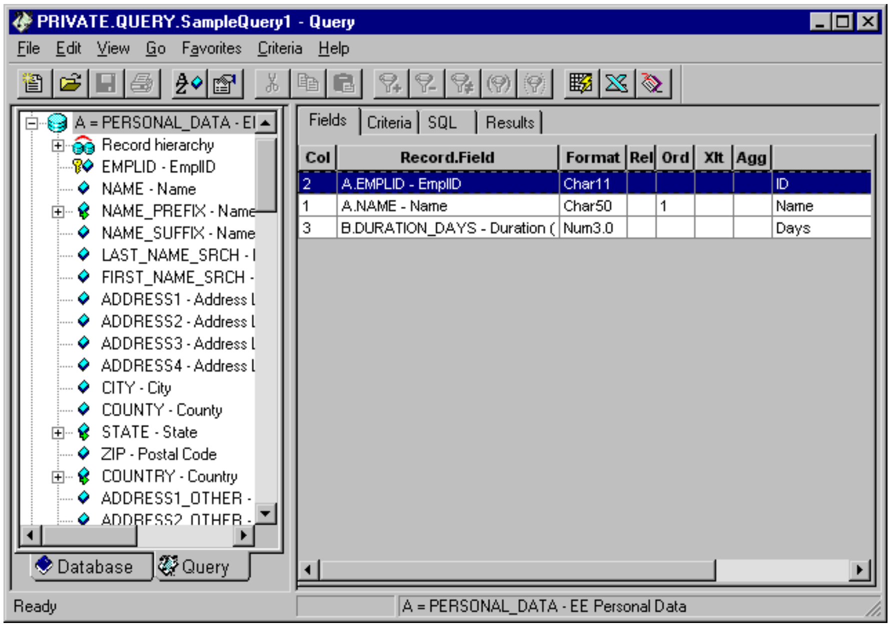
Sort Fields By:

○ Record.Field ○ Field Name
● Col Order ○ Order By Number

Preferences dialog box – Designer View tab

Record.Field

The **Record.Field** option displays fields in the order they appear in the record components. If your query uses fields from more than one record, the fields from record A are sorted before those of record B, and so on.

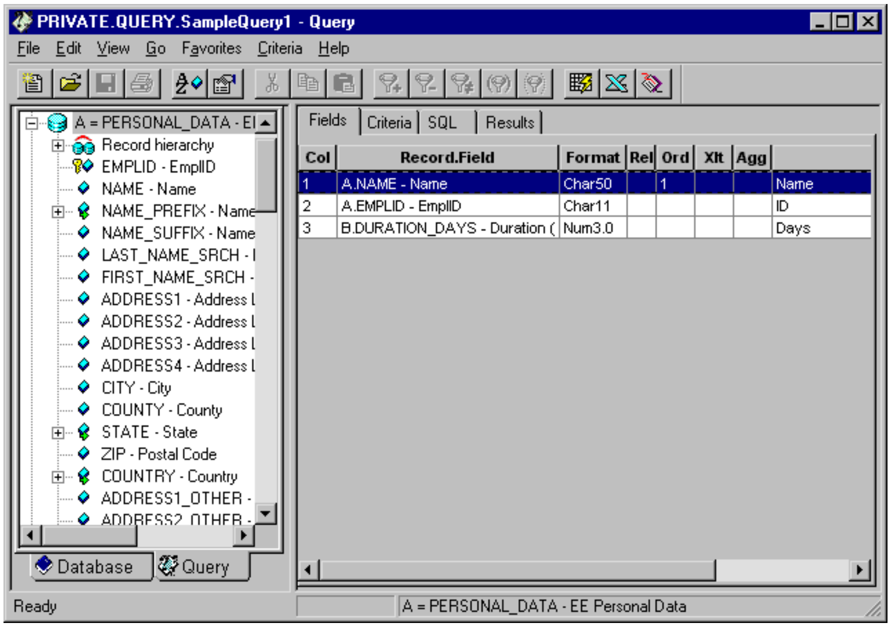


Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
2	A.EMPLID - EmplID	Char11					ID
1	A.NAME - Name	Char50		1			Name
3	B.DURATION_DAYS - Duration (	Num3.0					Days

“Record.Field” view

Col Order

The Col Order option displays the fields in your query in the order they will be displayed in your output. If you’re using a view that displays fields that you haven’t selected for output, these fields follow the selected fields and appear in record order. Col Order is the default option.



Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
1	A.NAME - Name	Char50		1			Name
2	A.EMPLID - EmplID	Char11					ID
3	B.DURATION_DAYS - Duration (	Num3.0					Days

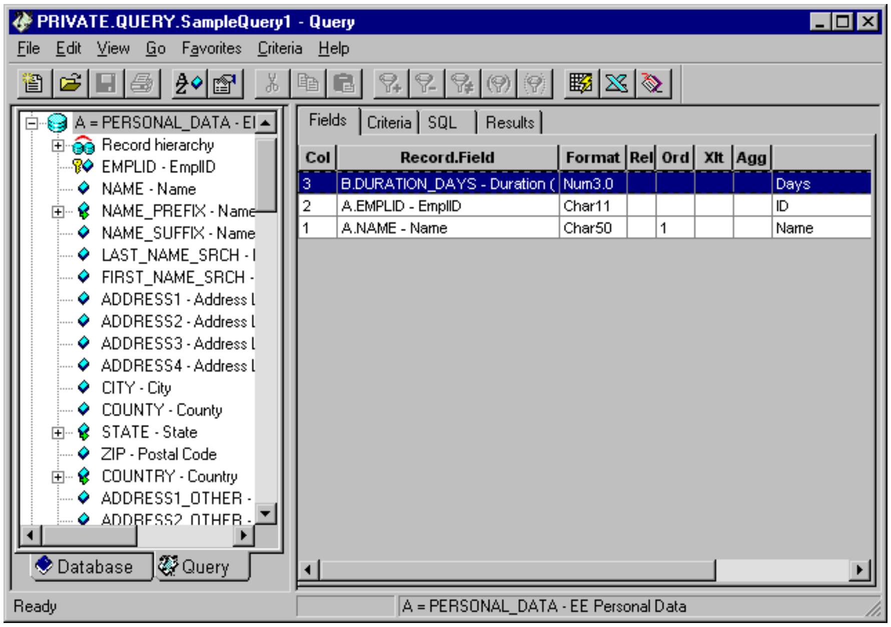
Col Order view

When the **Col Order** view option is active, you can change the column order by dragging and dropping the field rows.

For more information, see Formatting Query Output.

Field Name

The Field Name option displays the fields in alphabetical order, regardless of which record component the field came from.



Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
3	B.DURATION_DAYS - Duration (	Num3.0					Days
2	A.EMPLID - EmplID	Char11					ID
1	A.NAME - Name	Char50		1			Name

Ready

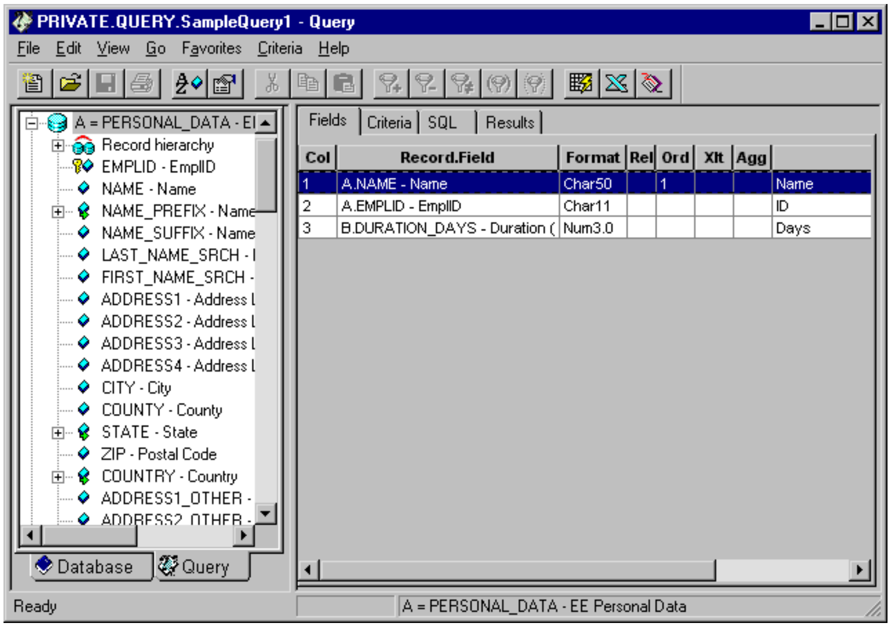
A = PERSONAL_DATA - EE Personal Data

Field Name view

Order By Number

The Order By Number option displays the fields you've selected to use for ordering the output rows at the top of the list, followed by the rest of the fields in record order. The Ord column shows which fields you've selected for ordering.

For more information on sort order fields, see Formatting Query Output.



Col	Record.Field	Format	Rel	Ord	Xlt	Agg	
1	A.NAME - Name	Char50		1			Name
2	A.EMPLID - EmplID	Char11					ID
3	B.DURATION_DAYS - Duration (	Num3.0					Days

Order by Number view

Result Tab Options

You use the Result Tab options to determine whether or not to show row headings on the Result tab. You can use the Run query when result tab is selected option to automatically run the current query anytime the user clicks the Result tab.

Formatting Query Output

Once you've selected the fields to include in the query, you're ready to make the query output pretty. PeopleSoft Query offers several options for formatting the query results. You can do the following:

- Change the column headings.
- Display Translate table values in place of codes.
- Specify a sort order for the result rows.

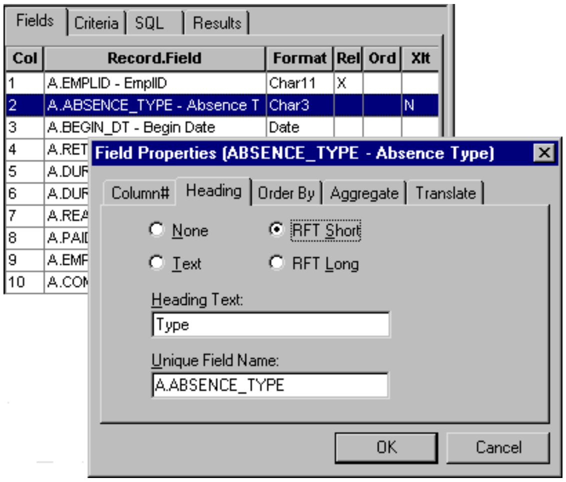
Headings

When you select a field for query output, Query automatically assigns a heading to display at the top of the column. By default, it uses the short name of the field from the record definition (RFT Short). You can, however, change the heading to any text you'd like.

To display or change column headings:

7. Right-click anywhere on the field in the designer view, and select **Heading** from the pop-up menu.

   The Field Properties dialog box appears, with the Heading tab active.



Col	Record.Field	Format	Rel	Ord	Xlt
1	A.EMPLID - EmplID	Char11	X		
2	A.ABSENCE_TYPE - Absence T	Char3			N
3	A.BEGIN_DT - Begin Date	Date			
4	A.RET				
5	A.DUF				
6	A.DUF				
7	A.REA				
8	A.PAIL				
9	A.EMF				
10	A.COM				

Field Properties (ABSENCE_TYPE - Absence Type)

○ None ● RFT Short

○ Text ○ RFT Long

Heading Text: Type

Unique Field Name: A.ABSENCE_TYPE

Changing column headings

   This dialog box is where you can provide a heading text and a unique field name. If you do not provide a unique field name, Query provides a default, which is generally acceptable. The default name will be the record field name appended with the record name.

8. Select the appropriate heading option.

   The following list describes the options available for column headings.

None The column does not have a heading.

Text The column heading is the text you've entered in the text box.

RFT Short The column heading is the short name from the record definition.

RFT Long The column heading is the long name from the record definition.

Heading Text Text displayed at the top of the column.

Unique Field Name Used for translations. There is no need to change this field.

Note. If you start typing text into the text box, Query automatically selects the **Text** option.

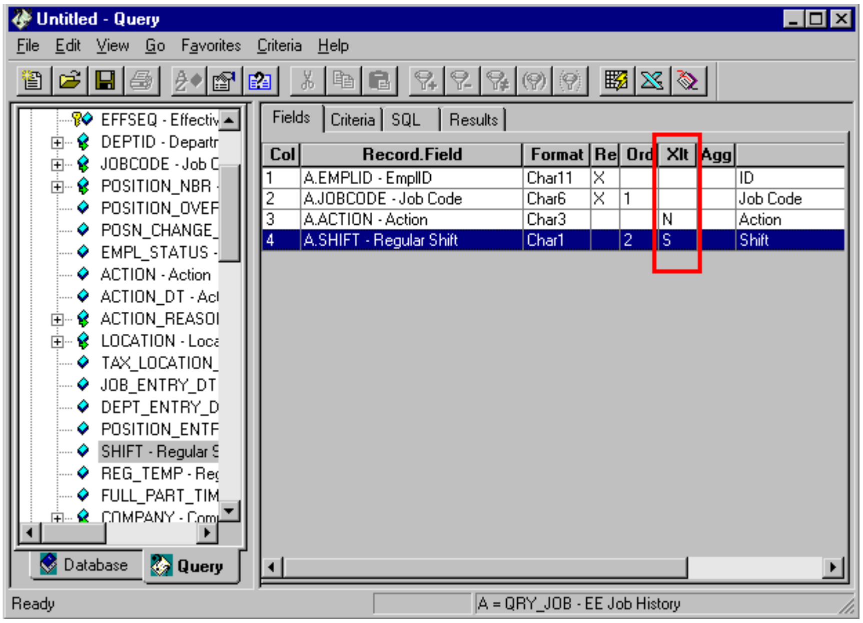
PeopleSoft Query provides globalization features such as language-sensitive query names and column headings. For more information, see Globalization.

Translate Table Values

The table you're querying may include fields that use the Translate table. If so, it means that the field itself contains a short code of some kind, for which the Translate table provides a set of corresponding values. For example, if the table includes an EFF_STATUS field, the field has one of the values *A* or *I*, the Translate table "translates" these codes into *Active* and *Inactive*.

In your query results, you might want to display the translated value rather than the code; for example, Active instead of A. It's easy to tell PeopleSoft Query to make this substitution.

If a field has values on the Translate table, a letter appears in the Xlt column for that field. By default, the letter is *N* (for none), meaning that Query will not be replacing the codes with the corresponding translate values.

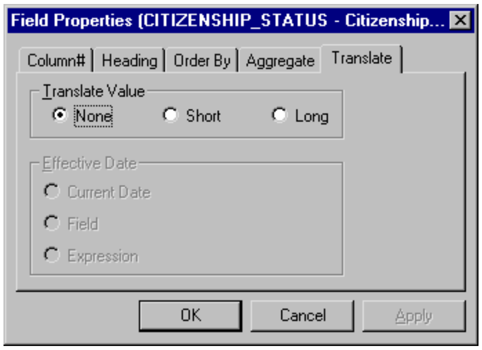


Col	Record.Field	Format	Re	Ord	Xlt	Agg	
1	A.EMPLID - EmplID	Char11	X				ID
2	A.JOBCODE - Job Code	Char6	X	1			Job Code
3	A.ACTION - Action	Char3			N		Action
4	A.SHIFT - Regular Shift	Char1		2	S		Shift

Ready   A = QRY_JOB - EE Job History

XLT Column

To use the Translate table values for a field, right-click anywhere on the field in the designer view, and select ***Translate*** from the pop-up menu. The Field Properties dialog box appears with the **Translate** tab active.


Field Properties (CITIZENSHIP_STATUS - Citizenship...
Column# | Heading | Order By | Aggregate | Translate
Translate Value
● None ○ Short ○ Long
Effective Date
○ Current Date
○ Field
○ Expression
OK
Cancel
Apply

Field Properties dialog box – Translate tab

Specify which translate value you want to display in the query results—***None, Short,*** or ***Long.***

Translate tables are effective-dated, so you need to select which effective date to use for it. For most tables, Query defaults to the current date, meaning that it will use the currently active list of Translate table values. However, if the table you're querying is also effective-dated, Query defaults to the value in the EFFDT field for a row; that is, for each row the query returns, Query will use the Translate table values that were active as of that row's effective date.

If neither of these effective date options are what you want, you have two more:

- If the table you're querying includes another date field, you can use the value in that field as the effective date for Translate table values. Select the Field radio button, then select the field name from the drop-down list box that appears next to it.

- You can use an expression to set the effective date for the Translate table. For example, you could enter a fixed effective date or prompt the user for one. Select the Expression radio button, then enter an expression in the dialog box that pops up.

For more information about expressions, see Advanced Query Options.

Click the **OK** button to close the Translate dialog box. In the designer view, an N, L, or S appears in the Xlt column to indicate which translate values will be used, if any.

Sort Order

You can select one or more fields to sort on for your query output.

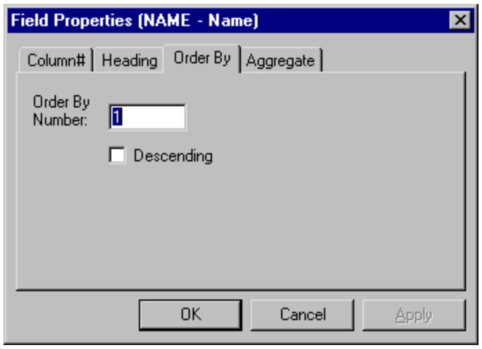
The simplest way to pick an "order by" field is to double-click in the **Ord** column on the field in the designer view. PeopleSoft Query puts a number in the column: if this field is the first sort field you've selected, a *1* appears in the Ord column, and the system will sort rows based on this field first. The next sort field you select will be numbered *2*, and so on. You can sort by as many fields as you like. Query will sort the rows by the first sort field, then use the second field to sort rows with the same value in the first field.

Note. The fields you use to sort the output rows don't have to be displayed. That is, you can choose fields that you haven't selected for output.

You can also change the sort order by right-clicking the field's row and selecting Order By from the pop-up menu.

The Field Properties dialog box appears with the Order By tab active.

Enter a value in the Order By Number field and then click OK.


Field Properties (NAME - Name)

Order By Number: 1

☐ Descending

Field Properties – Order By tab

This dialog box enables you to specify which sort field you want this field to be.

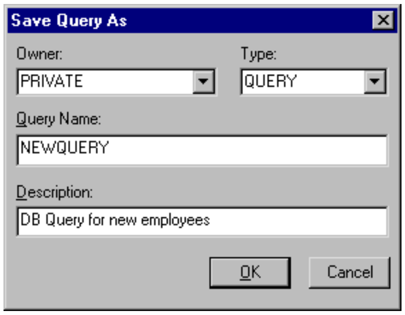
It also enables you to specify a descending sort by selecting the **Descending** check box. When you select a sort field by double clicking in the **Ord** column, Query sorts the rows in ascending order. If you want Query to sort them in descending order, you need to use the **Field Properties** dialog box to choose the sort field.

Note. When you're picking sort fields, you might find it useful to select the **Order By Number** option, which displays the sort fields in order.

Saving Queries

Once you've designed a query, you may want to save it so you can run it again later. To save your query either select **File, Save** (or **File, Save As**), or click the **Save** button .

The **Save Query As** dialog box appears.



Owner: PRIVATE

Type: QUERY

Query Name: NEWQUERY

Description: DB Query for new employees

Save Query As dialog box

You use this dialog box to make a copy of an existing query, saving it under a different name.

Owner A Private query allows only the creator of the query to open, run, modify, or delete the query. A Public query can be open, run, modified, or deleted by any user with access to those records involved in the query.

Type Generally, the type Query is used to denote ad hoc queries. There are also workflow queries —ROLEQRY and DBAGQRY—that are used in workflow queries. If you do not have access to create workflow queries, you will not see these options.

Query Name Make the name as descriptive as possible. Query names are uppercase and can be up to thirty characters in length. You cannot use special characters (commas, slashes, and so on) or include spaces (other than an underscore) in query names.

Description You can add a description to your query using up to thirty character, mixed case, and you may use special characters.

Note. You can click the **Properties** button at any time to chage an open query’s Owner and type.

PeopleSoft Query provides globalization features such as language-sensitive query names and column headings. For more information, see Globalization.

To save a query:

9. Specify the Owner and Type of query you want this to be.

10. Type a name in the **Query Name** field.

11. Type a narrative description of the query in the **Description** field.

12. After you've saved a query for the first time, select **File, Save** to save the currently open query using the same name, overwriting the previously saved version.

Use this feature to save your changes as you refine your query.

For more information about query profiles, see Setting up Query Security.

Printing Queries

It's a good idea to maintain documentation about the queries you create; particularly common ones that you run all the time. When someone has a question about what exactly a query is doing, they can refer to the documentation. PeopleSoft Query enables you to print a query object (meaning the definition of the query, not the results).

To print a query, select **File, Print (Current Query Object)** to print the query that you currently have open.

Running Queries

After designing a query, you'll want to run it to see the results. Query provides several output options, each of which can be extremely useful depending on your needs.

Running Predefined Queries

Sometimes all you need PeopleSoft Query for is to run a query that you or someone else has previously defined. You can run a query from any PeopleTools application and you may also run a query from the PeopleSoft Internet Architecture.

For more information about running a query from your browser, see Running a Query on the Web.

To run an existing query:

13. From any PeopleSoft application, select **Go, PeopleTools, Query.**

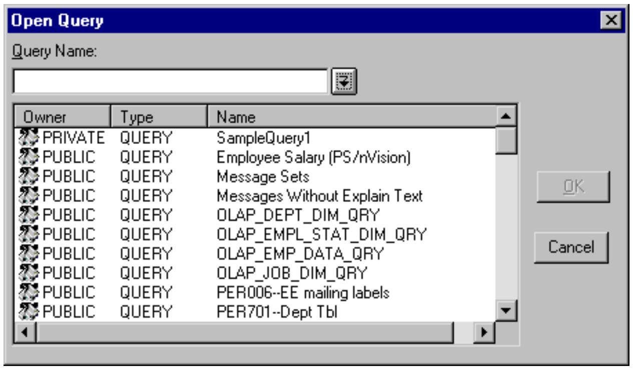
    PeopleSoft Query appears.

14. Select **File, Open** or click 📂 to display the **Open Query** dialog box.

You must open a query before you can run it. The list of queries will initially be blank, but once you specify a query name (or click the drop-down button), you'll see all the *private* queries you've saved, plus all *public* ones that are accessible to you.

Note. A public query is accessible when: 1) you have been granted access to the public query through Object Security; and 2) you have access to all records used by the public query.

You can get a partial list of available queries by entering the first several characters of the query name, and then clicking the list box button.



Owner	Type	Name
PRIVATE	QUERY	SampleQuery1
PUBLIC	QUERY	Employee Salary (PS/nVision)
PUBLIC	QUERY	Message Sets
PUBLIC	QUERY	Messages Without Explain Text
PUBLIC	QUERY	OLAP_DEPT_DIM_QRY
PUBLIC	QUERY	OLAP_EMPL_STAT_DIM_QRY
PUBLIC	QUERY	OLAP_EMP_DATA_QRY
PUBLIC	QUERY	OLAP_JOB_DIM_QRY
PUBLIC	QUERY	PER006--EE mailing labels
PUBLIC	QUERY	PER701--Dept Tbl

Open Query dialog box

Note. The **Query Name** is case-sensitive, so be sure to accurately enter the first several characters of the desired query before clicking the list box button.

15. Highlight the query you want to open in the list box, and then click **OK**.

    You can also double-click on the query name to open it. The query appears in the component view, and the designer view displays the fields as specified in your preferences.

For more information on view preferences, see View Preferences.

16. Select **File, Run (Current Query)** or click to run the query.

    This will send query results to the grid (on the **Results** tab) for immediate viewing. If you want to select one of Query's other output options, see "Selecting an Output Option" below.

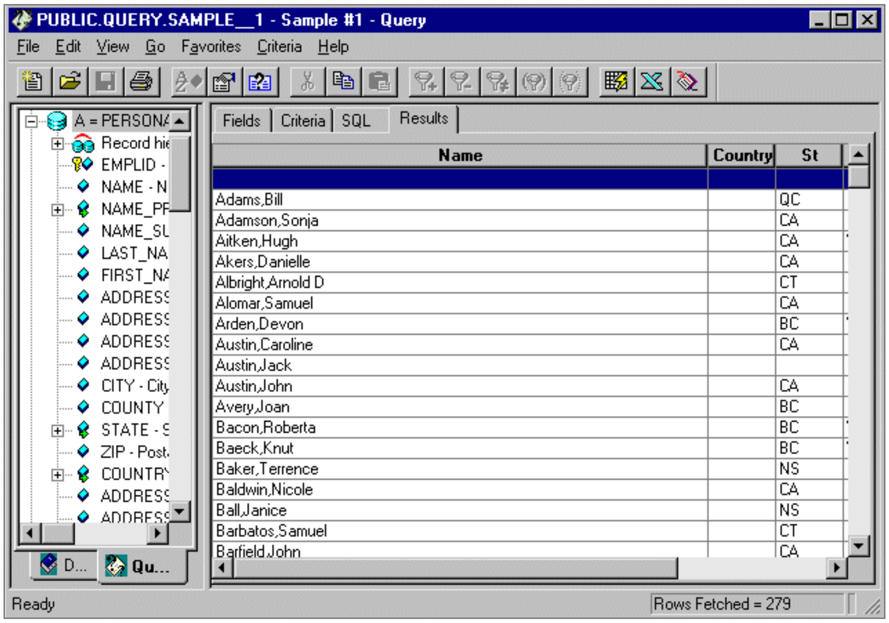
Note. You can also set your view preferences to run the current query when you click on the **Result** tab in the designer view.

Once you have a result set displayed, you can click on any column heading to order the result set by the values in that column. Clicking the same column heading again will restore the result set to its original order.

Note. If you can't find a public query in the Open Query dialog box, it most likely means that the query uses a record definition you haven't been granted access to.

Viewing Query Results

By default, Query results displayed on the Result tab in the designer view for immediate viewing. If you want to select one of Query's other output options, see "Selecting an Output Option" below.



Name	Country	St
Adams,Bill		QC
Adamson,Sonja		CA
Aitken,Hugh		CA
Akers,Danielle		CA
Albright,Arnold D		CT
Alomar,Samuel		CA
Arden,Devon		BC
Austin,Caroline		CA
Austin,Jack		
Austin,John		CA
Avery,Joan		BC
Bacon,Roberta		BC
Baeck,Knut		BC
Baker,Terrence		NS
Baldwin,Nicole		CA
Ball,Janice		NS
Barbatos,Samuel		CT
Barfield,John		CA

Ready

Rows Fetched = 279

Sample #1 Query - Results tab

Selecting an Output Option

PeopleSoft Query can pass the results of a query to three different places. The output options are as follows.

Output Option	*Toolbar Button*	*Description*
Result tab		Select **File, Run (Current Query)** or click the toolbar button to send the output to the Results tab of the designer view. This option is useful to get just the

		results you want as you design and refine queriesWhen your query is perfect, you can use one of the other output options.
Microsoft Excel		Select **File, Run to Excel** or click the toolbar button to send the output to Microsoft Excel (using QueryLink, discussed below).
Crystal Reports		Select **File, Run to Crystal** or click the toolbar button to send the output to Crystal Reports, a report formatter, where you can use predefined formats or create new ones to print the results of your query.

