Human Resources Management System

Query Basics (V8.9)
University of Houston System

Training Participation Guide
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query Basics (V8.19)</td>
<td>i</td>
</tr>
<tr>
<td>Chapter 1 – Introduction</td>
<td>4</td>
</tr>
<tr>
<td>Uses for Queries</td>
<td></td>
</tr>
<tr>
<td>CHAPTER 2 – Accessing a Query</td>
<td>5</td>
</tr>
<tr>
<td>Running an Existing Query from Query Manager</td>
<td>6</td>
</tr>
<tr>
<td>CHAPTER 3 - Creating a New Query</td>
<td>14</td>
</tr>
<tr>
<td>Creating a new Query</td>
<td>15</td>
</tr>
<tr>
<td>Editing Field Properties</td>
<td>21</td>
</tr>
<tr>
<td>Translate Values</td>
<td>24</td>
</tr>
<tr>
<td>Changing the Column and Output Order</td>
<td>29</td>
</tr>
<tr>
<td>CHAPTER 4 - Criteria</td>
<td>31</td>
</tr>
<tr>
<td>Setting up Criteria</td>
<td>32</td>
</tr>
<tr>
<td>Adding Criteria</td>
<td>36</td>
</tr>
<tr>
<td>Using: Equal To</td>
<td>36</td>
</tr>
<tr>
<td>Using: In List</td>
<td>39</td>
</tr>
<tr>
<td>Using: Between</td>
<td>43</td>
</tr>
<tr>
<td>Using: Like</td>
<td>44</td>
</tr>
<tr>
<td>Deleting Criteria</td>
<td>45</td>
</tr>
<tr>
<td>Boolean Expressions</td>
<td>45</td>
</tr>
<tr>
<td>Use of Parentheses</td>
<td>45</td>
</tr>
<tr>
<td>Order of Execution</td>
<td>46</td>
</tr>
<tr>
<td>CHAPTER 5 – Working With Multiple Tables and Run-Time Prompting</td>
<td>47</td>
</tr>
<tr>
<td>Multiple Tables With Run-time Prompts</td>
<td>48</td>
</tr>
<tr>
<td>Joins</td>
<td>48</td>
</tr>
<tr>
<td>CHAPTER 6 - Predefined Calculations</td>
<td>60</td>
</tr>
<tr>
<td>Aggregate Functions</td>
<td>61</td>
</tr>
</tbody>
</table>
Agenda

- Introductions
- Basic Query Functions
  - Running a Query
  - Building a Query
- Specifying Criteria
- Working with Multiple Tables, Run-time Prompts
- Predefined Calculations
Chapter 1– Introduction

Overview

The class will first introduce users to PeopleSoft’s Query Manager tool. Query Manager enables users to seamlessly create queries, which can be viewed in a grid or Excel.

The basic functions of Query will be shown and also how to create a simple query by selecting a record and specific fields. Topics will cover how to modify column headings and how to retrieve the short or long description for a translate value.

Once the basics are covered, criteria requirements such as equal to, greater than, in list, between, and like will be introduced to retrieve specific information.

Runtime prompts give the ability to enter specific values for a designated field. These values are then used as criteria for retrieving the information.

Creating queries based on multiple tables using predefined joins, calculations, such as a count, a sum, or an average will also be covered.

Objectives

Welcome to Query Basic! This class contains the tools needed to learn the concepts and procedures involved in Query. At the end of this class, participants will be able to:

- Run Existing queries
- Create queries
- View query results
- Retrieve information from effective-dated tables
- Generate runtime prompts
- Specify criteria for retrieving data
- Access information from multiple tables
- Perform predefined calculations
CHAPTER 2 – Running an Existing Query

Overview

PeopleSoft Architecture provides users a simple way to run queries using your browser.

In this chapter you will understand how to search and access an existing query using the browser.

Objectives

By the end of this chapter, the participant will be able to:

- Access a query.
- Run a query.
- Find a record name
Running an Existing Query

Step 1:
Log on to PeopleSoft. Enter your Operator ID and password. Link is My.uh.edu/hrpt
Step 2:

Select: *Reporting Tools, Query Manager*

Useful Queries:
- HR_EMPL_LIST1
- HR_EMPL_LIST1
- HR_EMPL_LIST2
- HR_EMPL_LIST1
- HR_EMPL_LIST1
- HR_EMPL_LIST1
- HR_EMPL_LIST1
- HR_EMPL_LIST1
- HR_EMPL_LIST1
- HR_EMPL_LIST1

Step 3:

The Query Manager search page will appear. Choose HR_EMPL and click on SEARCH or Enter and list of queries will appear that begin with the search criteria that you entered.
Step 4: Note the information that appears for the query. Here is what you see when you click on EDIT.

