PeopleSoft Query 201
Intermediate

Data Services

Oct 2016
# Table of Contents

- Course Goal 0
- Course Learning Objectives ................................................................. 0
- How to Use This Manual ......................................................................... 0
- Section 1 Introduction ........................................................................... 1
- Adding Criteria to a Left Outer Join ....................................................... 1
- Effective Data .......................................................................................... 3
- Effective Date, Effective Sequence, and Effective Status ......................... 3
- Effective Date and Query ......................................................................... 3
- Effective Status and Query ....................................................................... 5
- Effective Sequence and Query .................................................................. 6
- Effective Dates and Left Outer Joins ....................................................... 7
- Multiple Effective Dated Records .......................................................... 8
- Queries of Queries .................................................................................. 10
- Finding Record and Field Information ................................................... 13
- CTRL+SHIFT+J shortcut keys ................................................................. 13
- CTRL+SHIFT+C shortcut keys ................................................................. 15
- PS Chrome Extension ............................................................................. 15
- Field Information Icon ........................................................................... 16
- Page Information Icon .......................................................................... 16
- Using the metaLink Data Dictionary ....................................................... 17
- Logging into the metaLink Data Dictionary ............................................ 18
- Using the metaLink Data Dictionary ....................................................... 18
- Record Information ................................................................................ 19
- Field Information .................................................................................... 20
- Verifying your Query Results ................................................................. 22
- Section 1 Review .................................................................................... 23
- Section 1 Knowledge Test ....................................................................... 24
- Section 1 Exercises Campus Solutions .................................................. 26
- Section 1 Exercises Human Capital ....................................................... 27
- Section 1 Exercises Financials ............................................................... 28
- Section 2 Introduction ............................................................................ 29
- Aggregate Functions .............................................................................. 29
- Applying Aggregate Functions ............................................................... 30
- Having criteria ........................................................................................ 31
- Using the Having tab ............................................................................. 31
- Expressions 32

---

**Page | ii**
<table>
<thead>
<tr>
<th>Section</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Section 2</td>
<td>Exercises Campus Solutions</td>
</tr>
<tr>
<td>Section 3</td>
<td>Introduction</td>
</tr>
<tr>
<td>Section 2</td>
<td>Exercises Financials</td>
</tr>
<tr>
<td>Section 2</td>
<td>Exercises Human Capital</td>
</tr>
<tr>
<td>Section 2</td>
<td>Review</td>
</tr>
<tr>
<td>Section 2</td>
<td>Knowledge Test</td>
</tr>
<tr>
<td>Section 3</td>
<td>Introduction</td>
</tr>
<tr>
<td>Section 2</td>
<td>Bind records in Population Selection queries</td>
</tr>
<tr>
<td>Section 2</td>
<td>Pop Select Query Protocol</td>
</tr>
<tr>
<td>Section 3</td>
<td>Introduction</td>
</tr>
<tr>
<td>Section 2</td>
<td>Exercises Financials</td>
</tr>
<tr>
<td>Section 2</td>
<td>Exercises Human Capital</td>
</tr>
<tr>
<td>Section 3</td>
<td>Introduction</td>
</tr>
<tr>
<td>Resources</td>
<td>Adding Information to MetaLink</td>
</tr>
<tr>
<td>BI Publisher Introduction</td>
<td></td>
</tr>
<tr>
<td>BI Publisher Query Report Viewer</td>
<td></td>
</tr>
<tr>
<td>BI Publisher Query Report Scheduler</td>
<td></td>
</tr>
<tr>
<td>BI Publisher Report Search</td>
<td></td>
</tr>
<tr>
<td>Section 3</td>
<td>Review</td>
</tr>
<tr>
<td>BI Publisher Introduction</td>
<td></td>
</tr>
<tr>
<td>BI Publisher Query Report Viewer</td>
<td></td>
</tr>
<tr>
<td>BI Publisher Query Report Scheduler</td>
<td></td>
</tr>
<tr>
<td>BI Publisher Report Search</td>
<td></td>
</tr>
<tr>
<td>Section 3</td>
<td>Review</td>
</tr>
</tbody>
</table>
Course Goal

The goal of this course is to empower query Developers with the tools and skills needed to create and manage more complex queries.

Course Learning Objectives

At the end of this course users will:

• Be able to add criteria to a Left Outer Join.
• Be able to apply effective data logic to queries.
• Utilize queries of queries to effectively search for query data information.
• Efficiently find record and field information using the front end of PeopleSoft in conjunction with other tools.
• Gain the knowledge and skills to check your query results to be certain of query accuracy.
• Effectively apply Aggregate Functions to query results.
• Utilize the Having tab to add criteria to fields using Aggregate Functions.
• Be able to create Expressions to allow complex refining of query results.
• Understand BIND records and how they affect query development.
• Effectively use MetaLink to search and create query metadata.
• Understand how to run reports created with BI Publisher.

How to Use This Manual

The PS Query 201 Training Guide is comprised of 3 sections.

• Section 1 covers when and how criteria should be added to a join and how effective data can affect query results and how to use this functionality to achieve the desired output. We will also learn about how to use a query to search queries for specific information as well as how to find record and field information from the front end of PeopleSoft. Finally we will see how to best check the accuracy of query results.
• Section 2 is where we will learn how to apply Aggregate Functions and use the Having tab to add criteria to fields where Aggregate Functions have been applied. Finally, we will cover using Expressions in queries which will allow for complex refining of results.
• Section 3 details the effect of BIND records on queries and how to use MetaLink for searching query, record and field, Report and Script metadata as well as how to run BI Publisher reports.

Each section will include:

• Section Introduction
• Section Review
• Knowledge test
• Hands on exercises where appropriate
**Section 1 Introduction**

Welcome to PeopleSoft query 201 Intermediate Course!

At the end of this section you be able to overcome some common challenges in query development by learning how to:

- Be able to add criteria to a Left Outer Join.
- Be able to apply effective data logic to queries.
- Utilize queries of queries to effectively search for query data information.
- Efficiently find record and field information using the front end of PeopleSoft in conjunction with other tools.
- Gain the knowledge and skills to check your query results to be certain of query accuracy.

**Adding Criteria to a Left Outer Join**

When adding criteria to a query that contains a Left Outer Join you will notice that there will be the option of where to place the join in the SQL code; on the **ON** clause of the Left Outer Join or on the **WHERE** clause. The **ON** clause applies the criteria to the results of the Left Outer Join whereas the **WHERE** clause applies the criteria to the entire result set.

The basic rule of thumb is to always place the criteria on the clause of the record corresponding to the field used in Expression Type 1. In the example to the right, we are placing criteria on the Expression 1 Type field **ACAD_CAREER** from the B record. Thus, when indicating where to place the criteria we will select the corresponding **ON** clause of outer join B. If the field belongs to the A record, which is always the base record, then the criteria will belong to the **WHERE** clause.

Say you want a complete list of students but only want to see their grade if they took Math. To do this you create a query from a student record and then create a Left Outer Join with a second record that will display the subject and grades received.

Since you only want to see students Math grade you will need to create criteria for this. Where you place that Criterion is KEY to how PeopleSoft query will determine the result set. Basically, how PeopleSoft query determines the results is entirely dependent on the placement of the criteria in the SQL code; either on the **WHERE** clause or the **ON** clause.

The reason for this is because there is a subtle but marked distinction in how PeopleSoft query displays the results of the query based on the position of the criteria clause. Again, The **ON** clause applies the criteria only to the results of the Left Outer Join whereas the **WHERE** clause applies the criteria to the entire query result set.

To see how this difference works, let’s take a look at an example of how this placement of the code will affect the results. Let’s look at a query where the Criterion has been placed on the **ON** clause of Outer Join B as it should be. The **ON** clause applies the criteria only to the Left Outer Join and is actually applied before the join is completed. What this means is that the **ON** clause will compile a list where either Math or NULL is the result and then the Left Outer Join will compare that list to record A to determine the match. So the end result is that the query will display all students from record A and then display either Math or NULL in the subject field along with the corresponding grade if they took Math exactly as we intended.
Criteria placed on the ON Clause.

Results correctly display both MATH and NULL.

Now let’s look at a query where the criteria has been placed on the WHERE clause. In the initial calculations done by PeopleSoft query, the Left Outer Join returns all students, even if they didn't take Math. However, if they didn't take Math, then the joined row that is returned by the Left Outer Join will have NULLs in all the columns from the grades table. So as PeopleSoft query continues the calculations and the WHERE clause comes along, it displays only those rows where Math is the subject since we put the condition of Math on the Where clause. So, since NULL isn't Math, students who didn't take Math disappear from the final calculated result set.

So, in essence by placing the criteria on the WHERE clause we are negating the Left Outer Join and asking PeopleSoft query to only return results where the student took Math which is not what we wanted to happen.

Criteria placed on the WHERE Clause.

Results incorrectly display only MATH.
Here is a chart that details how PeopleSoft query processes a request to place criteria on the ON Clause as well as how it deals with a request to place criteria on the WHERE Clause when using a Left Outer Join in query creation.

**Effective Data**
There is frequently a need to keep track of historical changes to information within the database. The student status will change as he or she enrolls or graduates, etc. Name, addresses, service indicators, or amounts are also often kept as historical references. There may also be a need to store something that will be effective in the future.

**Effective Date, Effective Sequence, and Effective Status**
PeopleSoft uses three different fields in many records to enable having data be effective only at certain times – Effective Date (EFFDT), Status as of Effective Date (EFF_STATUS) and Effective Sequence (EFFSEQ).

Effective-dated records/tables contain the field EFFDT. This is the most commonly used of the three fields and is used throughout PeopleSoft applications to give data a historical perspective and allows for viewing data changes over time. Whenever users add a row of data to the table, they specify the date on which that data becomes effective. PeopleSoft will keep the new date as the current effective date and the previous row as a historical effective date.

Effective status indicates which row is currently active and which rows were previously active (inactive) for configuration history. This field is generally used in configuration tables.

Effective sequence tracks the sequence of changes in rows for transactional history. Generally used in transactional tables, this field is used when same day changes are expected.

**Effective Date and Query**
With Effective dated records, rows of data are classified in one of three categories:

1. **Current** – The latest effective date that is less than or equal to today’s date. Only the current row will display in the query results.
2. **History** – The effective date is less than the current effective date.
3. **Future** – The effective date is greater than the current date.

The EFFDT field allows users to display the following in their query results based on the criteria selected for the EFFDT field:

- All rows, regardless of their effective dates.
- Rows which are currently effective.
- Rows which were effective in the past.
- Rows which will be effective in the future.
**EFFECTIVE DATE BASE DATE**

You may notice some records displaying an effective date of 01/01/1901. This is the default effective date added by PeopleSoft when there is no other effective date available and indicates the first instance of the effective dated item.

**EFFECTIVE DATE CRITERIA**

When you select an effective dated record for your query a pop up widow will appear letting you know that an Effective Date criteria has been automatically added for this record.

![Message]

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

This message is telling you that the database will automatically give you the very latest dated row for the criteria. It is equal to or less than the current date (today) and will display only results which meet this Criterion. To return results that differ from the default, you may remove or alter the Criterion that has been automatically created.

**EFFECTIVE DATE OPERATORS**

As previously stated automatically added Effective Date criteria will return one effective dated row of information per item which will be the latest dated row. You can then select different options to compare the effective date against such as a date in the future or the past. While all of the standard operators are available for use with criteria, you will find that there are some special Effective Date Condition Types also available. Each of these operators will return the last effective dated result equal to or less than today by default.

Special Effective Date Condition Types (operators) are:

- **Eff Date <=** Returns the one row that is closest or equal to the Expression 2 date value without exceeding it. If Expression 2 is “Current Date”, then query will retrieve the most current row from the table.
- **Eff Date <** Returns the one row that is closest to the Expression 2 date value without exceeding it.
- **Eff Date >=** Returns the one row that is the nearest to being equal to or greater than the Expression 2 date.
- **Eff Date >** Returns the one row that is nearest to and greater than the Expression 2 date.
- **First Effective Date** Returns the row with the oldest effective date, usually the first row that is entered for an item.
- **Last Effective Date** Returns the row with the latest effective date, even if that date is still in the future.
Note that when using the special Effective Date Condition Types there are only 4 options for Expression 2 Type in the Edit criteria Properties Page:

- **Field** – Will pull up a list of available date fields in the records that can be used as the comparison value.
- **Expression** – Allows for the creation of an Expression as the comparison value.
- **Constant** – Displays a date field where the date to be used as the comparison value is entered.
- **Current Date** – Defaults to the current date as the comparison value.

Again, the standard Condition Types are available so you are able to select between certain dates, NULL, etc. Two operators of note are “not less than” and “not greater than”. These operators function the same as “equal to or less than” or “equal to or greater than”. The “not less than” operator will return all rows equal to or higher than the Expression 2 date (future dated rows) and “not greater than” will return all rows equal to or less than the Expression 2 date (historical rows).

**Effective Status and Query**

The EFF_STATUS works in conjunction with the EFFDT field and provides a status as of the effective date. The EFF_STATUS field has two values: Active and Inactive. You are able to specify if you would like to see only rows that are active or inactive by using criteria on your EFF_STATUS field. No automatic criteria are added for this field and the default is to show all available statuses, both active and inactive.

<table>
<thead>
<tr>
<th>ID</th>
<th>Addr Type</th>
<th>Eff Date</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>48</td>
<td>BUSN</td>
<td>12/27/2016</td>
<td>A</td>
</tr>
<tr>
<td>38</td>
<td>BUSN</td>
<td>12/27/2018</td>
<td>A</td>
</tr>
<tr>
<td>60</td>
<td>BUSN</td>
<td>12/23/2016</td>
<td>A</td>
</tr>
<tr>
<td>37</td>
<td>BUSN</td>
<td>12/22/2016</td>
<td>A</td>
</tr>
<tr>
<td>30</td>
<td>BUSN</td>
<td>12/22/2016</td>
<td>A</td>
</tr>
<tr>
<td>35</td>
<td>BUSN</td>
<td>12/20/2016</td>
<td>A</td>
</tr>
<tr>
<td>34</td>
<td>BUSN</td>
<td>12/22/2016</td>
<td>A</td>
</tr>
<tr>
<td>31</td>
<td>BUSN</td>
<td>12/20/2016</td>
<td>A</td>
</tr>
<tr>
<td>30</td>
<td>BUSN</td>
<td>12/19/2016</td>
<td>A</td>
</tr>
<tr>
<td>85</td>
<td>BUSN</td>
<td>12/16/2016</td>
<td>A</td>
</tr>
<tr>
<td>90</td>
<td>BUSN</td>
<td>12/14/2016</td>
<td>A</td>
</tr>
<tr>
<td>20</td>
<td>BUSN</td>
<td>12/14/2016</td>
<td>A</td>
</tr>
</tbody>
</table>

Statuses can also be entered into the database before they will become effective or after the date for the status change has already passed. For example, the academic plan below was set to change to inactive as of 6/29/16.

