Query and Advanced Query Manual
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subqueries</td>
<td>77</td>
</tr>
<tr>
<td>Unions</td>
<td>92</td>
</tr>
<tr>
<td>Managing Queries</td>
<td>100</td>
</tr>
<tr>
<td>Schedule a Query</td>
<td>100</td>
</tr>
<tr>
<td>Email a Query</td>
<td>105</td>
</tr>
<tr>
<td>Adding Queries to My Favorite Queries</td>
<td>106</td>
</tr>
<tr>
<td>Copying a Query to Another User’s List of Queries</td>
<td>107</td>
</tr>
<tr>
<td>Moving a Query to an Organization Folder</td>
<td>109</td>
</tr>
<tr>
<td>Renaming a Private Query</td>
<td>111</td>
</tr>
</tbody>
</table>
Queries can be as simple or as complex as necessary, and they can be one-time ad-hoc queries or queries used repeatedly. Running a query asks the system to search the database for an answer. The database is comprised of records (tables), fields (columns), key fields, and field data (rows). The results generated in a query will depend on the records (tables) selected.

This manual covers the basic concepts of viewing public queries as well as modifying or creating new queries. Users should have a basic understanding of how to use PeopleSoft applications. We recommend completing PantherSoft Fundamentals as well as having a full understanding of data and transactions at FIU. To register for training, check the FIU Develop website at https://develop.fiu.edu/.

Types of Queries

PeopleSoft Query provides the following types of queries:

- **User queries**: Create and run queries to retrieve data from the database directly from Windows-based Query Designer, or the web-based Query Manager/Query Viewer applications. These are the primary queries open to the entire user community.

- **Reporting queries**: Reporting queries are the same as user queries, except that they are designed to be used by another reporting tool. Reporting queries can be used as data sources for reports, PS/nVision, or Query Manager.

- **Process queries**: These are queries that are intended to run periodically by batch processes, using PeopleSoft Application Engine and the Query API (application programming interface).
Query Terminology

**Criteria:** Refines a query by specifying conditions that the retrieved data must meet. This helps the query run the same way each time.

**Prompts:** Extracts specific information without changing the results of the query. A field that requests a required value to run the query.

**Record Definitions:** Design specifications that determine the structure of a PeopleSoft application data table and online processing. In the PeopleSoft database, tables are represented as record definitions. In PeopleSoft Query, tables are also called records.

**Records/Tables:** Made up of columns (Fields) and rows (Data). Columns determine how the data will be stored. Rows represent the actual data stored in the database. Oracle’s PeopleSoft refers to Tables as Records.

**SetID:** Code that is used to group and share configuration data across the application. At FIU, most of the control records (i.e., Accounts, Department ID, Activity #) are under SetID ‘FIU01’ and allows the entire university to use those values.

**Fields:** Column headers within a query that store single pieces of information for each row. (TRAVEL_AUTH_ID is a field.)

**Rows:** Data in the field that contains all the information for a unique combination of key values on the table i.e. Travel Authorization number “0000201458” is the data in the field.

**Key Fields:** Fields within a record that hold unique data which identifies that record from all the other records in the file or database e.g., Account number, Transaction IDs, and Name are typical key fields. As an identifier, each key value must be unique in each record. Every record has a key field.

Roles and Security

The following users can access the Query Tools functionality:

**Query Super Users:** Users who develop queries within PeopleSoft for the user community. They can save Public queries that can be accessed by any user at FIU. They may also have job specific duties that require them to use, develop, and write queries on a regular basis.

**Query Managers:** User with the knowledge and skills received from the training and certification quiz to create and modify queries. They can only save their new or modified queries as Private queries.

**Query Viewers:** Users who can only run public queries but cannot develop or modify queries.

Users have access to data based on security settings within each PantherSoft application. See below for the process to request Query Viewer or Query Manager security role within each application. Query Super Users are provided their role based on job specification.

**Note:** Access is primarily granted within the Reporting Database and not within Production. However, users can request access to view, create, and modify queries in the Production environment.
Query Access within PS Human Resources

Users requesting either **Query Viewer** or **Query Manager** within the Human Resources application must have their HR supervisor put in the request with either the Human Resources Help Desk 305-348-2181 or Information Technology 305-348-2284. Users must attend the related training and complete a certification quiz for final approval of the security role. Access is also only granted within the Reporting database.

Query Access within PS Financials

To gain **Query Viewer** access within PantherSoft Financials to view or run public queries, users must obtain **Access Role 001 Inquiry and Reporting**. By default, this role provides access only within the Reporting Database.

Users should login to PantherSoft Financials and navigate with the NavBar to **Main Menu > Access Request System > Access Request**. Click on the **Add a New Value** tab and **Add** to create a new request. Choose 001 Inquiry and Reporting role and include a business reason under the **Description** field as to why the user is requesting the role. All roles route to the individual's HR Supervisor for approval.

To obtain the security role for **Query Manager**, users must attend the Query 101 training listed in **FIU Develop**, score 80% or higher on the certification quiz, and send an email to **controller@fiu.edu** copying their HR supervisor. The HR supervisor must reply approving the request to receive the role. Access is only granted within the Reporting database.

**Note:** When access to Production is needed, users can send an email to **controller@fiu.edu** copying their HR supervisor. The HR supervisor must reply approving the request.

Query Access within PS Campus Solutions

Users requesting either **Query Viewer** or **Query Manager** access within the Campus Solutions application must login to the Campus Solutions application and request access through the **End User Security Request Tile**. There are two roles within Job Role Campus Community to Run Queries – RPT and Create Queries – RPT. Either must be approved by an HR supervisor and are only granted within the Reporting database. Users must attend the training and pass the certification quiz with 80% or higher for final approval of the security role. For additional information or support please contact **sfsuppor@fiu.edu**.

Reporting Database and VPN Access

The Reporting Database is a PantherSoft Financials environment used exclusively to run and process data. Data and information on reports is 24hrs behind the usual production environment. The Reporting Database can be found through the **Reporting Database** tile as well as a stand-alone URL.

1. Within PantherSoft Financials and from the **Employee Self Service** homepage, click **Reporting Database** Tile.
This will navigate to the Reporting Database environment, proceed with Query Tools tile.

Alternatively, users can navigate to the Reporting Database directly through the Databases website. Visit [http://go.fiu.edu/fiudatabases](http://go.fiu.edu/fiudatabases)
**Query Viewer**

Query Viewer allows users read-only access to pre-existing queries created by other users in the PantherSoft applications. Query results can be previewed in a web browser or exported to an Excel spreadsheet. Query Viewer is available both within the Reporting database and Production environments. See the [Roles and Security](#) section for information regarding gaining access to Query Viewer within a specified environment (Human Resources, Financials, Campus Solutions).

