PS Query Tool: From Finding Data to Writing Expressions

Session #31379
March 17, 2013
Your Presenters

• Shawn Trauntvein
• Worked nineteen years fulltime for Student Financial or Financial Aid offices.
• Nine years experience as a functional lead for Financial Aid.
  – Five years as functional lead at 4-year Public
  – Four years as functional lead at 4-year Private
BYU is a private university, with about 30,000 undergraduate students enrolled in four year programs, and about 3,000 graduate students (including a law school and business school).
BYU & PeopleSoft/Oracle

PS Application:
HRMS
9.10.00.000 &
Campus Solutions
9.00.22.000

Tools Release:
8.51.20
Overview

A entry level look at where to find the data, how to write basic queries, and then moving to some more intermediate capabilities.
Agenda/Contents

- How to start
- Where to find the data
- Writing a Query
- Joins
- Subquery
- Expressions
Navigation

Query Manager

Enter any information you have and click Search. Leave fields blank for a list of all values.

Find an Existing Query | Create New Query

*Search By: Query Name

Search Advanced Search

Find an Existing Query | Create New Query
Basic Search for a Query

• A percent symbol ‘%’ acts in a search as a wild card without length. An underscore also acts as a one position wild card.

• Queries saved as “private” will be listed first in a search, followed by those saved as “public”.

• If a folder was added, you may use that in the search also.
Basic Search for a Query

• Use basic search if you know all or part of the name.
• Most of the PeopleSoft delivered queries start with FA followed by a number that corresponds to the Crystal Report.
• For institutional queries, it is useful to use a naming convention for the first couple of characters.
You can find a report that might do something like you want. The query name will match the process name.
For some reports, there may actually be a couple of queries that are used to produce the results.
Basic Search for a Query - Institutional

If you use a naming convention, it makes it easier to do searches. In the results, private queries will be listed first, then public ones.
Basic Search for a Query

The use of folders will also allow you to refine a search.
Advanced Search for a Query

• Use advanced search if you are looking for any query where you remember part of the description or a folder it belongs to.

• You may alternately search or refine your search if you know a particular record or field used in the query.

• You may also search for only public or private queries.
In this instance a double underscore is used to look for a query that had an ‘L’ two positions after the ‘BYUFA’. Also this includes the record name and is restricted to public queries.
Records (Tables)

• You can find records that might be useful in a query in a few ways:
  – Look at a previously written query
  – Use data stored in delivered PeopleSoft records for record, page or field setup for either delivered or institutional records
  – Ask another institution for their SQL (View SQL)
  – PeopleBooks shows some records
Records (Tables)

Always test records to make sure they have data and that the data is consistent before adding to a query or to production security tree.
Once you have identified a delivered query, you can look at how it was constructed to see how the records and criteria were used.

It is a good idea if you are going to experiment with changes to always do a ‘Save As’ and give it a new name so you don’t modify the delivered report.
Delivered Query
Delivered Query

**Query Name:** FA928  
**Description:** System Generated ISIRs

<table>
<thead>
<tr>
<th>Prompt</th>
<th>Customize</th>
<th>Find</th>
<th>First</th>
<th>Last</th>
<th>Edit</th>
<th>Delete</th>
</tr>
</thead>
<tbody>
<tr>
<td>:1 = OPRID - Operator Id</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>:2 = RUN_CNTL_ID - Run Control ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
</tbody>
</table>

**Buttons:**
- Save
- Save As
- New Query
- Preferences
- Properties
- Publish as Feed
- New Union

**Return To Search**
### Delivered Query

**Query Name:** FA928

**Description:** System Generated ISIRs

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

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Format</th>
<th>Ord</th>
<th>XLAT</th>
<th>Aggr</th>
<th>Heading Text</th>
<th>Add Criteria</th>
<th>Edit</th>
<th>Delete</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPS_SCHOOL_CODE - School Code</td>
<td>Char6</td>
<td>1</td>
<td></td>
<td></td>
<td>School Cd</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>IWD_BATCH_NBR - Batch Number</td>
<td>Char23</td>
<td></td>
<td></td>
<td></td>
<td>Batch Number</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>AECQUEUEINSTANCE - EC Queue Instance</td>
<td>Num9.0</td>
<td>2</td>
<td></td>
<td></td>
<td>Queue Inst</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>AWD_AWARD_YEAR - Award Year</td>
<td>Char4</td>
<td></td>
<td></td>
<td></td>
<td>Award Year</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>IWD_CREATE_DATE - File Create Date</td>
<td>Date</td>
<td></td>
<td></td>
<td></td>
<td>Create Date</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>EMPLID - EmpID</td>
<td>Char11</td>
<td></td>
<td></td>
<td></td>
<td>ID</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>CPS_RECORD_TYPE - Record Type</td>
<td>Char2</td>
<td>N</td>
<td></td>
<td></td>
<td>Rec Type</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>IWD_STD_LAST_NAME - Last Name</td>
<td>Char16</td>
<td>3</td>
<td></td>
<td></td>
<td>Last Name</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>IWD_STD_FIRST_NM02 - Student's First Name</td>
<td>Char12</td>
<td></td>
<td></td>
<td></td>
<td>First Name</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>B.SSN - Social Security #</td>
<td>Char9</td>
<td></td>
<td></td>
<td></td>
<td>SSN</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>B.BIRTHDATE - Date of Birth</td>
<td>Date</td>
<td></td>
<td></td>
<td></td>
<td>Birthdate</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>BISIRLOAD_STATUS - Load Status</td>
<td>Char1</td>
<td>L</td>
<td></td>
<td></td>
<td>Status</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>ORIG_SSN - Original SSN</td>
<td>Char9</td>
<td></td>
<td></td>
<td></td>
<td>SSN</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>NAME_CD - Name Code</td>
<td>Char2</td>
<td></td>
<td></td>
<td></td>
<td>Name CD</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>IWD_TRANS_NBR - Transaction Number</td>
<td>Char2</td>
<td></td>
<td></td>
<td></td>
<td>Trans Nbr</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>BISIR_SEQ_NO - Sequence Number</td>
<td>Num5.0</td>
<td></td>
<td></td>
<td></td>
<td>Seq Number</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>TRANS_PROCESS_DT - Transaction Process Date</td>
<td>Date</td>
<td></td>
<td></td>
<td></td>
<td>Process Dt</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>FROM_DATE - From Date</td>
<td>Date</td>
<td></td>
<td></td>
<td></td>
<td>From</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
<tr>
<td>TO_DATE - Latest Date to be sent to PYR</td>
<td>Date</td>
<td></td>
<td></td>
<td></td>
<td>Last Date</td>
<td></td>
<td>Edit</td>
<td></td>
</tr>
</tbody>
</table>
## Delivered Query

**Query Name:** FA928  
**Description:** System Generated ISIRS

<table>
<thead>
<tr>
<th>Logical</th>
<th>Criteria</th>
<th>Expression 1</th>
<th>Condition Type</th>
<th>Expression 2</th>
<th>Customize</th>
<th>Edit</th>
<th>Delete</th>
</tr>
</thead>
<tbody>
<tr>
<td>AND</td>
<td>G.OPRID - User ID</td>
<td>equal to</td>
<td>.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>G.RUN_CNTL_ID - Run Control ID</td>
<td>equal to</td>
<td>.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>G.ISIR_LOAD_DTTM - ISIR Load Date:Time</td>
<td>not less than</td>
<td>G.FROM_DATETIME - From DateTime</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>G.ISIR_LOAD_DTTM - ISIR Load Date:Time</td>
<td>not greater than</td>
<td>G.TO_DATETIME - To DateTime</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.EC.TRANSACTIONID - EC Transaction ID</td>
<td>equal to</td>
<td>B.EC.TRANSACTIONID - EC Transaction ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.EC.QUEUEINSTANCE - EC Queue Instance</td>
<td>equal to</td>
<td>B.EC.QUEUEINSTANCE - EC Queue Instance</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.EC.TRANSACTIONOUTSW - Inbound / Outbound Switch</td>
<td>equal to</td>
<td>B.EC.TRANSACTIONOUTSW - Inbound / Outbound Switch</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>B.EC.TRANSACTIONID - EC Transaction ID</td>
<td>equal to</td>
<td>C.EC.TRANSACTIONID - EC Transaction ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>B.EC.QUEUEINSTANCE - EC Queue Instance</td>
<td>equal to</td>
<td>C.EC.QUEUEINSTANCE - EC Queue Instance</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>B.EC.TRANSACTIONOUTSW - Inbound / Outbound Switch</td>
<td>equal to</td>
<td>C.EC.TRANSACTIONOUTSW - Inbound / Outbound Switch</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>B.ISIR_SEQ_NO - Sequence Number</td>
<td>equal to</td>
<td>C.ISIR_SEQ_NO - Sequence Number</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.EC.TRANSACTIONID - EC Transaction ID</td>
<td>equal to</td>
<td>G.EC.TRANSACTIONID - EC Transaction ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.EC.TRANSACTIONID - EC Transaction ID</td>
<td>equal to</td>
<td>E.EC.TRANSACTIONID - EC Transaction ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.EC.QUEUEINSTANCE - EC Queue Instance</td>
<td>equal to</td>
<td>E.EC.QUEUEINSTANCE - EC Queue Instance</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.EC.TRANSACTIONOUTSW - Inbound / Outbound Switch</td>
<td>equal to</td>
<td>E.EC.TRANSACTIONOUTSW - Inbound / Outbound Switch</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
If you don’t want to look at each of the tabs and know some SQL, you can cut the chase and go straight to this view.
Search PeopleSoft Records

