

Import Tools

The Import Tools module provides a set of tools designed to simplify the process of moving data from a school's previous SIS into Focus SIS.

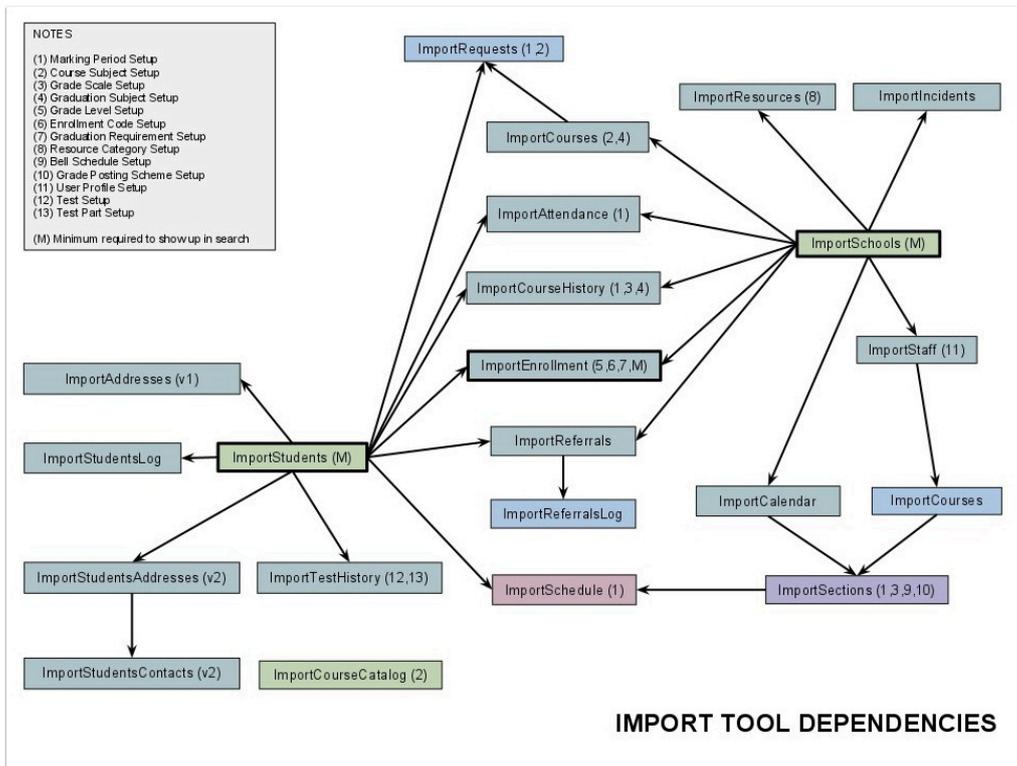
Introduction

Understanding Database Relations

Before importing data into Focus, it is important to understand how the relationships between the different import tools will affect the order in which the import tools are run. For example, Focus High School would like to import their Student Attendance information. Focus High School has not yet imported their Student Demographic information. How can a student's attendance be recorded if there is no information that the student exists? The answer is simply that the Student Demographic Information must be imported before Focus High School can import their Student Attendance data.

The chart, displayed below, illustrates the relationships between the import tools. When an arrow leads to an import tool, such as Import Attendance, the arrow illustrates that the tool that the arrow is drawn from, such as Import Students, must be completed first. In the case of Import Attendance, a Student Import must be completed as well as a School Import.

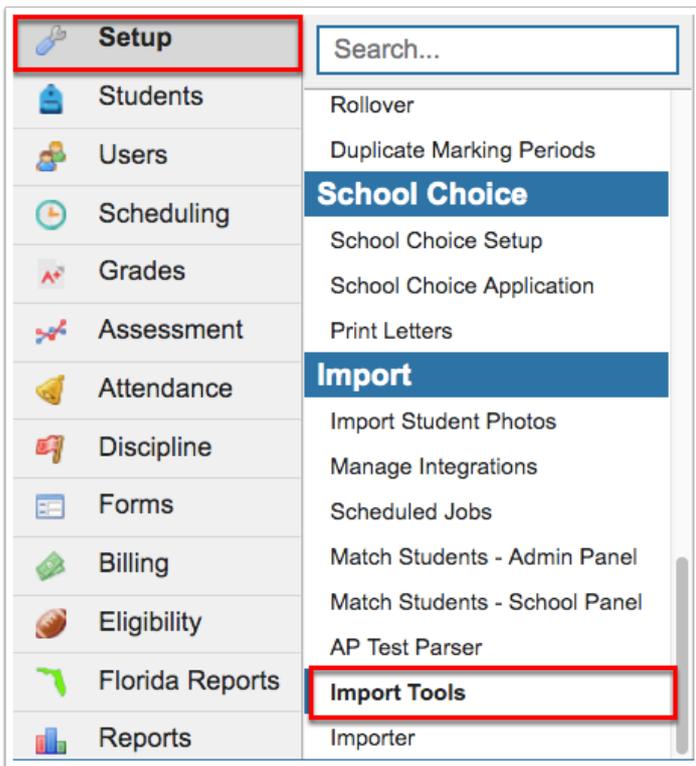
In the top left section of the Import Tools Dependencies chart, notes are provided for the corresponding parenthetical codes listed after the import tools' title. In the Import Attendance example, the notes indicate marking periods must be setup before this tool can be used.



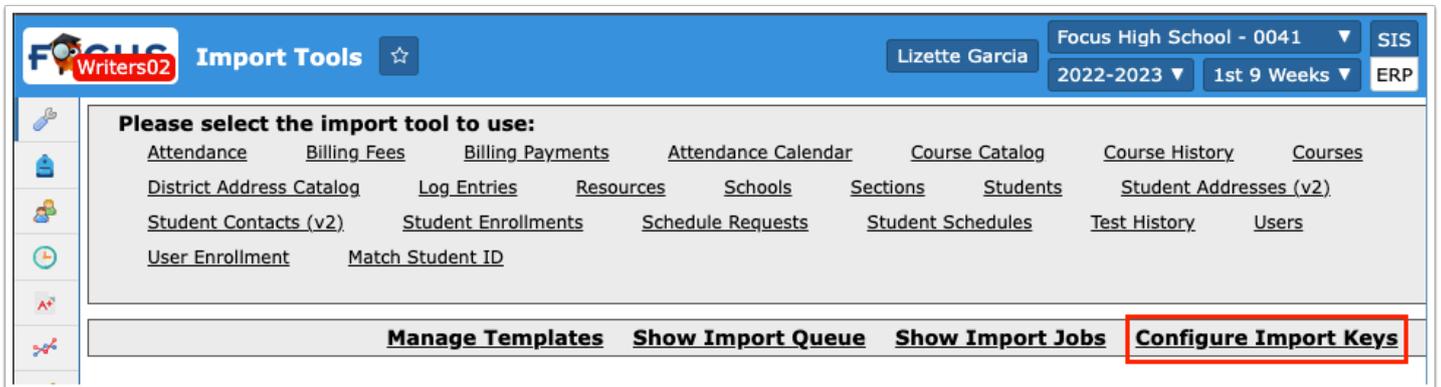
Configure Import Keys

The Configure Import Keys page allows the import tool user to specify which field in the Focus database table should be used for mapping data related to that table.

1. In the **Setup** menu, click **Import Tools**.



2. From the Import Tools screen, click **Configure Import Keys**.



When migrating data from another student information system, each key should be set to the primary key used by the legacy system.

Import Key Setup

Please assign the field that contains the primary key from the legacy system.

Import Tool	Legacy Key Field
Student Addresses	Address
Course Catalog	Short name
Course Subjects	Short name
District Address Catalog	Choice pool
Resources	Short name
School Periods	Short name
Schools (static)	School Number (CUSTOM_327)
Student Logs	Completed at
Users Logs	Completed at
Students	TERMS ID (CUSTOM_53)
Users	Custom 200000001 (CUSTOM_200000001)
User Enrollment	Custom 200000001 (CUSTOM_200000001)

Reset Keys to Focus defaults.