**Running or Viewing a Query**

To begin running or viewing a query using Query Viewer, login to the Reporting Database. It can be accessed via [http://go.fiu.edu/fludatabases](http://go.fiu.edu/fludatabases) or directly through a production application. Note that the Reporting Database requires login to the VPN.

From PantherSoft Financials Production environment, click the Reporting Database Tile.

1. Within the PSFSRPT environment, click the Query Tools tile.
2. Click on **Query Viewer** and search for the name of a query. Click on **Advanced Search** to filter results.

Advanced Search view:

Use the **Search By** drop-down to find queries by other parameters.
Query Name: The name given to a query. There are no spaces, uses capital letters. The naming convention at FIU should have queries begin with FIU and underscore and the name of the department. EX: FIU_FSSS_ER_QUEUE

Description: A brief description given to a query

Uses Record Name: Searches for a query that includes an associated Record name.

Uses Field Name: Searches for a query that includes the name of a specific field.

Access Group Name: Searches for a query associated to an Access Group

Folder Name: Searches for the name of the folder the query is stored in.

Owner (Public or Private): Searches for either private or public queries. Queries saved as private are not searchable by users who do not have public query access.

Type: Search for User, Archive, Process, or Role queries.

3. After entering search parameters, click Search. Results will appear below.

4. Choose the Query Name to run or view associated data. Decide which Run To option, either HTML or Excel. Run to HTML displays the query results on screen. Run to Excel displays the result in an excel spreadsheet. Run to XML is not used at FIU.
Many queries involve additional criteria called prompts. These are fields or specific information needed to further refine the results of the query. Prompts may be required to complete running the query.

Queries that are **Run to HTML** may display this way:

![Image of Run to HTML example]

Queries that are **Run to Excel** may display this way. If a prompt is present, enter the information required then click View Results:

![Image of Run to Excel example]
Query Manager

Users with **Query Manager** security role can create or edit queries without needing to use Standard Query Language (SQL) programming statements. When creating a query, users must navigate to **Query Manager** to edit a query or to access the link for creating a new query.

Create a New Query/Modify Existing Query

Users should login to the Reporting Database first. It can be accessed via [http://go.fiu.edu/fiudatabases](http://go.fiu.edu/fiudatabases) or directly through a production application. Note that the Reporting Database requires login to the VPN.

From PantherSoft Financials Production environment, click the **Reporting Database** Tile.

1. Within the PSFSRPT environment, click the **Query Tools** tile.
2. Query Manager is the default navigation. Click **Create New Query** link.

![Query Manager](image1)

3. A new unsaved query will appear with tabs across the top.

![Query Tools](image2)

**Records Tab** – Adds or joins records to the query.

**Query Tab** – Shows which records are already selected for the query.

**Expressions** – Displays the Expressions if any, that have been added to the query and can create expressions.

**Prompts** – Allows user to add or edit prompts.

**Fields** – Shows the fields currently selected and provides options to create criteria and further work with the fields.

**Criteria** – Shows the specific parameter set that refines how the query will run each time.

**Having** – Create, view, edit criteria based on Aggregated fields.

**View SQL** – Displays the SQL language of the query.

**Run** - Runs the query according to set criteria, records, and fields selected.

Along the bottom of the **Query Tools** pages are useful links: **Save As**, **New Query**, **Preferences**, **Properties**, **Publish as Feed**, **Publish as Pivot Grid**, and **New Union**.
Save As: Provides the option to save the query.

New Query: Replaces the current open window with the creating a new query options.

Preferences: Provides Query Preferences such as Name Style and whether to enable Auto Join

Properties: Provides Query Properties including save, name/rename options, last updated date/time, and last update user id as well as Distinct and Security Join Optimizer options.

Publish as Feed/Publish as Pivot Grid options are not available
Searching and Selecting Records

The first step in creating a new query is to choose a Record. Selecting the proper record to add to a query involves familiarity with the data and intuition about the information that users wish to search. To aid with selecting records, there is a list of frequently used Records.

1. After clicking on the Create New Query link, users must search for and select a record. Use the Search By drop-down and/or begins with field to enter the name of a Record. Click Search. The Advanced Search link shows more search options.

The same search options, like partial entries and using the wildcard learned in Fundamentals, work within Query as well. Users can also perform a partial search by entering part of a record name or description in the Search By field.

![Advanced Search options](image)

Advanced Search options:
2. Search results will appear. Click **Add Record** to choose the desired record. Click **Show Fields** to display a list of all the fields within the record available for selection.

**Show Fields:** Displays a preview of the record’s fields. Users can click **Return** to go back to the **Records** tab page.
Add Record: Adds/Selects the record; it will appear under the Query tab.
Saving a Query

When editing or creating queries, any changes or edits do not save automatically. In addition, saving a query is only available within Query Manager. Users must scroll to the bottom of the screen and access the Save As link.

1. To save a Query via Query Manager select Save As.

2. Enter a short name for the query in the Query field. Query names must follow the standard naming convention: FIU_DEPARTMENT_QUERY_DESCRIPTION; cannot have spaces or symbols.
3. Enter information that will help describe and identify what the query is about in the **Description** field.

4. Select a **Query Type** from the drop-down menu. Standard queries are designated as *User* queries. The *Archive, Process, or Role* options apply to Workflow queries. Workflow is not currently implemented by FIU. Select the default option: *User*. 
5. The Owner field will default to: *Private*. *Private* indicates that only the user ID that created the query can open, run, modify, or delete the query while *Public* means that any user with access to the records used by the query can run, modify, or delete the query. The ability to save a query as *Public* will vary depending on a user's level of security.

![Image of query setup interface showing the Owner field set to private]

6. Enter the Query Definition which can be a long description, i.e. “This query shows all the Hazardous Materials Ship To locations.” Then, click **OK**.

![Image of query setup interface showing the Query Definition field]

---

Version 6.0.0  February 2024
The **Query** Tab includes options related to records already chosen for a query. Users can click the folder icon next to a record to expand or collapse and show fields.

Check boxes to add fields, uncheck boxes to remove fields from the query. Once a record folder is expanded, there are also buttons for **Check All** and **Uncheck All**. It may be helpful to check all fields to view the data contained in those fields.

The key icon next to a field indicates this is a key field of the record.
The **Use as Criteria** icon is located to the right of the fields. By clicking on this icon, criteria can be created for that field.

Some fields will have blue links next to them. These provide the ability to create joins to additional records, called **Related Record Join**.
For each joined record, there are two options: to Delete a record with the minus sign and to join a child table to its parent table via the Hierarchy Join link.