• Use what you know about a query, page, record or field to search delivered PeopleSoft records where Oracle or your DBAs have recorded data

• PeopleSoft database records that contain this data begin with “PS”.
Search PeopleSoft Records

• Can use part of a record name, a field name, or a panel (page) name or other data to search.

• Unlike searching with a query, you will need to use the ‘%’ wildcard more often—including at the end of your search parameter.
You may search for queries based on information that is stored in the database.
Search PeopleSoft Records: Query

- **PSQRYDEFN**: person who created it, date created, last person to update it, last time query was updated.
- **PSQRYEXPR**: any non-join criteria in a query.
- **PSQRYFIELD**: Record name, field name for any query (similar to advanced search).
- **PSQRYRECORD**: Record name and join types.
- **PSQRYSELECT**: record counts, field counts, criteria counts, having criteria counts.
If you know the page (panel) name you can find the records and fields used on that page by joining these two records.
Search PeopleSoft Records—Page

---

**Query Name:** BYUFA_RECORD_BY_PAGE_NAME

**Description:**
Click folder next to record to show fields. Check fields to add to query. Uncheck fields to remove from query. Add additional records by clicking the records tab. When finished click the fields tab.

**Chosen Records**

- **Alias**: PSPNLFIELD - Panel Field
  - **Record**: Hierarchy Join
- **Alias**: PSDBFIELD - Database Field
  - **Record**: Hierarchy Join

*Buttons:*
- Expand All Records
- Collapse All Records
- Save
  - Save As
  - New Query
  - Preferences
  - Properties
  - Publish as Feed
  - New Union

*Buttons (Bottom):*
- Return To Search
Search PeopleSoft Records—Page
### Financial Aid Status

- **ID:**
- **Aid Year:** Financial Aid Year 2011-2012
- **Institution:** BYU
- **National ID:**
- **Campus ID:** MAIN
- **Dependency:** Dependent w/Primary EFC
- **EFC Status:** Official
- **App Date:** 05/15/2011
- **Cor Status:** Accepted  Date: 11/30/2011

### Packaging Status Summary
- **Aid Application Status:** Active
- **Package Status:** Completed
- **SAP:** Meets SAP
- **Review Status:** Complete
- **INST Verification Status:** Select
- **Verification Flag:** Complete
- **Verification Status:** Reprocess

### Checklists

<table>
<thead>
<tr>
<th>Item Code</th>
<th>Status Date</th>
<th>Description</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>F2PTAX</td>
<td>11/23/2011</td>
<td>Parent(s) 2010 Tax Return</td>
<td>Completed</td>
</tr>
<tr>
<td>F2STAX</td>
<td>11/23/2011</td>
<td>Student's 2010 Tax Return</td>
<td>Completed</td>
</tr>
<tr>
<td>F2IVFD</td>
<td>11/23/2011</td>
<td>Dependent Verification Form</td>
<td>Completed</td>
</tr>
<tr>
<td>F2OASM</td>
<td>05/19/2011</td>
<td>Self-Reported Aid Summary</td>
<td>Completed</td>
</tr>
</tbody>
</table>

### Communications

<table>
<thead>
<tr>
<th>Method</th>
<th>Date Completed</th>
<th>Direction</th>
<th>Letter Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-Mail</td>
<td>05/17/2011</td>
<td>Outgoing</td>
<td>F03</td>
<td>Need Institutional App</td>
</tr>
<tr>
<td>E-Mail</td>
<td>05/18/2011</td>
<td>Outgoing</td>
<td>F03</td>
<td>Need Institutional App</td>
</tr>
<tr>
<td>E-Mail</td>
<td>05/18/2011</td>
<td>Outgoing</td>
<td>F03</td>
<td>Need Institutional App</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Browser</td>
<td>IE/7.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Operating System</td>
<td>WINXP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Browser Compression</td>
<td>ON (gzip)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tools Release</td>
<td>8.51.12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Release</td>
<td>HRMS and Campus Solutions 9.00.22.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service Pack</td>
<td>22</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Page</td>
<td>SS_FA_AU_STATUS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Component</td>
<td>SS_FA_AU_STATUS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Menu</td>
<td>PACKAGE_AID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Database Name</td>
<td>SAPRD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Database Type</td>
<td>ORACLE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Server</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Component Buffer Size (KB)</td>
<td>535</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