<table>
<thead>
<tr>
<th>Acad Plan</th>
<th>Eff Date</th>
<th>Status</th>
<th>Descrip</th>
<th>Plan Type</th>
<th>Acad Prog</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEMIMAPT</td>
<td>06/29/2016</td>
<td>I</td>
<td>Industrial Mechanics AAS</td>
<td>MAJ</td>
<td>PRFTC</td>
</tr>
<tr>
<td>HEMIMAPT</td>
<td>01/01/1901</td>
<td>A</td>
<td>Industrial Mechanics AAS</td>
<td>MAJ</td>
<td>PRFTC</td>
</tr>
</tbody>
</table>
Effective Sequence and Query

Effective Sequence (EFFSEQ) also works in conjunction with EFFDT and enables the entry of more than one row with the same effective date. The system assigns a unique sequence number to each row that has the same effective date. So if more than one event occurs for a given key on the same date, the EFFSEQ is used as a key to uniquely identify each row. The earliest event is numbered 0 and the highest number in this key is the most current for the given date. By default when using EFFSEQ only the last action will display. The criteria that is automatically added will display as Eff Date <= Current Date (EffSeq = Last) as shown below.

With the effective sequence set to last the results will only display one row with the last entry for the effective sequence.

In order to display a different sequencing, select to edit the criteria and then choose to change the default of Last and select either First for the first sequenced entry or All for all sequenced rows.

The criteria will then display as shown below:

Now the results for the query will display all available sequencing information for the entry.
Effective Dates and Left Outer Joins

The criteria for effective dating will always be placed on the WHERE clause which is a notable exception to the rule for where to place criteria in the SQL code that we learned in the Adding criteria to a Left Outer Join module of this manual.

When using a Standard join there is no need to add any special criteria to your query, however, if using a Left Outer Join, you must provide an option for the effective date to be NULL in order for the Left Outer Join to function as desired. This is because the result of any Left Outer Join will either contain data or be NULL and all EFFDT criteria is automatically added to the WHERE clause thus there must be an option for the EFFDT to be NULL as well or only records that meet the EFFDT criteria will display – negating the Left Outer Join.

As an example, take record STDNT_CAR_TERM as record A and STDNT_GRPS_HIST Left Outer Joined as record B. The desired result is to see all students regardless if the student belongs to a particular student group. For the Left Outer Join to function correctly and display blanks if there is not a match between records, we must add the additional EFFDT can be NULL criteria and group it with the existing EFFDT criteria with a logical operator of OR. If this step is skipped, the Left Outer Join will be negated and the join will behave as a standard join.

Let’s take a closer look at this. In the example below, effective dated criteria has been automatically added by PeopleSoft to effective dated record B, STDNT_GRPS_HIST, which has been Left Outer Joined to record A STDNT_CAR_TERM.

![Query Name: QCS_TRAIN_LEFT_OUTER_EFFDT](image)

Even though record B has been Left Outer Joined to record A, there is no option for the EFFDT to also be NULL. So if the query is ran it will only return results where there is an effective dated match between both records. In essence, it will behave as a standard join. Remember EFFDT criteria is added to the WHERE clause so the EFFDT criteria is being applied to the entire result set.

![Query Results](image)
However, if we add the option for the effective date to be NULL and group the two Effective Date criteria together with an OR logical operator then the Left Outer Join functionality will be restored and the results returned will show blank for any rows where there is not an effective dated match between records.

Once the Criterion has been added in the correct manner the results will now display as a Left Outer Join.

Multiple Effective Dated Records

When joining two effective dated records together it is very important to understand how PeopleSoft query reads effective dating logic as this will determine what data is returned by your query.
PeopleSoft query will read the Effective Date criteria used in EACH record and return results based on that criteria regardless of join type (both standard and Left Outer Joins allow this). Meaning that if, for example, record A is asking to return results that are equal to or less than today and record B is asking to return results from a specific time period, query is able to handle this request easily even though the Effective Date criteria is not the same for each record. In the example below record A (JOB) is pulling employees who have an active employee effective date of equal to or less the current date. Record B (LIFE_ADD_BEN) is pulling employees whose effective date of a benefit enrollment is between 1/1/16 and 12/31/16. The join is a standard join type so there is no need to add NULL as an effective date option for record B. Any and all subsequent effective dated records Left Outer Joined must allow the effective date to be NULL.

When the query is ran it returns results for active employees who added life or accidental death insurance in the year 2016.
Queries of Queries

It can be very useful to have the ability to write queries that return results ABOUT a particular query or group of queries. Many people are interested in finding queries written by themselves or someone else in their institution. Or perhaps you would like to find when a particular query was last ran or who created the query. There are in fact records that contain information about queries themselves that we can use to write queries. For example, to search the Definition field for institution code you would write a query that searches the Definition field. What record does this information live in? There are actually many records that hold query data and they all start with PSQRY. As of 8.54 there are the following query records available in ctcLink.

- PSQRYBIND – Stores all run-time prompt data
- PSQRYBINDLANG - Related language record for PSQRYBIND
- PSQRYCRITERIA – Stores all criteria Expressions in code format
- PSQRYDEFN – Stores high-level query definitions with version numbers.
- PSQRYFIELD – Stores all fields used in all aspects of query operation
- PSQRYDEL – Stores operator ID, query name and version
- PSQRYEXECLOG – Stores information on the execution of queries
- PSQRYEXPR – Stores the text associated with each criteria Expression
- PSQRYFIELD – Stores all fields used in all aspects of query operation
- PSQRYFIELDLANG - Related language record for PSQRYFIELD
- PSQRYLINK – Stores the relationships to child queries
- PSQRYRECORD – Stores all records used in all aspects of query creation
- PSQRYSELECT – Stores all SELECT requirements by select type, i.e. union, subselect, join,....
- PSQRYSTATS – Stores run time statistics

In addition there are also records which store page or panel information.

- PSPNLDEFN - Panel Definition
- PSPNLDEFN_VW - Panel Definition View
- PSPNLFIELD - Panel field
- PSPNLFIELD_VW - Panel field View
- PSPNLGDEFNLANG - Panel Group Related Lang
- PSPNLGROUP - Panel Group
- PSPNLGROUPLANG - Panel Group Related Language
- PSPNLGRPDEFN - Panel Group Definition
- PSPNHTMLAREA - Panel HTML Area
- PSPNLTREECTRL - Panel Tree Control

So as you can see there are quite a few tables available for use in creating a query of queries and all sorts of useful information we can extract. To create a query of queries using the PSQRYDEFN record, in Query Manager select to create a new query with PSQRYDEFN as the A record. Join PSQRYSTATS as record B. Make your field selections and execute the query.
The snapshot below shows the results of a query in FS created as in the example above.

To make the query more valuable you can make the definition field searchable by adding a wildcard prompt on the DESCRLONG (Definition) field and you can then search for queries created by a certain college code, email address or name. (Step by Step found in this Manual in the Simplified Instructions Glossary).

It is also possible to use a query of queries when searching for record and field information as we will see in the next module.

Using the PSPNLFIELD record users are able to search out record and field information regarding a PeopleSoft front end page name. While, there are queries already created to use for this purpose we will still review the steps here.

Go to Query Manager in the Pillar where you would like to create the query and opt to “Create New Query”.

Search for the record: PSPNLFIELD

Make your field selections
Create a prompt where end users will be able to enter in a specific page name found in the front end of PeopleSoft.

Save changes to your query and run it to test.

<table>
<thead>
<tr>
<th>Page Name = STDN_T_GROUPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Panel Name</td>
</tr>
<tr>
<td>-------------</td>
</tr>
<tr>
<td>1 STDTN_GROUPS</td>
</tr>
<tr>
<td>2 STDTN_GROUPS</td>
</tr>
<tr>
<td>3 STDTN_GROUPS</td>
</tr>
<tr>
<td>4 STDTN_GROUPS</td>
</tr>
<tr>
<td>5 STDTN_GROUPS</td>
</tr>
<tr>
<td>6 STDTN_GROUPS</td>
</tr>
<tr>
<td>7 STDTN_GROUPS</td>
</tr>
<tr>
<td>8 STDTN_GROUPS</td>
</tr>
<tr>
<td>9 STDTN_GROUPS</td>
</tr>
<tr>
<td>10 STDTN_GROUPS</td>
</tr>
<tr>
<td>11 STDTN_GROUPS</td>
</tr>
<tr>
<td>12 STDTN_GROUPS</td>
</tr>
<tr>
<td>13 STDTN_GROUPS</td>
</tr>
<tr>
<td>14 STDTN_GROUPS</td>
</tr>
<tr>
<td>15 STDTN_GROUPS</td>
</tr>
</tbody>
</table>
Finding Record and Field Information

There may be times when you know a specific field is needed in a query but you don’t know which record the field lives in or you may not know the exact field name used by the database. Perhaps you have a record in mind but want to find out more information about it first. There are a few ways to find information on records and fields from the front end and/or using other methods.

- CTRL+SHIFT+J shortcut keys
- CTRL +SHIFT+C shortcut keys
- PSChrome Extension
- Metalink

The idea is to find the page name that the field is found on and then search that page name in a PSPNLFIELD query for details regarding the name of the field and the record where the field can be found or to search out this record and field information directly from the front end, depending on the process.

CTRL+SHIFT+J shortcut keys

Note: this process is not available in the production environment.

CTRL+SHIFT+J allows users to find the page name that can then be looked up in a PSPNLFIELD query to find record and field information. This process requires the user to press the Control key simultaneously with the Shift key and the J key to bring up a data sheet that details information specific to the page requested. For example, if the user navigates to the Student Groups page (Main Menu  records and Enrollment  Career and Program Information  Student Groups). And there wants to see field and record information from that page to use in their query, the user could press CTRL+SHIFT+J to access the data sheet for that page. NOTE: in Chrome the user must input CTRL+SHIFT+J TWICE for the process to work.
The data sheet lists the name of the PeopleSoft page which can be used to find which fields can be found on that page. In this example the page name is STDNT_GRPS. Note or copy the page name and then navigate to PeopleSoft Query Manager or Query Viewer.

Each Pillar has a query already available where users are able to search information related to a particular page name. Those queries are:

- QCS_PSPNLFIELD
- QHC_PSPNLFIELD
- QFS_PSPNLFIELD

In our example we are in the CS Pillar so we would search for the query - QCS_PSPNLFIELD in Query Manager or Query Viewer.

Run the QCS_PSPNLFIELD query to HTML and when prompted, enter the page name in the Prompt window then click View Results.

The results display the names of the fields found in the page and the record where each field lives. The column Label Text holds the field description while the Label ID column holds the field name as used in the database. The record where the field is found is in the record column.

Note: some records displayed are derived and cannot be used in this process. A "derived table" is essentially a statement-local temporary table created by means of a subquery in the FROM clause of a SQL SELECT statement. It exists only in memory and behaves like a standard view or table but cannot be used as a data source in PeopleSoft query. WRK tables behave in a similar fashion and can also not be used as a data source for PeopleSoft query.
CTRL+SHIFT+C shortcut keys

The CTRL+SHIFT+C option will only work in Chrome and Firefox. This tool is not available in Internet Explorer.

The shortcut keys work in a similar fashion to the CTRL+SHIFT+J shortcut, however instead of displaying the Panel Page, they display the name of the record where a field on the front end will store its data when the mouse is hovered over the field. The example below shows that the field Reason on the Enrollment page will write to the record STDNT_ENRL and the field name is ENRL_ACTN_RSN.

PS Chrome Extension

Install this extension and use it anywhere in the front end to get record and field information. Go to https://google.com/chrome/webstore from within the Chrome browser.

Click on The Chrome Webstore hyperlink under Distribution Options.

Search for "PeopleSoft" in the search bar.

Click on “Add to Chrome” next to the PSChrome Extension.
Now you are ready to start using the PSChrome Chrome Extension in PeopleSoft. You will see the following group of icons now available for use when working in PeopleSoft.

To use the icons, navigate to the page where the field you want to query on is found in the front end of PeopleSoft. For example, for Student Groups go to Main Menu ➔ records and Enrollment ➔ Career and Program Information ➔ Student Groups.

Field Information Icon 🌴

The field Information Icon will return the name of the field as it appears in the database. To use the field click on it and the screen will change to display red dotted line boxes around the names of the field as the record where it lives as well as any current values will display.

Click inside the red dotted line box then scroll to the bottom of your screen to see the resulting information. The name of the field and the value currently entered will display.

Page Information Icon 📚

The Page Information Icon will display the page information when clicked, including the Page Name which can be used in the PSPNLFIELD queries.

Using the same example of Student Groups used before, let’s see how to retrieve the page name using the Page Information Icon. After navigating to the Student Groups Page (Main Menu ➔ records and Enrollment ➔ Career and Program Information ➔ Student Groups) click on the Page Information Icon.
The Page Information will display, including the Page Name which can be used in the PSPNLFIELD queries.

Here, you would again simply navigate to Query Manager or Query Viewer to use the PSPNFIELD query.

Run the QCS_PSPNLFIELD query to HTML and when prompted, enter the page name in the Prompt window then click View Results.

The results display the names of the fields found in the page and the record where each field lives.

Note: some records displayed are derived and cannot be used in this process. A "derived table" is essentially a statement-local temporary table created by means of a subquery in the FROM clause of a SQL SELECT statement. It exists only in memory and behaves like a standard view or table but cannot be used in PeopleSoft query.

**Using the metaLink Data Dictionary**

With any PeopleSoft implementation one of the most difficult thing for users responsible for Query and Report development is to determine what data lives where and/or to understand the meaning and usage of each record and field. The sheer number of available tables and fields can be overwhelming for anyone attempting to learn the data structure. In addition end users reading reports may have questions about the exact nature of the data they are looking at. What each particular date field means, for example.

