

Memorized Queries in WinSearch

Overview

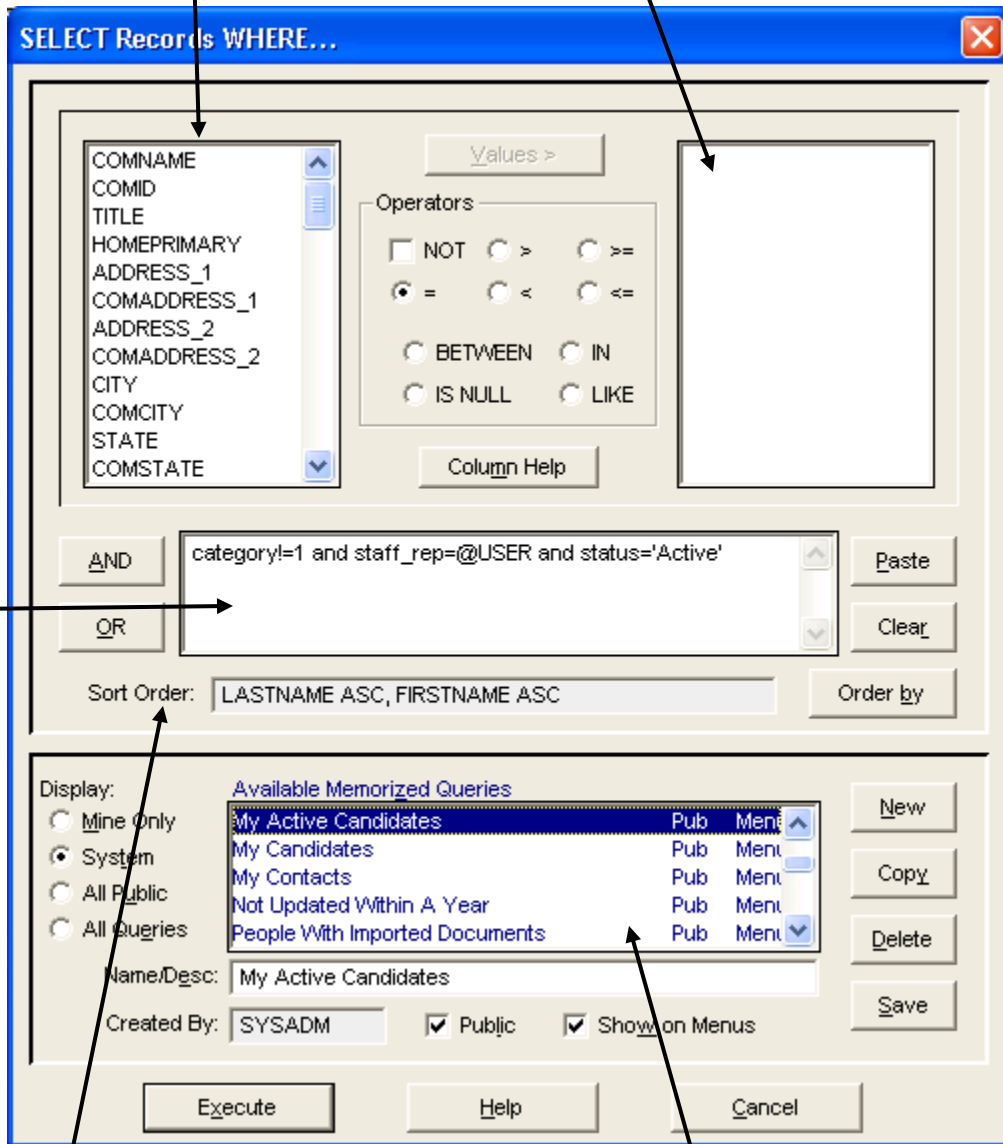
- Memorized, or saved, Queries can be set up for those queries that you create and use repetitively.
- Memorized Queries are available on menus for the following entities:
 - Persons
 - Companies
 - Job Orders
 - Searches
 - Activities (from the Calendar menu or Daily Planner)
- There are three types of Memorized Queries
 - **System Queries** – These queries are public SYSADM queries that are available to any WinSearch user.
 - **Public Queries** – These queries are public queries set up by users other than SYSADM that are made available to all WinSearch user.
 - **Private Queries** – These queries are set up to be available only to the WinSearch user who created it.