For example, school ID is typically defined as CUSTOM_327 in the Schools table. This allows you to specify your own local school number as the key value to be used when importing data into tables that connect to the school's table (such as, enrollment, discipline, course history), instead of using the Focus assigned school ID.

Each table key can be changed by selecting the pull-down from any of the import tools available on the Configure Import Keys screen.

Fields & Parameters

Each of the Focus import tools has a set of required fields, static fields, and runtime parameters.

A **required field** is a field that must be mapped for the import to successfully run. Additional fields may be flagged as required through the user interface, and if this is the case, those fields must be included in the import file and mapped to their corresponding field in Focus.

Static fields are fields that default to a fixed value when not provided in the mapping template. If the field is provided in the mapping template, the value will become whatever the import file has stored for each of the records.

Runtime parameters are options that provide the import tools with specific information about the data being imported. For example, a common runtime parameter is date format. This parameter will tell the import tool how the date fields are being formatted in the import file so that the tool can properly modify the fields to be inserted into Focus. It is important to make the proper selection for each of the runtime parameters to make a successful import.

Below is the import tool upload form for Course History. From this screen, a number of different runtime parameters can be set including calculate GPA, base class rank on weighted or unweighted GPA, and weight GPA by credits. Each tool will have a variety of different runtime parameters.

Focus Writers02 Import Tools Lizette Garcia Focus High School - 0041 2022-2023 1st 9 Weeks SIS ERP

Please select the import tool to use:

Attendance Billing Fees Billing Payments Attendance Calendar Course Catalog **Course History** Courses
 District Address Catalog Log Entries Resources Schools Sections Students Student Addresses (v2)
 Student Contacts (v2) Student Enrollments Schedule Requests Student Schedules Test History Users User Enrollment
 Match Student ID

Manage Templates Show Import Queue Show Import Jobs Configure Import Keys

Please select upload file options

Note: The file uploaded for this import should be sorted by School. [Other Options](#)

Source file Choose File No file chosen

File extension Tab-Delimited CSV Fixed-Width

File contains header information Yes No

Import methods Update or Insert

Date column options Import Date from 1 column

Custom year format YYYY

Custom date format YYYYMMDD

Upload File [Export Import Templates](#)

Runtime Parameters

Focus Import Tools use a variety of setup parameters to customize how a particular process is run. Below is a list of available options.

Base Class Rank On

Base class rank on

Weighted GPA Unweighted GPA

Possible Values: Weighted GPA (default), Unweighted GPA
Applies To: Course History
HTTP Parameter: rank=(WEIGHTED_GPA | GPA)
Command Line Parameter: -k (WEIGHTED_GPA | GPA)

As Of SVN Revision: 14775

Tells the import tool to rank students according to whether or not the GPA is weighted.

Calculate GPA

Calculate GPA <input type="radio"/> Yes <input checked="" type="radio"/> No	Possible Values: Yes, No (default) Applies To: Course History HTTP Parameter: calc_gpa=(yes no) Command Line Parameter: -g (yes no)
---	--

As Of SVN Revision: 14775

Tells the import tool to perform a GPA calculation on affected records after the import is completed.

Contact Name Options

Contact Name options <input type="text" value="Multiple columns"/>	Possible Values: Mutiple Columns (default), Single Column (last/middle/first), Single Column (first/middle/last) Applies To: Student Addresses (v1), Student Contacts (v2) HTTP Parameter: contact_name_type=([blank] f/m/l l/m/f) Command Line Parameter: -n ([blank] f/m/l l/m/f)
--	--

As Of SVN Revision: 10512

By default, contact names are provided by the import file in separate columns, split by first, middle, and last name. When one of the single column options is selected, the contact name is parsed according to the order selected.

Custom Date Format

Please select upload file options

Source file
 No file chosen

File extension
 Tab-Delimited CSV Fixed-Width

File contains header information
 Yes No [Other Options](#)

Import methods
 ▼

Date column options
 ▼

Custom year format
 ▼

Custom date format
 ▼

[Export Import Templates](#)

Possible Values: Standard Format (default), YYYYMMDD, YYMMDD, DDMMYYYY, MMDDYYYY, MMDDYY

Applies To: All Tools

HTTP Parameter: date_format=([blank] | yyyyymmdd | yymmdd | ddmmyyyy | mmddyyyy)

Command Line Parameter: -o ([blank] | yyyyymmdd | yymmdd | ddmmyyyy | mmddyyyy)

As Of SVN Revision: 10512

Determines how date columns are interpreted. The standard format is simply the usual database format, which is YYYY-MM-DD. All others are according to their size and order, using YYYY as a four-digit year, YY as a two-digit year with 19 (> 50) or 20 (< 50) implied. MM is a two-digit month, and DD is a two-digit year. Both MM and DD are zero-filled.

Custom Year Format

Please select upload file options

Source file
 No file chosen

File extension
 Tab-Delimited CSV Fixed-Width

File contains header information
 Yes No [Other Options](#)

Import methods
 ▼

Date column options
 ▼

Custom year format
 ▼

Custom date format
 ▼

[Export Import Templates](#)

Possible Values: Start Syear (default), End Syear, Period
Applies To: Attendance Calendar
HTTP Parameter: syear_type=(one | two | three)
Command Line Parameter: -s (one | two | three)

As Of SVN Revision: 12882

Determines how SYEAR fields are interpreted. For Start Syear, the SYEAR value is the year that the school year begins in. For End Syear, the SYEAR value is the year that the school year ends. For Period, the SYEAR value is the year that the marking period belongs to.

Date Column Options

Please select upload file options

Source file
 No file chosen

File extension
 Tab-Delimited CSV Fixed-Width

File contains header information
 Yes No [Other Options](#)

Import methods
 ▼

Date column options
 ▼

Custom year format
 ▼

Custom date format
 ▼

[Export Import Templates](#)

Possible Values: Import Date From 1 Column (default), Import Date From 3 Columns
Applies To: All Tools
HTTP Parameter: date_type=(one | three)
Command Line Parameter: -a (one | three)

As Of SVN Revision: 10512

When set to the default value, any dates are derived from a single column. However, when Import Date From 3 Columns is selected, the date is compiled from three separate columns representing month, day and year.

File Contains Header Info

Please select upload file options

Source file
 No file chosen

File extension
 Tab-Delimited CSV Fixed-Width

File contains header information
 Yes No [Other Options](#)

Import methods

Date column options

Custom year format

Custom date format

[Export Import Templates](#)

Possible Values: Yes, No (default)
Applies To: All Tools
HTTP Parameter: file_header=(yes | no)
Command Line Parameter: -d (yes | no)

As Of SVN Revision: 10512

Tells the import tool whether or not the first row of the import file contains column headers. This can become useful during the mapping phase of the import process, but otherwise any header row is ignored by the import tools.

A NOTE ABOUT HEADERS: In some file types, particularly CSV files, a difference between the number of columns in the header row and the number of columns in all other rows, can cause formatting issues on the mapping page. This tends to happen when all of the columns are defined in the header, but blank columns are left out of the data rows. A workaround solution to this is to either use a tab-delimited format, to include commas as placeholders at the end of each record to make the column counts the same, or to include a filler column at the end of

each row, to force the intermediate columns to be generated. The last option works well in files coming from Microsoft Excel, Open Office Calc, or a similar spreadsheet program.

File Extension

Please select upload file options

Source file
Choose File No file chosen

File extension
 Tab-Delimited CSV Fixed-Width

File contains header information
 Yes No [Other Options](#)

Import methods
Update or Insert

Date column options
Import Date from 1 column

Custom year format
YYYY

Custom date format
YYYYMMDD

Upload File [Export Import Templates](#)

Possible Values: CSV (default), Fixed Width, Tab Delimited
Applies To: All Tools
HTTP Parameter: file_type=(csv | txt | tab)
Command Line Parameter: -y (csv | txt | tab)

As Of SVN Revision: 10512

Determines the type and format of the import file. CSV files have each field separated by a comma, with optional double-quotes surrounding string values, and generally have the file extension of .CSV. Tab delimited files are similar, but use the table character to separate fields. These files will often have the file extension of .TXT or .TAB. Finally, fixed width files are an older format, where each field starts at a predetermined character column position. These files will usually have the .TXT extension.

