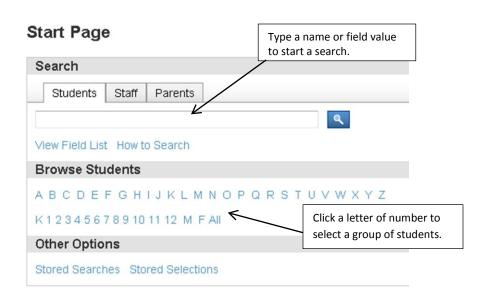## Smart Search

Smart Search is a PowerSchool auto-completion feature that works with the Search field. Once enabled, as you type in the Search field, the feature creates a suggestion menu below the field, relating to possible student or staff names, fields, or stored searches. Use the cursor to select your choice, or continue typing in the field. Smart Search must first be enabled at the district level. Once enabled at the district, it must be enabled at your school. Your security settings may affect your ability to enable the feature or the entry choices for the field.

Function	Description
Change password	Change current password to a new one.
Default sign in page	Sets the default sign in page.
Initial student screen	Sets the initial student screen.
Interface	Sets interface options.

Setup
School
System
Personalize

Smart Search Options
Enable Smart Search ☑
Include Inactive Student/Staff Results ☑

Basic Searches

Start Page

Search
Students | Staff | Parents

Type a name or field value to start a search.

View Field List   How to Search

Browse Students
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
K 1 2 3 4 5 6 7 8 9 10 11 12 M F All

Click a letter of number to select a group of students.

Other Options
Stored Searches   Stored Selections

Search Commands

There are three parts to a search: Field Name, Comparator and Search Argument.

Field Name	Comparator	Search Argument
First_Name	=	Maria

Other Comparators:

Comparator	Means This	Does This	Example
=	Equals	Searches for an exact match of the search argument.	**first_name=Mike** (would find all students with the first name of Mike)
<	Less Than	Searches for all matches less than the search argument.	**grade_level<5** (would find all students in grades K-4)
>	Greater Than	Searches for all matches greater than the search argument.	**grade_level >9** (would find all 10-12 grade students)
<=	Less Than or Equal to	Searches for all matches less than or equal to the search argument.	**grade_level <=4** (would find all students 4th grade and below)
>=	Greater Than or Equal to	Searches for all matches greater than or equal to the search argument.	**grade_level >= 10** (would find all students grade 10-12)
#	Does Not Equal	Searches for all matches that are anything other than the search argument.	**Wrestling#1** to find all students who are NOT involved in Wrestling (all students except wrestlers appear in you selection)
in	One of the values is present in the field	Searches for all the matches that contain one of the search arguments. Each search criteria must be separated by a comma.	**Last_Name in Smith,Jones** (Searches for students whose last names are Smith or Jones)
contains	The value is contained in the filed	Searches for matches where the search argument is anywhere in the field.	**street contains Maple Dr** (searches for students living on Maple Dr)
!contain	The value is not contained in the field	Searches for matches where the search argument is not anywhere in the field.	**street !contain Maple Dr** (searches for students not living on Maple Dr)
@	Wildcard	Replaces unknown information in the search argument	**First_Name=jacq@** (searches for any student whose first name begins with jacq – such as Jacquine and Jacqueline)

PowerSchool

Compound Searches

PowerSchool allows for the combining of multiple searches, called Compound Searches. This is done by separating each search with a semi colon (;).
For Example: **Grade_level=1;gender=M**

Search Codes

Search Codes are powerful searches that require PowerSchool to perform a calculation in the database to return the search results.

Search Code	Does This	Example
*birthday	Searches for students whose birthday is today, on a certain date or within a specific range	*birthday=today *birthday=10/28 *birthday>=10/1;*birthday<=10/31
*as_of	Searches for students who were active on the specific date	*as_of=11/23/2011
*not_enrolled_in_period	Searches for students who are not enrolled in a course for the specific period	*not_enrolled_in_period=4
*enrolled_in	Searches for students who are currently enrolled in a specified course or section	*enrolled_in=CTR2010 *enrolled_in=CTR2010.2
*not_enrolled_in	Searches for students who are not currently enrolled in a specified course	*not_enrolled_in=CTR1010
*has_completed_course	Searches for students who have at least one historical grade entry for the specified course	*has_completed_course=CTR1010
*has_not_completed	Searches for students who do not have any historical grade entries for the specified course	*has_not_completed=CTR1010
*cumulative_credit_hours	Searches for students with the specified number of credit hours	*cumulative_credit_hours=12 *cumulative_credit_hours<15 *cumulative_credit_hours<85
*number_of_classes	Searches for students who are currently enrolled in the specified number of classes	*number_of_classes=8 *number_of_classes<7 *number_of_classes#4
*hours_requested	Searches for students who have course requests for the specified number of credit hours	*hours_requested=6 *hours_requested<5 *hours_requested#8
*special_program	Searches for students who are enrolled in the specified special program	*special_program=PUF

Search Prefixes

There are three different Prefixes you can use to enhance searches.

Prefix	Use
/	**This will include all students including those who are inactive. For example *as_of=9/30/20112 will find those students who were in your school on September 30, 2011 who are still active but /*as_of=9/30/2011 will find all of the students who were active on September 30, 2011 even if they have left your school**
&	**This will search within the group you have found. For example /Enroll_status=2 will find the students in your school who have been transferred out. If from there you wanted to narrow down the search to those who left since September 30, 2011, you would then search &/exitdate>9/30/2011**
+	**This will add students to your list. For example AB_spcl_edu_stu_1=45 would give you all of the students who are coded 45 - Deaf but if you also wanted the students who were coded 46 – Blind, you would then search +AB_spcl_edu_stu_1=46**

Useful Searches

Comand	Use
Alert_medical#	Find students with medical alert information
/enroll_status= -1	Find pre-registered students
enroll_status= 0	Find all active students (you don't need the / for active students)
/enroll_status= 1	Find inactive students
/enroll_status= 2	Find transferred out students
/enroll_status= 3	Find graduated students

Multi School Enrollment Searches

Search Expression	Result
*secondarystudents=all	A list of students in all grades who are taking classes in the current school, but not enrolled in the current school
*secondarystudents=	A list of students in all grades who are taking classes in the current school, but not enrolled in the current school
*secondarystudents=10	A list of students in grade 10 who are taking classes in the current school, but not enrolled in the current school
*allstudents=all	A list of students in all grades who are taking classes in the current school, including those students enrolled in the current
*allstudents=	A list of students in all grades who are taking classes in the current school, including those students enrolled in the current
*allstudents=10	A list of students in grade 10 who are taking classes in the current school, including those students enrolled in the current school

Fee Searches

Search Expression	Result
*fee.id=noschool	The students who are not related to a school [Fee] record
*fee.id=nocourse	The students who are not related to a course [Fee] record
*fee.fee_balance=partial	The students who have at least one fee with a partial balance (partially paid)
*fee_balance.balance=partial	The students who have a partial global balance (partially paid)
*fee_balance.balance=0	The students who have a global balance equal to 0 Note: The [Fee_balance] record for a student is created when a transaction is created. Some students may not have a [Fee_balance] record because they never had a transaction created. This expression looks for [Fee_balance] records with a zero balance. The students who do not have a [Fee_balance] record will not appear in the results even though their balance is zero
*fee_balance.balance=null	The students who have no [Fee_balance] record. Note: This expression complements *fee_balance.balance=0 by returning the students who have no [Fee_balance] record

Stored Searches

Create a stored search when you know you're going to run the same set of search commands routinely.

1. On the Start Page, click **Stored Searches**
2. Click **New**
3. Give the search a descriptive title including your initials to indicate it is yours
4. In the Search instructions box, enter the search commands
5. If you are saving a compound search, enter one command per line
6. Click **Submit**

Stored Selections

Create a stored selection for a group of students that you will work with on a regular basis.

1. Select a group of students
2. From the Group Functions menu, click **Save Stored Selection**
3. Give the selection a name that describes the group and use your initials to indicate it is yours
4. From the list of options on the stored Selections page, click **Save the current selection with a new name**
5. Click **Submit**

Notes: -

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____