The obvious answer to this challenge is, of course, a Data Dictionary. A Data Dictionary is a repository of detailed information about the data itself such as relationships to other data, definitions, usage, format, etc. Oracle defines it as a collection of tables with metadata. What is metadata? It is data about data. An example would be the field EMPLID. What type of field is it? Is it a character field, a numeric field? How many characters can it contain? What are valid entries into the field? The Data Dictionary gives users a place to search all of that information and more regarding the data that they use daily. In addition, the Data Dictionary also provides a place for information regarding data relationships.
To meet the need for a Data Dictionary for our Query Developers and end users to have access to the information they need regarding PeopleSoft data structure and metadata, Greg Gamble, of the Application Development team at the State Board developed a web application called metaLink. It is a .Net site on Framework 4.0, using C sharp on code pages. SQL Server is the Database. Having the web app developed in house allows us amazing flexibility.

The MetaLink app will support the college system by providing key information about the data stored within the tables and columns in Golden Gate, which are replicated from the ctcLink production database. The key information is critical to the accurate use of PeopleSoft data for reporting and/or analysis.

The key features of MetaLink allows designated users to:

- Document structures and properties for a given record or field.
- View completed data dictionary reports.
- View the ctcLink reporting catalog.
- Search for available PeopleSoft data queries.

**Logging into the metaLink Data Dictionary**

The Data Dictionary does require a login as it details PeopleSoft data structure which is proprietary. The Data Dictionary provides the ability to search records by pillar or to search our State Board Master Data which is used for our data warehouse.

In order to gain access to MetaLink you must first request a password. Please submit all requests for MetaLink passwords to: Carmen McKenzie or Paula McDaniel. cmckenzie@sbctc.edu or pmcdaniel@sbctc.edu.

Navigate to [http://dataservicesmetalink.sbctc.edu](http://dataservicesmetalink.sbctc.edu)

Click on Login in the upper right hand corner of the screen and enter in your login credentials.

You will now be able to view metadata, edit metadata or edit your account information. Click on Meta Data to go to the Reporting Library.

**Using the metaLink Data Dictionary**

To view record information, click on View Current Data Dictionary. Note: at this point in the training we are focusing on finding record and field information. The other areas of metaLink will be explained later in this class.
Select a pillar then choose to either display only common tables or only prompt tables and then select the table and click View Report. Note that record names in MetaLink are slightly different than the record names in PeopleSoft query as all record names in MetaLink begin with PS_.

There are two main sections: Table Information and column Information. Table Information holds details regarding the record while column Information holds details regarding the fields of the table.

Record Information

Record details are found under the header TABLE INFORMATION. The available fields are:

- Table Name – Name of the record.
- Alternative Table Name - The laymen name for the record. For example, the record STDNT_ENRL is commonly referred to as the Student Enrollment table.
- Table Description – Description of the record and what it is used for. Any pertinent information regarding the record.
- Business Use Description – Description of the business use of the record.
• Pillar – the Pillar the record belongs to.
• Module – the Module the record pertains to.
• Prompt Table - indicates if the record is used as a prompt table.
• Effective Dated Table - indicates if the record contains the EFFDT field.
• Commonly Used Table - indicates if the record has been flagged as commonly used.

Many of the tables only contain information on the Table Name and Pillar which is why it is so important to add information if you have it. If we look at the table BEN_HEALTH_SRCH in the HCM pillar we can see some of the great information that has been added to metoLink about the record. The information in the Table Description field of this record provides users with vital knowledge about which fields to use to create a join with another record. Without this information a user could inadvertently join on the wrong field and thus create a query with incorrect or misleading results.

Field Information

Field details are found under the header COLUMN INFORMATION. The available fields are:
• Alternative column Name - The laymen name for the column. For example, the column name STRM is commonly referred to as Session/Term.
• Description - The description of the field and what it is used for. Any pertinent information regarding the field should be included.
• Primary Key - Indicates if this field is part of the primary key for the record (table). The combination of primary key fields represent the uniqueness of the row in the record.
• Data Type - A classification identifying one of various types of data, such as integer, text or decimal, that determines the possible values for that type.
• Size - The length or scale of the field value. Size is directly related to the Data Type.
• Nullable - Indicates of the field can contain null or blank values.
• Data Classification - The WA State Office of the Chief Information Officer (OCIO) data classification. Category 1 thru 4 are used with Category 1 being public information and category 4 being confidential information requiring special handling.

• For a detailed description of each category, see the Data Classification Brief here: [http://www.sbctc.edu/resources/documents/colleges-staff/data-services/data-warehouse/DataClassificationBrief.pdf](http://www.sbctc.edu/resources/documents/colleges-staff/data-services/data-warehouse/DataClassificationBrief.pdf)

• History: Includes any historical changes to the field such as inclusions or exclusions of values.

• Accountable Commission: Designated by the Data Governance Committee and indicates which system commission is ultimately accountable for this specific data element.

• Examples of Use: Provides examples of how/when to use the field. Information about auto-joins and how they might affect the query, etc.

• Converted: The table and field name of the legacy source record if converted during implementation.

• Common column: Used only by SBCTC.

• Notes – Any other notes, pertinent information regarding the field.

• Valid Values – field values and their meanings.

You will see how the information presented could be useful. For example, when studying the tables, we see that the STDNT_GRPS table does not have an Effective Status yet the STDNT_GRPS_HIST does have Effective Status. This means that whenever you need a query to display students in an inactive status you must use the STDNT_GRPS_HIST table. Also, note the STDNT_GROUP_TBL defines student group codes making it an excellent choice for a prompt table.

We will cover the other options available in MetaLink in Section 3 of this manual.
Verifying your Query Results

With all of the time and effort that goes into creating a query one of the most important aspects of query creation comes at the very end; verifying your results. Below is a list of ways that you can ensure your query results are correct.

- **Check the Front End**
  - Select a unique identifier (EMPLID, Account Number, etc.) and go to the front end of PeopleSoft and check the information available there against the information retrieved in your query results.

- **Compare**
  - Compare your results with known reports that are up-to-date and accurate.

- **Perform a "sanity check"**
  - Ask yourself if the information you are presenting is reasonable. For example, if there are generally no classes at your institution that enroll more than 100 students per class, then a query returning a class enrollment of 250 students is probably not correct.

- **Estimate**
  - Try to estimate likely results before executing the query.

- **Break it down**
  - Break down large amounts of data to its components, and then add them to be sure that the entire amount is accounted for. For example, if searching for undergraduate Hispanic female students, look at: all Hispanic students, male Hispanic students, etc. to check the accuracy of your query.

- **Ask a SME (Subject Matter Expert)**
  - Ask someone who knows the data to check your results.

- **Get Detailed Information from the Report Requester**
  - Ask the requestor of the report what results to expect. Ask if they can suggest an existing report that you can use against which to check your results. If your query is getting results that are not anywhere near what was expected then your query may be in error.

- **Become an Expert**
  - Become as familiar and knowledgeable as you can about your data records. Participate in the Data Services WebEx’s and trainings. Also, consider using MetaLink as a source to learn as much as you can about the records you use.

- **Review the Output**
  - Reviewing the query results is always a good idea and sometimes it will help you to spot mistakes.

- **Select a single Account or EMPLID**
  - Check the data in the query against the front end of the system.
Section 1 Review

In this section we learned about the WHERE clause and ON clause and how to place criteria appropriately to a Left Outer Join and to apply effective date logic to queries and Left Outer Joins allowing understanding of how this logic can affect the results from your queries.

We also discussed important information on how to find information about where a field found in the front end of the system lives in the backend and how you can use this information in your query development.

Finally we learned how to create and use queries which search existing queries for specified information and how to check the results of your queries to ensure accurate reporting.

Notes:
Section 1 Knowledge Test

1. The **ON** clause of a Left Outer Join applies the criteria to the: __________________________

2. The **WHERE** clause applies the criteria to the: __________________________

3. Why is it important where the SQL code for criteria is placed on a Left Outer Join?

   __________________________

   __________________________

4. What three fields are used in determining effective logic in PeopleSoft query?

   __________________________   __________________________   __________________________

5. What does EFFSEQ track?

   A. Effective Date
   B. Effective Status
   C. Changes to an effective dated row that take place on the same EFFDT

6. With Effective dated queries, rows of data are classified in one of three categories. These are:

   1. __________________________

   2. __________________________

   3. __________________________

7. If creating a Left Outer Join using an Effective dated record, users must provide an option for the effective date to be ________________ in order for the Left Outer Join to function as desired.

8. If one effective dated record in a query has the effective dated criteria set a certain way, all of the other effective dated records in the query must have their effective dated criteria set up in the same way.

   True   False

9. What are 4 ways to find information about records and fields?

   1. __________________________

   2. __________________________

   3. __________________________
4. 

10. records that begin with PSQRY all contain data about ________________.

11. What are three ways to verify your query results? Choose the three that you think you will find the most beneficial.

1. ________________

2. ________________

3. ________________
Section 1 Exercises Campus Solutions

1. Create a query using the effective dated record STDNT_GRPS_HIST and update the EFFDT criteria to only see results which have an EFFDT of between the two dates – 1/1/2016 and 1/1/2017. Save your changes. Write the name of your query here: ______________________________

2. Create a new query and use the record STDNT_ATTR_TBL as your base record then add FA_APP_SRC_TBL as a Left Outer Join. Arrange the Effective Date criteria for record B to have the option for the effective date to be NULL for the second record. Save your changes. Write the name of your query here: ______________________________

3. Go to the Student Enrollment page by following the menu path Main Menu → records and Enrollment → Enroll Students → Enrollment. Enter the EMPLID 101000242. Using CNTL+SHIFT+J find the page name to use in the QCS_PSPNLFIELD query to determine the database name of the field “Reason” and what record it lives in. Write the database field name and record here: ______________________________

4. Navigate back to the Student Enrollment page by following the menu path Main Menu → records and Enrollment → Enroll Students → Enrollment. Enter the EMPLID 101000242. This time use CTRL+SHIFT+C to find out the name of the “Program” field in the database and the record where it lives. Write the database field name and record name here: ______________________________

5. Go to Query Manager and create a new query using the record PSQRYSTATS. Select all of the fields and then run the query to determine how many times query CTC_SR_PS_ID_LOOKUP has been ran and its last update date/time. Discard your changes. Write the number of times ran and last update date/time. ______________________________
Section 1 Exercises Human Capital

1. Create a query using the effective dated record ADDRESSES and update the EFFDT criteria to only see results which have an EFFDT of between the two dates – 1/1/2013 and 1/1/2014. Save your changes. Write the name of your query here:

2. Create a new query and use the record JOBCODE_TBL as your base record then add PRIMARY_JOBS_VW as a Left Outer Join. Select any fields based on your preference. Arrange the Effective Date criteria to add the option for the effective date to be NULL for the B record. Save your changes. Write the name of your query here:

3. Go to the Phone Numbers page by following the menu path Main Menu → Self Service → Personal Information → Phone Numbers. Enter the EMPLID 101004733. Using CNTL+SHIFT+J find the page name to use in the QHC_PSPNLFIELD query to determine the database name of the field “Telephone” and what record it lives in. Write the database field name and record here:

4. Go to the Phone Numbers page by following the menu path Main Menu → Self Service → Personal Information → Phone Numbers. Enter the EMPLID 101004733. This time use CTRL+SHIFT+C to find out the name of the “Benefit Plan” field in the database and the record where it lives. Write the database field name and record here:

5. Go to Query Manager and create a new query using the record PSQRYSTATS. Select all of the fields and then run the query to determine how many times query QHC_PY_PAY_EARNINGS has been ran and its last update date/time. Discard your changes. Write the number of times ran and last update date/time:
## Section 1 Exercises Financials

1. Create a query using the effective dated record ASSET_CLASS_TBL and update the EFFDT criteria to only see results which have an EFFDT of between the two dates – 1/1/2013 and 1/1/2014. Save your changes. Write the name of your query here:

2. Create a new query and use the record PO_HDR as your base record then add VENDOR_ADDR as a Left Outer Join. Arrange the Effective Date criteria to add the option for the effective date to be NULL for the second record. Save your changes. Write the name of your query here:

3. Go to the Items page by following the menu path Main Menu ➔ Items ➔ Review Item Information ➔ Items. Search using a SETID of WACTC. Select Item 005-70 Pumice Stone. Using CNTL+SHIFT+J find the page name to use in the QFS_PSPNLFIELD query to determine the database name of the field “Item ID” and what record it lives in. Write the database field name and record here:

4. Navigate back to the Items page by following the menu path Main Menu ➔ Items ➔ Review Item Information ➔ Items. Search using a SETID of WACTC. Select Item 005-70 Pumice Stone. This time use CTRL+SHIFT+C to find out the name of the “Item Group” field in the database and the record where it lives. Write the database field name and record here:

5. Go to Query Manager and create a new query using the record PSQRYSTATS. Select all of the fields and then run the query to determine how many times query CTC_WA220_KK_PRJ_DTL_EXP_REV has been ran and its last update date/time. Discard your changes. Write the number of times ran and last update date/time.
Section 2 Introduction

At the end of this section you will be able to:

- Effectively apply Aggregate Functions to query results.
- Utilize the Having tab to add criteria to fields using Aggregate Functions.
- Be able to create Expressions to allow complex refining of query results.

Aggregate Functions

An Aggregate Function is a type of operator that returns a single value based on multiple rows of data. When your query includes one or more Aggregate Functions, PeopleSoft query collects related rows and displays a single row that summarizes their contents. If, for example, you were to create a query that would include General Ledger Account and Amount Posted. You might want to see all posted activity to the account over a given amount of time. Without any Aggregate Functions, this query would return one row for every account and amount combination over the given time frame.

<table>
<thead>
<tr>
<th>Unit</th>
<th>Ledger</th>
<th>Account</th>
<th>Currency</th>
<th>Year</th>
<th>Total Amt</th>
<th>Base Curr</th>
</tr>
</thead>
<tbody>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>-665,810 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>502,210 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>310,910 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>-1311,220 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>1745,550 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>-1004,970 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>267,010 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>737,860 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>105,950 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2015</td>
<td>-312,300 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2015</td>
<td>312,300 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2015</td>
<td>23,870 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2015</td>
<td>63,150 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2015</td>
<td>-84,840 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2013</td>
<td>-306,570 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>981,740 USD</td>
<td></td>
</tr>
</tbody>
</table>

If you apply the Aggregate Function Sum to the Amount field, the query can be narrowed down to display one row that summarizes the amount for each account over the given time frame.