Query Description: After you click on EDIT you can view a description of the query by clicking on PROPERTIES.
Step 5: Here is what you will see when you click on PROPERTIES.

Query Name – This appears on the field QUERY
Query Description – This a brief description of the query.
Folder – Disregard for now.
Query Type – Leave as USER (default)
Owner – There are two choices Private and Public. A Private query is one that belongs only to you. A Public query is one that anyone can run and modify.
Query Definition – This can be used for a more precise description of the query.
Step 6:  Let's click on CANCEL and return back to the fields and look at some of the tabs and what they do.

Let’s look at the Query Design Elements before we run the query.

Records Page – This is where the Records/Tables are that are used in the Query.
Query Page – This are the Records/Tables that are currently being used in the Query.
Expressions Page – Use this tab to use Mathematical calculations. (Not Covered)
Prompts Page – Runtime prompts are added to the query.
Fields Page
The Fields page displays the fields used in the query.
Criteria Page - This is where you define specifically what data you want to extract using Operators (Equal to, Like, Greater than, etc).

Having Page
The Having page new criteria is created for fields that have aggregate functions associated with them (Not Covered)

View SQL Page
The View SQL page displays the SQL statement created as the query is assembled. Query uses the SQL (Structured Query Language) syntax to generate the query. This page is view only and typically is not used by end-users.
Run – Use this tab to Run the Query and view results.
**Step 7:** Let’s continue to look at the components of the Query.

**Col** – This means column. It’s where the fields will appear when the query is run.

**Record.FieldName** – This is the naming convention that PeopleSoft uses for this/these particular fields.

**Format** – This tells you what type of field it is and how many spaces are used.

**Ord** – When you see a number (like the number 1 in any of these fields), it means that when this query runs it will place the fields in numerical or alphabetical order depending on the field.

**XLAT** – This means translate. When a letter appears in this column, it means that this particular field can be translated to a more meaningful value.

**Agg** – This means aggregate. It is used to perform simple mathematical calculations.

**Heading Text** – This is the heading that you will see in the query when you run it.

**Add Criteria** – This is used to add criteria to your query.

**Edit** – This is used to edit the heading, add aggregate calculations, etc.

**Delete** – Means delete.
Step 8: Now let’s look at the list of commands under the fields.

Save – Save
Save As – Means you want to make a copy of this query and save it under a different name.
New Query – Means that you want to get out and create a new query.
Preferences – This sets up how you want to look at your Records/Tables.
Properties – See Step 5 above.
New Union – When you want to have two different sets of criteria from the same table you can use a Union (Not Covered).
Step 9: Now let’s run our query and look at the results. Click on RUN. Note the links: View All, Rerun Query, Download to Excel.
CHAPTER 3 - Creating a New Query

Overview

Some of the basic concepts and features of Query will be shown, including how to create a query from one table and manipulate data in a query.

Options for displaying the Translate Table fields: long descriptions, short descriptions, and code values will also be explored.

Objectives

By the end of this chapter, the participant will be able to:

- Create a simple one table query
- Select and manipulate fields from a specific record
Creating a new Query

The first step in creating a query is selecting a record. The selected record establishes the primary focus of a query. To create a query select the **JOB** record.

**Step 1:**
Click on **Reporting Tools, Query Manager, Create New Query**
Here is what the screen looks like when you click on Create New Query.

Where it states Record Begins With enter Job and hit Enter. Click on Add Record to add the Records/Table to the query.
Here is what the screen looks like when you add the Job Table to the query.

**Step 2:** Click Ok on the automatic criteria and select the fields to be displayed in the query.

Select: EMPID, DEPT ID, HIRE_DT, JOBCODE, and EMPL_STATUS. Select the box located to the left of each field that you wish to add to your query content. You can sort the fields by clicking on AZ link.

Example:

☑ EMPID - EmpID
To view the fields that you have selected click on Fields.

**Step 3:** Now let’s run the query and look at the results. Click on Run tab.
Step 4: Let’s name the query before we modify Field Properties. Click on the Fields tab and click on Save. Use a standard naming convention like your initials: Ex. RL_(descriptive term)

Fill in the Query name, description, leave folder blank, Query Type (accept default), Owner (public or private), and Query Definition. You only have to fill out the boxes with an *.
Now let’s modify the field properties. After you save the query, it will return to the Fields.