continue
Search PeopleSoft Records—Page

BYUFA_RECORD_BY_PAGE_NAME

Page Name: SS_FA_AU_STATUS

OK  Cancel
### Search PeopleSoft Records—Page

**Page Name:** SS_FA_AU_STATUS

<table>
<thead>
<tr>
<th>Occurs Lvl</th>
<th>Record</th>
<th>Field Name</th>
<th>Label Text</th>
<th>Length</th>
<th>Dec Posns</th>
<th>Field Nbr</th>
<th>Field ID</th>
<th>Panel Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SCC_PERS_SA_VW</td>
<td>FERPA</td>
<td>FERPA</td>
<td>1</td>
<td>0</td>
<td>12</td>
<td>12</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>2</td>
<td>STDN_TAD_SRCH</td>
<td>NAME</td>
<td>Name</td>
<td>50</td>
<td>0</td>
<td>13</td>
<td>13</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>3</td>
<td>STDN_TAD_SRCH</td>
<td>EMPLID</td>
<td>ID</td>
<td>11</td>
<td>0</td>
<td>14</td>
<td>14</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>4</td>
<td>STDN_TAD_SRCH</td>
<td>AID_YEAR</td>
<td>Aid Year</td>
<td>4</td>
<td>0</td>
<td>15</td>
<td>15</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>5</td>
<td>AID_YEAR_TBL</td>
<td>DESCR</td>
<td>Aid Year Description</td>
<td>30</td>
<td>0</td>
<td>16</td>
<td>16</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>6</td>
<td>STDN_TAD_SRCH</td>
<td>INSTITUTION</td>
<td>Institution</td>
<td>5</td>
<td>0</td>
<td>17</td>
<td>17</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>7</td>
<td>STDN_TAD_SRCH</td>
<td>NATIONAL_ID</td>
<td>National ID</td>
<td>20</td>
<td>0</td>
<td>19</td>
<td>19</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>8</td>
<td>ISIR_CNTRL_VW</td>
<td>DT_APP_RECEIVED</td>
<td>App Date</td>
<td>10</td>
<td>0</td>
<td>20</td>
<td>20</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>9</td>
<td>ISIR_CNTRL_VW</td>
<td>CORRECTN_STATUS</td>
<td>Cor Status</td>
<td>1</td>
<td>0</td>
<td>21</td>
<td>21</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>10</td>
<td>ISIR_CNTRL_VW</td>
<td>CORR_STAT_DT</td>
<td>Date</td>
<td>10</td>
<td>0</td>
<td>22</td>
<td>22</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>11</td>
<td>DERIVED_SA</td>
<td>STDN_TAD_PKG_FB3</td>
<td>TERM</td>
<td>1</td>
<td>0</td>
<td>23</td>
<td>23</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>12</td>
<td>DERIVED_FA_LBPL</td>
<td>FA_ISIR_INFO_LBL</td>
<td>ISIR Information</td>
<td>1</td>
<td>0</td>
<td>24</td>
<td>24</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>13</td>
<td>DERIVED_FA_LBPL</td>
<td>FA_DB_MATCHES_LBL</td>
<td>Database Matches</td>
<td>1</td>
<td>0</td>
<td>25</td>
<td>25</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>14</td>
<td>DERIVED_FA_LBPL</td>
<td>FA_AWARD_LBL</td>
<td>Financial Aid Awarded</td>
<td>1</td>
<td>0</td>
<td>26</td>
<td>26</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>15</td>
<td>DERIVED_FA_LBPL</td>
<td>FA_NEED_SMRY_LBL</td>
<td>Need Summary</td>
<td>1</td>
<td>0</td>
<td>27</td>
<td>27</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>16</td>
<td>DERIVED_FA_AWD</td>
<td>AWARD_NOTIF_BTN</td>
<td>Pell Information</td>
<td>1</td>
<td>0</td>
<td>28</td>
<td>28</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>17</td>
<td>DERIVED_FA_LBPL</td>
<td>FA_PELL_LBL</td>
<td>Pell Information</td>
<td>1</td>
<td>0</td>
<td>29</td>
<td>29</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>18</td>
<td>DERIVED_FA_SS_A</td>
<td>FA_ADMIN_REQ_LBL</td>
<td>Administrative Request</td>
<td>1</td>
<td>0</td>
<td>30</td>
<td>30</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>19</td>
<td>DERIVED_FA_SS_A</td>
<td>CURRENCY_CD</td>
<td>Currency Code</td>
<td>3</td>
<td>0</td>
<td>31</td>
<td>31</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>20</td>
<td>DERIVED_FA_AWD</td>
<td>EMPLID</td>
<td>ID</td>
<td>11</td>
<td>0</td>
<td>32</td>
<td>32</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>21</td>
<td>DERIVED_FA_AWD</td>
<td>INSTITUTION</td>
<td></td>
<td>5</td>
<td>0</td>
<td>33</td>
<td>33</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>22</td>
<td>DERIVED_FA_AWD</td>
<td>AID_YEAR</td>
<td></td>
<td>4</td>
<td>0</td>
<td>34</td>
<td>34</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>23</td>
<td>STDN_AWD_PER</td>
<td>EFC_STATUS</td>
<td>EFC Status</td>
<td>10</td>
<td>0</td>
<td>35</td>
<td>35</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>24</td>
<td>PSXLATITEM</td>
<td>XLAT_SHORTNAME</td>
<td>Translate Short Name</td>
<td>10</td>
<td>0</td>
<td>36</td>
<td>36</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>25</td>
<td>STDN_AWD_ATR</td>
<td>SFL_REVIEW_STATUS</td>
<td>Review Status</td>
<td>1</td>
<td>0</td>
<td>37</td>
<td>37</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>26</td>
<td>STDN_AWD_ATR</td>
<td>QVRIF_SELECT</td>
<td>INST Verification Status</td>
<td>1</td>
<td>0</td>
<td>38</td>
<td>38</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>27</td>
<td>ISIR_CNTRL_VW</td>
<td>DEPNDNCY_STAT</td>
<td>Dependency</td>
<td>1</td>
<td>0</td>
<td>39</td>
<td>39</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>28</td>
<td>STDN_AWD_ATR</td>
<td>VERIFICATION_STATUS</td>
<td>Verification Flag</td>
<td>1</td>
<td>0</td>
<td>40</td>
<td>40</td>
<td>SS_FA_AU_STATUS</td>
</tr>
<tr>
<td>29</td>
<td>STDN_AWD_ATR</td>
<td>VERIFICATION_STATUS_CODE</td>
<td>Verification Status</td>
<td>1</td>
<td>0</td>
<td>41</td>
<td>41</td>
<td>SS_FA_AU_STATUS</td>
</tr>
</tbody>
</table>
Search PeopleSoft Records: Record and Field Setup

If you know part of the record name, or the field name, you may use one or both of these records to find all of the matching records.
Search PeopleSoft Records—Record
If you know part of the record name, you may search for all of the records that are similarly named. You must use a % at the end also, if there may be anything after.
### Search PeopleSoft Records—Record

<table>
<thead>
<tr>
<th>Record</th>
<th>Fld</th>
<th>Record Use</th>
<th>Rec Type</th>
<th>Rec Descr</th>
<th>Parent Rec</th>
<th>SQL Table</th>
<th>Object Owner ID</th>
<th>Descr</th>
</tr>
</thead>
<tbody>
<tr>
<td>SFA_CODEDIT_LNG</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>COD Edit Codes</td>
<td>SFA_COD_EDIT</td>
<td>SFA</td>
<td></td>
<td>Related Language record for COD Edit Codes by Aid Year.</td>
</tr>
<tr>
<td>SFA_COD_ATT_SCHL</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>COD Attended School</td>
<td>SFA_COD_RPT_SCHL</td>
<td>SFA</td>
<td></td>
<td>Contains Attended School Entity Identification Information.</td>
</tr>
<tr>
<td>SFA_COD_ATT_RSP</td>
<td>9</td>
<td>0</td>
<td>0</td>
<td>COD Attended School Response</td>
<td>SFA_COD_ATT_SCHL</td>
<td>SFA</td>
<td></td>
<td>Contains warnings and error codes related to the COD document.</td>
</tr>
<tr>
<td>SFA_COD_AWD_RSP</td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>COD Award Response</td>
<td>SFA_COD_AWARD</td>
<td>SFA</td>
<td></td>
<td>Contains warnings and error codes related to the award.</td>
</tr>
<tr>
<td>SFA_COD_BOR_PHN</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>COD Borrower Phone Information</td>
<td>SFA_COD_BORROWR</td>
<td>SFA</td>
<td></td>
<td>Contains Borrower Phone Information.</td>
</tr>
<tr>
<td>SFA_COD_BOR_RSP</td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>COD Borrower Response</td>
<td>SFA_COD_BORROWR</td>
<td>SFA</td>
<td></td>
<td>Contains warnings and error codes related to the borrower for the award.</td>
</tr>
<tr>
<td>SFA_COD_DOCMTNT</td>
<td>18</td>
<td>0</td>
<td>0</td>
<td>COD Document Information</td>
<td>SFA_COD_DOCMTNT</td>
<td>SFA</td>
<td></td>
<td>Parent Table to all other Common Origination and Disbursement Tables. Contains general information about the COD document.</td>
</tr>
<tr>
<td>SFA_COD_INB_AET</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>Publish AE Cache Record</td>
<td>SFA_COD_INB_AET</td>
<td>SFAAG</td>
<td></td>
<td>Contains warnings and error codes related to the Loan Information.</td>
</tr>
<tr>
<td>SFA_COD_IN_RSP</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>COD Loan Information Response</td>
<td>SFA_COD_IN_RSP</td>
<td>SFA</td>
<td></td>
<td>Contains warnings and error codes related to the Loan Information.</td>
</tr>
<tr>
<td>SFA_COD_LOOKUP</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>COD Lookup Table</td>
<td>SFA_COD_LOOKUP</td>
<td>SFA_L</td>
<td></td>
<td>Contains cross reference information. Links COD tag names to PS COD identifiers.</td>
</tr>
<tr>
<td>SFA_COD_OUT_AET</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>Publish AE Cache Record</td>
<td>SFA_COD_OUT_AET</td>
<td>SFAAG</td>
<td></td>
<td>Contains warnings and error codes related to the Coding document.</td>
</tr>
<tr>
<td>SFA_COD_DOC_RSP</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>COD Document Response</td>
<td>SFA_COD_DOCMTNT</td>
<td>SFA</td>
<td></td>
<td>Contains warnings and error codes related to the COD document.</td>
</tr>
<tr>
<td>SFA_COD_DISB_RSP</td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>COD Disbursement Response</td>
<td>SFA_COD_DISB</td>
<td>SFA</td>
<td></td>
<td>Contains warnings and error codes related to the disbursement.</td>
</tr>
<tr>
<td>SFA_COD_EDIT</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>COD Edit Codes</td>
<td>SFA_COD_EDIT</td>
<td>SFA</td>
<td></td>
<td>This table holds COD Edit Codes by Aid Year.</td>
</tr>
</tbody>
</table>
Search PeopleSoft Records

Once you find a record you may want to use, search to see if it is already in your query security. If not, you will need to have it added to your query tree, and then test whether or not it will meet your needs.
BYU has a naming convention to start all records created by our IT with a ‘Y’. Using the institutional naming convention for records, you can also use that to search for useful records.
Other Tools

There are other tools that may help locate records:

- PS Web Spy Lite (application)
- View Source Code in Internet Explorer
- Tools > Web Developer > Inspect (Firefox)
- Right click > Inspect Element (Chrome)
- Application Designer
- There are always other ways.
Cautions

There are of cautions when locating records through data:

• Sometimes the records that show are not reliable. One example is that STDNT_AWD_PKG may show as the source, but that actually is not a reliable record for award data.