<table>
<thead>
<tr>
<th>Unit</th>
<th>Ledger</th>
<th>Account</th>
<th>Currency</th>
<th>Year</th>
<th>Sum Total Amt</th>
<th>Base Curr</th>
</tr>
</thead>
<tbody>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2013</td>
<td>46776567320 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2014</td>
<td>-12406565840 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2015</td>
<td>5830548110 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101110</td>
<td>USD</td>
<td>2016</td>
<td>3465696960 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101120</td>
<td>USD</td>
<td>2013</td>
<td>20600430 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101120</td>
<td>USD</td>
<td>2014</td>
<td>233865460 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101120</td>
<td>USD</td>
<td>2015</td>
<td>144044820 USD</td>
<td></td>
</tr>
<tr>
<td>WA170</td>
<td>LOCAL</td>
<td>101120</td>
<td>USD</td>
<td>2016</td>
<td>14029750 USD</td>
<td></td>
</tr>
</tbody>
</table>

Note: If you do not want PeopleSoft 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.
In the same way, if you wanted to total the amount by Fund Code/Account/Year simply add in the Fund code to the query to find the results with the Aggregate Function Sum still on the Amount field.

<table>
<thead>
<tr>
<th>Unit</th>
<th>Ledger</th>
<th>Account</th>
<th>Fund</th>
<th>Currency</th>
<th>Year</th>
<th>Sum Total Amt</th>
<th>Base Curr</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>WA170</td>
<td>101110</td>
<td>057</td>
<td>USD</td>
<td>2016</td>
<td>0.000 USD</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>WA170</td>
<td>101110</td>
<td>145</td>
<td>USD</td>
<td>2013</td>
<td>0.000 USD</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>WA170</td>
<td>101110</td>
<td>145</td>
<td>USD</td>
<td>2014</td>
<td>-13008.000 USD</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>WA170</td>
<td>101110</td>
<td>145</td>
<td>USD</td>
<td>2015</td>
<td>10.000 USD</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>WA170</td>
<td>101110</td>
<td>147</td>
<td>USD</td>
<td>2016</td>
<td>0.000 USD</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>WA170</td>
<td>101110</td>
<td>147</td>
<td>USD</td>
<td>2013</td>
<td>0.000 USD</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>WA170</td>
<td>101110</td>
<td>147</td>
<td>USD</td>
<td>2014</td>
<td>7244340.850 USD</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>WA170</td>
<td>101110</td>
<td>147</td>
<td>USD</td>
<td>2015</td>
<td>915251.100 USD</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>WA170</td>
<td>101110</td>
<td>147</td>
<td>USD</td>
<td>2016</td>
<td>0.000 USD</td>
<td></td>
</tr>
</tbody>
</table>

When you apply an Aggregate Function to a field, you are redefining how PeopleSoft query uses the field throughout the query. Essentially, the query replaces the field, wherever it occurs, with the results of the function. If you select the field as a display column, PeopleSoft query displays the aggregate values. If you use the field as an order by column, the application organizes the results in an order that is based on the aggregate values.

Aggregates cannot be used to count or sum in every type of query. If you are selecting multiple fields that will generate unique results the count/sum may not produce the results you are expecting. If the data requested is unique there is no way for the system to present the unique data and provide a count.

**Applying Aggregate Functions**

You can apply the following Aggregate Functions to a field:

- **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.

To apply an Aggregate Function, navigate to the fields Tab in your query and select the field where you would like to apply the function. Click on the Edit button to the right of the field. The Edit field Properties box will appear where you can select the Aggregate Function. Click on Ok and then run your query.

Now the query will display one row that summarizes the aggregate request.
Having criteria

SQL does not support the use of Aggregate Functions in WHERE clauses. Because the WHERE clause is where standard criteria is automatically added by PeopleSoft query this means that once you have added an Aggregate Function to a field you can no longer use that field for standard criteria. Hence, the need for Having criteria. The Having tab creates a HAVING clause where criteria are placed in the SQL code for results based on an Aggregate Function.

In SQL, a HAVING clause is similar to a WHERE clause for rows of data that have been aggregated into a single row of output. The system evaluates WHERE clauses by looking at the individual table rows before they are grouped by the Aggregate Function, and then it evaluates HAVING clauses after applying the function. So if you want to add criteria the value that is returned by the function, you must define a HAVING criterion.

When you click the Add criteria icon from the fields or query pages for an aggregate field, you will be taken directly to the Having tab instead of the criteria Tab. If you try to add criteria to the aggregated field through the Add criteria tab, the aggregated field will not display as an option from the field selection screen. You will add your criteria filters using the Having tab in the same way that you add criteria using the criteria Tab. Keep in mind that PeopleSoft query compares the result of applying the Aggregate Function to the comparison value.

Using the Having tab

Continuing with the example from the Aggregate module of this manual, we will walk through how to display only GL Accounts that have posted more than $10,000.

To add Having criteria to your query, click on the Add Criteria Icon to the right of the field with the aggregate value. You will be directed to the Having tab. Enter the values as you would normally in the criteria Tab. All options available on the criteria Tab remain available on the Having tab.

In the example to the right we are looking for values equal to or greater than 10,000 by using the “not less than” operator.

Run your query to test the results.
Expressions

Expressions are calculations that PeopleSoft query performs as part of a query. Use them when you must calculate a value that PeopleSoft query does not provide by default—for example, to add the values from two fields together or to multiply a field value by a constant. In essence, an Expression is some form of operation involving numbers, character strings, or dates and times. It can use values taken from specific fields in a table, constant values or a combination. Expressions can define constants or can derive calculations. Character string can be concatenated (linked together) into a new field. Numerical or date columns can have mathematical operations applied.

The functions in query Expressions are special types of command words written in SQL. They are usually one-word commands which manipulate data items and return a single value which can be determined by various input parameters. A typical Expression appears as in the example to the right.

You work with an Expression as if it were a field in the query, meaning you can select it to display in the output, change its column heading, or choose it as an “order by” column.

In Query Manager, you can use Expressions in two ways:
- As comparison values in selection criteria.
- As columns (fields) in the query output.

How to Write Expressions

Before learning about all the different Expression types, let’s take a look at some formatting basics on how to write Expressions. As we have already discussed, Expressions are calculations using functions. Functions are special types of command words written in SQL. You will be learning many of these command words.
(functions) later in this module but we must first learn the formatting basics of SQL Expressions before we will be able to effectively use Expressions in our queries.

Below are some of the most common delimiters used and their meanings. Understanding these delimiters and their meanings will help in comprehending Expressions.

<table>
<thead>
<tr>
<th>Delimiter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>Addition operator</td>
</tr>
<tr>
<td>'</td>
<td>Character string delimiter</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>/</td>
<td>Division operator</td>
</tr>
<tr>
<td>(</td>
<td>Expression or list delimiter (begin)</td>
</tr>
<tr>
<td>)</td>
<td>Expression or list delimiter (end)</td>
</tr>
<tr>
<td>,</td>
<td>Item separator</td>
</tr>
<tr>
<td>*</td>
<td>Multiplication operator</td>
</tr>
<tr>
<td>=</td>
<td>Relational operator (equal)</td>
</tr>
<tr>
<td>&lt;&gt;</td>
<td>Relational operator (not equal)</td>
</tr>
<tr>
<td>&lt;</td>
<td>Relational operator (less than)</td>
</tr>
<tr>
<td>&gt;</td>
<td>Relational operator (greater than)</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Relational operator (less than or equal)</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Relational operator (greater than or equal)</td>
</tr>
<tr>
<td>;</td>
<td>Statement terminator</td>
</tr>
<tr>
<td>-</td>
<td>Subtraction operator</td>
</tr>
</tbody>
</table>
Using Expressions as Comparison Values for criteria

You may use an Expression as the Expression 1 or Expression 2 Type on both the criteria Properties or Having criteria Properties pages.

- The value in the selected Expression 1 is compared to an Expression 2 you enter
- When you’ve selected Expression as the Expression 2 comparison type, as shown to the right, the Define Expression dialog box appears. In the text box, enter a valid SQL Expression.
- To use a field or Prompt in the Expression, click the Add field link or the Add Prompt link.

Note: while it is important to understand the concept of using an Expression as a Comparison Value it is not often used in query development at SBCTC.

Using Expressions as columns in query Output

Expressions can also be (and are much more commonly) used as columns in your query through the Expressions Tab.

Note: if using an existing field from your query as part of the function, the Expression Type MUST match the Format of the selected field. It is best to review the field Format of the before creating the Expression.

To see the field Format, click on the fields tab and look in the Format column. In the record ANTC_AID_TERM, note the field A. NEW_AWARD_AMT is SNm11.2 (Signed Number, 11 characters, 2 decimals) so any Expression that uses that field must have an Expression Type of Signed Number.

To add an Expression to your query, navigate to the Expressions Tab and click on “Add Expression”. Here you will select the Expression Type.

Available options are:
- Character
- Datetime
- Drilling URL
- Long Character
- Date
- Number
- Signed Number
- Time
Next, select the length and, if applicable, the decimal points for the Expression and enter in the Expression function in the Expression Text field. If the Expression contains an aggregate value such as SUM, AVG, or COUNT then select the Aggregate Function check box. Once done, click OK.

![Edit Expression Properties](image)

Note: The Aggregate Checkbox has some exceptions to the rule of selecting it if using an aggregate value. Be prepared to search for other ways to make your query return the expected results.

Click on the hyperlink “Use as field” to use this Expression in your query as a field. Note: as this calculated data is now considered a field by PeopleSoft query you are also able to add additional criteria to the field through the criteria tab (or Add criteria Icon) and edit the field properties in the fields tab.

![Query Name](image)

Once you have clicked “Use as field” you will be taken directly to the fields tab where you will be able to edit the field properties as usual. The field name will default to the actual Expression function entered in the Expression Text field. It is recommended to change this to a more user friendly field name. In this case it has been changed it to “Monthly Award”. The Record.Fieldname will remain as the Expression Text.
Expression Types

There are five general types of Expressions used in PeopleSoft query.
- Conversion
- String/Character
- Numeric
- Date and Time
- Advanced

Conversion Expressions
Conversion: Change or convert values from one data type to another (character to numeric, numeric to character, character to date or date to character)

<table>
<thead>
<tr>
<th>Function</th>
<th>Usage</th>
<th>Expression Example</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>TO_CHAR</td>
<td>Converts a number or date to a string.</td>
<td>TO_CHAR((SYSDATE),'mm-dd-yyyy')</td>
<td>01-11-2017</td>
</tr>
<tr>
<td>TO_CHAR</td>
<td>Converts a number or date to a string.</td>
<td>TO_CHAR((A.ACTION_DT),'mm-dd-yyyy')</td>
<td>June 05, 2016</td>
</tr>
<tr>
<td>TO_NUMBER</td>
<td>Converts a string to a number.</td>
<td>TO_NUMBER('12345.67','99999.99')</td>
<td>12345.67</td>
</tr>
</tbody>
</table>

String Functions
String (also referred to as Character): perform operations on a string (char/varchar) input value and return a string or numeric value.

<table>
<thead>
<tr>
<th>Function</th>
<th>Usage</th>
<th>Example</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONCATENATION</td>
<td>Concatenate two or more strings together into a single string.</td>
<td>CONCAT('Data Services', 'Rocks!')</td>
<td>Data Services Rocks!</td>
</tr>
<tr>
<td>CONCATENATION</td>
<td>Concatenate two or more strings together into a single string.</td>
<td>'Data Services'</td>
<td></td>
</tr>
<tr>
<td>INITCAP</td>
<td>Sets the first character in each word to uppercase and the rest to lowercase.</td>
<td>INITCAP(A.ACAD_PROG)</td>
<td>Acadm</td>
</tr>
<tr>
<td>LOWER</td>
<td>Converts a string to all lowercase characters.</td>
<td>LOWER(A.ACAD_PROG)</td>
<td>acadm</td>
</tr>
<tr>
<td>UPPER</td>
<td>Converts a string to all uppercase characters.</td>
<td>UPPER(A.CITY)</td>
<td>SEATTLE</td>
</tr>
<tr>
<td>SUBSTR</td>
<td>Extracts a portion of a string or field.</td>
<td>SUBSTR(A.VENDOR_ID, 7,4)</td>
<td>4116</td>
</tr>
<tr>
<td>LENGTH</td>
<td>Returns the number of characters in a string or field.</td>
<td>LENGTH(A.A.VENDOR_ID)</td>
<td>10</td>
</tr>
<tr>
<td>TRIM</td>
<td>Removes all specified characters either from the beginning of a string, the end of a string or both.</td>
<td>TRIM(' Tech ' TRIM.LEADING '000' FROM '000123') TRIM.TRAILING '1' FROM 'Tech1')</td>
<td>Tech 123</td>
</tr>
</tbody>
</table>

Page | 36
## Numeric and Date Functions

**Numeric**: performs operations on numeric values and returns numeric values, accurate to 38 decimal points

<table>
<thead>
<tr>
<th>Function</th>
<th>Usage</th>
<th>Example</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ROUND</strong></td>
<td>Function returns a number rounded to a certain number of decimal places.</td>
<td>ROUND(A.CUM_GPA,1) – starting field is 3.6689</td>
<td>3.7</td>
</tr>
<tr>
<td><strong>TRUNC</strong></td>
<td>Returns a number truncated to certain number of decimal places.</td>
<td>TRUNC(A.CUM_GPA,1) – starting field is 3.6689</td>
<td>3.6</td>
</tr>
<tr>
<td><strong>SYSDATE</strong></td>
<td>Returns the current system date and time on your local database.</td>
<td>SYSDATE - where today’s date is 01/11/2017</td>
<td>2017-01-11</td>
</tr>
<tr>
<td><strong>ADD_MONTHS</strong></td>
<td>Returns a date with a specified number of months added.</td>
<td>ADD_MONTHS(SYSDATE,3) - where today’s date is 01/11/2017</td>
<td>2017-04-11</td>
</tr>
<tr>
<td><strong>MONTHS_BETWEEN</strong></td>
<td>Returns number of months between date1 and date2.</td>
<td>MONTHS_BETWEEN(SYSDATE,A.ACTION_DT) - where today’s date is 01/11/2017 and ACTION_DT is 06/05/2012</td>
<td>55</td>
</tr>
<tr>
<td><strong>GREATEST</strong></td>
<td>Returns the greatest from a list of one or more Expressions.</td>
<td>GREATEST(A.UNT_TAKEN_PRGRSS,A.UNT_INPROG_GPA,A.TOT_INPROG_NOGPA) – where UNT_TAKEN_PRGRSS = 5.500 and UNT_INPROG_GPA = 4.500 and TOT_INPROG_NOGPA = 1.000</td>
<td>5.500</td>
</tr>
<tr>
<td><strong>LEAST</strong></td>
<td>Returns the least (the smallest) from a list of Expressions. This function is used for multiple values in the same row. See the MIN function if you want the smallest value from a group of rows.</td>
<td>LEAST(A.UNT_TAKEN_PRGRSS,A.UNT_INPROG_GPA,A.TOT_INPROG_NOGPA) – where UNT_TAKEN_PRGRSS = 5.500 and UNT_INPROG_GPA = 4.500 and TOT_INPROG_NOGPA = 1.000</td>
<td>1.000</td>
</tr>
</tbody>
</table>
Advanced Functions
Like the name implies, advanced functions are used for more advanced functionality. Advanced functions can achieve results such as returning the least or the greatest or they can be used to define if/then statements or limit the number of rows returned, to name just a few.