Here is a description of each of the tags of the Fields.

Col- Order in which the fields are displayed

Record Fieldname- Record alias and the field name in the database.

Format- the way the field is formatted in the database

Ord- Indicates whether a sort order has been selected

XLAT- Indicates if a field is translatable (N-None, S-short, L-long)

Heading Text- output column headings

Add Criteria- Click on the to add a new criteria

Edit- Click to edit field properties

Delete- deletes the field from the query
Let’s change the Column heading for HIRE_DT.

Click on Edit where the field HIRE_DT appears.

After you click on Edit for HIRE_DT this screen appears. Click on the radio button Text and enter the new heading.
Here is what the screen looks like. Click Ok. After you click Ok you will be returned to fields. Click Save to save your changes.

Now you can see that the Heading Text has been changed on your query.
Let’s Run the query to verify that our Heading has changed.
Now let’s Translate a Value. Click on Fields tab to return to the Fields. If a letter appears in the XLAT column then that field can be translated. (N-None, S-Short, L-Long). Let’s translate the EMPL_STATUS field.
Click on Edit by the EMPL_STATUS field. When you click on Edit you will see the Translate Value box.
Click on the radio button Long and click Ok. You will be returned to the Fields.

Notice the letter L in the XLAT column by field EMPL_STATUS.
Changing the Column and Output Sort Order

Here is the current Column and Output order. Let’s start by changing the column order. Click on the Column Order button.
The screen looks like this after you click on Column Order button.

Delete and Fill in the numbers as desired to change the Column Output. Let put the EMPL_ID first then the HIRE_DT. Put a 1 to the right of the field EMPL_ID and a 2 to the right of the field HIRE_DT. Click Ok and you will be returned to Fields.
Note that the columns have now changed. EMPL_ID is now in the first column and HIRE_DT is in the second column.

Now let’s sort the Output for this query. Click on Sort Order button. You will see this screen. Simply enter the number 1 by the field that you want to Sort. Let’s sort by EMPL_ID.
Here is what the screen looks like. Click OK and you will be returned to the Fields.

Note the 1 in the Ord Column by the Name field.
CHAPTER 4 - Criteria

Overview

Adding criteria rows to a query will return specific rows of data. Within these criteria rows, fields can be compared to find data of equal values, values greater or less than the field, values in a list, values in a range, and much more.

Objectives

By the end of this chapter, participants will be able to:

- Understand the various methods of creating criteria.
- Identify the three different ways to add criteria to a query.
- Explain the fields on the Edit Criteria Properties page.
- Add criteria to a query using several condition types.
- Explain how effective dates can be used as criteria.
- Explain the logic in which criteria are evaluated.
Setting up Criteria

Using Criteria, you can define the specific information you want to retrieve and filter out unwanted data. Once you decide on what field you need to establish additional criteria, you can add criteria three different ways.

1. Using the Query page.
2. Using the Fields page.
3. Using the Criteria page

Below and on the next page are examples of each of these pages.

Query Page

Move to the Query page

Notice the criteria buttons
Fields Page
Move to the Fields page
Notice the criteria buttons

Criteria Page
Move to the Criteria page
Notice the criteria buttons
Add Criteria

Pressing the Add Criteria button takes you to the Edit Criteria Properties page, where you define your criteria.
**Criteria Condition Type and Expression Combinations**

**Equal To**
The Equal To condition finds fields having a value that matches the value specified.

<table>
<thead>
<tr>
<th>Condition Type</th>
<th>Expression 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equal to</td>
<td>Constant value</td>
</tr>
<tr>
<td>Not equal to</td>
<td>Field</td>
</tr>
<tr>
<td>Greater than</td>
<td>Expression</td>
</tr>
<tr>
<td>Not greater than</td>
<td>Sub query</td>
</tr>
<tr>
<td>Less than</td>
<td>Prompt</td>
</tr>
<tr>
<td>Not less than</td>
<td>Constant Value, Prompt, Field, etc.</td>
</tr>
</tbody>
</table>

**In List**
The In List condition finds fields having a value that matches any one of the values in a list of values. With this option, a prompt allows the creation of a list, with the Edit List dialog box.