Click the Sort fields alphabetically button once to list fields in alphabetical order. Click the button again to return to original sort. Fields displayed top to bottom on this screen will run left to right within the query if chosen.
Field Tab

The Field Tab provides options related to the fields already chosen within the query. Users can change field properties using Reorder/Sort and Edit to change the labels of the field.

Editing Field Properties

Users can alter the way the field name appears within the query by clicking on Edit within the field line. This will edit the field heading in the query, but it does not change the field name on the record.

Options appear under the Heading box. Choose one of the following options to edit the name of a field, then click OK.
No Heading: The column in the query will not have a heading; the field name will not show.

Text: The column heading in the query is any customized text that is typed within the **Heading Text** field. The field will be customized to what is typed.

RFT Short: The column heading is the short name from the record definition; what appears to the left of the dash. Example: Descr

RFT Long: The column heading is the long name from the record definition; what appears to the right of the dash. Example: Description
Users can change the order the columns (fields) display within the query as well as how the data is sorted within those columns. Click the **Reorder/Sort** button for those options.

The **Edit Field Ordering** box will appear. Fields appear in the order they were chosen from top to bottom. They will display when the query runs from left to right.

To change the order of the columns (fields), use the **New Column** boxes next to the column or field to be changed. Enter a number next to the field that is to be first, second, third. In the example below the 4th column will now be Column 1, followed by Column 6th. Columns left blank are automatically assigned a consecutive number. The same number cannot be used on multiple fields. Click **OK** if that is all that is being changed.

To change any column or field to sort the data in **descending** order, simply check the box next to the field(s) within the **Descending** column. Then, click **OK** if nothing else is to be changed.
The **New Order By column** is used to sort the associated field data selected in ascending order. Users will need to enter 1 in the **New Order By** column next to the field that needs to be sorted first. If additional fields need to be sorted, add sequential numbers to indicate the order you want to sort by. If a field is left blank or a 0 is entered, no sorting will take place. In the example below, the query results (rows) will show A. DEPTID field sorted from smaller to larger value. Click **OK** if there are no other changes.
Criteria Tab

The Criteria Tab provides the options to add specific conditions to a query which will refine and return specific rows of data. Without criteria, users can retrieve every row of data within a table or record by simply selecting the fields and running a query.

Select the Criteria Tab and click the Add Criteria button.

The Edit Criteria Properties page appears.
Criteria Properties

Expression 1 Type: Defines the criteria being added. Can be a Field or Expression.

Expression 1: The Field or Expression the Criteria is being compared to.

Expression 2 Type: Defines where the Comparison Value the Criteria is being compared against comes from; usually, Field, Expression, Prompt, Constant or Subquery though the options can change based on the Condition Type selected.

Expression 2: The Value the Criteria will be compared against.

<table>
<thead>
<tr>
<th>Value Type</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>The value in the selected field is compared to the value in another field, usually a field in another record component.</td>
</tr>
<tr>
<td>Expression</td>
<td>The value in the selected field is compared to an expression you enter, which PeopleSoft Query evaluates once for each row before comparing the result to the value in the selected field.</td>
</tr>
<tr>
<td>Constant</td>
<td>The value in the selected field is compared to a single fixed value.</td>
</tr>
<tr>
<td>Subquery</td>
<td>The value in the selected field is compared to the data returned by a subquery.</td>
</tr>
<tr>
<td>Prompt</td>
<td>The value in the selected field is compared to a value that you enter when running the query.</td>
</tr>
</tbody>
</table>

By default, the Field option is selected in the Choose Expression 1 Type box. Select a specific field from a table/record that has been chosen within the query by clicking on the search glass icon under Expression 1, Choose Record and Field.
The **Select a field** window appears providing all the possible records and fields that are available within the query. Users may need to click on the Show Fields button to view available fields if the query has more than one record. Click the desired field to select it.

The chosen field appears under **Record Alias.Fieldname**
# Condition Types

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

<table>
<thead>
<tr>
<th>Condition Types</th>
<th>When It Returns a Row</th>
</tr>
</thead>
<tbody>
<tr>
<td>between</td>
<td>The value in the selected record field falls between two comparison values. The range is inclusive.</td>
</tr>
<tr>
<td>equal to</td>
<td>The value in the selected record field exactly matches the comparison value.</td>
</tr>
<tr>
<td>exists</td>
<td>This condition 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 returns any data, PeopleSoft Query returns the corresponding row.</td>
</tr>
<tr>
<td>greater than</td>
<td>The value in the record field is greater than the comparison value.</td>
</tr>
<tr>
<td>in list (see subquery section)</td>
<td>The value(s) in the selected record field matches the comparison value(s) in a list. The query will show the results of that match.</td>
</tr>
<tr>
<td>in tree</td>
<td>The value in the selected record field appears as a node in a tree created with PeopleSoft Tree Manager. The comparison value for this condition is a tree or branch of a tree PeopleSoft Query is to search.</td>
</tr>
<tr>
<td>is null</td>
<td>The selected record field doesn’t have a value in it. Don’t specify a comparison value for this condition. Key fields, required fields, character fields, and numeric fields do not allow null values.</td>
</tr>
<tr>
<td>less than</td>
<td>The value in the record field is less than the comparison value.</td>
</tr>
<tr>
<td>like</td>
<td>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 PeopleSoft Query recognizes are % and _. % matches any string of zero or more characters, i.e., C% matches any string starting with C, including C alone. _ matches any single character, i.e., ones matches any five-character string ending with ones, such as Jones or Cones.</td>
</tr>
</tbody>
</table>

**Condition Type: Equal To Example**

In the screenshot below, using Field DEPID-Department from record A, **Condition Type**: equal to, we can select Field, Expression, Constant, Prompt, or Subquery to add a value to compare with Expression 1.
**Condition Type: In Tree Example**

**Condition Type** “in tree”: Enables a hierarchical tree to be chosen, and within the tree, the nodes containing specific values. *In Tree* is used to return the records that match or appear in the specified tree.

Some of the most commonly used trees at FIU are:
- ACCOUNTROLLUP
- ACTIVITY_HIERARCHY
- DEPT_ROLLUP
- STUFINANCIALS – Student Financials in Campus Solutions

*In Tree* can be used to answer questions such as: What are all the Travel Authorizations within the College of Medicine? Criteria could be added using “in tree” instead of having to list every department number.