<table>
<thead>
<tr>
<th>Function</th>
<th>Usage</th>
<th>Example</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>NVL</td>
<td>Allows you to substitute a value when a NULL value is encountered.</td>
<td>Example 1: NVL(SUPPLIER_CITY, ‘n/a’)</td>
<td>If Supplier City is populated this will return the city name, if NULL it will return ‘n/a’.</td>
</tr>
<tr>
<td>NVL2</td>
<td>Extends the functionality found in the NVL function. It lets you substitute a value when a NULL value is encountered as well as when a non-NULL value is encountered.</td>
<td>Example 1: NVL2(SUPPLIER_CITY, 'Completed', ‘n/a’)</td>
<td>If Supplier City is populated this will return ‘Completed’, if NULL it will return ‘n/a’.</td>
</tr>
<tr>
<td>ROWNUM</td>
<td>Assigns a number indicating the order in which each row is returned by a query.</td>
<td>ROWNUM</td>
<td>2931</td>
</tr>
<tr>
<td>ROWNUM</td>
<td>Limits the number of rows returned in a result set.</td>
<td>ROWNUM &lt; 10</td>
<td>Returns 9 rows of results.</td>
</tr>
<tr>
<td>CASE</td>
<td>Performs the functionality of an “IF-THEN-ELSE” statement</td>
<td>CASE WHEN A.ACAD_CAREER = 'UGRD' THEN 'Undergraduate Student' WHEN A.ACAD_CAREER = 'GRAD' THEN 'Graduate Student' WHEN A.ACAD_CAREER = 'SRSR' THEN 'Running Start Student' ELSE 'Check Career' END</td>
<td>Undergraduate Student</td>
</tr>
</tbody>
</table>
Section 2 Review

In this section we learned how to effectively apply Aggregate Functions to query results as well as use the Having tab to add criteria to fields using Aggregate Functions.

Finally, we learned basic Expression formatting and how to create Expressions to allow complex refining of query results.

Section 2 Knowledge Test

1. If an Aggregate Function is applied to a field then query will replace the field, wherever it occurs, with the results of the function. (circle one)
   True  False

2. To apply an Aggregate Function, navigate to the _________________________ tab and select the edit button next to the selected field.

3. SQL does not support adding criteria to a field with an Aggregate Function. (circle one)
   True  False

4. In order to place criteria on the results of the Aggregate Function users should create the criteria on the _________________________ tab.

5. Expressions are _________________ that PeopleSoft query performs as part of a query.

6. In Query Manager, you can use Expressions in two ways:
   • As _________________________ in selection criteria.
   • As _________________________ in the query output.

7. What are the four Expression types?
   1. _________________________
   2. _________________________
   3. _________________________
   4. _________________________
Section 2 Exercises Campus Solutions

→ Create a new query using the record STDNT_ENRL joined with ITEM_DUE_SF via a standard join type. Select the following fields in your query.

- A.INSTITUTION - Academic Institution
- A.STDNT_ENRL_STATUS - Student Enrollment Status
- A.STATUS_DT - Status Date
- A.ENRL_ADD_DT - Enrollment Add Date
- B.SA_ID_TYPE - ID Type
- B.ITEM_NBR - Item Nbr
- B.DUE_DT - Due Date
- B.DUE_AMT - Due Amount

→ Go to the fields tab and edit the properties of the DUE_AMT field to apply the Aggregate Function SUM to this field.

→ Add criteria to the DUE_AMT Aggregate Function results to display only amounts equal to or greater $200,000 using the Add criteria icon. Note that this will take you directly to the Having tab. Hint: to create criteria of equal to or greater than use the operator “not less than”.

→ Add a prompt to your query to allow the end user to select the institution. Add in a default institution of WA220. Make sure to label the prompt accordingly.

→ Review the View SQL tab and note that there is now a WHERE clause as well as a HAVING clause.

→ Go to the Expressions tab and create a CASE Expression to show how much over $100,000 the amount due is. Also make sure to check the Aggregate box as this field has an aggregate value associated with it. Remember to use the correct terminology.

- CASE
- WHEN
- THEN
- END

→ Select to use the Expression as a field and then go to the fields tab to change the Heading to say “Amount due over $100,000”

→ Save your changes

→ Run your query
Section 2 Exercises Human Capital

→ Create a new query using the record FED_TAX_DATA joined with DIVERSITY via a standard join type. Select the following fields in your query.

- A.SPECIAL_FWT_STATUS - Special FWT Status
- A.FWT_MAR_STATUS - FWT Marital Status
- A.FWT_ADDL_AMT - FWT Additional Amount
- B.CAN_VISBL_MINORITY - Visible Minority

→ Go to the fields tab and edit the properties of the FWT_ADDL_AMT field to apply the Aggregate Function SUM to this field.

→ Add criteria to the FWT_ADDL_AMT Aggregate Function results to display only amounts equal to or greater $100 using the Add criteria icon. Note that this will take you directly to the Having tab. Hint: to create criteria of equal to or greater than use the operator “not less than”.

→ Add a prompt to your query to allow the end user to select the company. Add in a default company of 170. Make sure to label the prompt accordingly.

→ Review the View SQL tab and note that there is now a WHERE clause as well as a HAVING clause.

→ Go to the Expressions tab and create a CASE Expression to show how much over $5,000 the amount due is. Make sure to select the correct Expression Type (hint: go to the fields tab first and look in the Format column). Also make sure to check the Aggregate box as this field has an aggregate value associated with it. Remember to use the correct terminology.

- CASE
- WHEN
- THEN
- END

→ Select to use the Expression as a field and then go to the fields tab to change the Heading to say “FWT over $5000”

→ Save your changes

→ Run your query
Section 2 Exercises Financials

→ Create a new query using the record VR_BAL_AR_VW joined with VENDOR_LOC via a standard join type. Select the following fields in your query.
  • A.VENDOR_ID - Supplier ID
  • A.BUSINESS_UNIT - Business Unit
  • A.GROSS_AMT - Gross Invoice Amount
  • B.VNDR_LOC - Supplier Location
  • A.VCHR_COUNT - Count of Vouchers
  • B.EFF_STATUS - Status as of Effective Date

→ Go to the fields tab and edit the properties of the GROSS_AMT field to apply the Aggregate Function SUM to this field.

→ Add criteria to the GROSS_AMT Aggregate Function results to display only amounts equal to or greater $10000 using the Add criteria icon. Note that this will take you directly to the Having tab. Hint: to create criteria of equal to or greater than use the operator “not less than”.

→ Add a prompt to your query to allow the end user to select the business unit. Add in a default business unit of WA220. Make sure to label the prompt accordingly.

→ Review the View SQL tab and note that there is now a WHERE clause as well as a HAVING clause.

→ Go to the Expressions tab and create a CASE Expression to show how much over $10,000 the amount due is. Make sure to select the correct Expression Type (hint: go to the fields tab first and look in the Format column). Also make sure to check the Aggregate box as this field has an aggregate value associated with it. Remember to use the correct terminology.
  • CASE
  • WHEN
  • THEN
  • END

→ Select to use the Expression as a field and then go to the fields tab to change the Heading to say “Gross Invoice > $10,000”

→ Save your changes

→ Run your query
Section 3 Introduction

At the end of this section you will be able to:

- Understand BIND records and how they affect query development.
- Effectively use Metalink to search query metadata.
- Have knowledge of how to run reports created with BI Publisher.

BIND records in Population Selection queries

Population selection (Pop Select) is a feature in PeopleSoft that allows for the selection of specific individuals or organization ID’s to be selected through a query for a particular batch process based on specific parameters. For example, if an acceptance letter should be sent to all newly accepted students there is a batch process for this which uses a specific query that retrieves all new students flagged as application accepted. BIND records are used in conjunction with Pop Select processes. Each batch process will require certain fields be used in the query in order to properly execute and the corresponding BIND record contains these required fields.

What is a BIND record?

A BIND record is simply a database record that stores the required field/fields that are needed for a particular population select process to run successfully. Any and all key fields in a BIND record MUST be displayed in the query result set for the batch process associated with the query to properly execute.

Let’s take a look at a BIND record to examine in more detail. If we pull up the query: CTC_SR_MIS_NATID_POPSEL we see that there are 5 records used in the query and one of them is a BIND record (SCC_PS_STRM_BND).
If we expand the SCC_PS_STRM_BND record we see that there are 4 key fields associated to it. We just learned that any and all key fields in a BIND record must be displayed in the query result set, however in the example below only two of the key fields have been selected for output.

The reason for this is that as long as the fields are selected to be a part of the result set they can be selected from ANY record in the query. If we expand record A, STDN’T_CAR_TERM, we will see thatEMPLID and INSTITUTION, the two missing key fields of the BIND record, have been selected here. Thus we have fulfilled the requirement of displaying each and every key field from the BIND record in the query result set.

Pop Select Query Protocol

Pop Select queries have the following protocol:
1. Identify the query/Batch Process relationship. If unsure what the relationship is submit a ticket to ERP Support to ensure that the correct query is tied to the correct process.
2. All key fields in the BIND record must be displayed in the query results.
3. Only display the BIND record key fields in the query results.
   a. In some batch processes if any other column is displayed besides a BIND record key field it will cause the process to fail. If it is desired to use the Pop Select query for data values in addition to its use in the batch process it is recommended to create two separate queries.
4. Always include POP or POPSEL in the query name at the end of the query. For example, CTC_SR_MIS_NATID_POPSEL.
5. Describe the query as a Pop Select query in the description.
6. Provide a full Definition, including the batch process the query is tied to.
**MetaLink Query, Report and Pivot Grid Search**

One of the great things about MetaLink is that it allows users to search for queries, reports and pivot grids across a variety of fields from a single input source. Users are able to search across all pillars, by a single Folder or by all Folders, across Description, Definition and field name. The plethora of available options and the ease of searching is one of the best features of MetaLink. In PeopleSoft query for example, the only way to search by query definition is by creating a query to do so while MetaLink allows query definition searching with no additional effort required.

To search MetaLink for existing queries, reports and pivot grids in Production click on the Meta Data tab if you are logged into the system.

If you are not logged in as a logon is not required for this section simply click on Search Tools.

Next click on View Query, Report and Pivot Grid search hyperlink.

From here you will enter your search criteria.

It is important to note that users enter the search criteria in one page to search three different objects; queries, reports and pivot grids. However, each search criteria may not actually apply to all three objects. For
example, Modules will search reports only while Folders will only search queries. However, Pillars will search queries, reports and pivot grids. If the search criteria does not apply to a certain tab then an exhaustive list of complete inventory will display instead of a filtered list of search results. How will a user know which search criteria applies to which tabs? There is a Definitions box to the right of the search criteria which explains which search criteria affects which tab.

### Definitions
- **Pillar** - Will search Queries, Reports and Pivot Grids.
- **Module** - Will search Reports.
  - Pivot Grid and Queries will display ALL current inventory.
- **Folders** - Will search Queries.
  - Pivot Grid and Reports will display ALL current inventory.
- **Field Names** - Will search Queries and Pivot Grids.
  - Reports will display ALL current inventory.
  - Use a comma (,) as an OR separator to search multiple fields.
  - AND separator searches are not available.
- **Free Text** - Will search Queries, Reports and Pivot Grids.
  - Free Text Search allows using up to four search terms.
    - Use a comma as an OR separator to search using multiple search terms.
    - Use a single space as an AND separator to search using multiple search terms.

* All search boxes are separated by AND conditional operator logic. *

### Search Criteria Boxes

**All Pillars**

If you want to search a specific pillar you are able to select it in the “All Pillars” box. If you want to search across all three pillars simply leave “All Pillars” selected. Queries, reports and pivot grid results will be returned based on your selection.
Selecting a module will narrow the search to only return Reports based on the selected module. The search will produce results in the Queries and Pivot Grids tabs, however the results will simply be a list of all current queries and pivot grids, not a list of results based on Module. Instead, the only tab which will reflect a filtered list of results will be the Reports tab.

In the next box you are able to search for queries in either a specific folder or across all folders if “All Folders” is selected. The search will produce results in the Reports and Pivot Grids tabs, however the results will simply be a list of all current reports and pivot grids, not a list of results based on Folder. Instead, the only tab which will reflect a filtered list of results will be the Queries tab.
Field Names

In the Fields box users are able to enter field names to find results. Commas act as an “OR” separator allowing for multiple fields to be searched, however it is not possible to search multiple fields based on an "AND" logic at this time. The search will produce results in the Queries and Pivot Grids tabs, results in the Reports tab will simply be a list of all current reports, not a list of results based on the Field/s entered.
Free Text Search

The “Free Text Search” box will search across all columns in all three tabs returning any results which contains the word or words entered. Up to five terms can be separated by a comma for an "OR" logic separation or a space which acts as an "AND" separation logic.

Combine Search criteria

In the same way as the Advanced Search in PeopleSoft all of the fields used for searching in MetaLink can be combined with other fields to narrow down your search results to find specifically the queries, reports or pivot grids you are most interested in. Just make sure to keep in mind that some of the search fields are designed to only search either queries, reports or pivot grids.

Search Results

Each tab will either display results or a message saying no results found if the search applies to that tab. If the search does not apply to the tab then a complete list of current inventory will display. For example, if a search is performed on the Modules box then a complete, exhaustive list of all queries will display as the Modules search does not apply to queries as shown below.
It gets really interesting however, if another search criteria has been selected. Following the same example, let’s say that we selected the Pillar of FIN and then selected a Module. Then, the list of queries (and pivot grids) will be exhaustive and complete WITHIN the Finance module.

