

Intermediate Query for Elementary & Secondary Schools Conference 2015

Session Description: The "TOTAL", "KEEP", "SKIP" commands; printing labels, letters, saving and loading query statements, using "AND" or "OR", and multiple table queries at the elementary and secondary level.

1. Query

Functions

What is a Table?

How is a Table Created?

Why is There More Than One Table?

Database Table Information Report

Query Statements – Using One Table

Using Multiple Table Queries

Viewing Data From the Datasheet View

Special Characters

2. Functions

Creating a Report

Labels

Customizing Labels

Query Letter Editor

Letters

Envelopes

Create an ASCII File

Change Data

Save Query Statements

Load Statements

3. Example Queries

4. Query Commands

Total Command

SKIP or KEEP

Using a SKIP Command

Using a KEEP Command

Using STARTUP

FUNCTIONS

There are buttons in a column on the right side of the **QUERY** form. These buttons are used to perform functions to generate reports, labels, files, etc. **after** the query statement has been generated. The **FUNCTIONS** available to perform different tasks are:

- **NEW** clears the text box area of any query currently displayed to enter a new query.
- **RUN** will generate the query statement and display the data.
- **EXCEL** will create an Excel spreadsheet from the data generated. This option is only available in Aeries.net.
- **REPORT** will generate a formatted report from the data generated.
- **LABELS** will generate formatted labels from the data generated.
- **LETTERS** will generate a formatted letter previously created in the **Query Text Editor** program from the data generated.
- **LETTER EDITOR** will store formatted letters created in the **Query Text Editor** program.
- **ENVELOPE** will generate a format to be printed on a legal envelope from the data generated. This option is only available in AeriesCS.
- **TO FILE** will create a text file onto a disc from the data generated. This option is only available in AeriesCS.
- **CHANGE** allows you to make quick changes to data in select fields. This option is only available in AeriesCS.
- **SAVE** allows you to save the query for later use.
- **LOAD** will allow you to re-load a query previously saved.
- **STARTUP** will store a **KEEP** statement that can be setup to run automatically on the opening of Aeries **WITHOUT** having to enter query. This option is only available in AeriesCS.
- **RESET** will reset the **KEEP** and **SKIP** statements.
- **TABLES** will re-load and update data tables. This option is only available in AeriesCS.
- **EXIT** will exit the **QUERY** form.

WHAT IS A TABLE?

A table is a file that stores certain data entered into the **Student Information System**. The **Query** form below displays some of the **tables** in the **Student Information System** database.

The screenshot shows the 'Tables' section of the Aeries Query form. At the top, there is a text input field labeled 'Enter table code' with the word 'LIST' typed into it. Below this, a table is displayed with the following data:

COMMANDS		TABLES		FIELDS		SPECIAL	
LIST	Display output	TBL	Table Name	FLD	Field Name	?	Description
TOTAL	Display totals	ABS	Absence Code Table			\	New Line
SKIP	Bypass records	ACT	Activities and Awards				Barcode

To the right of the table, there is a vertical column of buttons labeled 'FUNCTIONS': NEW, RUN, REPORT, LABELS, and LETTERS. The 'LABELS' button is highlighted with a blue background.

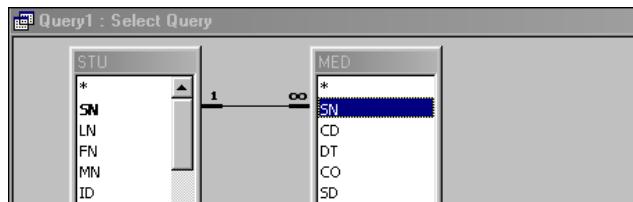
HOW IS A TABLE CREATED?

A table initially is empty and contains NO data. The form displayed below is the **Teachers** form. When you click on **Add** a Teacher number is assigned. After all data is entered and **ENTER** is pressed a record is created in the **TCH** table and technically the table is created. As each teacher is entered a unique teacher number is assigned and a new record is created. The table will now store all new records entered.

Tch#	Teacher Name	First Name	Last Name	Title	Room	Tag	Max	Total	Left	Boys	Girls	
1	Aldrich	Mary	Aldrich		1		20	15	5	4	11	
ElecTag	E-mail Address		Class ID	Staff ID	Highly Qlfid Tchr	Staff ID 2	HQT	Staff ID 3	HQT			
G			0	990001		0		0				
Website			Low Grd	High Grd	User1	User2	User3	User4	User5	User6	User7	User8
			0	0								

WHY IS THERE MORE THAN ONE TABLE?

Instead of creating a large table containing all data, smaller tables are created for different types of data. For instance the medical file contains the student number, medical code, details, etc. To access data from different tables a field is used to create a relationship between tables. Such as, the **MED** and **STU** tables contain the student number (**SN**) which creates a connection between tables.



If the same student number is located in the **MED** table data can be selected for the student, such as a medical condition the student has. This helps save space having multiple tables with a joining field due to the fact that student data is not duplicated, such as the student's first name and last name in both tables. It also eliminates double entry of student data into all tables.

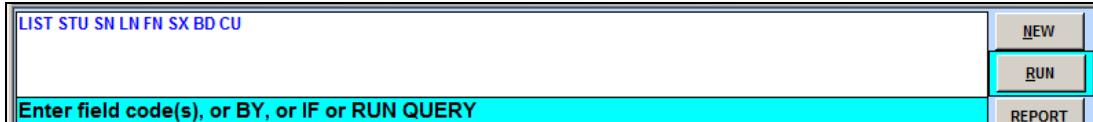
DATABASE TABLE INFORMATION REPORT

The **Database Table Information Report** can be printed from **View All Reports**. This report can print all tables located within Aeries. The report will provide details about the table and fields as well as any related table and field name. This report can be a valuable tool when building a multiple table query.

Database Table Information			8/12/2014	
Table Code:	ATT	Table Name: Attendance Data	Record Count:	6479
Table Description			Data Type	District Managed
Maintains Dates of Enrollment, Absences and Absence Reasons, and Un-enrollment			Data	N
Relationships	Field Name	Foreign Table	Foreign Field	
ABS	CD	ATT	AL	
STU	SN	ATT	SN	
DAY	DY	ATT	DY	
TCH	TN	ATT	TN	
Field Code	Field Heading	Field Description	Type of Data	Maximum
SN	Student#	Student number	Numeric	2,147,483,648
DY	Day#	Day number	Numeric	32,767
CD	Ent/Lv	Enter/leave code	Text	1

QUERY STATEMENT USING ONE TABLE

A **Query Statement** can be created with one or more tables. The following query statement utilizes one table, which only accesses data from the **STU** table.

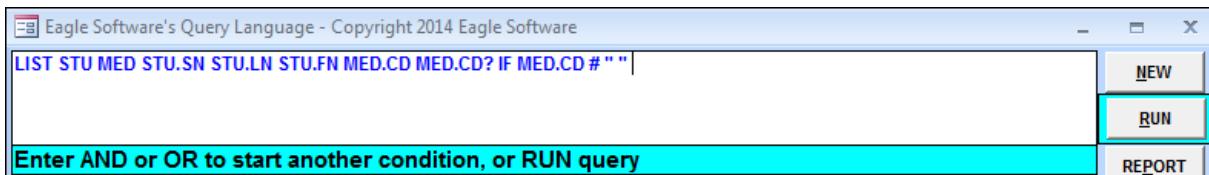


When the query is generated the following will display with the data selected.

Student#	Last Name	First Name	Sex	Birthdate	Tchr Num
1	Aadasian	Robert	M	5/17/2003	628
2	Aadasian	Ruben	M	12/23/2008	730
3	Aguilar	Jonathan	M	7/16/2004	526
4	Aguilar	LaTonya	F	10/13/2005	423

USING MULTIPLE TABLE QUERIES

A **Multiple Table Query** uses more than one table to access different information. For example, the query below will create a list of students with a medical condition by accessing data from the **STU** and **MED** tables.



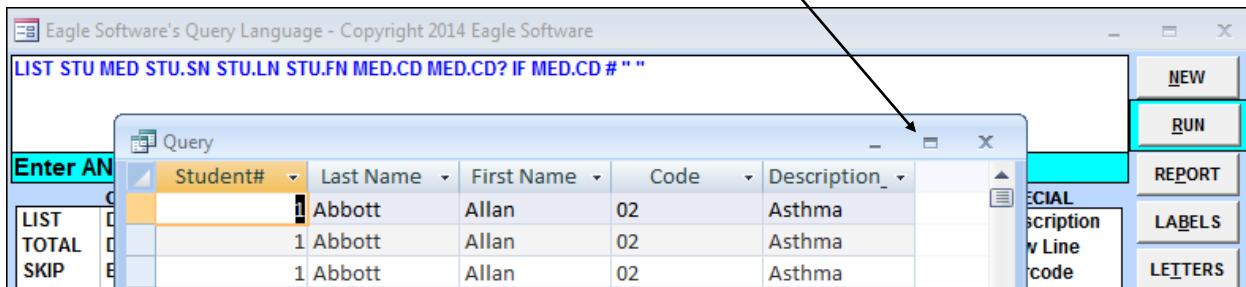
Notice that when more than one table is used in a query the table name comes before the field name. For example, **MED.CD**. This is due to the possibility of two tables having a field with the same name but containing different data. The following is the definition for the last Query:

LIST	= display on the screen
STU MED	= using two tables, student data and medical
STU.SN	= student number from the STU table
STU.LN	= student last name from the STU table
STU.FN	= student first name from the STU table
MED.CD	= medical code from the MED table
MED.CD?	= medical code description from the COD table
IF MED.CD # " "	= if the medical code is not blank

This Query will display the student number, last name, first name, medical code, and medical comments for students with a medical problem.