<table>
<thead>
<tr>
<th>Condition Type</th>
<th>Expression 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>In list</td>
<td>Choose from a List of Values</td>
</tr>
<tr>
<td>Not in list</td>
<td>Sub query</td>
</tr>
</tbody>
</table>

**Between**
The Between condition selects fields having a value that is between two specified values. This is an inclusive range where the upper and lower values are included in the search.

<table>
<thead>
<tr>
<th>Condition Type</th>
<th>Expression 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Between</td>
<td>Const-const</td>
</tr>
<tr>
<td></td>
<td>Const-field</td>
</tr>
<tr>
<td></td>
<td>Const-expr</td>
</tr>
<tr>
<td></td>
<td>Field-const</td>
</tr>
<tr>
<td></td>
<td>Field-field</td>
</tr>
<tr>
<td></td>
<td>Field-expr</td>
</tr>
<tr>
<td></td>
<td>Expr-const</td>
</tr>
<tr>
<td></td>
<td>Expr-field</td>
</tr>
<tr>
<td></td>
<td>Expr-expr</td>
</tr>
</tbody>
</table>

**Like**
The Like operator retrieves data containing fields that match specified portions of a character string. The Like operator is case-sensitive and uses wildcard characters to search for data.

<table>
<thead>
<tr>
<th>Condition Type</th>
<th>Expression 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Like</td>
<td>Constant (with wildcards, see below)</td>
</tr>
<tr>
<td>Not like</td>
<td>Prompt</td>
</tr>
</tbody>
</table>

**Wildcard characters:**

- % Any string of zero or more characters. For example, C% finds any string beginning with the letter C.

- - Any single character. For example, “-ones” will find any string of five characters ending with “ones” such as Jones and Cones.
**Boolean Expressions**

Boolean expressions are used to further define criteria rows. By default, an AND Boolean is added each time a new criteria expression is added. Edit Logical column of the row can change the Boolean value. Choose the value under the Logical column in the criteria page.

- AND
- AND NOT
- OR
- OR NOT

**Parentheses**

When there is more than one criteria row, parentheses can be used to control the order in which Query executes the criteria rows.

**Order of Execution**

Query processes criteria in a certain order based on what Boolean expressions are being used. Query will first execute criteria enclosed in parentheses, then NOT criteria, followed by AND criteria, and finally OR criteria.
Adding Criteria

Every row of data for the selected record may not be desired in the output of the query. By defining criteria rows, selective information will be retrieved.

Using: Equal To

Step 1:
Let’s work with the current query to define criteria using Equal To.

Step 2:
Click on the criteria funnel symbol for the field ACTION_REASON-Reason Code.
Step 3:
This will take you to a screen where you will define the criteria. In the box labeled Expression 2, Define Constant enter REH.

Step 4:
The screen should look like this, then click OK, then Run the query. Now Save your changes.
Step 5:

After clicking on Run. The results should look like this:
Using: In List

The In list operator finds fields having a value that matches any one of the values in a list of values. With this option, a prompt is presented to create a list with the Edit List dialog box. Using the same query, modify the criteria to retrieve employees that are designated as Staff or Professional employee classes.

Step 1:

Let’s go back to the query and add criteria to the query. The Employee Class is not one of the fields that we have chosen to display. We have to go and get that field from the Query tab.
Step 2:
Expand the record JOB by clicking on the + symbol. If the fields for JOB are not in alphabetical order, click on the button to sort.

Step 3:
Find the field in JOB called EMPL_CLASS, click on box to display the field in the query, and click on the criteria funnel .
Step 4:
Change the Condition Type to In List. Under Expression 2, Edit list, click on the to add values.

Step 5:
Add values A and K by clicking on the Add Value link on the right-hand side. Click Ok and then OK again.
Step 6:
You should be back at the main query where we will run the query.

Step 7:
Let’s Run and then Save the query. The results should look like this screen:

Note: The results are less than when the query was first run. All these Staff or Professional Employee classes.
Using: Between

The Between operator selects fields having a value that is between two specified values. This is an inclusive range where the upper and lower values are included in the search.

Step 1:

Locate the field, HIRE_DT from the JOB table, click on the box to display the field, and add it to the criteria page by clicking on .

Change the Operator to Between by using the drop arrow.

Leave the defaults in Choose Expression 2 Type as Const-Const.

Enter this value in Expression 1: (dates will be provided) and this value in Expression 2: (dates will be provided)
Click Ok. Click Run. Now Save the Query. The results will look like this:

Note: The results are getting smaller and smaller as the criteria get more specific.