Import Methods

Please select upload file options

Source file
Choose File No file chosen

File extension
 Tab-Delimited CSV Fixed-Width

File contains header information
 Yes No [Other Options](#)

Import methods
✓ Update or Insert
Update Only
Delete and Insert
Insert Only

Date column options
Import Date from 1 column

Custom year format
YYYY

Custom date format
YYYYMMDD

Upload File [Export Import Templates](#)

Insert Only: Import rows are checked against existing records in the database, and are inserted if they don't exist. If they do exist, they are discarded. For example, if you are putting a student into the database for the first time, they will be entered; however, if the student already exists in the database nothing will change.

Insert All: Import rows are inserted into the database, without regard to existing records. This option may cause SQL errors in the event that non-unique database keys are set up on fields other than the primary ID field.

Update or Insert: Import rows are checked against existing records in the database, and are inserted if they don't exist, or updated if they do.

Update Only: Import rows are checked against existing records in the database, and updated if they already exist. If they don't exist, they are discarded.

Delete and Insert: Import rows are checked against existing records in the database, and are inserted if they don't exist. If they do exist, they are deleted from the database then inserted. This import method is different from the update methods because it will not preserve any data from the original record, even if the new record does not contain a field that the old record does.

Synchronize: Certain import tools have the ability to try and synchronize data, as opposed to standard inserts or updates. Synchronize operations check a set of predefined, tool-specific key fields to determine whether or not prior records exist. Based on those criteria, it will insert, update or replace certain records. Each of those key sets can be separated into two components: the legacy key, and the sync criteria. Once those are determined, one of the following actions is taken:

1. If the legacy key and the sync criteria match, then the record is updated.
2. If the legacy key does not match, the record is updated.
3. If the legacy key matches, but the sync criterion doesn't, any record that matches the legacy key is deleted from the system, and the new record is inserted.

Possible Values: Update or Insert, Insert Only, Update Only, Insert All, Delete and Insert, Synchronize

Applies To: All Tools (certain methods only)

HTTP Parameter: import_method=(`[blank]` | insert | update | insert_all | delete_insert | sync)

Command Line Parameter: -e (`[blank]` | insert | update | insert_all | delete_insert | sync)

As Of SVN Revision: 12028 (Update or Insert, Insert Only, Delete and Insert, Update Only)

As Of SVN Revision: 12733 (Synchronize)

As Of SVN Revision: 13905 (Insert All)

Import Test Parts By

Import Test Parts by

Possible Values: Short Name (default), Subject Code
Applies To: Test History (All Tests selected)
HTTP Parameter: part_id=(SHORT_NAME | SUBJECT_CODE)
Command Line Parameter: -i (SHORT_NAME | SUBJECT_CODE)

As Of SVN Revision: 11994

Tells the import script where to fetch the key for the test part from.

Limit Row Count

Custom year format
Custom date format
Enter Test Administration Date?
Enter Test School Year?
Include In Transcript? No Yes
Testing Mode No Yes
Output SQL To Text No Yes
Timestamp output files Yes No
Import Test Parts by
Memory cap (in megabytes)
0 = no memory limit
Statistics frequency
Also affects transaction size
Limit row count
0 = process all rows
 [Export Import Templates](#)

Possible Values: Any integer; a 0 or any number greater than the number of rows in the file will import all rows.
Applies To: All Tools
HTTP Parameter: limit_rows={x}
Command Line Parameter: -x {x}

As Of SVN Revision: 15574

Limits import file processing to the first x rows in the file. A value of 0 (default) processes all rows normally.

Memory Cap

Custom year format YYYY	Custom date format YYYYMMDD	Enter Test Administration Date? <input type="checkbox"/>	Enter Test School Year? <input type="checkbox"/>	Include In Transcript? <input checked="" type="radio"/> No <input type="radio"/> Yes
Timestamp output files <input checked="" type="radio"/> Yes <input type="radio"/> No	Import Test Parts by Short Name	Memory cap (in megabytes) 0 0 = no memory limit		Output SQL To Text <input checked="" type="radio"/> No <input type="radio"/> Yes
Limit row count 0 0 = process all rows				Statistics frequency 1000 rows Also affects transaction size
Upload File				Export Import Templates

Possible Values: Any integer, up to the memory capacity of the server

Applies To: All Tools

HTTP Parameter: memory_cap={x}

Command Line Parameter: -b {x}

As Of SVN Revision: 14439

When memory usage reaches the number assigned in memory cap, all pending queries are committed to the database, memory structures are cleared out, and the import process resumes from the next record in the import. A value of 0 in this field disables the memory cap, and the process will use all available memory.

Primary Key Options

	Date column options Import Date from 1 column	Custom date format YYYYMMDD	// Other options
Primary key options ✓ Custom Primary Key SIS Primary Key	Testing Mode <input checked="" type="radio"/> No <input type="radio"/> Yes	Output SQL To Text <input checked="" type="radio"/> No <input type="radio"/> Yes	Send Unmatched Students To Schools <input checked="" type="radio"/> No <input type="radio"/> Yes
Limit row count 0 0 = process all rows	Timestamp output files <input checked="" type="radio"/> Yes <input type="radio"/> No	Memory cap (in megabytes) 0 0 = no memory limit	Statistics frequency 1000 rows Also affects transaction size
Upload File			Export Import Templates

Possible Values: SIS Primary Key, Custom Primary Key
Applies To: All Tools
HTTP Parameter: pk_type=(sis | custom)
Command Line Parameter: -p (sis | custom)

As Of SVN Revision: 10512

Determines how the primary key for the Focus records is assigned. For SIS Primary Key, the local ID from the import file is converted to an integer value. That value is assigned to the primary ID field in Focus. The original local ID is stored in a special field within the database table. For Custom Primary Key, the local ID is stored in a special field within the database table, and the Focus primary ID is auto-generated from a database sequence object.

Note: This field doesn't show up for all tools, but is still active. In tools that a choice is not offered, the default is always Custom Primary Key.

Section Day Column Options

Sections Day column options

Import Days from 1 column

Possible Values: Import Days From 1 Column (default), Import Days From 7 Columns
Applies To: Sections, Student Schedules
HTTP Parameter: section_day_type ([blank] | seven)
Command Line Parameter: -c ([blank] | seven)

As Of SVN Revision: 10512

By default, section day data is stored in one composite column. When set to Import Days From 7 Columns, each day value is stored in its own column in the import file.

Source File

Please select upload file options

Note: The file uploaded for this import should be sorted by School.

Source file
Choose File No file chosen

File extension
 Tab-Delimited CSV Fixed-Width

File contains header information
 Yes No

Import methods
Update or Insert

Date column options
Import Date from 1 column

Custom year format
YYYY

Custom date format
YYYYMMDD

Testing Mode
 No Yes

Output SQL To Text
 No Yes

[Other Options](#)
// Other options

Possible Values: A file from the local computer, or one that resides on the server.

Applies To: All Tools

HTTP Parameter: read_file={x}

Command Line Parameter: -r {x}

As Of SVN Revision: 10512

This can either be a file selected from the local machine, or a file that resides on the server. For server-based files, they will be located in one of the following directories:

- Attendance: uploaded-assets/export/attendance
- Attendance Calendar: uploaded-assets/export/calendar
- Course Catalog: uploaded-assets/export/catalog
- Course History: uploaded-assets/export/course_history
- Courses: uploaded-assets/export/courses
- Discipline Incidents: uploaded-assets/export/discipline_incidents
- Discipline Referrals: uploaded-assets/export/referrals
- District Address Catalog: uploaded-assets/export/geodata
- Log Entries (Referrals): uploaded-assets/export/referral_log_entry
- Log Entries (Students): uploaded-assets/export/student_log_entry
- Match Student ID Tool: uploaded-assets/export/matches
- Resources: uploaded-assets/export/resources
- Schools: uploaded-assets/export/schools
- Sections: uploaded-assets/export/courses
- Students: uploaded-assets/export/students
- Student Addresses: uploaded-assets/export/addresses
- Student Enrollments: uploaded-assets/export/enrollment
- Schedule Requests: uploaded-assets/export/requests
- Student Schedules: uploaded-assets/export/schedule
- Test History: uploaded-assets/export/test_history
- Users: uploaded-assets/export/staff

Testing Mode

Please select upload file options

Note: The file uploaded for this import should be sorted by School.

Source file <input type="button" value="Choose File"/> No file chosen	File extension <input checked="" type="radio"/> Tab-Delimited <input type="radio"/> CSV <input type="radio"/> Fixed-Width	File contains header information <input type="radio"/> Yes <input checked="" type="radio"/> No
Custom year format YYYY ▾	Import methods Update or Insert ▾	Date column options Import Date from 1 column ▾
Custom date format YYYYMMDD ▾	Testing Mode <input checked="" type="radio"/> No <input type="radio"/> Yes	Output SQL To Text <input checked="" type="radio"/> No <input type="radio"/> Yes

[Other Options](#)
// Other options

Possible Values: Yes, No (default)
Applies To: All Tools
HTTP Parameter: test_mode=(yes | no)
Command Line Parameter: -m (yes | no)

As Of SVN Revision: 12838

When testing mode is enabled, all database commands that update, insert or delete records from the system will be suppressed. Please note that this may affect certain processes in some tools that depend on DML statements being executed successfully.

Time Stamp Output Files

Source file <input type="button" value="Choose File"/> No file chosen	File extension <input checked="" type="radio"/> Tab-Delimited <input type="radio"/> CSV <input type="radio"/> Fixed-Width	File contains header information <input type="radio"/> Yes <input checked="" type="radio"/> No
	Import methods Update or Insert ▼	Date column options Import Date from 1 column ▼
Custom year format YYYY ▼	Custom date format YYYYMMDD ▼	Testing Mode <input checked="" type="radio"/> No <input type="radio"/> Yes
	Timestamp output files <input checked="" type="radio"/> Yes <input type="radio"/> No	Output SQL To Text <input checked="" type="radio"/> No <input type="radio"/> Yes
		Memory cap (in megabytes) 0 0 = no memory limit

[Other Options](#)
// Other options

Possible Values: Yes, No (default)
Applies To: All Tools
HTTP Parameter: stamp_log=(yes | no)
Command Line Parameter: -f (yes | no)

As Of SVN Revision: 14855

Output files are normally named according to the import file (exceptions) or the database table (finish and log) that they operate on. When this option is active, an additional date/time stamp is added to the file names to make them unique.

Weight GPA by Credits

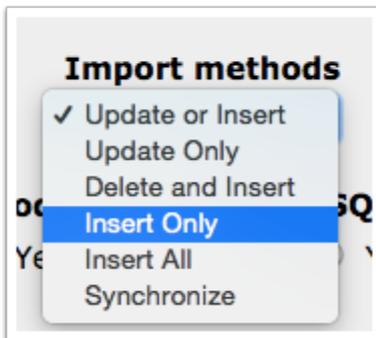
Weight GPA by Credits <input checked="" type="radio"/> Yes <input type="radio"/> No	Possible Values: Yes (default), No Applies To: Course History HTTP Parameter: weight_by_credits=(yes no) Command Line Parameter: -w (yes no)
---	---

As Of SVN Revision: 14775

Tells the import tool whether or not to weight the students GPA by credits.

Import Methods

Each Focus import tool allows for one or more different import methods. Each method performs the data import in a different way.



Insert Only: Import rows are checked against existing records in the database, and are inserted if they don't exist. If they do exist, they are discarded. For example, if you are putting a student into the database for the first time, they will be entered; however, if the student already exists in the database nothing will change.

Insert All: Import rows are inserted into the database, without regard to existing records. This option may cause SQL errors in the event that non-unique database keys are set up on fields other than the primary ID field.

Update or Insert: Import rows are checked against existing records in the database, and are inserted if they don't exist, or updated if they do.

Update Only: Import rows are checked against existing records in the database, and updated if they already exist. If they don't exist, they are discarded.

Delete and Insert: Import rows are checked against existing records in the database, and are inserted if they don't exist. If they do exist, they are deleted from the database then inserted. This import method is different from the update methods because it will not preserve any data from the original record, even if the new record does not contain a field that the old record does.

Synchronize: Certain import tools have the ability to try and synchronize data, as opposed to standard inserts or updates. Synchronize operations check a set of predefined, tool-specific key fields to determine whether or not prior records exist. Based on those criteria, it will insert, update or replace certain records. Each of those key sets can be separated into two components: the legacy key, and the sync criteria. Once those are determined, one of the following actions is taken:

1. If the legacy key and the sync criteria match, then the record is updated.
2. If the legacy key does not match, the record is updated.
3. If the legacy key matches, but the sync criterion doesn't, any record that matches the legacy key is deleted from the system, and the new record is inserted.

Selecting the Import File

When selecting a file to import, there are two options for the user to choose from.

1. The source file can be selected and uploaded from the user's local machine.
2. The file can be selected from the server. Files stored on the server are kept in directories of uploaded-assets as defined by the import tool. See below:
 - Attendance: uploaded-assets/export/attendance
 - Attendance Calendar: uploaded-assets/export/calendar
 - Course Catalog: uploaded-assets/export/catalog
 - Course History: uploaded-assets/export/course_history
 - Courses: uploaded-assets/export/courses
 - Discipline Incidents: uploaded-assets/export/discipline_incidents
 - Discipline Referrals: uploaded-assets/export/referrals
 - District Address Catalog: uploaded-assets/export/geodata
 - Log Entries (Referrals): uploaded-assets/export/referral_log_entry
 - Log Entries (Students): uploaded-assets/export/student_log_entry
 - Match Students: uploaded-assets/export/matches
 - Resources: uploaded-assets/export/resources
 - Schools: uploaded-assets/export/schools
 - Sections: uploaded-assets/export/courses
 - Students: uploaded-assets/export/students
 - Student Addresses: uploaded-assets/export/addresses
 - Student Enrollments: uploaded-assets/export/enrollment
 - Schedule Requests: uploaded-assets/export/requests
 - Student Schedules: uploaded-assets/export/schedule
 - Test History: uploaded-assets/export/test_history
 - Users: uploaded-assets/export/staff

with a blank screen following your import, do not be alarmed. The import is still happening in Focus.

If the file being imported does not time out the browser, the next screen will provide you with the option of downloading the output files.

Marking Periods

For importing any file with marking periods, the import tool will automatically translate the following marking periods. The column on the left is the value in the import file, and the column on the right being the short name setup in Focus.

1 => 'S1'

2 => 'S2'

3 => 'FY'

4 => 'SS1'

5 => 'SS2'

6 => 'Q1'

7 => 'Q2'

8 => 'Q3'

9 => 'Q4'

'B' => 'T1'

'C' => 'T2'

'D' => 'T3'

'E' => 'QM1'

'F' => 'QM2'

'G' => 'QM3'

'H' => 'QM4'

'I' => 'QM5'

'J' => 'SW1'

'K' => 'SW2'

'L' => 'SW3'

'M' => 'SW4'

'N' => 'SW5'

'O' => 'SW6'

'S' => 'SS'

The Course History import can be used to import semester exam grades. The marking period short name should have a leading E (without quotes) to flag the import tools that the grade is for a semester exam.

The Course History import tool will automatically create marking periods for previous years of course history based on the marking periods setup in the current school year for each school. So the only marking periods that must be setup are all marking periods that match the short names in the course history import files for each school in the current school year.