The Windows clipboard is essentially a fourth output option.

To select an output option:

17. Select the result rows from the grid that you wish to copy to the clipboard.

You can select a range of rows by holding down the SHIFT key and dragging the cursor down. Individual rows can be select by holding down the CTRL key and clicking to select each row.

18. After selecting the rows you want to copy, click the **Copy** button on the toolbar.

QueryLink

QueryLink enables you to send the results of a query directly to a Microsoft Excel spreadsheet, bypassing the need for a PS/nVision layout. Any query results sent to Excel through QueryLink are based on the QUERY.XLT Excel template. Therefore, if you want to make any permanent changes to spreadsheets derived through QueryLink, make the changes to QUERY.XLT.

For more information**,** see Comparing Tabular Layouts, Matrix Layouts, and QueryLink. You'll find a table that contrasts the features and functions of PS/nVision layouts and QueryLink.

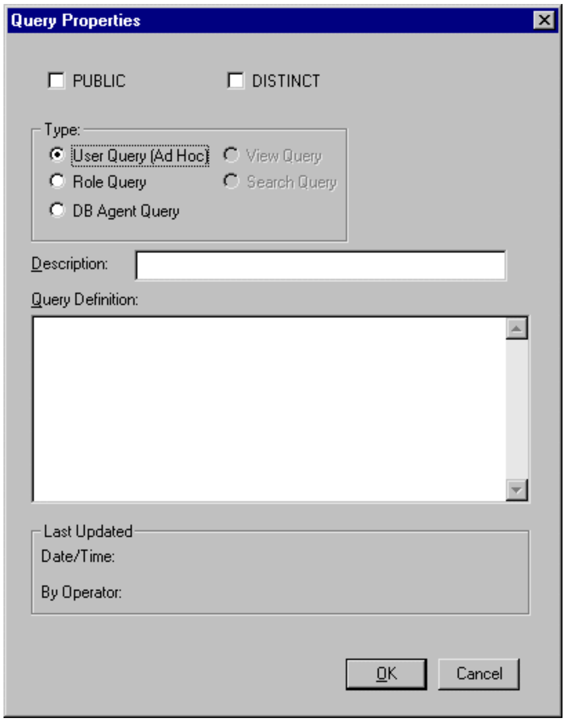
Changing Query Properties

You can change the properties of a query at any time.

To change query properties:

19. Either select **File, Properties** or click the **Properties** button on the toolbar to view or change the properties of the currently open query.

The Query Properties dialog box appears.



□ PUBLIC □ DISTINCT

Type:
● User Query (Ad Hoc) ○ View Query
○ Role Query ○ Search Query
○ DB Agent Query

Description:

Query Definition:

Last Updated
Date/Time:
By Operator:

Query Properties dialog box

You use this dialog to define the properties of a query.

Selecting the PUBLIC option enables other users to access the query, if they have access to public queries.

Note. To bring up the Query Properties dialog box, you can also right-click on the query component and select Properties from the pop-up menu, or select **Edit, Properties** (with the query component selected).

For more information on query security, see Setting up Query Security. For more information on the DISTINCT option, see Advanced Query Options.

The query type can be any of the following:

- **User Query** (ad hoc). This is a “regular” query (any query which is not one of the special types of queries listed below is a user query.)
- Role Query.

- DB Agent Query.

For more information on Role Queries and DB Agent Queries, see Advanced Query Options.

The View Query button is only valid when creating a view query from the Application Designer. The Search Query button is only valid when creating a search query from the Windows client. If you launched Query directly from the **Go, PeopleTools** menu, then these options are grayed out.

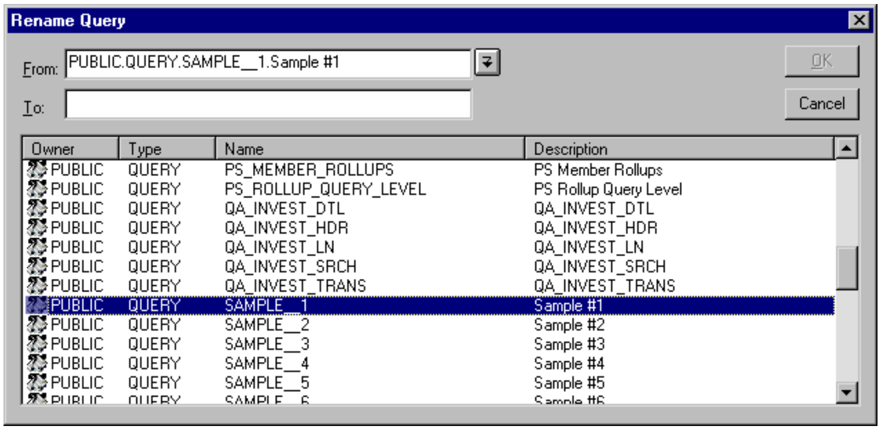
Renaming Queries

You can change the name of a query at any time.

To rename an existing query:

20. Select **File**, **Rename**.

The **Rename Query** dialog box appears.



From: PUBLIC.QUERY.SAMPLE__1.Sample #1

To:

Owner	Type	Name	Description
PUBLIC	QUERY	PS_MEMBER_ROLLUPS	PS Member Rollups
PUBLIC	QUERY	PS_ROLLUP_QUERY_LEVEL	PS Rollup Query Level
PUBLIC	QUERY	QA_INVEST_DTL	QA_INVEST_DTL
PUBLIC	QUERY	QA_INVEST_HDR	QA_INVEST_HDR
PUBLIC	QUERY	QA_INVEST_LN	QA_INVEST_LN
PUBLIC	QUERY	QA_INVEST_SRCH	QA_INVEST_SRCH
PUBLIC	QUERY	QA_INVEST_TRANS	QA_INVEST_TRANS
PUBLIC	QUERY	SAMPLE__1	Sample #1
PUBLIC	QUERY	SAMPLE__2	Sample #2
PUBLIC	QUERY	SAMPLE__3	Sample #3
PUBLIC	QUERY	SAMPLE__4	Sample #4
PUBLIC	QUERY	SAMPLE__5	Sample #5
PUBLIC	QUERY	SAMPLE__6	Sample #6

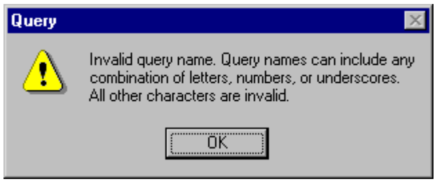
Rename Query dialog box

The **Rename Query** dialog box is where you select an existing query and specify a new name for it.

21. Type the first few letters of the query you are interested in or simply click the button to return a list of available queries.

22. Click once on the query you want to rename from the list, then type a new name in the **To:** field.

Note. Query names can include letters, numbers, and underscores. If you use any other invalid characters (such as a period or asterisk), the following message appears.



Query

Invalid query name. Query names can include any combination of letters, numbers, or underscores. All other characters are invalid.

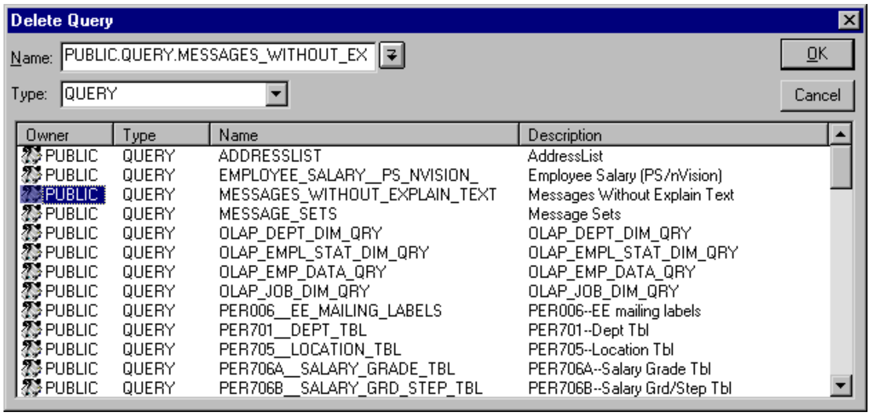
23. Once you have entered a valid query name, click OK.

Deleting Queries

To delete a query:

24. Select **File**, **Delete**.

    The **Delete Query** dialog box appears.



Delete Query

Name: PUBLIC.QUERY.MESSAGES_WITHOUT_EX

Type: QUERY

Owner	Type	Name	Description
PUBLIC	QUERY	ADDRESSLIST	AddressList
PUBLIC	QUERY	EMPLOYEE_SALARY__PS_NVISION_	Employee Salary (PS/nVision)
PUBLIC	QUERY	MESSAGES_WITHOUT_EXPLAIN_TEXT	Messages Without Explain Text
PUBLIC	QUERY	MESSAGE_SETS	Message Sets
PUBLIC	QUERY	OLAP_DEPT_DIM_QRY	OLAP_DEPT_DIM_QRY
PUBLIC	QUERY	OLAP_EMPL_STAT_DIM_QRY	OLAP_EMPL_STAT_DIM_QRY
PUBLIC	QUERY	OLAP_EMP_DATA_QRY	OLAP_EMP_DATA_QRY
PUBLIC	QUERY	OLAP_JOB_DIM_QRY	OLAP_JOB_DIM_QRY
PUBLIC	QUERY	PER006__EE_MAILING_LABELS	PER006--EE mailing labels
PUBLIC	QUERY	PER701__DEPT_TBL	PER701--Dept Tbl
PUBLIC	QUERY	PER705__LOCATION_TBL	PER705--Location Tbl
PUBLIC	QUERY	PER706A__SALARY_GRADE_TBL	PER706A--Salary Grade Tbl
PUBLIC	QUERY	PER706B__SALARY_GRD_STEP_TBL	PER706B--Salary Grd/Step Tbl

Delete Query dialog box

25. Search for the query or enter the query name.

26. Click OK.

You use this dialog box in the same way as the Open Query dialog box.

CHAPTER 4

Specifying Query Selection Criteria

In most cases, you don't want to retrieve every row of data in a table. Your database contains a lot of information, and a complete list of entries is unlikely to answer the question that's motivating you to write a query.

To selectively retrieve just the data you want, you define *selection criteria*. Selection criteria refine your query by specifying conditions that the retrieved data must meet. For example, you can specify that the system only retrieve phone numbers with a certain area code instead of all phone numbers. In this chapter, you'll learn how to define selection criteria for your queries.

Note. If you are familiar with SQL, it might help to know that a criteria expression defines a WHERE clause for the query's SELECT statement.

Choosing Selection Criteria

Because your PeopleSoft database stores data in tables, you can identify every individual piece of data by saying what *column* (field) and *row* (record) it's in. When you create a query, you pick the data you want by specifying which columns and rows you want the system to retrieve.

You select which *columns* of data you want using the query component view and the Fields tab. You select the columns by choosing fields from record components.

For more information**,** see Creating and Running Simple Queries

If you run the query after selecting the fields, the system retrieves *all* the data in those columns; that is, it retrieves the data from every row in the table or tables.

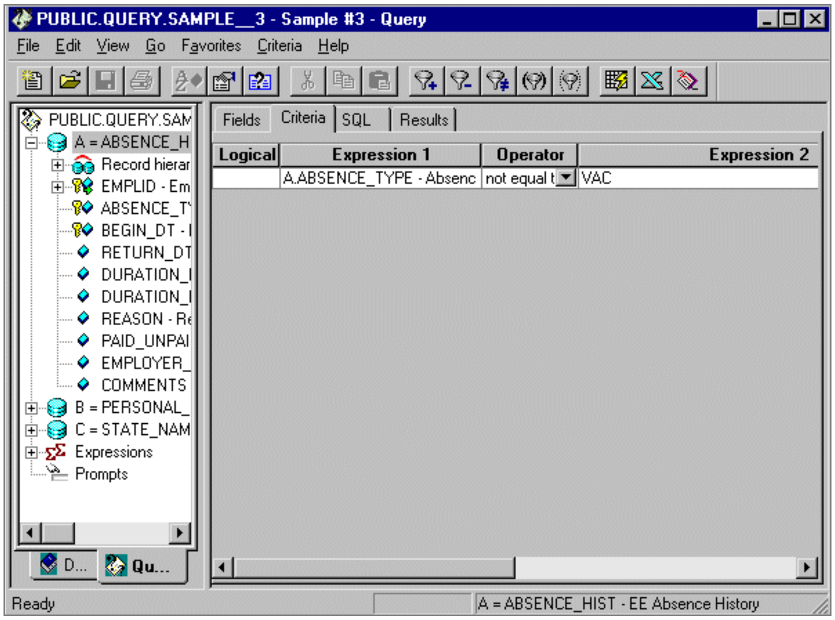
You select which *rows* of data you want by adding selection criteria to the query, using the Criteria tab in Query's designer view. The selection criteria serve as a test that the system applies to each row of data in the tables you're querying. If the row passes the test, the system retrieves it; if the row doesn't pass, the system doesn't retrieve it.

For example, suppose you needed the names of all PeopleStore customers who aren't PeopleSoft employees. You'd start by creating a query that retrieves the Name and Company fields from the Customer table. At this point, you could run the query to get a list of all customers, then scan the list for rows where the company name is not *PeopleSoft*. But wouldn't it be better to get the system to do this work for you? You can add a selection criterion that tells Query to scan for rows where the company name is not *PeopleSoft*.

In most cases, a selection criterion compares the value in one of a row's fields to a reference value. In the example from the previous paragraph, you would compare the value in the Company field to the constant value *PeopleSoft*. In other situations, you might compare the value to the value in another record field or to a value that the user enters when running the query.

Using the Criteria Tab

You define selection criteria using the Criteria tab in Query's designer view. Here's what the Criteria tab looks like for our example query, Sample #3 Query.



Logical	Expression 1	Operator	Expression 2
	A.ABSENCE_TYPE - Absenc	not equal t	VAC

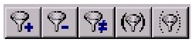
Sample #3 Query – Criteria tab

On the **Criteria** tab, each field's row defines one criterion expression. There are four columns on this tab that you use to define the specifics of each criterion: **Logical, Expression 1, Operator,** and **Expression 2.** These columns and their descriptions are listed in the table below. Because there are two columns labeled *Expression*, we've numbered them 1 and 2 in the table. Expression (1) refers to the first expression column (the one farther to the left), and Expression (2) refers to the second expression column.

Criteria	***Description***
Logical	Any rows after the first row must include either an *AND* or *OR* logical value in this column to specify whether you want the rows to meet this criterion in addition to other criteria you've defined or as an alternative criterion. The first criterion you define doesn't have a value in this box. The default for subsequent criteria is AND.

Criteria	***Description***
Expression (1)	In most cases, the left-hand expression (the one to the left of the Operator column) will be a field that's value you want to base the selection criterion on. You can also enter any expression text by right-clicking on the grid cell and selecting Expression.
Operator	Select the appropriate operator from the drop-down list, such *as equal to* or *greater than* in the example above. See Selecting Operators below for a complete list of operators and their descriptions.
Expression (2)	When you right-click in the second Expression column (the one to the right of the Operator column), a pop-up menu appears enabling you to choose from the available value types. Examples of value types are constants, fields, and runtime prompts. The available value types differ depending on which comparison operator you select. After you select a value type, a dialog box appropriate for the value type you've selected appears. This enables you to enter the value with which you want to compare Expression(1).

Notice that five special buttons on the Query toolbar become active when the **Criteria** tab is selected:



The meaning of each of these buttons is as follows:

Criteria Button	***Description***
	Adds a new row of criteria directly below the highlighted row.
	Deletes the currently highlighted row of criteria.
	Enables you to specify a criterion that a row should *not* meet. To negate a criterion, highlight it and click this button in the toolbar. (See Selecting Operators for an important note about this option.)
	Enables you to group criteria to ensure that Query checks them in the appropriate order. Left and right parentheses are used to designate a group of criteria.
	Enables you to ungroup criteria that were previously grouped.

For more information about defining a selection criterion, see the next section, Entering Selection Criteria. For details about using *AND* and *OR*, as well as using parentheses to group criteria, see Relating Multiple Criteria.

Entering Selection Criteria

In most cases, you don't want all the rows of data from the table, just the rows that meet certain conditions. To limit the returned rows, you define *selection criteria*. If you know some SQL, you'll notice that the criteria correspond to the WHERE clause of the SELECT statement.

To enter selection criteria:

1. Select the field component you want to base the selection criterion on (this will be the left-hand Expression).

Note. If necessary, you can also use any valid SQL expression on which to base the selection criterion.

To view or add selection criteria, click the Criteria tab in the designer view. This tab displays one row for each query criterion. Then select the field you want to base the selection criterion on from the query component view. There are several ways to select the field:

- Drag the field from the component view onto the Criteria tab in the designer view.
- Right-click on the field and choose **New Criteria** from the pop-up menu.
- Click once on the field and select **Edit**, **New Criteria** from the Query menu.

2. Select an operator.

The operator determines how Query compares the values of the first (left-hand) expression to the second (right-hand) expression. Click the drop-down button in the **Operator** column and select the appropriate comparison operator from the pop-up menu.

For more information about the available operators, see Selecting Operators.

3. Enter the comparison value or values in the second (right-hand) Expression column.

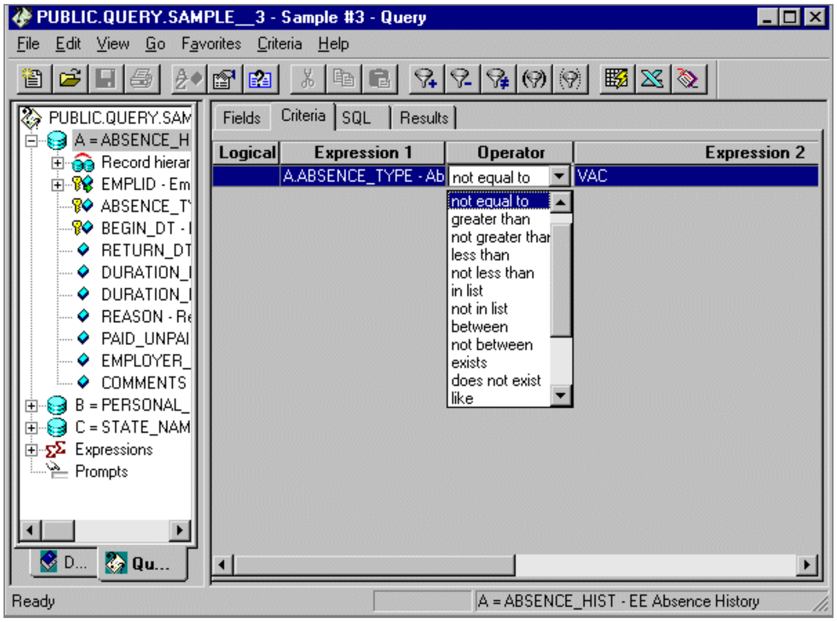
Entering comparison values is a two-step process. First, you specify what type of value you want to compare the selected field to by right clicking in the second Expression column and selecting a value type from the pop-up menu. Then, you enter the comparison value or values in the dialog box that appears. The procedure for entering values depends on what type of values you're entering.

For more information**,** see Entering Comparison Values.

If the field you're using has an edit table you can press F4 from the Constant dialog box to get a list of valid values. You can then select a constant from this list.

Selecting Operators

The operator specifies a relationship that needs to exist between the record field and the comparison values. You select an operator by clicking the drop-down arrow in the Operator column for the field, and then choosing the desired operator from the list.



Logical	Expression 1	Operator	Expression 2
	A.ABSENCE_TYPE - Ab	not equal to	VAC

Selecting an operator

The table below describes the available operators. For each of the operators, Query also offers a "not" option that reverses its effect. For example, ***not equal to*** returns all rows that ***equal to*** would *not* return.

Note. It's always a better idea to use the "not" version of an operator rather than the NOT operator on the entire criterion. When you use NOT, PeopleSoft Query can't use SQL indexes to speed up the data search. When you use the "not" version of an operator, Query can translate it into a SQL expression that allows it to use the indexes.

Operator	***When It Returns a Row***
equal to	The value in the selected record field exactly matches the comparison value.
greater than	The value in the record field is greater than the comparison value.
less than	The value in the record field is less than the comparison value.
in list	The value in the selected record field matches one of the comparison values in a list. See Entering Comparison Values for more information about creating the list.
between	The value in the selected record field falls between two comparison values. The range is inclusive.
exists	This operator is different from the others, in that it doesn't compare a record field to the comparison value. The comparison value is a subquery. If the subquery

Operator	***When It Returns a Row***
	returns any data, PeopleSoft Query returns the corresponding row. You don’t select a record component or field for this operator. For more information about subqueries, see Advanced Query Options.
like	The value in the selected field matches a specified string pattern. The comparison value may be a string that contains wildcard characters. The wildcard characters that Query recognizes are: % Matches any string of 0 or more characters. For example ‘C%’ matches any string starting with C, including C alone _ Matches any single character. For example, ‘_ones’ matches any five-character string ending with “ones,” such as Jones or Cones. Query also recognizes any wildcard characters that your database software supports. See your database management system documentation for details. To use one of the wildcard characters as a literal character (for example, to include a % in your string), precede the character with a \. For example, ‘percent\%\’.
is null	The selected record field doesn’t have a value in it. You don’t specify a comparison value for this operator. Key fields, required fields, Character fields, and Numeric fields do not allow null values.
in tree	The value in the selected record field appears as a node in a tree created with the Tree Manager. The comparison value for this operator is a tree or branch of a tree that you want Query to search. See Entering Comparison Values for more information about selecting which tree branches to use.

Note. If you’ve selected the EFFDT field on an effective-dated table, PeopleSoft Query also offers some special effective date operators.

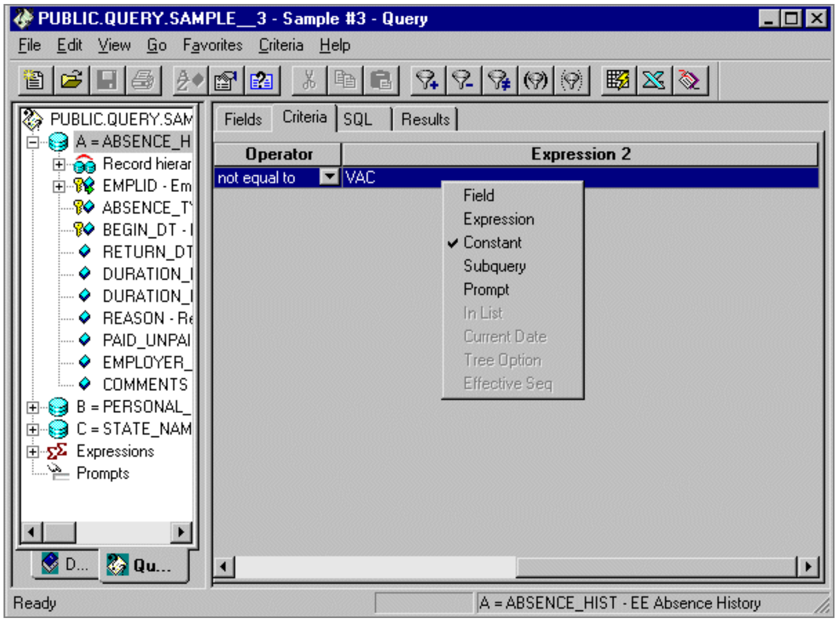
For more information, see Specifying Effective Date Criteria.

Entering Comparison Values

The procedure for entering comparison values differs depending on what kind of value you're entering. If you're comparing one field to another, you need to pick the second record field; if you're comparing the rows to a constant value, you need to enter the constant.

To enter comparison values:

4. Select the value type by right clicking in the second **Expression** column for the field's row on the **Criteria** tab.



Sample #3 Query – Expression 2 list

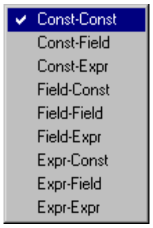
The table below describes all the available value types.

Value Type	***What Query Compares the Value in the Selected Field To***
Field	The value in another field, usually a field in another record component. To compare the values from fields in two records, you need to join the record components. See Advanced Query Options.
Expression	An expression you enter, which Query evaluates before comparing the result to the value in the selected field. Query

Value Type	***What Query Compares the Value in the Selected Field To***
	evaluates the expression once for each row it tests.
Constant	A single fixed value.
Subquery	The data returned by a subquery. For information about subqueries, see Advanced Query Options.
Prompt	A value that a user enters when he or she runs the query. The system displays a dialog box asking for a value.
In List	A list of values that you enter. This value type is available only when the selected operator is *in list* or *not in list*.
Current Date	The current date on the database server.
Tree Option	A selected set of tree nodes. This value type is available only when the selected operator is *in tree* or *not in tree*.
Effective Seq	Used on some effective-dated records, the effective sequence is a sequencing number provided to further refine the effective date.

Note. Not all value types are available for all operators. For example, when you select the *exists* operator, Subquery is the only available value type. After you select an operator, PeopleSoft Query displays only the value types that are available for that operator.

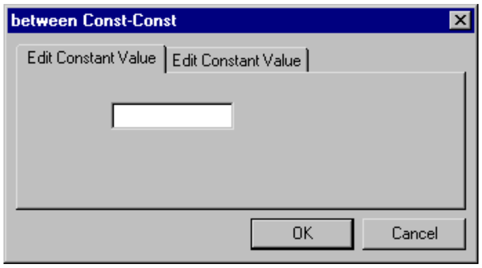
5. When you select the *between* operator, the list of value types is somewhat different than the list used for the other operators.



Value List for the *between* operator

6. After you make your selection from the list above, a *between* dialog box appears, enabling you to enter the appropriate criteria.

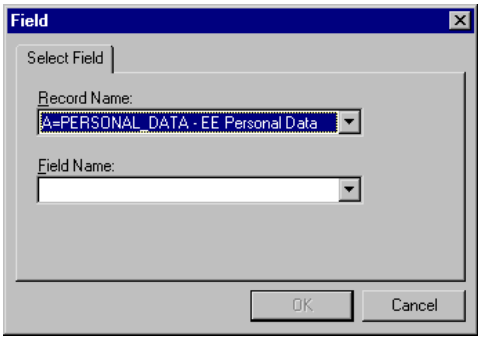
   The exact dialog box you see depends on the value you selected from the list.



Between Const-Const dialog box

Field

To compare the value in the selected record field to the value in another field, you need to pick the second field. When you’ve selected **Field** as the value type, the Select Field dialog box appears.



Record Name: A=PERSONAL_DATA - EE Personal Data

Field Name:

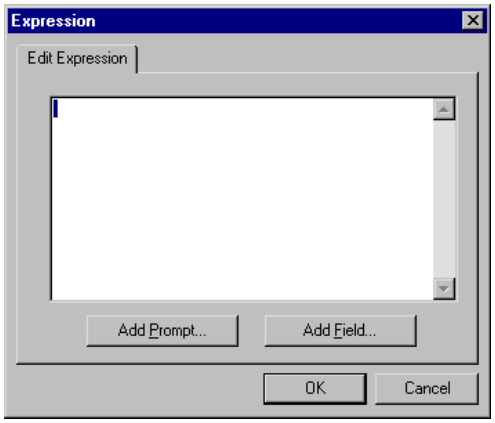
Select Field dialog box

The **Record Name** box lists all the record components that are part of the current query. Select the record component and the field, then click the **OK** button. The selected field name appears in the second **Expression** column of that field’s row.