![Query Report & Pivot Grid Search](image)

It can be a bit confusing at first but with practice it will soon make sense!

**Queries Tab Results**

The Queries Tab will show queries that match the search criteria (or all queries as explained above). The list will include the:

- Pillar
- Folder
- Query Name
- Description
- Long Description (Definition)
The query name will be in the form of a hyperlink which can be clicked to bring up additional details about the query.

The additional details available via the hyperlink are the table/field pairs which shows which records and fields were used in the creation of the query.
Reports Tab Results

The Reports Tab will show reports that match the search criteria (or all reports as previously explained). The list will include the:

- Pillar
- Module
- Report ID
- Report Name
- Description

The report name will be in the form of a hyperlink which can be clicked to bring up additional details about the report.

The additional details available via the hyperlink are:

- Report ID
- Business Process Number
- CEMLI Number
- Report Name
- Report Description
- Report Type
- PeopleSoft Navigation
Report Details

Report ID:
FA854A

Business Process Number:
23k

CEMLI#:
No Number Available

Report Name:
Direct Lending Loans on Hold Report

Report Description:
Review and monitor loans placed on hold. Loans are placed on hold by the system when rejected origination acknowledgement, origination change acknowledgement, and disbursement acknowledgement files are received. You can also place loans on hold manually. Use after every import function.

Report Type:
BI Publisher

PeopleSoft Navigation:
Main Menu>>Financial Aid>>Loans>>Direct Lending Reconciliation>>Loans on Hold Report

Pivot Grids Tab Results

The Pivot Grids Tab will show pivot grids that match the search criteria (or all pivot grids as previously explained). The list will include the:

- Pillar
- Object Name (Pivot Grid Name)
- Description
- Long Description
The pivot grid name will be in the form of a hyperlink which can be clicked to bring up additional details about the pivot grid.

The additional details available via the hyperlink are table/field pairs which show which records and fields were used in the creation of the pivot grid.
The MetaLink Script Search allows for searching for the underlying SQL script used in the creation of Views. Users can search by View Name and/or by Pillar. Click on Meta Data to go to the Reporting Library. Click on View Script Search.

Here you will enter your search terms.

The results will display as show below.
Click on the hyperlink of the View Name to see the underlying SQL code.

Resources

While it is now possible to search commonly used tables in metaLink there is also a Resources page where full documents of commonly used tables are available by pillar.

To access these documents simply click on Resources when accessing the home page.
This brings the user to the Resources page where documents are available for download.
Adding Information to MetaLink

How is it populated? Well, while we in Data Services are definitely the responsible party for making sure it is populated, we are also harnessing the knowledge and expertise of our users... all of our users, not just Query Developers. The app is basically a wiki utilizing crowdsourcing as its source of information. We leverage our communities of practice, functional analysts, project staff... anyone who knows (or finds out something interesting) about the data is able to share their knowledge. As we gain information about data structure and business uses of different fields and records we have an obligation to all our fellow users to share it. It is quick and easy to add in information.

Click on Edit Meta Data

To update the record – select only the Pillar and record. To update a field within a record – select the Pillar, record and field you would like to modify. Type in the new information.

Scroll to the bottom of the screen when finished and click Save Data.

BI Publisher Introduction

Oracle Business Intelligence Publisher (BI Publisher, formerly XML Publisher) is an enterprise reporting solution that streamlines report and form generation. BI Publisher enables you to design and create report layout templates with the more common desktop applications of Microsoft Word and Adobe Acrobat, and renders XML data based on those templates. With a single template, it can generate reports in many formats (PDF, RTF, Excel, HTML, and so on)
in many languages. This approach to reporting can dramatically reduce report maintenance, enabling power business users to adjust report templates without involvement of IT resources.

In essence BI Publisher allows the State Board to create actual formatted reports using queries, Connected queries or other data sources and then allow the end users the option of how the report should be rendered. Users are able to select to view a report on the Web, in Excel, Word or Adobe. BI Publisher is basically a translator of data. It takes the raw format of query results and translates those results into a nicely formatted report.

The BI Publisher report naming convention prefixes will be:

- BCS - Campus Solutions
- BFS - Finance
- BHC - Human Capital Management

The rest of the BIP report name will follow the standard query naming convention.

For the purpose of this class we are going to be looking at how to view query based BI Publisher reports through query Report, how to run a BI Publisher report using Query Report Scheduler and using BI Publisher Report search.

**BI Publisher Query Report Viewer**

The BI Publisher Query Report Viewer allows end users to run query based BI Publisher reports. Note: BI Publisher reports with data sources other than query do not appear in search results and will need to be scheduled to run.

Navigate to Query Report Viewer: Main Menu → Reporting Tools → BI Publisher → Query Report Viewer

There are two search methods: Basic Search and Advanced Search.

**Basic Search**

Basic Search allows users to search by:

- Data Source ID
- Data Source Owner
- Object Owner ID
- Report Category ID
- Report Description
- Report Name
- Template Type

Wildcards % and _ are also available to use in your searches here.
“Begins With” is the only available operator in Basic Search.

Advanced Search

The Advanced Search allows for the same searching options but with different operators available. Users can combine search criteria in the Advanced Search to narrow the results returned.

Wildcards % and _ are also available to use in your searches here.

Enter your search criteria or simply hit enter for a list of all available reports.
Users are able to select how the report should be displayed in the Format column. Available options are:

- PDF
- HTM
- RTF
- XLS

Once the Format selection is made, click on “View Report” to see the report results.

BI Publisher Query Report Scheduler

The Query Report Scheduler is similar to Schedule Query in that users are able to schedule reports to run at a specific time/date or immediately. All BI Publisher reports with data sources of query or Connected query are available to be run through this tool.

Note: Do not schedule any queries or Reports to run between the hours of 1 am and 3 am as this time is reserved for IT.

The BI Publisher Query Report Scheduler is very similar in use and appearance to Schedule Query. Each report is assigned a Run Control ID which will tell the system when, where and how to run the report saving the report parameters (prompts) to the Run Control ID. Run Control ID’s are personal and are not shared across the system.

Adding a New Run Control ID

Navigate to Query Report Viewer: Main Menu → Reporting Tools → BI Publisher → Query Report Scheduler
If adding a new Run Control ID, select the Add a New Value tab. Enter the Run Control ID and click on “Add”. This will take you to the Query Report Scheduler Page.

![Query Report Scheduler](image)

On this page select the Report Name. Either type it in or click on the magnifying glass next to the Report Name field to search for your report. Once you have made the selection if there are prompts in the report you will be asked to enter in the values.

![Query Report Scheduler](image)

Note that once the Report Name is entered the remainder of the fields will fill in automatically with the exception of Template as of Date where users can select a particular version of the template they want the report to be based on. Click on Run in the upper right corner to run the report.

![Query Report Scheduler](image)
This will take you to the Process Scheduler Request page where final selections are made including the Date and Time to run the report as well as Type and Format. Once done with the selections click OK to schedule the report.

You will be directed back to the Query Report Scheduler page where you will be able to check the Process Monitor and the Report Manager for your report.

Using an Existing Run Control ID

Navigate to Query Report Viewer: Main Menu → Reporting Tools → BI Publisher → Query Report Scheduler

From the main Query Report Scheduler page you are also able to search for an existing Run Control ID.

There are two search methods: Basic Search and Advanced Search.

**BASIC SEARCH**

Basic Search allows users to search by:
- Data Source ID
- Data Source Type
• Description
• Language Code
• Report Name
• Run Control ID

Wildcards % and _ are also available to use in your searches here.

“Begins With” is the only available operator in Basic Search.

ADVANCED SEARCH

The Advanced Search allows for the same searching options but with different operators available. Users can combine search criteria in the Advanced Search to narrow the results returned.

Wildcards % and _ are also available to use in your searches here.

Enter your search criteria or simply hit enter for a list of all Run Control ID’s. Remember they are personal to you so only Run Control ID’s you have created will display here.

Click on the hyperlink of the Report you wish you run. This will take you to the Query Report Scheduler page where the report name will be filled in already. The parameters will also be saved though they can be changed by clicking on the Update Parameters hyperlink. Click on Run in the upper right corner to run the report.
This will take you to the Process Scheduler Request page where final selections are made including the Date and Time to run the report as well as Type and Format. Once done with the selections click OK to schedule the report.

You will be directed back to the Query Report Scheduler page where you will be able to check the BI Publisher Process Monitor and the Report Manager for your report.
BI Publisher Report Search

The BI Publisher Report Search will allow users to search specifically for the results of their scheduled BI Publisher reports. It is similar to Report Manager. In the example to the right the search was made for all BI Publisher Reports ran within the last 30 days.

Also note that users are able to access the BI Publisher Process Monitor and BI Publisher Report Manager from this page.

Click on the hyperlink in the Report column to go the Report Index page to view the report.

To download and/or view the report click on the Report Name in the File List box.

It is interesting to note that while there appears to be separate Process Monitors and Report Viewers for BI Publisher with different menu paths they are actually the same query Process Monitor and Report Viewer as found when scheduling standard queries. You will be able to view both BI Publisher reports and standard scheduled queries from either navigation. At the same time, only BI Publisher run Control ID’s will display when finding an existing value in BI Publisher Query Report Scheduler and only Run Control ID’s for standard queries will display in Schedule Query.
Section 3 Review

In this section we learned to understand BIND records and how they affect Population Selection query development.

We also looked at how to effectively use MetaLink to search for and create query metadata.

Finally we saw how to run reports created with BI Publisher.

Section 3 Knowledge Test

1. What is a BIND record?
   ___________________________________________________________
   ___________________________________________________________

2. Population selection (Pop Select) is a feature in PeopleSoft that allows for the selection of specific individuals or organization ID's to be selected for a particular batch process based on specific parameters.
   True False

3. Any and all _______________ _______________ in a BIND record MUST be displayed in the query result set for the batch process associated with the query to properly execute.

4. If creating a Pop Select query, how should the query be named? (What designation should appear at the end of the query name)?
   ___________________________________________________________
   ___________________________________________________________

5. MetaLink is a data dictionary provided by Data Services where users can search for _______________, _______________ and _______________.

6. MetaLink allows users to search queries using a variety of methods, some of which are not available in standard PeopleSoft query searching?
   True False

7. One of the great functions of BI Publisher is that it allows end users the option of how to view the final report. The four output options are:
   1. _______________
   2. _______________
   3. _______________
   4. _______________

8. BI Publisher Query Report Viewer only allows for viewing BI Publisher reports which have _______________ as a data source.

9. The Query Report Scheduler is similar to ________________________________.
Section 3 Exercises Campus Solutions

1. Navigate to Data Services MetaLink and click on the Meta Data tab. In the Current Data Dictionary find the field GRADING_BASIS in the PS_ACAD_CAR_TBL record. What does the Value OPT stand for in this field?

2. Going to query Search look in the CS Pillar in the Student Financials Folder for queries that have the code 890 in the definition and use the fields: ADVR_LOAD_ENRL. How many queries match these criteria?

3. Now go to Report Search and look in the CS Pillar in the Financial Aid folder for queries which have a name or description with “Student”. How many reports match these criteria?

4. When searching in MetaLink a space acts as an ___________ separator and a comma acts as an ___________ separator.

5. MetaLink allows users to search across all pillars by Folder, Description, Definition and query Name?
   True False

6. The BI Publisher report naming convention prefix for Campus Solutions will be:
   ____________

7. BI Publisher Query Report Viewer allows end users to run ____________ based BI Publisher reports.
Section 3 Exercises Human Capital

1. Navigate to Data Services MetaLink and click on the Meta Data tab. In the Current Data Dictionary find the field EMPLID_TYPE in the PS_PAY_CHECK record. What does the Value E stand for in this field?

2. Going to query Search look in the HCM Pillar in All Folders for queries that have the code 890 in the definition and use the fields: ACCOUNT. How many queries match these criteria?

3. Now go to Report Search and look in the HCM Pillar in the HR Core Objects folder for queries which have a name or description with “Salary”. How many reports match these criteria?

4. When searching in MetaLink a space acts as an ____________ separator and a comma acts as an ____________ separator.

5. MetaLink allows users to search across all pillars by Folder, Description, Definition and query Name? 
   True  False

6. The BI Publisher report naming convention prefix for Campus Solutions will be: 

7. BI Publisher Query Report Viewer allows end users to run ________________ based BI Publisher reports.
Section 3 Exercises Financials

1. Navigate to Data Services MetaLink and click on the Meta Data tab. In the Current Data Dictionary find the field VAT_ACCOUNT_FLG in the PS_GL_ACCOUNT_TBL record. What does the Value N stand for in this field?

2. Going to query Search look in the FIN Pillar in the Expenses folder for queries that have the code 890 in the definition and use the fields: ACCOUNT. How many queries match these criteria?

3. Now go to Report Search and look in the FIN Pillar in the Payables folder for queries which have a name or description with “1099”. How many reports match these criteria?

4. When searching in MetaLink a space acts as an ____________ separator and a comma acts as an ____________ separator.

5. MetaLink allows users to search across all pillars by Folder, Description, Definition and query Name?
   True    False

6. The BI Publisher report naming convention prefix for Campus Solutions will be:

   ____________

7. BI Publisher Query Report Viewer allows end users to run ________________ based BI Publisher reports.
Course Review

The goal of this course was to empower query Developers with the tools and skills needed to create and manage more complex queries.

The objectives for this course were to:

- Add criteria to a Left Outer Join.
- Be able to apply effective data logic to queries.
- Utilize queries of queries to effectively search for query data information.
- Efficiently find record and field information using the front end of PeopleSoft in conjunction with other tools.
- Gain the knowledge and skills to check your query results to be certain of query accuracy.
- Effectively apply Aggregate Functions to query results.
- Utilize the Having tab to add criteria to fields using Aggregate Functions.
- Be able to create Expressions to allow complex refining of query results.
- Understand BIND records and how they affect query development.
- Effectively use MetaLink to search and create query metadata.
- Understand how to run reports created with BI Publisher.

To meet the Goal and Objectives we learned in Section 1 about:

- The WHERE clause and ON clause and how to place criteria appropriately to a Left Outer Join and to apply effective date logic to queries allowing understanding of how this logic can affect the results from your queries.
- How to create and use queries which search existing queries for specified information. We also learned how to find record and field information using the front end of PeopleSoft and the PSChrome Extension and how you can use this information in your query development.
- How to check the results of your queries to ensure accurate reporting.

In Section 2 we learned how to:

- Effectively apply Aggregate Functions to query results as well as use the Having tab to add criteria to fields using Aggregate Functions.
- How to create Expressions to allow complex refining of query results.

