



Wisconsin PowerSchool  
User's Group Meeting  
November 18<sup>th</sup>-19<sup>th</sup>, 2009

*Kalahari Resort, Wisconsin Dells*

# Custom Screens/HTML

## Creating Custom Fields and Screens in PowerSchool

PowerSchool provides an easy-to-use interface for creating basic custom fields and screens. This feature is accessed by selecting System > Custom Fields / Screens from the PowerSchool administrative home page.

Students	Description
<a href="#">Custom Student Screens</a>	Provides quick setup of new student screens.
<a href="#">Student Fields</a>	Creates and edits custom fields.

  

Staff	Description
<a href="#">Custom Staff Screens</a>	Provides quick setup of new staff screens.
<a href="#">Staff Fields</a>	Creates and edits custom fields.

  

Courses	Description
<a href="#">Course Fields</a>	Creates and edits custom fields.

  

Sections	Description
<a href="#">Section Fields</a>	Creates and edits custom fields.

### Creating a Custom Field

Click on the field link for the type of custom field that you would like to create. A list of current custom fields will be displayed. Click the “new” button at the top of the list to create a new field.

<a href="#">New</a>	
Name	Description
<a href="#">12_Basketball</a>	
<a href="#">2_dose_hepb</a>	
<a href="#">2yr</a>	
<a href="#">4yr</a>	
<a href="#">ACT.12</a>	
<a href="#">ACT1OTHER_09</a>	

Enter the name of the field that you wish to create and a description of the data that will be stored in this field. The field name should be representative the data that it will hold and cannot contain spaces. If spaces are needed, substitute an underscore ( \_ ).

Option	Value
Field Name	<input type="text" value="student_email"/>
Description	<input type="text" value="Student's email address"/>

Click submit to create the custom field. The custom field will not appear in the field list until at least one student has been given a value for that field.

### Creating a Custom Screen

Click on the screen link for the type of custom screen that you would like to create (student or staff). A list of current custom screens will be displayed. Click the “new” button at the top of the list to create a new screen.

Screen	Edit fields
<a href="#">3rd Friday - PI1563 Data</a>	<a href="#">Edit Fields</a>
<a href="#">After-School Job</a>	<a href="#">Edit Fields</a>
<a href="#">Bicycle Registration</a>	<a href="#">Edit Fields</a>
<a href="#">English Proficiency</a>	<a href="#">Edit Fields</a>
<a href="#">Lunch Status</a>	<a href="#">Edit Fields</a>
<a href="#">Original Student ID</a>	<a href="#">Edit Fields</a>

Enter the name of the new screen.

Name

[Submit](#)

PowerSchool will return to the custom screens list. The screen has been created, and fields must now be added to the new screen. Click on "Edit Fields" for the new screen.

[New](#)

Screen	Edit fields
<a href="#">3rd Friday - PI1563 Data</a>	<a href="#">Edit Fields</a>
<a href="#">After-School Job</a>	<a href="#">Edit Fields</a>
<a href="#">Bicycle Registration</a>	<a href="#">Edit Fields</a>
<a href="#">English Proficiency</a>	<a href="#">Edit Fields</a>
<a href="#">Lunch Status</a>	<a href="#">Edit Fields</a>
<a href="#">New Student Screen</a>	<a href="#">Edit Fields</a>
<a href="#">Original Student ID</a>	<a href="#">Edit Fields</a>

The custom screen editor will appear indicating that there are no fields for this custom screen. Click the “New” button at the top of the screen create a new field.

Option	Value
Label	<input type="text"/>
Field Name ( <a href="#">Fields</a> )	<input type="text"/>
Sort Order	<input type="text" value="1"/>
Input Type	<input type="text" value="Entry field"/>
Width (applies only to 'entry field' and 'entry box' types)	<input type="text"/> (width in characters)
Height (applies only to 'entry box' type)	<input type="text"/> (height in lines)
Data for Popup or Radio Buttons (press return after each entry)	<input type="text"/>

**Label** – Enter the label that should appear for this field on the custom screen.

**Field Name** – Click on the Fields link and select the desired field.

**Sort Order** – Indicate the order you would like this field to appear on the custom screen.

**Input Type** – Select the type of input that will appear for this field. The next section describes the six types of entry fields.

**Width** – For entry fields and entry boxes, indicate the width in characters that will be available for data entry.

**Height** – For entry boxes, indicate the height in lines that will be available for data entry.

**Data** – This section allows the user to enter predefined values that will be available for selection on the custom screen when the input type is radio buttons or pop-up menu.

## Input Types:

**Entry Field** - This option will create a text box in which the user can enter text free-form.

**Static Field** – This option will create a field that will display a field value, but not allow the user to update the value of that field.

**Checkbox** – This option will create a box that the user may choose to check. This option is used for fields that have true or false values.

**Pop-up Menu** – This option will create a list of values from which the user can choose one.

**Radio Buttons** – This option will create a series of buttons from which the user can select one.

**Entry Box** – This is a larger version of the Entry Field. Use this option for descriptions and other longer input fields.

### Sample Custom Screen

The image displays a sample custom screen with the following components:

- Entry Field:** A small text input box.
- Static Field (Gender) F:** A field displaying the value 'F'.
- Checkbox:** An unchecked checkbox.
- Pop-up Menu:** A dropdown menu showing 'Option 1'.
- Radio Buttons:** Three radio buttons labeled 'Radio 1', 'Radio 2', and 'Radio 3', all of which are unselected.
- Entry Box:** A large text input area.
- Submit:** A blue button with the text 'Submit'.

## **Basic HTML in PowerSchool**

HTML is text that is interpreted by a web browser. This text tells the browser what to render on the user's screen. Most HTML is a series of opening and closing tags. An opening tag has the format `<tag>`, and a closing tag has the format `</tag>`. The bold tag (`<b>`) is a very commonly used HTML tag. If text should be rendered as bold in the browser, it can be enclosed in this tag as follows:

```
<b>This text will be bold when rendered in a browser</b>
```

Browsers do not render the actual tags. They use the tags to determine how to render the content between them.

A very commonly used structure is the HTML table. A table is represented with `<table></table>` tags. Nested within the `<table>` tags are table row (`<tr>`) tags. Nested within the table row tags are table data (`<td>`) tags.

Another feature used by PowerSchool is the `<input>` tag. This tag will render an input field in the browser in which the user can enter data. There are several types of inputs, but the most commonly used is "text". This type of input will render a text box for user input. A text input tag has the following format:

```
<input type="text" name="input_name" value="">
```

In PowerSchool, the name of the input designates the field that will hold the entered data. Preceding the field name, the table number must be included in square brackets. The table number for the students table is 01. Therefore, if we want to create an input for a student's first name (called `First_Name` in PowerSchool), the input tag would look like the following:

```
<input type="text" name="[01]First_Name" value="">
```

The "value" portion of the tag can be left equal to empty quotes, but must be present for the field to function properly.

The following is a sample table with a structure similar to that of many of the tables in PowerSchool.

```
<table border="solid" cellpadding="5">
  <tr>
    <td>First Name</td>
    <td><input type="text" name="[01]First_Name" value=""></td>
  </tr>
  <tr>
    <td>Last Name</td>
    <td><input type="text" name="[01]Last_Name" value=""></td>
  </tr>
</table>
```

How does this code render in a browser?

First Name	<input type="text"/>
Last Name	<input type="text"/>

This is a very basic introduction to HTML intended to get users more comfortable with viewing the code behind PowerSchool pages. For a more in depth look, try the W3Schools web site referenced at the end of this document or simply search the web for HTML tutorials.

## **Customizing Existing Screens in PowerSchool**

PowerSchool maintains two web roots (file structures) on the server for accessing screens for display. The live web root contains all of the pages (HTML files) needed by the application. There is also a custom web root where custom pages reside. PowerSchool will always look to the custom web root for a page before the live web root. Therefore, to customize an existing page, the first step is to copy that page from the live web root to the custom web root.

The live web root is located on the server at `PowerSchoolPremier/system/server/resources/web_root`.

The custom web root is located on the server at `PowerSchoolPremier/data/custom/web_root`.

### **Add Email Address to the Demographics Screen**

To demonstrate the process of modifying an existing PowerSchool screen, this section will step through the process of adding a student's email address to the demographics page. We will assume that a custom student field has been created named "email".

The first step is to copy the existing demographics page from the live web root to the custom web root. The demographics page is located in the live web root at `admin/students/generaldemographics.html`. This file must be copied and placed in the same location in the custom web root.

In this demonstration we will add the email field immediately after the student's name at the top of the page. The `generaldemographics.html` file in the custom web root can be opened with a text editor and modified. We first need to find the table row in the original table with the student's name. We must then add another row to the table. The first cell in the new row will have the label "Email Address". The second will have the input for the student's email address. Notice the name given to the text input for the email address. "[01]" indicates that this data will be stored in the students table (table 1). "email" indicates that this data belongs to the field named email.

```
<tr class="^[evenoddrrow]">
  <td class="bold">Name (last, first MI)</td>
  <td>
    <input type="text" name="[01]last_name" value="" size="17" maxlength="20" id="lastName">,
    <input type="text" name="[01]first_name" value="" size="17" maxlength="20" id="firstName">
    <input type="text" name="[01]middle_name" value="" size="15" maxlength="15" id="middleName">
  </td>
</tr>
<tr class="^[evenoddrrow]">
  <td class="bold">Email Address</td>
  <td><input type="text" name="[01]email" value=""></td>
</tr>
```

New Code

Save the altered file in the custom web root and open the demographics page in PowerSchool. The email field will now appear beneath the student's name.

Name (last, first MI)	Sample	Student	
Email Address	student@school.org		

### Oops... I made a mistake. Now what?

Because all customization is done with a copy of the original document, there is no fear of irrevocably breaking a PowerSchool page. If you find that a customization attempt has not gone as planned and need to restore the page to its original state, simply remove the page from the custom web root. PowerSchool will then display the original page from the live web root.

## PowerSchool Customization Resources

### Custom Fields and Screens User Guide for PowerSchool

This is an article available on PowerSource. Perform a search for this title. This article details the steps for creating custom fields and screens in PowerSchool.

**Customizing PowerSchool 101** - [www.sisresources.com/index.php?/SIS-Resources/customizing-101-articles.html](http://www.sisresources.com/index.php?/SIS-Resources/customizing-101-articles.html)

This site provides several how-to articles on PowerSchool customization including the creation of custom fields and screens and basic HTML.

**W3schools.com** - [www.w3schools.com/html/default.asp](http://www.w3schools.com/html/default.asp)

This website provides an excellent introduction to HTML. Short chapters introduce the user to all of the must-know HTML features with a generous number of practical examples. Try it yourself functions allow the user to alter code and observe the results with the click of a button.