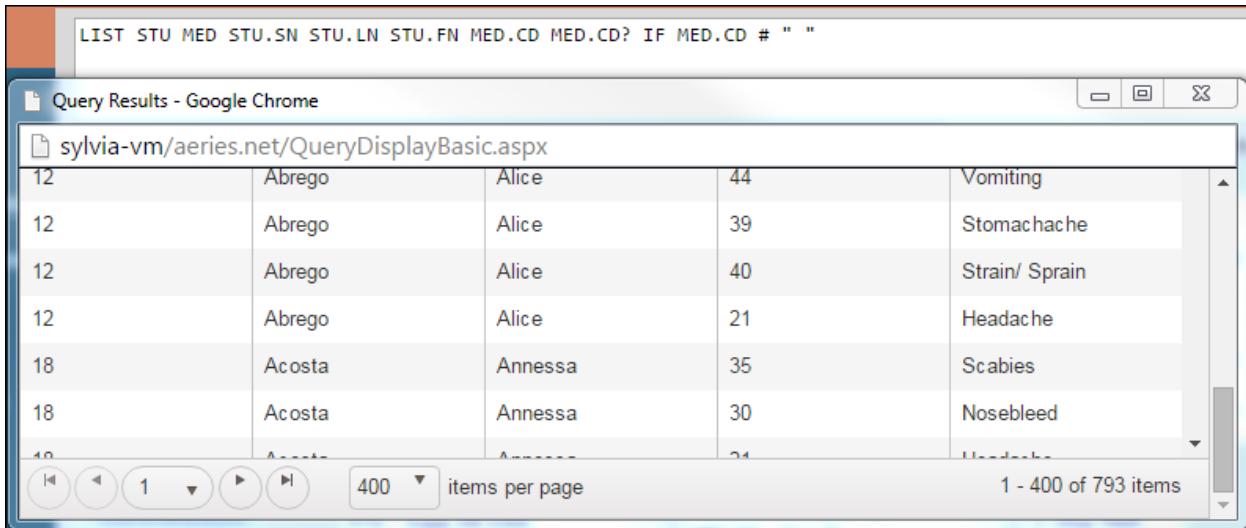
VIEWING DATA FROM THE DATASHEET VIEW

After a query is generated the data selected will display in a datasheet that resembles Excel. To maximize the form and **View the Data** click the mouse on the **middle box** in the right corner. The data will then display maximized on the screen.



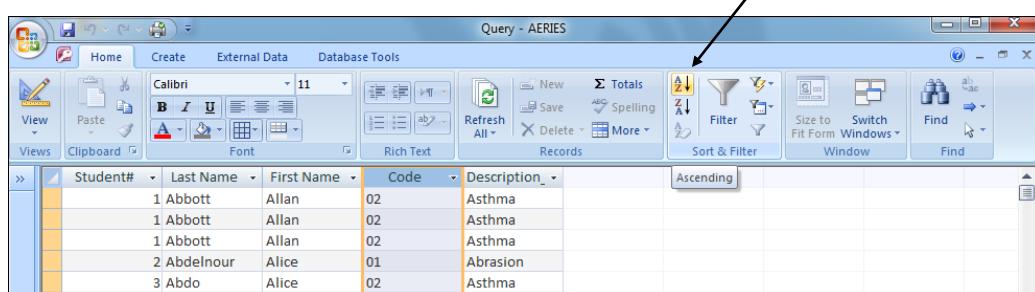
The data can be filtered to only display select data. Also, the data displayed can be sorted in ascending or descending order.

Aeries.Net returns results in a similar format using a pop-up window with scroll bars to the right and the bottom of the window. A maximum of 400 records will display per page. Clicking on the page numbers at the bottom of the window will display the next set of results.

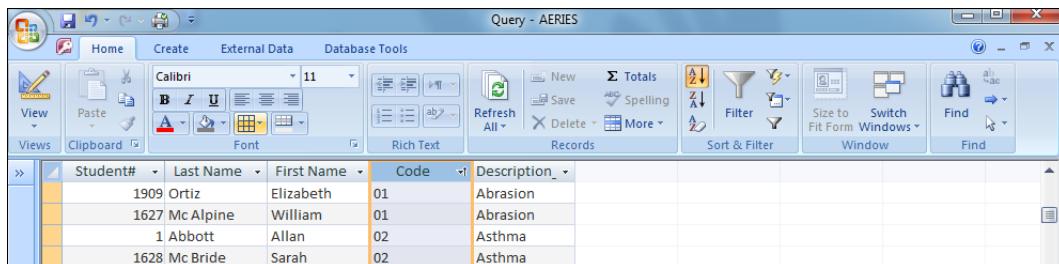


SORT THE DATA DISPLAYED

To change the sort order of the data in AeriesCS click the mouse on the column heading and the entire column will be hi-lighted. To sort the column in ascending order, click the mouse on the **A-Z** button on the toolbar.

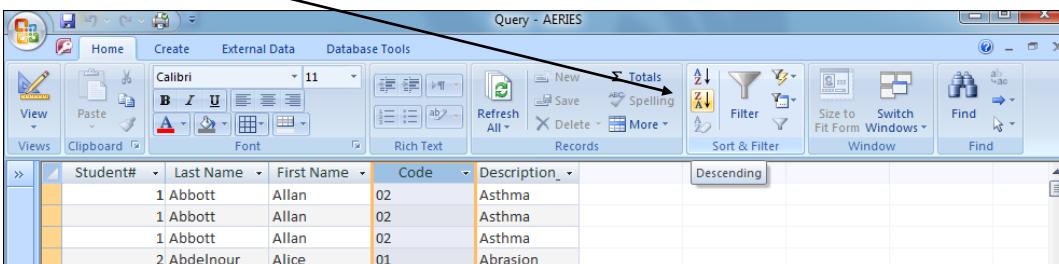


The data will then be resorted in ascending order by the column selected.



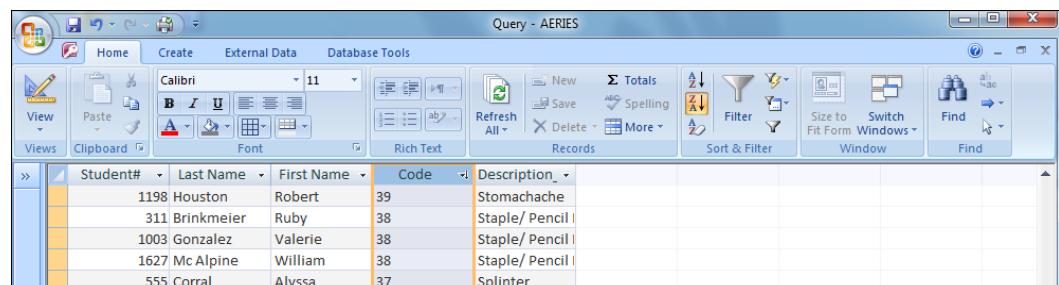
Student#	Last Name	First Name	Code	Description_
1909	Ortiz	Elizabeth	01	Abrasion
1627	Mc Alpine	William	01	Abrasion
1	Abbott	Allan	02	Asthma
1628	Mc Bride	Sarah	02	Asthma

To sort the column in descending order, click the mouse on the column heading and click the mouse on the Z-A button on the toolbar.



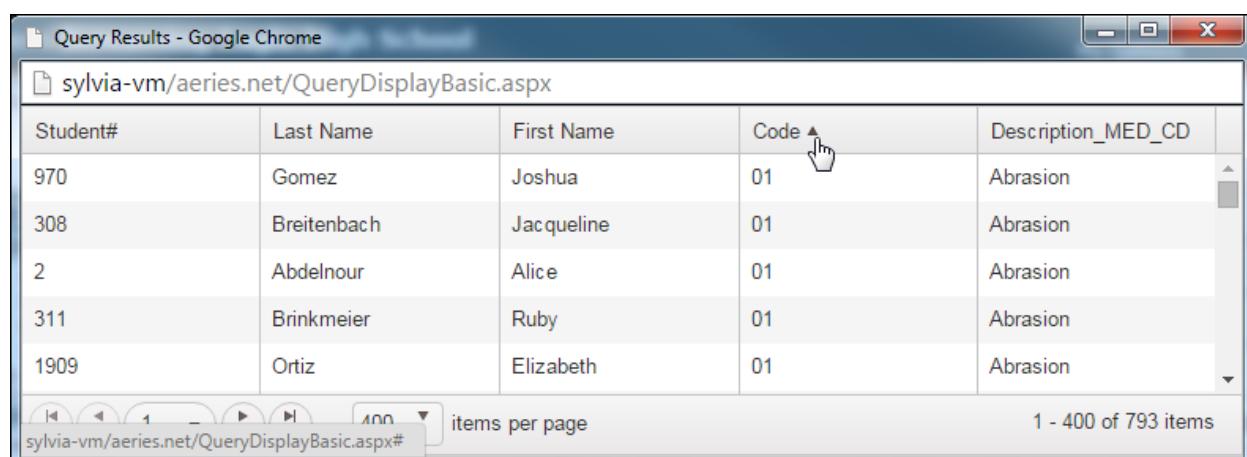
Student#	Last Name	First Name	Code	Description_
1	Abbott	Allan	02	Asthma
1	Abbott	Allan	02	Asthma
1	Abbott	Allan	02	Asthma
2	Abdelnour	Alice	01	Abrasion

The data will then be resorted in descending order by the column selected.



Student#	Last Name	First Name	Code	Description_
1198	Houston	Robert	39	Stomachache
311	Brinkmeier	Ruby	38	Staple/ Pencil
1003	Gonzalez	Valerie	38	Staple/ Pencil
1627	Mc Alpine	William	38	Staple/ Pencil
555	Corral	Alyssa	37	Splinter

In Aeries.net clicking on the column header will sort the data in ascending order by that column. Clicking it a second time will sort the data in descending order by that column, and clicking it a third time removes the sort on that column.



Student#	Last Name	First Name	Code	Description_MED_CD
970	Gomez	Joshua	01	Abrasion
308	Breitenbach	Jacqueline	01	Abrasion
2	Abdelnour	Alice	01	Abrasion
311	Brinkmeier	Ruby	01	Abrasion
1909	Ortiz	Elizabeth	01	Abrasion

FILTER THE DATA DISPLAYED

To filter specific data in AeriesCS, click the mouse on information within a column. For example, Code 21 was selected below. Click the mouse on the **Filter by Selection** button on the toolbar.

The screenshot shows the AeriesCS application window titled 'Query - AERIES'. The 'Sort & Filter' toolbar is visible at the top, with the 'Filter by Selection' button highlighted. A dropdown menu is open, showing filtering options based on the selected value '21' in the 'Code' column. The main grid displays student data, with the 'Code' column showing various values including '21'.

The only data that will display will be the data selected. For example, Code 21 was hi-lighted and now only the students with a Code 21 in the MED table display.

The screenshot shows the AeriesCS application window after applying the filter. Only the rows where the 'Code' value is '21' are displayed in the grid. The other rows are hidden.

To remove the filter from the data displayed, click the mouse on **Remove Filter** button on the toolbar and all data will now be displayed.

The screenshot shows the AeriesCS application window with the 'Remove Filter' button on the 'Sort & Filter' toolbar highlighted. The grid now displays all the original data rows again.

To search for specific data, click the mouse within a column. For example use the **Last Name** field. Click the mouse on the **Find** button on the toolbar. Enter the Last Name in the **Find What** field. Click the mouse on the **Find Next** button. The cursor will display in the first record located. To continue the search, click the mouse on the **Find Next** button until the record is located.

The screenshot shows the AeriesCS application window with the 'Find' dialog box open. The 'Find What' field contains 'Abesamis'. The 'Look In' dropdown is set to 'Last Name'. The 'Match' dropdown is set to 'Any Part of Field'. The 'Search' dropdown is set to 'All'. The 'Find' button is highlighted. The main grid shows student data, with one row for 'Abesamis' highlighted.

A print out can also be generated from this form by clicking the mouse on the printer icon immediately above the data box. The formatting for this printout will be exactly the same as the screen display. A nicely formatted report can also be generated. Clicking the **LOWER "X"** in the upper right hand corner of the screen to exit the screen and choosing **REPORT** from the **QUERY** form.

SPECIAL CHARACTERS

There are Special Characters that can be utilized within a query statement that will provide shortcuts to the final query generated. These characters are displayed on the right hand side of the form under **Special**.

SPECIAL	
?	Description
\	New Line
	Barcode
	Day
	Month
	Year
	Years (Age)
NM	Full Name
MI	Middle Initial

- ? will connect a field to the **COD** table and display the code description.
- \ forces the print line to drop down one line
- **Barcode** will generate a number into a scanning barcode on a label **Rswide39** font must be in the windows/fonts/directory
- **Day** will only display the Day from the date
- **Month** will only display the Month from the date
- **Year** will only display the Year from the date
- **Years** will calculate the number of years from the date
- **NM** will join **LN** with a **comma** and the **FN**, for example: Abbott, Allan
- **MI** will only display the first letter of the **MN**

The following is an example of a query utilizing Special Characters listed above:

Eagle Software's Query Language - Copyright 2014 Eagle Software

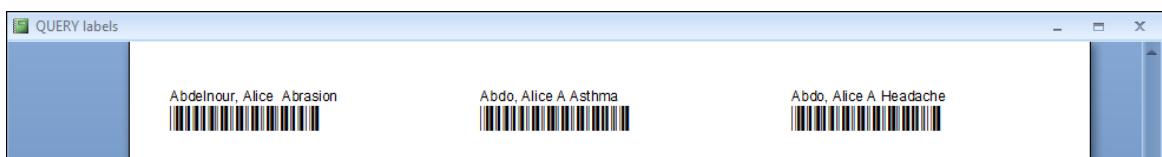
LIST STU MED STU.NM STU.MI MED.CD?\ BARCODE STU.ID IF MED.CD # ""

Query

Student Name	M/I	Description_MED_CD	Expr1003	Expr1004	ID#
Abbott, Allan	J	Asthma	\	BARCODE	99400001
Abdelnour, Alice		Abrasion	\	BARCODE	99400002
Abdo, Alice	A	Asthma	\	BARCODE	99400003

Labels

After the query statement is closed and Labels is selected for Avery 5160 labels the following will display. The \ has forced a new line and **Barcode** has converted the student ID number into a scanning barcode.



The following example uses the **NM** and **MI** functions: Notice that **NM** will join the Last Name and First Name together. The **MI** will strip off all letters from the Middle Name and only display the first initial.

Eagle Software's Query Language - Copyright 2014 Eagle Software

LIST STU LN FN NM MN MI

Query

Last Name	First Name	Student Name	Middle Name	M/I
Arias	Christopher	Arias, Christopher	R	R
Arias	Jean-Jacques	Arias, Jean-Jacques	Carlos	C
Arias	Norman	Arias, Norman	Adam	A

SPECIAL

Description
New Line
Barcode

The question mark (?) will pull the Code Descriptions for the field from the COD table. However the Code values for every field are not ALL stored within the COD table for that field. There are some fields that contain the exact same codes and the descriptions that pull this information from another field.

For example the Correspondence Language field will pull the codes and descriptions from the Home Language field. The query is created a little differently. After entering the field name, question mark (?) you will need to add the field where the codes are stored. The query below has: CL?HL which will have the Correspondence Language field pull the codes and descriptions from the Home Language field.

The screenshot shows the Eagle Software's Query Language interface. The main window displays a table titled 'Query' with columns: Student#, Last Name, First Name, CorrLng, and Description_CL. The data includes three rows: 12 Abrego, Alice, 01, Spanish; 15 Aceves, Steven, 01, Spanish; and 16 Acharya, Josiah, 99, Other non-English. To the right of the table is a context menu with options: NEW, RUN, REPORT, LABELS, and LETTERS. The 'RUN' button is highlighted with a blue border.

The **DAY**, **MONTH**, **YEAR** and **YEARS** are functions that will allow you to split the day, month or year from a **Date** field and can also total the number of years from a **Date** field and today's date.

The query below has **ED.DAY ED.MONTH ED.YEAR** in the query and when selected will only display the Day, Month or Year of the **Date** field. The **ED.YEARS** will calculate the total number of years from the **Date** field using today's date.

The screenshot shows the Eagle Software's Query Language interface. The main window displays a table titled 'Query' with columns: Enter Date, Day, Month, Year, and Years. The data includes three rows: 9/5/2011, 5, 9, 2011, 2; 9/6/2012, 6, 9, 2012, 1; and 8/23/2013, 23, 8, 2013, 0. To the right of the table is a context menu with options: NEW, RUN, REPORT, LABELS, and LETTERS. The 'RUN' button is highlighted with a blue border.

CREATING A REPORT

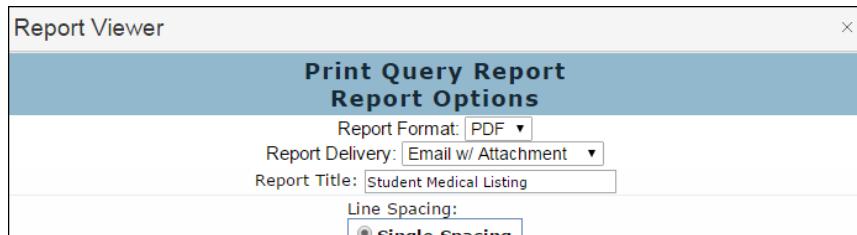
After running the query statement a formal report can be generated by clicking the mouse on the **REPORT** button.

The screenshot shows the Eagle Software's Query Language interface. The main window displays a table titled 'Query' with columns: Student#, Last Name, First Name, CorrLng, and Description_CL. The data includes three rows: 12 Abrego, Alice, 01, Spanish; 15 Aceves, Steven, 01, Spanish; and 16 Acharya, Josiah, 99, Other non-English. To the right of the table is a context menu with options: NEW, RUN, REPORT, LABELS, and LETTERS. The 'REPORT' button is highlighted with a blue border.

The following box will display. In the **Query Report Parameters** box, type the **Report Title** that will print at the top of the report. Select the **Report Line Spacing**, Single, Double or Triple spaced.

The screenshot shows the 'Query Report Parameters' dialog box. It contains a 'Report Title (if any)' input field with the value 'Student Medical Listing'. Below it is a 'Report line spacing' section with three options: 'Single spaced' (checked), 'Double spaced', and 'Triple spaced'. At the bottom are 'OK' and 'Cancel' buttons.

Aeries.net includes an additional **Report Format** and **Report Delivery** options. Reports can be formatted as a PDF, Word Document (RTF), Spreadsheet (XLS) or Text file (TXT).



The **Report Delivery** options are:

- **Email w/ Attachment** – an email is sent to the user notifying them of the report parameters and the completion of the report
- **Email w/o Attachment** – an email is sent to the user notifying them of the report parameters and the completion of the report along with an attachment of the report
- **None** – no emails are sent to the user.

The following report will be displayed on the screen. To print the report, click the mouse on the **Printer** icon at the top left corner of the screen: The report displayed on the screen will be printed.

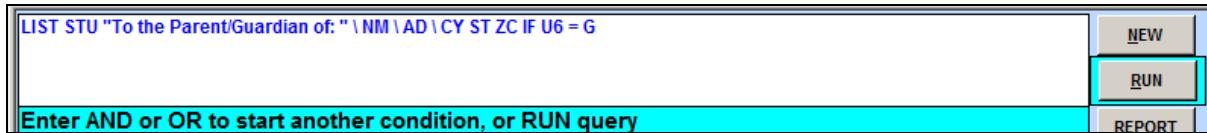
Screaming Eagle High School				8/12/2014
2014-2015		Student Medical Listing		Page 1
Student#	Last Name	First Name	Code	Comments
1	Abbott	Allan	02	Needed Inhaler
1	Abbott	Allan	02	Needed Inhaler
1	Abbott	Allan	02	Needed Inhaler
2	Abdelnour	Alice	01	Fell while playing soccer. Severe abrasion to hip. Cleaned/BA. Called parent for permission for student to return to class.

LABELS BUTTON

After running a query statement, the **LABELS** button can be utilized to create different types of labels, such as, mailing labels, student folder labels, etc. Listed below is an example of a **Query** to print address labels to the parents of Gate students.

To create address labels or any label containing more than one line, the '\n' must be used in the query statement. This will force the query to drop down a line when printing the labels. To have a text expression print, double-quotes must be placed around the text expression.

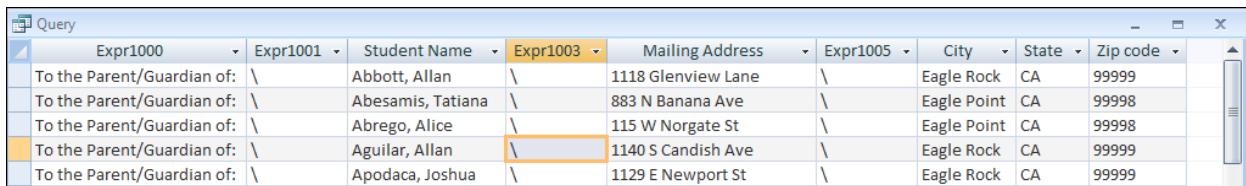
For Example: “To the Parents/Guardians of: ”



The following is the definition for the query above:

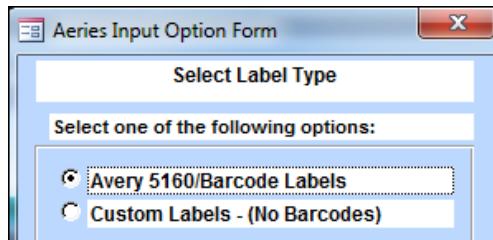
LIST	= display on the screen
STU	= using the student data table
"To the Parents"	= creates a text statement
\	= drops down to the next line
NM	= creates students full name from the LN and FN in STU table
\	= drops down to the next line
AD	= students address from the STU table
\	= drops down to the next line
CY	= students city from the STU table
ST	= students state from the STU table
ZC	= students zip code from the STU table
IF U6 = G	= only select if the U6 field contains a G for GATE students (this field is only an example for GATE)

To create the query, click the mouse on the **RUN** button. The following **ACCESS** query will display. The text expression and the '\' entered create their own field for each student record. The '\' field is read by query and forces the program to drop down a line when printing.



	Expr1000	Expr1001	Student Name	Expr1003	Mailing Address	Expr1005	City	State	Zip code
	To the Parent/Guardian of:	\	Abbott, Allan	\	1118 Glenview Lane	\	Eagle Rock	CA	99999
	To the Parent/Guardian of:	\	Abesamis, Tatiana	\	883 N Banana Ave	\	Eagle Point	CA	99998
	To the Parent/Guardian of:	\	Abrego, Alice	\	115 W Norgate St	\	Eagle Point	CA	99998
	To the Parent/Guardian of:	\	Aguilar, Allan	\	1140 S Candish Ave	\	Eagle Rock	CA	99999
	To the Parent/Guardian of:	\	Apodaca, Joshua	\	1129 E Newport St	\	Eagle Rock	CA	99999

After the query statement has been generated, click the mouse on the **LABELS** button and the following selection box will display. The default will be for **Avery 5160** labels. Click the mouse on the **OK** button.



Depending on the size of your query, the following message might display. This message is only an indication that all data may not print on a label. Click the mouse on the **OK** button.



In AeriesCS the following **QUERY Labels** will display. Verify the label data is correct and will fit on the label selected. To print the labels, click the mouse on the **Printer** icon at the top of the screen.

QUERY labels		
To the Parent/Guardian of: Abbott, Allan 1118 Glenview Lane Eagle Rock CA 99999	To the Parent/Guardian of: Abesamis, Tatiana 883 N Banana Ave Eagle Point CA 99998	To the Parent/Guardian of: Abrego, Alice 115 W Norgate St Eagle Point CA 99998
To the Parent/Guardian of: Aguilar, Allan 1140 S Candish Ave Eagle Rock CA 99999	To the Parent/Guardian of: Apodaca, Joshua 1129 E Newport St Eagle Rock CA 99999	To the Parent/Guardian of: Aquino, Kenneth 1149 E Claraday Eagle Rock CA 99999

In Aeries.net a report is generated to create the labels onto Avery 5160 labels. To properly print out the labels we suggest creating a PDF of the labels and setting the "Print Scaling" option to "None".

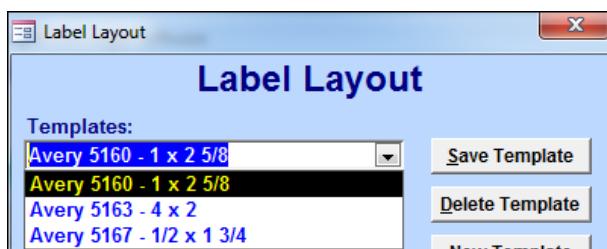


CUSTOMIZING LABELS

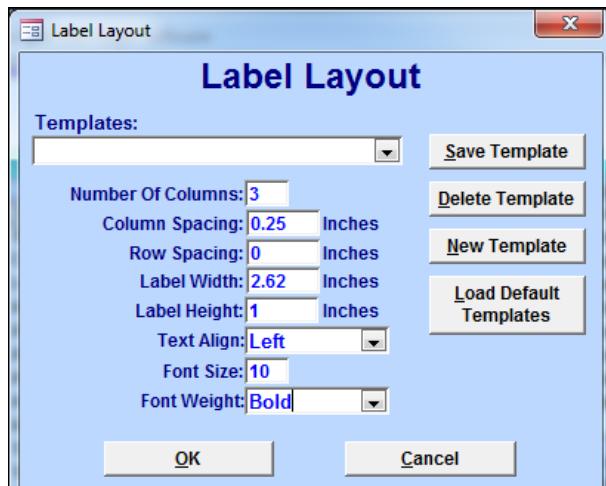
To customize your label click the mouse on **Custom Labels** and click the mouse on the **OK** button.
Please note that Barcodes cannot be printed on a custom label.



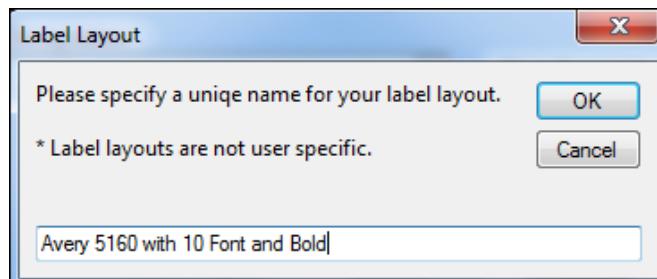
The following form will display. This form will allow you to create custom label templates that can be **Saved**. Default templates for Avery labels can also be loaded by clicking the **Load Default Templates** button. Once loaded the drop down will display all available templates.



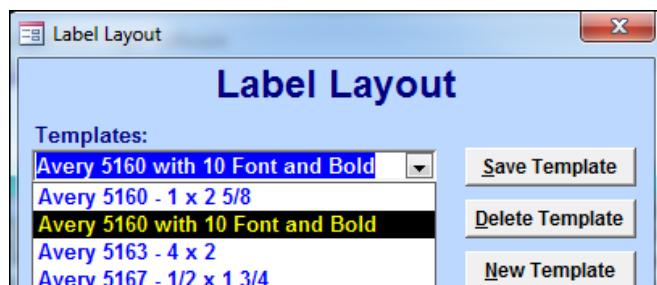
To create a new template from a saved template, select a template from the drop down and enter the label format. For example, the Avery 5160 default template was selected from the drop down. The Font Size was changed to 10 and Font Weight to Bold.



Click the mouse on the **Save Template** button and the following message will display. Enter the name of the new template which **MUST** be different than one in the drop down listing. Click the mouse on the **OK** button and a message will display indicating that it was saved.



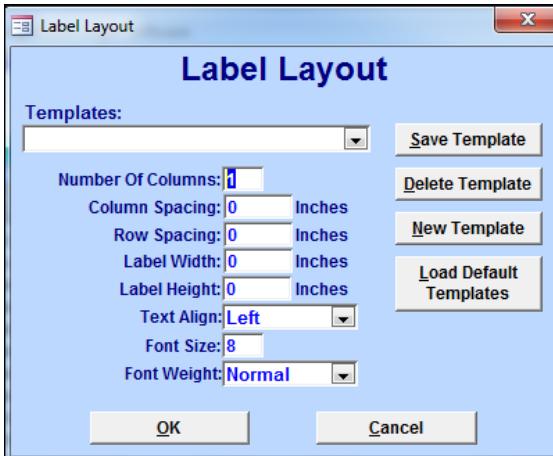
After the template is saved it will display in the drop down. To print these labels select the template and click the mouse on the **OK** button.



The Labels will be generated with the template selected. For example, this is an Avery 5160 that now has 10 Font and is Bold.

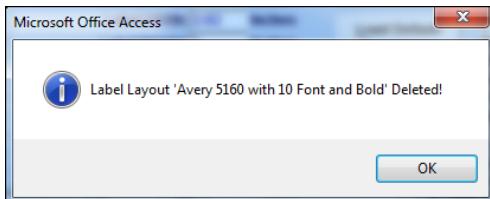
To the Parent/Guardian of: Abbott, Allan 1118 Glenview Lane Eagle Rock CA 99999	To the Parent/Guardian of: Abesamis, Tatiana 883 N Banana Ave Eagle Point CA 99998	To the Parent/Guardian of: Abrego, Alice 115 W Norgate St Eagle Point CA 99998
--	---	---

To create a new template with no defaults from another label click the mouse on the **New Template** button. Notice that the defaults changed. Enter **ALL** sizes into the fields displayed and click the mouse on the **Save Template** button. A message will display to enter the name of the **Template**. Click the mouse on the **OK** button and the template will now display in the drop down.



NOTE: Remember to always print out one page on regular paper and compare it to the labels to ensure it formatted properly.

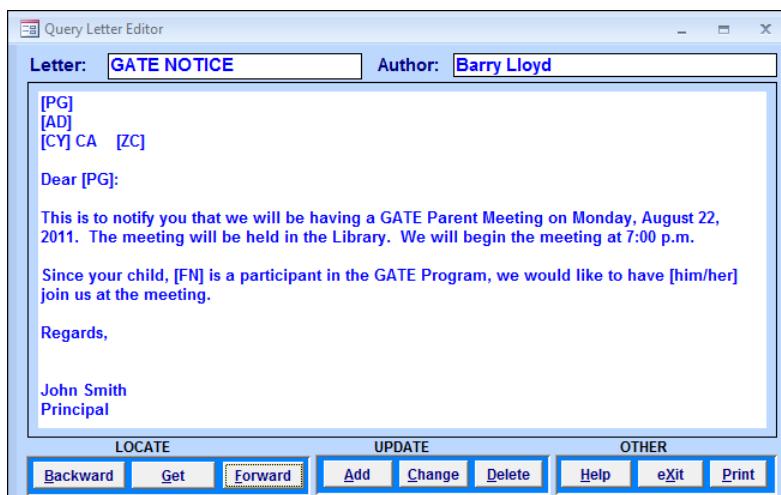
To delete a **Saved Template** click the mouse on the drop down and select the template. Click the mouse on the **Delete Template** button and the following message will display. Click the mouse on the **OK** button and the template will no longer display in the drop down.



LETTER EDITOR BUTTON

The **LETTER EDITOR** allows you to create letters that will contain text along with merged data from a query statement. Once the letter has been created in the **Query Letter Text Editor**, a query statement **must** be generated to pull the data from the tables and fields selected.

The **LETTER** button is utilized to display the letters available. The letter is selected and the data is merged into the letter.



In Aeries.net a list of letters appears to the left of the letter editor. Click on the **Letter ID** to view a letter. The **Add** button can be used to add a new letter.

The screenshot shows the 'Query Letter Editor' window. On the left, there is a list of letter IDs: 'First Day Welcome', 'GATE NOTICE' (which is selected and highlighted in green), and 'Graduation Day Info'. The main area displays the content of the selected letter: 'GATE NOTICE' by 'Barry Lloyd'. The letter text is as follows:

[PG]
[AD]
[CY] CA [ZC]

Dear [PG]:

This is to notify you that we will be having a GATE Parent Meeting on Monday, August 22, 2011. The meeting will be held in the Library. We will begin the meeting at 7:00 p.m.

Since your child, [FN] is a participant in the GATE Program, we would like to have [him/her] join us at the meeting.

Regards,

At the bottom of the window, there are buttons for 'Add' (circled in blue), 'Change', 'Delete', 'Print', and 'Return to Query'.

Prior to creating a letter it must be determined which fields will be utilized. Click the mouse on the **LETTER EDITOR** button. Click the mouse on the **Add** button and enter the letter name in the **Letter:** field. Press **Tab** and the **Author:** field will automatically display the current user's name. Press the **Tab** key until the **cursor** displays at the top of the white text box.

The upper left hand corner usually contains who the letter will be sent to, for example to the Parent. On the first line type **[PG]** for the Parent/Guardian name and press **Enter**. The cursor will drop down to the second line. Type **[AD]** for the address and press **Enter**. The cursor will drop down to the third line. Type **[CY] [ST] [ZC]** and press **Enter**.

A school heading for the letter will automatically be created with the school name, address, and the current date at the top of the letter. **This heading will not display in the letter editor but will print when submitted.**

NOTE: ALL field codes MUST have a square bracket surrounding them and there must be a space between each field but not within the brackets.

Type the remainder of the letter combined with text and any field codes. An example of a letter is displayed below. A query function is available that will insert **[he/she]** or **[his/her]** into the letter for the proper sex but the sex field **must** be used in the query statement.

The screenshot shows the 'Query Letter Editor' window with the letter 'GATE NOTICE' by 'John Smith'. The letter text is as follows:

[PG]
[AD]
[CY] CA [ZC]

Dear [PG]

This is to notify you that we will be having a GATE Parent Meeting on Monday, August 20, 2011. the meeting will be held in the Library. We will begin the meeting at 7:00 p.m.

Since your child, [FN] is a participant in the GATE Program, we would like to have [him/her] join us at the meeting.

Regards,

The text 'This is to notify you that we will be having a GATE Parent Meeting on Monday, August 20, 2011. the meeting will be held in the Library. We will begin the meeting at 7:00 p.m.' and 'Since your child, [FN] is a participant in the GATE Program, we would like to have [him/her] join us at the meeting.' is displayed in blue, indicating it is a query placeholder.

For example, on the third line where **him/her** is used, if the sex code for a student is **F** the above letter will read "we would like to have her join us".

After the letter is complete, click the **Exit** button at the bottom of the form. A query can now be generated through the **Query** option and the **Letter** function is utilized to generate the letters.

NOTE: Each field used MUST be in the query statement or will be blank when the letter prints.

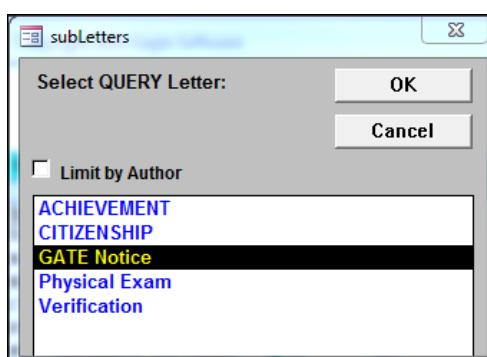
LETTERS BUTTON

The **LETTERS** option is used to generate a formatted letter created in the **Query Letter Editor**. Fields are added into the content of the letter. When the query is generated these fields will be included in the query statement to be merged into the letter. After all fields are selected including the **SX** field, for the **[him/her]** statement, click the mouse on the **RUN** button. Generate the query and verify all data was selected and close out the query statement.

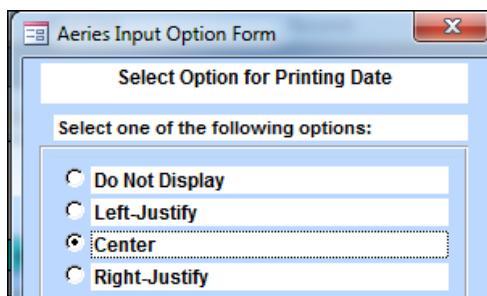


NOTE: ALL fields setup in the letter MUST be in the query or will be blank when the letter prints.

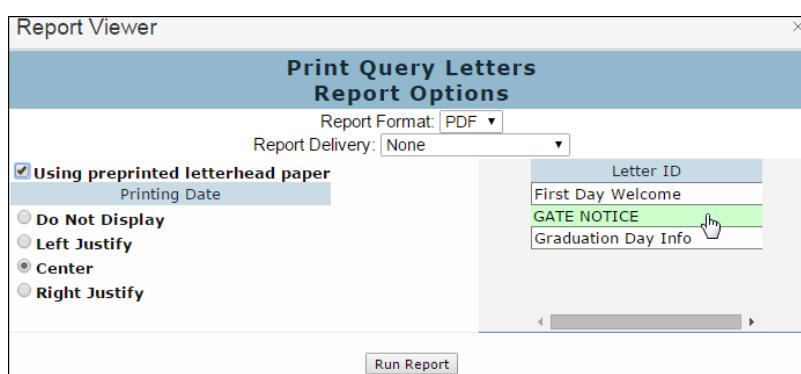
Click the mouse on the **LETTER** button and the following selection box will display. Click the mouse on the letter to be generated and click the mouse on the **OK** button.



A message will display to select whether or not the letterhead needs to be printed. If pre-printed letterhead will be used click the mouse on the **No** button. To print letterhead click the mouse on the **Yes** button. Select where on the letter the date will be printed and click the mouse on the **OK** button.



In Aeries.net all of the selections are on one screen.



The letter will be generated and the data selected in the query will be merged into the letter. The following is an example of the Gate Letter. Notice the **SX** field was selected in the query statement and **HER** was entered into the letter due to this student being a female.

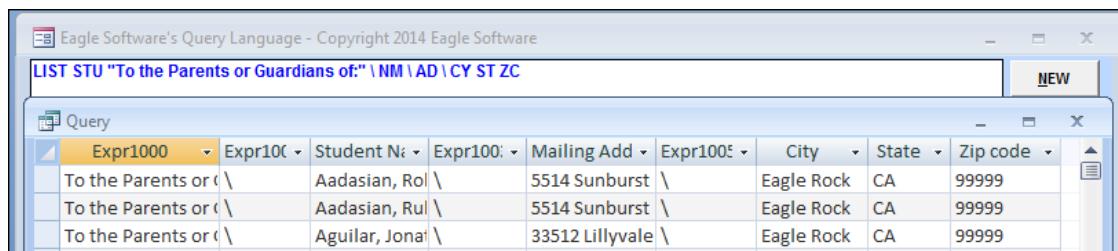
Golden Eagle Elementary School 9950 Comet Street Eagle Point, CA. 95990
<p style="text-align: center;">August 13, 2014</p> <p>M/M A Aguilar 33512 Lillyvale Ave Eagle Point CA 99998</p> <p>Dear M/M A Aguilar</p> <p>This is to notify you that we will be having a GATE Parent Meeting on Monday, August 20, 2011. The meeting will be held in the Library. We will be holding the meeting at 7:00 p.m.</p> <p>Since your child Stephanie is a participant in the GATE Program, we would like to have her join us at the meeting.</p>

REMEMBER - ALL fields in the letter MUST be in the query statement or will be blank when the letter prints. For example, the CY (city) field was not selected for this letter and City is blank.

Golden Eagle Elementary School 9950 Comet Street Eagle Point, CA. 95990
<p style="text-align: center;">August 13, 2014</p> <p>M/M A Aguilar 33512 Lillyvale Ave CA 99999</p>

ENVELOPES

When creating envelopes the query generated is the same as creating an address label. Envelopes containing more than one line must utilize the '\>' in the query statement to force the query to drop down a line when printing the envelopes. After the query has been generated click the mouse on the 'X' to close out the query. Click the mouse on the **Envelope** button.

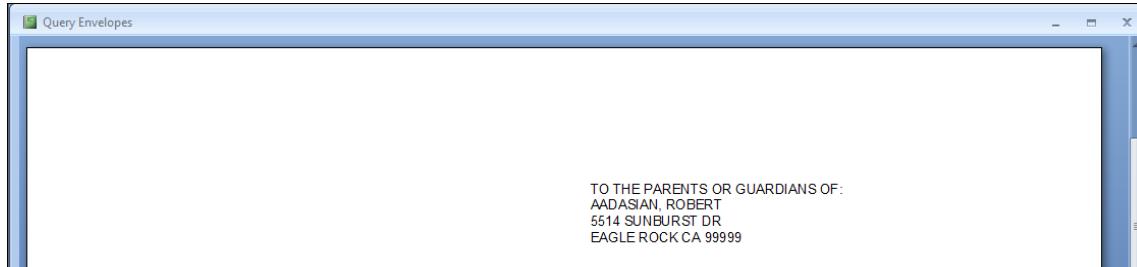


The following selection form will display. This will determine where the address should print which depends upon how the envelopes are loaded into the printer. Some feed trays load the envelope on the left side and some load the envelope in the center.

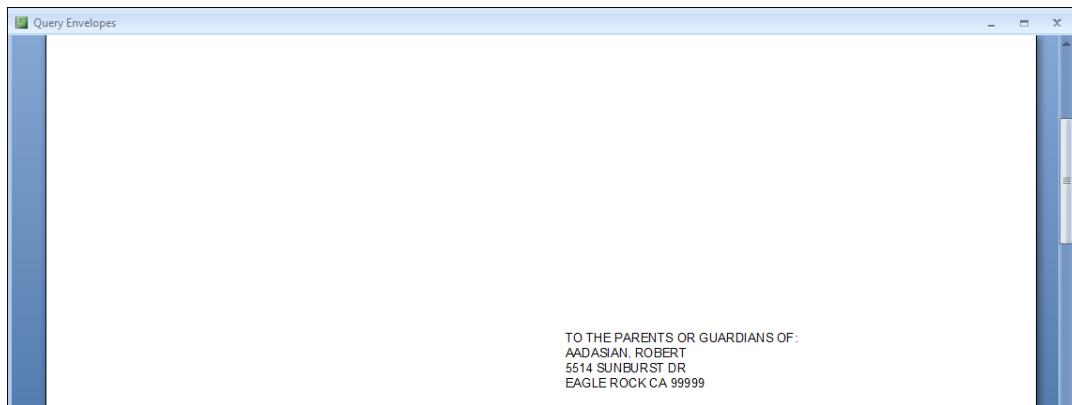


After the selection is made, click the mouse on the **OK** button. The **Check Printer Lineup** message will display. Select **Yes** or **No** depending on whether or not you need to test the printer lineup. When done with the **Printer Lineup**, click the mouse on the **Exit** button.

The screen will display full screen. The information selected will display differently depending upon which option was selected. The following is for Option 1 and will print at the top for the Left Side of the Feeder Tray.



The following is Option 2 which displays in the middle for the Center Feeder Tray.

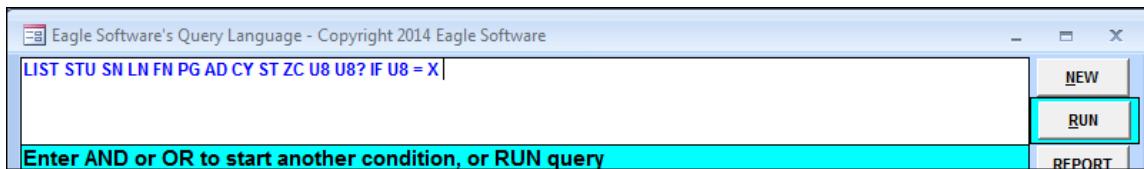


NOTE: Each Printer is different and may require different options to be set in order for the envelope to print.

CREATING AN ASCII FILE

A query statement can be generated and an ASCII file can be created and copied to your C: drive or to a floppy disc on your A: drive. Type a **LIST** query statement in the query text box. Certain criteria can be utilized for specific needs.

For example, the query below is utilizing the **U8** field, which has been used to flag parents that wish to be excluded from all mailings or contacts.

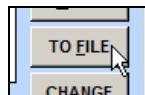


NOTE: Notice a “?” is used after U8 in the query statement. If a code is setup in the COD table for the field selected, the query will display the description. For example, Mail Exclusion.

This query will generate records of parents requesting Mail Exclusion. After the query has been entered, click the mouse on the **RUN** button. Verify the query statement has the correct data as selected and that only flagged students were captured. For example, an **X** in the **U8** field is for mail exclusion. Once the data is verified, close this form by clicking the bottom “X” in the upper right-hand corner.

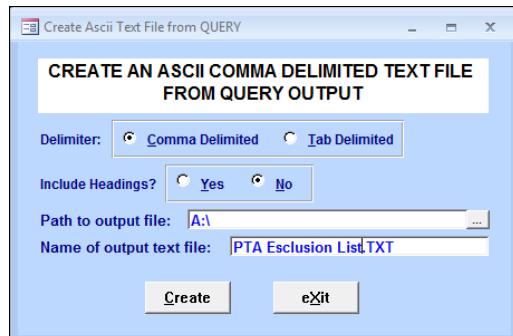
Student#	Last Name	First Name	Parent/guard	Mailing Add	City	State	Zip code	User8	Description_
332 Gattdua	Jake	M/M G Gattdua	1132 E Adams F	Eagle Rock	CA	99999	X		Mail Exclusion
968 Gomez	Allan	M/M G Gomez	22130 E Vmentor	Eagle Rock	CA	99999	X		Mail Exclusion
1859 Nunez	Ethan	M/M N Nunez	1142 E Old Blvd	Eagle Rock	CA	99999	X		Mail Exclusion
2330 Schlichting	Jonathan	M/M S Schlicht	22101 E Mesario	Eagle Rock	CA	99999	X		Mail Exclusion
2538 Thropay	Javier	M/M T Thropay	2279 Palomino	Eagle Rock	CA	99999	X		Mail Exclusion

Click the mouse on the **TO FILE** button. A text box will display allowing either a comma delimited or tab delimited file to be created. A comma-delimited file places a comma between each of the fields in the text file that were selected in the query. A tab-delimited file places a tab between each of the fields selected.



NOTE: The type of text file must be determined prior to selecting the TO FILE option especially when creating this file for an outside vendor.

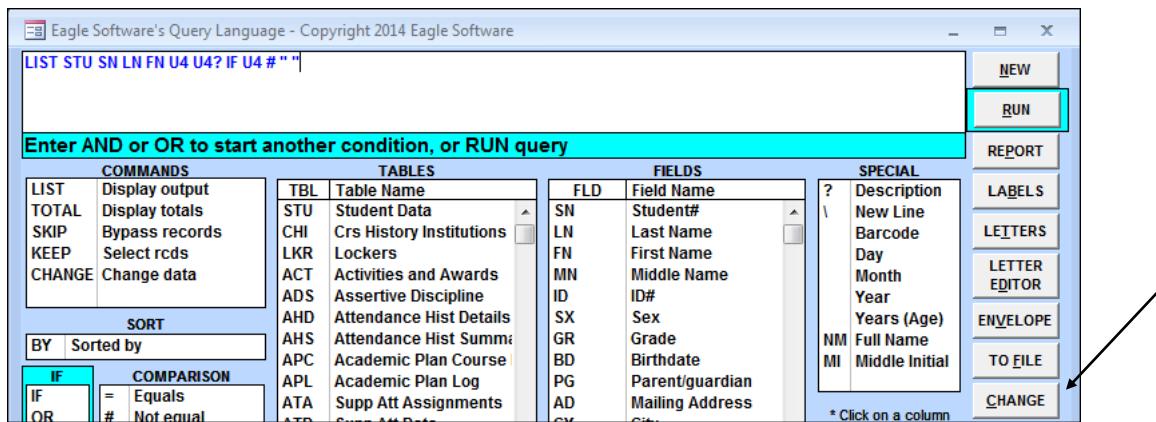
The default for the type of file is **Comma Delimited**. To change the type to tab delimited, click the mouse on the **Tab Delimited** button. To include the field name headings in the file, click the mouse on the **Yes** button. The default for the output path will be **A:**. To change the path type the drive letter and a colon, example **C:**.



In the **Name of output text file:**, type the text file name and **.TXT**. Click the mouse on the **Create** button. A message will display at the bottom of the form that the file is being created. Once the file has been created, a message will display. Click the mouse on the **OK** button.

CHANGE BUTTON

Major changes can be made to student data by utilizing the **CHANGE** button. Type a **LIST** query statement in the query text box and click the mouse on the **RUN** button. For example, the query below is searching for all students that have a code in the **U4** field. Return to the query form and click the mouse on the **CHANGE** button.



The following form will display with the fields selected in the query. Data can be changed within certain fields. For example, the Student Number, First and Last Name fields cannot be changed. Fields selected can also have **Tab** stops set to change the fields to display only.

The screenshot shows the "Query Change Form" window. The title bar reads "Query Change Form". The header section displays "2014-2015", "Screaming Eagle High School", and "8/13/2014". Below the header is a table with five columns: "Student#", "Last Name", "First Name", "User4", and "Description_U4". The table contains four rows of data:

Student#	Last Name	First Name	User4	Description_U4
1	Abbott	Allan	N	No Photo Release
2	Abdelnour	Alice	Y	Photo Approved
3	Abdo	Alice	Y	Photo Approved
4	Abdo	Arnold	Y	Photo Approved

Click the mouse on the field headings to change the **Tab** stops. The data will change to a darker blue. For example, the **First Name**, **Last Name** and **Description** fields will be used for verification so the **Tab** will not stop within these fields.

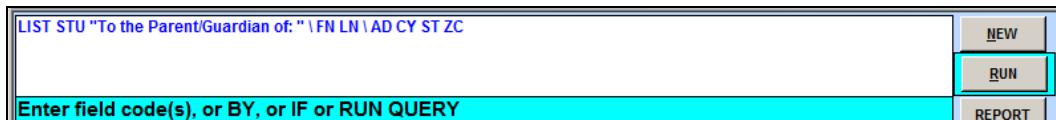
The screenshot shows the "Query Change Form" window again. The table data is identical to the previous screenshot, but the "User4" column for all rows is highlighted with a darker blue color, indicating active tab stops.

Press the **Tab** key. The cursor will only move up and down the **U4** field. Type the changes to the data and press the **Tab** key. After all changes are complete click the mouse on the **EXIT** button.

NOTE: Care MUST be taken when using this option. Data is changed instantly in the database.

SAVE QUERY STATEMENTS

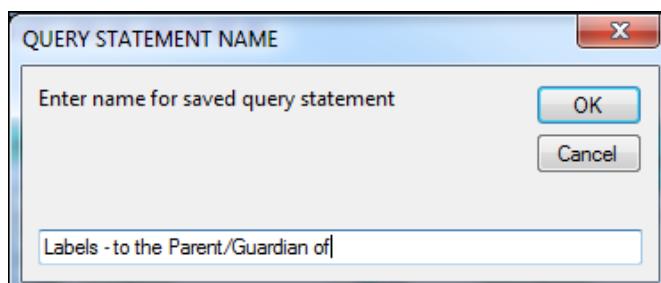
Once a query statement has been created and generated it can be saved and accessed for later use. It is suggested that only queries generated frequently and complicated queries be saved. Enter the **LIST** query and click the mouse on the **RUN** button. Close the query.



To save a query, click the mouse on the **SAVE** button.



The following text box will display. Type a descriptive name for the query in the text box. Click the **OK** button to save your **Query**. Remember - When a Report query is saved the query name will default to print on the heading of the report but can be changed.



USING THE LOAD BUTTON

To reload a saved query statement, click the **LOAD** button on the **QUERY** form. The following screen will display in AeriesCS. Saved queries can be displayed by the query name, author name, limited by query name, limit by author, or limited by table. Queries can also be viewed or imported from other school databases.

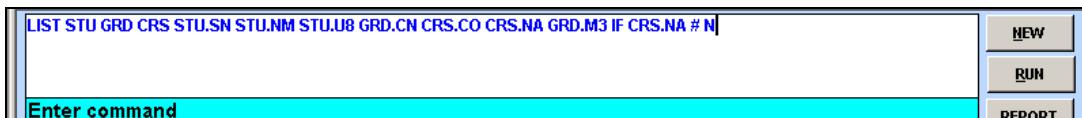
To select and run a **Query**, click the **Gray box** to the left of the **Query title**. A **black arrow** will display. Click the **OK** button at the bottom of the box.

Select Query to be Loaded		
<input type="checkbox"/> Limit by Query Name		<input type="checkbox"/> View / Import Query from Any School
<input type="checkbox"/> Limit by Author		<input type="checkbox"/> View / Import Query from 9-12 Schools
<input type="checkbox"/> Limit to Favorites		<input type="checkbox"/> Limit by School
Fav	Query Name	Author
	11th Grade Students	Demo
	LIST STU FN LN PG IF GR = 11	
▶	2nd period student list	Demo
	LIST STU SEC MST TCH CRS STU.LN STU.FN TCH.TE MST.RM MST.PD IF MST.PD = 2	
	9th Grade Student Enrollment	Demo
	LIST STU LN FN GR IF GR = 9	

To select the **Query** in Aeries.net and run it, click on the query record.

Sort By:	Query Name	Query Name Filter:	Author Filter:
Query Name	Author	Query	
11th Grade Students	Demo	LIST STU FN LN PG IF GR = 11	
2nd period student list	Demo	LIST STU SEC MST TCH CRS STU.LN STU.FN TCH.TE MST.RM MST.PD IF MST.PD = 2	
9th Grade Student Enrollment	Demo	LIST STU LN FN GR IF GR = 9	

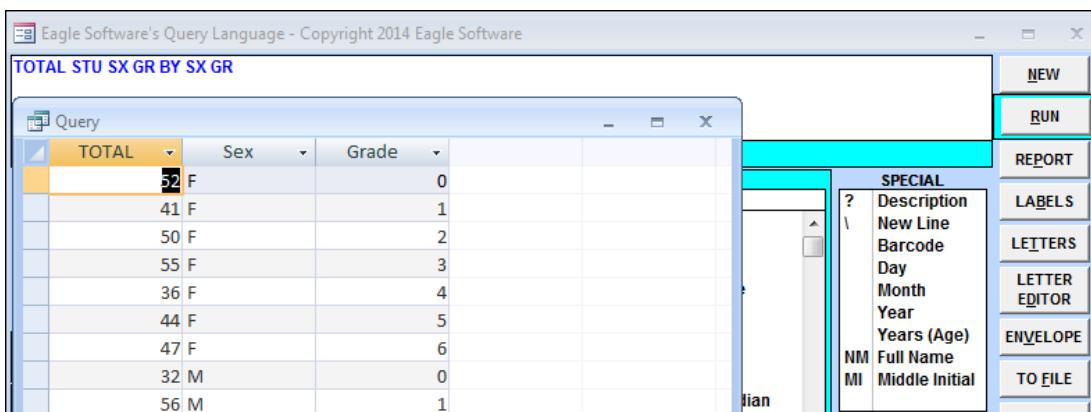
The query statement will display on the **Query** form. Take note the **RUN** button does not contain a border. Enter a **Space** after the query statement and the query will be loaded. Click the mouse on the **RUN** button.



TOTAL QUERY STATEMENT

The **TOTAL** command will perform a count of select data generated in a query statement. When running a **TOTAL** query the fields selected **MUST** be followed with a **BY** statement and **ALL** the fields selected. For example: **TOTAL STU SX GR BY SX GR**

This query will generate a count of the number of male/female students in each grade level.



TOTAL	Sex	Grade
52	F	0
41	F	1
50	F	2
55	F	3
36	F	4
44	F	5
47	F	6
32	M	0
56	M	1

Reports can also be generated for **TOTAL** queries using the **Report** button.

Golden Eagle Elementary School			8/13/2014
2014-2015 Total Male/Female Students by Grade			Page 1
TOTAL	Sex	Grade	
52	F	0	
41	F	1	
50	F	2	
55	F	3	
36	F	4	
44	F	5	
47	F	6	
32	M	0	
56	M	1	

USER SUPPLIED QUERIES

Queries have been posted out on our web site that have been created and sent to us by Aeries users. These queries can be located at the website <http://www.aeries.com/techsupport.asp> under **Queries**. These Queries can be hi-lighted, copied and pasted into Query. The query can then be generated and saved.

EXAMPLE QUERIES FOR SECONDARY SCHOOLS

Listing of code descriptions stored in the **COD** table for one field (EC) but used by another field (EC2)

LIST STU SN NM EC2? EC

Parents who have a different last name than the student

LIST STU NM GR PG IF PG ; LN

Search for an address with a certain street name

LIST STU NM PG AD IF AD : "(Type in the street name)"

CBEDS - to find Students who have a missing Ethnic Code

LIST STU SN LN FN EC GR SX BY EC

Search for a parent name different than a student's last name

LIST STU LN FN PG IF PG : "(Type the Last Name of the Parent)"

List a total number of students by Ethnic Code and by Sex

TOTAL STU EC SX BY EC SX

Class list by teacher and by sex

LIST STU TCH STU.SN STU.LN STU.FN STU.GR STU.SX TCH.TE BY TCH.TE STU.SX

Students excluded from mailings (**Example has student flagged with X in U4**)

LIST STU SN LN FN TL PG U4 IF U4 = "X"

Students with a health concern

LIST STU MED STU.SN STU.LN STU.FN MED.CD MED.CO IF MED.CD # " "

Class Directory by Teacher

LIST STU TCH STU.NM STU.AD STU.CY STU.ZC STU.TL STU.PG TCH.TE BY TCH.TE

Labels by grade and grid code

LIST STU FN LN \ AD \ CY ST ZC BY GR GC

Listing of gate students (**Example has student flagged with "X" in U7**)

LIST STU SN LN FN SX GR U7 IF U7 = G

Search for students with missing health immunizations

LIST STU IMM STU.SN STU.LN STU.FN STU.GR IF IMM.P1 = NULL

Listing of teachers and courses by teachers for a select period

LIST MST CRS TCH MST.TN TCH.TE MST.RM MST.SE CRS.CO BY TCH.TE IF MST.PD = 1

List all students in grade 9 -11 with math course info (Math courses range from 4501-4699 and the semester is for everything but Fall)

**LIST STU SEC MST CRS TCH STU.SN STU.LN STU.FN STU.GR CRS.CO TCH.TE MST.PD BY
STU.LN IF CRS.CN > 4500 AND CRS.CN < 4700 AND MST.SM # "F" AND STU.GR < 12**

EXAMPLE QUERIES FOR ELEMENTARY SCHOOLS

Listing of code descriptions stored in the **COD** table for one field (EC) but used by another field (EC2)
LIST STU SN NM EC2? EC

Parents who have a different last name than the student
LIST STU NM GR PG IF PG ; LN

Search for an address with a certain street name
LIST STU NM PG AD IF AD : "(Type in the street name you are searching for)"

CBEDS - to find Students who have a missing Ethnic Code
LIST STU SN LN FN EC GR SX BY EC

Search for a parent name different than a student's last name
LIST STU LN FN PG IF PG : "(Type the Parents Last Name you are looking for)"

List a total number of students by Ethnic Code and by Sex
TOTAL STU EC SX BY SX EC

Class list by teacher and by sex
LIST STU TCH STU.SN STU.LN STU.FN STU.GR STU.SX TCH.TE BY TCH.TE STU.SX

Students excluded from mailings (**Example has student flagged with X in U4**)
LIST STU SN LN FN TL PG U4 IF U4 = "X"

Students with a health concern
LIST STU MED STU.SN STU.LN STU.FN MED.CD MED.CO IF MED.CD # " "

Class Directory by Teacher
LIST STU TCH STU.NM STU.AD STU.CY STU.ZC STU.TL STU.PG TCH.TE BY TCH.TE

Labels by grade and grid code
LIST STU FN LN \ AD \ CY ST ZC BY GR GC

Listing of gate students (**Example has student flagged with "X" in U7**)
LIST STU SN LN FN SX GR U7 IF U7 = G

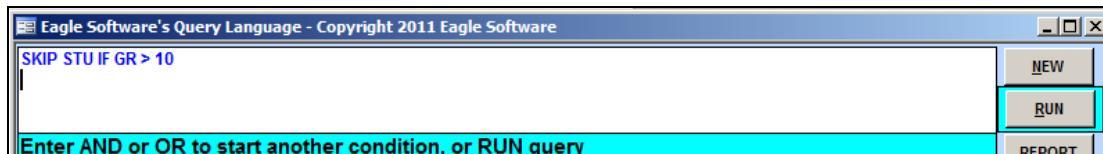
Search for students with missing health immunizations
LIST STU IMM STU.SN STU.LN STU.FN STU.GR IF IMM.P1 = NULL

USING SKIP OR KEEP COMMANDS

The **SKIP** and **KEEP** commands are used to create temporary tables that will only contain certain “groups” of data such as a particular grade. The **SKIP** can be used to “hold aside” students in grades 11 and 12. Queries can then be generated and the student data accessed will only be for students currently in grades less than grade 11. For example: **SKIP STU IF GR > 10**

USING SKIP COMMAND

Using the **SKIP** command, the query or student data will not access the students selected. In the query option type: **SKIP STU IF GR > 10** and click the mouse on the **RUN** button.



The following form will display. Click the mouse on the scroll bar and hold the button down. Drag the **scroll bar** to the bottom of the form. Only students that are less than grade 11 display.

Student#	Last Name	First Name	Middle Name	ID#	Sex	Grade	Birth
54	Almodovar	Armando	Timothy	99400054	M	8	1
55	Alonzo	Clarisa	Berdine	99400055	F	10	
58	Alvarado	Jocelyn		99400058	F	9	6
59	Alvarado	Nicholas	Steven	99400059	M	8	
60	Alvarado	William		99400060	M	10	7
63	Alyunan	Tracy		99400063	F	10	
64	Amador	Isaac	Thomas	99400064	M	8	1

To exit and close this screen click the bottom “X” in the upper right-hand corner. The **QUERY** form will now display. Notice that the **SKIP** statement entered displays.



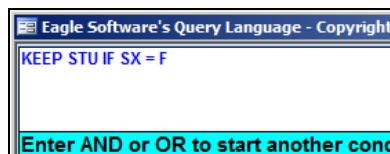
In the middle of the form “**Records Are Being Skipped**” will display. Any further queries or student data will skip all grades greater than grade 10. To give an example of how a **SKIP** statement works, type **LIST STU SN LN FN GR BY GR**. Click the mouse on the **RUN** button. Click the mouse on the scroll bar and drag the **scroll bar** to the bottom of the form.

Notice that the last records displayed will only display up to grade 10. To exit and close this screen, click the “X” in the upper right-hand corner.

Student#	Last Name	First Name	Grade
2728	Williams	Amber	10
2732	Williams	Ly	10
2736	Wilson	Kimberly	10
2748	Woods	Erin	10

USING KEEP

The **KEEP** command will give you access to only the students selected. In the query option type **KEEP STU IF SX = "F"**. Click the **RUN** button.



Click the mouse on the **scroll bar** and drag to the bottom of the form. The only students displayed are female. Close the screen by clicking the bottom "X" in the upper right-hand corner.

Student#	Last Name	First Name	Middle Nam	ID#	Sex	Grade
2774	Zapata	Jennie	S	99402774	F	
2775	Zarate	Linda		99402775	F	
2777	Zarate	Stefanie	S	99402777	F	
2778	Zavaleta	Dolores		99402778	F	

The **QUERY** form will now display. The **KEEP** statement will display. In the middle of the form "**Records Are Being Skipped**" will display. Any further queries or student data will only access female students.



ACCESSING DATA USING KEEP

To give you an example of how a **KEEP** statement works accessing data, click the mouse on the **EXIT** button. The **Control Panel** will display. The message, **RECORDS ARE BEING SKIPPED** will display at the bottom of the form. In Aeries.net the message will display above the navigation menu.



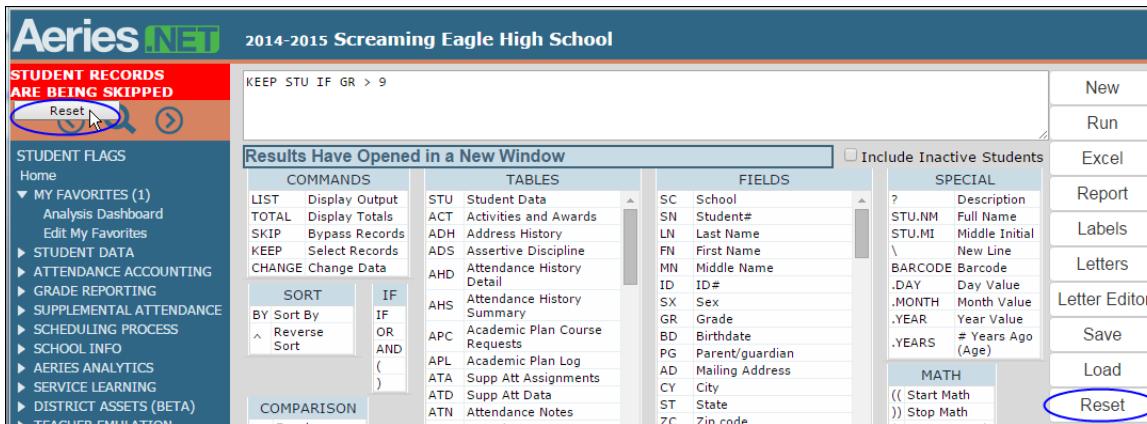
Click the mouse on **Student Data**. The **Student Data** form will display. In the middle of the form, **RECORDS BEING SKIPPED** will display. Click the mouse on the **Forward** button. Only female students will display.

A screenshot of the Aeries.net Student Data form for the year 2014-2015 at Screaming Eagle High School. The date 8/13/2014 is shown in the top right. The form has tabs for 'Student Data 1 (STU)', 'Student Data 2 (STU)', 'Student Data 3 (STU)', 'Secondary Student Data (SSD)', and 'Student Documents (DOC)'. The 'Student Data 1 (STU)' tab is active. It contains fields for Student ID (000012), Last Name (Abrego), First Name (Alice), Middle Name (A), Suffix (None), Perm ID No. (099400012), Sex (F), Grade (9), Birthdate (12/01/2000), and Age (13). There are also sections for Parent/Guardian, Area/Telephone, Primary Contact 1, Primary Contact 2, Residence and Mailing Address, GridCd, Interdist xfer / District, Schi Enter Dt, Dist Enter Dt, and a photo of the student. At the bottom of the form, a red banner displays the message 'RECORDS ARE BEING SKIPPED'. There are also checkboxes for 'Check Status', 'Show Inactives', and 'Show Adjust'.

RESET SKIP OR KEEP

When complete with running queries and/or reports that include a SKIP or KEEP, you must RESET the query by clicking the **RESET** button in the lower right corner of the form. “Records Are Being Skipped” will no longer display and further queries or accessing data will include all students. If reset is not performed the only data accessible will be the query entered for the SKIP or KEEP command.

In Aeries.net the message indicating that records are being skipped is displayed above the navigation menu. Hovering over the message displays a **RESET** button that can be used to clear out the limitation on the records. The **RESET** button in the Query functions can also be used.



UNDO LAST KEEP/SKIP

When utilizing the **KEEP/SKIP** function multiple **KEEP/SKIP** query statements can be generated. For example, listed below there are two **KEEP/SKIP** query statements that have been generated.



When a query is generated the first **KEEP** statement will locate only female students and the second **SKIP** statement will locate students (female only) with a grade less than 11. There is an **Undo Last KEEP/SKIP** button to the right of the statement box that will remove the last **KEEP/SKIP** statement entered.

Click the mouse on the **Undo Last KEEP/SKIP** button and the last statement generated will be removed. Now when a query is generated the first **KEEP** statement will locate only female students.

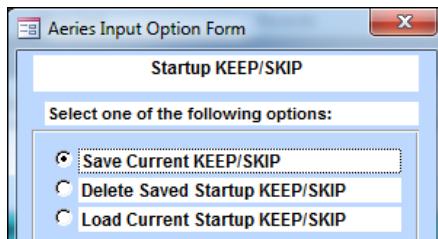


USING STARTUP

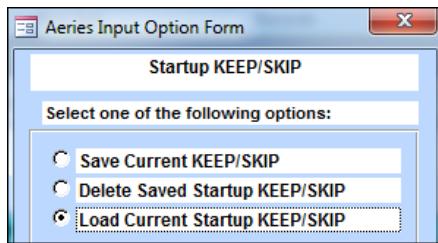
The **STARTUP** function is used to save a **KEEP/SKIP** statement that is user specific. When the user signs into Aeries the **KEEP/SKIP** statement will automatically be generated. For example, counselors may only want to look at students assigned to them and not all students.

To **SAVE** a current **KEEP/SKIP** statement for **Startup**, enter a **KEEP/SKIP** statement and click the mouse on the **RUN** button. The **KEEP/SKIP** statement will display in the statement box.

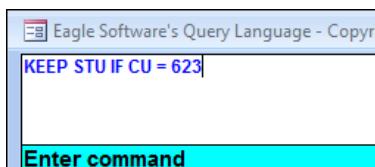
Click the mouse on the **STARTUP** button. The following selection box will display. Select the **SAVE Current KEEP/SKIP** option and click the mouse on the **OK** button.



To **LOAD** the **KEEP/SKIP** that was saved, click the mouse on the **STARTUP** button. Select **LOAD Current KEEP/SKIP** and click the mouse on the **OK** button.



The saved query statement will display. Click the mouse on the **RUN** button and the **KEEP/SKIP** statement previously saved will be generated. Exit the query form.



The **Aeries Control Panel** will display **Records are Being SKIPPED**. Exit Aeries and sign back in. The **KEEP/SKIP** statement will automatically be generated. The only data that can be accessed will be in regards to the **KEEP/SKIP** statement loaded.

For example the statement above is **KEEP STU IF CU = 623**. The only students that will display on the **Student Data** form will be the students that have been assigned to counselor 623.



To view **ALL** students click the mouse on the **RESET** button. The button will now display **Apply Startup KEEP/SKIP** and all student information can be viewed. To re-load the Startup query click the mouse on the **Apply Startup KEEP/SKIP** button and the query will be generated.



To remove the **Startup KEEP/SKIP** statement, click the mouse on the **STARTUP** button. Click the mouse on the **Delete Saved Startup KEEP/SKIP** button. Click the mouse on the **OK** button and the **Startup KEEP/SKIP** will be deleted.