Welcome!

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

- **Goal**: At the end of this section you will be able to:
  - **Look up**: Look up common tables to use in Query development.
  - **Create**: Create simple Queries.
  - **Join**: Join multiple records together to create more complex Queries.
  - **Make**: Make Queries more efficient by using Prompts.

Log in

https://csmtt92pr-oci.wsu.edu/

Query2020
Using PS Query Viewer

Main Menu > Reporting Tools > Query > Query Viewer

Search for a Query

Advanced Search
Searching Using Wildcards

Using Wildcards to search
_
 replaces a single character
%
 replaces a string of characters. It can be used in front of or in back of the search term.

Where do you start?

First thing in creating a query is that you need to find the record that holds the information that you want to extract so you can go through and you can search or you can use the PS query QuickStart guide which has a list of core tables to help you get started.
QUERY NAMING CONVENTION

Based on the WSU Data Standards, here are some suggested prefixes to use for naming your services:

- A: Admissions
- F: Financial Aid
- R: Student Records
- S: Student Financials
- V: Advising
- P: Payroll Services

WSU_A_ENROLLED_NO_ADVISOR

QUERY DEVELOPMENT

**QUERY DESCRIPTION AND DEFINITION**

**Description**

30 Characters

General description of the query

Use key words to facilitate searching

Can search in Query Viewer

**Runtime prompts**

Are your query prompts of a tabular nature? If so, you should limit the number of prompts to 20. In Query Builder, the maximum number of prompts is 35. In Query Viewer, the maximum number of prompts is 100. The best way to ensure your query is efficient is to run it and verify that the run-time is less than one minute, preferable just a few seconds.

**PUBLIC or PRIVATE Folders**

Folders can be used to categorize and organize Queries. Queries may only be stored in one folder at a time. Testing in the Test Environment is recommended.
TERMNOLOGY

Relational Database  Record/Table  Column/Field

Query  SQL  Criteria

Join  Primary Key

Creating a Simple Query

Tabs that we will be using:
Records  Fields
Query  Criteria
Prompts  View SQL
Run

Query Manager
Records

The purpose of the Records tab is to provide locations where users can search for and select one or more records to use in their queries.

Search for and add Records to your Query.

Review Record and Field Information

CTRL + SHIFT + C shortcut keys
Inspect Element
The purpose of the Query tab is to provide a location where all the records used in the query can be displayed and where fields can be selected to use in the query.

Adding the Prompt from Query tab

From the Query tab select the funnel by the field you wish to add criteria/prompt.

Edit Criteria Properties

Once the funnel has been clicked this page opens and you change the Choose Expression 2 Type from Constant to Prompt. You then can choose New Prompt or if you already have a prompt in the query choose the magnifying glass and select the prompt.
Creating Prompts Tab

Field Name – Name of field to be prompted

Name – Name of field to be prompted

Heading Type – Long or Short version of the Field Name or Aplik.

Type – Select the type of table

Heading Text – Enter the text for the Prompt Header

Prompt – Not the actual text for the prompt, but the options for the type of prompt.

Table – Select the table to be used for the prompt.

Unique Prompt Name – Field name automatically by the system and should not be changed

Length – Determine the field's length

Decimals – Number of decimals allowed for numeric prompts

Edit Type – Define the prompt type of field edit

No Table – No Data Validation (Optional)

Prompt Table – Data Validation (Required)

Translate Table – Select from a dropdown list

Yes/No Table – This will produce a yes/no prompt

Using Prompts from Criteria button

Adding Prompts through the "Add Criteria" Button automatically adds the Prompt as Criteria.

Adding Prompts through the Prompts tab will add the Prompt however, the prompt will have to be manually added as Criteria.

Prompt Tab

Table allows the Prompts tab to display and create prompts

Prompts are specific to each query and must be added for each new query.
Field Tab

The purpose of the Fields tab is to show all of the fields selected for the query and to determine how those fields should be displayed in the query results.

Change the:
- Order
- Sort Options
- Column Headers
- Select Translated Values

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.
THE PURPOSE OF THE VIEW SQL TAB IS TO ALLOW USERS TO REVIEW THE SQL STATEMENTS BEING USED IN THE QUERY.

Save Your Query
Always save your Query before running it and save frequently as you are working on it.

THE PURPOSE OF THE RUN TAB IS TO ALLOW USERS TO VIEW THE RESULTS OF THE QUERY.
If you are seeing duplicate rows in your query, you can use distinct in Properties to limit your results.

Caveat: This doesn’t work if you have unique information displaying in your rows.

Standard Joins

Standard joins only display rows where there is a match between both records, meaning it is possible not all of the rows from Record A will be displayed.
Effective Date

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, graduates, etc. Name, address, 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 (EFFDT) specifies the date on which data becomes effective.

Effective status (EFF_STATUS) indicates which row is currently active and which rows were previously active (inactive) for configuration history.

Effective sequence (EFFSEQ) tracks the sequence of changes in rows for transactional history.

Effective Data

Effective dated records are classified in one of three categories:

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.

History - The effective date is less than the current effective date.

Future - The effective date is greater than the current date.
### Effective Date Criteria

<table>
<thead>
<tr>
<th>Effective Date Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EffDate</td>
<td>Returns the one row that is closest or equal to the Expression 2 date value without exceeding it.</td>
</tr>
<tr>
<td>EffDate&lt;=</td>
<td>Returns the one row that is closest or equal to the Expression 2 date value without exceeding it.</td>
</tr>
<tr>
<td>EffDate&lt;</td>
<td>Returns the one row that is closest to the Expression 2 date value without exceeding it.</td>
</tr>
<tr>
<td>EffDate&gt;=</td>
<td>Returns the one row that is closest to the Expression 2 date value without exceeding it.</td>
</tr>
<tr>
<td>EffDate&gt;</td>
<td>Returns the one row that is closest to and greater than the Expression 2 date.</td>
</tr>
<tr>
<td>First Effective Date</td>
<td>Returns the row with the oldest effective date, usually the first row that is entered for an item.</td>
</tr>
<tr>
<td>Last Effective Date</td>
<td>Returns the row with the latest effective date, even if that date is in the future.</td>
</tr>
</tbody>
</table>

---

Create your Own Query!