Follow the steps to use **Condition Type** “in tree”:

1. After clicking **Add New Criteria**, choose **Expression 1 Type: Field**, **Choose Record and Field**: A.DEPTID-Department, select the “in tree” **Condition Type**.
2. Choose Tree Option under **Choose Expression 2 Type** and click the **New Node List** link under **Select Tree Node List** to display the Select Tree page.
3. Click on Search or type in the Tree name. **Only trees to which a user has been granted access will be listed.** Click the name of the desired tree.
4. Click the **Add to Node Selection List** icon to select the Node. Folders may be expandable as well as users can use the expand hierarchy icon if displayed.

Users can remove nodes from the list by clicking the **Remove from List** icon. To display the selected tree branch, click the **Find** icon.
5. Scroll to the bottom of the page and click **OK**. The selected tree SetID, tree name, effective date, and selected nodes appear in the Select Tree dialog box.

### Understanding Effective Dates

Effective dates keep history, current, and future information in tables. The system categorizes effective-dated rows into three basic types:

- **Future**
  Data rows that have effective dates greater than the system date—usually today’s date. There can be more than one.

- **Current**
  The data row with the greatest effective date less than or equal to today’s (system) date. Only one row is the current row.

- **History**
  Data rows that have effective dates less than the effective date of the current data row. There can be more than one.

When adding data with effective dates, the query defaults to **currently effective** rows of data—the row where the effective date is less than or equal to today’s date. The query may not need to see the historic rows, which are no longer accurate, or future-dated rows, which aren’t in effect yet.

![Select Tree Node List](image)

An effective date criteria has been automatically added for this effective dated record. (139,60)

**To specify effective date criteria:**

1. When you choose the record that has EFFDT as a key field, Query Manager automatically creates default criteria and adds that criteria to the Criteria page.
2. If you choose one of the comparison options, choose to compare each row’s effective date against today’s date or a date other than today.

- Select **Current Date** to compare each row’s effective date against today’s date.
- Select **Constant** to display the Constant box so that you can enter a date. Select this option when you want to see the rows that were effective as of a past date or that will be effective on a future date.
- Select **Expression** to display the Edit Expression box so that you can enter a SQL expression that evaluates the date entered. Select 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 box.
- Select **Field** to display the Select Field box so that you can select the record field that holds the date to which you want to compare effective dates. Select this option when you want to see the rows that were effective at the same time as another 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.
- Select **First Effective Date** to return the row with the oldest effective date, usually the first row entered for an item.
- Select **Last Effective Date** to return the row with the latest effective date, even if that date is still in the future.
Prompts Tab

A run-time prompt allows values to be entered for a specific field at the time the query is run. This acts as a criterion. The query will display only those rows of information that match the value entered in the prompt.

1. Prompts can be created from the Add Criteria icon associated with a field on the Field or Query Tab. Another way to create a prompt is from the Criteria Tab. Click the Add Criteria button and choose the field with which the prompt will be associated.
2. From the **Choose Expression 2 Type** box, select **Prompt**.

3. In the **Define Prompt** box, click the **New Prompt** link.

The **Edit Prompt Properties** window appears. PeopleSoft Query looks to the record definition for information about the field chosen and completes the rest of the dialog box based on its properties. There are basic properties about how the prompted field will display, which can be changed. This includes the **Heading Type**, **Type**, **Heading Text**, **Format**.

**Heading Type**: RFT Short, RFT Long, or text options are available.
**Type:** This indicates what kind of field was chosen.

**Heading Text:** Where the name of the prompted field can be changed.

**Unique Prompt Name** A default value generated by Query Manager for globalization.

By default, prompts use the **Edit Type Prompt Table**. The system will then use a prompt table to choose values based on the field. Under **Prompt Table**, the name of a table will display. If the **Edit Type** is **Translate Table**, the value in the list box determines the values used. The **Edit Type No Table Edit** will allow the user running the query to enter manually any value.

Use the drop-down **Edit Type** and choose **No Table Edit**. Click the search glass icon under **Prompt Table** and select or confirm **No Value**.
Click **OK** to exit and then **OK** again to confirm the choices.

Prompts are viewable as criteria under the **Criteria** Tab indicated by the symbol “:1”. The number depends on the order of the prompts.

Version 6.0.0  February 2024
Prompts are also viewable under the **Prompt** Tab. They can be fully edited and deleted there.
Run Tab

To process the query and view the results, click the Run Tab. Results will display below the tabs. There are also options to View All, Rerun Query, and Download to Excel. FIU does not utilize Download to XML.

View All: Click this link to view all rows and use scroll bar to navigate.

Rerun Query: Click this link to rerun your query in the preview pane. If any changes were made to the query since the last preview, the query must be rerun to see the effect of the changes.

Download to Excel: Click this link to download the query to Excel.

Validating Query Results

A query is defined as a request for information from a database. The queries and query types mentioned in this manual are ‘select’ queries; select queries are data retrieval queries. When data is retrieved and results are displayed, the results should be validated.

While there is no defined step-by-step method of validation, there are a few things that a user can do to determine if the results are accurate.

Some helpful validation tips:

- **Some queries should not have results but there are results**: this may mean that a process may need to be run for the query not to have any results, i.e., running Budget Checking, Voucher Posting, Payment Posting, Combo Build, Journal Edit.
• **Some queries result in an error:** A correction is needed to clear the results of the query. After correcting the error, re-run the query, and it should yield different or no results, i.e., correcting a Journal Entry that is in error status and changing the status to ‘Valid’.

• Reconcile data in a query with standard reports. User nVision reports to confirm data, i.e., AP-Voucher Reports.

• If planning to combine two tables, run each table separately, for one department or an account, then filter in Excel or use a v-lookup function to compare the two. Run the query and then estimate how many rows to expect, and make sure that is what results after combining the tables.

• Use one chartfield and follow it through the queries. When joining new tables, make sure all the data is there and nothing is missing, i.e., same number of rows. Check which lines are missing and validate that the query is correct to exclude those lines.

• Validate some transactions from query results comparing them with what is in PeopleSoft menus or view pages, making sure the fields match, for example that the PO line is open.

• Use the Show Fields option.

• Check all the fields and see what the output is by running the query.
  - Which fields display data?
  - What data is displayed?

• Look at the Criteria, Records, and SQL language of another query that might do the same thing but with different documents.
Working with Multiple Records and Data

In many cases, the desired output data comes from more than one table or record of data. The tables must be linked together to retrieve the correct output or at least added/joined to the query.

PeopleSoft Query allows queries created to include multiple tables joins. Joins allow data to be retrieved from more than one underlying table, presenting the data as if it came from one. When joins are completed, the records involved are linked based on common fields. The process for joining tables differs depending on how the tables being joined are related to each other. PeopleSoft Query recognizes four types of joins:

- **Record Hierarchy 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).