• It takes time to look, it is better to have an idea of what records to use.
Frequently Used Records in Financial Aid

There are two types of records we will focus on:

• Student data records—use to analyze activity from ISIR load to budgets, awarding and disbursing.

• Setup records—use to check setup during aid year rollover and bundle testing. Check for errors or missing data.
Frequently Used—Student Aid Year

- **STUDENT_AID**: Record that stores Aid Year Activation for a student. When a student is Activated for an aid year, and entry is made in this record and in STDNT_AID_ATRBT. There is one row of data per year. A student can be aid year activated when an ISIR loads, by institutional batch process, or manually.

- **STDNT_AID_ATRBT**: Record shows packaging status summary data, including SAP status, verification status, database match flags, career (but be careful because only shows if awarded), and other data related to eligibility. There is one row of data per aid year and the record is created during aid year activation.
ISIR Suspense Records (contain data for ISIRs that have been inbounded (whether or not loaded):

- **ISIR_00_1_EC**: Contains student and Parent Data, also EMPLID if a match was found.
- **ISIR_00_2_EC**: Contains dependency, primary EFC (and other calculations) and Database matches.
- **ISIR_00_3_EC**: Contains NSLDS data from ISIR.
- **ISIR_00_HEAD_EC**: Contains Award Year, Record Type and Create Date.
- **ISIR_00_TRAL_EC**: Similar to header. (These only have EMPLID if match in ISIR Load process.)
Loaded ISIR Records (most used), all are effective dated and, except for ISIR_COMMENTS, there is one effective dated row for each transaction:

- **ISIR_COMMENTS**: Contains comment codes from ISIR, one row for each comment code per ISIR.
- **ISIR_COMPUTED**: Primary EFC, Prorated EFC, and other calculations.
- **ISIR_CONTROL**: ISIR Received date (used for determining Pell Eligibility), Processing Dates, Pell Eligibility Flag, database matches, and IRS data retrieval codes.
- **ISIR_PARENT**: Parental Data Reported on ISIR
- **ISIR_STUDENT**: Dependency, and student/spouse data.
STDNT_FA_TERM: This record houses all of the enrollment-based financial aid eligibility detail by term for a student. There may be multiple effective-dated rows per term, but we are usually looking for the last effective date/sequenced row for each term. STDNT_FA_TERM should build for each admitted student after being aid year activated. It builds in projection mode if the student has not registered for classes for a term. This table also has data for locked and unlocked hours (UGRD), and for attempted and completed total hour for term and lifetime, and is where the budget assign flag is set for a student.
Frequently Used—Student Need

- **STDNT_AWD_PER**: Shows Federal EFC, Federal Cost of Attendance, Federal Need, Sum of Federal Need-based and Special Aid, Overawards and other data by award period. There are two rows of data for each student per year, one for the standard academic period, and one for the non-standard academic period. Join with AID_YR_CAR_TERM to match STRMs to an Award Period. Sometimes you have to do math with some of the numbers to see if the need summary should be updated.
Frequently Used—Student Budget

- **STDNT_TERM_BUDGET**: This record shows the data that you usually see on Budget Maintenance in the top section, including the EFFDT, the FED_TERM_COA, PELL_TERM_COA and SFA_PELTRM_COA. There may be multiple rows per term, but we are usually looking for the max effective-date/sequence for each term.

- **STDNT_BUDGET_IT**: This record shows the data that you usually see on Budget Maintenance in the bottom section, including the budget category and code, and the amounts (amount, Pell amount, LHT Pell amount). There is one row for each budget category for each effective dated row, but we are usually looking for the max effective-date/sequence for each term.
Frequently Used—Student Awards

- **STDNT_AWARDS**: Has multiple rows of data per aid year, depending on whether an item type has ever been offered. One row for each career/item type combination per year. Shows the current offered, accepted, authorized, disbursed amounts for each award.

- **STDNT_AWRD_DISB**: Has most of the same data as STDNT_AWARDS, but is done by Disbursement ID, and is tied to an STRM. This is one record to use to get term specific award data.
Frequently Used—Student Awards

Records containing award and disbursement data:

- **STDNT_DISB_VW1**: This record combines three other records to give term specific award data for student awards. This was the original way to get this data before STRM was added to **STDNT_AWRD_DISB**. One row per term per item type/career combination per year.

- **STDNT_DISB_AWD**: Do not use this record for general reporting, it is used for manual awarding and will produce incorrect results.
Frequently Used—Student Awards

- **STDNT_AWRD_ACTV**: This record records a row of data for each action that takes place on an award. Records it by aid year for offer, accept, cancel or decline actions. Records it by disbursement id for authorization and disbursement actions. Do not auto-join on disbursement id unless you are only looking for authorization or disbursement activity. This is an auditing record that permanently records every adjustment that is made. Multiple rows typically per item type/career from offer, to accept/decline, to authorize and disburse.
Frequently Used—Student Aggregates

- **STDNT_AGGR_AY**: Aggregate Aid by Level and Year
- **STDNT_AGGR_LIFE**: Aggregate Aid by Aid Year and Aggregate Area. One row per aggregate area per year.
- **STDNT_AGGR_SCHL**: Aggregate Aid by School for Aid Year, Level, and School, shows loans you have input from other schools. Can have multiple rows per aggregate area and year.
- **NSLDS_FAT_AGGR**: Loan Aggregate Data, one row per ISIR or loaded FAT History report.
- **NSLDS_GEN**: Overpayment and default data.
- **NSLDS_PELL**: Pell Detail Data, one row per reported Pell detail.
- **NSLDS_LN_DTL_EC**: Loan Detail Data, one row per reported loan.

FA History records are better sources for detail when working in NSLDS_PELL and NSLDS_LN_DTL_EC.
Frequently Used—Origination

• **LOAN_ORIGINATN**: Records Loan Origination data for loan type that is used in creating the file that goes to the lender or COD. A loan must originate before it disburses.

• **LOAN_ORIG_DTL**: Contains additional information used in creating file. Also ties back to Item Type, and contains loan transmission status, including rejects.

• **PELL_ORIGINATN**: Records basic Pell Origination data that is used in creating the file that goes to COD. Pell can disburse without origination. Includes transmission status, including rejects.

• **PELL_ORIG_DTL**: Contains additional information used in creating file that goes to COD.
Frequently Used—Disbursement

• **STDNT_DISB_MSG**: Records Messages for Awards that have run through manual or batch authorization (based on when award achieves accepted status, and on authorization calendar). If authorized, there will be a disbursement message code that identifies it as disbursed. If any other message is triggered during authorization or during an authorization override, that will also be there. Join with PSMSGCATDEFN to get text of message numbers.

• **LOAN_DISBMNT** and **LOAN_DISB_ACTN**: These record disbursement data for loans and are generated during loan origination and by COD Inbounding. LOAN_DISB_ACTN is where (besides disbursements) checks are manually logged as RCVD (received). Use to identify COD rejected disbursements.

• **PELL_DISBMNT** and **PELL_DISB_ACTN**: These records record disbursement data for Pell and are generated during Pell origination and COD Inbounding. Use to identify COD rejected disbursements.
Frequently Used—Setup Aid Year

- FED_AID_YR_TBL: Aid Year Table
- AID_YR_TBL: Valid Careers for Aid Year
- AID_YR_CAREER: Valid Careers for Aid Year
- AID_YR_PROG: Valid Programs for Aid Year
- TIV_SCHOOL_TBL: School Codes Table
- INST_SCHOOL_CDS: School Codes for Institution
- SCHOOL_CAREER: Careers for School Codes
- FA_APP_SRC_TYP: Application Source Rank
- FA_APP_SRC_TBL: Application Source Rank
- FA_APP_SRC_RANK: Application Source Rank
- SS_FA_INSTALL: Self Service Options
Frequently Used—Setup Agg, FA Term

Aggregate Aid Limits
- AGGR_LIMIT_TBL: Aggregate Aid Limits
- AGGR_LVL_LIMITS: Aggregate Aid Limits

FA Term
- FATRMP_CAREER: FA Term Setup
- FATRMP_CAR_TERM: FA Term Setup
- AID_YR_CAR_TERM: Valid Terms for Career
Frequently Used—Setup Budgets