Advanced SQL Query Window

- Memorized Queries are created in the Advanced SQL Query window.
- The Advanced SQL Query window is accessible from any form or table.
- Please note: When executing a query from the Advanced SQL Query window, if you access the window from a form the results are displayed in a form, (from a table, the results are in a table).
- Go to **QUERY** → **ADVANCED SQL QUERY** to find the following screen:

Fields List Box: A List box of fields available to you for building your memorized query.

Values List Box: List of Values that correspond to the field selected in the Fields List Box.



Sort Order: Designate how you would like to see the resulting records sorted.

Memorized Queries: A list of your existing Memorized Queries.

Query Where Text Box: The syntax for the query that you are executing.

Creating a Memorized Query

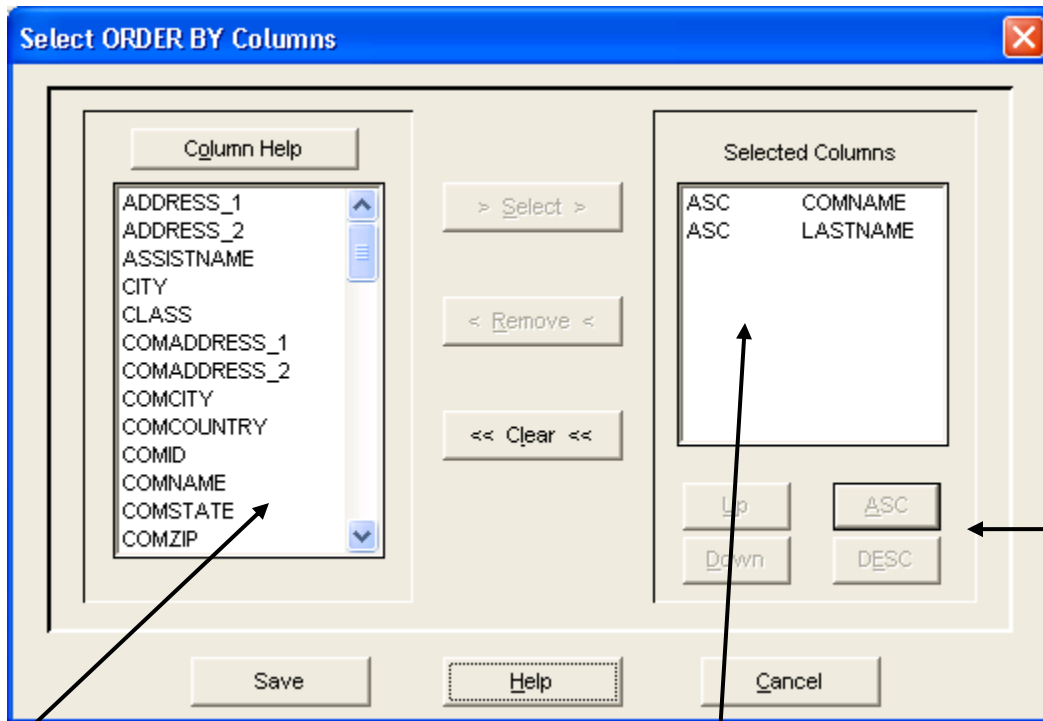
1. Open the form or table window where you want to see the resulting data.
2. Select the **QUERY → ADVANCED SQL QUERY** command to open the **Select Records Where...** dialog box. Enter the criteria that selects the data you want returned. (Examples to follow)
3. Set the order for how you want the information returned in the form or table window using the **Order By** button. (See **Setting Sort Order** section below).
4. Select **Execute** and verify that the data is correct.
5. Select **QUERY → ADVANCED SQL QUERY** command to open the **Select Records Where...** dialog box again. The previous criteria you entered should still be visible in the Query Where text box.
6. Select the **New** button in the bottom half of the window.
7. Type the name for this new query in the **Description** text box.
8. Select whether you want this memorized query to be available for other WinSearch users by selecting or clearing the **Public** check box.
9. Select whether you want this memorized query to appear on the menus by selecting **Show on Menus** check box. (The menus for memorized queries will display up to 30 memorized queries.)
10. Select **Save** when you are finished.