- **Related Record Join**, which joins records from non-hierarchical records that are related by common fields. For example, description tables for common codes are related records.

- **Any Record Join (Standard Inner Join)**, which joins any two tables in the database.

- **Left Outer Join**, which joins any two tables in the database similar to an ‘Any Record Join’. However, in a left outer join, all rows of the first (left) record are present in the result set, even if there are no matches in the joining record.

In Query, predefined joins can be generated as a **Record Hierarchy Join** or a **Related Record Join**. Since these types of joins are predefined, **no criteria are needed to manually link the records**.

The Auto Join Wizard should be enabled in Query Preferences. PeopleSoft Query automatically attempts to join the new record to the existing record by looking for matching columns on the two records when **Any Record Join or Left Outer Join** are attempted.
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 key fields.)

To create a Record Hierarchy Join:

1. From the **Query Tab**, choose the base record for the query and click the **Hierarchy Join** link to the right of the base record.

All the records that have a parent/child relationship with the selected base record appear. Click the name of the table listed that should be joined.
At this point, any field can be selected from either table into one query.
Related Record Join

In a related record join, two tables are joined based on a shared field that isn't necessarily a key field. For example, if a field has a prompt table defined for it, then PeopleSoft Query displays a join link to the right of the shared field.

To create a related record join:

1. From the Query tab, choose the base record for the query and click the required Related Record Join link.

2. Select Standard Join and click OK.
Effective data criteria will be added, click **OK**.

The join is reflected on the Query Tab with the added record.

**Standard Join**

The Standard Join is typically used when the names of the records that should be joined are known. This join is also used when our query requires additional information and there is no need to retain the original results. In other words, the standard join filters and add fields to the query. Using Query Manager, a join will be created between two records by adding the initial base record, selecting fields, defining criteria, and then returning to the Record tab to select the second record.

**To create any record join:**

1. Choose the base record for the query and select the appropriate fields and criteria.

2. Return to the Record page to select the second record. The Auto Join Wizard will attempt to join the new record to the existing record by finding matching fields between the two tables.

In this Standard Join example, we are going to join the REQ_HDR table and the PO_LINE_DISTRIBUTION table to see all the Requisitions that have sourced to a Purchase Order.
Standard Join Example

Business Question: What are all the requisitions that have sourced to a Purchase Order between 02/15/2023 and 02/28/2023?

1. Main Menu > Reporting Database > Query Tools > Query Manager
2. Click Create a New Query
3. Search for and add **REQ_HDR** record

4. Select the **REQ_ID** field and click funnel shape next to **ENTERED_DT** field to add criteria without selecting the field.
5. Change the condition type to “between”
   *Verify Expression 2 Type Box indicates “Const-Const”
6. Enter “02/15/2023” in the first date box
7. Enter “02/28/2023” in the second date box

8. Click OK
9. Click Save
10. Click Run

When we first ran the query with the date criteria on theREQ_HDR record by itself, we received 764 results. These results show all Requisitions entered between our date parameters.

Create a STANDARD JOIN with the PO_LINE_DISTRIBUT table to find all the requisitions that sourced to PO. The two tables will link at Business Unit, and REQ_ID, and we will select the PO_ID field from the PO_LINE_DISTRIBUT. The system will bring back the data that matches Req ID and a PO ID. If there is no PO number, the Requisition number will not show.

11. Click on the Records tab
12. Search for PO_LINE_DISTRIBUT

*By selecting show fields next to the PO_LINE_DISTRIBUT record, user can see that there is a PO_ID and REQ_ID field.
13. Click Join Record

14. System defaults to Standard Join

15. Click the A= REQ_HDR-Requisition Header record link.

The system shows that the two records are going to join at Business Unit and Requisition ID and bring the results from the additional field(s) selected, which in this case will be PO_ID.

Results will show all Requisitions that have PO IDs associated. Those requisitions that do not, will be filtered out. (Join to filter and get additional fields)
16. Click Add Criteria
17. Choose the PO ID from Record B

Performing the Standard Join yields 1969 results. All these results have PO#. Now we notice that there is repetition. The same req# is repeated with the same PO# because we used the PO_LINE_DISTRIBUT record, and the results are by line. If the requisition has multiple lines, the results repeat the PO# according to that # of lines.
We are going to use **Distinct** to get rid of the duplicates. The distinct feature is intuitive enough to **not** remove the duplicate requisitions that have sourced to *different* PO#s.

**Using Distinct**

20. Click any tab except Run  
21. Click the Properties link at the bottom of the page

22. Check off the Distinct Box
23. Click OK

24. Click Save
25. Click Run

When we run the results again, the results decrease to 815 due to no duplicates.
Left Outer Join

In a Left Outer Join, all rows of the first (left) record are present in the result set even if there are no matches in the joining record. This allows the integrity of the data from the first table though none of those records exist in the secondary tables that are being joined. A left outer join will not filter any rows when the query is run.

To create left outer joins:

1. Choose the base record for the query and select the appropriate fields and criteria.
2. Return to the Record page to select the second record. Also, select a Related Record Join Link. The Auto Join Wizard will attempt to join the new record to the existing record by finding matching fields between the two tables.
3. Click the Join Record link on the same row as the joining record.
4. Select Join to get additional fields only (Left outer join).

Left Outer Join Example

Business Question: What are all the requisitions between 02/15/2023 and 02/28/2023 which may or may have not been sourced to a PO?

By performing a LEFT Outer Join, we should keep the integrity of the results of the first table (all requisitions entered between 2/15/2023 and 2/28/2023); however now the results should include all the requisitions, whether sourced or not to a PO due to the “join to get additional fields only”. There is no filtering associated with Left Outer Join.

Main Menu>Reporting Tools>Query>Query Manager

1. Click Create New Query
2. Search for and Add the REQ_HDR
3. Choose the REQ_ID field
4. Click the Criteria Funnel next to the ENTERED_DT field
5. Change the condition type to “Between”.
6. Enter “02/15/2023” in Date 1
7. Enter “02/28/2023” in the Date 2 box
8. Click OK
9. Save the Query
10. Click Run
11. Click records to search for and select Join the PO_LINE_DISTRIBUT record
12. Click Join Record
13. Select the Join to get additional fields only (Left Outer Join) 
14. Click the A=REQ_HDR link

The criteria show the join will occur where there is a BUSINESS_UNIT and a REQ_ID field.
15. Choose PO ID field
16. Click Properties and choose Distinct again
17. Click Save
18. Click Run

The results increase because now the query includes Requisitions that have not sourced to a Purchase Order.
Aggregate Functions in Query

An Aggregate is a property used to help perform computations or mathematical functions on a set of values to summarize or roll up the data. For example, rather than running a query that lists separate rows of data by expense account and journal monetary amounts in those accounts, an aggregate could be used to obtain the total monetary amount by expense account.