PeopleSoft Query checks whether the field you’ve selected has the same data type as the field you’re comparing it to. If it doesn’t, Query displays a warning message so that you know the values might never match.

Expression

If you select Expression as the value type, you can enter any valid SQL expression as the comparison value in the Edit Expression dialog box that appears.



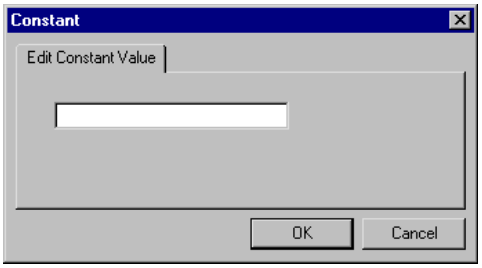
Edit Expression dialog box

You can type any valid SQL expression in the text box. If you want to add a field or user prompt to the expression, click the **Add Prompt** or **Add Field** button.

The **Add Prompt** and **Add Field** buttons display the same dialog boxes as you see when you're adding a field or prompt as a comparison value: the Add Prompt button displays the Run-time Prompt dialog box, and the Add Field button displays the Select Field dialog box. The only difference is that Query adds the field or prompt to your expression rather than using it directly as the comparison value.

Constant

Comparing the value from the first (left-hand) expression to a constant value is simple. When you select Constant as the value type, the Edit Constant Value dialog box appears.



Edit Constant Value dialog box

Enter the value you want to compare the first expression to in the text box. Notice that Query sizes the text box based on the length of the first expression you're comparing the value to, and that all the edit characteristics match the first expression. For example, if you're comparing a constant to an uppercase character field, the text box displays all characters as uppercase. Also, if the first expression has an associated prompt table, the standard prompt options are active—F4, SHIFT+F4, and CTRL+F4. (F4 displays a list of all valid codes, SHIFT+F4 displays a partial list of valid codes, and CTRL+F4 prompts for alternate search criteria.)

When you've entered the value, click **OK** to close the dialog box. The comparison value appears in the expression column of the field's row.

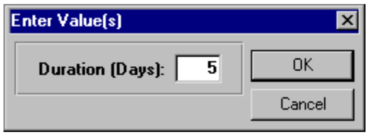
Subquery

When you select *Subquery* as the value type, PeopleSoft Query displays the database component view, just like when you're starting a new query. In fact, that's what you're doing—starting a new query.

For more information about creating subqueries, see Advanced Query Options.

Prompt

Adding a prompt allows you to further refine a query when you run it. For example, suppose you wanted to change the query we created in Chapter 3, SampleQuery1, so that you could prompt the user running the query to enter a value for the duration of the vacation. The query we created *always* retrieves rows for employees who had taken vacation in excess of 15 days, because we defined the query with a constant value of 15 on which to make a comparison. But if you could add a prompt to the query that allowed the user to enter *any* duration, then the query would return employees based on the value provided by the user when running the query. This obviously makes a query much more flexible. When you run the query, a dialog box would appear so that you could specify a duration.

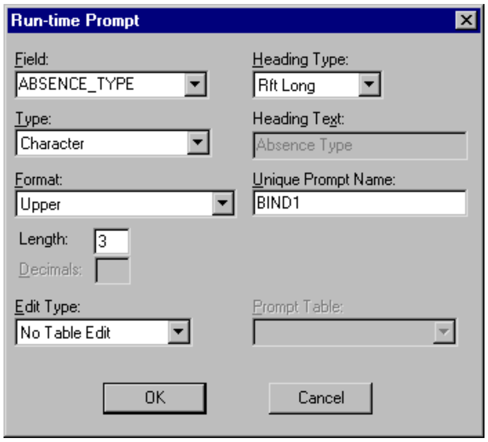


Duration (Days): 5

Enter Value(s) prompt box

The user can enter a duration into the text box. The query uses the value the user enters as the comparison value for the criterion that included the prompt.

When designing your query, if you select *Prompt* as the value type, the Run-time Prompt dialog box appears.



Field: ABSENCE_TYPE
Heading Type: Rft Long
Type: Character
Heading Text: Absence Type
Format: Upper
Unique Prompt Name: BIND1
Length: 3
Edit Type: No Table Edit

Run-time Prompt dialog box

You use the Run-time Prompt dialog box to define the prompt dialog box that Query displays to ask the user for a comparison value. In most cases, it already displays the correct information, and you won’t need to change it.

The Field list box shows the name of the field you’re prompting for. Query looks to the record definition for information about this field and fills out the rest of the dialog box based on its properties.

The Edit Type list box defines the type of field edit for the specified field. Query defaults to the type of edit specified in the record definition for the field. In general, we recommend that you don’t change its value here. If you’re unhappy with the edit type, use Application Designer to change its record definition so that the change is made consistently throughout PeopleTools.

If the Edit Type is ***Prompt Table***, the Prompt Table list box specifies the prompt table to use. If the Edit Type is *Translate Table*, the values used are determined by the field in the Field list box. Query assumes that the specified field has Translate Table values associated with it, and that the field is identified as a Translate Table field in its record definition.

Note. When using a prompt table on a field from a record definition with multiple keys, you must prompt for all higher-level keys before lower-level keys. Query needs values for the higher-level keys to generate the correct prompt list. Because of this complication, we recommend that you don’t use multi-key prompt tables.

The Heading Text box displays the label for the text box where the user will enter the comparison value. By default, Query uses the long name of the field from the record definition (RFT Long). You can, however, change the label to any text you’d like. To change the text, select ***Text*** from the Heading Type list box, then type the new label in the Heading Text box. The Heading Type list box also gives you the option of using the short name from the record definition (RFT Short).

The Unique Prompt Name is a default value generated by Query for globalization. This value can only be set by a base language user in order to uniquely identify a query prompt parameter.

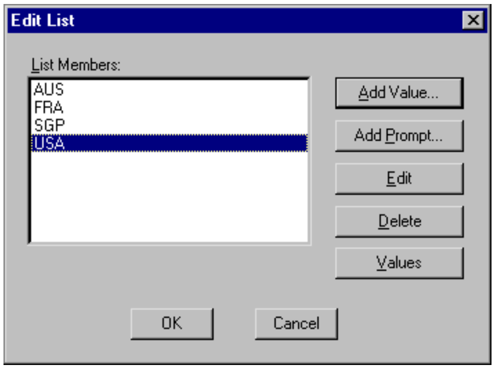
The Type, Length, Decimals, and Format boxes all display the properties of the field you're prompting for. Query will use these same properties to determine the size of the input text box and the type of input it will accept.

If the field you're prompting for has an associated prompt table (even if it's the Translate table), the Edit Table list box shows its name. When there's an associated prompt table, the user can press **F4** in the prompt dialog box to select a valid value from the table.

List

If you selected the ***in list*** or ***not in list*** operator, the only available value type is ***In List*** or ***Subquery.*** When you select In List value type, the Edit List dialog box appears.

You use this dialog box to build a list of values for Query to compare to the value from the first expression. To add a comparison value to the list, click the Add Value button and enter the value in the Constant dialog box. Repeat this step for each value you want in the list.



AUS
FRA
SGP
USA

Adding Values to the Edit List dialog box

In the designer view, the items defined with the Add Value button appear in the (left-hand) Expression 2 column.

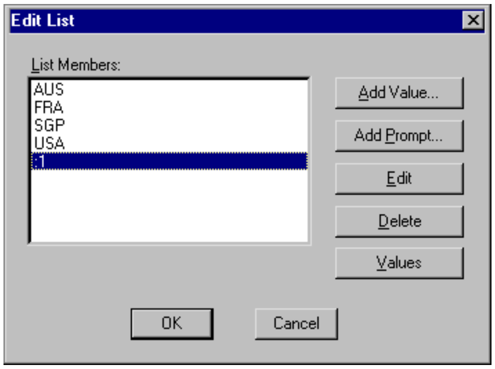
You can also add one or more prompts to the list, so that users can enter the comparison values when they run the query.

To add prompts to the Expression List:

7. Click the Add Prompt button to display the Run-time Prompt dialog box.

8. Complete the dialog box as described in the section on the Prompt value type.

When you click **OK** on the **Run-time Prompt** dialog box, the prompt appears in the List Members list box.



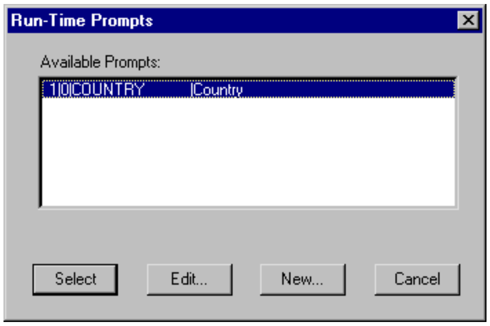
List Members:
AUS
FRA
SGP
USA
:1

Edit List dialog box

If you selected the *in list* operator, you may want users to be able to enter more than one value to search for. So you'll want to add more than one prompt.

9. To add another prompt, click the **Add Prompt** button again.

   Since you already have a prompt in place, a different dialog box appears, showing the prompt you've already added.



Available Prompts:
1|0|COUNTRY |Country

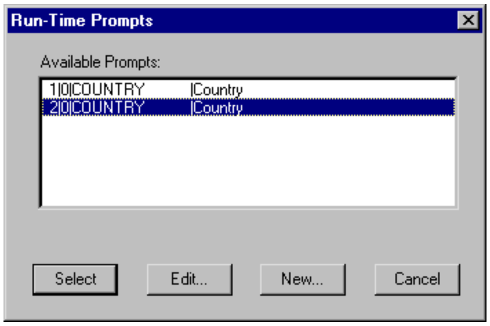
Run-Time Prompts dialog box

10. To add the next prompt, click the **New** button and complete the **Run-time Prompt** dialog box.

    The settings for this second prompt may be identical to those for the first prompt—after all, they ask for values for the same record field—unless you want to provide a different label for this prompt in the **Heading Text** box.

    When you click **OK** on the **Run-Time Prompts** dialog box, the second prompt appears in the **Available Prompts** list.

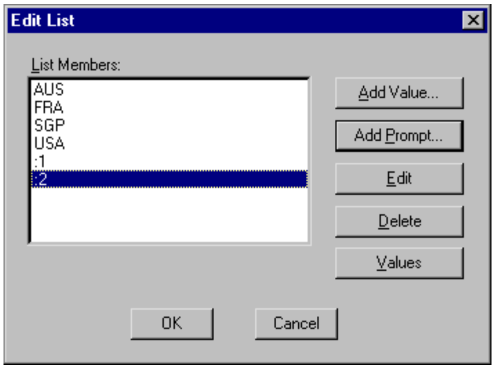
11. Highlight the second prompt and click the **Select** button to add it to the list of comparison values.


Run-Time Prompts

Available Prompts:

1|0|COUNTRY |Country
2|0|COUNTRY |Country

The prompt then appears in the **Edit List** dialog box.


Edit List

List Members:

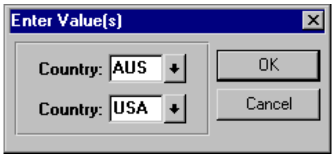
AUS
FRA
SGP
USA
:1
:2

Edit List dialog box

12. Repeat this process for each prompt you want to add.

13. When you're done, click **OK** to close the **Edit List** dialog box.

When a user runs a query with multiple prompts, they'll see a dialog box prompt dialog box similar to the following. Here, the user has been prompted for and has entered two countries to include in the query.


Enter Value(s)

Country: AUS

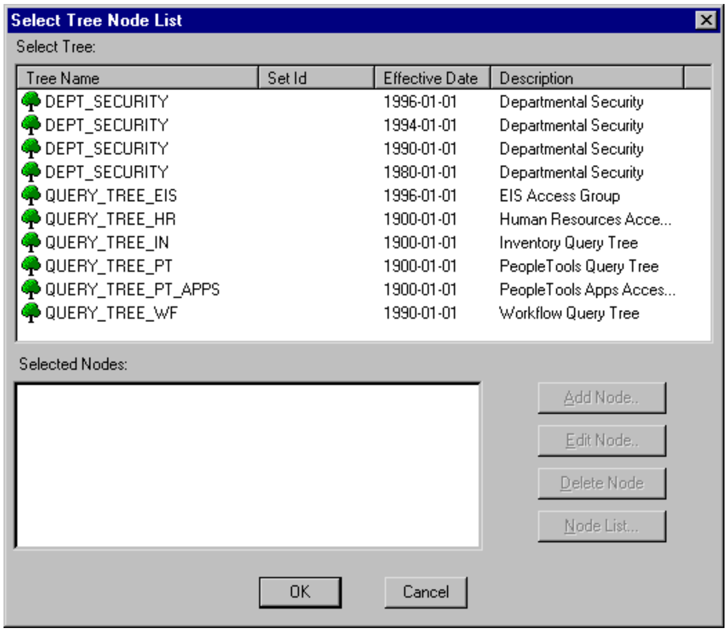
Country: USA

Enter Value(s) dialog box

Tree Option

If you selected the in tree or not in tree operator, the only available value type is Tree Option. The *in tree* and *not in tree* operators let you check whether the value in the first (left-hand) Expression column appears in a given tree, or in specific branches of the tree. For example, you could restrict your query to records where the DEPTID field identifies a retail banking administration department, that is, one of the nodes that appears under retail banking administration node in the DEPT_SECURITY tree.

When you select *Tree Option* as the value type in the second (right-hand) Expression column, the Select Tree Node List dialog box appears.

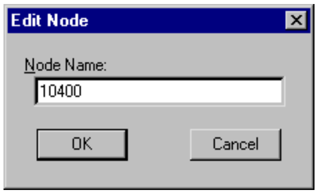


Tree Name	Set Id	Effective Date	Description
DEPT_SECURITY		1996-01-01	Departmental Security
DEPT_SECURITY		1994-01-01	Departmental Security
DEPT_SECURITY		1990-01-01	Departmental Security
DEPT_SECURITY		1980-01-01	Departmental Security
QUERY_TREE_EIS		1996-01-01	EIS Access Group
QUERY_TREE_HR		1900-01-01	Human Resources Acce...
QUERY_TREE_IN		1900-01-01	Inventory Query Tree
QUERY_TREE_PT		1900-01-01	PeopleTools Query Tree
QUERY_TREE_PT_APPS		1900-01-01	PeopleTools Apps Acces...
QUERY_TREE_WF		1990-01-01	Workflow Query Tree

Select Tree Node List dialog box

You use this dialog box to select which branches of the tree PeopleSoft Query will check. There are two ways to add a tree node to the Selected Nodes list.

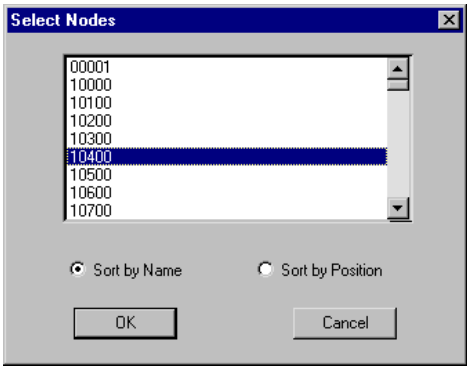
You can highlight the desired tree, and click the Add Node button and enter the node name in the dialog box that appears.



Node Name: 10400

Edit Node dialog box

You can also highlight the desired tree, and then click the Node List button. The Select Nodes dialog box appears, enabling you to select the desired tree node(s) from a list of those in the highlighted tree. To help you find the node you want, the Select Nodes dialog box allows you to display the node names in alphabetical order (Sort by Name) or organized as they are in the tree hierarchy (Sort by Position). Add as many of the tree nodes as you like. When you add a high-level node to the list, you don’t need to add any of the lower-level nodes that appear under that high-level node. When you’re done, click OK to close the Select Nodes dialog box.



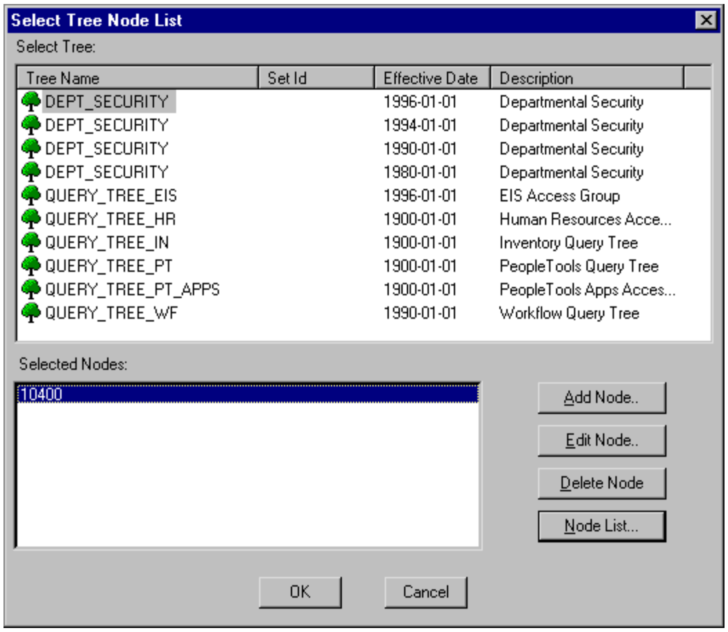
Select Nodes

00001
10000
10100
10200
10300
10400
10500
10600
10700

● Sort by Name ○ Sort by Position

Select Nodes dialog Box

The node(s) you entered or selected from the list appear in the **Selected Nodes** list box:



Select Tree Node List

Select Tree:

Tree Name	Set Id	Effective Date	Description
DEPT_SECURITY		1996-01-01	Departmental Security
DEPT_SECURITY		1994-01-01	Departmental Security
DEPT_SECURITY		1990-01-01	Departmental Security
DEPT_SECURITY		1980-01-01	Departmental Security
QUERY_TREE_EIS		1996-01-01	EIS Access Group
QUERY_TREE_HR		1900-01-01	Human Resources Acce...
QUERY_TREE_IN		1900-01-01	Inventory Query Tree
QUERY_TREE_PT		1900-01-01	PeopleTools Query Tree
QUERY_TREE_PT_APPS		1900-01-01	PeopleTools Apps Acces...
QUERY_TREE_WF		1990-01-01	Workflow Query Tree

Selected Nodes:

10400

Selected Nodes Appear in the “Selected Nodes” List Box

Specifying Effective Date Criteria

Effective-dated tables have record definitions that include the field EFFDT. This Effective Date field is used throughout the PeopleSoft applications to give you a historical perspective, allowing you to see how the data has changed over time. Whenever someone adds a row of data to the table, they specify the date on which that data becomes effective; whenever someone changes a row of data, they specify a new effective date, and the system retains the previous version of the row as history.

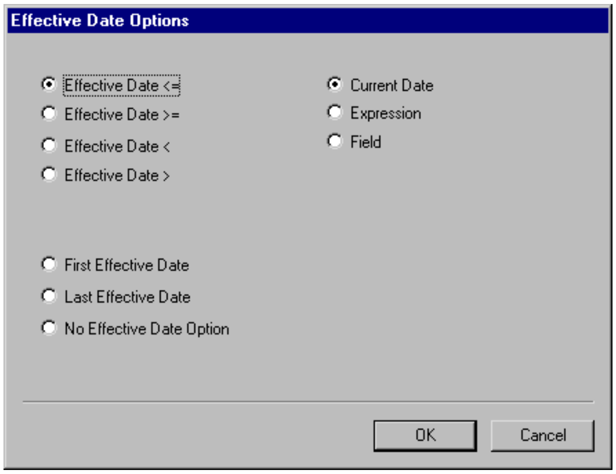
For more information about effective dating and how PeopleSoft applications use it, see the documentation for your PeopleSoft applications.

When you’re using a PeopleSoft application for day-to-day processing, you usually want the system to give you the *currently effective* rows of data—that is, the rows whose EFFDT value is closest to today’s date without being in the future. You don’t want to see the history rows, which are no longer accurate, nor do you want to see future-dated rows, which aren’t yet in effect.

When you’re querying an effective-dated table, though, you may well want to see some rows that aren’t currently in effect. You might want to see all the rows, regardless of their effective date. Or you might want to see the rows that were effective as of some date in the past.

To specify effective date criteria:

14. When you choose the record component for an effective-dated table, the Effective Date Options dialog box appears.



● Effective Date <=
○ Effective Date >=
○ Effective Date <
○ Effective Date >

● Current Date
○ Expression
○ Field

○ First Effective Date
○ Last Effective Date
○ No Effective Date Option

Effective Date Options dialog box

You use this dialog box to specify which row of data Query retrieves for each item in the table. The default is the currently effective row—the row where the effective date is less than or equal to today’s date.

15. Choose from the following options.

If you choose one of the comparison options (the four Effective Date radio buttons on the upper left), you can also choose to compare each row’s effective date against today’s date, or a date other than today, using one of the buttons on the right:

- If you select Current Date, the comparison is made against today’s date.
- If you select Constant**,** Query displays the Edit Expression dialog box so that you can enter a date or a SQL expression that evaluates to a date.

For more information about this dialog box, see Entering Comparison Values.

- Use the Constant option when you want to see the rows that were effective as of a past date or will be effective on some future date. Also use this option if you want to prompt users for an effective date when they run the query: you can add a prompt to the expression you define in the Edit Expression dialog box.
- If you select Field**,** Query displays the Select Field dialog box so that you can select the record field that holds the date you want to compare effective dates to.

For more information about the Select Field dialog box, see Entering Comparison Values.

- Use the **Field** option when you want to see the rows that were effective at the same time as some other record. For example, if you're reviewing the list of products on a customer order, you'll want to see the products that were effective on the date of the order.

  The three options in the lower left part of the dialog box don't compare a row's effective date to any other date.

- First Effective Date returns the row with the oldest effective date, usually the first row entered for an item.
- Last Effective Date returns the row with the latest effective date, even if that date is still in the future.
- No Effective Date Option returns all rows, regardless of their effective dates.

Note. All options except No Effective Date Option return a single row for each item on the table. If you want a subset of the rows (say, all future-dated rows or all history rows), select No Effective Date Option in this dialog box, then enter a selection criterion on the EFFDT field. Use the standard comparison operators rather than the Eff Date comparison operators.

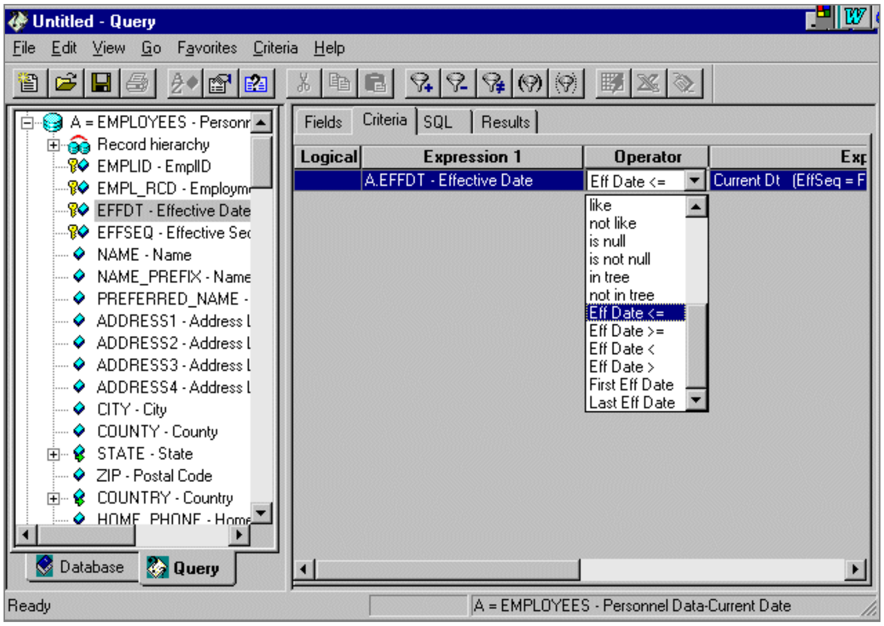
For more information on selection criteria, see Entering Selection Criteria.

16. When you've made your selection, click **OK** to open the record component.

17. If you selected anything other than **No Effective Date Option**, Query automatically adds a selection criterion for the query.

To see the criterion, select the **Criteria** tab to display the criterion.

Notice that when the selected record field is EFFDT, the list of operators includes several Eff Date operators, corresponding to the options in the Effective Date Options dialog box.



Logical	Expression 1	Operator	Exp
	A.EFFDT - Effective Date	Eff Date <=	Current Dt (EffSeq = F

List of Comparison Operators for the EFFDT Field

Keep in mind that the effective date operators work a little bit differently than the standard comparison operators, in that they always return a single effective-dated row. For example, **Eff Date <=** returns the one row whose EFFDT value is most recent, whereas **not greater than** would return the currently active row *and* all history rows.

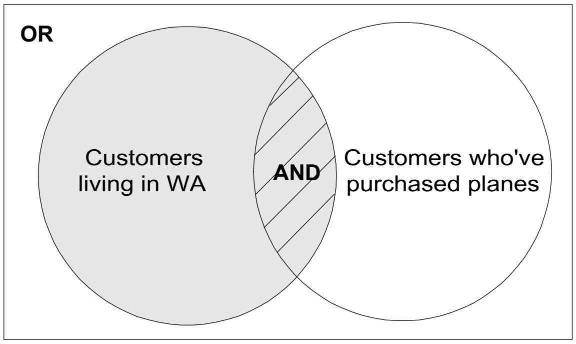
Relating Multiple Criteria

One feature that adds power and flexibility to Query is the ability to relate multiple criteria in very specific ways that you define. You can define these relationships using the “AND” and “OR” operators, as well as by grouping criteria with parentheses.

The AND and OR Logical Operators

When you specify two or more selection criteria for a query, you need to tell PeopleSoft Query how to coordinate the different criteria. For example, suppose you’re querying the list of your customers, and you’ve defined two criteria: one that selects customers from the state of Washington and another that selects customers who have purchased airplanes. You may want Query to return only those rows that meet *both* conditions (customers in Washington who’ve purchased airplanes), or you may want the rows that meet *either one* of the conditions (*all* Washington customers plus *all* customers who’ve purchased airplanes).

When your query includes multiple criteria, you link them together using either AND or OR. When you link two criteria with AND, it means that a row must meet the first criterion ***and*** the second criterion in order for Query to return it. When you link two criteria with OR, it means that a row must meet the first criterion ***or*** the second criterion, not necessarily both.



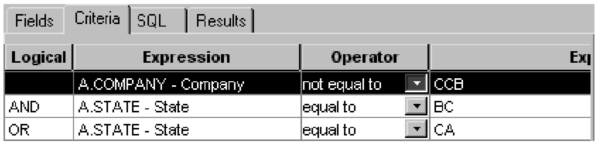
Rows returned by AND and OR

By default, Query assumes that you want those rows that meet all the criteria you specify. When you add a new criterion, Query displays AND in the Logical column on the Criteria tab. To link the criterion using OR instead, simply double-click in the Logical column for the field's row. Double clicking in this column enables you to toggle back-and-forth between the *AND* and *OR* operators as necessary.

Grouping Criteria with Parentheses

When your query includes multiple criteria, Query checks them according to the rules of logic, which tell it to evaluate criteria linked by ANDs before those linked by ORs. When all the criteria are linked by ANDs, this order will always give you the correct result. When you mix in one or more ORs, however, it may not give you the result you want.