Setting Sort Order

Here you designate how you would like the resulting records ordered. Typically it's alphabetically by last name for persons and alphabetically by company name for companies and job orders.

1. Click on the **Order By** button on the Advance SQL Query window, after you have entered the query syntax.
2. There are three options: **Use Defaults**, **Clear Entry**, **Open Order By**
3. To change the sort order click **Open Order By**. (If you change your mind down the road you can always restore the defaults by clicking **Use Defaults**.)

Here is the **Select ORDER BY Columns** Screen:



List of Columns – displays all the columns you can use to sort the data

Selected Columns – columns you have selected to display the data.

ASC vs. DSC – ascending or descending order

4. From the **List of Columns** on the left, highlight the columns you want to sort by and hit the **Select** button in the middle of the screen to move them to **Selected Columns**.
5. To remove one column from **Selected Columns** click on it and hit the **Remove** button. To clear all columns and start over hit **Clear**.
6. Use the **Up** and **Down** buttons below **Selected Columns** to change the order of the columns, whichever is listed first is the primary sort method.
7. By default the values are listed in Ascending order, to change to Descending highlight the column under Selected Columns and click the **DESC** button.
8. Hit **Save** when you are finished.

Memorized Query Examples

The following are examples of popular Memorized Queries often requested by our customers.

Please Note: Some of these Memorized Queries refer to configurable fields that may be setup differently in your WinSearch database.

Persons Queries

Candidates with placements for “Filled” job orders

```
id in (select entid from act_entities where enttype='$PER$' and actid in (select id from activities, act_entities where id=actid and actcode='Placement' and enttype='$ORD$' and entid in (select id from orders where status='Filled'))))
```

*Note the Status on this; you will need to make sure it is what you have set in your database.

Candidates submitted for a specific job order

```
id in (select entid from act_entities where enttype='$PER$' and actid in (select id from activities where actcode='Resume Submitted' and actid in (select actid from act_entities where enttype='$ORD$' and id=227)))
```

*You can adjust the activity code and job order ID number

Candidates connected to a specific job order

```
id in (select entid from act_entities where enttype='$PER$' and actid in (select actid from act_entities where enttype='$ORD$' and id=227))
```

*Again on this one you would adjust the job order ID number

Perm Apps Added This Week

```
Category=2 and create_date>=@SYSDATE-7
```

Perm Apps Added This Month

```
Category=2 and create_date>=@MONTHBEG(@SYSDATE)
```

Perm Apps Added This Year

```
Category=2 and create_date>=@YEARBEG(@SYSDATE)
```

*You can easily change this query to use a different category. Use 1 for Contact, 2 for Perm Applicant, 3 for Contractor, 4 for Temporary, 5 for Employee, 6 for Other.

People available this month

@DATE(datepart(year, getdate()), @MONTH(firstavailable), @IF(@MONTH(firstavailable)-2, DAY(firstavailable), @IF(DATEPART(day, firstavailable)-29, DAY(firstavailable), 28)))

between

@SYSDATE-convert(int, dbo.ssChoose8(datepart(dw, getdate())-1, 6, 0, 1, 2, 3, 4, 5))+7 and

@SYSDATE-convert(int, dbo.ssChoose8(datepart(dw, getdate())-1, 6, 0, 1, 2, 3, 4, 5))+13 and firstavailable is not null

People With Birthdates in the next 14 days (Birthdate is a UDF):

id in (select udfentid from udfdata where convert(datetime, @IF(charindex('Feb 29', left(udfdate, 6)), 'Feb 28', left(udfdate, 6))+', ' + ltrim(str(datepart(year, @SYSDATE), 10, 0))) between @SYSDATE and @SYSDATE+14 AND udfid=1006)

* Replace 1006 with the number of the UDF.