Using: Like

**Step 1:**

Locate the field, BUSINESS_TITLE from the PER_ORG_ASGN table, click on the box to display the field, and add it to the criteria by clicking on ✅.

Change the Operator to Like by using the drop arrow

In the Expression 2, type %Ins%, and then click on OK.

The % sign is a wildcard.
Step 2:

Save and run the Query.

The result will be this:

Deleting Criteria

To delete a row of criteria, click on the button in the Delete column.

Boolean Expressions

Boolean expressions are used to further define criteria rows. The Boolean expressions used in Query include AND, OR, NOT, and parentheses.

By default, an AND Boolean is added each time new criteria is added to an expression. Click on the drop down menu to change the Boolean value.

Use of Parentheses

When there is more than one criteria row, parentheses can be used to group and control the order in which Query executes the criteria rows. In order add a parenthesis, click on the Criteria tab and then click on Group Criteria.

The following window will appear:
Order of Execution

Query processes criteria in a certain order based on what Boolean expressions are being used. Query will first execute criteria enclosed in parentheses, then NOT criteria, followed by AND criteria, and finally OR criteria.
CHAPTER 5 – Working With Multiple Tables and Run-Time Prompting

Overview

When writing queries, it is very simple to retrieve information from one table. In many cases, data will need to be retrieved from one or more tables or specify criteria in the query from a second table. In these cases, two tables will need to be linked in one query. Working with multiple tables is almost as easy as working with one.

A run-time prompt allows users to enter a value for a specific field at the time the report is executed. The report will display only those rows of information that match the value entered at the prompt.

Objectives

By the end of this chapter, participants will be able to:

- Create run-time prompts
- Know the processes involved with creating a query from multiple tables.
- Join multiple tables in a single query.
Multiple Tables With Run-time Prompts

Joins

A join enables users to retrieve data from two or more records or specify criteria from more than one record. Whenever a join is performed, the records involved are linked based on common fields.

In Query, predefined joins can be generated as a Hierarchical join or a Related Record join. Since these types of joins are predefined, criteria does not need to be added to manually link the records.

**Record Hierarchy**

A Hierarchical join uses records that are parents or children of each other. The hierarchical relationship is defined by the **Parent Record** in the Application Designer.

**Related Record**

Related Record joins use records from non-hierarchical records that are related by common fields. For example, description tables for common codes are related records. This relationship is determined by the Prompt Table edit defined for a field in the Application Designer.

Step 1:

Let's create a new query.

Choose the **JOB** record

Add record

Collapse the **JOB** (table A)

Click on the records page
Select the PERSONAL_DTA_VW and Join Record.

When you select Join Record for PERSONAL_DTA_VW you will see this screen. Accept the Join Type default (Standard Join) and click on A=JOB – EE Job History.
After you click on A=JOB – EE Job History you will get the Auto Join Criteria screen. Click on Add Criteria.

You have now joined the two tables together: the Job Table and the PERSONAL_DTA_VW table.
Step 2:
Select the fields

- From `PERSONAL_DTA_VW` (Table B) select: `EMPLID` and `NAME`

- From `JOB` (Table A) select: `DEPTID` and `POSITION_NBR`

Step 3:
Let’s create the first Prompt.
Set the Criteria by clicking on the Field’s tab, find `DEPTID` and click on the funnel 🛠️.
Leave the Condition Type as Equal To.
Choose Expression 2 Type as Prompt.
Click on New Prompt under Expression 2, Define Prompt.
Edit Prompt Properties appears.

The first time a prompt is defined in a report, a Runtime Prompt definition dialog box will be displayed.

- Verify the parameters
- Select No Table Edit in Edit Type field.
- Click OK
- Click OK on next page that appears and you are returned to the fields.

Heading Types:

- **Rft Long**: The long field name from the record definition.
- **Rft Short**: The sort field name from the record definition.
- **Text**: Any text that is desired.

Edit Types:

- **No Table Edit**: Value entered in prompt dialog box is not validated.
- **Prompt Table**: Will be picked as the default if one is defined on the record definition. Allows validation against the prompt table, which enables F4 capabilities.
- **Translate Table**: Will be picked if the field is validated against the Translate Table. Enables F4 prompt in dialog box
- **Yes/No**: Standard validation for fields represented by check boxes on panels.

If the edit type is Prompt Table, make sure the Prompt Table field is the correct record for users to prompt against.