When applied to fields, Aggregate Functions redefine how that field is used throughout the query. PeopleSoft Query replaces the field, wherever it occurs, with the results of the function.

The table below lists the aggregate functions that can be applied to a field.

<table>
<thead>
<tr>
<th>Aggregate Function</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>The field does not have an aggregate function defined (Default)</td>
</tr>
<tr>
<td>Sum</td>
<td>Adds related values in each row and displays the total.</td>
</tr>
<tr>
<td>Count</td>
<td>Counts the number of rows.</td>
</tr>
<tr>
<td>Min (Minimum)</td>
<td>Checks the value from each row and returns the lowest one.</td>
</tr>
<tr>
<td>Max (Maximum)</td>
<td>Checks the value from each row and returns the highest one.</td>
</tr>
<tr>
<td>Average</td>
<td>Adds the related values in each row and divides the result by the number of rows.</td>
</tr>
<tr>
<td>Count Distinct</td>
<td>Counts the number of non-null value rows in the query result, and the duplicated rows are counted once</td>
</tr>
</tbody>
</table>

Aggregate Functions Example

**Business Question:** What was the total monetary amount for each journal by account on 02/17/2023? The Aggregate feature of SUM will narrow down multiple journal line data rows into one row of information.

1. Navigate to Query Manager
2. Search and add the JRNL_LN record
3. Select the **Business Unit**, **Account**, and **Monetary Amount** fields.
4. Click Save (at the bottom of the screen).
5. Click the **Criteria Funnel** next to **JOURNAL_DATE** field (to add the 2/17/2023 criteria).
6. Enter “02/17/2023” as the date. Then click **OK**.

7. Click the **Run** tab.

Results show every transaction (or journal line) that was recorded on that date (02/17/2023) by account. To view Monetary Amount totals by account, an aggregate sum function can be added to group the monetary amount values to form a single value by account.
8. Click the **Fields** Tab. Click **Edit** next to the Monetary Amount Field.

9. In the **Edit Field Properties** box choose from the **Aggregates**, “Sum.” Click **OK**.
10. Click **Save** and **Run**. The new results show the rows organized by single accounts. Also, the monetary amount field has changed to a Sum Amount.
**Having Criteria**

**Having** criteria is used when associated to a field with an aggregate function. Aggregated fields cannot use the regular selection criteria (**Criteria** tab); therefore, any criteria must be created using the **Having** tab.

**Having Criteria Example**

**Business Question:** What are all the accounts that net to zero from related journal lines on 02/17/2023?

After displaying the results from the **Aggregate example**, follow these steps to create the Having criteria:

1. Click the **Having** tab then choose **Add Having Criteria**.

2. Use the lookup glass in **Expression 1, Choose Record and Field** box to search for the **Monetary Amount** field. Notice it is the only field that shows up because it is the only 'aggregated' field.
Click the **A. MONETARY_AMOUNT – Monetary Amount** field.

3. Ensure Condition Type is at “equal to"
4. Constant Value = 0.00
5. Click OK
6. Click the **Run** tab. Results show the summed amounts equal to zero.
Expressions

Expressions are calculations that PeopleSoft Query performs as part of a query. They are used to display a value of a field differently than the way the field is stored or used in the record. Examples include using it to perform calculations other than the predefined aggregates, calculations against aggregated values, filter data by using expressions as selection criteria or access information from external sources.

There are two basic types of expressions, using them as columns in the query output (as field) or in criteria to filter out data. To use expressions as fields, select the **Expressions** tab, select the expression type and any expression text.

**Expressions Example**

**Business Scenario:** I need my query to show a field with the names of the employees who entered journals on 02-17-2023. Using values drawn from specific fields (First Name and Last Name), we will add an expression that displays both fields as one field in one separate column we will name: Full Name.

1. Main Menu>Reporting Tools>Query>Query Manager
2. Click Create New Query
3. Search for the JRNL_HEADER record
4. Add record

5. Select the BUSINESS_UNIT, JOURNAL_ID, JOURNAL_DATE, and scroll down to select OPR_ID.
6. Click the Criteria Funnel next to JOURNAL_DATE
7. Enter the date 02/17/2023 in the Define Constant Field
8. Click OK
9. Click the Records Tab
10. Search for the PERSONAL_DATA record
11. Click on Join Record

12. Leave default at Standard Join
13. Click the JRNL_HEADER link
Since there are no join conditions (common fields) for the two records to join, we must create the join manually. We are going to add criteria to make EMPL ID to OPR ID equal, and the records will join there.

14. Select the LAST_NAME and FIRST_NAME fields

15. Click the Fields Tab
16. Click the Criteria Funnel next to OPR_ID

17. Choose Field in the Expression 2 type
18. Use the Lookup glass to choose the EMPL_ID field from B PERSONAL_DATA record.
19. Click OK
20. Select the Expressions tab and Click Add Expressions

21. Use Expression Type: Character and modify Length to increase number of characters allowed, i.e., 30 to accommodate long names.

22. Click Add Field
23. Choose the B.FIRST_NAME field from the PERSONAL_DATA record (click show fields)
24. Type the **Concatenate formula**: (space) || (space) ‘(space)’ (space)|| (space)
25. Click on Add Field and select B.LAST_NAME
26. Click OK

![Edit Expression Properties](image)

27. Select Use as Field

![Query Tools](image)
28. Notice the Field has now been added with the expression listed as Heading Text
29. Delete LAST_NAME and FIRST_NAME and Click on Edit to change the field heading to Full Name.

30. Click Save
31. Click Run
**Common Expressions Syntax**

To tell the system how to “express” a value, the input must be specific.

<table>
<thead>
<tr>
<th>sub string</th>
<th>SUBSTR(‘ABCDEFG’,3,4) ➔ CDEF</th>
<th>give me part of the string ABCDEFG which starts in position 3 (C) and is 4 letters long ➔ CDEF</th>
</tr>
</thead>
</table>
| multiply   | A.Cost_per_unit * 500 ➔ use as Field | 1.1. give me the total cost and use the total cost as a new column in the query result  
1.2. can use the total cost in a criterion, such as to only show total cost > 5000 |
| concatenate| A.Dept_ID || ‘ - ‘ || A.Description | 1.1. This merges field results (data) together show me the result in the form:  
110401000 – Controller |
| length     | LENGTH(A.DEPT_ID) ➔ 9 |                                                                                   |
| Trim       | Trim(A.Req_ID) ➔ remove leading zeros |                                                                                   |
| System date| SYSDATE – A.APRROVAL_DATE ➔ how many days ago approved |                                                                                   |
A subquery, sometimes called a sub-SELECT, is a query whose results are used as comparison value for the selected criteria by another query. Subqueries can return a single value, a list of values, or check for existence. The condition type used in the subquery criteria will determine the subquery type.