- RUN_CNTL_BDGT_P: Budget Assign Run Control
- BUDGET_CATG_TBL: Budget Categories
- BDGT_ASSIGN_CNTL: Budget Assignment
- RUN_CNTL_FABDGT: Budget Assign Run Control
- BDGT_ASSIGN_TERM: Budget Assignment
- BUDGET_ITEM_TBL: Budget Items
- BUDGET_GRP_TBL: Budget Groups
- BUDGET_ITEM: Budget Items
- BDGT_TREE_CATG: Budget Trees (We don’t use)
- BDGT_FORMLA_TYP: Budget Formulas
- BDGT_FORMLA_TBL: Budget Formulas
- BDGT_FORMLA_DFN: Budget Formulas
- BDGT_TREE_TBL: Budget Trees (We don’t use)
- BDGT_FRMULA_GRP: Budget Formulas
- BDGT_ASSIGN_CATG: Budget Assignment
Frequently Used—Setup FA Item Types

- ITEM_TYPE_FA: Financial Aid Item Types (Rules)
- ITEM_TYPE_FISCL: Fiscal Item Types (Limits)
- ITM_TP_TERM_LMT: Financial Aid Item Types
- LN_FEE_TBL: Loan Fee Table
- ITEM_TYP_FA_FEE: Financial Aid Item Types
- DISB_PLAN_TBL: Disbursement Plan Table
- DISB_ID_TBL: Disbursement ID Table
- DISB_SPLIT_CD: Disbursement Split Codes
- DISB_ID_SPLIT: Disbursement Split Code Formulas
- ITEM_TP_FA_DISB: Financial Aid Item Types
- AWD_MESSAGE_TBL: Award Messages
Frequently Used—Setup Disb Calendar

• DISB_CAL_TBL: Disbursement Calendar
• AUTH_CAL_TBL: Authorization Calendar
• AUTH_CAL_DATA: Authorization Calendar
• DISB_CAL_DATA: Disbursement Calendar

This setup is used to determine which terms by career should be included when authorization or disbursement are run. This setup is one way to limit the term/career combinations for which an individual or batch can authorize or disburse aid.
Frequently Used—Setup Disb Rules

• DISB_RULE_GL: Global Disbursement Rules
• DISB_RULE_SRV: Global Disbursement Rules for Negative Service Indicators
• DISB_RULE_USR: Global Disbursement Rules for User Edits
• DISB_RULE_TRK: Global Disbursement Rules for Tracking Groups
Frequently Used—Setup Disb Rules

- DISB_RULE_ITM: Item Type Disbursement Rules
- DISB_RULE_CHK: Item Type Disbursement Rules for Checklists
- DISB_RULE_ITS: Item Type Disbursement Rules for Service Indicators
- DISB_RULE_ITU: Item Type Disbursement Rules for User Edit Messages
- DISB_RULE_ITK: Item Type Disbursement Rules for Tracking Groups
<table>
<thead>
<tr>
<th><strong>Frequently Used—Setup Packaging</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>• AWD_ADJ_RSN_TBL: Award Adjustment Reasons</td>
</tr>
<tr>
<td>• FA_INST_DEFN: Institutional Mass Packaging</td>
</tr>
<tr>
<td>• FA_INST_CRIT: Institutional Mass Packaging</td>
</tr>
<tr>
<td>• FA_INST_CRITLST: Institutional Mass Packaging</td>
</tr>
<tr>
<td>• PKG_RTNG_CMP_TBL: Packaging Rating Components</td>
</tr>
<tr>
<td>• PKG_PLAN_TBL: Packaging Plan</td>
</tr>
<tr>
<td>• PKG_RULE_TBL: Packaging Plan</td>
</tr>
<tr>
<td>• FA_BPKG_DEFN: Mass Packaging Definition</td>
</tr>
<tr>
<td>• FA_BPKG_CRIT: Mass Packaging Definition</td>
</tr>
<tr>
<td>• FA_BPKG_CRITLST: Mass Packaging Definition</td>
</tr>
</tbody>
</table>
Frequently Used—Setup Packaging

- PKG_EQUITY_LMT: Pkg Equity Item Types
- PKG_EQUITY_ITM: Pkg Equity Item Types
- PKG_REL_ITM_TBL: Pkg Related Item Type Group
- PKG_REL_ITM_TYP: Pkg Related Item Type Group
- RSTRC_AID: Restricted Aid Table
- RSTRC_AID_ORG: Restricted Aid Table
- RSTRC_AID_COM: Restricted Aid Table
- RSTRC_AID_MEM: Restricted Aid Table
Frequently Used—Setup R2T4

- RTRN_TIV_SETUP: Return of TIV RULES
- RTRN_TIV_SETUP0: Return of TIV RULES
- RTRN_TIV_SETUP2: Return of TIV RULES
- RTRN_TIV_SETUP3: Return of TIV RULES
Frequently Used—Setup ISIR

- ISIR_SRCH_ADD: ISIR Data LoadParms
- ISIR_SRCH_PRM: ISIR Data LoadParms
- ISIR_INST_CNTRL: ISIR Data LoadParms
- INAS_FM_ASMPTN: INAS Assumption Codes
Frequently Used—Setup Pell

- PELL_PMT_OPTION: Pell Payment Setup
- PELL_PMT_CAREER: Pell Payment Detail
- PELL_ATTENDING: Pell ID Attending
- PELL_COMMT_TBL: Pell Comment Codes
Frequently Used—Setup Loans

- LN_TYPE_TBL: Loan Type Table
- LN_TO_NSLDS_TYP: Loan Type Table
- LN_ITEM_TBL: Loan Type Table
- LN_PNOTE_TBL: Loan Type Table
- LN_ACTNTYPE_TBL: Loan Action Code Table
- LN_ACTN_PRG_TBL: Loan Action Code Table
- LN_ACTNMSG_TBL: Loan Action Code Table
- LN_ACTNCD_TBL: Loan Action Code Table
- LN_DESTPROF_TBL: Loan Destination Profile
- LN_DEST_CATGRY: Loan Destination Profile
- LN_DEST_EDIT: Loan Destination Profile
- LN_EDIT_DFLT: Loan Edit Defaults
- LN_EDITDFLT_DTL: Loan Edit Defaults
- LN_EDIT_TBL: Loan Edits/Messages
- SFA_LN_CNSL_SU2: Loan Item Type
Frequently Used—Setup Loans

- LN_INSTUTN_TBL: Loan Institution
- LN_INSTN_CNTACT: Loan Institution
- LN_INST_DEST: Loan Institution
- LN_RPT_ITEM: Loan Report Definitions
- LN_RPT_COL: Loan Report Definitions
- LN_RPT_PKG: Loan Report Packages
- LN_RPT_LNK02: Loan Report Packages
- LN_DL_CHG_XREF: Direct Lending Change fields
- LN_DL_INST_PARM: Loan DL Inst Change Hold/Susp
- LN_PN_TYPE_TBL: DL Serial Prom Note Table
- SFA_COD_LN_ATTD: Attended School Routing ID
- SFA_LNDR_SEL: Lender Select Setup
- SFA_LNDR_DTL: Lender Select Detail
- SFA_LN_DTL: Lender Select Detail
- LN_AGENCY_DFLT: Loan Agency Defaults
Sources of Setup Tables

Tables are added sometimes, so you can also find lists of setup tables in:

• PeopleBooks
• In the log file after doing aid year rollover (useful if you roll each piece separately)
• Also, after doing these slides I found that David Dannert had done a spreadsheet of 9.0 setup tables that is posted on HEUG:

http://www.heug.org/p/do/sd/topic=3&sid=8509
Creating a Query—Start

Click on “Create a New Query”
Creating a Query—Select Record Record

Search for the record you wish to use then click on “Add Record”.
Creating a Query—Pick Display Fields

Select the Fields you want to display.
Creating a Query—Order/Sort Fields

Arrange the fields in the wanted order for columns, and for sort order.

- **Col**
  - A.EMPLID - EmplID
  - A.AID_YEAR - Aid Year
  - A.ITEM_TYPE - Item Type
  - A.ACAD_CAREER - Academic Career
  - A.ACTION_DTTM - Override Date Time
  - A.AWARD_DISB_ACTION - Disb ID Action
  - A.DISB_AMOUNT - Disbursed Amount

- **Heading Text**
  - ID
  - Aid Yr
  - Item Type
  - Career
  - Action Date Time
  - Action
  - Disb Amt

- **Format**
  - Char11
  - Char4
  - Char12
  - Char4
  - DateTm
  - Char1
  - SNM11.2

- **Ord**
  - N
  - N
  - N

**Note:**
- Arrange the fields as desired for columns and sorting.
- Use the 'Reorder / Sort' button to adjust the order.
Creating a Query—Order/Sort Fields

To reorder, you only have to enter the fields you want moved. The column number specifies order on spreadsheet.