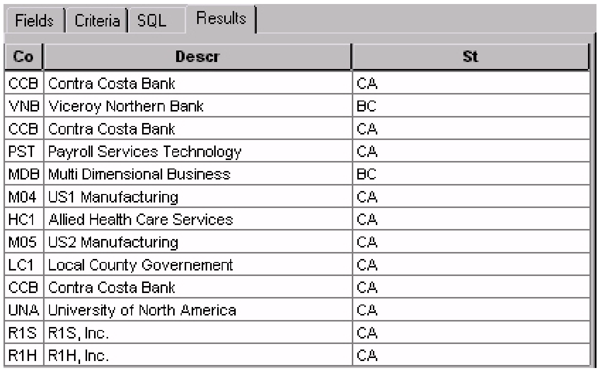
For example, suppose you wanted a list companies other than Contra Costa Bank (CCB) from either British Columbia (BC) or California (CA). You enter these criteria:



Logical	Expression	Operator	Ex
	A.COMPANY - Company	not equal to	CCB
AND	A.STATE - State	equal to	BC
OR	A.STATE - State	equal to	CA

These criteria gives the wrong result

This set of criteria gives you a list of companies in British Columbia (except for Contra Costa Bank) and *all* companies in California (including Contra Costa Bank).



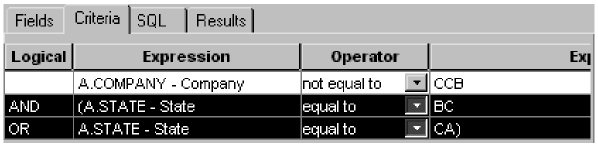
Co	Descr	St
CCB	Contra Costa Bank	CA
VNB	Viceroy Northern Bank	BC
CCB	Contra Costa Bank	CA
PST	Payroll Services Technology	CA
MDB	Multi Dimensional Business	BC
M04	US1 Manufacturing	CA
HC1	Allied Health Care Services	CA
M05	US2 Manufacturing	CA
LC1	Local County Governement	CA
CCB	Contra Costa Bank	CA
UNA	University of North America	CA
R1S	R1S, Inc.	CA
R1H	R1H, Inc.	CA

Incorrect results obtained because criteria were not grouped properly

Why didn't it give the result you wanted? Because Query evaluates the criteria in the order they appear. It looks for rows where the company name isn't Contra Costa Bank *and* the state is British Columbia *or* rows where the state is California.

What you want in this case is for Query to look for rows where the state is British Columbia ***or*** California, *and* the company name isn't Contra Costa Bank. That is, you want it to evaluate the OR before the AND. You accomplish this feat by adding parentheses to the list of criteria. When a list of criteria includes parentheses, Query evaluates the criteria inside the parentheses before the criteria outside the parentheses.

On the Criteria tab, the opening parenthesis appears just before the field name and the close parenthesis appears just after the comparison value. For example, here's the set of criteria that will give us the result we want:



Logical	Expression	Operator	Ex
	A.COMPANY - Company	not equal to	CCB
AND	(A.STATE - State	equal to	BC
OR	A.STATE - State	equal to	CA)

Adding parentheses around the last two criteria

To group criteria:

18. Highlight the rows you want to group by holding down the **SHIFT** key while pressing the up or down arrow keys.

19. Either click the Group Criteria button (?) on the toolbar or select **Criteria, Group.**

A parenthesis appears at the beginning of the *first* Expression column for the first row you selected and at the end of *second* Expression column for the last row you selected. Looking at the highlighted rows in the example above, you can see that the *AND* operator *precedes* the parentheses, while the *OR* operator is located *within* the parentheses.

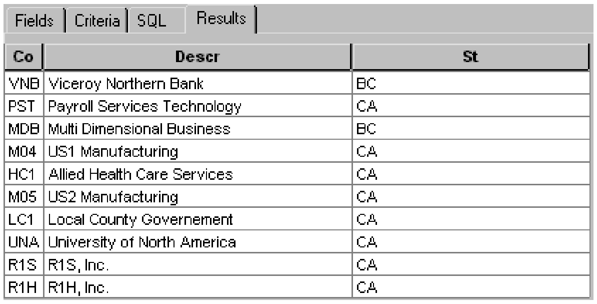
If you add parentheses to a row that already has a parenthesis in it, Query increments the number of parentheses in the row. You can add as many parentheses as you need to.

To ungroup criteria:

20. Either click the Ungroup Criteria button on the toolbar or select **Criteria, Ungroup**.

The parentheses will be removed, indicating that the criteria have been ungrouped.

The result set below matches our intended results. Notice that Contra Costa Bank (CCB) is no longer included in the results.



Co	Descr	St
VNB	Viceroy Northern Bank	BC
PST	Payroll Services Technology	CA
MDB	Multi Dimensional Business	BC
M04	US1 Manufacturing	CA
HC1	Allied Health Care Services	CA
M05	US2 Manufacturing	CA
LC1	Local County Governement	CA
UNA	University of North America	CA
R1S	R1S, Inc.	CA
R1H	R1H, Inc.	CA

Correct results obtained by using parentheses to group criteria

CHAPTER 5

Advanced Query Options

In this section, we’ll discuss features of PeopleSoft Query you can use to create more sophisticated queries. Keep in mind that a familiarity with SQL is helpful when writing more advanced queries. You should be aware that it’s possible to create queries that return an infinite number of rows or that don’t produce any rows at all. Query makes creating queries easy, but doesn’t prevent you from creating “runaway” queries.

Creating Workflow Queries

Workflow queries are queries that PeopleSoft Workflow uses as part of automated business processes. There are two types of workflow queries:

- *Role queries*, which PeopleSoft Workflow runs to determine which users should receive work items.
- *Database agent queries*, which PeopleSoft Workflow runs to check the database for conditions that need to trigger a business event.

Note. All headings in database agent queries must be text, rather than derived from the record field text (RFT) values.

Because these types of queries are run by the system rather than by a user, the PeopleSoft application doesn’t run the same security checks before running a workflow query as is does for other queries. In particular, it doesn’t check which access groups the user running the query has access to—because the user is a workflow agent. (For this reason, you may not want to allow all users to create workflow queries.

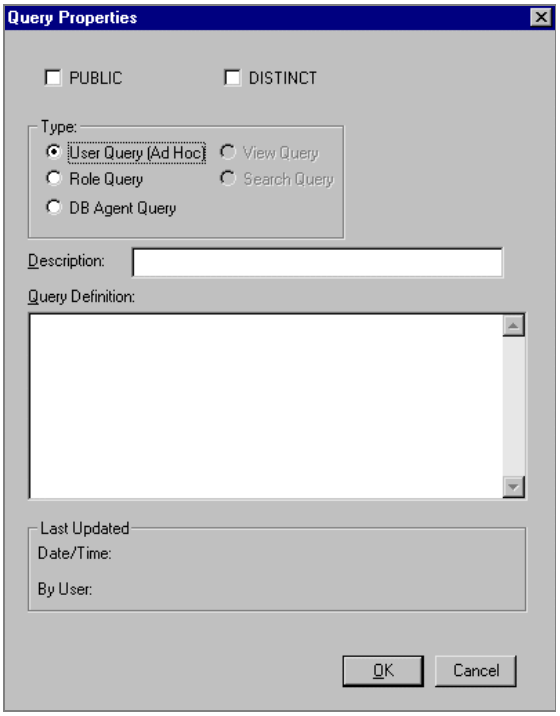
For more information on controlling user access to this feature using query profiles, see Setting up Query Security.

Note. If you’re going to use a query as a role query or database agent query, you *must* change it to the appropriate type. If you don’t, the query is likely to fail at run time.

To create a workflow query:

1. Select either **File, Properties** or click the button on the toolbar to identify the currently open query as a role query or database agent query.

   The **Query Properties** dialog box appears.



Query Properties

☐ PUBLIC ☐ DISTINCT

Type:
● User Query (Ad Hoc) ○ View Query
○ Role Query ○ Search Query
○ DB Agent Query

Description:

Query Definition:

Last Updated
Date/Time:
By User:

Query Properties dialog box

2. Select either the **Role Query** or **DB Agent Query** option and click **OK**.

The User Query (Ad Hoc) option refers to any query that’s not a workflow query.

It is recommended that you use a naming convention to distinguish workflow queries from other queries. The Role Query option refers to the fact that PeopleSoft names its role queries with names that begin with [ROLE]. The DB Agent Query option refers to the fact that PeopleSoft names its database agent queries with names that begin with [DBAG].

Aggregate Functions

In a standard query, each row in the result set that PeopleSoft Query returns corresponds to an individual row in the table you’re querying. Sometimes, though, what you want is a summary of the information in multiple rows. For example, you might want to know how many customers you have in each state. You can query for this kind of summary information using *aggregate functions*.

Understanding Aggregate Functions

An aggregate function is a special type of operator that returns a single value based on multiple rows of data. When your query includes one or more aggregate functions, Query collects together related rows and displays a single row that summarizes their contents.

For example, suppose you have an Order table that includes (among other fields) a Customer ID and an Amount for each item ordered. You'd like to find out how much each customer has ordered, so you create a query that selects the Customer ID and Amount fields. Without any aggregate functions, this query would return the same number of rows as there were in the table: If Stuart Schumacher ordered 10 items, you'd see 10 rows with his ID in the Customer ID column. On the other hand, if you apply the aggregate function Sum to the Amount field, you'll get just one row for each Customer ID. Query takes all the rows with the same value in the non-aggregated column (Customer ID) and collapses them into a single row. The value of the Amount field in Stuart Schumacher's row would be the sum of the values from the 10 rows.

The table below lists the aggregate functions you can apply to a field using Query.

Aggregate Function	***Action***
Sum	Adds the values from each row and displays the total.
Count	Counts the number of rows.
Min	Checks the value from each row and returns the lowest one.
Max	Checks the value from each row and returns the highest one.
Average	Adds the values from each row and divides the result by the number of rows.

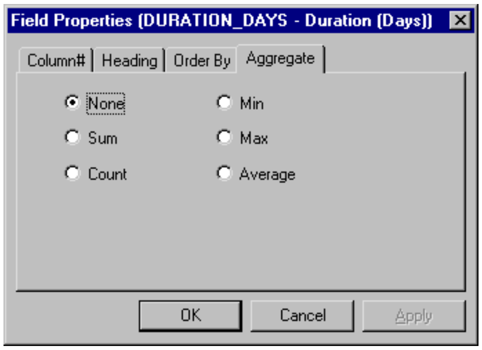
Note. In addition to the aggregate functions above, any aggregate function supported by the underlying database can be used by creating an expression component.

To apply an aggregate function to a field:

3. Click the Fields tab in the designer view to make sure that it is the active tab.

4. Either double-click in the field's Agg column, or right-click anywhere in the appropriate row and select **Aggregate** from the pop-up menu.

   The **Field Properties** dialog box is displayed with the **Aggregate** tab active.


Field Properties (DURATION_DAYS - Duration (Days))
Column# | Heading | Order By | Aggregate
● None
○ Min
○ Sum
○ Max
○ Count
○ Average
OK
Cancel
Apply

Field Properties dialog box- Aggregate tab

5. Pick the aggregate function you want to use for this field and click **OK.**

The abbreviation for the selected function appears in the Agg column. If a function is not available for the field you've selected, it is grayed out. For example, you can't use Sum with a character field.

Aggregate Functions and Selection Criteria

When you apply an aggregate function to a field, you're redefining how PeopleSoft Query uses the field throughout the query. Essentially, Query replaces the field, wherever it occurs, with the results of the function. If you select the field as a display column, Query displays the aggregate values; if you use the field as an "order by" column, Query orders the results based on the aggregate values.

Note. If you don't want Query to redefine the field in this way—for example, if you want to display both the individual row values *and* the results of the aggregate function—create an expression that includes the aggregate function rather than applying the function directly to the field.

For more information**,** see Expressions.

HAVING Criteria

SQL doesn't support the use of aggregate functions in WHERE clauses. So, once you've applied an aggregate function to a field, you can't use that field in your selection criteria, which correspond to a SQL WHERE clause. So what do you do when you want to select rows based on the results of an aggregate function? Suppose, for example, you want a list of the departments whose minimum salary is greater than $100,000. For these cases, Query enables you to create *HAVING criteria*.

In SQL, a HAVING clause is like a WHERE clause for table rows that have been grouped together into a single row of output. The system evaluates WHERE clauses by looking at the individual table rows *before* they're grouped together by the aggregate function, then it evaluates HAVING clauses *after* applying the function. So, if you want to check the value returned by the function, you need to define a HAVING criterion.

If your query includes one or more aggregate functions, follow the procedure below to define your selection criteria.

To define selection criteria in a query with an aggregate function:

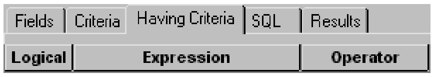
6. In the designer view, select the Criteria tab to display the selection criteria.

7. Define the selection criteria for the fields that *don't* have aggregate functions applied to them.

For more information**,** see Specifying Query Selection Criteria.

8. Select Criteria, Having to display the Having Criteria tab.

The Having Criteria tab appears to the right of the Criteria tab.



You add selection criteria using the **Having Criteria** tab in the same way you add selection criteria with the **Criteria** tab. However when you add Having Criteria from an aggregate field, new criteria will be added to the Having Criteria tab instead of the Criteria tab.

9. Define the selection criteria for the fields that have aggregate functions applied to them.

Keep in mind that Query will be comparing *the result of applying the aggregate function* to the comparison value.

Expressions

When looking at any query component view, you may have noticed the next-to-last item, *Expressions*. (*Prompts* is always the very last item.) *Expressions* are calculations that PeopleSoft Query performs as part of a query. You use them when you need to calculate a value that Query doesn't provide by default. For example, you might want to add the values from two fields together or multiply a field value by a constant.

In PeopleSoft Query, you can use expressions in two ways:

- As comparison values in selection criteria.
- As columns in the query output.

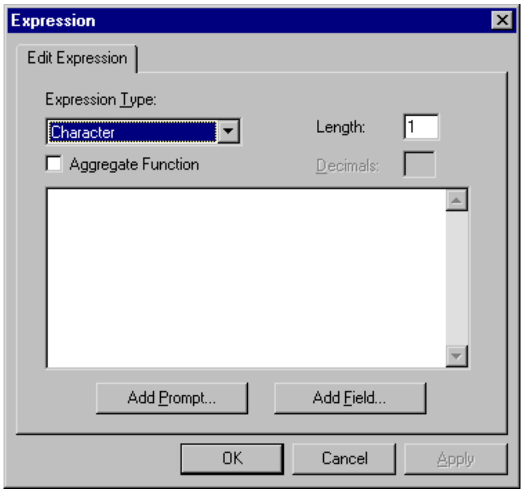
For more information on the use of expressions as comparison values, see Specifying Query Selection Criteria. This section describes the use of expressions in the query output.

To add an expression to a query:

10. Select ***Expressions*** in the query component view and then select **Edit, New Expression.**

Note. You can also right-click on ***Expressions*** and then select **New Expression** from the pop-up menu.

The **Expression** dialog box is displayed.



Expression dialog box

11. In the **Expression Type** list box, identify the data type of the value this expression will return.

Query needs this information to correctly use and format the result.

12. If you select ***Character***, enter the maximum length of the string in the Length text box that appears to the right of the Expression Type box.

13. If you select a ***Number*** or ***Signed Number*** data type, the Length box is replaced by an Integers box. Enter the maximum size of the number in the Integers and Decimal text boxes that appear.

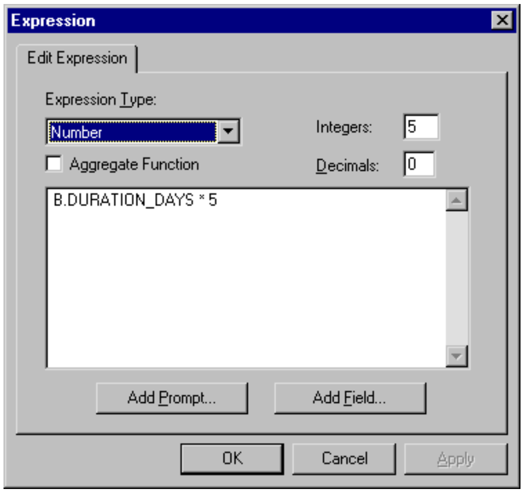
14. Enter the *total* number of digits in the Integer box and the number of digits to the right of the decimal point in the Decimal box.

For example, a number expressed as 1000.00 has a total of *six* digits, and there are *two* digits to the right of the decimal point.

15. Select the Aggregate Function check box if your expression includes an aggregate function, such as SUM, AVG, or COUNT.

The large text box in the middle of the dialog box is where you enter the expression. You can type directly in the box.

16. To include a record field in the expression, you can either type the field name or click the **Add Field** button and select the field from the **Field** dialog box.



B.DURATION_DAYS * 5

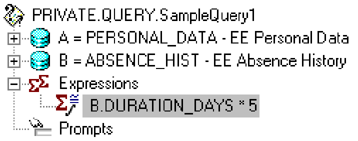
Expression dialog box

Note. To add a prompt to an expression, click the **Add Prompt** button and complete the Run-time Prompt dialog box.

For more information about prompts and the Run-time Prompt dialog box, see Specifying Query Selection Criteria.

17. When you're done entering the expression, click **OK** to close the **Expression** dialog box.

    The expression appears in the query component view under "Expressions."


PRIVATE.QUERY.SampleQuery1
A = PERSONAL_DATA - EE Personal Data
B = ABSENCE_HIST - EE Absence History
Expressions
B.DURATION_DAYS * 5
Prompts

Viewing Expressions in the Component View

Now you can treat this expression just as if it were a field in the query. You can select it for output, change its column heading, or choose it as an "order by" column.

For more information about the available options, see Creating and Running Simple Queries.
.

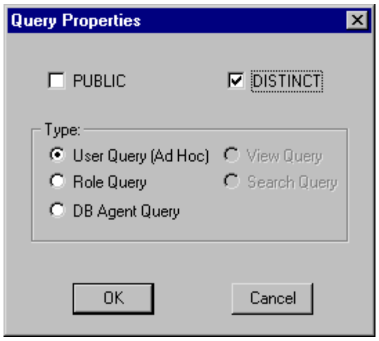
Using the Distinct Option

You can use the Distinct option to eliminate duplicate rows in a query result. Some queries will return the same row more than once because it satisfies the query in more than one case. The Distinct option removes the duplicate rows from the result set. If two returned rows are identical in all selected columns, Query displays only one of the rows.

To make a query distinct:

18. Select **File, Properties** or click the button.

    The **Query Properties** dialog box appears.


Query Properties

□ PUBLIC ☑ DISTINCT

Type:
● User Query (Ad Hoc) ○ View Query
○ Role Query ○ Search Query
○ DB Agent Query

Query Properties dialog box

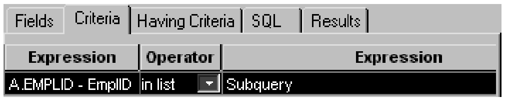
19. Select the **DISTINCT** option and then click **OK.**

For example, if you query the JOB record component to return all JOBCODES, you'll get a bunch of rows that look identical because some employees have the same JOBCODE. If you select **DISTINCT**, you'll only receive each JOBCODE once.

Subqueries

A *subquery*, sometimes called a *sub-SELECT*, is a query whose results are used by another query. The main query uses the subquery's result set as a comparison value for a selection criterion.

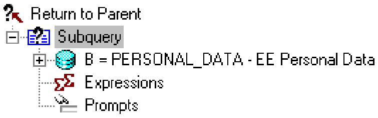
You create a subquery when you need to compare a field value to the results of a second query. Suppose, for example, that you want a list of employees who aren't members of any professional organizations. For each employee in the PERSONAL_DATA table, you need to find out whether their EMPLID is in the MEMBERSHIP table. In other words, you need to compare the value in the PERSONAL_DATA.EMPLID field to the results of a subquery that selects the EMPLID values from the MEMBERSHIP table.



Expression	Operator	Expression
A.EMPLID - EmplID	in list	Subquery

Using a subquery

You create subqueries just like other queries. When you select **Subquery** as the comparison value, PeopleSoft Query displays a special database component view, so that you can select a record component. From there, you follow the same steps as for any query.



Component view for a subquery

At the top of the subquery's component view is an item called, *Return to Parent*. Click this item to return to the main query. Then, if you want to return to the subquery's component view, click on the *Subquery* item in the **Expression** column.

Note. A subquery can return data from only one record component.

Joins

PeopleSoft Query enables you to create queries that include multiple-table *joins*. Joins retrieve data from more than one underlying table, presenting the data as if it came from one. Query links the tables based on common columns and links the rows on the two tables by common values in the shared columns.

Joins are what make relational databases relational. Using joins, you define relationships among fields when you query the records, not when you create the records. Because PeopleSoft records are highly normalized (they each describe one kind of entity), it's easy to use Query to create joins.

We saw an example of a join in Using PeopleSoft Query when we created a query to find all employees who took vacations of more than three weeks. The data we needed to find these employees wasn't all on one table. Employee data is in the PERSONAL_DATA record, and the days off are in the ABSENCE_HIST table. So we had to join these two tables.

The procedure for joining tables differs depending on how the tables being joined are related to each other. PeopleSoft Query recognizes three types of joins:

- **Hierarchical join**, which joins a parent table to a child table. (A *child table* is a table that uses all the same key fields as its parent, plus one or more additional keys.) You specify the parent/child relationship using the Parent Record Name option in the Application Designer.

For more information, see Setting up Query Security.

- **Related record join**, which joins a prompt table to a table that uses the codes from that table.
- **Any record join**, which joins any two tables in the database.

Note. If you join a record component that you later decide to remove from your query, you can highlight the record in the **Record** list box and select **Join, Remove Record.**

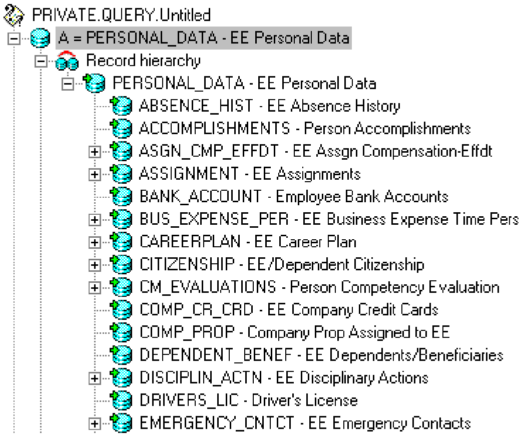
Creating Hierarchical Joins

You use the query component tree to create a join query.

To create a join query:

20. After selecting one of the record components from the database component view, highlight it in the query component view, and expand the record hierarchy by clicking the "+" to the left of the *Record hierarchy* symbol .

You'll see all the records that have common information.



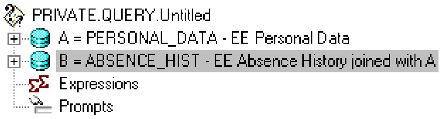
```
PRIVATE.QUERY.Untitled
⊟ A = PERSONAL_DATA - EE Personal Data
  ⊟ Record hierarchy
    ⊟ PERSONAL_DATA - EE Personal Data
        ABSENCE_HIST - EE Absence History
        ACCOMPLISHMENTS - Person Accomplishments
      ⊞ ASGN_CMP_EFFDT - EE Assgn Compensation-Effdt
      ⊞ ASSIGNMENT - EE Assignments
        BANK_ACCOUNT - Employee Bank Accounts
      ⊞ BUS_EXPENSE_PER - EE Business Expense Time Pers
      ⊞ CAREERPLAN - EE Career Plan
      ⊞ CITIZENSHIP - EE/Dependent Citizenship
      ⊞ CM_EVALUATIONS - Person Competency Evaluation
        COMP_CR_CRD - EE Company Credit Cards
        COMP_PROP - Company Prop Assigned to EE
        DEPENDENT_BENEF - EE Dependents/Beneficiaries
      ⊞ DISCIPLIN_ACTN - EE Disciplinary Actions
        DRIVERS_LIC - Driver's License
      ⊞ EMERGENCY_CNTCT - EE Emergency Contacts
```

Viewing the Record Hierarchy

The hierarchy shown in the query component view is not related to the query tree hierarchy shown when viewing access groups in the component view. It reflects an actual relationship between the record components, as defined in Application Designer using the Parent Record Name feature. For more information, see Setting up Query Security.

21. Double-click the record component you want to add to your query, or drag it into the designer view.

The query component view displays the record components you’ve joined. Since Query knows how these record components are related to each other, you don’t have to do anything more to complete the join.



PRIVATE.QUERY.Untitled
A = PERSONAL_DATA - EE Personal Data
B = ABSENCE_HIST - EE Absence History joined with A
Expressions
Prompts

Two record components in a hierarchical join

Creating Related Record Joins

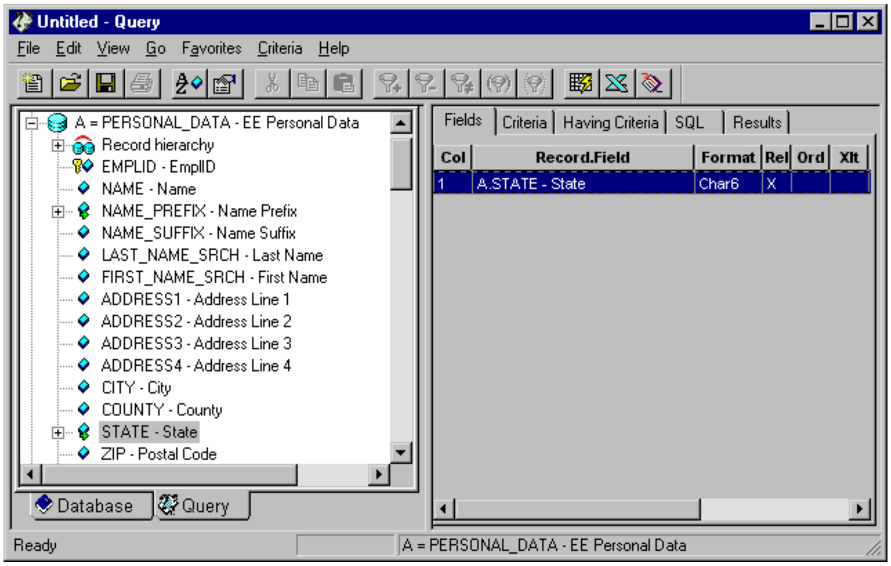
In a related record join, you join two tables based on a shared field that isn’t necessarily a key field. When you add a field to a record definition in the Application Designer, you have the option of identifying a prompt table for it. The system uses the prompt table to validate the data users enter into the field and to simplify data entry.

PeopleSoft Query displays fields that have related records in two ways:

- On the **Fields** tab in the designer view, it displays an *X* in the ***Rel*** column for each field that has a prompt table associated with it, such as the STATE field in the PERSONAL_DATA record component.

- In the query component view, it shows the related record symbol for the field.

You can see both of these visual cues in the example below:


Untitled - Query
File Edit View Go Favorites Criteria Help
Fields Criteria Having Criteria SQL Results
Database Query

A = PERSONAL_DATA - EE Personal Data
Record hierarchy
EMPLID - EmplID
NAME - Name
NAME_PREFIX - Name Prefix
NAME_SUFFIX - Name Suffix
LAST_NAME_SRCH - Last Name
FIRST_NAME_SRCH - First Name
ADDRESS1 - Address Line 1
ADDRESS2 - Address Line 2
ADDRESS3 - Address Line 3
ADDRESS4 - Address Line 4
CITY - City
COUNTY - County
STATE - State
ZIP - Postal Code

Col	Record.Field	Format	Rel	Ord	Xlt
1	A.STATE - State	Char6	X		

Ready A = PERSONAL_DATA - EE Personal Data

Query indicates related records in the query component view and in the **Rel** column

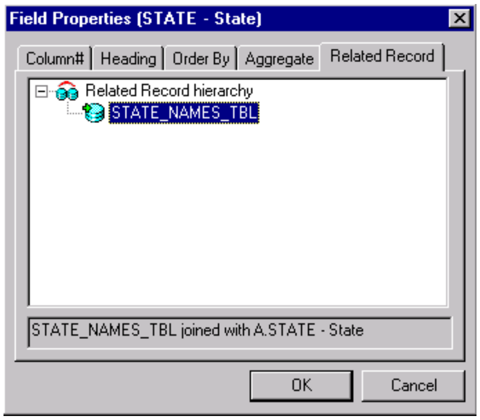
To create a related record join:

22. To join a related record component to the current record, expand the related record in the query component view.

23. Next, either double-click the related record or drag it into the designer view.

The query component view displays the record components you've joined, including the name of the field the two records share. Since Query knows how these record components are related to each other, you don't have to do anything more to complete the join.

You can also accomplish related record joins through the Field Properties dialog box.



STATE_NAMES_TBL joined with A.STATE - State

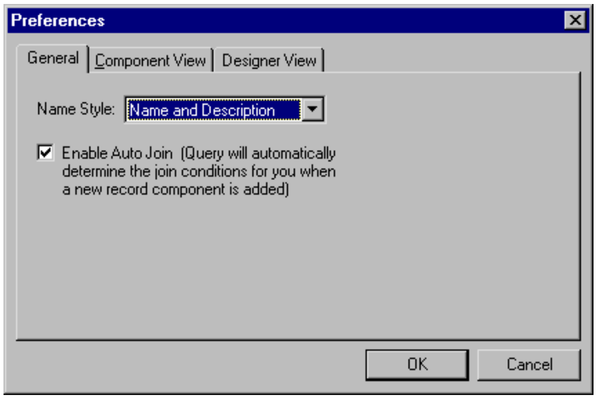
Field Properties dialog box

If a field is associated with a prompt table, double clicking on the Rel column in the designer view will display the related record hierarchy for that field.

Creating Any Record Joins with Auto Join

An Any Record join allows you to join the current record to any other record component to which you have access. You can choose whether or not you want Query to determine the conditions of the Any Record join by enabling the Auto Join option. When you select **View**, **Preferences** in Query, the Enable Auto Join option is automatically checked.

Auto Join first checks for a parent-child relationship between the base and join record. If one exists, a hierarchical join will be done. If no parent-child relationship exists, Auto Join then attempts to join on all matching key fields (excluding EFFDT and EFFSEQ).



Name Style: Name and Description

☑ Enable Auto Join (Query will automatically determine the join conditions for you when a new record component is added)

Preferences dialog box

To create an Any Record join with Auto Join:

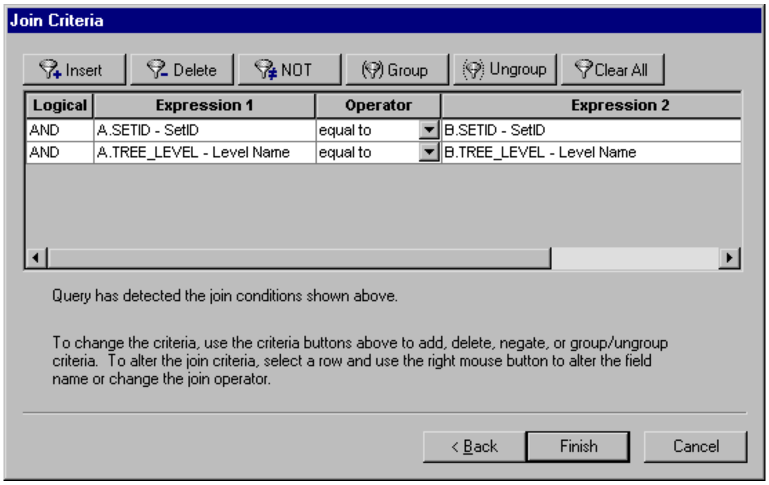
24. Select the first record component to add to your query from the database component view.

25. Click on the **Database** tab to go back to the database component view, and select the additional record component by double-clicking it.

    The Effective Date Options dialog box is displayed. This is the first dialog box in the Auto Join wizard, which guides you through the process of selecting criteria and setting up join conditions.

26. Select the appropriate effective date options, and click the **Next** button.

    The **Join Criteria** dialog box appears.



Logical	Expression 1	Operator	Expression 2
AND	A.SETID - SetID	equal to	B.SETID - SetID
AND	A.TREE_LEVEL - Level Name	equal to	B.TREE_LEVEL - Level Name

Query has detected the join conditions shown above.

To change the criteria, use the criteria buttons above to add, delete, negate, or group/ungroup criteria. To alter the join criteria, select a row and use the right mouse button to alter the field name or change the join operator.

Join Criteria dialog box

You use this dialog to create an Any Record join.

27. Follow the directions on the dialog box to modify the join conditions, and if you are finished, click the **Finish** button.

To go back a step in the wizard, click the **Back** button.

Note. You need to understand SQL to ensure that you create logical and efficient Any Record joins.

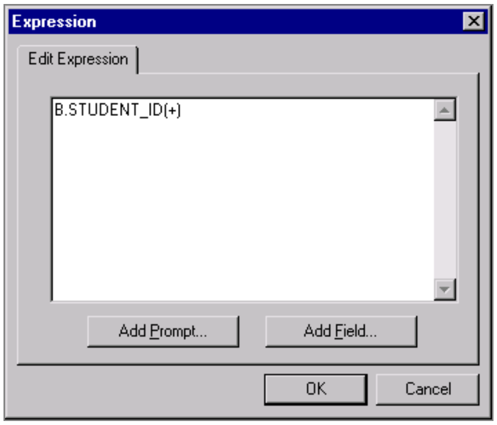
Creating an Outer Join

You must include a plus sign in parentheses (+) after the key field(s) of the subordinate record in the criteria that link the records. To produce this syntax, you cannot use the predefined joins. You must perform an *Any Join* and code an Expression that contains the (+) instead of a field.

To produce an outer join using Studend_Data & Student_Eval:

28. Click the *Criteria* tab and drag and drop Student_ID from the primary record, Student_Data.

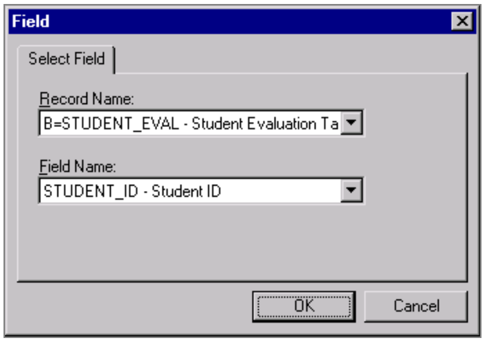
29. Accept the default Operator of *Equal To,* and right-click in the *Expression (2)* column. Select *Expression* from the drop-down menu, and the Edit Expression dialog box is displayed.



B.STUDENT_ID(+)

Expression dialog box

30. Click *Add Field*. Select Student_Eval for the Record Name and Student_ID for the Field Name.



Field dialog box

31. Manually type in (+) after the subordinate field.

Outer Join Equivalent

Even though you can only perform an Outer join in Query on select platforms, you can achieve the effect of an Outer join if your platform doesn't allow that selection. In the following example, we'll retrieve a list of all students and any evaluation information on file, and create a query using a subquery and a union. The first select will be retrieving those students who have not filled out an evaluation and the second select will be retrieving those students who have filled out an evaluation.

To perform an Outer join equivalent:

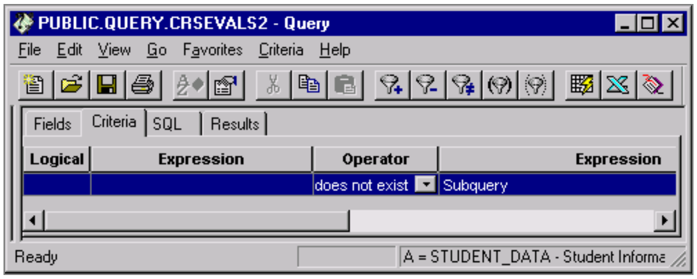
32. Start with ***Student_Data*** and select ***Customer_ID*** and ***Name.***

Since you'll be selecting two different fields from the Student_Eval record and you want them to have their own columns, you'll need to create two dummy fields for your first record.

33. Create two different expressions and select them for output.

34. Save the query as CRSEVALS2.

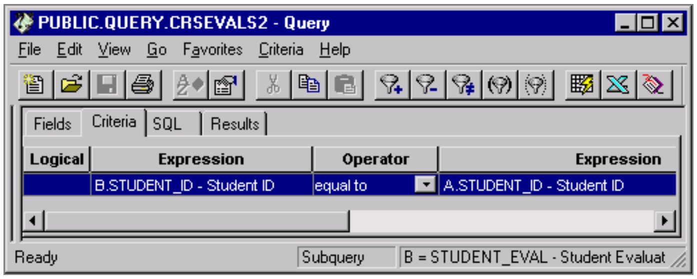
Now create your subquery to find those students who have not filled out an evaluation.


PUBLIC.QUERY.CRSEVALS2 - Query
File Edit View Go Favorites Criteria Help
Fields Criteria SQL Results
Ready
A = STUDENT_DATA - Student Informa

Logical	Expression	Operator	Expression
		does not exist	Subquery

Query CRESEVALS2

Subquery:


PUBLIC.QUERY.CRSEVALS2 - Query
File Edit View Go Favorites Criteria Help
Fields
Criteria
SQL
Results

Logical	Expression	Operator	Expression
	B.STUDENT_ID - Student ID	equal to	A.STUDENT_ID - Student ID

Ready | Subquery | B = STUDENT_EVAL - Student Evaluat

Subquery for CRESEVALS2

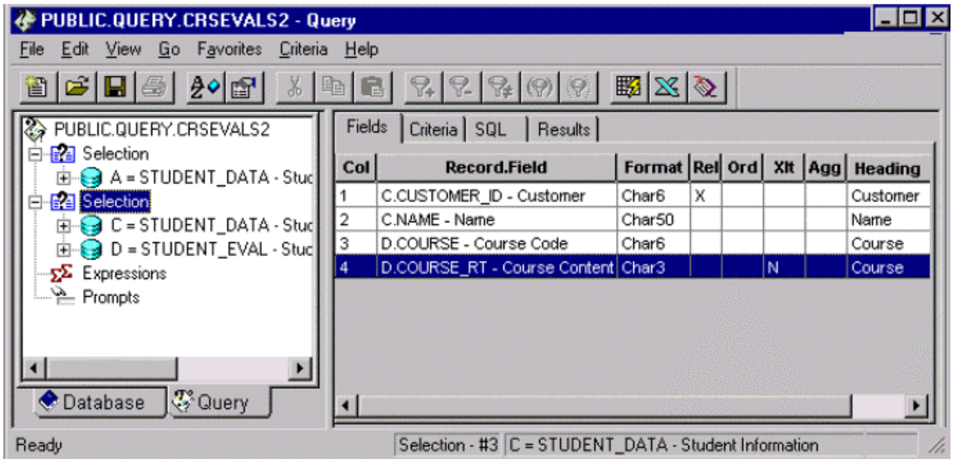
Your first select is now complete. If you run the query at this point, you should only see the 155 students that did not fill out an evaluation.

35. To join in the students who did fill out an evaluation, create a union.

36. Start with Student_Data and select Customer_ID and Name for output.

37. Next join Student_Eval using a Record Hierarchy join.

38. Select Course and Course_Rt for output.


PUBLIC.QUERY.CRSEVALS2 - Query
File Edit View Go Favorites Criteria Help
Database
Query
Fields
Criteria
SQL
Results

PUBLIC.QUERY.CRSEVALS2
Selection
A = STUDENT_DATA - Stuc
Selection
C = STUDENT_DATA - Stuc
D = STUDENT_EVAL - Stuc
Expressions
Prompts

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	C.CUSTOMER_ID - Customer	Char6	X				Customer
2	C.NAME - Name	Char50					Name
3	D.COURSE - Course Code	Char6					Course
4	D.COURSE_RT - Course Content	Char3			N		Course

Ready | Selection - #3 | C = STUDENT_DATA - Student Information

Query page with Student Information

Unions

Unions allow you to get the results from two or more separate queries at the same time. You can only create a union of multiple queries when the queries have the following common elements:

- The same number of selected fields.
- The same data types for all fields.
- The same display order for the columns.

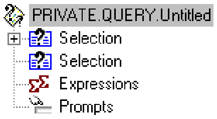
Note. You need to understand SQL to ensure that you create logical union queries.

To create a union:

39. Select the query component, and then either select **Edit, New Union** from the main menu or click the Create a new union toolbar button .

Query then automatically switches to the database tab of the component view so that you can start defining the second query. You define the query just as you do any other query.

When you're defining a union, the query component view shows each query as a "Selection" component. This enables you to move easily from one select statement to another.



```
PRIVATE.QUERY.Untitled
  Selection
  Selection
  Expressions
  Prompts
```

Viewing selection components in a union

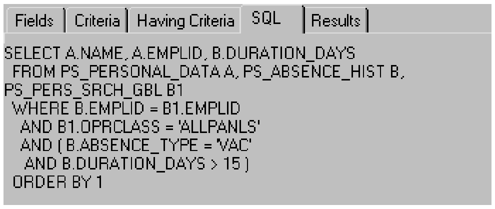
When you're working on a union, each individual selection component looks like an independent query, and for the most part they *are* independent. However, the first selection component in the union—the one you started before selecting **New Union**—has a special status. Query determines the ordering of the rows and columns based on what you specify for the first selection. It also uses the column headings defined for the first selection.

Viewing the Underlying SQL Statements

Like all client applications, PeopleSoft Query communicates with the database using SQL. As you define your query using the various panels and dialog boxes, Query is working behind the scenes to convert your instructions into a SQL statement that retrieves the data you want. If you're familiar with SQL, you might find it helpful to review the SQL statement that Query builds.

To view an underlying SQL statement:

40. Select the **SQL** tab in the designer view to see the SQL SELECT statement that Query creates.



```
SELECT A.NAME, A.EMPLID, B.DURATION_DAYS
 FROM PS_PERSONAL_DATA A, PS_ABSENCE_HIST B,
PS_PERS_SRCH_GBL B1
 WHERE B.EMPLID = B1.EMPLID
  AND B1.OPRCLASS = 'ALLPANLS'
  AND ( B.ABSENCE_TYPE = 'VAC'
   AND B.DURATION_DAYS > 15 )
 ORDER BY 1
```

Viewing the SQL statement for your query

You can't edit the text in the SQL dialog box, but you can copy it to the Microsoft Windows clipboard.

41. Highlight the text you want by dragging the mouse over it, then right-click and select **Copy** from the pop-up menu.

Note. You can also use the standard Windows keyboard shortcut (CTRL+C) to copy the text to the clipboard or use the Copy button on the toolbar or select **Edit, Copy**.

---

42. After you've copied the text to the Windows clipboard using one of these methods, you can paste it into another file or application.

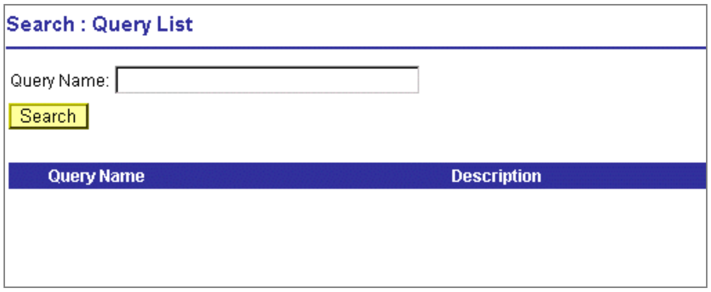
CHAPTER 6

Running a Query on the Web

With PeopleSoft 8, you can run a predefined query from your browser and view it online.

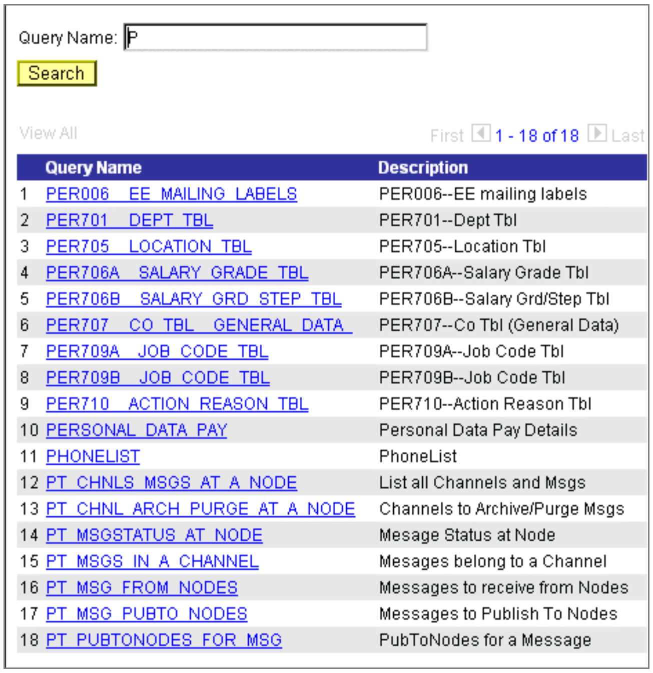
To run a predefined query using the PeopleSoft Internet Architecture:

1. Open your browser and select **PeopleTools, PSQuery** to display the Search: Query List page.



Search: Query List page

2. If you know the name of the query you want to run, type it in the **Query Name** field. If you want to search for a query, leave the field blank and click **Search** to display a list of all queries. You can also enter the first letter of a query name to narrow your search. In this example, we've searched for queries that begin with ***P***.



	Query Name	Description
1	PER006__EE_MAILING_LABELS	PER006--EE mailing labels
2	PER701__DEPT_TBL	PER701--Dept Tbl
3	PER705__LOCATION_TBL	PER705--Location Tbl
4	PER706A__SALARY_GRADE_TBL	PER706A--Salary Grade Tbl
5	PER706B__SALARY_GRD_STEP_TBL	PER706B--Salary Grd/Step Tbl
6	PER707__CO_TBL__GENERAL_DATA_	PER707--Co Tbl (General Data)
7	PER709A__JOB_CODE_TBL	PER709A--Job Code Tbl
8	PER709B__JOB_CODE_TBL	PER709B--Job Code Tbl
9	PER710__ACTION_REASON_TBL	PER710--Action Reason Tbl
10	PERSONAL_DATA_PAY	Personal Data Pay Details
11	PHONELIST	PhoneList
12	PT_CHNLS_MSGS_AT_A_NODE	List all Channels and Msgs
13	PT_CHNL_ARCH_PURGE_AT_A_NODE	Channels to Archive/Purge Msgs
14	PT_MSGSTATUS_AT_NODE	Mesage Status at Node
15	PT_MSGS_IN_A_CHANNEL	Mesages belong to a Channel
16	PT_MSG_FROM_NODES	Messages to receive from Nodes
17	PT_MSG_PUBTO_NODES	Messages to Publish To Nodes
18	PT_PUBTONODES_FOR_MSG	PubToNodes for a Message

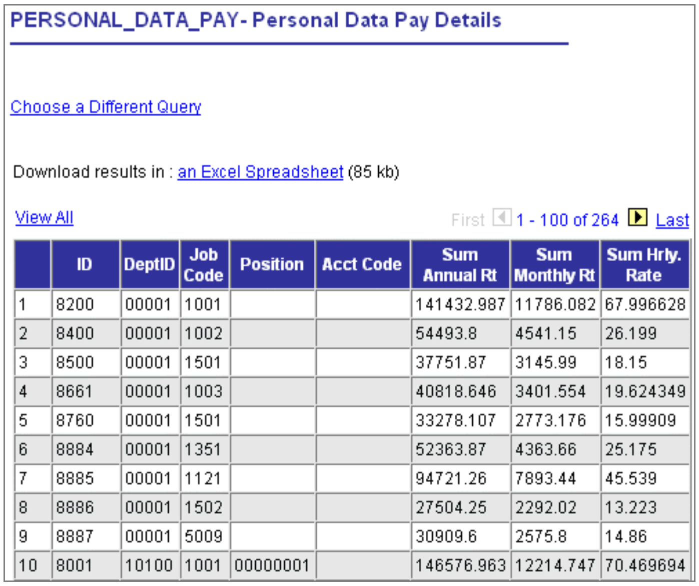
Query List page

The Query List page appears. The Query List shows the name of the query and a short description of the query.

3. Scroll to the name of the query that you want to run.

   By default, only the first 100 queries appear on the page. To see more of the list, select the **Next** or **Last.** To display all of the queries, select **View All** and use the scrollbar to go through the rest of the list**.**

4. Click the query name from the list of queries. In this example, we'll run the **PERSONAL_DATA_PAY** query.



PERSONAL_DATA_PAY- Personal Data Pay Details

Choose a Different Query

Download results in : an Excel Spreadsheet (85 kb)

	ID	DeptID	Job Code	Position	Acct Code	Sum Annual Rt	Sum Monthly Rt	Sum Hrly. Rate
1	8200	00001	1001			141432.987	11786.082	67.996628
2	8400	00001	1002			54493.8	4541.15	26.199
3	8500	00001	1501			37751.87	3145.99	18.15
4	8661	00001	1003			40818.646	3401.554	19.624349
5	8760	00001	1501			33278.107	2773.176	15.99909
6	8884	00001	1351			52363.87	4363.66	25.175
7	8885	00001	1121			94721.26	7893.44	45.539
8	8886	00001	1502			27504.25	2292.02	13.223
9	8887	00001	5009			30909.6	2575.8	14.86
10	8001	10100	1001	00000001		146576.963	12214.747	70.469694

PERSONAL_DATA_PAY query page

Downloading the Query as an Excel Spreadsheet

You have the option of downloading your query as an Excel spreadsheet. Click the text ***an Excel Spreadsheet*** to open the file in your browser in Excel. If you are using Excel 2000, the Excel spreadsheet will appear in your browser. To save it, select **File, Save As.** If you are using Office 97, the **File download** dialog box appears. You have the choice of opening the file in your browser or saving it to disk.

Troubleshooting the Download in Excel Format Option

If your Query does not download to Excel correctly, there are several things you can check. Depending on your version of Excel and your environment or configuration, you may have to use one of the following workarounds.

Excel 97

If Excel does not launch when you choose to view the spreadsheet in your browser, launch Excel separately and leave it running in the background. Try the download again and your query will appear in Excel format in your browser. You do not need to launch Excel to save the query to your hard disk as an Excel spreadsheet.

Excel 2000

In some environments, Seagate Crystal Report Wizard must be disabled to download a query in Excel format. To do this, select the **Tools, Add-Ins** in from the Excel toolbar. Deselect the checkbox for Seagate Crystal Report Wizard. After you click **OK,** you can reselect the Report Wizard option without affecting the query download option in the future.

Performing a Lookup

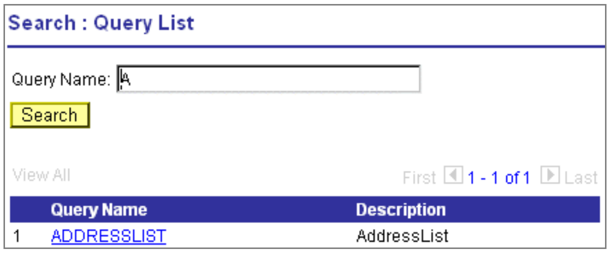
There are some fields in Internet Query that allow you to search for certain field values. If you can perform a search, the 🔍 button appears on the screen.

For example, you might know an employee’s last name, but not have their employee ID handy. Using the 🔍 button, you can enter the information that you *do* have to find the required information that you *don’t* have.

In this example, we will perform a basic and an advanced lookup using ADDRESSLIST.

To perform a basic lookup:

5. Click **ADDRESSLIST** query name from the Query List.



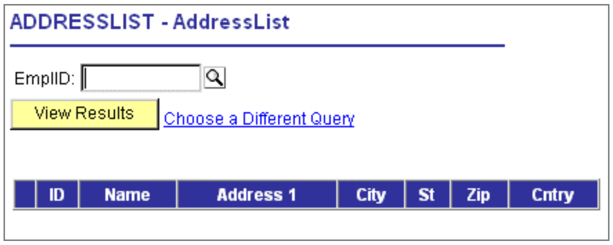
Search : Query List

Query Name: A

	Query Name	Description
1	ADDRESSLIST	AddressList

Search: Query List page

6. If you know the employee ID for the address you are looking up, type it in the **EmplID** field. If you need to search for the employee ID, click 🔍.



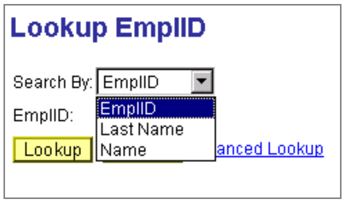
ADDRESSLIST - AddressList

EmplID:

	ID	Name	Address 1	City	St	Zip	Cntry

ADDRESSLIST page

7. On the Lookup page, click the drop-down arrow to find more search values. In this example, you can select **EmplID, Name,** or **Last Name.** (For more search values, click **Advanced Lookup.)**



Search By: EmplID

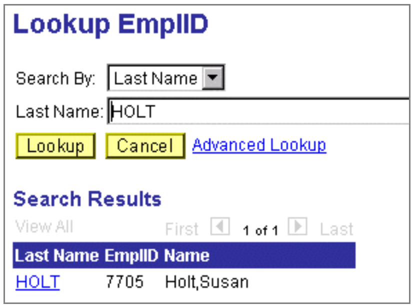
Lookup EmplID page

Note. To find **all** values for this field, leave the search field blank and click **Lookup.** You can also display all of the search fields at once by clicking the link for **Advanced Lookup.**

8. Select the value to **Search By** and click **Lookup.**

   In this example, we know the employee's last name, but not their employee ID.

   Your lookup information appears on the page.



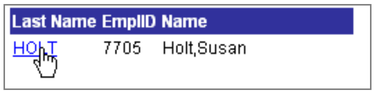
Search By: Last Name

Last Name: HOLT

Last Name	EmplID	Name
HOLT	7705	Holt,Susan

Lookup EmplID page

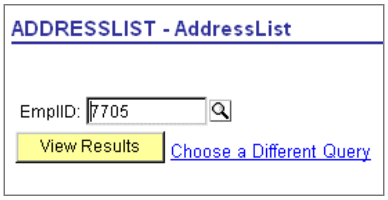
9. Select the **link** in the result field.