Type, format, and length all default from the field definition in the database.
Step 4:
Here is what the criteria looks like. Run query and enter a department. Save the Query (recommendation)
INITIALS_DATE(mmd dyy)_DESCRIPTION

Step 5:
View Results.
Multiple Tables and Multiple Prompts

Step 1:

Let’s take the same query that we created above and add another table to it and also another prompt.

Let’s go to the Records Tab on the query.

Step 2: Search Page

Type DEPT as shown
Click Search
Locate DEPT_TBL - Departments. Table should be at the bottom of the list.
Click Join Record link to add record to query. Link appears on same

Note: Clicking on the link Show Fields for a Record allows you to view the fields for the record. Viewing the fields may help you determine if the record is the one you want to use in your query.
Step 3: After you select Join Record, you will see a screen that looks like this. Accept the default and click on A=JOB – EE Job History.

Step 4: Accept the Auto Join Criteria and click on Add Criteria.
Step 5:
After you click on the Add Criteria button, the tables have been joined and the query returns to the Query tab.

Step 6:
Scroll down and add the SETID_LOCATION – Location SetID field to the query from the DEPT_TBL that you just added.
Step 7:
Now let’s make this field another Prompt.

Click on Fields in the Query and find SETID_LOCATION.

Step 8:
Click on the Add Criteria funnel for the SETID_LOCATION field.

Set the Condition Type to Equal To. Set the Choose Expression 2 Type to Prompt and click on New Prompt under Expression 2, Define Prompt.
Step 9:
Choose No Table Edit under Edit Type and click OK.

Step 10:
Click OK on the next screen also and you will be returned to the Fields. Confirms the Prompt. Note where it states Expression 2, Define Prompt. You see a number two. That means that this is the second of two prompts.
Step 11:

Save and Run the Query. The screen should look like this before you enter the information of the two prompts.
CHAPTER 6 - Predefined Calculations

Overview

Aggregate simply means:
  o To gather into a mass, sum, or whole
  o To amount to; total
Instead of returning many rows of data, a count of rows or sum of a numeric field can be achieved by using the aggregate function.

Objectives

By the end of this chapter, participants will be able to:

• Use predefined aggregate functions in a query
Aggregate Functions

An aggregate function is a predefined summary calculation in Query. An aggregate function returns a single value for multiple rows of output.

| None | Sum | Count | Min | Max | Avg |

To calculate the average Annual Salary of Personnel for your department create a new query from the JOB – EE Job History table.

SUM. Here is how the function SUM works.

Create New Query

Choose the JOB – EE Job History table and select the following fields:
- DEPTID
- ANNUAL_RT

- Set the Criteria for YOUR department number.
- Click on fields and Edit the field Annual_Rt
- Click SUM in Aggregate category.
- Click OK
Some functions may be grayed-out. If so, those functions cannot be applied to the particular field selected (for example, SUM cannot be done on a character field).

The output will display the SUM of all annual rates for YOUR department.

Run and Save the Query. Click on the Fields tab to be able to see the Save. The results should look like this:

Note the word Sum in the Fields, in the Agg column.

COUNT.

Now let’s modify this query to show an example of COUNT.

- Let’s take the same query and add the field EMPLID
- Go back to the Query tab and find EMPLID and select it.
Here are the fields that we have selected for the query.

- Click on the Edit button next to EMPLID
- Change the Aggregate function to COUNT
- Click OK
The query should look like this prior to Saving and Running.

Note: Note the word COUNT in the aggregate column next to the EMPLID field.

Let’s run and save the query.

The results should look like this:

Now, you have a sum of the annual rates for all the employees in a specified Department and the count of employees in the department.
MINIMUM

Now, let’s see how Minimum works in a query.

We will use the same existing query and modify one of the fields.

Let’s click on Fields.

Let’s edit the field Annual_Rt. Instead of sum, we will change it to Min. Click OK after the change and the screen should look like this.

Note: The Agg column by field ANNUAL_RT has the abbreviation Min.
Let’s save the changes and run the query. The results should look similar to these:
MAXIMUM

Now let’s do the same thing to this field and covert it to Maximum.

Let’s edit the Annual_Rt field and chose Max.

Click Ok

Save and Run the query.

The results should look something like this:
AVERAGE

Now let’s do the same thing to this field and covert it to Average.

Let’s edit the Annual_Rt field and chose Average

Click Ok

Save and Run the query.
The results should look something like this: