

***Team Approach 5.0.1***  
***Understanding Queries***

---

---

***Team Approach 5.0.1***  
***Understanding Queries***

---

---

# Contents

<b>Query Concepts .....</b>	<b>3</b>
About This Chapter .....	3
Overview of Queries .....	3
SQL Statement .....	4
Standard and Ad Hoc Queries .....	4
Query Privileges.....	4
Parts of a Query.....	5
Query Status.....	6
Query Naming Convention.....	7
Source Code Queries.....	8
Parameterized Queries .....	8
Reports that Run with Queries .....	8
Overview of the Query Entry Screen.....	9
First Page of the Query Entry Screen.....	9
Second Page of the Query Entry Screen.....	11
Query Type .....	13
Data TA Returns Based on the Query Type .....	14
Selection Criteria .....	17
Include and Exclude Queries.....	18
Include Queries.....	18
Exclude Queries.....	19
Address Selection Rules .....	21
Expected Arrival Date .....	21
Selection Rules Application.....	21
System Preferences (Queries) .....	22
Special Keys (Query Entry Screen) .....	22
Special Keys in the Include Block and Exclude Block.....	23
Special Keys in the Query Field Name Field .....	23
Special Keys in the Values Field .....	23
<b>Queries Step-by-step.....</b>	<b>24</b>
Finding a Query.....	24
Working with User Groups (Queries) .....	25
Finding a User Group (Queries) .....	26
Creating a Query .....	26
Copying Over Your Query with an Existing Query .....	27
Working with Include and Exclude Queries.....	27
Adding and Removing Include Queries .....	28
Adding and Removing Exclude Queries .....	28
Using Previously Retrieved Records for an Include or Exclude Query.....	28
Zooming to View the Details of an Include or Exclude Query .....	29
Setting Selection Criteria .....	29
Working with Queryable Fields.....	30
Selecting an Operator and Entering a Value .....	32
Special Treatment of Nulls for a Selection Criterion .....	34
Connecting Multiple Selection Criteria.....	40
Adding to and Deleting from a Set of Selection Criteria .....	43

# **Team Approach 5.0.1**

## **Understanding Queries**

---

---

Setting Processing Options on the Query Entry (Detail) Screen.....	44
Setting a Query Effective Date .....	44
Setting a Preferred Address Type .....	45
Linking the Query to a Source Code .....	45
Calculating Ask Amounts .....	46
Returning Special Action Information .....	47
Marking the Query as Public or Private .....	47
Marking the Query as an Include or Exclude Query .....	47
Allowing Other Users to Edit Retrieved Records .....	48
Setting the SQL Edited Field .....	48
Setting an Expiration Date for the Retrieved Records.....	48
Enabling the Query to Search for Inactive Records .....	48
Viewing and Editing the SQL Statement .....	49
Testing a Query.....	49
Testing a Query to Generate the Query's SQL Statement .....	49
Testing a Query to See If It Is Selecting the Correct Records.....	50
Reviewing the Query's Purpose, Schedule, or Related Output .....	50
Entering and Viewing a Note to Describe the Query.....	50
Accessing the User-defined Output Entry Screen.....	51
Accessing the Query and Output Schedule Screen .....	51
Deleting a Query .....	51
Marking a Query for Deletion .....	51
Unmarking a Query Marked for Deletion .....	52
<b>Definition of Ownership .....</b>	<b>52</b>

## Query Concepts

[About This Chapter](#)

[Overview of Queries](#)

[Overview of the Query Entry Screen](#)

[Query Type](#)

[Selection Criteria](#)

[Include and Exclude Queries](#)

[Address Selection Rules](#)

[System Preferences \(Queries\)](#)

[Special Keys \(Query Entry Screen\)](#)

### About This Chapter

The Understanding Queries chapter is meant to familiarize you with the concept of a query, and provide instruction on how to use the Query Entry screen.

This chapter contains many examples. Due to the sophistication of the Team Approach® (TA) query facility, not all examples are described in exhaustive detail. When an example is included in a section to illustrate a concept, the example provides the level of detail appropriate to the concept, i.e., whether the section provides an overview of a concept or provides a detailed description of the concept.

See the Writing Queries: Samples and Guidelines chapter for a description of the standard queries delivered with TA. See the Understanding Outputs chapter for an explanation of outputs and how to create a user-defined output. See the [Scheduling Queries and Viewing Records Retrieved chapter](#) to learn how to schedule queries, user-defined outputs, and update procedures. See the Understanding Update Procedures chapter for an explanation of each update procedure.

### Overview of Queries

In TA, a query is a "question" you ask the TA database to find a set of TA records. The question you ask is composed of a set of criteria. TA compares the records stored in TA to the criteria you set up and selects only those records that meet the criteria.

For example, in TA you store the account type for each account. You could write a query to find each national foundation stored in TA; you are asking the question "Which accounts are national foundations?" For this query, you would set up the criterion: "the account type equals the national foundation account type." When you run this query, TA selects only those accounts assigned the national foundation account type.

If you want to mail a letter to each national foundation, you could run this query with a user-defined output. A user-defined output identifies which information to use from the records selected by the query. You could set up a user-defined output to pull an account's salutation and mailing address. In this example, the query selects the accounts that are national foundations, and the user-defined output pulls the salutation and mailing information for each national foundation account selected.

Often, you want to find a set of records to complete one of the following tasks.

- Prospecting, e.g., pulling a list of prospects for a capital campaign
- Setting up an appeal, e.g., sending an additional gift mailing
- Collecting a pledge, e.g. sending an installment bill
- Cultivating a group of donors, e.g., sending a newsletter to your major donors
- Reporting, e.g., sending a direct mail vendor a report listing the number of pieces per ZIP Code for a mailing

# Team Approach 5.0.1

## Understanding Queries

---

### See Also:

- [SQL Statement](#)
- [Standard and Ad Hoc Queries](#)
- [Query Privileges](#)
- [Parts of a Query](#)
- [Query Status](#)
- [Query Naming Convention](#)
- [Source Code Queries](#)
- [Parameterized Queries](#)
- [Reports that Run with Queries](#)

### SQL Statement

TA's Query Entry screen enables you to create complex database queries without needing a programmer. When you test a query you create, TA takes the information you entered on the Query Entry screen and generates the actual SQL statement to use for selecting the records. "SQL" stands for Structured Query Language, which is a standardized programming language used to access and store information in a database.

### Standard and Ad Hoc Queries

A "standard" query refers to one you expect to run more than once. For example, your organization may send renewal notices on a monthly basis. Your organization can set up a query that is used every month to send those notices. TA is delivered with a set of standard queries that you can copy and either use as they are or modify to meet your organization's needs.

Two kinds of standard queries exist: those that are run on a regular basis (e.g., monthly) and those that find a standard set of records.

- An example of a standard query that is run on a regular basis may be your organization's membership renewal query.
- An example of a standard query that finds a standard set of records is one that finds your organization's major donors (e.g., donors who have given over \$1,000). Your organization can use this standard query to exclude your organization's major donors accounts from queries that select accounts for mass-marketing mailings.

An "ad hoc" query refers to one you might use only once to gather information for a particular purpose.

You create standard queries and ad hoc queries in the same way using the Query Entry screen.

### Query Privileges

The query facility of TA enables you to create, copy, update, test, and schedule queries. In addition, you can also save the results from a query. Who "owns" the query and whether the query is marked as private determines which of these functions you can perform with a query.

- A query is owned by the user who created it. If a query is associated with a user group, then it is also owned by the members of that user group.
- A query owner can mark the query private to indicate that it can only be used by one of the query owners.

This table illustrates query privileges.

	<u>Owner</u>	Not Owner/ Public	Not Owner/ Private	pbadmin/ Public
<b>View</b>	X	X		X
<b>Update</b>	X			
<b>Test</b>	X			
<b>Copy</b>	X	X		X
<b>Schedule/ Run</b>	X			X

As you can see from the table, if you own the query, you can update, copy, test, schedule, and save the results of the query. (These results refer to those from testing the query or from executing the query once it has been tested.) Otherwise, you can

- copy any other owner's public queries, and
- schedule only PADMIN's queries.

## See Also:

[When to Use the PADMIN User](#)

## When to Use the PADMIN User (Queries)

TA is delivered with the username "PADMIN." All standard queries delivered with TA were created using the PADMIN user. Staff at your organization should log into TA using the PADMIN username and password to create each query that should be accessible to all staff members, e.g., regular production queries such as your organization's membership renewal query and acknowledgment query. Staff at your organization should create queries specific to their job responsibilities using their own user names; staff can copy the PADMIN queries to create their own queries.

## Parts of a Query

A query, whether it is a standard query or an ad hoc query, can contain the following four parts.

- The query type
- One or more active queries in the Include block
- One or more active queries in the Exclude block
- One or more field criteria in the Selection Criteria block

The query *must* be assigned a query type.

If you do not enter any active queries in the Include block or in the Exclude block, you must enter the field criteria in the Selection Criteria block.

If you do not set up any field criteria in the Selection Criteria block, you must at least enter one or more existing queries in the Include block or in the Exclude block.

## See Also:

[How TA Selects Records](#)

# Team Approach 5.0.1

## Understanding Queries

---

### How TA Selects Records

TA uses the query type, the selection criteria, the include queries, and the exclude queries to select the records as shown below.

1. TA starts with the query type to determine from where records should be selected.
2. After determining from where to select records, TA selects the records that meet the criteria entered in the Selection Criteria block. If no include queries are used, TA skips to step #5.
3. If one or more include queries are entered in the Include block, TA runs the include queries.
4. TA checks the results of the include queries against the results of the selection criteria and selects only those records that meet *both* the criteria represented by the query or queries in the Include block *and* the criteria entered in the Selection Criteria block.
5. If one or more exclude queries are entered in the Exclude block, TA runs the exclude queries.
6. TA removes the records that meet the criteria represented by the query or queries in the Exclude block from the records that were selected by the selection criteria and include queries.
7. TA eliminates any duplicate records.

**Note:** When there is more than one include query or more than one exclude query, TA handles the records in the way that you specify (e.g., TA either selects every record selected by the exclude queries, or just the records common to all of the exclude queries).

### Query Status

For each query, TA displays the status of the query in the upper right corner of the Query Entry screen. You cannot update the Status field; it is a display-only field.

#### See Also:

- [Active Status](#)
- [Changed Status](#)
- [Delete Status](#)
- [Errors Status](#)
- [Inactive Status](#)

### Active Status

TA activates a query, i.e., sets the query's status to "A" for active, in two situations.

- You successfully test a query you create, i.e., the test is error-free. See "[Testing a Query](#)".
- You copy an existing query. You can only copy an active query. The status remains active until you change any of the information for the query.

Once a query is activated, you can schedule the query to run, copy the query, and use the query in the Include block or Exclude block.

### Changed Status

TA sets the status to "C" for changed in three situations.

- You create a query without copying an existing query.

- You change any of the following information for a query: any field on the first page *except* for the query's description, the effective date of the query, the address type to select for the query, the telephone type to select for the query, and the SQL edited indicator to select for the query. The last four are located on the second page of the Query Entry screen.
- You unmark a query marked for deletion.

You cannot schedule the query to run, copy the query, and use the query in the Include block or Exclude block until you activate the query. To activate the query, you must test the query. See "[Testing a Query](#)".

See "[Unmarking a Query Marked for Deletion](#)".

## Delete Status

TA sets a query's status to "D" for delete when you click <Delete> on the toolbar. See "[Marking a Query for Deletion](#)".

**Note:** Make sure your cursor is not in the Selection Criteria block when you click <Delete>; if it is, TA deletes the selection criterion rather than changing the status of the query.

You cannot schedule the query to run, copy the query, and use the query in the Include block or Exclude block until you unmark the query for deletion and activate the query.

See "[Unmarking a Query Marked for Deletion](#)". To activate the query, you must test the query. See "[Testing a Query](#)".

## Errors Status

TA sets a query's status to "E" for errors when you test the query and it contains errors. A query with the error status cannot be run. If the query with the error status is an include or exclude query for another active query, that active query can't be run.

See "[Testing a Query](#)".

## Inactive Status

TA sets a query's status to "I" for inactive when you exit a query whose status is changed. When you try to exit a query with the status of changed, TA displays the message "Query was changed and not tested. Leave record or test query first?" If you click <Leave Record> to leave the query without testing it, TA changes the status of the query to "I" for inactive.

You cannot schedule the query to run, copy the query, and use the query in the Include block or Exclude block until you activate the query. To activate the query, you must test the query. See "[Testing a Query](#)".

## Query Naming Convention

The name of each standard query delivered with TA begins with the letters "TA." You should follow the conventions your organization has set up to name queries, e.g., begin each query you create with your initials or with the initials of the department that will use the query.

**Note:** If you are creating a source code query instead of using one of TA's automation features to create the query, check the [Prefix for Source Code](#) system preference to see if your organization has set this system preference. This system preference contains the characters your organization wants to use as a prefix for each source code query when one of TA's automation features is used to create a source code query.

# Team Approach 5.0.1

## Understanding Queries

---

### Source Code Queries

A source code query is a query your organization creates to pull accounts to be solicited where the query is tied a source code that represents a specific solicitation. TA contains four automation features to create source code queries. You can have TA create a source code query when:

- You create a source code directly from the initiative level of the marketing hierarchy using the Initiative Entry screen.
- You create a source code directly from the effort level of the marketing hierarchy using the Effort Entry screen.
- You create a source code directly from the source code level of the marketing hierarchy using the Source Entry screen.
- You create a source code by copying an existing source code using TA's Copy Sources and Queries by Effort screen.

In each of these situations, you can indicate that in addition to creating the source code, TA should create the source code query. When you have TA create the source code query using one of these features, TA sets the name of the query to SOURCECODE and sets the source code on the query. For example, if you use one of TA's automation features to create a query for the source code "AMA990101001," then TA sets the name of the query to AMA990101001 and sets the Source field for the query to AMA990101001.

If your organization sets the [Prefix for Source Code](#) system preference, TA creates the source code query using the value of this system preference as a prefix for the query name. TA sets the query name to PFX SOURCECODE where "PFX" is the prefix and "SOURCECODE" is the source code for which the query was created. For example, if you use one of TA's automation features to create a query for the source code "AMA990101001" and your organization set the Prefix for Source Code system preference to "ASK," then TA sets the name of the query to ASK AMA990101001 and sets the Source field for the query to AMA990101001.

See "[Creating the Source Code](#)" and "[Linking the Query to a Source Code](#)".

### Parameterized Queries

Parameterized queries allow you to create a query with parameters in place of fixed values in the selection criteria. These parameters function as placeholders when the query is being composed and tested. Each time the query is scheduled to be run, you can give different values to the parameters. This allows you to create one query where you might have had to create several very similar ones.

For example, you might want to create a query that retrieves accounts whose last gift was made during a given calendar year, without specifying the year. You can do this by creating a parameterized query. You can create parameterized values such as ":BEGINDATE" and ":ENDDATE" with text prompts for the appropriate values (e.g., "Enter the first date of a calendar year" and "Enter the last date of the same calendar year").

See "[Using a Parameterized Value](#)".

### Reports that Run with Queries

Below is a list of reports that run only with queries and those that can run standalone or with queries.

#### Reports That Only Run with Queries

- 1 Dimensional Statistics Report
- 2 Dimensional Statistics Report
- Account Overview Report
- Accounts by Benefit Report

Address Report  
 Charge Output Report  
 Count by Benefit Report  
 Effort Check List  
 Multi-Part Installment Reminder  
 Multi-Part Pledge Bill  
 Query Retrieved Effort Check List  
 Query Retrieved Gift Summary Report  
 Stock Valuation Letter

**Reports that can Run Standalone or with Queries**

Accounts	Account File Statistics Report Account Profile Report
Gifts and Pledges	Donor Giving Summary Report Gift Summary Report Transaction Summary Report
Interactions	Interaction Summary Report Open Tickler Listing
Planned Giving	Planned Giving Profile Report
Events and Volunteers	Registration and Participation Report
Management Reports	Account Statistics Income Comparison Report Monthly Income Analysis Report Pledge and Paid Statistics Report Revenue Statistics by Marketing Code Revenue Statistics

**Overview of the Query Entry Screen**

You can view and work with queries on the Query Entry screen. You can access the Query Entry screen by selecting Query Entry from the Queries and Outputs module on the Main Menu. You can also access the Query Entry screen by clicking <Queries> from any of the following screens: User-defined Output Entry, Query and Output Schedule, Query Retrieved, and Source Code Entry.

**See Also:**

- [First Page of the Query Entry Screen](#)
- [Second Page of the Query Entry Screen](#)

**First Page of the Query Entry Screen**

**Fields that Describe the Query on the Query Entry Screen**

The top half of the Query Entry screen contains the following fields that describe the query.

Field	Description
-------	-------------

## Team Approach 5.0.1

### Understanding Queries

Field	Description
Query	The name of the query that was assigned when the query was created. You cannot change this name once it is created.
Status	The status of the query (e.g., "A" for active, "C" for changed"). Only Active queries can be scheduled and run. If you edit a query, TA automatically changes the status to Changed. Click on <Test> (and then Run) to activate the query.
Description	The description of the query. You can edit this description at any time.
<a href="#">Query Type</a>	The query type (e.g., transactions, accounts). This field determines the number of records that are returned.
Owner	The brief name of the user who created the query and who can edit and run it.
User Group	The user group that owns the query. All users in this group are also allowed to edit and run the query.

#### Blocks on the Query Entry Screen

On the bottom half of the Query Entry screen, there are three blocks of information that can be used to identify the records the query should select.

Block	Description
Include block	When you add query to this block, you are effectively running a query against another query. Use Includes when you want to select records out the query in the Include block.
Exclude block	The Exclude block contains any queries that you might use as a basis for excluding from other queries. Common examples of exclude queries are deceased names or accounts with invalid addresses.  See " <a href="#">Include and Exclude Queries</a> ".
Selection Criteria block	This is the main query. This block contains a list of conditions to check for in the database. If you have queries in the Exclude block, then TA finds the records you specify in the main query, then excludes those records that are found in any of the exclude queries.  If you have a query in the Include block, TA selects out records from the Include based upon the records selected in the main query.  See " <a href="#">Setting Selection Criteria</a> " for more information.

#### Buttons on the Query Entry Screen

Button	Click to
Page 2	access the Query Entry (Detail) screen.
Copy From	copy over the query with a different query.
Test Query	test that the query is composed correctly, and to generate the SQL statement.
Related	<p>If the query is used as an Include or an Exclude with other queries, this button is enabled. Click on it to see an information window that lists the other queries that reference the query you are currently working with. This window displays information such as Query Names, Related Queries, Owner and when the query was last run.</p> <p>Click on the &lt;Report&gt; button to view or print the information that displays in the Related Queries window.</p>
View Recs	access the Query Retrieved screen and view the records retrieved by the query.
Outputs	access the User-defined Output Entry screen.
Schedule	access the Query and Output Schedule screen.

## Second Page of the Query Entry Screen

The second page of the Query Entry screen is the Query Entry (Detail) screen. You can access this page of the screen by clicking the <Page 2> button on the Query Entry screen.

**Note:** See "[Setting Processing Options on the Query Entry \(Detail\) Screen](#)" for details about these fields.

## Fields that Describe the Query on the Query Entry (Detail) Screen

Field	Description
Query	The name of the query.
Status	The status of the query (e.g., "A" for active, "C" for changed).
Description	The description of the query.
Query Type	The query type (e.g., transactions, accounts)
Owner	The brief name of the user who created the query.
Query Universe	The name of the table or view to use for this query (e.g., "CURRENT_TRANSACTIONS" for active transactions only, or "TRANSACTIONS" for all transactions).

## Fields that Control the Processing Options on the Query Entry (Detail) Screen

The following fields control processing options for the query.

## Team Approach 5.0.1

### Understanding Queries

Field	Option
Effective Date	To set the effective date of the query, which is the date TA uses to determine the correct address to use for each account.
Address Type	To indicate which type of address to return for each record selected by the query.
E-mail Type	To indicate which type of e-mail address to return for each record selected by the query.
Telephone Type	<i>This option will be implemented in future releases of TA.</i>
Source	To link the query to a particular solicitation.
Ask Amount Rules	To have TA calculate the ask amounts for each account pulled.
Special Action?	To have TA include the special action information for the records selected by the query
Private?	To indicate whether the query is private or public. If it set to private, only the owner can view and run the query.
Include/Exclude?	To indicate whether the query can be used as an include query, exclude query, both, or neither.
Edit Retrieved?	To indicate whether you want to let other users edit the records selected by the query.
SQL Edited?	To indicate if the query can continue to use an edited SQL statement after the query is tested.
Expiration Date	To indicate the date on which the records saved to the Query Retrieved table should expire.

#### Information From When the Query Is Tested or Run

TA sets two informational fields on the Query Entry (Detail) screen when the query is tested or run.

Field	Description
# Recs Saved	The Number of Records Saved field displays the number of records saved to the Query Retrieved table when the query was last tested or run. You cannot edit this field.
Date Tested	The Date Tested field displays the date the query was last tested or run. You cannot edit this field.

#### Buttons on the Query Entry (Detail) Screen

There are two buttons on the Query Entry (Detail) screen.

Button	Click to
--------	----------

- Page 1            access the first page of the Query Entry screen.
- View SQL        view the SQL statement once the query has been tested.

## Query Type

The query type identifies the kind of record to select. In other words, of all the types of records in TA, which TA should search to match the criteria you enter. TA includes eight query types described in the following table. Also, you may find the information in "[Identifying Queries and Outputs You Can Schedule Together](#)" helpful.

Query Type	Use	Sample Standard Queries Based on this Query Type
<a href="#">Accounts</a>	To find a list of accounts. This query type does not list individuals within an account who match the criteria.	No Appeals, EFT Members, Renewal Notice #1 Template
<a href="#">Names</a>	To find a list of names within an account.	Eliminator Template
<a href="#">Addresses</a>	To find a list of addresses.	Not applicable
<a href="#">Transactions</a>	To find a list of transactions and the account responsible (i.e., the donor account) for each transaction.	Acknowledgments, Installment Pledge Bills
<a href="#">Benefits</a>	To find a list of benefits awarded to an account and the transaction that awarded the benefit.	Club Mailing Template, Member Card Template, Program Guide Template
<a href="#">Interactions</a>	To find a list of interactions.	Customer Service Missed Response Template
<a href="#">Sources</a>	To find a list of sources.	Not applicable
<a href="#">Tributes</a>	To find a list of tribute transactions and both the account responsible (i.e., the donor account) and the account notified for each transaction.	Not applicable
<a href="#">Planned Giving</a>	To find a list of planned giving records (e.g., all bequests, or all charitable gift annuities)	Not applicable
Registrations	To find records or information displayed on the Registration Entry screen. Also use this query type with <a href="#">Event Registration Procedures</a> .	
Volunteer Availability Volunteer Schedule	To find general or event-related volunteer records or those where a volunteer availability records were created.	

For example, suppose Peter Clark gave three \$1,500 gifts to your organization during the current calendar year.

# Team Approach 5.0.1

## Understanding Queries

---

- If you create a query to find *each account* that gave over \$1,000 during the current calendar year, the query type is "Accounts."
- If you create a query to find *each gift* over \$1,000 received during the current calendar year, the query type is "Transactions."

When each query is run, the first returns one record for Peter Clark's account whereas the second returns three records, one for each gift from Peter Clark.

### See Also:

[Data TA Returns Based on the Query Type](#)

## Data TA Returns Based on the Query Type

When you assign a query type to a query, TA automatically returns a set of data about the records selected by the query.

**Note:** For each query type except sources, TA returns an address. In most cases, TA follows a set of rules to return the correct address such as the address that is active and should be used as of the date you run the query. The address returned is usually the preferred address. TA references "[Address Selection Rules](#)" to find correct addresses based upon seasonal and preferred address contact information. See [Setting a Preferred Address Type](#) for more information. (Note that TA does not follow these rules for address-level queries, because TA allows you to specify your own criteria for selecting addresses when you use the addresses query type.)

## Query Universe

Each query type has at least one query universe associated with it. The query universe determines which table (or view) TA searches when performing the query. For example, when performing a query of the Transactions query type, you can select the "CURRENT\_TRANSACTIONS" view to search for active transactions only, or the "TRANSACTIONS" table to search for all transactions (including inactive ones).

### See Also:

[Accounts Query Type](#)

[Names Query Type](#)

[Addresses Query Type](#)

[Transactions Query Type](#)

[Benefits Query Type](#)

[Interactions Query Type](#)

[Sources Query Type](#)

[Tributes Query Type](#)

[Planned Giving Query Type](#)

## Accounts Query Type

TA includes the following information for each record retrieved by an account-level query.

- The Account ID
- The account's name
- The account's address
- The account's default salutation

## Names Query Type

TA includes the following information for each record retrieved by a name-level query.

- The Account ID
- The account's name
- The person's name
- The person's address
- The person's default salutation

## Addresses Query Type

TA includes the following information for each record retrieved by an address-level query.

- The Account ID
- The address
- The name associated with the address
- The default salutation for the name associated with the address

An example of when you might want to create an addresses query is when you want to output multiple addresses from an account to aid in a merge/purge process. Or you can create an addresses query to give you more flexibility in specifying the addresses you'd like to include in an output file, as TA bypasses the standard [address selection rules](#) with address queries. Note that it generally makes sense to run address-level queries with address-level outputs.

## Transactions Query Type

TA includes inactive transactions if you set the Query Universe field on the query to "TRANSACTIONS." If you set the Query Universe field on the query to "CURRENT\_TRANSACTIONS," TA includes active transactions only.

If you reference the Addresses table in the selection criteria of your transactions query, the join to Addresses can be based only on the Account ID of the transaction. That is, by default the address specified on the transaction is selected and you can change this default. For example, choosing ZIP CODE defaults to ZIP CODE (GIVER), press Ctrl-Alt-F9 to display these alternate join options: ZIP CODE (ACK, ANY ADDRESS); ZIP CODE (ACKNOWLEDGED); ZIP CODE (GIVER); ZIP CODE (GIVER, ANY ADDRESS). If you selected the ZIP CODE (GIVER, ANY ADDRESS) option, the query joins to Addresses only on Account ID, thus considering any address on the account, not just the address on the transaction.

When you choose an alternate join option for a criteria line, you see a message such as: "Should all Address fields in this query be for the Giver, Any Address?", with options <Yes> and <No>. Based on your selection, TA sets the join option on other criteria lines from the same table.

TA includes the following information for each record retrieved by a transaction-level query.

- The Account ID
- The name associated with the transaction
- The address
- The salutation
- The gift date
- The pledge amount

## ***Team Approach 5.0.1***

### ***Understanding Queries***

---

- The payment amount
- The source
- The ask amounts

If you set up the query to pull the special action information

- The special action code
- The special action comments

### **Benefits Query Type**

TA includes the following information for each record retrieved by a benefit-level query.

**Note:** TA includes inactive benefits if you set the Query Universe field on the query to "TRANSACTION\_BENEFITS." If you set the Query Universe field on the query to "CURRENT\_TRANS\_BENEFITS," TA includes active benefits only.

For the benefit:

- The Account ID of the benefit recipient
- The name of the benefit recipient
- The address of the benefit recipient
- The salutation for the benefit recipient
- The benefit code
- The start and expire dates, if applicable

For the gift that awarded the benefit:

- The gift date
- The pledge amount
- The payment amount
- The source code

### **Interactions Query Type**

TA includes the following information for each record retrieved by an interaction-level query.

- The Account ID
- The name associated with the interaction
- The address

- The salutation
- The interaction date
- The interaction sequence number
- The interaction page number
- The source code for the interaction

## Sources Query Type

TA includes only the source code for each record retrieved by a source-level query.

## Tributes Query Type

TA includes the following information for each record retrieved by a tribute-level query.

- Name of donor
- Address of donor
- Donor's salutation
- Tribute transaction information
- Account ID of notified party
- Name of notified party

## Planned Giving Query Type

TA includes the following information for each record retrieved by a planned giving-level query.

- The Account ID
- The name associated with the planned gift record
- The address associated with the planned gift record
- The default salutation for the name associated with the planned gift record
- The planned gift sequence, i.e., the number that uniquely identifies a specific planned gift for an account (not currently displayed on the Query Retrieved Detail window)

## Selection Criteria

The selection criteria are the first (and sometimes only) step in determining which records the query should select. By entering criteria lines in the Selection Criteria block of the Query Entry screen, you specify a list of conditions to check for in the database (i.e., fields in the database and their values). You indicate the value for which you are searching by using an operator and either a specific value or a list of values.

You can enter as many criteria lines as needed to select the appropriate records. A selection criterion line is composed of three parts.

## Team Approach 5.0.1

### Understanding Queries

---

- The query field name
- The operator
- The value

For more information about working with selection criteria, see "[Setting Selection Criteria](#)".

### Include and Exclude Queries

Include and exclude queries are queries that can be used within a query to help the query select records.

#### See Also:

[Include Queries](#)

[Exclude Queries](#)

### Include Queries

An include query is an active query that has been identified for use in the Include block on the Query Entry screen. Include queries can help you build complex queries in that they enable you to compartmentalize complicated sets of selection criteria without you having to organize all of the criteria in one Selection Criteria block. Include queries are also used when you want your query to select from different types of records (e.g., using an interactions-type include query within an accounts-type query).

For example, suppose you want to do a special renewal mailing for your annual members who are also prospects for a capital campaign. You can do this by writing a benefits-level query to find the active anchor benefits (e.g., Benefit Assigned = "MEM" and Expiration Date (Account Activities) (Recipient's) <= "Effective Date"). Within this query, you can use an accounts-level include query to find a group of accounts being cultivated for the capital campaign (e.g., Project = "C" for capital campaign, Stage = "CU" for cultivation, and Ask Amount (Account Projects) >= 5000).

If you enter an include query in the Include block, TA runs this query, compares the results with the selection criteria entered in the Selection Criteria block and any other include queries entered in the Include block, and selects only those records that meet *both* the criteria represented by the query or queries in the Include block *and* the criteria entered in the Selection Criteria block.

**Note:** Include queries are generally only used in complex queries. For example, you might want to use an include query if the criteria used in the include query take up more than two or three lines. However, if the criteria only involve one or two lines, your query will probably run faster if you enter this information directly in the Selection Criteria block.

When you create a query, you can enter into the Include block one or more active queries to which you have access *and* for which the Include/Exclude field is set to either "Include" or to "Both." You have access to all of the [queries you own](#) and to those that have been marked as public. See "[Marking the Query as an Include or Exclude Query](#)".

#### See Also:

[Union and Intersection Fields for Include Queries](#)

[Using Saved Records from Include Queries](#)

### Union and Intersection Fields for Include Queries

If you enter more than one query in the Include block, you need to specify how TA handles the sets of records selected by the queries (i.e., whether TA uses every record selected by the include queries, or just those records common to all of the include queries). You can enter one of the following two operators.

UNION	Use every record selected by all of the include queries, eliminating any duplicate records. For example, you enter two queries in this block. The first query selects 1,000 records, and the second query selects 2,500 records. Of the 3,500 total records, 100 of the same records are selected by both queries. TA selects a total of 3,400 records.
INTERSECTION	Use only those records common to all of the include queries. For example, you enter three queries in this block. If a record appears in the set of records selected by the first query, the second query, and the third query, use this record. If a record appears only in the set of records selected by one or two of the queries, do not use the record.

**Note:** Union is equivalent to "OR" in the selection criteria and Intersection is equivalent to "AND."

When you create a new query, TA defaults the Union/Intersection field to "UNION."

## Using Saved Records from Include Queries

When you use an include query, you can specify that you want to use the saved records from that include query. Using the saved records from an include query may save processing time when you are running many queries that use the include query. In addition, you can use this feature, for example, to ensure that two distinct mailings are sent to the same set of accounts. You could complete the following steps to use a list of invitees for both the "save-the-date" mailing and the formal invitation to the event.

1. Two months before an event, you create a query to select the accounts that you want to invite to the event.
2. You set the expiration date of the records selected to the date of the event. See "[Setting an Expiration Date for the Retrieved Records](#)".
3. You schedule this query to run because you want to send this group a save-the-date notice. When you schedule this query, you don't schedule a user-defined output to run with the query because you want to review the list first before sending the save-the-date notice; instead, you indicate that you want TA to save the records.
4. The query runs.
5. You review (and modify) the list of accounts selected by the query.
6. You run the appropriate user-defined output with the saved records (i.e., the modified list of accounts) to send the save-the-date notice to this group of people.
7. In the next week, you receive notification from some people that cannot attend.
8. When you create the query to send the invitation several weeks later, you would include the query used for the save-the-date notice and mark, in the Include block, the checkbox to the left of the save-the-date query to use the saved records from that query. You could also exclude the people who have an interaction saying they cannot attend.

## Exclude Queries

An exclude query is an active query that has been identified for use in the Exclude block on the Query Entry screen. Exclude queries identify the records that TA should remove from the set of records selected by the selection criteria and include queries. That is, of all the records currently selected by the selection criteria and include queries, *exclude* those records that meet the criteria of the query or queries in the Exclude block on the Query Entry screen.

A set of standard exclude queries is delivered with TA. A common use of the exclude query feature is to exclude invalid records. For example, one of TA's standard queries finds all accounts with invalid addresses. Your organization can use the invalid address query as an exclude for each query your organization creates to select

# Team Approach 5.0.1

## Understanding Queries

---

accounts that should receive a mailing, i.e. all accounts with invalid addresses are removed from the set of accounts selected for the mailing.

When you create a query, you can enter into the Exclude block one or more queries to which you have access *and* for which the Include/Exclude field is set to either "Exclude" or to "Both." You have access to all of the [queries you own](#) and to those that have been marked as public. See "[Marking the Query as an Include or Exclude Query](#)".

### See Also:

[Union and Intersection Fields for Exclude Queries](#)

[Using Saved Records from Exclude Queries](#)

## Union and Intersection Fields for Exclude Queries

If you enter more than one query in the Exclude block, you need to specify how TA handles the sets of records selected by the multiple queries (i.e., whether TA uses every record selected by the exclude queries, or just those records common to all of the exclude queries). You can enter one of the following two operators.

UNION	Use every record selected by all of the exclude queries, eliminating any duplicate records. For example, you enter two queries in this block. The first query selects 1,000 records, and the second query selects 2,500 records. Of the 3,500 total records, 100 of the same records are selected by both queries. TA selects a total of 3,400 records.
INTERSECTION	Use only those records common to all of the exclude queries. For example, you enter three queries in this block. If a record appears in the set of records selected by the first query, the second query, and the third query, use this record. If a record appears only in the set of records selected by one or two of the queries, do not use the record.

**Note:** Union is equivalent to "OR" in the selection criteria and Intersection is equivalent to "AND."

When you create a new query, TA defaults the Union/Intersection field to "UNION."

## Using Saved Records from Exclude Queries

When you use an exclude query, you can specify that you want to use the saved records from that exclude query. For example, you can use this feature to save processing time when running a series of renewal queries. That is, you can run a set of exclude queries once and then use the saved records with each of the renewal queries in your series.

1. Create the queries that you will run in your renewal series.
2. For each of the queries in your renewal series, enter the exclude queries that you want to use in the Exclude block, and mark the checkboxes to the left of each of the exclude queries to use the saved records from those queries. For example, you might want to use the following exclude queries: TA INVALID ACCOUNTS, TA NO MAIL APPEALS, and TA NO RENEWAL APPEAL.
3. Schedule the exclude queries you want to use for your renewals (e.g., TA INVALID ACCOUNTS, TA NO MAIL APPEALS, and TA NO RENEWAL APPEAL).

**Note:** Be sure to indicate that you want TA to save the records from each of these queries.

4. Schedule each of the renewal queries to run after the exclude queries.
5. TA runs the exclude queries and saves the records.

- TA runs each query in the renewal series. From each renewal query, TA removes the records that were saved by the exclude queries.

## Address Selection Rules

For each record selected by a query, TA tries to select the appropriate address based on a set of address selection rules. TA references the value in the Address Type field on the Query Entry (Detail) screen when applying these rules. See [Setting a Preferred Address Type](#).

### See Also:

[Expected Arrival Date](#)

[Selection Rules Application](#)

## Expected Arrival Date

In TA, you can assign a date range during which an address is valid and you can indicate that the address should be used for mailings during that time period. When your organization runs a query for a mailing, the query must have a date on which the mailing is expected to arrive at the address so that TA does not select an address that is not valid for that arrival date. TA determines the expected arrival date as follows:

- If you have entered the effective date of the query, TA sets the expected arrival date to the effective date.
- If you have not entered the effective date of the query but have scheduled the query to run, TA sets the expected arrival date to the scheduled run date for the query. See the [Scheduling and Viewing a Query and Output chapter](#) to learn how to set the scheduled date for a query.

## Selection Rules Application

When your organization sets up a query for a mailing using any query type except addresses or sources, the query type indicates whether the mailing is being sent to the account or to a particular person within the account. TA applies Rules 1 and 2 to all queries to select a valid address for each account or name selected by the query. TA then applies Rules 3 through 6 as needed. That is, if TA does not select an address by applying Rule 3, TA applies Rule 4. If TA selects an address by applying Rule 4, TA does not apply the remaining rules. In addition, if more than one address meets the rule, TA selects the first address it finds that meets the rule.

**Note:** When you use the addresses query type, TA does not apply these address selection rules, and instead allows you to specify your own criteria for picking addresses. TA also does not apply the address selection rules for queries using the sources query type, because no address is returned by sources-level queries.

### Rule 1: Select Active, Mailable Addresses

TA selects only from those addresses that are active (i.e., the Status field for the address is set to "A" for active). Five statuses are available for an address: active, delete, inactive, old, and undeliverable.

In addition, TA selects only from those addresses that do not have the "do not mail" flag set (i.e., the Do Not Mail? field for the address is blank or set to "N" for no). You can mark an address as one that should not receive any mail by setting the Do Not Mail? field to "Y" for yes.

### Rule 2: Select Valid Addresses for the Account or Name

Given the addresses TA selected based on Rule 1, TA only selects those addresses associated with the selected account or the selected name. That is, for each address, you can designate that the address belongs to the account as a whole or to a specific name within the account. If your query is selecting accounts, then TA selects only those addresses assigned to the account as a whole. If your query is selecting names, then TA selects only those addresses assigned to that name.

TA uses Rules 1 and 2 to filter out addresses that cannot be used for the query. For example, an account may have six addresses. Of the six, one has a status of undeliverable, one has a status of old, two are associated with the

# Team Approach 5.0.1

## Understanding Queries

---

account as a whole, and the remaining two are each associated with a different person in the account. TA "throws out" the undeliverable and old addresses based on Rule 1. If the query is searching for accounts, TA throws out the last two addresses because each is associated with a particular person. If the query is searching for names, TA throws out the addresses associated with the account as a whole. If more than one address remains, TA selects the appropriate one based on Rules 3 to 6.

### Rule 3: Select the Address Matching the Query's Address Type and Expected Arrival Date

TA selects the address that matches the address type specified in the query and that is valid for the expected arrival date. TA determines whether the address is valid for the expected arrival date by checking the address' date range as follows.

- If a date range is assigned to the address, the expected arrival date falls within this range, and the Use During These Dates? checkbox is marked, then TA selects this address.
- If no date range is assigned to the address, then TA selects this address.

### Rule 4: Select the Address Stored on the Record Selected by the Query

If the record the query selects includes an address, TA uses that address.

### Rule 5: Select the Preferred Address If No Address Is Selected

If no address meets any of the selection criteria for Rules 3 and 4, TA selects the preferred address for the account if the query is selecting accounts, or for the name if the query is selecting names.

### Rule 6: Do Not Mail If No Address Is Selected

If TA cannot find a valid address to select and one or more address fields are used in the user-defined output, the record will be selected by the query but not included in the output file.

### Choosing to Include Invalid Addresses

You may want to include invalid addresses in your output file, e.g., if you run a query to select accounts for which you have only inactive or undeliverable addresses and you want to output the address. You can choose to include invalid addresses when you schedule your query and output by marking the Include Invalid Addresses? checkbox on the File Output Options window, which you access by zooming from the Output Name field on the Query and Output Schedule screen. When you mark the Include Invalid Addresses? checkbox, TA broadens the address selection rules to include addresses marked Do Not Mail as well as those containing any status except "D" for delete.

## System Preferences (Queries)

The following table lists each system preference that affects queries you create or update. See the [Setting System Preferences chapter](#) for an explanation of each system preference.

System Preference	See
Days Until Queried Records Expire	<a href="#">Setting an Expiration Date for the Retrieved Records</a>
Default Copy Query	<a href="#">Creating a Query</a>
Prefix for Source Code Queries	<a href="#">Query Naming Convention</a> <a href="#">Source Code Queries</a>

## Special Keys (Query Entry Screen)

When you use the Query Entry screen, selected actions require the use of a specific key or key combination from the Include block, the Exclude block, the Query Field Name field, and the Values field.

### See Also:

[Special Keys in the Include Block and Exclude Block](#)

[Special Keys in the Query Field Name Field](#)

[Special Keys in the Values Field](#)

## Special Keys in the Include Block and Exclude Block

The following table lists the key(s) you need to press to perform a particular task from either the Include block or the Exclude block.

The action or task you want to perform	The key(s) to press
View the query. See " <a href="#">Zooming to View the Details of an Include or Exclude Query</a> "	[F10]
Remove a query from the block. See " <a href="#">Adding and Removing Include Queries</a> " and " <a href="#">Adding and Removing Exclude Queries</a> "	[Shift] [F6]

## Special Keys in the Query Field Name Field

The following table lists the key(s) you need to press to perform a particular task from the Query Field Name field.

The action or task you want to perform	The key(s) to press
Create a new line to enter a new criterion within a set of existing criteria. See " <a href="#">Adding a Selection Criterion Within the Set of Criteria</a> "	[Ctrl] [T]
Delete a criterion. See " <a href="#">Deleting a Selection Criterion</a> "	[Shift] [F6]
View the list of queryable fields. See " <a href="#">Working with Queryable Fields</a> "	[F9]
View the related fields for the queryable field you selected when query type is benefits or transactions. See " <a href="#">Changing Which Account is Referenced</a> "	[Ctrl] [Alt] [F9]

## Special Keys in the Values Field

The following table lists the key(s) you need to press to perform a particular task from the Values field.

The action or task you want to perform	The key(s) to press
Create a new line to enter a new criterion within a set of existing criteria. See " <a href="#">Adding a Selection Criterion Within the Set of Criteria</a> "	[Ctrl] [T]
Create a new values list for the queryable field selected in the Query Field Name field. See " <a href="#">Creating a Values List</a> "	[F10]
Delete a criterion. See " <a href="#">Deleting a Selection Criterion</a> "	[Shift] [F6]

# Team Approach 5.0.1

## Understanding Queries

---

---

The action or task you want to perform	The key(s) to press
Indicate that the value to be entered in the Values field should remain in mixed case. See " <a href="#">Indicating the Use of Mixed Case Value</a> "	[F10]
Indicate how to treat a null value for the queryable field selected in the Query Field Name field. See " <a href="#">Special Treatment of Nulls for a Selection Criterion</a> "	[F10]
Update one or more values lists for the queryable field selected in the Query Field Name field. See " <a href="#">Updating a Values List</a> "	[F10]
View the list of literals for the queryable field selected in the Query Field Name field. See " <a href="#">Values Lists</a> "	[F9]
View the list of fields for the queryable field selected in the Query Field Name field.	[Alt] [F9]
View the values for a values list entered in the Values field.	[F10]

### Queries Step-by-step

[Finding a Query](#)

[Working with User Groups \(Queries\)](#)

[Creating a Query](#)

[Working with Include and Exclude Queries](#)

[Setting Selection Criteria](#)

[Setting Processing Options on the Query Entry \(Detail\) Screen](#)

[Viewing and Editing the SQL Statement](#)

[Testing a Query](#)

[Reviewing the Query's Purpose, Schedule, or Related Output](#)

[Deleting a Query](#)

### Finding a Query

To find a query, complete the following steps.

1. Access the Query Entry screen.
  - If you were not working with a query previously, TA displays the Find Queries window.
  - If you were working with a query during your current session, TA displays the query with which you were working on the Query Entry screen. To find a different query, click <Find> on the TA toolbar. TA displays the Find Queries window.

2. On the Find Queries window, complete the fields that you want to use to specify which query to display (see table below).
 

**Note:** The query must meet all of the criteria you specify in the Find Queries window, so only set those fields that you want to use to limit the queries TA selects. You can press the [F9] key for a list of values in all of the fields.
3. Click <Find>. TA displays the Query Entry screen with the query you selected. If you selected a set of queries, TA displays the first query of the set of queries you selected. You can move from one query to the next by clicking <Prev> or <Next> on the toolbar as appropriate

### Fields on the Find Queries Window

Field	Option
Query	<p>You can find queries based on the name of the query. You can enter the query name or press the [F9] key for a list of queries to which you have access.</p> <p><b>Note:</b> You can also use the percent sign ("%") as a wildcard character in the query name to find a set of queries. The percent sign represents all characters following those you enter for the query name. For example, you can enter "TA ACK%" to find all of the standard TA acknowledgment queries. You can enter "TA%REMINDER%" to find the standard TA bill queries such as the pledge bill query and the installment bill query.</p>
Query Type	You can find queries based on the query type. You can enter the type or press the [F9] key for a list of query types.
Owner	You can find queries based on the user who created the query. You can enter the brief name of the owner or press the [F9] key for a list of brief names.
User Group	You can find queries based on the user group that owns the query. You can enter the user group code or press the [F9] key for a list of user groups to which you belong.

**Note:** The queries selected do not include other users' private queries.

See "[Query Type.](#)" See "[Query Privileges.](#)" See "[Marking the Query as Public or Private.](#)"

### Working with User Groups (Queries)

A user group can be assigned to a query to allow multiple users to ["own" the query.](#) You can work with a user group on the User Group Entry screen.

#### Creating a User Group

1. Access the User Group Entry screen. TA displays the Find User Groups window.
2. Click <New> to access the Create New User Group window. You can enter a name up to 12 characters in length including spaces, underscores, and periods. This name must be unique.
3. Click <Create>. TA displays the user group on the User Group Entry screen.
4. Enter a description of the user group in the Description field.
5. Assign members to the user group.

# Team Approach 5.0.1

## Understanding Queries

---

### Assigning Members to a User Group

Once a user group has been created, you can assign members to it.

1. In the User Assignments block, click on the first blank line.  
**Note:** If no blank lines appear, highlight any user name in the User Assignments block and press the [F6] key or click the <New> button on the toolbar to add a blank line.
2. Enter the brief name of the user to assign to the group. You can press the [F9] key to select from the list of user names.
3. Press the [Tab] key. TA displays the user's name line to the right of the user's brief name.
4. Repeat steps 1 through 3 for each additional user that you want to assign to the group.  
**Note:** You can remove a user from a user group by highlighting their brief name in the User Assignments block and pressing [Shift] [F6] key combination.

### See Also:

[Finding a User Group \(Queries\)](#)

### Finding a User Group (Queries)

1. Access the User Group Entry screen. TA displays the Find User Groups window.
2. Enter the name of the user group in the User Group field. You can press the [F9] key to select from the list of user groups.  
**Note:** You can use the percent sign ("%") as a wildcard character in the User Group field to find a set of user groups. The percent sign represents all characters following those you enter for the user group name.
3. Click <Find>.

### Creating a Query

You can create a query from scratch, or you can create a query by copying an existing query. You can copy any [query that you own](#) or any public query owned by another user, regardless of the status of the query. See "[Marking the Query as Public or Private](#)".

To create a new query, complete the following steps.

1. Access the Query Entry screen.
  - If you were not working with a query previously, TA displays the Find Queries window from which you should click <New> to access the Create New Query window.
  - If you were working with a query during your current session, TA displays the query with which you were working on the Query Entry screen. To create a new query, click <New> on the toolbar to access the Create New Query window.
2. Enter the name of the new query in the New Query Name field. This name must be unique. You can enter a name up to 20 characters in length including spaces, underscores, and periods.
3. Enter the query type in the Query Type field. You can press the [F9] key for a list of query types.

4. If you want to base this new query on an existing query, enter that query name in the Copy Existing Query field, otherwise leave this field blank.

**Note:** If your organization set the [Default Copy Query system preference](#), then TA defaults this field to the name of an existing query. Follow your organization's policy as to whether to delete the name of this query or not.

5. Click <Create>.
6. The new query appears on the Query Entry screen; you can enter the appropriate include queries, exclude queries, and selection criteria for this new query.

**Note:** If you try to create a query using the name of an existing query, TA displays a standard message that asks if you want to view the existing record or create a new one.

## See Also:

[Copying Over Your Query with an Existing Query](#)

## Copying Over Your Query with an Existing Query

If you create a query and then decide that you want to base the new query on an existing query, you can replace your new query with a copy of the query you identify. You can then modify the fields as you choose.

To copy an existing query, complete the following steps.

1. Access the query you want to replace on the Query Entry Screen. See "[Finding a Query](#)".
2. Click <Copy From> to access the Enter Query Name to Copy From window.
3. Enter the name of the query you want to copy. You can press the [F9] key to view the alphabetical list of queries you can copy. These queries consist of those queries marked as public and those [queries you own](#).
4. Click <OK>.
5. On the Query Entry screen, TA sets the include queries, exclude queries, and selection criteria for this new query based on the query you copied. You can modify the query.

**Note:** TA overwrites any include queries, exclude queries, and selection criteria you entered with those from the query you copy.

## Working with Include and Exclude Queries

You can do the following with include and exclude queries on the Query Entry screen:

- Add include or exclude queries
- Remove include or exclude queries
- Specify whether previously retrieved records should be used for an include or exclude query
- Zoom to view the details of the include or exclude query
- If you are viewing a query that is referenced in other queries (either as an Include or an Exclude) the **Related** button is enabled and you can view an information window that list the related queries.

In the Related Queries window you can click on the Reports button to view or print the **Related Queries report**. (This report displays the same information that is in this window.) You can also run the Related Queries report from the Main Menu for one or multiple queries where you want to view this information.

See "[Include Queries](#)". See "[Exclude Queries](#)".

# Team Approach 5.0.1

## Understanding Queries

---

### See Also:

[Adding and Removing Include Queries](#)

[Adding and Removing Exclude Queries](#)

[Using Previously Retrieved Records for an Include or Exclude Query](#)

[Zooming to View the Details of an Include or Exclude Query](#)

## Adding and Removing Include Queries

### Adding an Include Query

You can enter the name of one or more queries to be used as includes in the Include block. To view the list of queries that you can include, press the [F9] key from the Include field in the Include block. TA displays the Query Sets to Include window. Select the query you want to include, and click <OK>.

**Note:** The list on the Query Sets to Include window consists of those queries that have been designated for use as an include query; the list may contain public queries and [queries you own](#). The list does not contain other users' private queries that have been designated for use as an include query. If you do not see the name of the query you want to use as an include, follow your organization's policy regarding changing the designation of the query.

See "[Marking the Query as Public or Private](#)." See "[Marking the Query as an Include or Exclude Query](#)." See "[Query Privileges](#)."

### Removing an Include Query

You can remove an include query from the Include block, e.g., if you select the incorrect query to include, by highlighting the query and pressing the [Shift] [F6] key combination.

## Adding and Removing Exclude Queries

### Adding an Exclude Query

You can enter the name of one or more queries to be used as excludes in the Exclude block. To view the list of queries that you can exclude, press the [F9] key from the Exclude field in the Exclude block. TA displays the Query Sets to Exclude window. Select the query you want to exclude, and click <OK>. The records selected by the exclude queries are removed from those records selected by the selection criteria and include queries.

**Note:** The list on the Query Sets to Exclude window consists of those queries that have been designated for use as an exclude query; the list may contain public queries and [queries you own](#). The list does not contain other user's private queries that have been designated for use as an exclude query. If you do not see the name of the query you want to use as an exclude, follow your organization's policy regarding changing the designation of the query.

See "[Marking the Query as Public or Private](#)." See "[Marking the Query as an Include or Exclude Query](#)." See "[Query Privileges](#)."

### Removing an Exclude Query

You can remove an exclude query from the Exclude block, e.g., if you select the incorrect query to exclude, by highlighting the query and pressing the [Shift] [F6] key combination.

## Using Previously Retrieved Records for an Include or Exclude Query

You can mark the checkbox to the left of an include query in the Include block or an exclude query in the Exclude block to indicate that you want to use the saved records from that query rather than running the query again.

## Zooming to View the Details of an Include or Exclude Query

After you have selected a query to include or exclude, you can view the details of the query by clicking the <Zoom> button to the left of the query name. TA displays the Include/Exclude Zoom window, which enables you to view the include or exclude query's criteria without leaving the query with which you are working. From the Include/Exclude Zoom window, you cannot update any information about the include or exclude query. To close the window and return to the Query Entry screen, click <Close>.

## Setting Selection Criteria

The selection criteria are the first (and sometimes only) step in determining which records the query should select. By entering criteria lines in the Selection Criteria block of the Query Entry screen, you specify a list of one or more fields that should be searched in each of the records selected by the query type, and the value for which you are searching in each field. You indicate the value for which you are searching by using a comparison operator and either a specific value or a list of values.

You can enter as many criteria lines as needed to select the appropriate records.

A selection criterion line is composed of three parts.

- The query field name
- The operator
- The value

If you enter more than one selection criterion, you must indicate how to combine the criteria.

When entering selection criteria, the basic steps are as follows:

1. Enter a field name in the Query Field Name field. You can press the [F9] key for a list of queryable fields.
2. Enter an operator in the Operator field. You can press the [F9] key for a list of operators.
3. Enter a value (or values) in the Values field. Depending on the field with which you are working, you may be able to press the [F9] key for a list of values.
4. If you want to add additional selection criteria, enter "And," "Or," or "+Not" in the And/Or field.
5. Repeat steps 1 to 4 on the next blank line for each selection criteria you enter.

For example, to find each account that has a current giving level amount over \$1,000 *and* is also interested in education, you would enter the following criteria using the connector "And."

Query Field Name	Operator	Field Value	Connector
CURRENT GIVING LEVEL AMOUNT	>	1000	AND
INTEREST CODE	=	EDUCATN	

You could also find each account that has a current giving level amount over \$1,000 *or* is interested in education (or meets both criteria) by changing the connector to "Or."

Query Field Name	Operator	Field Value	Connector
CURRENT GIVING LEVEL AMOUNT	>	1000	OR
INTEREST CODE	=	EDUCATN	

**See Also:**

# Team Approach 5.0.1

## Understanding Queries

---

[Working with Queryable Fields](#)

[Selecting an Operator and Entering a Value](#)

[Connecting Multiple Selection Criteria](#)

[Adding to and Deleting from a Set of Selection Criteria](#)

### Working with Queryable Fields

For each selection criterion you set up, you must select a queryable field that TA evaluates to select records. The fields from which you select are tied to the query type. To view the list of fields appropriate for the query type, press the [F9] key in the Query Field Name field. The list includes the following two types of fields: database fields and calculated fields.

### See Also:

[Database Fields](#)

[Calculated Fields](#)

[Parenthetical Information in the List of Queryable Fields](#)

### Database Fields

Database fields are stored in the database tables. The value for a database field is either set by you using a TA entry screen or calculated by TA based on information you enter into TA. For example, TA sets the total additional gifts the donor has given this calendar year, an account activity field, by summing each additional gift for this calendar year that you enter for the account.

### Calculated Fields (Query & Outputs)

All calculated fields begin with the "at" sign ("") to distinguish them from the database fields. The ability to create calculated fields is a powerful feature of TA that allows for great extensibility of the query tool. A calculated field:

- Performs a calculation on database fields to create a value not stored in the database. The value of a calculated field is calculated at the time the query runs.
- Selects a particular database field based on customized criteria.

In general, the calculated fields based on customized criteria are used in user-defined outputs but are also available for queries. For more information about creating new calculated fields or definitions of existing fields, please see ['About Calculated Fields.'](#)

The queryable field "Pledge Payment Due" is an example of a calculated field that performs a calculation on database fields at the time the query is run; Pledge Payment Due represents the amount due on a pledge transaction.

The queryable field "Current Mem Yr Amt Total" is an example of a field based on customized criteria; Current Mem Yr Amt Total sums the total of a donor's giving to the "R" activity type, the "T" activity type, and the "X" activity type. If your organization does not use these three activity types, your organization should not use this queryable field.

### Parenthetical Information in the List of Queryable Fields

TA may include parenthetical information following the field name for some fields included in the list of queryable fields. One of the following kinds of information is noted in the parentheses.

- The information identifies the specific record that will be selected by the query.
- The information identifies the database table that the query will reference when a field appears in multiple tables.
- The information identifies the type of record.

The Last Fiscal Year Paid Amount field is an example of a calculated field that has been programmed to sum the total payment amount for the previous fiscal year for the annual giving activity type (i.e., activity type = "A") and the matching gift activity type (i.e., activity type = "G"). In the list of queryable fields, this field appears as "LAST FISCAL YEAR PAID (ANNUAL AND MATCHING)." If your organization does not use these two activity types, your organization should not use this calculated field.

The Account ID field is an example of a database field included in many database tables. When the Account ID field is included in the list of queryable fields, the table name, such as Accounts, Names, or Transactions, is included in parentheses.

If your organization uses attributes to help define benefits and/or service providers, or has a list of questions that appear on the Inquiry and Referral screen, you will see these fields followed by the identifiers Benefit Attributes, Service Provider Attributes, or Questionnaire.

See the Understanding Calculated Fields chapter for an explanation of the calculated fields delivered with TA.

**Note:** When the query type for the query is set to "Benefits" or to "Transactions," TA may add parenthetical information to the right of the field name for some fields you can select.

## See Also:

[Parenthetical Information in Benefits Queries](#)

[Parenthetical Information in Transactions Queries](#)

[Changing Which Account is Referenced](#)

## Parenthetical Information in Benefits Queries

For the Benefits query type, the query returns benefit information for awarded benefits. An awarded benefit has information related to the account of the giver of the benefit and the account of the recipient of the benefit. Most of the time the giver and the recipient are the same, but sometimes, as with a gift membership, they are not.

For a benefits query, when you select a database field that relates to account or transaction information, TA adds "(RECIPIENT'S)" as the suffix to the field name to indicate that by default TA will return the recipient's information rather than the giver's information. For example, if you select the field "STREET NAME" from the list of queryable fields, TA displays this field as "STREET NAME (RECIPIENT'S)" to indicate that this criterion will be based on the street name of the address for the benefit recipient.

TA adds "(BENEFIT ATTRIBUTE)" as the suffix to the field name if the field represents an attribute that helps define the benefit.

## Parenthetical Information in Transactions Queries

For the Transactions query type, the transaction information includes the account of the giver of the gift and the account acknowledged for the gift. Most of the time the giver and the account acknowledged are the same, but sometimes they are not (e.g., soft credit transactions).

For this query type, TA adds "(GIVER'S)" as the suffix to the field name to indicate that by default TA will return the giver's information rather than the acknowledged's information. For example, if you select the field "STREET NAME" from the list of queryable fields, TA displays this field as "STREET NAME (GIVER'S)" to indicate that this criterion will be based on the street name of the address for the giver.

## Changing Which Account is Referenced

For benefits and transactions queries, you can change which account TA references, e.g., you want TA to evaluate the giver information instead of the recipient information for a benefit query.

- For benefits queries, the related fields are "GIVER'S" and "RECIPIENT'S."

# Team Approach 5.0.1

## Understanding Queries

---

- For transactions queries, the related fields are "GIVER'S" and "ACKNOWLEDGED'S."

To change which account is referenced, complete the following steps.

1. Highlight the queryable field.
2. Press the [Ctrl] [Alt] [F9] key combination to access the Related Fields window.
3. The Related Fields window lists the fields from which you can select. The pairs of related fields are "GIVER'S" and "RECIPIENT'S" for a benefits query, and "GIVER'S" and "ACKNOWLEDGED'S" for a transaction query.
4. Select the appropriate field.
5. Click <OK> to return to the Query Entry screen.
6. The queryable field you selected appears in the Query Field Name field.

### Selecting an Operator and Entering a Value

Once you have selected the queryable field that you want TA to evaluate, you need to enter the operator and value for the field. This operator identifies how to compare each record's value for the selected field to the value(s) you enter in the Values field on the Query Entry screen.

On the Query Entry screen, you can press the [F9] key from the Operator field to see the operators. TA includes six mathematical operators and eight comparison operators.

### See Also:

[Using a Mathematical Operator](#)

[Using the Comparison Operators BETWEEN and NOT BETWEEN](#)

[Using the LIST Operators \(IN LIST, NOT IN LIST, and LIST CONTAINS\)](#)

[Using the Comparison Operators IS NULL and IS NOT NULL](#)

[Using the Comparison Operators LIKE and NOT LIKE](#)

[Indicating the Use of Mixed Case Value](#)

[Using a Parameterized Value](#)

### Using a Mathematical Operator

The six mathematical operators and their definitions are:

Operator	Definition
=	Equal to the value in the Values field
>=	Greater than or equal to the value in the Values field
<=	Less than or equal to the value in the Values field
>	Greater than the value in the Values field
<	Less than the value in the Values field
<>	Not equal to the value in the Values field

TA selects each record with a value in the queryable field that meets the criterion. If the field is blank for the record being evaluated, TA does not select the record unless you "assign a meaning" to that blank value; see "[Special Treatment of Nulls for a Selection Criterion](#)".

When you use a mathematical operator, you must enter into the Values field either a literal value or a field name.

## See Also:

- [Entering a Literal Value](#)
- [Entering a Field Name](#)
- [Viewing the List of Valid Literal Values and Fields](#)
- [Entering a Date](#)
- [Special Treatment of Nulls for a Selection Criterion](#)

## Entering a Literal Value

When you use a mathematical operator, you can enter a literal value into the Values field. The phrase "literal value" refers to a valid value for the queryable field you selected. If you selected a database field, then the value must be one that can be stored in the field. If you selected a calculated field, then the value must be one that TA could calculate for the field.

- For a validated field, the literal values are the valid codes you can enter for the field. The phrase "validated field" refers to a field for which you can view a list of values.
- For numeric fields, the literal value is a valid number for the field.
- For alphanumeric (text) fields, the literal value is the text that might be stored in the field.
- For date fields, the literal value is a date. See "[Entering a Date](#)".

When entering a literal value, you should understand what values are valid for the queryable field you selected. TA's query facility does not prevent you from entering a literal value that is invalid for the field.

**Note:** For any literal value you enter into the Values field, TA uppercases the value when you exit the field. See "[Indicating the Use of Mixed Case Value](#)".

For example, suppose you create an accounts query (the query type is Accounts) and select "ACCOUNT CATEGORY" as the queryable field to evaluate with "=" as the operator. You could enter a literal value as the field value; in this situation, the valid literal values are the account category codes: "I" for Individual/Family, "C" for Corporation, "F" for Foundation, "G" for Government, and "O" for Other. You could set up the criterion "ACCOUNT CATEGORY = I" to select accounts assigned to the Individual/Family account category.

**Note:** In this example, TA's query facility does not prevent you from entering a "V" in the Values field where "V" is not a valid account category code. For validated fields, you can press the [F9] key to view the list of valid literal values to ensure that the literal value you enter is valid for the queryable field. See "[Viewing the List of Valid Literal Values and Fields](#)".

## Entering a Field Name

When you use a mathematical operator, you can enter a field name into the Values field. The field name refers to a database field that is tied to the query type of the query. All fields that can be used as a value in the Values field are bracketed using curly brackets, e.g., {PAYMENT AMOUNT (GIVER'S)}, so that you can easily distinguish a field name from a literal value.

For example, suppose you create a transactions query (the query type is Transactions) and select "PLEDGE AMOUNT" as the field to evaluate with ">=" as the operator. You could enter a field name as the field value. In this situation, the field "PAYMENT AMOUNT (GIVER'S)" is a valid queryable field for a transactions query. You could set up the criterion "PLEDGE AMOUNT >= {PAYMENT AMOUNT (GIVER'S)}" to select transactions for which the amount pledged is greater than or equal to the amount paid.

## Viewing the List of Valid Literal Values and Fields

You can view either the list of valid literal values for some queryable fields or view the valid field names for the query type.

# Team Approach 5.0.1

## Understanding Queries

---

- To see the list of valid literal values for the queryable field you selected, press the [F9] key from the Values field.
- To see the list of all valid field names for the query, press the [Alt] [F9] key combination from the Values field.

If no list exists for the queryable field, TA displays the message "List of values not available for Query Field name specified." in the status bar.

In addition, if one or more values lists exist for the queryable field, the values lists appear at the top of the list of literal values. See "[Values Lists](#)".

### Entering a Date

When you use a mathematical operator, you can enter a date as a literal value, e.g., 05/01/99, or you can use one of two of the following keywords, if appropriate.

<b>Effective Date</b>	Use this phrase when the field criterion is evaluated against the date specified in the Effective Date field on page 2 of the Query Entry screen. For example, "EXPIRATION DATE = EFFECTIVE DATE" refers to those records whose expiration date is the date in the Effective Date field for this query.
<b>Today</b>	Use this key word when the field criterion is evaluated against the date the query is run. For example, "EXPIRATION DATE = TODAY" refers to those records whose expiration date is today's date. You should use this keyword carefully since you can create a query on one day but not schedule the query to run until a later date.

If you enter a date as a literal value, enter it in MM/DD/YY format where "MM" is the month, "DD" is the day, and "YY" is the year. If you enter a keyword, you can type the keyword into the Values field or press the [F9] key to view the available keywords.

See "[Setting a Query Effective Date](#)".

### Special Treatment of Nulls for a Selection Criterion

A field that is "null" means that the field is blank in the database. Since a field with a null value can have a different meaning depending on the context, you can determine if TA should include the record if it meets all of the other query criteria and the field is null. For example, you can query to find records that have a particular date range for Classifications. If an End Date is blank, it could mean that it is a current classification (no end date is assigned). To make sure that TA includes the record, you need to establish how TA handles the record if a blank End Date is encountered. In this example, a null end date should be translated to a date far in the future such as the year 9999.

You can set the Translate Null Values To field by pressing the [F10] key from the Values field to access the Field Values Options window. You can then enter into the Translate Null Values To field the value you want TA to use for null values. The default value for this field is "/s" which is used to indicate a single space. If you don't want nulls to be treated in a special manner, you can delete the value from the Translate Null Values To field. Click <Close> to close the Field Values Options window and return to the Query Entry screen.

Another Null Values example: the Hedge? field on a transaction can be blank, set to "Y" for yes, or set to "N" for no. Usually, this field is blank or set to yes. If the field is blank or set to no, the transaction is not a hedge. If the field is set to yes, the transaction is a hedge. If you want to select a set of transactions that aren't hedges and you enter "HEDGE <> Y" as the criterion, the query selects all transactions where some value is in the Hedge field but that value isn't "Y." The query would select all transactions where the Hedge field is set to no but would not select transactions where the hedge field is blank. To select both transactions where the Hedge field is set to no or is blank (since no

and blank have the same meaning), you would enter "HEDGE = N" and set the Translate Null Values To field to "N" meaning translate a null value to the value "N." By setting the Translate Null Values To field to "N," you indicate that when TA evaluates this field for a record, if the field is blank, treat the blank as if it were an "N" and use the translated value for the comparison.

**Note:** As an alternative to setting the Translate Null Values To field, you can enter two criteria grouped together to achieve the same results: "(HEDGE = N OR HEDGE IS NULL)." See "[Using Parentheses to Group Criteria for Processing](#)".

## Using the Comparison Operators BETWEEN and NOT BETWEEN

When you select from the BETWEEN and NOT BETWEEN operators, TA evaluates the record's value for the queryable field you selected in the Query Field Name field to see if the value is between or is not between the two values listed in the Values field. Use a comma to separate the values you enter in the Values field.

Use these operators to select records based on a range of values for the field you selected in the Query Field Name field. TA evaluates each record pulling those where the value of the field for the record is or isn't within the range; the starting and ending values are included in the range. For example, if you want to select transactions between 7/1/99 to 7/7/99, you would enter the criterion "GIFT DATE BETWEEN 7/1/99,7/7/99."

**Note:** If you enter a date, enter it in MM/DD/YY format where "MM" is the month, "DD" is the day, and "YY" is the year.

## Using the LIST Operators

When you select from the IN LIST and NOT IN LIST operators, TA evaluates the record's value for the queryable field you selected in the Query Field Name field to see if the value is in or is not in the list of values in the Values field. Use a comma to separate the values you enter in the Values field.

You can use the [F9] key to select the values for the list. If you press the [F9] key multiple times to make your selections, TA inserts the comma between the last value you selected and the newest value you select.

Use these operators to select records based on more than one value for the field you selected in the Query Field Name field. TA evaluates each record pulling those where the value of the field for the record is or isn't one of the values in the list of values you defined. For example, if you want to select stock and in-kind gift transactions, you would enter the criterion "PAYMENT METHOD (TRANSACTIONS) IN LIST IK, ST" where "IK" is an in-kind gift and "ST" is a stock gift.

**Note:** If you enter dates in the list, enter each one in MM/DD/YY format where "MM" is the month, "DD" is the day, and "YY" is the year.

Use the LIST CONTAINS operator **only** if your query is referencing **Service Provider Attributes**. For example, if you have the service provider attributes of 'Language'. A service provider might have employees who can handle requests in SPANISH, FRENCH, KOREAN. If you're going to run a query to look for all service providers that have someone available to speak KOREAN, your query line would look like this::

Language            LIST CONTAINS            KOREAN

### See Also:

[Values Lists](#)

## Values Lists

In addition to entering multiple values into the Values field when using the IN LIST or NOT IN LIST operators, the field you selected may have one or more values lists associated with it. A "values list" refers to a stored list of values that can be referenced using the name of the values list. You can set the Values field to the name of the values list instead of having to enter each value into the Values field. A values list begins with the "at" sign ("") followed by the name of the list, i.e., LIST NAME.

# Team Approach 5.0.1

## Understanding Queries

---

For example, your organization may set up a values list named "NESTATES" for the New England States. This values list includes the values "CT" for Connecticut, "MA" for Massachusetts, "ME" for Maine, "NH" for New Hampshire, "RI" for Rhode Island, and "VT" for Vermont. You could then enter the criterion "STATE (GIVER'S) IN LIST NESTATES" to select transactions of donors in New England.

When you press the [F9] key to see the list of valid literal values for the field you selected in the Query Field Name field, if any values lists exist for that field, the values lists are listed at the top, before the literal values. See "[Entering a Literal Value](#)".

### Viewing the Values Included in a Values List

If the name of a values list is entered in the Values field, you can view the values included in that list by pressing the [F10] key from the Values field to access the Field Values Options window. On the Field Values Options window, look at the Current Field Value field to see the values for the list you selected. This field is not updateable. You can click <Close> to close the Field Values Options window and return to the Query Entry screen.

### See Also:

[Creating a Values List](#)

[Updating a Values List](#)

[Viewing All Values Lists for a Queryable Field](#)

### Creating a Values List

If you want re-use a particular list of values consider creating a customized list. You can create a customized list of values for any queryable field. You can only use customized value lists with the IN LIST or NOT IN LIST operator. Your list can include the wildcard character (%) to handle ranges.

For example, you can create a list of specific ZIP codes. Instead of adding the ZIP codes as separate lines in your query, use your value list. Your query criteria line would be for example: ZIP code IN LIST NEZIPS (if your value list is of New England Zip codes).

Before you create a new values list for a queryable field, you may want to view the existing values lists for that field. See "[Viewing All Values Lists for a Queryable Field](#)".

1. Enter the field you are including in your query into the Query Field Name field.
2. Enter the operator IN LIST or NOT IN LIST, into the Operator field.
3. From the Values field, press the [F10] key to access the Field Values Options window.
4. From the Field Values Options window, click <Create or Change Values List>.
5. Click <New> on the Find Values List window to access the Create New Values List window.
6. Enter the unique name of the values list into the List Name field. You can enter up to 30 characters including spaces. If you enter a name already used for an existing values list, TA displays a standard message from which you can view the existing list on the Select Values List window or return to the Create New Values List window to change the value list name.
7. Enter the values for the list in the Values field. Use a comma to separate the values you enter in the Values field. Note that you can use the % if you want a range of values.
8. Click <Create> to create the list.

TA closes the Create New Values List window and returns to the Query Entry screen. The list you created is entered in the Values field with an "at" sign ("") in front of the name of the list, i.e., NEZIPS.

## Updating a Values List

To update a values list, complete the following steps.

1. Enter the queryable field into the Query Field Name field.
2. Enter the appropriate operator, IN LIST or NOT IN LIST, into the Operator field.
3. From the Values field, press the [F10] key to access the Field Values Options window.
4. On the Field Values Options window, click <Create or Change Values List> to access the Find Values List window.
5. On the Find Values List window, press the [F9] key from the List Name field to view the list of existing value lists for the field on which you want to query.
6. Select the values list you want to update.
7. Click <Find> from the Find Values List window to display the Select Values List window.
8. In the Values field to the right of the values list's name, update the values. You can press the [F9] key to view the list of literal values for the queryable field.
9. Click <OK> to close the Select Values Lists window and return to the Field Values Options window.
10. Click <Close> from the Field Values Options window to return to the Query Entry screen.

## Viewing All Values Lists for a Queryable Field

To view all values lists for a queryable field, complete the following steps.

1. Enter the queryable field into the Query Field Name field.
2. Enter the appropriate operator, IN LIST or NOT IN LIST, into the Operator field.
3. From the Values field, press the [F10] key to access the Field Values Options window.
4. On the Field Values Options window, click <Create or Change Values List> to access the Find Values List window.
5. On the Find Values List window, leave the List Name field blank.
6. Click <Find> from the Find Values List window to display the Select Values List window.
7. The Select Values List window displays all values lists for the queryable field.
8. Click <Close> from the Field Values Options window to return to the Query Entry screen.

For this queryable field, you can complete one of three tasks from the Select Values List window.

- Update an existing values list. See "[Updating a Values List](#)."
- Select an existing values list by clicking the <Zoom> button to the left of the list.
- Create a new values list by clicking <New>. See "[Creating a Values List](#)."

## Using the Comparison Operators IS NULL and IS NOT NULL

When you select from the IS NULL and IS NOT NULL operators, TA evaluates the record's value for the queryable field you selected in the Query Field Name field to see if the value is null or is not null. You do not list any values in the Values field when using either of these operators. "Null" means that the field does not contain a value, or, in other words, that the field is blank in the database. It doesn't mean the value is 0.

# Team Approach 5.0.1

## Understanding Queries

---

- Use the "IS NULL" operator to select records where the field does not contain a value, e.g., "DATE ACKNOWLEDGED IS NULL" to select those transactions that have yet to be acknowledged.
- Use the "IS NOT NULL" operator to select records where the field contains any value, e.g., "DATE ACKNOWLEDGED IS NOT NULL" to select those transactions that have been acknowledged but you don't care on what date the acknowledgment occurred.

### Using the Comparison Operators LIKE and NOT LIKE

When you select from the LIKE and NOT LIKE operators, TA evaluates the record's value for the queryable field you selected in the Query Field Name field to see if the value does or does not partially match the value listed in the Values field. When you select from these two operators, you need to enter the characters that TA should use plus the "wildcard character." In TA, the percent sign ("%") is the wildcard character and represents all characters that follow those you specified. The percent sign represents a wildcard only with the LIKE and NOT LIKE operators. TA interprets the percent sign literally elsewhere, i.e., meaning as an actual percent sign.

For example, you can set up the criterion "ALPHA NAME LIKE New%" for an accounts query to select all accounts with an alpha name that starts with "New" (e.g., Newman, Newton).

**Note:** You can also use the underscore symbol ("\_") to represent any single character in the position where the underscore appears. For example, assuming your organization uses a one-character activity type and the activity "AD" represents major donors, the criterion "SOURCE (TRANSACTIONS) LIKE AD\_990701001" selects transaction records with a source that represents the first segment of the first effort of all major donor campaigns during July of 1999 regardless of the campaign, i.e., all source codes with the first two characters of "AD" and the last nine characters of "990701001." The second character can be any character. The underscore represents a wildcard only with the LIKE and NOT LIKE operators. TA interprets the underscore literally elsewhere, i.e., meaning as an actual underscore.

### Indicating the Use of Mixed Case Value

When you enter a literal value into the Values field, TA automatically uppercases this value after you leave the field. TA capitalizes all of the characters you enter because many of the values you enter are code values, and all code values are stored in uppercase in the database. For some situations, you want to use all lowercase or mixed case in the Values field because the value in the database is stored in lowercase or mixed case. For example, comments, first names, middle names, and last names are stored in mixed case in the database.

You can indicate that TA should not uppercase the value in the Values field by completing the following steps *before* you enter the values.

1. From the Values field, press the [F10] key to access the Field Values Options window.
2. Mark the Allow Mixed Case Field Values checkbox.
3. Click <Close> to close the Field Values Options window and return to the Query Entry screen
4. Enter the value in mixed case into the Values field.

### Using a Parameterized Value

Parameterized values enable you to create a query with parameters (i.e., placeholders) in place of fixed values in the selection criteria. The field-values parameter acts as a placeholder for values that you enter later in the process (often during query and output scheduling). As you can wait until the scheduling stage to enter the values, you can use one parameterized query to select multiple sets of records. All field-values parameters begin with a colon (":").

**Note:** You can use a field-values parameter wherever you would use a fixed value in the Values field. A field-values parameter cannot be used to stand for a field name or a values list.

To use a parameterized value in the selection criterion, complete the following steps:

1. Enter a colon followed by any character string of up to 30 letters or numbers (e.g., :GIFT TYPE) in the Values field.
2. Use the [Tab] key to move out of the field. TA displays the Field Values Parameters window, with the parameterized value you entered in the Parameter field.
3. Enter the default value for the parameter (e.g., "NW" for new) in the Default Value field if you want TA to default this value when the query is scheduled (i.e., this value defaults when the query is scheduled, but the user can change the value). If there is no default value for the parameter, leave the Default Value field blank.
4. Enter a prompt in the Prompt field that will notify the user what value to enter when they schedule the query (e.g., "Enter the gift type").

**Note:** TA defaults the prompt to the name of the field-values parameter as you typed it in, but you will probably want to enter a more informative prompt.

5. If the parameterized value should be in mixed case, enter "Y" for yes in the Mix Case field. See ["Indicating the Use of Mixed Case Value"](#) which precedes this section for more information about mixed case. If the parameterized value should not be mixed case, leave the Mix Case field set to "N" for no.

### Example

For example, you might want to send different solicitations to the accounts whose last gift was made during one of the following calendar years: 1999, 1998, or 1997. You can create one parameterized query that is set up to select all of the accounts whose last gift was made during a date range, with prompts indicating that the date range constitutes a calendar year. The same query can be scheduled three times, each with the appropriate outputs for each solicitation.

For the above example, you might enter the following information on the Selection Criteria block.

Query Field Name	Operator	Values	And/Or
LAST GIFT DATE (ACCOUNT ACTIVITIES)	BETWEEN	:BEGINDATE, :ENDDATE	

When you use the [Tab] key to move out of the field, TA displays the Field Values Parameters window. To indicate that the date period is for a calendar year, you might enter the following information in the Field Values Parameters window.

Parameter	Default Value	Prompt
BEGINDATE	01/01/99	Enter the first date of a calendar year
ENDDATE	12/31/99	Enter the last date of a calendar year

### See Also:

[Updating the Default Value or Prompt for a Field Values Parameter](#)

### Updating the Default Value or Prompt for a Field Values Parameter

You can view and update both the default value and the prompt for a field values parameter by completing the following steps.

1. Move your cursor to the Values column in the Selection Criteria block.
2. Press the [F10] key to access the Field Values Options window.

# Team Approach 5.0.1

## Understanding Queries

---

3. Click <Parameters>. TA displays the Field Values Parameters window, where you can update the default value and the prompt for each parameter.
4. Click <Close> to return to the Query Entry screen.

### Connecting Multiple Selection Criteria

When you enter more than one selection criterion, you must indicate how TA should process the criteria. You can indicate how TA should process criteria by:

- Using the appropriate connector between two criteria lines.
- Grouping criteria lines by placing parentheses in the Left Parentheses field and the Right Parentheses field in the Selection Criteria block.

You *must* connect one criterion to the next one regardless of how many criteria you enter, but you do not have to use parentheses.

### See Also:

[Selecting a Connector](#)

[Using Parentheses to Group Criteria for Processing](#)

### Selecting a Connector

You must connect one criterion to the next using one of the following three connectors.

- And
- Or
- +Not

You enter the connector in the And/Or field. You can press the [F9] key from this field to view the list of connectors.

### See Also:

[Using And](#)

[Using Or](#)

[Using +Not](#)

### Using And

When you use "And" as the connector, you indicate that the record being evaluated must meet both the criterion on the line before the connector and the one on the line after the connector. For example, to find each account that has a current giving level amount over \$1,000 and is also interested in education, you would enter the following criteria using the connector And.

<u>Query Field Name</u>	<u>Operator</u>	<u>Values</u>	<u>And/Or</u>
CURRENT GIVING LEVEL AMOUNT	>	1000	AND
INTEREST CODE	=	EDUCATN	

If a record does not meet both criteria, the record is not selected by the query.

## Using Or

When you use "Or" as the connector, you indicate that the record being evaluated must meet either the criterion on the line before the connector or the one on the line after the connector. For example, to find each account that has a current giving level amount over \$1,000 or is interested in education, you would enter the following criteria using the connector Or.

Query Field Name	Operator	Values	And/Or
CURRENT GIVING LEVEL AMOUNT	>	1000	OR
INTEREST CODE	=	EDUCATN	

If a record meets either criteria, the record is selected by the query.

## Using +Not

When you use "+Not" as the connector, you indicate that the record being evaluated must meet the criterion on the line before the connector but not the one on the line after the connector. For example, to find each account that has a current giving level amount over \$1,000 and that has not expressed an interest in education, you would enter the following criteria using the connector +Not.

Query Field Name	Operator	Values	And/Or
CURRENT GIVING LEVEL AMOUNT	>	1000	+Not
INTEREST CODE	=	EDUCATN	

If a record meets the first criterion and doesn't have the information noted in the second criterion, the record is selected by the query. You should use the +Not connector when you want to select a record based on the absence of the criterion you enter.

Using the +Not operation *is not* the equivalent of writing the criteria in the following manner.

Query Field Name	Operator	Values	And/Or
CURRENT GIVING LEVEL AMOUNT	>	1000	AND
INTEREST CODE	<>	EDUCATN	

This set of criteria will select an account with an education interest if more than one interest exists for the account. In other words, when evaluating the above set of criteria,

- If the account's giving level is over \$1,000 *and*
- the account has two or more interests *and*
- one of those interests is education

then the account is still selected. Assume the account has only two interests, one for education and one for opera. When TA evaluates the account, the set of criteria is not true when TA looks at the education interest. But, when TA evaluates the criteria by looking at the opera interest, the set of criteria is true, which results in the account being selected.

## Using Parentheses to Group Criteria for Processing

You can group criteria together using parentheses. The parentheses indicate that TA should process all of the criteria inside the parentheses before processing any criteria outside of the parentheses.

## Team Approach 5.0.1

### Understanding Queries

For example, you could set up an accounts query with the following criteria using parentheses.

Query Field Name	Operator	Values	And/Or
( CURRENT GIVING LEVEL AMOUNT	>	1000	AND
INTEREST CODE	=	EDUCATN	) OR
( CURRENT MEMBER YEAR ADDITIONAL AMOUNT	>	500	OR
DEGREE MAJOR	=	EDUC	)

In this example, TA processes the selection criteria as follows.

1. TA processes the criteria in the first set of parentheses, evaluating each account to see if the account's current giving level amount is over \$1,000 and if the account has expressed an interest in education. The record must meet both criteria.
2. TA processes the criteria in the second set of parentheses, evaluating each account to see if the account's current member year additional amount is over \$500 or the account has an education degree. The record must meet one of the two criteria.
3. TA then evaluates the account to see if it meets #1 or meets #2. If the account meets either #1 or #2, the account is selected.

### See Also:

[Nesting Parentheses](#)

### Nesting Parentheses

You can "nest" parentheses up to five times meaning that the Left Parentheses field and the Right Parentheses field can accept up to five parentheses each. When you set up parentheses, review the nested sets to ensure that you have enclosed in parentheses the correct set of criteria. The following example includes one set of nested parentheses.

Query Field Name	Operator	Values	And/Or
( CURRENT GIVING LEVEL AMOUNT	>	1000	OR
( INTEREST CODE	=	EDUCATN	AND
DEGREE MAJOR	=	EDUC	) OR
CURRENT GIVING LEVEL AMOUNT	>	5000	)

In this example, TA processes the selection criteria as follows.

1. TA processes the criteria in the "innermost" set of parentheses (the nested set of parentheses), evaluating each account to see if the account has expressed an interest in education and the account has an education degree. The account must meet both criteria.
2. TA processes the criteria in the "outer" set of parentheses, evaluating each account to see if the account's current giving level amount is over \$1,000 or meets the criteria in the nested parentheses. The account must meet one of the two criteria.
3. TA then evaluates the account to see if it meets #2 or has a current giving level amount greater than \$5,000. If the account meets either #2 or the account's current giving level amount is over \$5,000, the record is selected.

## Adding to and Deleting from a Set of Selection Criteria

When you create a query, you can enter one or more selection criteria. You can modify the query by adding to or deleting from a set of selection criteria. When you add a selection criterion to the set, you can add the new criterion to the end of the set of criteria or within the set of criteria. You use the [Next Record] key to add the criterion in the first situation and the [Ctrl] [T] key combination to add the criterion in the second situation.

### See Also:

[Adding a Selection Criterion at the End of the Set of Criteria](#)

[Adding a Selection Criterion Within the Set of Criteria](#)

[Deleting a Selection Criterion](#)

## Adding a Selection Criterion at the End of the Set of Criteria

If you entered a selection criterion and no selection criterion follows it, you can press the [Next Record] key to go to the next blank line to enter the next selection criterion. For example, assume you have set up the following criteria for a query.

Query Field Name	Operator	Values	And/Or
CURRENT GIVING LEVEL AMOUNT	>	1000	OR
INTEREST CODE	=	EDUCATN	

To add the next criterion, enter the appropriate connector in the And/Or field (e.g., AND), then press the [Next Record] key to move to the blank line below the second criterion. The cursor moves from the And/Or field set to "AND" for the INTEREST CODE = EDUCATN criterion to the Query Field Name field on the blank line below. The result is three criteria lines, the last of which is blank.

Query Field Name	Operator	Values	And/Or
CURRENT GIVING LEVEL AMOUNT	>	1000	OR
INTEREST CODE	=	EDUCATN	AND
(blank line)			

## Adding a Selection Criterion Within the Set of Criteria

If you entered a set of selection criteria and you want to add a criterion within the set, you can press the [Ctrl] [T] key combination to insert a blank criterion line below the line in which the cursor sits. (Make sure your caps lock key is not on.)

For example, assume you have set up the following criteria for a query.

Query Field Name	Operator	Values	And/Or
CURRENT GIVING LEVEL AMOUNT	>	1000	OR
INTEREST CODE	=	EDUCATN	

To add a criterion between the first criterion and the second criterion, highlight the first criterion and press the [Ctrl] [T] key combination. TA creates a blank line between the current giving level amount criterion and the interest code criterion.

Query Field Name	Operator	Values	And/Or
CURRENT GIVING LEVEL AMOUNT	>	1000	OR
INTEREST CODE	=	EDUCATN	

# Team Approach 5.0.1

## Understanding Queries

Query Field Name	Operator	Values	And/Or
CURRENT GIVING LEVEL AMOUNT (blank line)	>	1000	OR
INTEREST CODE	=	EDUCATN	

### Deleting a Selection Criterion

If you entered a set of selection criteria and you want to delete a criterion, you can highlight the criterion you want to delete and press the [Shift] [F6] key combination. TA removes the criterion from the set of criteria. You can delete as many criteria as you need.

### Setting Processing Options on the Query Entry (Detail) Screen

You can set or update the additional processing options for a query on the Query Entry (Detail) screen, which can be accessed by clicking <Page 2> on the Query Entry screen.

### See Also:

- [Setting a Query Effective Date](#)
- [Setting a Preferred Address Type](#)
- [Linking the Query to a Source Code](#)
- [Calculating Ask Amounts](#)
- [Returning Special Action Information](#)
- [Marking the Query as Public or Private](#)
- [Marking the Query as an Include or Exclude Query](#)
- [Allowing Other Users to Edit Retrieved Records](#)
- [Setting the SQL Edited Field](#)
- [Setting an Expiration Date for the Retrieved Records](#)
- [Enabling the Query to Search for Inactive Records](#)

### Setting a Query Effective Date

When you create a query, the Effective Date field is blank. If the field is blank, TA uses the date the query runs as the effective date. If you enter a date into this field, enter the date in MM/DD/YY format where "MM" is the month, "DD" is the day, and "YY" is the year.

**Note:** If you don't set the Effective Date field when you create the query, you can set it on the Query and Output Schedule screen when you schedule the query to run. See the [Scheduling Queries and Viewing Records Retrieved chapter](#) to learn how to schedule a query to run.

TA uses the effective date of the query for two types of query processing.

- Selecting the correct address for a query assigned the Accounts, Names, Benefits, or Transactions query type. For example, TA looks for a valid seasonal address to use when selecting the correct address. TA looks for this address based on the effective date of the query.  
**Note:** If you run a query to select donors for a mailing, each donor's address is selected based on the effective date. If the mailing is not actually mailed for some number of days, the mailing may go to an address that is no longer the valid seasonal address.
- Calculating the value for a selection criterion based on a date. If the value in the Values field of the selection criterion is EFFECTIVE DATE, TA uses the value in the Effective Date field when the query runs. The effective date is also used by some of the calculated fields such as Months to Expire

(Transaction Benefits), Months to Expire (Account Activity) and Payment Date. See "[Calculated Fields](#)"

## Setting a Preferred Address Type

For all query types except addresses and sources, if you want TA to try to select an address based on a specific address type, you should enter that type in the Address Type field. You can view the list of address types by pressing the [F9] key.

For example, you can enter "P" for premium to have TA select the premium address if a valid premium address exists for the account, and if it is not excluded because of seasonal dates. If a premium address does not exist, or if it is not valid for the time period, TA uses the address selection rules to select a different address.

### See Also:

[Reference Information About the Address Type Field](#)

## Reference Information About the Address Type Field

For each address entered into TA, you can set:

- The address type, e.g., "H" for home.
- One or more specific uses of the address, e.g., "P" for use for mailing premiums.

For all query types except addresses and sources, TA's query facility is programmed to select one address for each account selected by the query. TA follows a set of rules to select the address appropriate to the effective date of the query. If you want TA to try to select an address based on a specific address type, you should enter that type in the Address Type field. For each account selected, TA applies the address selection rules. If the account has a valid address of the type you entered that isn't excluded because of seasonal dates, TA selects that address for the account. *If TA cannot find a valid address to select and one or more address fields are used in the user-defined output, the record will be selected by the query but not included in the output file.*

For example, your organization sets up a specific use code to identify the address to which your membership publication should be mailed. For some accounts, this code is entered into one of the Specific Use? fields for the address specified for the publication mailing; however, this information is not specified for every account. You create a query to select all accounts to which this month's membership publication should be mailed. You set the Address Type field on the Query Entry screen to the specific use code for the membership publication. For each account selected by the query, TA checks each address to see if one address has a Specific Use? field set to the membership publication specific use code. If the code is entered for an address, TA selects that address if it meets the expected arrival date of the query. Otherwise, TA follows the address selection rules to select the correct mailing address.

See "[Address Selection Rules](#)".

## Linking the Query to a Source Code

You can enter an active source code into the Source field or change the one set by TA. You can view the list of source codes by entering at least the first letter of the source code and pressing the [F9] key. The more letters from the beginning of the source code that you enter, the fewer are the source codes that appear in the list.

**Note:** If you don't set the Source field when you create the query, you can set it on the Query and Output Schedule screen when you schedule the query to run. See the [Scheduling Queries and Viewing Records Retrieved chapter](#) to learn how to schedule a query to run.

### See Also:

[Reference Information About the Source Field](#)

# Team Approach 5.0.1

## Understanding Queries

---

### Reference Information About the Source Field

The Source field contains the source code that ties the query to a particular solicitation. When you create a query to select the donors your organization will be soliciting for a gift, and your organization has set up the source codes to use, you can set the Source field to the source code to be used for the group of donors selected by the query. You should have one query per source code.

If, when your organization set up the source codes, your organization used one of TA's automatic query creation features, then TA:

- Created the query, naming it SOURCECODE where "SOURCECODE" is the actual source code, e.g., AMQ990801001.
- Set the Source field on the query to that source code. For example, TA would set the Source field to AMQ990801001 for the query AMQ990801001.

See "[Source Code Queries](#)".

**Note:** If your organization sets the [Prefix for Source Code](#) system preference, TA creates the source code query using the value of this system preference as a prefix for the query name. TA sets the query name to PFX SOURCECODE where "PFX" is the prefix and "SOURCECODE" is the source code for which the query was created.

### Calculating Ask Amounts

If you want TA to calculate the ask amounts with a query, enter the ask amount rules code in the Ask Amount Rules field. You can press the [F9] key to select from a list of ask amount rules. If you don't set this field, TA does not calculate the ask amounts.

**Note:** If you don't set this field when you create the query, you can set it on the Query and Output Schedule screen when you schedule the query to run. See the [Scheduling Queries and Viewing Records Retrieved chapter](#) to learn how to schedule a query to run.

### See Also:

[Reference Information About the Ask Amount Rules Field](#)

### Reference Information About the Ask Amount Rules Field

The Ask Amount Rules field identifies the ask amount rule to use with a solicitation. When you create a query to select the donors your organization will be soliciting for a gift, you can specify the rule TA should use to calculate the ask amounts to use for each donor. Using TA's output facility, the amounts can be printed on the reply device if your organization is using direct mail, or the amounts can be included in the file for a telemarketing campaign. If the Ask Amount Rules field is blank, TA does not calculate the ask amounts.

Your organization can set up multiple ask amount rules. Each rule specifies the formula that TA uses to calculate up to five ask amounts. Your organization can change the formula based on the donor's giving level, i.e., within one rule, you can set up one ask amount formula per giving level. Each ask amount formula can be based on a percentage of a gift amount your organization specifies or can be a specific dollar amount.

For example, your organization can set up the following rule. The ask amount is based on each donor's current member year amount. To calculate the five ask amounts for donor's from \$.01 to \$249.99, TA uses the following percentages: 100%, 105%, 110%, 115% and "Other." To calculate the five ask amounts for donors from \$250 to \$499.99, TA uses the following percentages: 90%, 100%, 110%, 120% and "Other." To calculate the five ask amounts for donors over \$500, TA uses the following percentages: 75%, 90%, 100%, 110% and "Other."

See the Setting Up Control Codes chapter to learn how to create an ask amount rule and how to view existing ask amount rules.

## Returning Special Action Information

*This option is valid only for queries assigned the Transactions or Benefits query type.*

When you create a query, TA defaults the Special Action? field to "N" for no. If you want the special action information (i.e., the special action code and the special action comment) returned with the records selected by the query, you can set the Special Action? field to "Y" for yes.

### See Also:

[Reference Information About the Special Action Field](#)

## Reference Information About the Special Action Field

The Special Action? field indicates whether the special action information should be returned with the records selected by the query. The special action information is composed of the special action code and the special action comment. You can enter this information for a gift or pledge using the Acknowledgment window from the Batch Detail Entry screen at the time of entry or using the Acknowledgment Information window from the Giving History Detail screen after the pledge or gift has been processed. Your organization sets up the special action codes. These codes can indicate, for example, that an acknowledgment should be hand-written by the solicitor.

## Marking the Query as Public or Private

The Private? field indicates if the query is private or public. A private query is accessible only to the [owner of the query](#). A public query is accessible to all users for viewing and copying.

When you create a query, TA defaults the Private? field to "N" for no, indicating that the query is public. If you want the query to be private, set the Private? field to "Y" for yes.

See "[Query Privileges](#)."

## Marking the Query as an Include or Exclude Query

When you create a query, you can indicate whether the query should be available as an include query, an exclude query, as both, or as none by setting the Include/Exclude? field. When you create a query, the Include/Exclude? field is blank. You can:

- Leave the field blank to indicate the query will not appear in the list of queries for the Include block and the query will not appear in the list of queries for the Exclude block.
- Set the field to "INCLUDE" to indicate the query will appear only in the list of queries for the Include block.
- Set the field to "EXCLUDE" to indicate the query will appear only in the list of queries for the Exclude block.
- Set the field to "BOTH" to indicate the query will appear in the list of queries for the Include block and in the list of queries for the Exclude block.

You can press the [F9] key from the Include/Exclude? field to see the list of valid values.

See "[Include and Exclude Queries](#)". See "[Working with Include and Exclude Queries](#)".

# Team Approach 5.0.1

## Understanding Queries

---

### Allowing Other Users to Edit Retrieved Records

When you run a query, you can indicate that the records selected should be saved. You can then edit the list, adding or removing records from it, and use the list for as many outputs as you need. How the Edit Retrieved? field is set determines whether or not other people can edit the records.

When you create a query, TA defaults the Edit Retrieved? field to "N" for no, which indicates that other users cannot edit the saved records from the query. To permit other users to edit the records saved by the query, set the Edit Retrieved? field to "Y" for yes.

See the [Scheduling Queries and Viewing Records Retrieved chapter](#) for an explanation of scheduling options for queries, including viewing and updating the set of records selected by a query.

**Note:** You can test run a query to select a sample of the records. See "[Testing a Query to See If It Is Selecting the Correct Records](#)".

### Setting the SQL Edited Field

When you test a query you create, TA takes the information you entered on the Query Entry screen and generates the SQL statement to use for selecting the records. However, if you edit the SQL statement (which you might do on rare occasions when working with a TA consultant or technical support staff member), TA sets the SQL Edited? field to "Y" for yes. When this field is set to yes, each time you test the query, TA only checks that the statement is valid; TA does not change the SQL statement. If you want TA to regenerate the SQL statement, you must change the field to "N" for no.

See "[SQL Statement](#)". See "[Testing a Query to Generate the Query's SQL Statement](#)".

### Setting an Expiration Date for the Retrieved Records

When you schedule a query, you can indicate that the records selected by the query should be saved. These records will be saved until the date entered in the Expiration Date field. When TA's end of day processing runs, TA deletes those saved records with an expiration date equal to the current date.

When you create a query, the Expiration Date field is blank. If you leave the Expiration Date field blank, TA sets the field when the query is run based on the [Days Until Queried Records Expire](#) system preference. If you want TA to save the records for a longer period, set the Expiration Date field to the date the records should be deleted by entering that date in MM/DD/YY format where "MM" is the month, "DD" is the day, and "YY" is the year.

For example, assume the system preference is set to 30. You create a query and leave the Expiration Date field blank. You schedule the query to run on 1/16/99. When the query runs, TA sets the expiration date for the query to 2/15/99. TA deletes the saved records the first time end of day processing is run after the expiration date.

See the [Scheduling Queries and Viewing Records Retrieved chapter](#) for an explanation of scheduling options for queries, including viewing and updating the set of records selected by a query.

### Enabling the Query to Search for Inactive Records

For either a Transactions or a Benefits query, you can choose to include inactive records in your search. To include inactive records, set the Query Universe field as follows:

- For a Transactions query, set the Query Universe field to "TRANSACTIONS."
- For a Benefits query, set the Query Universe field to "TRANSACTION\_BENEFITS."

You can press the [F9] key from the Query Universe field to see the list of valid values.

## Viewing and Editing the SQL Statement

You can view the SQL statement created by the query by clicking <View SQL>, which is located on the second page of the Query Entry screen. TA displays the Query SQL Text window. Since you can create complex queries, you can use the <Part 1>, <Part 2>, and <Part 3> buttons to view different parts of the statement. You can close this window by clicking <Close>.

If you edit the statement, TA sets the SQL Edited field to "Y" for yes; this field is located on the second page of the Query Entry screen. When you close the Query SQL Text window, TA saves the changes you made but does not update the information on the first page of the Query Entry screen to reflect these changes. See "[Setting the SQL Edited Field](#)".

## Testing a Query

Once you have set up a query you must test it so that TA can check that the query has been set up correctly and generate the SQL statement that actually selects the records. You can also retrieve a sample of records when you test a query to check that the query is selecting the correct records.

### See Also:

[Testing a Query to Generate the Query's SQL Statement](#)

[Testing a Query to See If It Is Selecting the Correct Records](#)

## Testing a Query to Generate the Query's SQL Statement

After you have set up the query, you must test the query to see that it is composed correctly and to generate the SQL statement. You can test the query by completing the following steps.

1. Click <Test Query> on the Query Entry screen to access the Test/Run Query window.
2. Leave the Maximum Number to Retrieve field blank.
3. Click <Run>.
4. TA displays a message prompting you to select whether or not to remove any previously existing records for the query. TA generates the SQL statement regardless of whether you click <Yes> or <No>; however, if you click <No>, TA leaves any records that were previously retrieved by the query in the Query Retrieved table.

When TA finishes composing this statement, TA displays the appropriate message to indicate whether the query generated a valid SQL statement. If the query generated a valid SQL statement, TA activates the query by setting the status of the query to "A" for active. The query can now be scheduled.

See the [Scheduling Queries and Viewing Records Retrieved](#) chapter for an explanation of scheduling options for queries, including viewing and updating the set of records selected by a query.

## Determining the Cause if the Test Ends In Error

If TA cannot generate a valid SQL statement, TA sets the query's status to "E" for errors. If a query's status is set to "E" for errors, complete the following steps to determine the cause of the error.

1. Check the SQL Edited field. If the field is set to "Y" for yes, then check the SQL statement to see if it is valid. See "[Viewing and Editing the SQL Statement](#)".
2. Find out from your Database Administrator or Systems Administrator if any problems exist with the database or network. Make sure to explain the query that is in error. If TA displayed a specific error message, include that message and the steps you took until the message appeared.
3. Contact TA Support. Make sure to include the name of the query. If TA displayed a specific error message, include that message and the steps you took until the message appeared.

# Team Approach 5.0.1

## Understanding Queries

---

### Testing a Query to See If It Is Selecting the Correct Records

When you create a query, you are asking a question. TA includes many fields on which you can query and enables you to set up complex queries. You can test the query to ensure that you have set up the question properly by looking at a sample of the records the query selects. If the records the query selects aren't the correct ones, then you need to modify the query. Since the query generated a valid SQL statement, you have set up a "legitimate" question but not the one you intended to ask.

You can view a sample of the records the query selects by completing the following steps.

1. Click <Test Query> on the Query Entry screen to access the Test/Run Query window.
2. Set the Maximum Number to Retrieve field to the appropriate sample size. You can enter a number up to 9999999. Target Software, Inc. recommends you enter a relatively small number of records to retrieve.
3. Click <Run>.

TA tests the query and runs it, selecting records that meet the query criteria. The length of time TA needs to test and run the query depends upon the nature of the query and the number of records requested. While TA is testing and running the query, you cannot work in any other part of TA; your screen is locked. When TA is finished, TA displays a message showing the number of records retrieved.

### Viewing the Records Selected by the Test Run

Once you test and run a query successfully, you can view the set of records selected by the query by clicking <View Records> on the first page of the Query Entry screen. TA displays the Query Retrieved screen with the records selected by the query.

See the [Scheduling Queries and Viewing Records Retrieved chapter](#) for an explanation of the Query Retrieved screen.

### Reviewing the Query's Purpose, Schedule, or Related Output

From the Query Entry screen, you can access a note about the query, the User-defined Output Entry screen, and the Query and Output Schedule screen.

#### See Also:

[Entering and Viewing a Note to Describe the Query](#)

[Accessing the User-defined Output Entry Screen](#)

[Accessing the Query and Output Schedule Screen](#)

### Entering and Viewing a Note to Describe the Query

After you have entered the information for a query, you can use the Notes field to describe

- the purpose of the query,
- the purpose of the include queries,
- the purpose of the exclude queries, and
- the purpose of the selection criteria.

By using the Notes field, you can enter a description "in English" of the query along with the reasoning for the structure of the query to enable other users to understand what the query will select and why it is set up as is. You can access the Notes field by clicking <Notes>.

## Accessing the User-defined Output Entry Screen

From the Query Entry screen, you can access the User-defined Output Entry screen by clicking <Outputs>.

You use the User-defined Output Entry screen to create a user-defined output that you want to run with the query. See the Understanding Outputs chapter for an explanation of what an output is and how to use the User-defined Output Entry screen.

## Accessing the Query and Output Schedule Screen

From the Query Entry screen, you can access the Query and Output Schedule screen by clicking <Schedule>.

You use the Query and Output Schedule screen to set up a schedule of which queries should be run with which outputs. See the [Scheduling Queries and Viewing Records Retrieved chapter](#) for an explanation of the Query and Output Schedule screen.

## Deleting a Query

You can mark a query for deletion if that query should no longer be used. A query marked for deletion is not deleted until your organization runs delete processing. If you mark a query for deletion and then decide that the query should not be deleted, you can "unmark" the query so that delete processing does not delete it.

### See Also:

[Marking a Query for Deletion](#)

[Unmarking a Query Marked for Deletion](#)

## Marking a Query for Deletion

*Before you mark a query for deletion, you should check to see if the query is used as an include or exclude in any other queries. If you mark the query for deletion, any other query that uses it will not run properly.*

If you want to mark a query for deletion, complete the following steps.

1. Access the Query Entry screen.
2. If you were not working with queries previously, TA displays a message from which you should click <Find> to find the query you want to delete. If your current query, i.e., the last query with which you worked during your current session, appears, click <Find> on the TA toolbar to find the query you want to delete. In both cases, the Find Query window appears.
3. Find the query you want to delete. See "[Finding a Query](#)".
 

**Note:** Make sure your cursor is not in the Selection Criteria block before you move on to step 4; if it is, TA deletes the selection criterion rather than changing the status of the query.
4. Click <Delete> on the TA toolbar.
5. TA displays the message "Are you sure you want to mark query for deletion?"
6. Click <OK> to mark the query for deletion. (Click <Cancel> if you don't want to mark the query for deletion; TA returns to the Query Entry screen.)
7. TA sets the query's status to "D" for delete. When your organization runs delete processing, this query will be deleted.

Once a query is marked for deletion, if it is marked as an include query, exclude query, or both, it is removed from the appropriate lists.

# Team Approach 5.0.1

## Understanding Queries

---

**Note:** When you mark a query for deletion, the <Delete> button on the toolbar changes to the <Undelete> button. See "[Unmarking a Query Marked for Deletion](#)" which follows this section to learn how to use the <Undelete> button.

### Unmarking a Query Marked for Deletion

If you want to unmark a query marked for deletion, complete the following steps.

1. Access the Query Entry screen.
2. If you were not working with queries previously, TA displays a message from which you should click <Find> to find the query you want to unmark. If your current query, i.e., the last query with which you worked during your current session, appears, click <Find> on the TA toolbar to find the query you want to unmark. In both cases, the Find Query window appears.
3. Find the query you want to activate. See "[Finding a Query](#)".
4. Click <Undelete> on the TA toolbar.
5. TA changes the query's status from "D" for delete to "C" for changed. The query is no longer marked for deletion but it must be tested before it can be run or used as an include or exclude query.

See "[Testing a Query](#)".

**Note:** When you unmark a query for marked for deletion, the <Undelete> button on the toolbar changes to the <Delete> button. See "[Marking a Query for Deletion](#)".

### Definition of Ownership

Queries and user-defined outputs are owned by the user who created them. If a query or user-defined output is associated with a user group, then it is also owned by the members of that user group.