In Section 3 we learned how to:

- Understand BIND records and how they affect query development.
- Effectively use MetaLink to search query metadata.
- Run reports created with BI Publisher

Now get out there and have fun developing top notch queries! 😊
**Simplified Instructions 101 Basics Skills**

Find below all of the step by step instructions provided in this manual.

**Scheduling through Query Viewer/Manager – Simplified Instructions for a New Run Control ID**

- Go to Query Manager or Query Viewer and search for the desired query and click on "Schedule " from the list of results
- The query name will automatically populate based on the query selected from Query Viewer or Query Manager.
- Click on Add New Value and enter a Run Control ID.
- Click on the “Add” button.
- If there are prompts (parameters) in the query a pop up box will appear where they can be filled in. To update existing parameters click on Update Parameters.
- Enter a description for the query Run Control ID in the Description field. This will be the Report Name.
- Click Apply
- On the Process Scheduler page select the Time Zone, Date and Time to run the query, if desired. Also change the default output Type and Format, if desired
- Click OK
- Note the Process Instance Number
- View the status of the run via Process Monitor
- View the results of the report via Report Manager

**Scheduling through Query Viewer/Manager – Simplified Instructions for an Existing Run Control ID**

- Go to Query Manager or Query Viewer and search for the desired query and click on "Schedule " from the list of results
- The query name will automatically populate based on the query selected from Query Viewer or Query Manager.
- Click on Find an Existing Value and search for the Run Control ID.
- Click on the the hyperlink for the Report Name or Run Control ID in the search results.
- If there are prompts (parameters) in the query a pop up box will appear where they can be filled in. To update existing parameters click on Update Parameters.
- Click Apply
- On the Process Scheduler page select the Time Zone, Date and Time to run the query, if desired. Also change the default output Type and Format, if desired
- Click OK
- Note the Process Instance Number
- View the status of the run via Process Monitor
- View the results of the report via Report Manager
Scheduling through Schedule Query Menu Path – Simplified Instructions for a New Run Control ID

- Navigate to Schedule Query: Main Menu → Reporting Tools → query → Schedule Query.
- Click on Add a New Value to create a new Run Control ID. Enter a Run Control ID.
- Click on the “Add” button.
- Enter the query name or search for the query.
- Enter a description for the query Run Control ID in the Description field. This will be the Report Name.
- If there are prompts (parameters) in the query a pop up box will appear where they can be filled in. To update existing parameters click on Update Parameters.
- Click Run.
- On the Process Scheduler page select the Time Zone, Date and Time to run the query, if desired.
- Also change the default output Type and Format, if desired.
- Click OK.
- Note the Process Instance Number.
- View the status of the run via Process Monitor.
- View the results of the report via Report Manager.

Scheduling through Schedule Query Menu Path – Simplified Instructions for an Existing Run Control ID

- Navigate to Schedule Query: Main Menu → Reporting Tools → query → Schedule Query.
- Click on Find an Existing Value and select the correct Run Control ID.
- If there are prompts (parameters) in the query a pop up box will appear where they can be filled in. To update existing parameters click on Update Parameters.
- Click Run.
- On the Process Scheduler page select the Time Zone, Date and Time to run the query, if desired. Also change the default output Type and Format, if desired.
- Click OK.
- Note the Process Instance Number.
- View the status of the run via Process Monitor.
- View the results of the report via Report Manager.
Creating a Simple query - Simplified Instructions

- Navigate to Query Manager.
- Click on “Create New query”.

On the records tab search for and select the record to use in the query.

On the query tab, select the fields from the selected record to use in the query.

On the fields tab, make any edits to the display of the heading text, translate values, field order or sort order.

Optional – On the criteria tab, add any desired criteria to the query.

Optional – On the Prompts tab (or from the fields tab) create any desired Prompts to use in the query.

Save Changes
Creating a Join - Simplified Instructions

• Determine the Join Method to use for creating the Join; Pre-defined, Hierarchy or Any Join.
  o Pre-Defined Joins are listed next to the field Name in the query tab.
  o Hierarchy Joins are listed next to the record Name in the query tab.
  o Any Joins are created by searching for related records in the records tab. **Any Joins are the preferred Method.**

• Create the Join by selecting the correct hyperlink.
• Select to use either a Standard Join or a Left Outer Join Type.

• If prompted, select/confirm the Auto Join criteria.

• If not prompted to Auto Join, go to the criteria tab and create the criteria to link record A to record B.

• Save changes.
Adding criteria - Simplified Instructions

- From the criteria tab click on “Add criteria” button.
- OR From the query or fields tab, click on the “Add criteria” icon.
- If not preselected, select Expression 1 Type
  
  ![Expression 1 Type](image)

  - Select the field or Expression to use as criteria (or create the New Expression) for Expression 1 depending on the prior selection.
  
  ![Expression 1](image)

  or

  ![Expression 1](image)

- Select the Condition Type.

  ![Condition Type](image)

- Select the Expression 2 Type.

  ![Expression 2 Type](image)

- Select the Expression 2 to use as criteria.

  ![Expression 2](image)

- If using a Left Outer Join, select where the criteria belongs, either the WHERE clause or on the Left Outer Join.

- Click OK.
- Select the Logical Operator for each criteria.

  ![Logical Operator](image)

- Group or Reorder the criteria as necessary.

  ![Group Criteria](image)
  ![Reorder Criteria](image)
Required Prompt with Lookup List – Simplified Instructions

To create a prompt that must be filled in and can be selected from a list of validated lookup values:
- From the fields tab click on the “Add criteria” icon.
- Select “Prompt” for Choose Expression 2 Type.
- Select Prompt Table” for the Edit Type.
- Look up the correct Prompt Table to use for the selected field Name and select it.
- Optional: change the Heading Text.
- Click “OK”.

Optional Prompt with Lookup List – Simplified Instructions

To create a prompt that can be left blank or selected from a list of lookup values:
- From the fields tab click on the “Add criteria” icon.
- Select “Prompt” for Choose Expression 2 Type.
- Click on “New Prompt”.
- Select “No Table Edit” for the Edit Type.
- Look up the correct Prompt Table to use for the selected field Name and select it.
- Optional: change the Heading Text.
- Click “OK”.
- Edit Your Optional Prompt in the criteria Tab
  - Navigate to the criteria tab and find your Prompt. Note the Prompt number.
  - Select Edit.
  - Change the selection in Choose Expression 2 Type from “Prompt” to “Expression”
  - Substituting your Prompt number for X insert the following into the Define Expression box.
    - :X OR :X = ‘ ’
  - Click “OK”.
- Make the final criteria modifications
  - Still in the “criteria” tab click “Group criteria” button.
  - Add left and right parenthesis to group the Prompt and Expression criteria:
  - Click “OK”.

Optional Prompt with No Lookup List – Simplified Instructions

To create a query where the Prompt can be left blank or typed in with no data validation (list of lookup values):
- From the fields tab click on the “Add criteria” icon.
- Select “Prompt” for Choose Expression 2 Type.
- Click on “New Prompt”.
- Select “No Table Edit” for the Edit Type.
- Click on the “Magnifying Glass” icon to select a Prompt table. Click on “No Value”.
- Optional: change the Heading Text.
Click “OK”.

**Edit Your Optional Prompt in the criteria Tab**
- Navigate to the criteria tab and find your Prompt. Note the Prompt number.
- Select Edit.
- Change the selection in *Choose Expression 2 Type* from “Prompt” to “Expression”
- Substituting your Prompt number for X insert the following into the *Define Expression* box.
  - :X OR :X = ‘ ’
- Click “OK”.

**Make the final criteria modifications**
- Still in the “criteria” tab click “Group criteria” button.
- Add left and right parenthesis to group the Prompt and Expression criteria:
- Click “Ok”.

### Date Prompt – Simplified Instructions

To create a Standard Date Prompt:

**Create the Date Prompt**
- Go to Prompts tab and click on “Add Prompt”.
- Leave the *field Name* blank.
- On *Type* drop down select *Date*.
- Optional: Type the name of the Prompt in the *Heading Text* field.
- On *Format* drop down select *None*.
- On *Edit Type* select *No Table Edit*.
- Click the ”Magnifying Glass” on the *Prompt Table* and click on the “No Value” button.
- Click the “Ok” button.

**Link the Date Prompt to the Selected field as query criteria**
- From the criteria Tab click on “Add criteria”.
- Select *field* for *Choose Expression 1 Type*.
- In Expression 1 select the Date field where to tie the Prompt by using the “Magnifying Glass” icon.
- Select the Operator.
- In “*Choose Expression 2 Type* select *Expression*.
- In Expression 2 click on “Add Prompt”.
- Select the Date Prompt previously created.
- Click “Ok”.
- Click “Save”

### Date Range Prompt – Simplified Instructions

To create a Date Range prompt where random date range can be select it is necessary to create 2 Date prompts then link them through a criteria added to the date field.

**Create the First Date Prompt**
- Go to the Prompts tab and click on “Add Prompt”.
  - Heading Text should be Text.
  - Type “From Date” in the Heading Text field.
  - Select Date on the Type drop down list.
  - On the Format drop down list, select None.
  - On Edit Type select No Table Edit.
  - For the Prompt Table click on the “Magnifying Glass” icon and click on the “No Value” button.
  - Click “Ok”.

- Create the Second Date Prompt
  - Go to the Prompts tab and click on “Add Prompt”.
  - Heading Text should be Text.
  - Type “To Date” in the Heading Text field.
  - Select Date on the Type drop down list.
  - On the Format drop down list, select None.
  - On Edit Type select No Table Edit.
  - For the Prompt Table click on the “Magnifying Glass” icon and click on the “No Value” button.
  - Click “Ok”.

- Match your Prompts to the Selected Date field and Add as query criteria
  - Go to the fields tab and identify the Date field to tie to your Prompts.
  - Click on the “Add criteria” icon.
  - As this was done from the fields tab Choose Expression 1 and Expression 1 – Choose record and field will be already filled in with the information from the selected field.
  - For Condition Type select Between from the drop down list.
  - For Choose Expression 2 Type select the Expr – Expr radio button.
  - On Expression 2 – Define Expression click “Add Prompt”.
  - Select and link the “From Date” Prompt.
  - On Expression2 – Define Expression 2 click “Add Prompt”.
  - Select and link the “To Date” Prompt.
  - “Save”.
Optional Numeric Prompt with No Lookup List – Simplified Instructions

This is a special case since the field used is numeric and the user wants to have the option to enter a numeric value without having a list to choose from or leave the field blank. This combination doesn’t work with a regular “Prompt-Expression” combination. It is necessary to convert the “Numeric” field to a “TO-CHAR” format to make this Prompt work.

• Create a new Expression in the Expression tab.
  o For Expression Type select Character from the drop down list.
  o Match the original numeric length in the Length field.
  o Enter TO_CHAR(X.FIELD_NAME) where X.FIELD_NAME is the name of the field.
  o Click “OK” button to save the new Expression.
  o Back on the Expression tab click “Use as field” link, to add the Expression as a field. Make sure to test it.

• Run your query to test the results.
  o If it runs correctly, re-name the Expression field and remove the original numeric field.

• Create the Optional Numeric Prompt
  • Back at the fields tab click the “Add criteria” icon next to the TO_CHAR field.
  • Choose Expression 1 Type and Expression 1 will be already filled in with the information from the selected field.
  • On Choose Expression 2 Type select the Prompt radio button.
  • Click “New Prompt” button to create a new Prompt.
  • Leave field Name blank.
  • Optional: Name the Prompt in the Heading Text field.
  • Match the original numeric length in the Length field.
  • Make sure Edit Type is No Table Edit.
  • Leave Prompt Table blank.
  • Click the “OK” button.

• Add as query criteria
  o Navigate to the criteria tab and find your Prompt. Note the Prompt number.
  o Select Edit.
  o Change the selection in Choose Expression 2 Type from “Prompt” to “Expression”
  o Substituting your Prompt number for X insert the following into the Define Expression box.
    ▪ :X OR :X = ‘ ’
  o Click “OK”.

• Make the final criteria modifications
  o Still in the “criteria” tab click “Group criteria” button.
  o Add left and right parenthesis to group the Prompt and Expression criteria:
    [AND] (to_char(A.STD_HOURS) equal to (? OR ? = ‘ ’)) [Edit]
  o Click “OK”.
Prompt with wildcard (%)– Simplified Instructions

This prompt allows users to select either one value or multiple values. For example, if using Union Codes the user could type in W% to search all codes beginning with W or the partial code 17% to find all codes starting with 17 or simply % to search all possible codes.

Create a new Expression in the Expression tab.
- In Expression Text enter: ‘ ’ (apostrophe-space-apostrophe).
- Click “OK”.

Add your Expression as an Optional Prompt to the query criteria
- Go to the criteria tab click the “Add criteria” button.
- Select Expression in Expression1 Type.
- Click on the "Magnifying Glass" icon to search for the desired Expression in Expression 1 – and click to select.
- Select Prompt in Choose Expression 2 Type.
- Select “New Prompt” in Expression 2 – Define Prompt to create the Optional Prompt.
- Leave the field Name blank.
- Select Text from the Heading Text drop down list.
- Type the Prompt Name in the Heading Text field. Be sure to include instructions regarding Wildcard.
- The Edit Type field should be No Table Edit.
- Click on the “Magnifying Glass” icon to select a Prompt table. Click on “No Value”.

Match the Selected field to the Optional Prompt
- Go to the fields tab and click on the “Add criteria” icon next to the field to be used for the Prompt.
- Select the Condition Type of Like.
- Select Prompt for the Choose Expression 2 Type.
- Click on the "Magnifying Glass" icon to search for the Optional Prompt and click to select it.
- Click "OK".

Make the final criteria modifications
- Navigate to the criteria Tab.
- Click the “Group criteria” button.
- Add left and right parenthesis to group both criteria:
- Click “OK”.
- Change the operator in the Logical column to Or and click "Save".

Simplified Instructions 201 Intermediate Skills

Adding criteria to a Left Outer Join – Simplified Instructions

- Using a query that has a Left Outer Join, select a field from the record which has been Left Outer Joined and add criteria to it, either from the criteria tab or the Add criteria Icon.
- In the “criteria Belongs To” box select the Outer Join clause that matches the record alias (for example, record B)
Adding EFFDT criteria to a Left Outer Join – Simplified Instructions

- Using a query that has a Left Outer Join where record B contains and EFFDT. Add additional criteria to the EFFDT field of record B where the EFFDT is NULL.
- Group both the EFFDT criteria with an OR Logical Operator.