Single Value Subquery
- Condition type that produces a single value (equal to, min, max, etc.). It can be used with a subquery or in standard criteria.

In List Subquery
- In List/Not In List condition type. It can be used with a subquery or in standard criteria.

Exists Subquery
- Exists/Does Not Exist condition type. This condition type is used exclusively with subqueries. When the condition type “Exists” is selected, the only option for Expression 2 Type is “Subquery” and there is no Expression 1 as there is no comparison. The subquery will evaluate either a true or false response.

The in list Condition Type, which is used in the subquery example provided below, prompts users to search for or create a list using the Edit List dialog box to identify the set of values your query will use for comparison when applied to a standard criteria.

To build a list of values within your query, choose the record and field under Record Alias.Fieldname. Select Condition Type: in list and click the search glass icon within the Edit List box for a list options.

In the example below, in list will be used as condition type for Expression 2 Type: Subquery, which allows to create a new query within your original query using a new record.

Subquery Example

Business Question: Our department needs to identify employees who are travel proxies and expense managers for the university.
Start by creating a top query with all employees who are travel proxies. Then, the subquery will use the results of the top query (employees who are proxies) and criteria to show who are also expense managers.

The first part of a subquery is to build the top query. Use the following steps as an example.

1. Navigate to **Query Manager** and Create New Query.

2. Search for and Add the **EX_EE_AUTH_TBL** record.
3. Choose the **AUTHORIZED_OPRID** field. Then, click the funnel next to the AUTHORIZED_OPRID field.

4. Change **Condition Type** to “not equal to”. This will help to eliminate employees who are proxy for themselves, leaving only employees that are proxy for others for the top query results.
5. Ensure “Field” is selected in Expression Type 2 box. Under the Choose Record and Field box and Record Alias.Fieldname, use the lookup glass to select the EMPL_ID field.

Select **A.EMPLID – Empl ID**.
6. Next, click the **Criteria** Tab and click **Add Criteria** button.
7. Use the lookup glass in the **Expression 1** box to select the **AUTHORIZED_OPRID** field.

Click **A.AUTHORIZED_OPRID – Authorized User ID.**
8. Change the condition type to “in list”. The “in list” condition type finds fields having a value that matches any one of the values in a list of values. With this option, there is an option to create a list or subquery within the Choose Expression 2 Type box.

9. Choose “Subquery” in the Expression 2 Type box. Select the Define/Edit Subquery link under Define Subquery, Expression 2 box.
10. By defining a subquery, the screen re-directs to the **Records** tab. It will indicate that the “working on selection subquery for ….” Type the name of another record in the **begins with** box. In this case it is “EX_APPRVR.” Click **Search**. Then select that record, click **Add Record**.

11. Click **Select** next to the **APPROVER_OPRID** field.
12. This will automatically land on the **Fields** Tab. Click on the funnel to create criteria to make B.APPROVER_OPRID equal to A.AUTHORIZED_OPRID.

13. Choose "Field" in the **Expression 2 Type** box.
14. Select the lookup glass under **Choose Record and Field, Record Alias.Fieldname.**

Select **A.AUTHORIZED_OPRID – Authorized User ID.**
15. Click **OK**.

16. Click the **Run** tab.
Result amounts may vary. At this point, the list shows employees who are proxies AND approvers.

<table>
<thead>
<tr>
<th>Row</th>
<th>Authorized ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1376914</td>
</tr>
<tr>
<td>2</td>
<td>1094596</td>
</tr>
<tr>
<td>3</td>
<td>1375808</td>
</tr>
<tr>
<td>4</td>
<td>1688670</td>
</tr>
<tr>
<td>5</td>
<td>6233833</td>
</tr>
<tr>
<td>6</td>
<td>1064596</td>
</tr>
<tr>
<td>7</td>
<td>1780463</td>
</tr>
<tr>
<td>8</td>
<td>1064596</td>
</tr>
<tr>
<td>9</td>
<td>1091040</td>
</tr>
<tr>
<td>10</td>
<td>1750463</td>
</tr>
<tr>
<td>11</td>
<td>1780463</td>
</tr>
<tr>
<td>12</td>
<td>1298015</td>
</tr>
<tr>
<td>13</td>
<td>1727902</td>
</tr>
<tr>
<td>14</td>
<td>2644829</td>
</tr>
<tr>
<td>15</td>
<td>1331764</td>
</tr>
<tr>
<td>16</td>
<td>1091040</td>
</tr>
<tr>
<td>17</td>
<td>1367108</td>
</tr>
<tr>
<td>18</td>
<td>1377640</td>
</tr>
<tr>
<td>19</td>
<td>1064596</td>
</tr>
<tr>
<td>20</td>
<td>1780463</td>
</tr>
<tr>
<td>21</td>
<td>1064596</td>
</tr>
<tr>
<td>22</td>
<td>1064596</td>
</tr>
</tbody>
</table>
17. To further refine the query, the approvers need to specifically be expense managers. Click the **Criteria** Tab and click the **Add Criteria** button.

18. Use the lookup glass to view records in the Expression 1 box

19. Choose Show Fields next to the B record

20. Choose APPROVER_PROFILE
21. Leave Condition type as equal to
22. Type “EXPENSE MANAGER” in the Constant box

(The reason why we are isolating the Expense Manager approver type is because approvers can be HR Supervisors, Project Managers, etc.)

23. Click Ok.
24. Click the Run tab.
Unions

Unions combine the data resulting from two or more separate queries. A union of two queries can take place only 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 (aka fields).

Therefore, before creating a union, it is useful to create the queries separately first to identify the information contained in the fields and ensure the same order and data type is selected. Please see the Union Example below.

Union Example

Business Question: We need to create a query that contains information for Project and Expense Managers in the University.
First identify the information contained in the fields, decide how many columns to select, and match the fields in both queries.

1. Create both queries separately using the EX_APPRV record for the first and the PROJECT_MGR record for the second query. Select the desired fields and view the results to determine the number of columns and the information that matches.

Note: The Query tab can be used to select and deselect fields as needed.
2. Create a New Query with a union. Search for EX_APPRVR- Approvers record. Click Add Record.

3. Choose the BUSINESS_UNIT, APPROVER_PROFILE, APPROVER_OPRID, and CHARTFIELD_FROM. The EX_APPRVR table is created first. The data in its fields will be combined in a union with the same number of fields and data type contained in the PROJECT_MGR table as indicated in the table shown in step 1.