Potential duplicate people

Firstname+lastname in (select firstname+lastname from persons group by firstname+lastname having count(firstname+lastname)>1)

People with duplicate email addresses

Id in (select id from emailaddresses where emailaddress in (select emailaddress from emailaddresses group by emailaddress having count(emailaddress)>1))

Candidates without Imported Documents

id not in (select id from impfile where enttype='\$PER\$')

Candidates without Resumes

(category=2 or category=3 or category=4) and id not in (select id from restext)

People with Blank Resumes

id in (select id from restext where @LENGTH(restext)<10)

People with resumes documents but no WSResume document

id not in (select id from restext) and id in (select id from impfile where enttype='\$PER\$')

Records with no Phone number

ID not in (select id from phonenumbers where entitytype='\$PER\$')

Records with no Email Address

ID not in (select id from emailaddresses)

People with Bad Email Addresses

id in (select id from emailaddresses where charindex('@', emailaddress)=0 or charindex('@', emailaddress)=1 or charindex('.', emailaddress)=0 or charindex('<', emailaddress)>0 or charindex('>', emailaddress)>0 or charindex(' ', emailaddress)>0 or charindex(',', emailaddress)>0 or right(emailaddress,1)='.' or right(emailaddress,1)='-')

Company Queries

Potential Duplicate Companies

name in (select name from company group by name having count(name)>1)

Potential duplicates as determined by phone number:

id in (select id from phonenumber where entitytype='\$COM\$' and entitytype+areacode+localnumber in (select entitytype+areacode+localnumber from phonenumber group by entitytype+areacode+localnumber having count(entitytype+areacode+localnumber)>1 and left(entitytype+areacode+localnumber ,5)='\$COM\$'))

Companies without people working at them:

id not in (select comid from persons)

Job Order Queries

Open Job Orders that have had Resume Submitted activities

status like 'Open%' and id in (select entid from act_entities where enttype='\$ORD\$' and actid in (select id from activities where actcode='Resume Submitted'))

Job Orders open between 1/1/09 and 7/1/09

STATUS = 'Open' or create_date between 1/1/09 and 7/1/09 or close_date between 1/1/09 and 7/1/09)

Orders Closed Last Week

close_date<@SYSDATE- datepart(weekday, @SYSDATE)-1 and close_date>=@SYSDATE- datepart(weekday, @SYSDATE)-8

Orders Closed Last Month

close_date<@MONTHBEG(@SYSDATE) and close_date>=@MONTHBEG(@MONTHBEG(@SYSDATE)-1)

Orders Closed Last Quarter

close_date<@QUARTERBEG(@SYSDATE) and
close_date>=@QUARTERBEG(@QUARTERBEG(@SYSDATE)-1)

Activity Queries

Placements made in a date range for a specific WinSearch user:

actcode='Placement' and apptdate>='10/01/2004' and apptdate<='10/31/2004' and id in (select
actid from act_users where staff_rep='SYSADM')

*Staffrep would be changed from SYSADM to whatever staffrep name you prefer.

Interview activities in a date range for candidates with a specific staff rep

staff_rep='GLAWRENCE' and id in (select entid from act_entities where enttype='\$PER\$' and actid
in (select id from activities where ACTCODE = 'Interview - Client' AND apptdate>= '1/1/05' and
apptdate<='6/30/05'))

*Again, staffrep name would be changed

All Completed Year to Date Tasks

APPTSTATUS='Completed' and apptdate<= @SYSDATE and apptdate>=@YEARBEG(@SYSDATE)

All Completed Activities For A Specific Group Of Users for the Current Month

APPTSTATUS='Completed' and apptdate>= @MONTHBEG(@SYSDATE) and apptstaffrep in
('USER1', 'USER2', 'USER3')

*Again, staffrep name would be changed

Built-In Functions

Besides using SQL Server functions in your queries, you can also use functions built into your WinSearch database. To use them, you must have the function name in all capital letters. These functions include:

@SYSDATE	@YEAR	@YEARBEG
@SYSTIME	@MONTH	@QUARTERBEG
@USER	@MONTHBEG	@WEEKBEG