For “order by” you have to enter every field that you want in the sort ordering, you can also specify descending.
Creating a Query—Edit Fields

From this tab you can also delete fields, or you can edit them to show a translate value if one exists.

Your column ordering and sort ordering choices display on the fields tab.
Creating a Query—Translates

For some fields you may choose to display a short or long translate value.
Creating a Query—Field Aggregates

On “Edit Field Properties” you may also specify when to aggregate data in some way.
Creating a Query—Aggregating

To aggregate data you will need to delete fields that will create distinct rows, and you will want to make sure that you will not get duplicate data that is being aggregated.
Creating a Query—Distinct

The distinct function is used to hide multiple rows of data if they are substantially the same, but if you are aggregating data the distinct is assumed and multiple rows will make the outcome unreliable.
Creating a Query—Criteria

You will also need to specify criteria that will be used in the select either from here or from query tab or from the criteria tab.

![Query Criteria Table]

<table>
<thead>
<tr>
<th>No.</th>
<th>Field Name</th>
<th>Format</th>
<th>Ord</th>
<th>XLAT</th>
<th>Heading Text</th>
<th>Add Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A.AID_YEAR - Aid Year</td>
<td>Char4</td>
<td>1</td>
<td></td>
<td>Aid Yr</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>A.EMPLID - EmplID</td>
<td>Char11</td>
<td>3</td>
<td></td>
<td>ID</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>A.ACAD_CAREER - Academic Career</td>
<td>Char4</td>
<td>2</td>
<td>N</td>
<td>Career</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>A.AWARD_DISB_ACTION - Disb ID Action</td>
<td>Char1</td>
<td>N</td>
<td></td>
<td>Action</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>A.DISB_AMOUNT - Disbursed Amount</td>
<td>S/NM11.2</td>
<td>Sum</td>
<td></td>
<td>Sum Disb Amt</td>
<td></td>
</tr>
</tbody>
</table>
Creating a Query—Criteria

This query tab is probably the easiest place to use for adding criteria, because all of the fields are available.
Creating a Query—Criteria

When adding a criteria, you can choose whether to compare the piece of data to another field, an expression (you write), a constant, a prompt (asks every time you run), or a subquery.

You also may choose from a variety of ways in which you want to check the data between expression 1 and expression 2 in the query, the standard is equal, but you may want to look for other relationships. Exist or not exist relations start a subquery.
Creating a Query—Criteria

Condition Types:

- **Equal To** (the most used, is typically used for joining records and for matching other data)
- **Not Equal To** (the result must not match [greater than or less than] the value entered or returned)
- **Between** (inclusive between, put two values and amount must be equal to either of the values or between the values entered or returned)
- **Not Between** (the value must not be between or equal to the two values entered or returned)
Creating a Query—Criteria

Condition Types:

• Less Than (the result must be less than the value entered or returned)
• Not Less Than (the result must be equal to or greater than the value entered or returned)
• Greater Than (the result must be greater than the value entered or returned)
• Not Greater Than (the result must be less than or equal to the value entered or returned)
Condition Types, continued:

- **Like** (very useful, allows you to specify characters and the ‘_’ and ‘%’ wild cards to find substrings)
- **Not Like** (allows you to specify characters and the ‘_’ and ‘%’ wild cards to exclude substrings)
- **Is Null** (character fields typically are a blank rather than null, this is useful mostly for effective-date fields)
- **Is Not Null** (again, most fields are not null)
Creating a Query—Criteria

Condition Types, continued:

• In List (allows you to list typically more than one value that will be checked)

• Not In List (allows you to list typically more than one value that will be excluded in the check)

• In Tree (We only have these for SF, I have not used)

• Not in Tree (We only have these for SF, I have not used)
Creating a Query—Criteria

Condition Types, continued:

• Exists (starts a subquery as an exist statement)
• Does not Exist (starts a subquery as a not exist statement)
Once you have added criteria, review the criteria tab checking for possible errors or missing criteria.

Fields like “Aid Year” are often either used as an equal to either a constant or a prompt, or as an “in list”. It depends on your preference.
Creating a Query—Joins

To join another record, go to the records tab and search for all or part of the record name.

To join another record to the existing query, click on the “Join Record” link. You can show the fields if you want to make sure it has a field you need.
Creating a Query—Joins

You will be given the option to do a standard or left-outer join. A standard join is used if there will be a one-to-one or one-to-many relation between the base record and the one being joined to it. A left outer join is used if there may not always be a record in the record being joined, and this will display data only if it exists.

In this case we will do a standard join. To perform the join, make sure you have selected the correct radio-button, then select the record you will join with from this list under “Join Record”.

Select join type and then record to join with STDNT_AWRD_DISB - Student Award Disbursements.

Join Type
- Join to filter and get additional fields (Standard Join)
- Join to get additional fields only (Left outer join)

Join Record
A = STDNT_AWRD_ACTV - Student Award Activity

Cancel
Creating a Query—Joins

The tool will suggest the join criteria, which usually consists of all of the “key” fields. You can deselect one by deselecting, but would usually only do this if you are going to use your own criteria on that field.

If you agree with the selected join criteria, click on the ‘Add Criteria’ button.

Be careful with disbursement ID on STDNT_AWRD_ACTV, it is only populated for actions of ‘H’ (authorize) and ‘P’ (disburse).
Creating a Query—Joins

If you want to change the order of the criteria to make it easier to review, you can click on ‘Reorder Criteria’. The CBO makes it so ordering here doesn’t really affect speed.

Check the criteria tab to review the criteria that now exist. Determine if you need to add additional criteria.

<table>
<thead>
<tr>
<th>Logical</th>
<th>Expression1</th>
<th>Condition Type</th>
<th>Expression 2</th>
<th>Edit</th>
<th>Delete</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A.AID_YEAR - Aid Year</td>
<td>equal to</td>
<td>2013</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.AWARD_DISB_ACTION - Disb ID Action</td>
<td>equal to</td>
<td>P</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.EMPLID - EmplID</td>
<td>equal to</td>
<td>B.EMPLID - EmplID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.INSTITUTION - Academic Institution</td>
<td>equal to</td>
<td>B.INSTITUTION - Academic Institution</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.AID_YEAR - Aid Year</td>
<td>equal to</td>
<td>B.AID_YEAR - Aid Year</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.ITEM_TYPE - Item Type</td>
<td>equal to</td>
<td>B.ITEM_TYPE - Item Type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.ACAD_CAREER - Academic Career</td>
<td>equal to</td>
<td>B.ACAD_CAREER - Academic Career</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>B.DISBURSEMENT_ID - Disbursement ID</td>
<td>equal to</td>
<td>A.DISBURSEMENT_ID - Disbursement ID</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Creating a Query—Joins

If you have specified criteria, and have selected fields to display, then you will be able to click on the “Run” tab to launch the query.

If you do a sum on one amount field, you will probably want to do a sum on other amount fields from the same record if you are displaying them.
Creating a Query—Joins

Here we want to find the disbursed amount by career and term for federal loans. We could put in every federal loan item type and then change it every time we add a new federal loan item type, or...

Using setup records to help with query results. Example: here we will start a new query with STDNT_AWRD_DISB (or could use STDNT_DISB_VW1), we will join with ITEM_TYPE_FA.

Here we want to find the disbursed amount by career and term for federal loans. We could put in every federal loan item type and then change it every time we add a new federal loan item type, or...
Creating a Query—Joins

We may do a join and use fields like these to specify the types of aid we want included:

- **FA_SOURCE** to pick Federal, State, Government, Private, Institutional, or Other;
- **Federal_ID** to pick out specific federal aid programs,
- **LOAN_PROGRAM** to select Direct Lending, FFELP, Health Professions, Perkins, Alternative, State, or University; and **FIN_AID_TYPE** of Waiver, WorkStudy, Athletic, Bursary, Fellowship, Grant, Loan or Scholarship.
Performing a left outer-join to add fields. Here we are starting with a query that shows students with Pell grants put on with PJ, that have an amount on for a particular term.
Next we want to display those that do or do not have a User Edit Message of PJPELL to hold up Pell disbursements until the amount is reviewed (since our Pell repackaging skips awards that are placed with PJ). We are going to do a left outer join to do this.
Creating a Query—Left Joins

This is how the criteria tab looks after doing the left outer join, but we want to add STRM and look for a specific User Edit Message, so we will need to add criteria.