4. 
5. Use the **Save As** link to save the query.

![Save As link highlighted in a query interface]

6. Check the fields order in **Fields**.

![Fields section in a query interface]

7. Click the **New Union** link in any of the tabs. It will not be available in the Run tab.

![New Union link highlighted in a query interface]
8. The **Records Tab** appears showing **Working on selection** "Union 1." arch for the PROJECT_MGR record. Click on **Add Record**.

**Note:** The **Subquery/Union Navigation** appears. It allows navigation between the **Top Level of Query** (EX_APPRV table) and **Union 1** (PROJECT_MGR table).

- Select the fields with similar data. Choose BUSINESS_UNIT, PROJ_ROLE, PROJECT_MANAGER, and PROJECT_ID. The Reorder/Sort functionality will allow you to organize the fields of the union to match the data in the Top Level of Query.

- Select the Fields tab and click on the **Sort/Reorder Button**.
As indicated in the table shown in step 1 PROJECT_ROLE matches APPROVER_PROFILE, PROJECT_MANAGER matches APPROVER OPRID, and PROJECT ID matches CHARTFIELD_FROM, so we need to add that order.
- Verify the fields in the union by clicking the **Fields** Tab. The fields used in the Top-Level of Query can be viewed by toggling. Click the **Subquery/Union Navigation** link.

- Click **Top Level of Query** to see Record A.
• The system goes back to the **Query** tab. Click the **Fields** tab to finish reviewing the Top-Level of Query fields.

![Image of query tool](image1.png)

• Before selecting run, make sure to answer the three important Union tests:

1. Do you have the same number of output columns? In this example there are 4.
2. Do you have the same data type? Yes, the data in the fields of the Top Level of Query match the data in the fields of Union 1. Refer to the screenshot and the table in step 1.
3. Are they in the same display order? Yes: BUSINESS_UNIT, PROJECT_ROLE / APPROVER_PROFILE, PROJECT_MANAGER / APPROVER_OPRID, and PROJECT_ID / CHARTFIELD_FROM.

![Image of query tool](image2.png)

Then click the **Run** tab.
The results show approximately 17468 rows of data for both Project Managers and Expense Managers and display the headings from the first record.
Managing Queries

Schedule a Query

Schedule Query enables users to run a query process in the background and deliver it at a predetermined day, time, and delivery method. There are two ways to schedule a query, from the Schedule Query navigation and from the Schedule link found in Search Results.

From the Schedule Query navigation within either the Reporting Database or Production environment following these steps:

1. Click the Add a New Value tab to create a Run Control ID. Give it a unique name and click Add. Spaces and symbols are not allowed in a run control name.

2. Select a query by clicking Search or type the name of the query into the Query Name field.
From the search results, select the query by clicking on the name.

Some queries may have prompts or criteria that must be entered before scheduling. Enter any required information.
3. The chosen query and parameters will be displayed. Click the Update Parameters link to make any changes and click Save. After saving, click Run.
4. The **Process Scheduler Request** screen appears. This screen has options for scheduling recurrence, date, time, and output options.

**Recurrence** options:

**Run Date** options: Use the calendar to choose a specific date.

Output options can be chosen under **Type** drop-down. Web and Email are the most common options. Email will email the query at the desired day and time. The Web option will produce the query result on screen.
**Format** is the type of file that will be produced. TXT for a CSV file or XLS for an Excel spreadsheet file are the most common.

5. After choosing all the parameters of the schedule, click **OK**.

A Process Instance number will appear, this confirms that the schedule is created and will process at the date and time and recurrence indicated.
Email a Query

To schedule a query to be emailed at specific days and times, follow steps 1-4 in the Schedule Query section.

1. On the Process Scheduler Request screen, make sure to choose Email as a Type instead of Web. Also, click the Distribution link to enter email address(es).

2. There are options to enter an Email Subject, Message Text, Email Address List. Click OK.

The email arrives at the scheduled date and time.
Adding Queries to My Favorite Queries

Place frequently used queries in the My Favorite Queries list for easy access. This option is available through Query Manager as well as Query Viewer.

To add a query to the My Favorite Queries list:

1. On the Query Manager Search Results page, select the query and add it to the My Favorite Queries list (click the box to the left of the query). It will appear as follows:

2. Select Add to Favorites from the Action drop-down list box.

3. Click Go.

The query appears in the My Favorite Queries list group box.
Copying a Query to Another User’s List of Queries

The Query Manager allows users to copy a private query to another individual to view that private query. It is important to note that users can only copy *Private* queries to another user’s list of queries.

If the intended user does not have permission to access all the records in a copied query, that query will not appear in the target user’s list of queries. Some queries are also created within one environment versus another e.g., Production vs Reporting. The intended user must also have access to that environment to view the query.

Once the query is copied, it will then appear in the intended user’s list of queries in Query Viewer.

Click on the **Query Tools** Tile, and **Query Manager**:  

**To copy a query to another user’s list of queries:**

1. On the Query Manager Search Results page, select or search for the query or queries that are intended for another user.

![Query Manager Search Results](image)

2. Select **Copy to User** from the **Action** drop-down list. Click **Go**.

![Query Manager Search Results](image)
3. Enter the Panther ID of the intended user. Click **OK**.

4. A message will appear indicating the query was successfully copied to the designated user.

5. The intended user must then navigate to the environment in which the query was created (Reporting or Production), go to **Query Tools**, then to **Query Viewer**, and search for the name of the query that was copied. Please note that no email is generated to advise the user about the copied query.
Moving a Query to an Organization Folder

Folders are a tool within Query Manager to help users organize queries under a common heading. To move a query to an organization folder:

1. On the Query Manager Search Results page, select the query or queries to move to an organization folder.

2. Select Move to Folder from the Action drop-down. Click Go.
3. The **Move to Folder** box will appear. Choose an existing folder from *All Folder dropdown* or choose to enter the name of a new folder. Click **OK**.

Queries moved into a specific named folder are now searchable within **Query Viewer or Query Manager** using the **Folder Name** Advanced Search field.
**Renaming a Private Query**

To rename a private query, a user must be the owner of the query. Navigate to **Query Manager**.

1. On the **Query Manager** page click **Search**. This will initiate the **Action** drop-down choices.

   ![Query Manager Screenshot](image1)

2. Select the checkbox next to the query to be renamed. Then choose the “Rename Selected” option from the Action drop-down. Click **Go**.

   ![Query Manager Screenshot](image2)

3. Enter the new desired name of the query in the New Name box, then click **Ok**.

   ![Rename Queries Screenshot](image3)