Last Name	EmplID	Name
HOLT	7705	Holt,Susan

Select the link

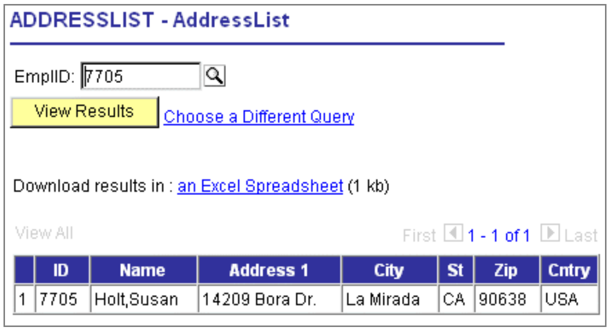
The Query page appears with the required value already filled in.



EmplID: 7705

ADDRESSLIST – AddressList page

10. Click **View Results**, and the query results appear on the page.



ADDRESSLIST - AddressList

EmplID: 7705

Download results in : an Excel Spreadsheet (1 kb)

	ID	Name	Address 1	City	St	Zip	Cntry
1	7705	Holt,Susan	14209 Bora Dr.	La Mirada	CA	90638	USA

ADDRESSLIST – AddressList page

CHAPTER 7

Using Query Designer on the Web

The addition of Query Designer to PeopleSoft Internet Architecture provides Web users a simple, intuitive way to create and run queries using their browser. Casual users have an interface that is familiar and easy to understand, with the more complex Query development features residing in the Windows client. Query view development is still available in Application Designer.

For more information about building views with Application Designer, see Building SQL Tables and Views.

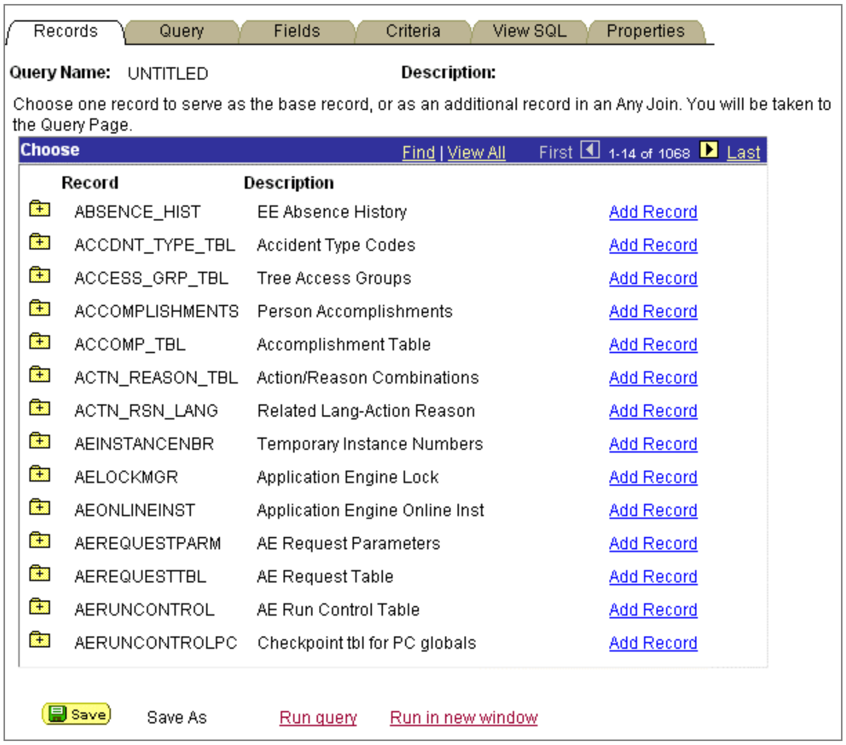
Creating a new Query on the Web

The steps used to create a query on the Web are very similar to creating a query in Windows. Now, instead of using drag-and-drop mouse actions, you'll use point-and-click to select records, fields, and criteria.

Open Query Designer by selecting **PeopleTools, Query Designer, Use, Create New Query.**

Selecting Record Components

On the Record page, you'll see a list of records to which you have access. This is where you choose the record that becomes the starting point for your query. After you select your base record, you are taken to the Query page so that you can select the fields you want in your output.



Query Name: UNTITLED Description:

Choose one record to serve as the base record, or as an additional record in an Any Join. You will be taken to the Query Page.

Record	Description	
ABSENCE_HIST	EE Absence History	Add Record
ACCDNT_TYPE_TBL	Accident Type Codes	Add Record
ACCESS_GRP_TBL	Tree Access Groups	Add Record
ACCOMPLISHMENTS	Person Accomplishments	Add Record
ACCOMP_TBL	Accomplishment Table	Add Record
ACTN_REASON_TBL	Action/Reason Combinations	Add Record
ACTN_RSN_LANG	Related Lang-Action Reason	Add Record
AEINSTANCENBR	Temporary Instance Numbers	Add Record
AELOCKMGR	Application Engine Lock	Add Record
AEONLINEINST	Application Engine Online Inst	Add Record
AEREQUESTPARM	AE Request Parameters	Add Record
AEREQUESTTBL	AE Request Table	Add Record
AERUNCONTROL	AE Run Control Table	Add Record
AERUNCONTROLPC	Checkpoint tbl for PC globals	Add Record

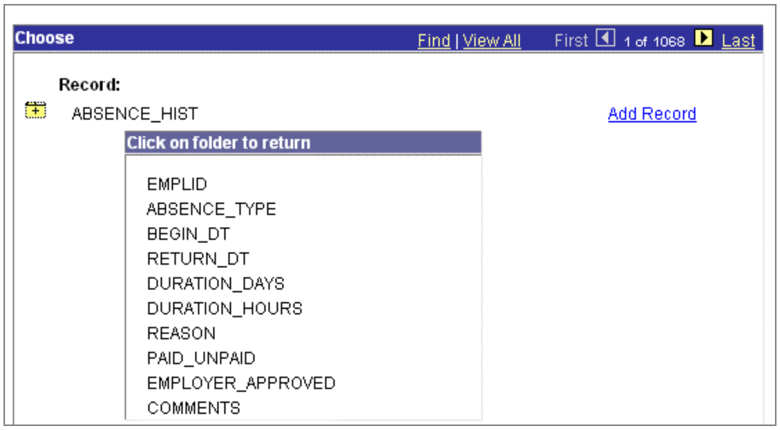
Record page

In many cases, you will have many more records than can be displayed on the screen. Selecting View All will cause all of your records to be displayed on the page, and will require you to scroll through a very long list to find your record. The easiest way to find a specific record is to select Find from the Navigation bar. The Find link will start your browser's Find utility where you can enter a search string and locate the record that you need.

Note. Enter as much of the record name as possible to find the correct record.

Viewing the fields associated with a record

You can look at the fields associated with a record by clicking the folder icon next to the record name. Query Designer expands the record so that you can see the fields and make sure that this record has the content that you want. Click the folder icon again to return to the database page.



Record:

ABSENCE_HIST

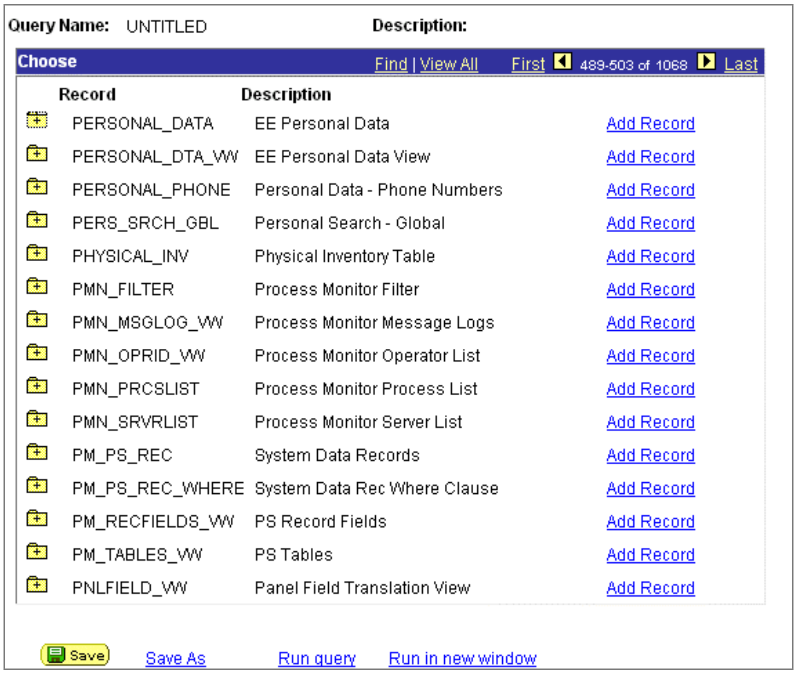
EMPLID
ABSENCE_TYPE
BEGIN_DT
RETURN_DT
DURATION_DAYS
DURATION_HOURS
REASON
PAID_UNPAID
EMPLOYER_APPROVED
COMMENTS

Choose Record page

To find a record:

1. Select PeopleTools, Query Designer, Use, Create New Query.
2. Select **Find** and enter **PERSONAL_DATA** in the Find dialog box.

   Query Designer displays the record at the top of the page.



Query Name: UNTITLED

Description:

Record	Description	
PERSONAL_DATA	EE Personal Data	Add Record
PERSONAL_DTA_VW	EE Personal Data View	Add Record
PERSONAL_PHONE	Personal Data - Phone Numbers	Add Record
PERS_SRCH_GBL	Personal Search - Global	Add Record
PHYSICAL_INV	Physical Inventory Table	Add Record
PMN_FILTER	Process Monitor Filter	Add Record
PMN_MSGLOG_VW	Process Monitor Message Logs	Add Record
PMN_OPRID_VW	Process Monitor Operator List	Add Record
PMN_PRCSLIST	Process Monitor Process List	Add Record
PMN_SRVRLIST	Process Monitor Server List	Add Record
PM_PS_REC	System Data Records	Add Record
PM_PS_REC_WHERE	System Data Rec Where Clause	Add Record
PM_RECFIELDS_VW	PS Record Fields	Add Record
PM_TABLES_VW	PS Tables	Add Record
PNLFIELD_VW	Panel Field Translation View	Add Record

Record page

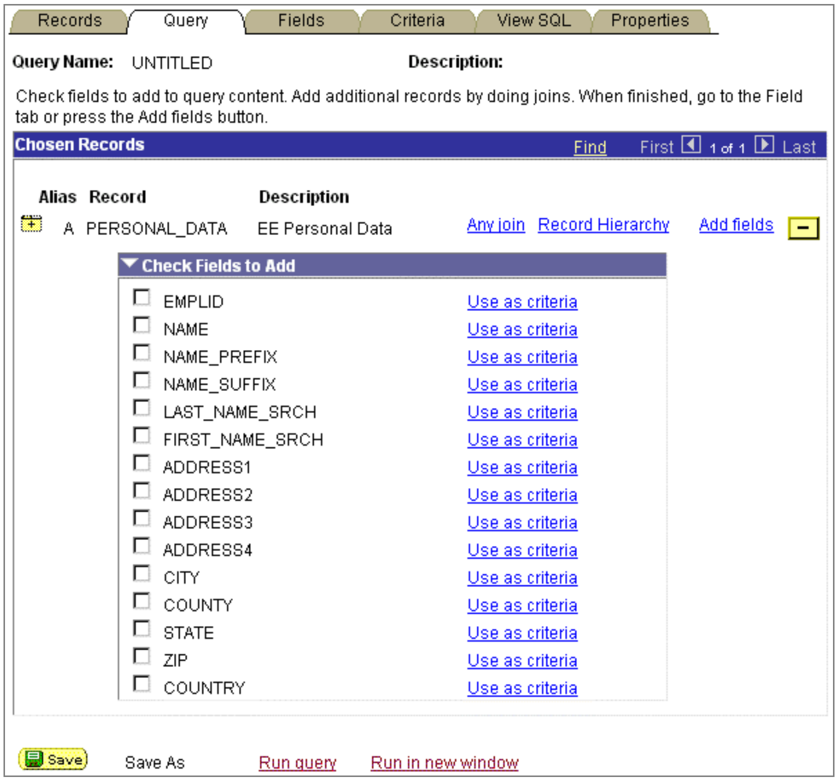
3. Click Add Record.

   The Query page appears listing the fields associated with that record.

Note. In most cases, you will only select one base record from this page. If you navigate back and select a second base record, you will be creating an Any Join for the two records.

Adding Fields and Criteria to your Query

Once you've selected a record, Query Designer automatically switches to the Query page. The Query page shows you all of the fields that are associated with the record and enables you to add or delete fields and criteria to your query.



Query Name: UNTITLED **Description:**

Check fields to add to query content. Add additional records by doing joins. When finished, go to the Field tab or press the Add fields button.

Chosen Records Find First 1 of 1 Last

Alias	Record	Description			
A	PERSONAL_DATA	EE Personal Data	Any join	Record Hierarchy	Add fields

Check Fields to Add

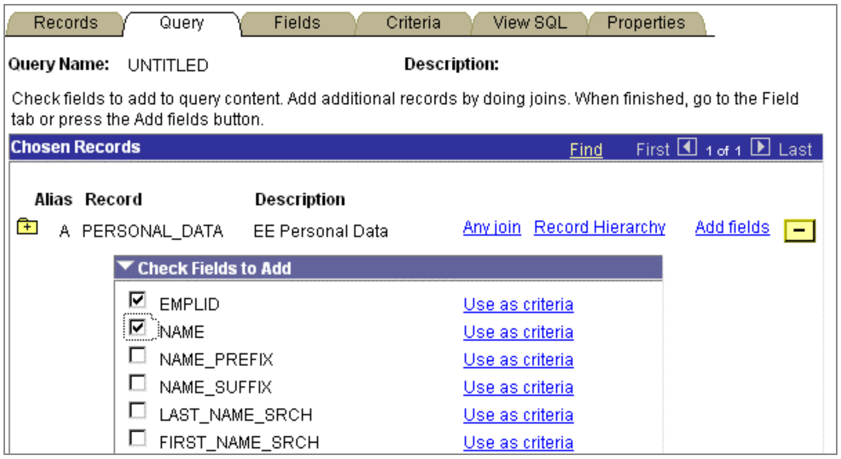
□ EMPLID	Use as criteria
□ NAME	Use as criteria
□ NAME_PREFIX	Use as criteria
□ NAME_SUFFIX	Use as criteria
□ LAST_NAME_SRCH	Use as criteria
□ FIRST_NAME_SRCH	Use as criteria
□ ADDRESS1	Use as criteria
□ ADDRESS2	Use as criteria
□ ADDRESS3	Use as criteria
□ ADDRESS4	Use as criteria
□ CITY	Use as criteria
□ COUNTY	Use as criteria
□ STATE	Use as criteria
□ ZIP	Use as criteria
□ COUNTRY	Use as criteria

Query page

To select fields for the query:

4. Click the checkbox by each field you want to include in the query output.

   In this example, we are selecting the EMPLID and NAME fields for query output.



Query Name: UNTITLED Description:

Check fields to add to query content. Add additional records by doing joins. When finished, go to the Field tab or press the Add fields button.

Chosen Records

Alias	Record	Description
A	PERSONAL_DATA	EE Personal Data

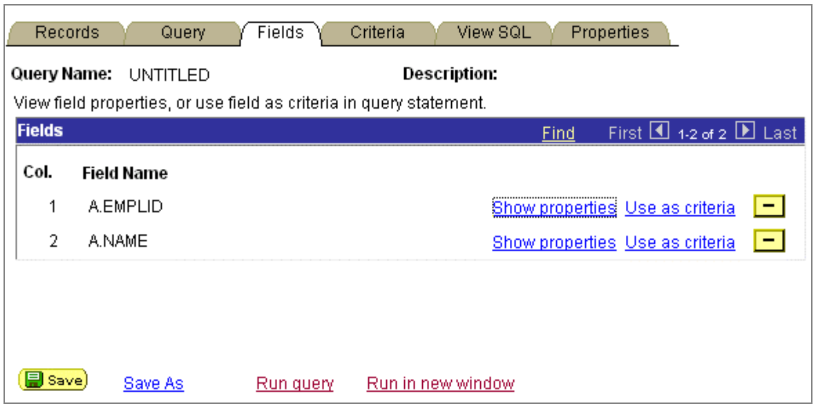
Check Fields to Add

- ☑ EMPLID
- ☑ NAME
- □ NAME_PREFIX
- □ NAME_SUFFIX
- □ LAST_NAME_SRCH
- □ FIRST_NAME_SRCH

Query page with EMPLID and NAME fields selected

5. Click Add Fields.

The Fields page appears listing all of the fields that you've selected for output.



Query Name: UNTITLED Description:

View field properties, or use field as criteria in query statement.

Fields

Col.	Field Name
1	A.EMPLID
2	A.NAME

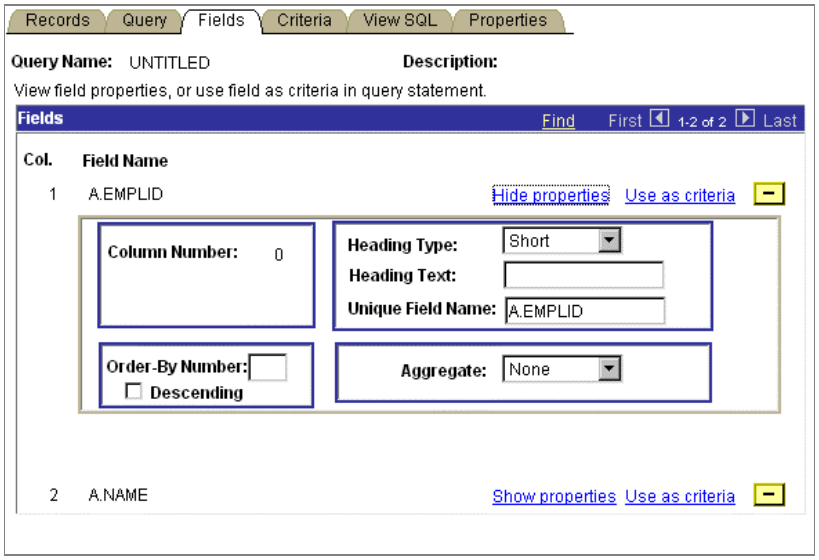
Fields page

Note. If you navigate to another page in Query Designer without clicking Add Fields, any field that you have selected from this page will be added to the query automatically.

Formatting Output

When you select a field for query output, the system automatically assigns a heading to display at the top of the column. By default, it uses the short name of the field from the record definition (RTF short). You can use the field properties to change the heading to any text you'd like.

By selecting the Show Properties link, you can view the properties of each field and make changes to headings, order-by numbers, and aggregate values.



Records | Query | Fields | Criteria | View SQL | Properties

Query Name: UNTITLED Description:

View field properties, or use field as criteria in query statement.

Fields Find First 1-2 of 2 Last

Col.	Field Name		
1	A.EMPLID	Hide properties	Use as criteria

Column Number: 0

Heading Type: Short

Heading Text:

Unique Field Name: A.EMPLID

Order-By Number:

□ Descending

Aggregate: None

Col.	Field Name		
2	A.NAME	Show properties	Use as criteria

Field Properties page

Also on this page, you can provide a heading text and a unique field name. If you do not provide a unique field name, Query provides a default name, which is a single-letter alias for the record followed with the record field name, for example A.NAME or B.EMPLID.

Note. Column ordering is not supported in Query Designer at this time. If you need to change the column order for a field, you must delete the fields that are out of order and add them individually in the order you want. Use the Query page to add the fields back into the query.

To change the properties for query output:

6. Select the appropriate heading option.

   The following list describes the options available for column headings.

Heading Type

None. The column does not have a heading.

Text. The column heading is the text you’ve entered in the text box.

Short. The column heading is the short name from the record definition.

Long. The column heading is the long name from the record definition.

Heading Text

Text displayed at the top of the column.

Unique Field Name

Used for translations. There is no need to change this field.

7. If you want to sort the output, enter a number in the order-by number field. You can also choose whether the sort is ascending or descending by selecting that checkbox.

8. If are using aggregate values, select the aggregate function value for the field.

   An aggregate function is a special type of operator that returns a single value based on multiple rows of data. When your query includes one or more aggregate functions, Query collects together related rows and displays a single row that summarizes their contents.

   The table below lists the aggregate functions you can apply to a field using Query Designer.

Aggregate Function	*Action*
Sum	Adds the values from each row and displays the total.
Count	Counts the number of rows.
Min	Checks the value from each row and returns the lowest one.
Max	Checks the value from each row and returns the highest one.
Average	Adds the values from each row and divides the result by the number of rows.
None	Will not use aggregate functions.
N/A	Doesn't' apply to this field.

Adding Criteria to a Query

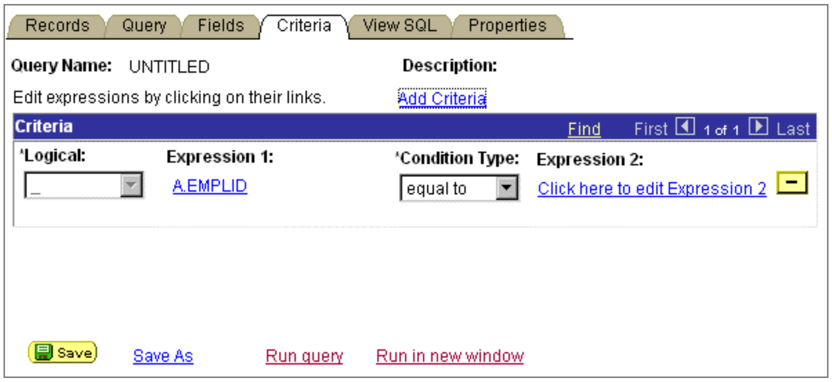
To use the field as criteria for the query, select **Use as Criteria** from either the Fields or the Query page. Query Designer opens the Criteria page for that field. You can specify the criteria for that field, and then return to the Fields page to add criteria to other fields, if you wish. A field from a record can be used as criteria even if you have not chosen that field for query output. You can also simply select the Criteria page to create criteria that is not based on a field.

In most cases, you don't want all the rows of data from the table, just the rows that meet certain criteria. To limit the returned rows, you define *selection criteria*. If you know some SQL, you'll notice that the criteria correspond to the WHERE clause of the SELECT statement.

For more information about using criteria in your queries, see Specifying Query Selection Criteria.

Note. Effective date operators are not currently supported in Query Designer. If you need to select something other than the current effective date from a table, we recommend that you build a view on the record with an effective-dated subquery in SQL view text. Grant security access for the record and use this record for additional effective dates.

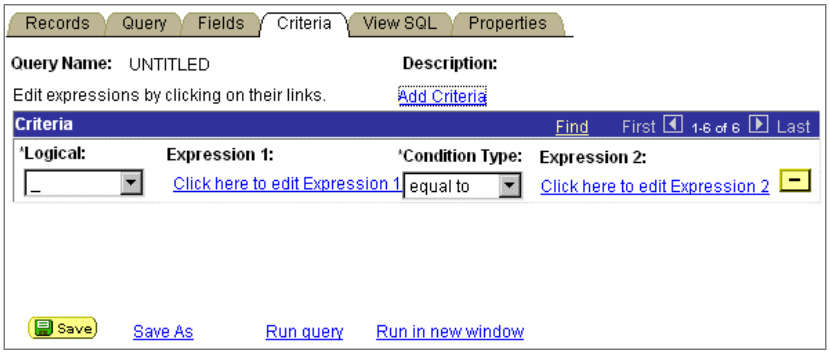
You can add criteria to your query in three different ways. From the Query and Fields pages, if you select Use as Criteria, you add the field as criteria and can edit the Expression 2 part of the criteria.



Query Name: UNTITLED
Description:
Edit expressions by clicking on their links.
Add Criteria
Criteria 1 of 1
*Logical: _
Expression 1: A.EMPLID
*Condition Type: equal to
Expression 2: Click here to edit Expression 2

Criteria page with a field as criteria

If you select Add Criteria from the Criteria page, you can edit both Expression 1 and Expression 2.



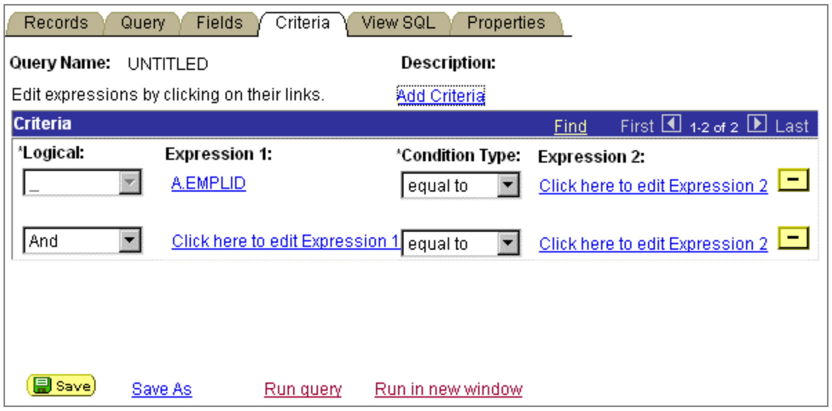
Query Name: UNTITLED
Description:
Edit expressions by clicking on their links.
Add Criteria
Criteria 1-6 of 6
*Logical: _
Expression 1: Click here to edit Expression 1
*Condition Type: equal to
Expression 2: Click here to edit Expression 2

Criteria page

To enter selection criteria:

9. Select **Use as Criteria** for the field you want to base the selection criterion on or select the **Add Criteria** link from the Criteria page.

The **Add Criteria** link adds a new row of criteria to the page that you can modify to suit your needs.



Query Name: UNTITLED
Description:
Edit expressions by clicking on their links.

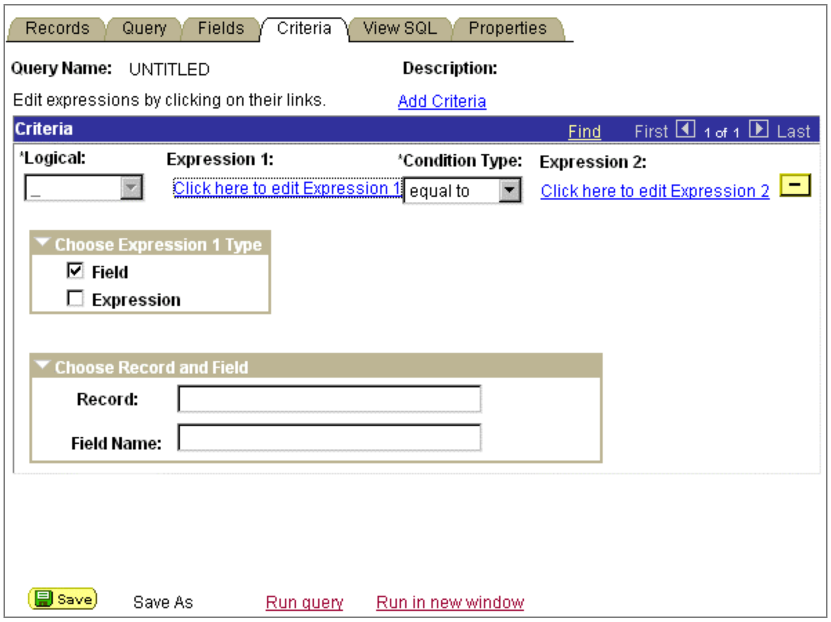
Criteria 1-2 of 2

*Logical:	Expression 1:	*Condition Type:	Expression 2:
_	A.EMPLID	equal to	Click here to edit Expression 2
And	Click here to edit Expression 1	equal to	Click here to edit Expression 2

Criteria page with added row for new criteria

10. Click the fieldname to edit the first (left-hand) expression. You have the choice of **Field** or **Expression**.

Select the Field checkbox to edit the field and record information.