Create a query of queries

- Create a query using one of the PSQRY tables found below.
  - PSQRYBIND – Stores all run-time prompt data
  - PSQRYBINDLANG - Related language record for PSQRYBIND
  - PSQRYCRITERIA – Stores all criteria Expressions in code format
  - PSQRYDEFN – Stores high-level query definitions with version numbers.
  - PSQRYFIELD – Stores all fields used in all aspects of query operation
  - PSQRYDEL – Stores operator ID, query name and version
  - PSQRYEXECLOG – Stores information on the execution of queries
  - PSQRYEXPR – Stores the text associated with each criteria Expression
  - PSQRYFIELD – Stores all fields used in all aspects of query operation
  - PSQRYFIELDLANG - Related language record for PSQRYFIELD
  - PSQRYLINK – Stores the relationships to child queries
  - PSQRYRECORD – Stores all records used in all aspects of query creation
  - PSQRYSELECT – Stores all SELECT requirements by select type, i.e. union, subselect, join,....
  - PSQRYSTATS – Stores run time statistics

In addition there are also records which store page or panel information.

  - PSPNLDEFN - Panel Definition
  - PSPNLDEFN_VW - Panel Definition View
  - PSPNLFIELD - Panel field
  - PSPNLFIELD_VW - Panel field View
  - PSPNLGDEFNLANG - Panel Group Related Lang
  - PSPNLGROUP - Panel Group
  - PSPNLGROUPLANG - Panel Group Related Language
  - PSPNLGRPDEFN - Panel Group Definition
  - PSPNLHTMLAREA - Panel HTML Area
  - PSPNLTREECTRL - Panel Tree Control

Create a query of queries with Definition Search Capabilities

To make the query more valuable you can make the definition field searchable by adding a wildcard prompt on the DESCRLONG (Definition) field and you can then search for queries created by a certain college code, email address or name.

Text fields are not searchable so we must convert it to a Character field. Hence, you will need to convert the DESCRLONG (Definition) field to a character field before being able to prompt with a wildcard on it. While we will
cover Expressions much more in depth later in the class, let’s see how to use them now to convert a Text field to a Character field.

On the fields tab of the query QFS_QRY_DEFINITION we see that the DESCRLONG, which is the query Definition field, is a Text field and so it must be converted to a Character field, which is searchable.

Go to the Expressions tab and click on Add Expression. In the Expression Test type in TO_CHAR(A.DESCRLONG). Make sure to update the length allowed. Click OK.

Now click on “Use as field” to use this new Expression as a field in your query.

Go to the fields tab and change the heading text of your field from TO_CHAR(A.DESCRLONG) to query Definition.
You will now be able to create a Wildcard Prompt on the Definition field. For step by step instructions on creating a Wild Card Prompt see below.

To create a Wildcard Prompt:

**CREATE A NEW EXPRESSION IN THE EXPRESSION TAB.**
- In *Expression Text* enter: ‘ ’ (apostrophe-space-apostrophe).
- Click “OK”.

**ADD YOUR EXPRESSION AS AN OPTIONAL PROMPT TO THE QUERY CRITERIA**
- Go to the criteria tab click the “Add criteria” button.
- Select *Expression* in *Expression 1 Type*.
- Click on the “Magnifying Glass” icon to search for the desired Expression in *Expression 1* – and click to select.
- Select *Prompt* in *Choose Expression 2 Type*.
- Select “New Prompt” in *Expression 2 – Define Prompt* to create the Optional Prompt.
- Leave the *field Name* blank.
- Select *Text* from the *Heading Text* drop down list.
- Type the Prompt Name in the *Heading Text* field. Be sure to include instructions regarding Wildcard.
- The *Edit Type* field should be *No Table Edit*.
- Click on the “Magnifying Glass” icon to select a Prompt table. Click on “No Value”.

**MATCH THE SELECTED FIELD TO THE OPTIONAL PROMPT**
- Go to the fields tab and click on the “Add criteria” icon next to the field to be used for the Prompt.
- Select the *Condition Type* of *Like*.
- Select *Prompt* for the *Choose Expression 2 Type*.
- Click on the “Magnifying Glass” icon to search for the Optional Prompt and click to select it.
- Click "OK".

**MAKE THE FINAL CRITERIA MODIFICATIONS**
- Navigate to the criteria Tab.
- Click the “Group criteria” button.
- Add left and right parenthesis to group both criteria:
- Click “OK”.
- Change the operator in the *Logical* column to *Or* and click "Save".

Once the query is complete you are able to search by any standard in the Definition field.
Searching record and field Information – CTRL+SHIFT+J

Note: this process is not available in the production environment.

- Navigate to the front end page where you are looking to find record and field information.
- Press CTRL+SHIFT+J to access the data sheet for that page.
  - NOTE: in Chrome the user must input CTRL+SHIFT+J TWICE for the process to work.
- Note or copy the page name and then navigate to PeopleSoft Query Manager or Query Viewer.
- Use one of the following queries as appropriate to the pillar:
  - QCS_PSPNLFIELD
  - QHC_PSPNLFIELD
  - QFS_PSPNLFIELD
- Run the QXX_PSPNLFIELD query to HTML and when prompted, enter the page name in the Prompt window then click View Results.
- The results display the names of the fields found in the page and the record where each field lives.

Searching record and field Information – CTRL+SHIFT+C

Note: this process is not available in Internet Explorer.

- Navigate to the front end page where you are looking to find record and field information.
- Press CTRL+SHIFT+C to access the information for that page.
- Hover your mouse over the desired field and look in the black box that appears above for the record and field information.

Searching record and field Information – PSCHROME Extension

- Install this extension and use it anywhere in the front end to get record and field information. Go to https://google.com/chrome/webstore from within the Chrome browser.
- Click on The Chrome Webstore hyperlink under Distribution Options.
- Search for "PeopleSoft" in the search bar.
- Click on “Add to Chrome” next to the PSChromeExtension.
- Now you are ready to start using the PSChrome Chrome Extension in PeopleSoft.

Using the PSChrome Extension Simplified Instructions

THE FIELD INFORMATION ICON

- Navigate to the front end page where you are looking to find record and field information.
- Click on the field Information icon
- Red dotted line boxes will appear around the fields
- Click inside the red dotted line box on the “I” icon then scroll to the bottom of your screen to see the resulting information. The name of the field and the value currently entered will display.

THE PAGE INFORMATION ICON

- Navigate to the front end page where you are looking to find record and field information.
- Click on the Page Information icon
- The Page information box will appear
- Note or copy the page name and then navigate to PeopleSoft Query Manager or Query Viewer.
• Use one of the following queries as appropriate to the pillar:
  o QCS_PSPNLFIELD
  o QHC_PSPNLFIELD
  o QFS_PSPNLFIELD
• Run the QXX_PSPNLFIELD query to HTML and when prompted, enter the page name in the Prompt window then click View Results.
• The results display the names of the fields found in the page and the record where each field lives.

Searching record and field Information – MetaLink
• Click on Meta Data to go to the Reporting Library.
• Click on View Current Data Dictionary.
• Select a pillar and table then click View Report.
• Table Information holds details regarding the record while column Information holds details regarding the fields of the table.

Applying Aggregate Functions
• Create a new query or edit an existing query
• Go to the fields tab
• Select the field to which to apply the Aggregate Function
• Click on the Edit box next to the selected field
• Click on the radio button corresponding to the desired Aggregate Function
• Click Ok

Applying Having criteria
• Create a new query or edit an existing query
• Go to the fields tab and select the field with the Aggregate Function to which to apply the Having criteria
• Click on the Add criteria icon
• Add the Having criteria to the field
• Click OK

Using an Expression as a column
• Create a new query or edit an existing query
• Note the field Format of the field to be used in the Expression
• Go to the Expression tab and add a new Expression
• Match the Expression Type to the field Format and if needed adjust the Length and Decimals of the Expression field
• Click the Aggregate Function box if applicable
• Enter the Expression in the Expression Text box
• Click OK
• Click the hyperlink “Use As field”
• Change the field Heading Text in the fields tab

BI Publisher Reports

• Navigate to Query Report Viewer: Main Menu → Reporting Tools → BI Publisher → Query Report Viewer
• Search for a Report
• Select the desired format in which to view the Report
• Click on View Report

Adding a new Run Control ID

• Navigate to Query Report Viewer: Main Menu → Reporting Tools → BI Publisher → Query Report Scheduler
• Click the Add a New Value tab. Enter the Run Control ID and click on “Add”.
• This will take you to the Query Report Scheduler Page.
• On this page select the Report Name. Either type it in or click on the magnifying glass next to the Report Name field to search for your report. Once you have made the selection if there are prompts in the report you will be asked to enter in the values.
• Note that once the Report Name is entered the remainder of the fields will fill in automatically with the exception of Template as of Date where users can select a particular version of the template they want the report to be based on. Click on Run in the upper right corner to run the report.
• This will take you to the Process Scheduler Request page where final selections are made including the Date and Time to run the report as well as Type and Format. Once done with the selections click OK to schedule the report.
• You will be directed back to the Query Report Scheduler page where you will be able to check the Process Monitor and the Report Manager for your report.

Using an Existing Run Control ID

• Navigate to Query Report Viewer: Main Menu → Reporting Tools → BI Publisher → Query Report Scheduler
• Search for an existing Run Control ID. To search for all Run Control ID’s simply press enter or click Search
• This will show you a list of all available Run Control ID’s.
• Click on the hyperlinked name of the desired Run Control ID.
• This will take you to the Query Report Scheduler page where the report name will be filled in already. The parameters will also be saved though they can be changed by clicking on the Update Parameters hyperlink.
• Click on Run in the upper right corner to run the report.
• This will take you to the Process Scheduler Request page where final selections are made including the Date and Time to run the report as well as Type and Format. Once done with the selections click OK to schedule the report.
• You will be directed back to the Query Report Scheduler page where you will be able to check the Process Monitor and the Report Manager for your report.
Glossary

**BIND record**: A record which binds a process which needs variable to run to a query which provides the necessary variables.

**column/field**: In a database context, a field is the same as a column. For example, a record of people could contain separate fields such as name, address, phone, etc.

**Criteria**: Specifying criteria in your query allows you to set conditions which limit the results returned by the query to only those data that you are interested in. Criteria are viewed and maintained on the “criteria” tab. Example: You may want to set criteria to limit your query to retrieve a relevant subset of data such as active undergraduate students as opposed to returning results for all active students.

**Condition Type (Edit criteria Properties Page)** – The operator used to determine how the criteria will be compared.

**Criteria Tab** - The purpose of the criteria tab is to allow users a place to create and/or view criteria that will refine the results of their queries.

**Expression 1 Type (Edit criteria Properties Page)** - Defines WHAT the criteria is being added to. Can be a field or an Expression.

**Expression 1 (Edit criteria Properties Page)** – The field or Expression the criteria is being compared to.

**Expression 2 Type (Edit criteria Properties Page)** - Defines where the VALUE the criteria is being compared against will come from. Generally from a field, Expression, Prompt, Constant or Subquery though the options can change based on the condition Type selected.

**Expression 2 (Edit criteria Properties Page)** – The Value the criteria will be compared against.

**Expressions Tab** – The purpose of the Expressions tab is to allow users a place01

**Expression Text** – Area where to write the Expression.

**Fields Tab** - The purpose of the fields tab is to show a list of all of the fields selected for the query and to determine how those fields should be displayed in the query results

**Having Tab** - The purpose of the Having tab is to allow users a place to create and/or view criteria that will refine the results of their queries for fields which have Aggregate Functions applied to them.

**Join**: The process of combining data from two or more records using matching keys.

**Primary Key**: A column in a record whose values uniquely identify the rows in the record. A primary key value cannot be NULL.

**Private query**: Private queries are only viewable by the individual who created the query. They can be shared with others individually, however.

**Prompts Tab** - The purpose of the Prompts tab is to create and display prompts

**PS query**: PeopleSoft query
Public query: Public queries are viewable and editable by any user with access to Query Manager and the proper record access. Public queries are available for use by many different users, so please do not save any changes that you make to a public query.

Query: A query is a SQL SELECT statement that reads data from records and views within the database, and returns the result set to the requester. PS queries cannot change data within the database.

Query Tab - The purpose of the query tab is to provide a location where all of the records used in the query can be displayed and where fields can be selected to use in the query.

Relational Database: A database system in which the database is organized and accessed according to the relationships between data items without the need for any consideration of physical orientation and relationship. Relationships between data items are expressed by means of tables (records).

Record/Table: records/Tables are the foundation of the query tool. A record stores data that is arranged by rows (entries) and columns (fields). For example, a record/table containing data about “people” would have a row for each individual person and columns (fields) for each piece of data stored for that individual (ex: name, address, phone). Records can be added to a query from the “records” tab.

Records Tab - The purpose of the records tab is to provide a location where users are able to search for and select one or more records to use in their queries.

Run Control ID – A unique system identifier that remembers parameters for a scheduled query.

Run Tab - The purpose of the Run tab is to allow users to view the results of the query.

SQL: Structured query Language (SQL) is a language that provides an interface to relational database systems. It was developed by IBM in the 1970s for use in System R. SQL is a de facto standard, as well as an ISO and ANSI standard. Some people pronounce SQL "sequel".

View SQL Tab - The purpose of the View SQL tab is to allow users to review the SQL statements being used in the query.
PS query 201 Training Evaluation

Please take a moment to answer the following questions. Your comments are an important contribution as we design learning experiences to meet your professional needs.

What will you do differently as a result of this training?

________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________

What do you feel were the strengths of this course?

________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________

What do you feel were the weaknesses of this course?

________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________

How can we improve this course?

________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________

What additional training-development in PS query do you require?

________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
Please rate the following statements using a 1 through 5 scale where:

1 = Agree Strongly  
5 = Disagree Strongly

<table>
<thead>
<tr>
<th>Question</th>
<th>1 (Agree Strongly)</th>
<th>2 (Agree)</th>
<th>3 (Neutral)</th>
<th>4 (Disagree)</th>
<th>5 (Strongly Disagree)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The course <strong>difficulty level</strong> was about right.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I can <strong>apply the information</strong> I learned in this course to my job.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The course met my professional educational needs.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The trainer <strong>actively involved</strong> me in the learning process.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>As a result of this training, I feel <strong>more confident</strong> in my capacity to develop queries.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>