Breakdown and Tips

Attendance

[Import Methods](#): Delete and Insert, Insert Only

Required Fields: Student ID, School Date

Static Fields: School Year (selected year), School ID (selected school)

Log File Name: ATTENDANCE_DAY.log

Local ID: STUDENT_ID, SCHOOL_DATE

IMPORTANT: Users should separate import files by school year. Files should be sorted by attendance date and then by school.

The attendance import will import both daily and period attendance. Importing attendance data is very processor intensive. All import files should be separated by year. Files should be sorted by attendance date and then school. Attendance imports should be run one file at a time when Focus is under heavy use, and 1 or 2 files at a time after hours. The maximum records per import file is 40,000. Larger files will need to be split out into 40,000 records per file.

Prior to importing Attendance it is important to make sure Attendance Codes are setup for all school years being imported, and that the short_name values for your codes setup in Focus match the values in the "Attendance Code" field in your import files.

If you choose the import method Delete and Insert it will delete both daily and period attendance for the students and attendance date (keys) in your file regardless of which fields (period or daily attendance) have been mapped.

Periods will need to be copied back to prior years before importing historic years period attendance.

See the [Runtime Parameters](#) for information on the Custom Date Format, Custom Year Format, Date Column Options, File Contains Header Info, File Extension, Limit Row Count, Memory Cap, Source File, Testing Mode, and the Time Stamp Output Files.

Attendance Calendar

[Import Methods](#): Delete and Insert, Insert Only

Required Fields: School ID, School Year, School Date

Static Fields: Minutes (999)

Log File Name: ATTENDANCE_CALENDAR.log

Local ID: SCHOOL_ID, SYEAR, SCHOOL_DATE

IMPORTANT: If importing for multiple schools, the import file must be sorted by School ID.

See the [Runtime Parameters](#) for information on the Custom Date Format, Custom Year Format, Date Column Options, File Contains Header Info, File Extension, Limit Row Count, Memory Cap, Source File, Testing Mode, and the Time Stamp Output Files.

Courses

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only, Synchronize

Required Fields: Local Course Number, School ID, Title, Course_ID

Static Fields: School Year (selected year)

Log File Name: COURSES.log

Local ID: SHORT_NAME, SCHOOL_ID, SYEAR

IMPORTANT: If importing for multiple schools, the import file must be sorted by School ID.

A Note on Courses Imports: Course subjects are generated based on the field "Graduation Subject", mapped in this import.

<u>American History</u>	[GRAD_SUBJECT_ID] => LEGACY:AH => FOCUS:192
<u>Biology</u>	[GRAD_SUBJECT_ID] => LEGACY:SC => FOCUS:185
<u>Career and Technical</u>	[GRAD_SUBJECT_ID] => LEGACY:VO => FOCUS:197
<u>Economics</u>	[GRAD_SUBJECT_ID] => LEGACY:EC => FOCUS:193
<u>Electives</u>	[GRAD_SUBJECT_ID] => LEGACY:EL => FOCUS:203
<u>English</u>	[GRAD_SUBJECT_ID] => LEGACY:EN => FOCUS:228

The courses import tool is used to import course data, and together with the sections import tool, makes up the master schedule. **Course numbers should not contain spaces or dashes as this will cause the student schedule records to reject.**

See the [Runtime Parameters](#) for information on the Custom Date Format, Primary Key Options, Custom Year Format, Date Column Options, File Contains Header Info, File Extension, Limit Row Count, Memory Cap, Source File, Testing Mode, and the Time Stamp Output Files.

Course Catalog

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only

Required Fields: Course Number (Short Name)

Static Fields: none

Log File Name: MASTER_COURSES.log

Local ID: SHORT_NAME (customizable in Configure Import Keys)

The course catalog import tool imports data into the course catalog which is used to validate any course and section data that is inputted into Focus. To utilize the course catalog the system preference for course catalog must be turned on.

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Primary Key Options, Source File, Memory Cap, and the Limit Row Count.

Course History

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only, Insert All, Synchronize

Required Field: Student ID, Marking Period ID5, Course Number

Static Fields: School Year (selected year), School ID (selected school), Course Weight (1)4

Log File Name: STUDENT_REPORT_CARD_GRADES.log

Local ID: STUDENT_ID, MARKING_PERIOD_ID, COURSE_NUM

IMPORTANT: If importing for multiple schools, the import file must be sorted by School ID

Certain fields, although not required, are very important flags for course history records. The COURSE_HISTORY column is a Y/N field used to flag records that should be included in the students transcript.

Instructor names are stored in a text field, and so the full name, not the staff_ID, of the teacher who taught the course will need to be included as a field in your file if you wish to have this information displayed.

To Import Exam Grades put an "E" in front of the marking period short name.

Directions for Large School Districts: The course history import is a very processor intensive import, as the course history table can easily contain millions of records and running the tool in Update/Insert requires the entire database of course history data to be loaded into local memory for comparison and processing. Because of this, the tool can quickly run out of memory; importing course history data in update/insert mode is currently not possible for large districts. Firstly, its recommended that data be deleted and then inserted on larger districts. Second, it is recommended that for each year of course history a separate file be created. When loading each file there are two options:

1. Select the school year from the drop-down menu at the top of the Focus menu. Then run the import but do not map the school year.
2. There must not be any of the course history records that are in the import file already in the database or the import tool will create duplicate records. Run the import in Insert Only mode instead of Update/Insert

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Calculate GPA, Weight GPA by Credits, Source File, Memory Cap, and the Limit Row Count.

Discipline Incidents

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only

Required Fields: none

Static Fields: School Year (selected year)

Log File Name: DISCIPLINE_INCIDENTS.log

Local ID: CUSTOM_19 (customizable in Configure Import Keys)

Discipline Incidents are used to tie multiple Referrals to a single incident. They are tied to Referrals on the field "Referral ID". Student ID is not mapped during the Discipline Incidents Import, as Student IDs are tied to Referrals during the Discipline Referral import (which must be done prior to this import).

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Primary Key Options, Source File, Memory Cap, and the Limit Row Count.

Discipline Referrals

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only, Synchronize

Required Fields: Student ID, Referral Date1

Static Fields: School Year (selected year), School ID (selected school)

Log File Name: DISCIPLINE_REFERRALS.log

Local ID: CUSTOM_83 (customizable in Configure Import Keys)

IMPORTANT: If importing for multiple schools, the import file must be sorted by School_ID

Discipline Referrals can be tied to discipline incidents by mapping the incident ID. The reporter field of the discipline referrals import file should match the key value assigned in configure import keys for the users table. If the previous SIS only stores the users names in the referrals entry, a custom field should be setup to store the reporter name and should be used for the import.

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Source File, Memory Cap, and the Limit Row Count.

District Address Catalog

[Import Methods](#): Insert Only

Required Fields: none

Static Fields: School Year (selected year)

Log File Name: ADDRESS_TO_DISTRICT.log

Local ID: PRIMARY_ID (customizable in Configure Import Keys)

This district address catalog is used to store the district zoning information. When the system preference for district address catalog is turned on, all new students added in Focus will have their address verified against the address catalog to confirm the student is in the appropriate district zone. Typically the district address catalog import file is acquired from the 911 services of the county.

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Source File, Memory Cap, and the Limit Row Count.

Log Entries (Referrals and Students)

[Import Methods](#): Insert Only

Required Fields: none

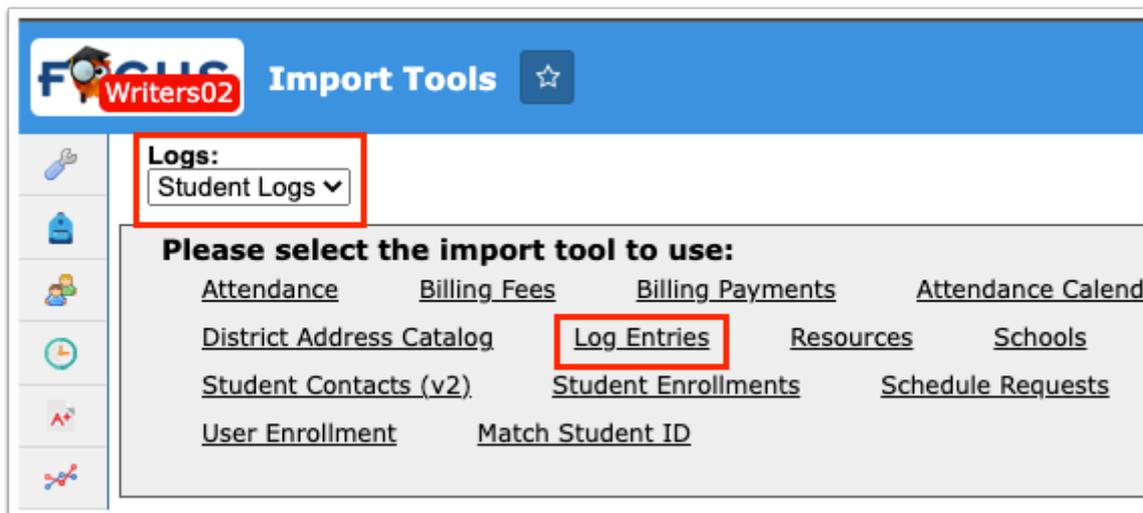
Static Fields: none

Log File Name: DISCIPLINE_REFERRALS_LOG_ENTRIES.log (referrals)

Log File Name: STUDENT_LOG_ENTRIES (students)

Local ID: CUSTOM_LOCAL_ID (customizable in Configure Import Keys)

Selecting the Log Entries Import Tool will bring up a pull-down menu populated with the 3 types of logging fields available.



Once you have made your selection, you will see the normal import tool screen with available runtime parameters.

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Source File, Memory Cap, and the Limit Row Count.

Match Student ID

Required Fields: First Name, Last Name, Gender, Date of Birth

Static Fields: none

Match Student ID is not an import, rather it is a tool for getting Student IDs into ACT or SAT files based on a student's first name, last name, gender, and date of birth. Match Student ID will automatically convert the gender codes 1 to M and 2 to F (for SAT) or 4 to M and 6 to F (for ACT). After you select the four required fields, press the match button to find the student's IDs.

For all the students that Match Student ID is unable to find, manually find the student by clicking Select Student to open the Focus student search screen.

Line	Student ID	Full Name	Gender	Birthday
2	Select Student	[REDACTED]	4	2012-03-20
3	Select Student	[REDACTED]	4	2012-03-20

After you have finished finding all of the students listed, press the Save button. This will save a new copy of the import file in your uploaded-assets/export/test_history/ directory with the Student IDs now in the file.

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Primary Key Options, Source File, Memory Cap, and the Limit Row Count.

Schools

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only

Required Fields: Local School ID, Title

Static Fields: none

Log File Name: SCHOOLS.log

Local ID: CUSTOM_327 (customizable in Configure Import Keys)

The Schools Import Tool is especially useful for districts with a large number of schools. To preserve the school numbers from your legacy system (or use your state DOE's Assigned numbers), make sure to set Primary Key to CUSTOM_327.

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Source File, Memory Cap, and the Limit Row Count.

Sections

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only, Synchronize

Required Fields: Local Course Period ID, Course ID

Static Fields: School Year (selected year), School ID (selected school), Course Weight (1), Grade Posting Scheme ID (0)

Log File Name: COURSE_PERIODS.log

Local ID: SYEAR, SHORT_NAME (when referred to from ImportSchedule & ImportSections)

Local ID: SYEAR, SCHOOL_ID, SHORT_NAME (when referred to from all other tools)

IMPORTANT: If importing for multiple schools, the import file must be sorted by School ID.

The sections import tool is the second part of the master schedule import, and should be run after the courses import. Section data is what assigns a teacher to a period and room, a student to a class, and contains other elements that are an integral part of scheduling. Most of the section data filters down to the students schedule when the schedule is built in Focus.

Marking periods should be setup prior to this import. Marking Periods must be included in the Sections import for the Student Schedules Import Tool to work.

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Primary Key Options, Section Day Column Options, Source File, Memory Cap, and the Limit Row Count.

Schedule Requests

Import Methods: Update or Insert, Update Only, Delete and Insert, Insert Only, Synchronize

Required Fields: Student ID, Course ID

Static Fields: School Year (selected year), School ID (selected school)

Log File Name: SCHEDULE_REQUESTS.log

Local ID: STUDENT_ID, COURSE_ID

IMPORTANT: If importing for multiple schools, the import file must be sorted by School ID

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Source File, Memory Cap, and the Limit Row Count.

Students

Import Methods: Update or Insert, Update Only, Delete and Insert, Insert Only

Required Fields: Local Student ID, First Name², Last Name²

Static Fields: none

Log File Name: STUDENTS.log

Local ID: CUSTOM_53 (customizable in Configure Import Keys)

When the user wishes to insert students using the same student ID in the import file as the student ID in Focus the user should change the Primary Key Option field to SIS Primary Key, and set the key to CUSTOM_53 in "Configure Import Keys."

When importing students in any mode that insert's records, the student's first name and last name are required fields.

When adding additional information to existing student records, make sure to always run this tool in "Update Only" mode.

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Primary Key Options, Source File, Memory Cap, and the Limit Row Count.

Student Addresses

Students can be tied to one or more addresses. Each address record contains a residential address. If the students mailing address is different from the residential address, the mailing address should be on the same record as the residential address. Each address record can be flagged for Mailing (a Y/N field). This determines whether an address record should be included when printing Avery labels. The Avery labels will automatically use the mailing address if it is listed.

It is recommended to import student addresses and all contacts using the same source file. This will ensure that contacts are assigned correctly and duplicate contact entries do not occur.

In your source file, contact names can be in either single or multiple columns. Name suffixes should be mapped separately.

One contact per student should be mapped per import. Additional contacts should be mapped one per student, running the tool as many times as necessary.

Version 2 (v2) separates the addresses and contacts into two separate imports. It also allows for updates to be run, in addition to inserts, which the previous version (v1) does not allow for.

If running Student Addresses (v1) and wanting to import multiple parents information it is important to set up additional fields for the addresses to map to for the additional parent data.

Student Addresses (v1)

Import Methods: Insert Only

Required Fields: none

Static Fields: none

Log File Name: ADDRESS.log

Local ID: ADDRESS (customizable in Configure Import Keys)

Student Addresses (v2)

Import Methods: Update & Insert, Insert Only, Update Only

Required Fields: Student ID

Static Fields: Residence (Y)

Log File Name: ADDRESS.log

Local ID: ADDRESS, ADDRESS2, CITY, STATE

Student Contacts: Students can have one or more contacts. Each contact record is tied to an address record. If the student contact is at the same address as the student, the address record in the contact file should be the same and flagged as Resides With Student. If the contact is at a different address than the student, this flag should not be used and the address should be

different than that of the students. The relationship to the student is marked with the Student Relation field. Focus has a number of pre-defined relationships, shown below:



If an option is used that is different than the Focus pre-determined relationships, the field will automatically be populated with the custom relationship.

Student Contacts (v2)

Import Methods: Update & Insert, Insert Only, Update Only

Required Fields: Full Name or First Name/Last Name

Static Fields: none

Log File Name: PEOPLE.log

Local ID: LAST_NAME, FIRST_NAME, MIDDLE_NAME, STUDENT_ID

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Contact Name Options, Source File, Memory Cap, and the Limit Row Count.

Student Enrollments

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only, Synchronize

Required Fields: Student ID, School ID

Static Fields: School Year (selected year), Start Date (current date)

Log File Name: STUDENT_ENROLLMENT.log

Local ID: SCHOOL_ID, STUDENT_ID

IMPORTANT: If importing for multiple schools, the import file must be sorted by School ID

Student enrollment data is what assigns a student to a school or schools for a specific enrollment period. Students will not be visible in Focus until at least one enrollment record is created for each student. The rolling retention option can be imported into Focus as the field Next School. Student enrollment records are required to have a school ID, enrollment code and enrollment date.

A Note on Student Enrollments Imports: The Rolling / Retention Options are mapped as Next School. Values for this field can be:

- The School ID or "Next grade at current school"

- The code 0 for Retain

- The code -1 for Do not enroll after this school year

For concurrent enrollment, use the field Second School and map to a Y value.

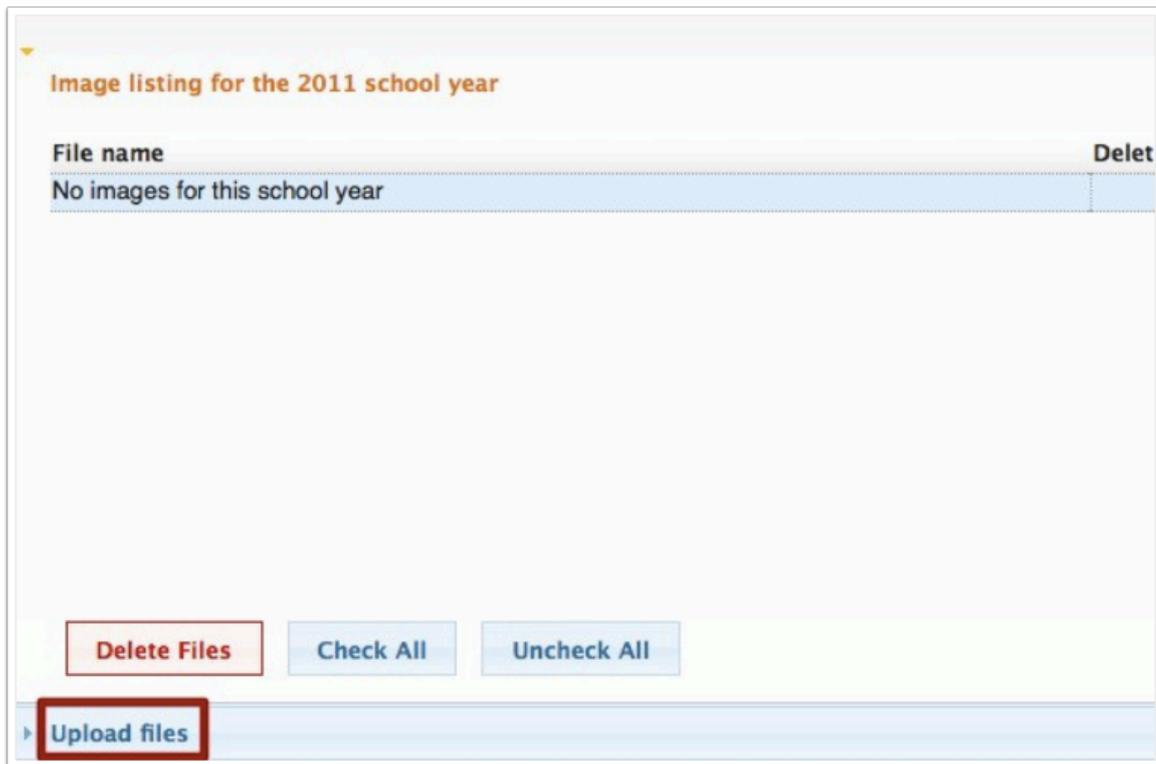
See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Source File, Memory Cap, and the Limit Row Count.

Student Photos

Student Photo filenames are formatted as XXXXXXXXXX.jpg where XXXXXXXXXX= Student ID. Student Photos will only be displayed for students with a .jpg matching their Student ID that has been uploaded via this Import Tool. There is a 10MB limit per image.

After selecting the Image Upload tool from the School Setup menu, you will see the screen in the image below (assuming you have no files uploaded currently. If you do, you will see a list of the files already uploaded).

First, click on Upload files.



The file format accepted is JPG. You may upload single or multiple JPG files, or a ZIP archive of JPG files.

Click on the "Select files for upload" button.



You will then get an Explorer window in Windows, or a Finder window in OSX(Mac). Navigate to the student photos you would like to upload. Select desired files.

The files you have chosen will be displayed as yellow boxes with name and filesize. Click on "Upload Images."



Once all files have been uploaded click on "Image listing for the 20xx school year."



You will then see all student photo files for the selected school year.



You can delete (or mass delete) photos from this screen as well.

If you have followed these steps exactly you should see a student photos displayed in the General tab of [Student Info](#) > Demographic.

Student Schedules

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only, Synchronize

Required Fields: Student ID, Marking Period ID5, Course Period ID

Static Fields: School Year (selected year), School ID (selected school), Start Date (current date)

Log File Name: SCHEDULE.log

Local ID: STUDENT_ID, COURSE_PERIOD_ID

IMPORTANT: If importing for multiple schools, the import file must be separated by School Year and sorted by School ID. Student Schedules should be imported one file at a time.

Student schedules imports are processor intensive. Student schedules assign students to sections in the master schedule.

The Student Schedules Import Tool uses a value called CoursePeriod-ID (a concatenation of the Course Number and the Section ID e.g. 1006700-001) to assign students to sections. Course numbers should not contain spaces or dashes as this will cause the student schedule records to reject.

When running a Student Schedules import in update mode, you must include a start date as the import tools will otherwise search for schedule records with the current date as the start date.

If you receive a DB Error Message as follows:

```
ERROR: current transaction is aborted, commands ignored until end of transaction block
```

This is because a previous import of student schedules failed, and was unable to recreate the required table indices. Please run the following queries and run the import again:

```
CREATE UNIQUE INDEX schedule_pkey ON schedule USING btree (syear, student_id, course_id, course_weight, course_period_id, start_date);
```

```
CREATE INDEX schedule_ind1 ON schedule USING btree (course_id, course_weight);
```

```
CREATE INDEX schedule_ind2 ON schedule USING btree (course_period_id);
```

```
CREATE INDEX schedule_ind3 ON schedule USING btree (student_id);
```

```
CREATE INDEX schedule_ind4 ON schedule USING btree (syear, school_id);
```

```
CREATE INDEX schedule_ind5 ON schedule USING btree (marking_period_id);
```

```
CREATE INDEX schedule_ind6 ON schedule USING btree (start_date, end_date);
```

```
CREATE INDEX schedule_ind7 ON schedule USING btree (id);
```

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Section Day Column Options, Source File, Memory Cap, and the Limit Row Count.

Test History

[Import Methods](#): Delete and Insert, Insert Only, Insert All, Synchronize

Required Fields: Student ID, Administration Date, Test ID3, Test Part ID3

Static Fields: Test ID (selected test)

Log File Name: TEST_HISTORY_ADMINISTRATIONS.log

Local ID: STUDENT_ID, ADMINISTRATION_DATE

Test History v1 is for people who had already built a template prior to 5.1.

Test History v2 is what we use for the portable mapping templates that can be shared between the schools and districts.

Please note: Test History v2 uses the Test "Short Name" field and Test Part "Subject Code" field as the keys for mapping, this means you must have a value setup in these fields in order import data with this version.

The test short names and test part short name\subject codes are setup under Assessment > [Standardized Tests](#).

Sorting:

For scores to group correctly your import file must be sorted by:

1. Student ID
2. Test Short Name
3. Administration Date

Test History Imports Can Be Run in Two Ways: The first way involves importing multiple types of tests from the same file. Test History files with multiple test types should be separated by school year and sorted by student ID then administration date. Importing test files with the wrong sort order will result in duplicate administrations for each test part. When importing multiple tests, each test is distinguished by the test short name that should match what is setup in Focus. The test parts can be identified by either the test part short name or test part subject code. Whichever part identifier is selected should match what is setup in Focus.

The second is for importing a specific test. No sorting is required with this method.

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Import Test Parts By, Source File, Memory Cap, and the Limit Row Count.

Users

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only

Required Fields: Local User ID

Static Fields: School Year (selected year), Profile ID (profile ID for Teacher)

Log File Name: USERS.log

Local ID: USERNAME (customizable in Configure Import Keys)

The Users Import Tool is used to import all faculty and staff. Profiles should be set up prior to this import, and your data should include a corresponding field. For the custom primary key, you will want to use either your legacy system staff_id, username (if this value is unique - no two users have the same value), or social security number.

Whichever field you choose to use as the custom primary key during this import should be present in all other imports where staff_id is mapped (most importantly sections).

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Primary Key Options, Source File, Memory Cap, and the Limit Row Count.

Resources

[Import Methods](#): Update or Insert, Update Only, Delete and Insert, Insert Only

Required Fields: none

Static Fields: School ID (selected school), Type (Y)

Log File Name: RESOURCES.log

Local ID: SHORT_NAME (customizable in Configure Import Keys)

Resources are where Focus typically stores room data. After importing, make sure to setup at least one Resource Category in School Setup via Setup > [Resource Scheduler/Rooms](#) > Setup the Resources (e.g. Rooms).

See the [Runtime Parameters](#) for information on the Testing Mode, Time Stamp Output Files, File Extension, File Contains Header Information, Date Column Options, Custom Year Format, Custom Date Format, Primary Key Options, Source File, Memory Cap, and the Limit Row Count.

Output Files

Each import tool generates a set of output files, to be used during the review of an import, as well for debugging issues that may arise. These come in the form of log files, exception files, and finish files. If Time Stamp Output Files is set to yes, each file will include a unique UNIX timestamp in the file name. If it is set to no, then a generic file name specific to that import tool will be used, and will be overwritten if it already exists.

Log Files

Log files contain a comprehensive list of operations performed during the execution of an import, and provides progress along each step of the process. This file is available for download from within Focus, once the import is complete, by clicking on the button below the final import statistics. Log files can be divided into three four sections:

Settings

Shows a listing of all options configured on the setup page of the import, along with any others that are set by default for a particular import tool. It also shows the start date and time, and the process ID of the PHP thread, which can be used to cancel an import.

2011-12-06 20:28:53

PID: 25353

```
Original File Name: Staff.txt
File Type:          Tab-Separated Values
Unique Log File Name: Yes
File Header Row:   Yes
Date Type:         From 1 Column
Year Format:       YYYY
Date Format:       yyyymmdd
Primary Key Type:  Custom Key Fields
Section Day Type:  Not Applicable For This Tool
Contact Name Type: Not Applicable For This Tool
Part ID Type:     Not Applicable For This Tool
School Year Type: Not Applicable For This Tool
Import Method:    insert/update
Template Name:    Staff_9_20
Import Tool Name:  Import/ImportStaff.php
Import Tool Location: /var/www/html/focus/modules/Import
Verbose Output:   Yes
Static Year:      Not Provided
Running Mode:     Live
Memory Cap:       Not enabled
Limit Rows:       Process all rows
Calculate GPA:    Not Applicable For This Tool
Base Class Rank On: Not Applicable For This Tool
Weight GPA By Credits: Not Applicable For This Tool
```

Preparation

Goes through each record in the import file one at a time. After initial processing, each entry is either going to be a message stating that the record already exists in the database in its current form, or its going to show a breakdown of what is new or changed in the record. That block of information is going to show five different pieces of information: any import tool diagnostic messages, including whether the record is being updated or inserted and any error messages; discarded data that has no corresponding column in the database, or already exists in the database in its current form; generated data that doesnt come directly from the import file, but is instead created on the fly based on other columns; unchanged data that is going into the database without any conversion or change; and converted data, which is being converted from one form to another, such as ID numbers.

```

Switching to school 1042 (EL - Swimming Pen Creek Elementary)

000221: Query Type Not Assigned; record ignored (record likely already
exists)

000222: Query Type Not Assigned; record ignored (record likely already
exists)

000223: Query Type Not Assigned; record ignored (record likely already
exists)

<pre>Array
(
  [IMPORT_TOOL_MESSAGES] => Array
  (
    [QUERY_TYPE] => UPDATE
  )

  [DISCARDED] => Array
  (
    [CUSTOM_556] => 100579
    [USERNAME] => srmcnamara
    [LAST_NAME] => MCNAMARA
    [FIRST_NAME] => SARAH
    [MIDDLE_NAME] => N
    [EMAIL] => srmcnamara@mail.clay.k12.fl.us
    [CUSTOM_607] => 0001063209
    [PROFILE_ID] => Teacher Pilot 2
    [SYEAR] =>
  )

  [GENERATED] => Array
  (
    [PROFILE] => teacher
    [STAFF_ID] => 131666
  )

  [CONVERTED] => Array
  (
    [SCHOOLS] => LEGACY:0451 => FOCUS:,1020,
  )
)
</pre>

```

Import

Where the actual database updates and inserts take place.

```

Query Pointer->7315/14620 (50.03%) - (1463 Query per: 48 seconds)

Imported 8000/14620 (54.72%)
Memory usage: 485.47 MB
Time elapsed: 576 s
Rows/sec: 327.16

```

Summary

A listing of how many records were affected by the import, memory status, and time elapsed.

```
<pre>Array
(
    [executed] => 160904
    [inserted] => 14620
    [updated] => 0
    [ignored] => 0
    [bad_data] => 0
    [obsoleted] =>
    [duplicates] => 0
)
</pre>

End Time: 2011-12-06 17:27:51
Calculated Time: 00:13:21
Memory in use before quitting the Import process: 393.43MB
```

Other information may be interspersed within the file, and is usually different from tool to tool.

Exception Files

Exception files are identical in format with the original import file, including column names and delimiter type, except that they will have an additional column at the end of each line of data for any error message that the record generated. The purpose of this file is to show any invalid data, and present it in a format where only those rows can be re-imported into the system after any issues have been resolved. Because the original columns are intact, any existing mapping template for that data should work without modification. Exception files will not be generated if there is no exception data and the import file has no column header row. If generated, this file is available for download from within Focus, once the import is complete, by clicking on the button below the final import statistics.

Reading Exceptions

If an exception file is generated, it is important to understand what the exception message means in order to correct the problem with the import. This document will not cover every possible exception, but instead focuses on some common exceptions.

1. Required Import Field When the import tools message field reads Required Import Field, this should indicate a problem with one or more of the fields on the line of the exception. Although each of these exceptions must be handled on a case by case basis, typically the line in question will contain a blank in a required field, or an otherwise invalid value.
2. Invalid Operation This exception is typically thrown during an INSERT ONLY import when the row of data already exists or during an UPDATE ONLY when a row of data does not exist.
3. Attempting to insert a record with a pre-existing key field. (Key Collision) Exceptions reading Key Collisions will occur when an INSERT is performed on a record with a key that already exists

4. Attempting to update a record without a pre-existing key field. (Key Missing) This exception usually occurs when running an UPDATE on a record that does not exist.

Finish Files

Finish files are 1-byte files placed in the same directory as the import data to indicate that the import job ran to completion. The contents of that file are a single space. When checking for finish time, refer to the file system date stamp on the file, not the timestamp in the file name. This file can only be viewed from with an ssh session on the hosting server, and is not available through the Focus interface.

Using Log Files During Imports

In order to track the process of an import, a UNIX command line program called tail can be used. To follow the import:

Use ssh to connect to the server hosting the Focus installation.

Issue the following command:

```
tail -f [web root]/uploaded-assets/import/logs/[import file log name]
```

where web root is the directory that the web server uses to hold its files (this varies between Focus installations and hosts), and import file log name is the log file being generated by the import tool, and ends with .log. As an example:

```
tail -f /var/www/html/uploaded-assets/import/logs/STUDENTS.log
```

This will direct all output being sent to the log file to the ssh session, and can be a great deal of data. For a more concise output with fewer details add `| egrep ^\w` to the end of the tail command, like this:

```
tail -f /var/www/html/uploaded-assets/import/logs/STUDENTS.log | egrep ^\w
```