Query Name: UNTITLED
Description:
Edit expressions by clicking on their links.

Criteria 1 of 1

*Logical:	Expression 1:	*Condition Type:	Expression 2:
_	Click here to edit Expression 1	equal to	Click here to edit Expression 2

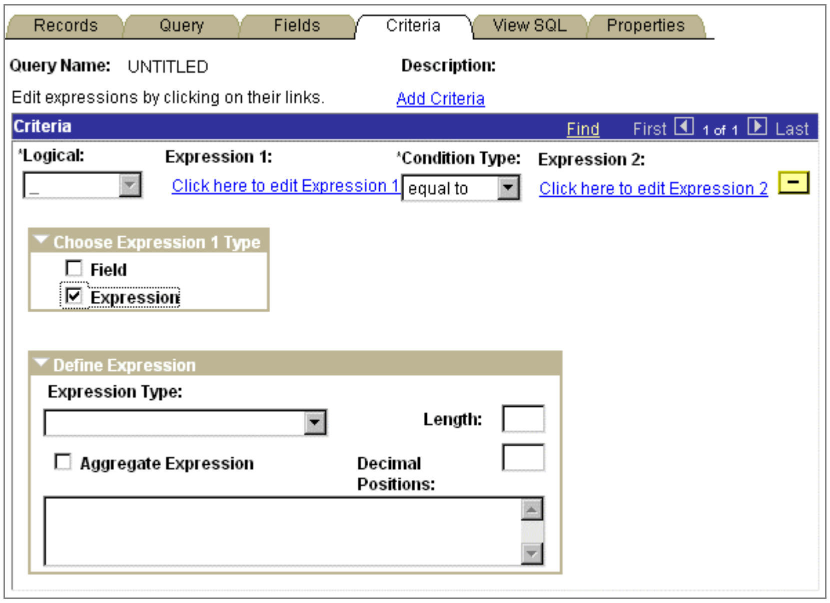
Choose Expression 1 Type
☑ Field
☐ Expression

Choose Record and Field
Record:
Field Name:

Criteria page with Field selected for Expression 1

11. If you are choosing an expression, enter that information.

Select an expression type from the drop-down list. If you are entering an aggregate value, make sure that you select the Aggregate Expression checkbox. You can also enter parameters for length and decimal positions. Finally, enter the expression in the textbox. Query Designer will insert it into the SQL for you.



Query Name: UNTITLED

Description:

Edit expressions by clicking on their links.

Add Criteria

Criteria 1 of 1

*Logical: _

Expression 1: Click here to edit Expression 1

*Condition Type: equal to

Expression 2: Click here to edit Expression 2

Choose Expression 1 Type

☐ Field

☑ Expression

Define Expression

Expression Type:

Length:

☐ Aggregate Expression

Decimal Positions:

Criteria page with Expression selected for Expression 1

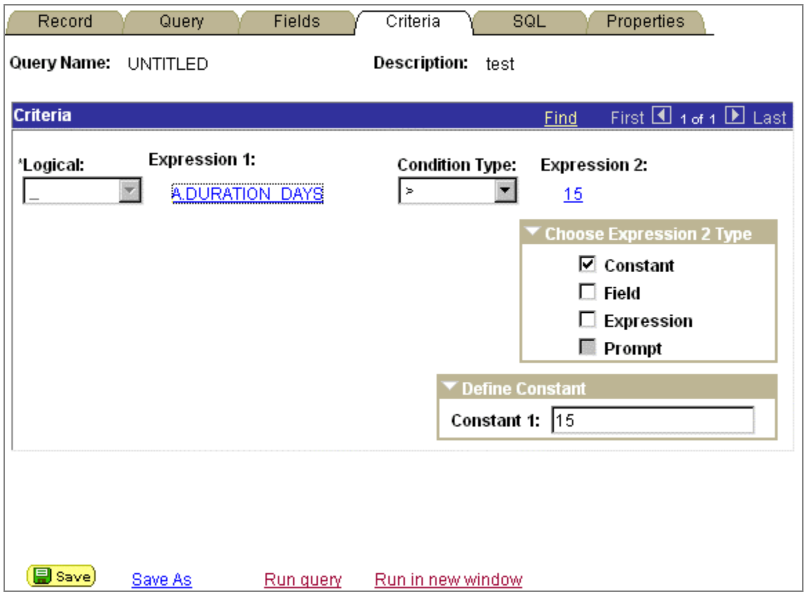
12. Select Condition Type.

The condition type determines how Query Designer compares the values of the first (left-hand) expression to the second (right-hand) expression. Click the drop-down button in the **Condition Type** column and select the appropriate comparison operator from the pop-up menu.

For more information about the available operators, see Selecting Operators.

13. Click the link to edit the second (right-hand) Expression column.

The Expression 2 Type and Choose Record and Field edit boxes appear on the Criteria page.



Query Name: UNTITLED
Description: test

Criteria

*Logical: _
Expression 1: A.DURATION_DAYS
Condition Type: >
Expression 2: 15

Choose Expression 2 Type

☑ Constant
☐ Field
☐ Expression
☐ Prompt

Define Constant

Constant 1: 15

Criteria page with Expression 2 edit boxes

The table below describes all the available value types.

Value Type	***What Query Compares the Value in the Selected Field To***
Field	The value in another field, usually a field in another record component. To compare the values from fields in two records, you need to join the record components. See Advanced Query Options.
Expression	An expression you enter, which Query evaluates before comparing the result to the value in the selected field. Query evaluates the expression once for each row it tests.
Constant	A single fixed value.

Note. Not all value types are available for all condition types. After you select a condition type, PeopleSoft Query displays only the value types that are available for that condition.

For more information, see Entering Comparison Values.

14. Typically, this is the point at which you would save your query.

For more information about saving your query, see Saving a Query.

Note. After you save a query, you cannot go back and delete the base record.

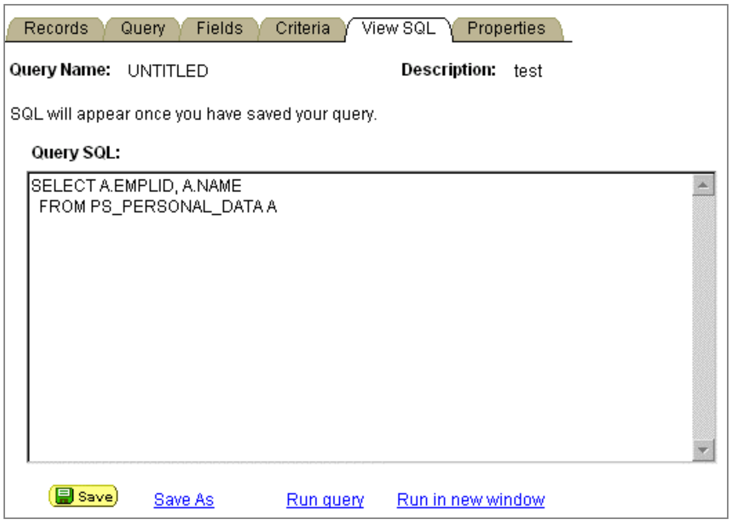
SQL Page

PeopleSoft Query communicates with the database using SQL. As you define your query using the various pages, Query is working behind the scenes to convert your instructions into a SQL statement that retrieves the data you want. If you're familiar with SQL, you might find it helpful to review the SQL statement that Query builds.

Note. In Query Designer, the SQL is not generated until you have saved your query for the first time. If this page is blank, save the query and the SQL will be generated for you.

To view an underlying SQL statement:

15. Select the **SQL** tab to see the SQL SELECT statement that Query creates.



Query Name: UNTITLED Description: test

SQL will appear once you have saved your query.

Query SQL:

```
SELECT A.EMPLID, A.NAME
 FROM PS_PERSONAL_DATA A
```

SQL page

You can't edit the text in the SQL dialog box, but you can copy it to the clipboard.

16. Highlight the text of the statement, and copy it using your browser's copy command.

17. Paste it into another application, if desired.

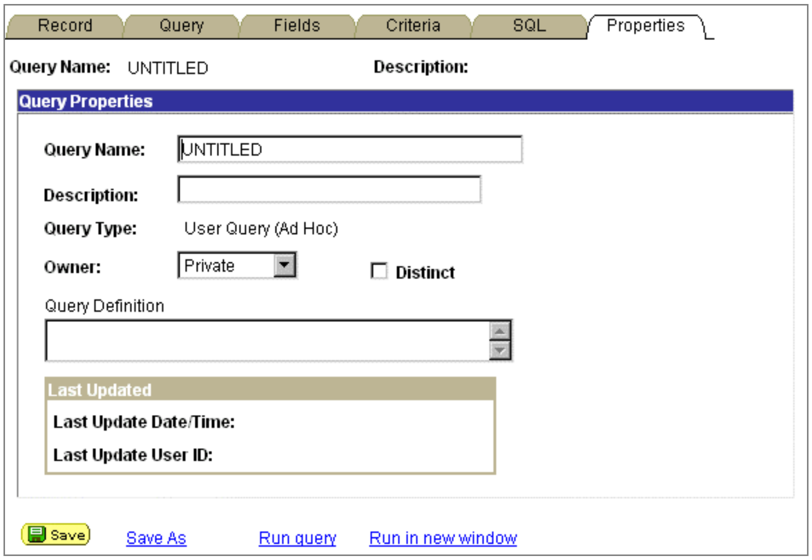
Defining Query Properties

When you save your query, you may want to record information about it so that you can use it again in the future. This information resides on the Properties page. You can change most of the properties of a query at any time. To change a private query to a public query, have your system administrator use Windows-based Query to change the query from Private to Public.

To define query properties:

18. Select the Properties tab.

The Properties page appears.



Query Name: UNTITLED Description:

Query Properties

Query Name: UNTITLED

Description:

Query Type: User Query (Ad Hoc)

Owner: Private ☐ Distinct

Query Definition

Last Updated

Last Update Date/Time:

Last Update User ID:

Query Properties page

19. Enter the name and description of your query.

All queries created in Query Designer are designated **User Query (Ad Hoc)**.

20. Select whether you query is ***Public*** or ***Private*** in the Owner field.

Private means that only the user ID that created the query can open, run, modify, or delete the query.

Public means that any user with access to the records used by the query can run, modify, or delete the query.

21. Select **Distinct** if you want to remove duplicate rows of output from your results.

22. Enter the Query Definition.

23. Query Designer will display the date, time, and user ID associated with any changes to this page in the Last Updated group box.

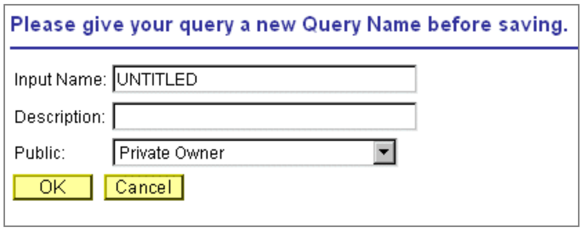
24. Select **Save.**

After you save the query, the Query Name field is grayed out.

For more information on query security, see Setting up Query Security. For more information on the DISTINCT option, see Advanced Query Options**.**

Saving a Query

You can save a query at any time after you have selected one record and at least one field for your query. Queries can be saved from the Properties page or by using the Save buttons on the other Query Designer pages. You must enter some basic information about the query before the system allows you to save it for the first time. If you select the Save button from a page other than the Properties page, you will need to enter the following information:



Please give your query a new Query Name before saving.

Input Name: UNTITLED

Description:

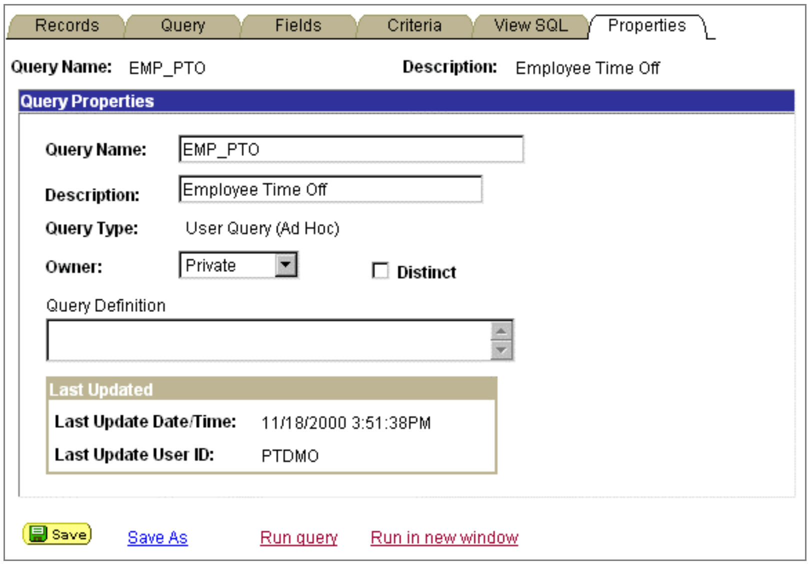
Public: Private Owner

Saving a query

Enter the Input or short name for the query and a description that will help you to identify it later on. (Description is not a required field.) Then, select whether you query is ***Public*** or ***Private*** in the Owner field. Private means that only the user ID that created the query can open, run, modify, or delete the query. Public means that any user with access to the records used by the query can run, modify, or delete the query.

Using **Save As** creates another instance of the query that you can modify and save under a different name. When you select Save As, the Properties page appears allowing you to change the name, description, and owner of the new query. After you make your changes, be sure to select **Save.**

Note. Some queries (ones with unions, subqueries, and prompts) are not supported for edit at this time. The Save As button will not be available for these queries. You can still run and print these queries from Query Designer, but you must use the Windows-based Query to edit them.



Query Name: EMP_PTO Description: Employee Time Off

Query Properties

Query Name: EMP_PTO

Description: Employee Time Off

Query Type: User Query (Ad Hoc)

Owner: Private □ Distinct

Query Definition

Last Updated

Last Update Date/Time: 11/18/2000 3:51:38PM

Last Update User ID: PTDMO

Properties page

Joining Records

Query Designer enables you to create queries that include multiple-table *joins*. Joins retrieve data from more than one table, presenting the data as if it came from one. Query links the tables based on common columns and links the rows on the two tables by common values in the shared columns.

Joins are what make relational databases relational. Using joins, you define relationships among fields when you query the records, not when you create the records. Because PeopleSoft records are highly normalized (they each describe one kind of entity), it's easy to use Query Designer to create joins.

The procedure for joining tables differs depending on how the tables being joined are related to each other. Query Designer recognizes two types of joins: Record Hierarchy and Any Record joins.

Creating a Record Hierarchy Join

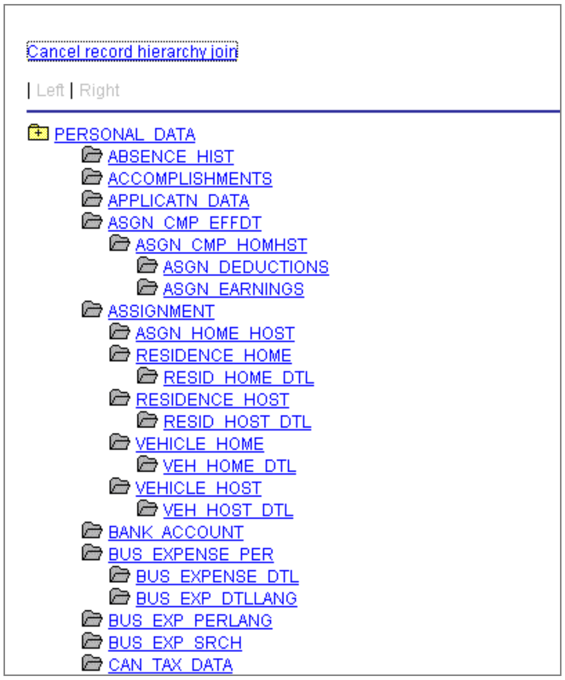
A Record Hierarchy joins a parent table to a child table. (A *child table* is a table that uses all the same key fields as its parent, plus one or more additional keys.) You specify the parent/child relationship using the Parent Record Name option in the Application Designer.

To create a Record Hierarchy Join:

25. Choose the base record for your query and select the appropriate fields and criteria.

26. From the Query page, select **Record Hierarchy.**

You will see all of the records that have a parent/child relationship with your selected record.



Cancel record hierarchy join

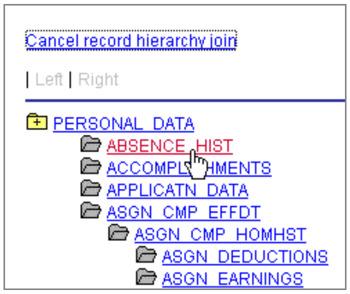
| Left | Right

```
PERSONAL_DATA
    ABSENCE_HIST
    ACCOMPLISHMENTS
    APPLICATN_DATA
    ASGN_CMP_EFFDT
        ASGN_CMP_HOMHST
            ASGN_DEDUCTIONS
            ASGN_EARNINGS
    ASSIGNMENT
        ASGN_HOME_HOST
        RESIDENCE_HOME
            RESID_HOME_DTL
        RESIDENCE_HOST
            RESID_HOST_DTL
        VEHICLE_HOME
            VEH_HOME_DTL
        VEHICLE_HOST
            VEH_HOST_DTL
    BANK_ACCOUNT
    BUS_EXPENSE_PER
        BUS_EXPENSE_DTL
        BUS_EXP_DTLLANG
    BUS_EXP_PERLANG
    BUS_EXP_SRCH
    CAN_TAX_DATA
```

Record tree

27. Select the second record for the join.

In this example, we are selecting ABSENCE_HIST.



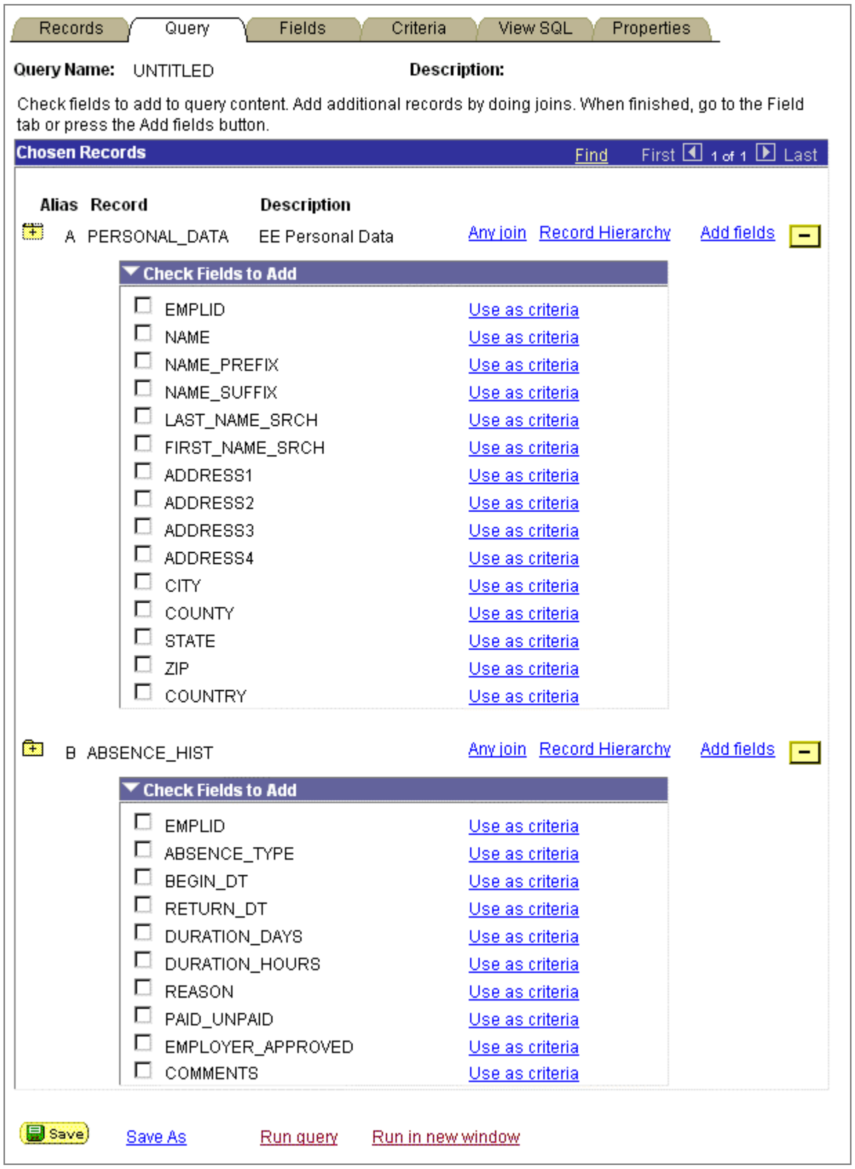
Cancel record hierarchy join

| Left | Right

```
PERSONAL_DATA
    ABSENCE_HIST
    ACCOMPLISHMENTS
    APPLICATN_DATA
    ASGN_CMP_EFFDT
        ASGN_CMP_HOMHST
            ASGN_DEDUCTIONS
            ASGN_EARNINGS
```

Selecting the second record

The join will be reflected on the Query page.



Query Name: UNTITLED **Description:**

Check fields to add to query content. Add additional records by doing joins. When finished, go to the Field tab or press the Add fields button.

Chosen Records Find First 1 of 1 Last

Alias	Record	Description			
A	PERSONAL_DATA	EE Personal Data	Any join	Record Hierarchy	Add fields

Check Fields to Add

□ EMPLID	Use as criteria
□ NAME	Use as criteria
□ NAME_PREFIX	Use as criteria
□ NAME_SUFFIX	Use as criteria
□ LAST_NAME_SRCH	Use as criteria
□ FIRST_NAME_SRCH	Use as criteria
□ ADDRESS1	Use as criteria
□ ADDRESS2	Use as criteria
□ ADDRESS3	Use as criteria
□ ADDRESS4	Use as criteria
□ CITY	Use as criteria
□ COUNTY	Use as criteria
□ STATE	Use as criteria
□ ZIP	Use as criteria
□ COUNTRY	Use as criteria

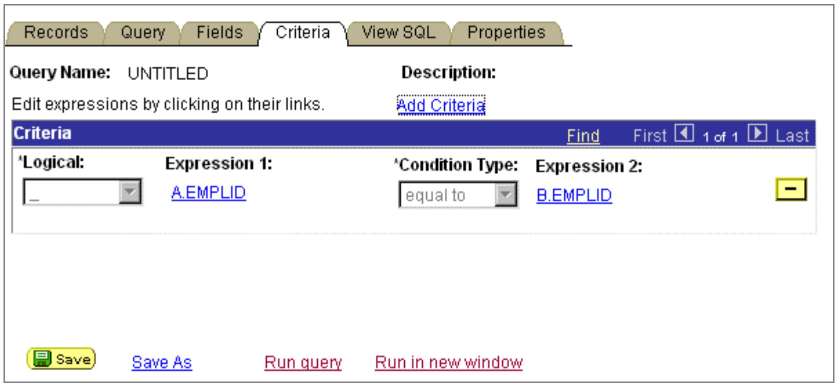
B	ABSENCE_HIST	Any join	Record Hierarchy	Add fields

Check Fields to Add

□ EMPLID	Use as criteria
□ ABSENCE_TYPE	Use as criteria
□ BEGIN_DT	Use as criteria
□ RETURN_DT	Use as criteria
□ DURATION_DAYS	Use as criteria
□ DURATION_HOURS	Use as criteria
□ REASON	Use as criteria
□ PAID_UNPAID	Use as criteria
□ EMPLOYER_APPROVED	Use as criteria
□ COMMENTS	Use as criteria

Query page

The Criteria page reflects the two key fields by which the two tables are joined.



Query Name: UNTITLED

Description:

Edit expressions by clicking on their links.

Criteria

*Logical:	Expression 1:	*Condition Type:	Expression 2:
_	A.EMPLID	equal to	B.EMPLID

Criteria page

Note. We do not recommend that you edit or delete automatically generated join criteria. Although the Delete button is not grayed out, deleting automatically created criteria can cause unpredictable results with your query.

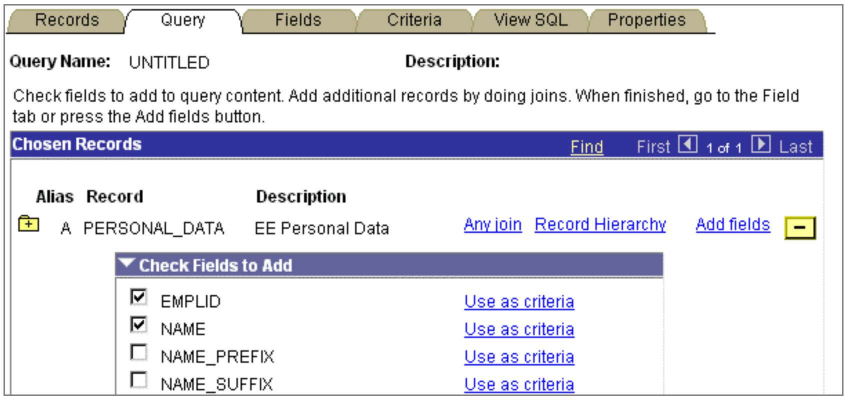
Creating an Any Join

Using Query Designer, you can create a join between two records (Any Join) in one of two ways. You can select your initial base record, define it's output fields and associated criteria, then return to the Record page and select the second record. You can also select and define your base record, then select the Any Join link and choose the second record.

To create an Any Join:

28. Choose the base record for your query and select the appropriate fields and criteria.

29. From the Query page, select **Any join**.



Query Name: UNTITLED

Description:

Check fields to add to query content. Add additional records by doing joins. When finished, go to the Field tab or press the Add fields button.

Chosen Records

Alias	Record	Description
A	PERSONAL_DATA	EE Personal Data

Check Fields to Add

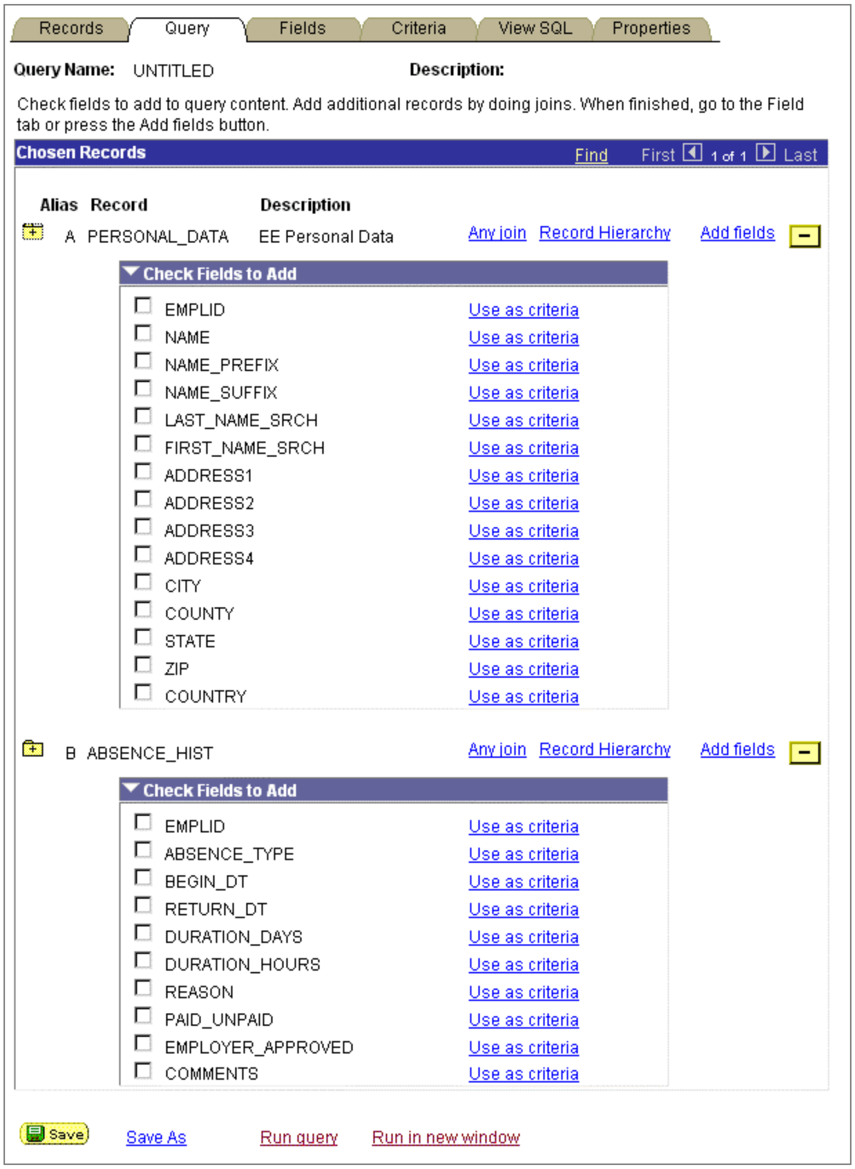
☑	EMPLID	Use as criteria
☑	NAME	Use as criteria
□	NAME_PREFIX	Use as criteria
□	NAME_SUFFIX	Use as criteria

Query page

In this example, we are creating a query joining PERSONAL_DATA with ABSENCE_HIST.

30. Navigate to the **ABSENCE_HIST** record on the Record page and select **Add Record.**

Query Designer will add the record to the query and display both records on the query page.



Query Name: UNTITLED **Description:**

Check fields to add to query content. Add additional records by doing joins. When finished, go to the Field tab or press the Add fields button.

Chosen Records Find First 1 of 1 Last

Alias	Record	Description			
A	PERSONAL_DATA	EE Personal Data	Any join	Record Hierarchy	Add fields

Check Fields to Add

☐ EMPLID Use as criteria
☐ NAME Use as criteria
☐ NAME_PREFIX Use as criteria
☐ NAME_SUFFIX Use as criteria
☐ LAST_NAME_SRCH Use as criteria
☐ FIRST_NAME_SRCH Use as criteria
☐ ADDRESS1 Use as criteria
☐ ADDRESS2 Use as criteria
☐ ADDRESS3 Use as criteria
☐ ADDRESS4 Use as criteria
☐ CITY Use as criteria
☐ COUNTY Use as criteria
☐ STATE Use as criteria
☐ ZIP Use as criteria
☐ COUNTRY Use as criteria

B ABSENCE_HIST Any join Record Hierarchy Add fields

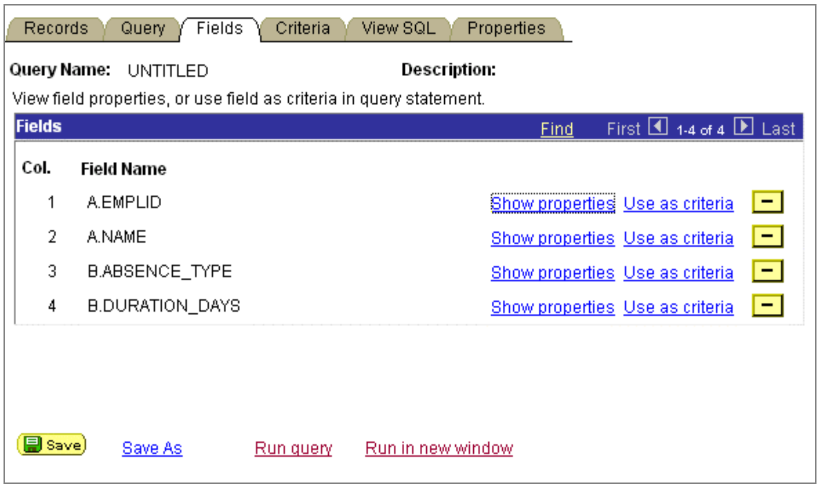
Check Fields to Add

☐ EMPLID Use as criteria
☐ ABSENCE_TYPE Use as criteria
☐ BEGIN_DT Use as criteria
☐ RETURN_DT Use as criteria
☐ DURATION_DAYS Use as criteria
☐ DURATION_HOURS Use as criteria
☐ REASON Use as criteria
☐ PAID_UNPAID Use as criteria
☐ EMPLOYER_APPROVED Use as criteria
☐ COMMENTS Use as criteria

Query page

31. Select the fields that you want to use in your query, and click **Add Fields**.

You'll notice that ABSENCE_TYPE and DURATION_DAYS are prefaced with *B* indicating that these are fields from the second record.



Query Name: UNTITLED Description:

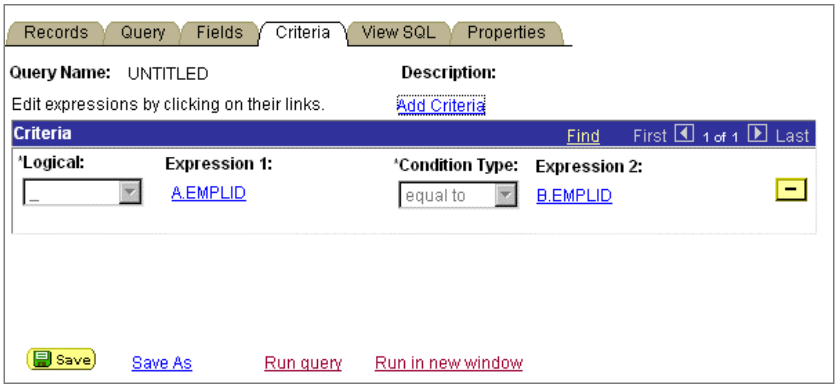
View field properties, or use field as criteria in query statement.

Fields 1-4 of 4

Col.	Field Name		
1	A.EMPLID	Show properties	Use as criteria
2	A.NAME	Show properties	Use as criteria
3	B.ABSENCE_TYPE	Show properties	Use as criteria
4	B.DURATION_DAYS	Show properties	Use as criteria

Fields page

32. Next, go to the Criteria page and select ***equal to*** as the Condition Type.



Query Name: UNTITLED Description:

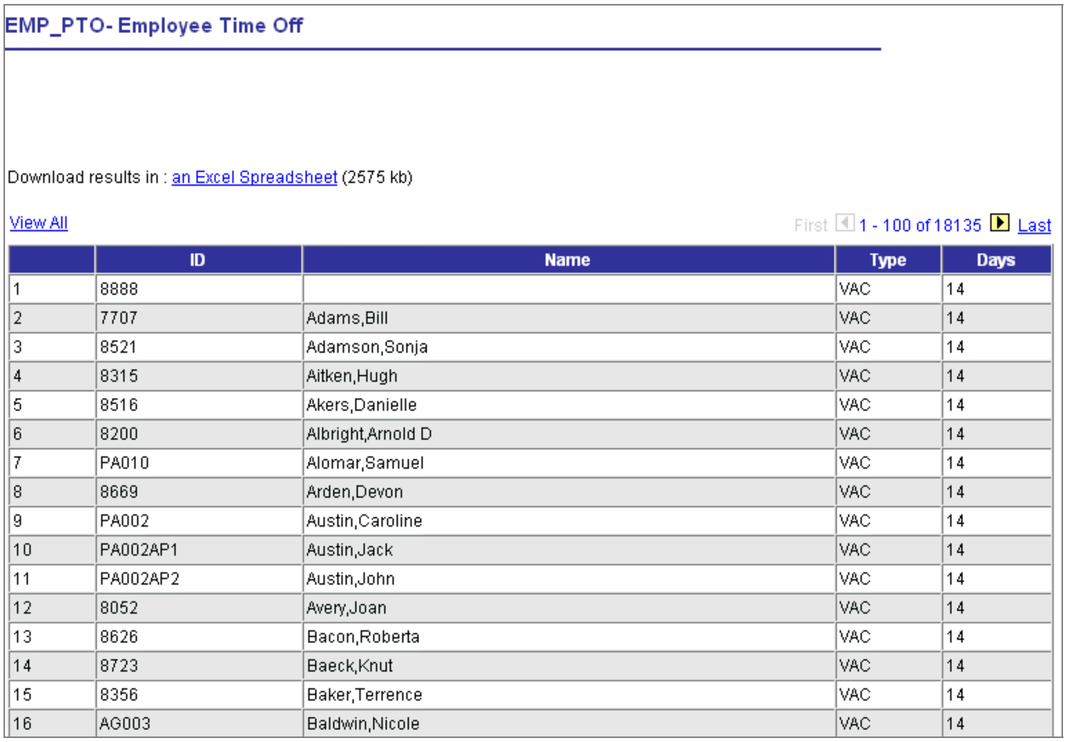
Edit expressions by clicking on their links.

Criteria 1 of 1

*Logical:	Expression 1:	*Condition Type:	Expression 2:
_	A.EMPLID	equal to	B.EMPLID

Criteria page

This is an example of query output from two joined records.



EMP_PTO- Employee Time Off

Download results in : an Excel Spreadsheet (2575 kb)

View All

	ID	Name	Type	Days
1	8888		VAC	14
2	7707	Adams,Bill	VAC	14
3	8521	Adamson,Sonja	VAC	14
4	8315	Aitken,Hugh	VAC	14
5	8516	Akers,Danielle	VAC	14
6	8200	Albright,Arnold D	VAC	14
7	PA010	Alomar,Samuel	VAC	14
8	8669	Arden,Devon	VAC	14
9	PA002	Austin,Caroline	VAC	14
10	PA002AP1	Austin,Jack	VAC	14
11	PA002AP2	Austin,John	VAC	14
12	8052	Avery,Joan	VAC	14
13	8626	Bacon,Roberta	VAC	14
14	8723	Baeck,Knut	VAC	14
15	8356	Baker,Terrence	VAC	14
16	AG003	Baldwin,Nicole	VAC	14

Query Output page

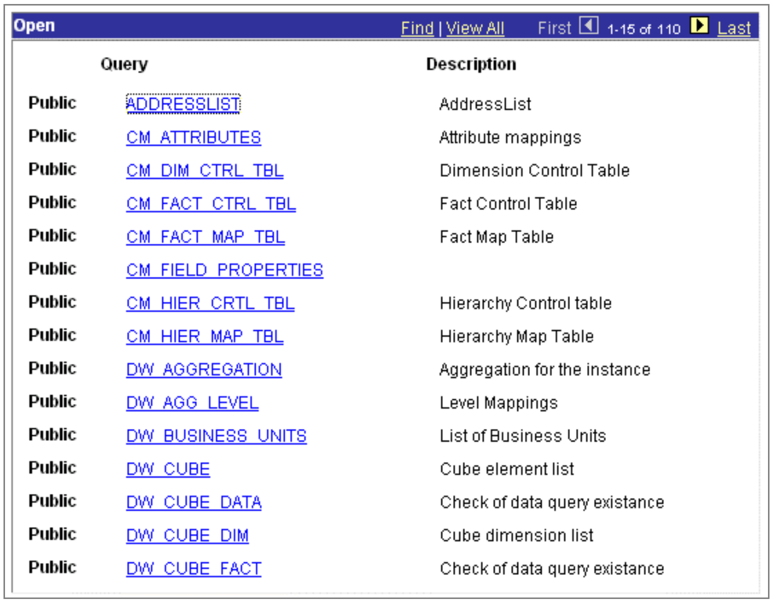
Updating an Existing Query

There may be times when you'd like to update a query to reflect new conditions or copy an existing query and modify the new query for other uses. You can open a query, examine it to see if it's suitable, and then use the Save As function to save it with a new name.

To update an existing query:

33. Select PeopleTools, Query Designer, Use, Update Existing Query.

The Open Query page appears listing all the queries you have security to view.



	Query	Description
Public	ADDRESSLIST	AddressList
Public	CM_ATTRIBUTES	Attribute mappings
Public	CM_DIM_CTRL_TBL	Dimension Control Table
Public	CM_FACT_CTRL_TBL	Fact Control Table
Public	CM_FACT_MAP_TBL	Fact Map Table
Public	CM_FIELD_PROPERTIES	
Public	CM_HIER_CRTL_TBL	Hierarchy Control table
Public	CM_HIER_MAP_TBL	Hierarchy Map Table
Public	DW_AGGREGATION	Aggregation for the instance
Public	DW_AGG_LEVEL	Level Mappings
Public	DW_BUSINESS_UNITS	List of Business Units
Public	DW_CUBE	Cube element list
Public	DW_CUBE_DATA	Check of data query existance
Public	DW_CUBE_DIM	Cube dimension list
Public	DW_CUBE_FACT	Check of data query existance

Open Query page

34. Select the link for the query you want to open.

35. Select **Save As** from the bottom of the Record page.

36. Enter the new information for the query on the Properties page.

37. Click **Save.**

Running a Query

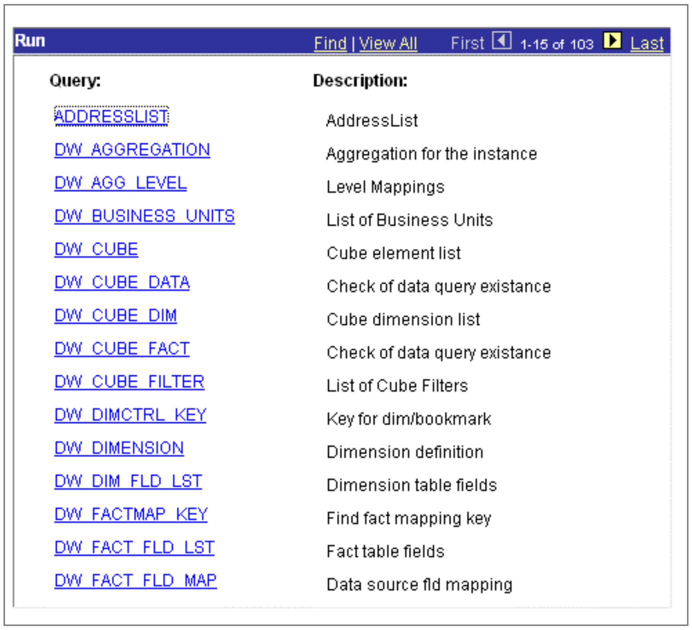
Using Query Designer to run your queries from a browser is as simple as clicking a link in your browser. Using the new functionality of PeopleSoft Internet Architecture, even the most inexperienced users can run queries in an environment that is familiar and easy to use.

Select PeopleTools, Query Designer, Use, Run Query from the menu or select the Run Query link from the pages that you use to create a query. If you run a query from within the Create Query pages, you must remember to save your query after you create it before you can run it successfully.

Query Designer displays only the queries that you have security to run.

To run a predefined query using the PeopleSoft Internet Architecture:

38. Open your browser and select **PeopleTools, Query Designer, Use, Run Query** to display the Run page.



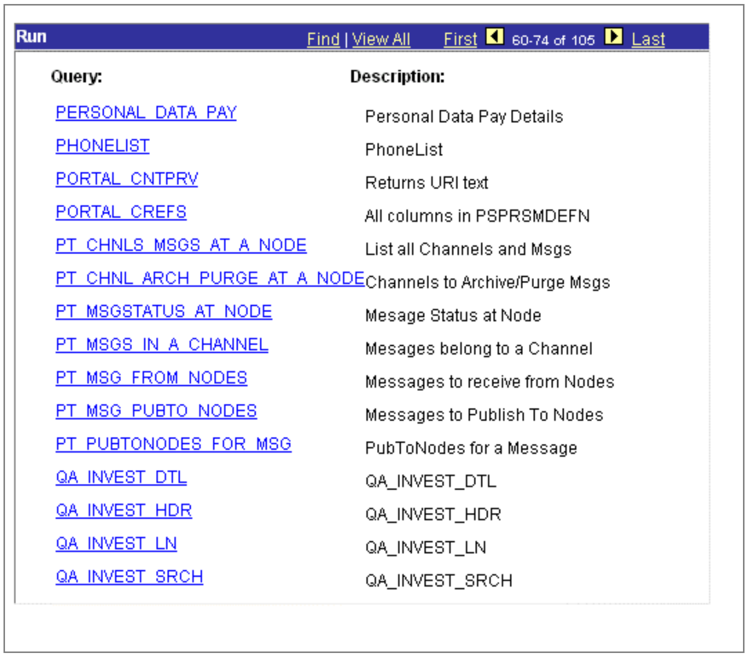
Query:	Description:
ADDRESSLIST	AddressList
DW_AGGREGATION	Aggregation for the instance
DW_AGG_LEVEL	Level Mappings
DW_BUSINESS_UNITS	List of Business Units
DW_CUBE	Cube element list
DW_CUBE_DATA	Check of data query existance
DW_CUBE_DIM	Cube dimension list
DW_CUBE_FACT	Check of data query existance
DW_CUBE_FILTER	List of Cube Filters
DW_DIMCTRL_KEY	Key for dim/bookmark
DW_DIMENSION	Dimension definition
DW_DIM_FLD_LST	Dimension table fields
DW_FACTMAP_KEY	Find fact mapping key
DW_FACT_FLD_LST	Fact table fields
DW_FACT_FLD_MAP	Data source fld mapping

Run page

In many cases, you will have many more queries than can be displayed on the screen. Selecting View All will cause all of your queries to be displayed on the page, and will require you to scroll through a very long list. The easiest way to find a specific record is to select Find from the Navigation bar. The Find link will start your browser's Find utility where you can enter a search string and locate the record that you need.

39. If you know the name of the query you want to run, select Find and type the name of the query.

You can also enter a portion of the query name to narrow your search. In this example, we've searched for queries that begin with ***PERSONAL***.

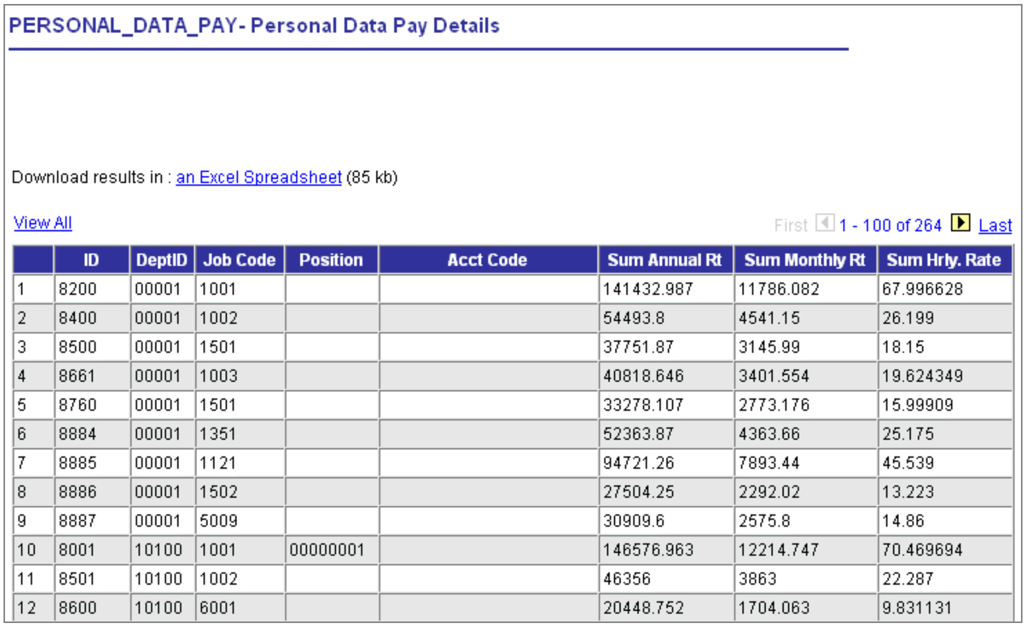


Run Find | View All First ◀ 60-74 of 105 ▶ Last

Query:	Description:
PERSONAL_DATA_PAY	Personal Data Pay Details
PHONELIST	PhoneList
PORTAL_CNTPRV	Returns URI text
PORTAL_CREFS	All columns in PSPRSMDEFN
PT_CHNLS_MSGS_AT_A_NODE	List all Channels and Msgs
PT_CHNL_ARCH_PURGE_AT_A_NODE	Channels to Archive/Purge Msgs
PT_MSGSTATUS_AT_NODE	Mesage Status at Node
PT_MSGS_IN_A_CHANNEL	Mesages belong to a Channel
PT_MSG_FROM_NODES	Messages to receive from Nodes
PT_MSG_PUBTO_NODES	Messages to Publish To Nodes
PT_PUBTONODES_FOR_MSG	PubToNodes for a Message
QA_INVEST_DTL	QA_INVEST_DTL
QA_INVEST_HDR	QA_INVEST_HDR
QA_INVEST_LN	QA_INVEST_LN
QA_INVEST_SRCH	QA_INVEST_SRCH

Run page

40. Click the query name from the list of queries. In this example, we'll run the **PERSONAL_DATA_PAY** query.



PERSONAL_DATA_PAY- Personal Data Pay Details

Download results in : an Excel Spreadsheet (85 kb)

View All First ◀ 1 - 100 of 264 ▶ Last

	ID	DeptID	Job Code	Position	Acct Code	Sum Annual Rt	Sum Monthly Rt	Sum Hrly. Rate
1	8200	00001	1001			141432.987	11786.082	67.996628
2	8400	00001	1002			54493.8	4541.15	26.199
3	8500	00001	1501			37751.87	3145.99	18.15
4	8661	00001	1003			40818.646	3401.554	19.624349
5	8760	00001	1501			33278.107	2773.176	15.99909
6	8884	00001	1351			52363.87	4363.66	25.175
7	8885	00001	1121			94721.26	7893.44	45.539
8	8886	00001	1502			27504.25	2292.02	13.223
9	8887	00001	5009			30909.6	2575.8	14.86
10	8001	10100	1001	00000001		146576.963	12214.747	70.469694
11	8501	10100	1002			46356	3863	22.287
12	8600	10100	6001			20448.752	1704.063	9.831131

PERSONAL_DATA_PAY query page

For more information on options for running your queries, see Creating and Running Simple Queries.

Downloading the Query as an Excel Spreadsheet

You have the option of downloading your query as an Excel spreadsheet. Click the text ***an Excel Spreadsheet*** to open the file in your browser in Excel. If you are using Excel 2000, the Excel spreadsheet will appear in your browser. To save it, select **File, Save As.** If you are using Office 97, the **File download** dialog box appears. You have the choice of opening the file in your browser or saving it to disk.

Troubleshooting the Download in Excel Format Option

If your Query does not download to Excel correctly, there are several things you can check. Depending on your version of Excel and your environment or configuration, you may have to use one of the following workarounds.

Excel 97

If Excel does not launch when you choose to view the spreadsheet in your browser, launch Excel separately and leave it running in the background. Try the download again and your query will appear in Excel format in your browser. You do not need to launch Excel to save the query to your hard disk as an Excel spreadsheet.

If you don’t get the prompt to allow you to select a download location for the file, follow these steps:

1. Double-click My Computer.
2. On the View menu, click Options.
3. On the File Types tab, click the appropriate file type in the Registered File Types box. In this case, select Microsoft Excel Worksheet.
4. Click Edit.
5. Click the Confirm Open After Download checkbox.
6. Click OK and then click OK again.

In some environments, Seagate Crystal Report Wizard must be disabled to download a query in Excel format. To do this, select the **Tools, Add-Ins** in from the Excel toolbar. Deselect the checkbox for Seagate Crystal Report Wizard. After you click **OK,** you can reselect the Report Wizard option without affecting the query download option in the future.

Printing a Query

To print a query, you must first have run the query and have it displayed in your browser. Select the Print icon from your browser or select File, Print from you browser menu. The query will be printed on your default printer.

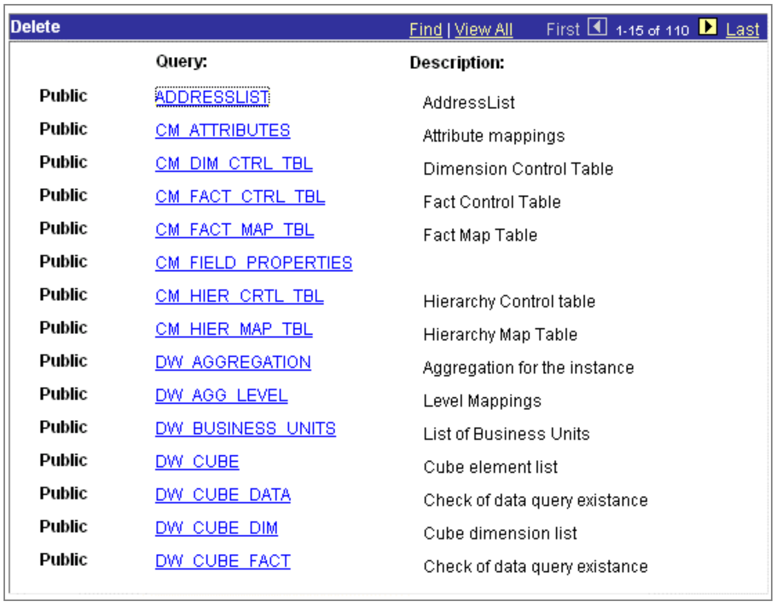
If you choose to download the query in Excel, you can print the query using Excel’s print function.

Deleting a Query

You can delete any public query that you have access to as well as any private query that you have created.

To delete a query:

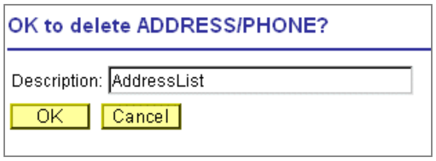
41. Select PeopleTools, Query Designer, Use, Delete Query.



	Query:	Description:
Public	ADDRESSLIST	AddressList
Public	CM_ATTRIBUTES	Attribute mappings
Public	CM_DIM_CTRL_TBL	Dimension Control Table
Public	CM_FACT_CTRL_TBL	Fact Control Table
Public	CM_FACT_MAP_TBL	Fact Map Table
Public	CM_FIELD_PROPERTIES	
Public	CM_HIER_CRTL_TBL	Hierarchy Control table
Public	CM_HIER_MAP_TBL	Hierarchy Map Table
Public	DW_AGGREGATION	Aggregation for the instance
Public	DW_AGG_LEVEL	Level Mappings
Public	DW_BUSINESS_UNITS	List of Business Units
Public	DW_CUBE	Cube element list
Public	DW_CUBE_DATA	Check of data query existance
Public	DW_CUBE_DIM	Cube dimension list
Public	DW_CUBE_FACT	Check of data query existance

Delete page

42. Select the query that you want to delete.


OK to delete ADDRESS/PHONE?

Description: AddressList

OK to delete page

43. Click **OK**.

You do not have to enter anything in the Description field.

Index