<table>
<thead>
<tr>
<th>Logical</th>
<th>Expression1</th>
<th>Condition Type</th>
<th>Expression 2</th>
<th>Edit</th>
<th>Delete</th>
<th>Belongs to</th>
</tr>
</thead>
<tbody>
<tr>
<td>AND</td>
<td>A.OFFER_AMOUNT - Offer Amount</td>
<td>greater than</td>
<td>0</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.ACA_PROF_JUDGEMENT - Professional Judgement</td>
<td>equal to</td>
<td>Y</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.AID_YEAR - Aid Year</td>
<td>equal to</td>
<td>B.AID_YEAR - Aid Year</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.ITEM_TYPE - Item Type</td>
<td>equal to</td>
<td>B.ITEM_TYPE - Item Type</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.SETID - SetID</td>
<td>equal to</td>
<td>A.SETID - SetID</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.EFFDT - Effective Date</td>
<td>Eff Date &lt;=</td>
<td>Current Date</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.FEDERAL_ID - Federal ID</td>
<td>equal to</td>
<td>PELL</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.EMPLID - EmplID</td>
<td>equal to</td>
<td>C.EMPLID - EmplID</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.INSTITUTION - Academic Institution</td>
<td>equal to</td>
<td>C.INSTITUTION - Academic Institution</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.AID_YEAR - Aid Year</td>
<td>equal to</td>
<td>C.AID_YEAR - Aid Year</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.ITEM_TYPE - Item Type</td>
<td>equal to</td>
<td>C.ITEM_TYPE - Item Type</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>A.ACA_CAREER - Academic Career</td>
<td>equal to</td>
<td>C.ACA_CAREER - Academic Career</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>C.OFFER_BALANCE - Offer Balance</td>
<td>greater than</td>
<td>0</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>C.STRM - Term</td>
<td>equal to</td>
<td>2125</td>
<td>Edit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AND</td>
<td>C.EMPLID - EmplID</td>
<td>equal to</td>
<td>D.EMPLID - ID</td>
<td>Edit</td>
<td></td>
<td>D</td>
</tr>
<tr>
<td>AND</td>
<td>C.INSTITUTION - Academic Institution</td>
<td>equal to</td>
<td>D.INSTITUTION - Academic Institution</td>
<td>Edit</td>
<td></td>
<td>D</td>
</tr>
<tr>
<td>AND</td>
<td>C.AID_YEAR - Aid Year</td>
<td>equal to</td>
<td>D.AID_YEAR - Aid Year</td>
<td>Edit</td>
<td></td>
<td>D</td>
</tr>
</tbody>
</table>

Notice the notation in the ‘Belongs to’ column.
Creating a Query—Left Joins

To keep from breaking the outerjoin, any criteria that uses record alias ‘D’ will need to be changed from a where clause to an outer join.
Creating a Query—Left Joins

This is how the criteria tab looks after doing the left outer join.
Creating a Query—Left Joins

Now the report will show every student that has Pell on for that term with PJ, and whether or not a User Edit of PJPELL has been placed for the term.
Doing a Manual Outer Join—required for joining effective-dated records.

Start by doing a regular join. Then edit each of the join criteria that joins the record to a previous one. Make sure the outer joined record is always on the right.

A manual outer-join is a special case of using expressions to create a query.
Doing a Manual Outer Join

Replace the field reference with an expression that includes the field followed by a “(+)”, as shown above. Do for each join criteria.
Doing a Manual Outer Join.

Click on “Add a Criteria” and add a check that allows for a null effective-dated row.
Doing a Manual Outer Join.

Change the “and” to an “or” between the two effective-date criteria, then group them together.
Doing a Manual Outer Join.

On the expression tab, click on “Add Expression” and add one for each additional criteria you will need to use for the outer-joined record.
Doing a Manual Outer Join.

On the funnel+ to use the expression as a criteria for the outer-joined record, and enter the parameter for the criteria (in this case the term).
Doing a Manual Outer Join.

Once you are done, the final version of the query should have an outer-join with additional criteria that looks something like this simplified version.
Other uses for expressions.

You can use part of the data in a field as a search criteria or display it as a field. ECTRANSID, for example, is something like ‘CPS_ISIR_2013’. But usually we only need part of that.
Other uses for expressions.

Our goal is to change this so we don’t have to always enter that full string, or display it. To begin, select “Add Expression” on the Expressions tab.
Other uses for expressions.

Type the function “SUBSTR(field, start-position, length). After typing SUBSTR you will open and close the rest of the expression with parenthesis. You can type in the alias.field, or select it using the ‘Add Field’ link.
Other uses for expressions.

Here we can select the alias.field if we did not type it in.
Other uses for expressions.

Now we must meet the rest of the formatting requirements for this expression. The definition for this expression is that the alias.field is followed by a comma, then the start position, then the number of characters to return.

- The start position can be a positive number (go right this many), or a negative number (go left from end this many).
- Your length will need to be as long as the number of characters you are returning.
Other uses for expressions.

Once we select okay on the previous page, we then end up with an expression that we can either display as a field or use as a criteria, or both.
Other uses for expressions.

Here we chose to display the expression as a field. This substring expression returns only the year portion of the Transaction ID, which takes less room in the output than the original field, so now we could remove the original field. We may also add this as a criteria.
Other uses for expressions.

Since we now have this specified as a four character field, we may use the aid year portion as a constant to select only the aid year we want...
Other uses for expressions.

or we may use it as a prompt in order to select only the aid year we want.
Other uses for expressions.

Since we now have this specified as a four character field, we can use the aid year portion as a constant or prompt in order to select only the aid year we want.
Other uses for expressions.

Converting (truncating) a date-time to a date, so that we can then use just the date as a criteria.
Other uses for expressions.

Using SYSDATE (system date on server) in a criteria. In this case is will look for anything less than today’s date minus 364 days.
Other uses for expressions.

Using SYSDATE (system date on server) in a criteria. [This query is used to sum the adjust amounts on STDNT_AWRD_ACTV to show the cumulative amount awarded this year, and as of the same date one year ago for the prior aid year.]
Other uses for expressions.

Use a record that has data that you want to do calculations on (amount fields).
Other uses for expressions.

Look at the format on the fields that you will be using, in this case we have a number and a signed number.
Other uses for expressions.

Use a similar format and enter the fields and function that you want used in the calculation. The net disbursement balance is the amount of the offer minus fees, so it is useful when trying to determine when loans are fully disbursed.
Other uses for expressions.

The expression can then be displayed as a field and/or used as a criteria.
Other uses for expressions.

Here we are displaying it as a field. This will show as ‘0’ for anyone where the award is fully disbursed, or as a positive or negative number if it is not.
Other uses for expressions.

The expression can also be used as a criteria since we are not aggregating data for that field. (If we were aggregating data, it would have to be used as a ‘Having’ criteria.)
Other uses for expressions.

Another way to use expressions is to use field data to calculate things like completion rates from FA Term data, keep in mind that you cannot divide by zero, so you have to eliminate that possibility.
Types of Expressions.

- Character/String Functions (CONCAT, LOWER, UPPER, SUBSTR)
- Conversion Functions (CAST, TO_DATE, TO_NUMBER)
- Mathematical Functions (MAX, MIN)
- Date Functions (SYSDATE)

Exist/Not Exist Statements.

- Use to check for the existence or non-existence of data in another record.
- If you are looking for data that doesn’t exist, you cannot do that with regular joins, so the ‘Does not exist’ statement is a good way to do that.
- The exist statement can be used to check for data, where that data does not need to be shown in a field on the report or where the data could impact attempts to aggregate data.
- An exist or not exist statement is like asking a ‘yes’/’no’ question.
Not Exist Statements.

First write the query that will identify the particular characteristics needed, in this case it is students with a Pell grant that was placed using Professional Judgment.
Add a criteria and change the condition type to ‘Does Not Exist’, then click on ‘Define/Edit Subquery’.
Not Exist Statements.

Do not click on ‘OK’, you have defined the subquery.
Not Exist Statements.

Add the record, in this case the record for User Edit Messages, and then tie the required keys back to the top level of the query.
You may join back to different records in the top level to get the pieces you need, then you may add additional criteria as needed. In this case we are trying to exclude students with a particular user edit on a particular term (since we don’t need to place the user edit again).
Not Exist Statements.

When we run the query we will get only those students that have (from the top level) a Pell grant that has been placed with a professional judgment override, and (from the subquery) that do not already have a PJPELL User Edit Message for the same term.
Start with the main select that you want to use. In this case we are looking for students that have federal need-based aid for the year. We want to look for need over-awards.
Exist Statements.

• STDNT_AWARDS has a field called AWARD_PERIOD with valid codes being ‘A’, ‘N’ or ‘B’.
• STDNT_AWD_PER also has the same field, but the valid codes are only ‘A’ or ‘N’.
• If you have packaged some students using an award period of ‘B’ (both), they will be dropped from the query if you use that as a join criteria.
• You could restrict the award period on STDNT_AWD_PER and have one query for the standard academic period and one for the non-standard academic period, but one report might be easier.
• If you are aggregating data in the top level of the query, and you join STDNT_AWD_PER and didn’t restrict it to just one academic period, two rows will be returned for every student. But even with the distinct, if you are doing aggregating data (like summing), the aggregate amount will be doubled because of the two rows.
• Using an exist statement will eliminate these types of problems.
Exist Statements.

So we are going to define a subquery that checks data in STDNT_AWD_PER.
Exist Statements.

Again, STDNT_AWRD_PER has two rows of data for every student for each year, one for the standard academic period and one for the non-standard academic period.
Exist Statements.

We will use join criteria to tie to key fields in the top level of the query, and then look for the need overaward. We are not designating award period, so we are asking if they have an over-award on at least one award period of the year. Even if an over-award is on both, it will still in effect only return one ‘Yes’.
So our results will show the sum of amounts of need-based federal loans, for any student that has an over-award on one of the award periods.
Having.

• Will still allow you to refine a query when you are aggregating data (min, max, sum, count) at the field level.
• PeopleSoft will usually do it automatically if you change the field setting before adding the criteria.
NSLDS Data comes in on ISIR and NSLDS files, then can be pushed to aggregates, we wanted to make sure we had the amount pushed before packaging.
We will do an outer-join to include students that may not exist in the second record (could do as an exist statement also, but we wanted to check the numbers initially).
So we make sure to add enough criteria using manual or left outer join (in this case manual) to only get the data from the second record that is applicable to the Pell for that year and student.
We will setup the field to do the summing the way we want, then add the criteria (you cannot add the criteria first and then setup the aggregate on the field).
Adding the criteria to the having, we can then specify that we only want students with a total in a certain range.
We wanted to make sure anyone over 500% had been pushed to aggregates, and that the amount looked correct. (This is a subset of our data, there are many more students.)
Example Using Several Elements.

Let’s say we want to get all of the direct loan disbursements that had taken place prior to a certain date (so use award activity record), to reconcile them with the data that comes with the DLSAS reports that come each month.
Example Using Several Elements.

Let’s start by making it so we can use the date substring of the date-time by using an expression to change it.
Example Using Several Elements.

So now our criteria tab looks like this. To match with the data that will come in from COD we need the loan identification number, so we need LOAN_ORIG_DTL.
Example Using Several Elements.

We also know that we are reconciling disbursements, so we are going to get a total of the disbursements that happen prior to the chosen date...
Example Using Several Elements.

So here we are creating a new expression, and will aggregate the disbursed amount for all disbursement rows from award activity that are dated prior to the date we enter at prompt.
Example Using Several Elements.

We are now displaying the data from our expression as a field. Next we will need to join with the COD detail, but we need to use an outer-join so that we don’t miss the loans that COD didn’t have.
Example Using Several Elements.

So we will add the loan level detail record...
Example Using Several Elements.

Then we want to change or enter join criteria to make it an outer-join. In this case we don’t have to change anything because it doesn’t find any join criteria, so we will set them.
First we will add our only join that will tie the data together using the loan application id and DL Loan ID (which have the same data, even though they have a different name. This has to be an outer join, so we will use an expression for record D.
Next we will add an expression that we will use for a criteria to pick the correct ECTRANSID from our data.
Example Using Several Elements.

Then we will use this as a criteria, and set the value.
Example Using Several Elements.

Finally, we want to be able to pick the correct date, so we will add another expression.
Example Using Several Elements.

For this one we are going to pick a date that is within a couple of days (has margin for differences exhibited by data in record) of the date that we enter in the prompt.
Example Using Several Elements.

In the end our criteria will look like this. All references to record alias D, will have the “(+)” to maintain the outer join.
Example Using Several Elements.

The field tab now looks like this. We can now run our report, and we will get every student that has had a disbursed amount on award activity prior to the first day of the month following the month we are reconciling (which is the date we enter at the prompt), and will match that data with the COD data (where it exists).
Now we can run this out to a spreadsheet, and can compare the A.DISB_AMOUNT sum to the DL SvcrNet. Any where there is nothing in the last column, or where the amount is different, needs to be reviewed and reconciled.
Writing a query for Pop Select or Pop Update is pretty much like writing any other query, except that you have to add a bind record. The bind record makes sure all of the needed fields are available for the process.
Only queries that include the bind record can be used in the pop select or pop update. Find the bind record by pulling up one of the delivered queries (start with ‘QA’).
On the query tab, we see the name of the bind record for the pop update process for the record we are trying to update. Depending on which record we are updating, there will often be different bind records.
Now we may take a query that we have already written to identify our population, and add the bind record at the end. It doesn’t matter if you start or end with the bind record, or put it somewhere in the middle, it just has to be there.
All of the key fields from the bind record must be included in the query. Additional fields may follow.
The finished criteria for the query look like appear as above and following...
This query is designed to find students that have changed loads before we have made the ‘initial Pell calculation’ when that calculation will occur after the census date, so that we may reset the load.
Once we have saved the query with the added bind record, we can pull it up and use it in the pop select or pop update. If you did a prompt for aid year or some other value, there will be a link to set the prompt values on the run control.
Query for Pop-Select

We have used Pop Selects or Pop Updates to:

• Set budget assign flags to ‘no’ if there is a professional judgment item inserted
• For updating or creating checklists or communications (caution if updating checklist item in a tracking group).
• Set processing status to “ready to package” for eligible students
• Set processing status from “repackage” to applied or packaged after external award load if repackaging is not needed.
• Hold loan originations sometimes
• Etc.
There is still a lot to learn, we haven’t even covered...

- Unions
- Connected Queries
- Drilling URLs
Drilling (if there’s time)

Here we are going to add an expression that shows as a link on one of the fields in the query data.
Drilling (if there’s time)

To do this you will need to know the menu, component, and page.

Expression Type: Drilling URL
Expression Text:
'/c/PACKAGE_AID.AWARD_ENTRY_MC.GBL?
Page=STNDT_AWARD_ENTRY3&Action=U&EMPLID=%A.EMPLID%%AID_YEAR=%A.AID_YEAR%
&INSTITUTION=BYU:A.EMPLID'
Drilling (if there’s time)

• In our database they all start with ‘/c/
• Menu.Component
• .GBL?
• Page = information
• &Action=U
• Provide the ‘key’ data for a search, but we can get it from fields in our query
• :Alias.Fieldname’ (this tells the query where to put the link, and ends the expression).
<table>
<thead>
<tr>
<th><strong>Browser</strong></th>
<th>IE/8.0</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Operating System</strong></td>
<td>WINXP</td>
</tr>
<tr>
<td><strong>Browser Compression</strong></td>
<td>ON (gzip)</td>
</tr>
<tr>
<td><strong>Tools Release</strong></td>
<td>8.51.20</td>
</tr>
<tr>
<td><strong>Application Release</strong></td>
<td>HRMS and Campus Solutions 9.00.22.000</td>
</tr>
<tr>
<td><strong>Service Pack</strong></td>
<td>22</td>
</tr>
<tr>
<td><strong>Page</strong></td>
<td>STDNT_AWARD_ENTRY3</td>
</tr>
<tr>
<td><strong>Component</strong></td>
<td>AWARD_ENTRY_MC</td>
</tr>
<tr>
<td><strong>Menu</strong></td>
<td>PACKAGE_AID</td>
</tr>
</tbody>
</table>
Drilling (if there’s time)

You will not need to use it as a field. If you have other types of expressions, they will be listed in a different section.
Drilling (if there’s time)

Our field setup and other tabs will look the same as they did before we added this drilling expression.
The EMPLIDs now show as links that will take whoever clicks on them straight to the Award Entry page. If downloaded to Excel, the links will remain.
Contacts

• Shawn Trauntvein
  – System Analyst
  – Financial Aid Office
  – Brigham Young University
  – E-mail: shawn_trauntvein@byu.edu
THIS PRESENTATION AND ALL ALLIANCE 2013 PRESENTATIONS ARE AVAILABLE FOR DOWNLOAD FROM THE CONFERENCE SITE AT WWW. ALLIANCE-CONFERENCE.COM

Presentations from previous meetings are also available