## PowerFields Menu Option

<table>
<thead>
<tr>
<th>eXpression</th>
<th>Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Substitution</td>
<td>Percent of Total</td>
</tr>
<tr>
<td>Margin</td>
<td>Benchmarks</td>
</tr>
<tr>
<td>Tabulator</td>
<td>Control Values</td>
</tr>
</tbody>
</table>
PowerFields

PowerFields allow you to extend the data. With PowerFields you may apply formulas to the data and specify the condition under which they occur.

The results of the PowerField calculation are displayed in a new column.

This column is color coded to maintain a delineation between imported data (original) and data that has been acted upon.
Click **Edit PowerFields** to edit any existing PowerField or add a new one of any type.

Use the **Add** options to quickly add a specific type of PowerField.
**PowerFields - eXpression**

**Add:** Add a new expression.

**Duplicate:** Add a new expression based on the expression currently highlighted in the grid on the left.

**Edit:** Edit the currently highlighted expression in the grid on the left.

**Delete:** Delete the currently highlighted expression in the grid on the left.

**Field Name:** This will be the header name for the new column that will contain the results of the expression.

**Data Type:** Determines how the data will be displayed. See following page for definitions.
PowerFields – Data Types

From the **Data Type** dropdown, select how the data in the expression should be displayed.

**Currency**: Displays numerical data with two digits after the decimal, and will round to the nearest two digits if more exist. **Example** 1.73

**Date**: Displays data in a date format. **Example** mm/dd/yyyy

**Decimal**: Displays numerical data with all available digits displayed after the decimal point. **Example** 1.734628937

**Freeform**: Displays data exactly as it has been entered. Data is treated as non-numerical. **Example** 5x7w5p

**Integer**: Displays numerical data as a whole number. If digits exist beyond the decimal point, they will be rounded to the next whole number. **Example** 1.743 becomes 2

**List**: Data will be available in a pick list format when filtered.

**Percent**: Displays numerical data in a percentage format. **Example** .74 becomes 74%

**Year**: Displays four digit numerical data as a year.

**Yes/No**: Displays data in a yes or no format such that a 0 becomes a No and a 1 becomes a Yes.
Click the **Edit** button under **Use this Expression** to open the eXpression builder.

The grid on the left contains all the columns of the DataBook. Double click or drag and drop to place them in the panel on the left.

Constants may also be added to the formula by entering a **Custom Value** or selecting a predefined value from **Math Value** or **Date Value** in the User Values box on the bottom left.

When two fields have been placed into the expression the operator drop down is displayed.
**Group**: Combines highlighted elements together for complex calculations. Click each element to highlight then click **Group**.

**Un-Group**: Removes the grouping.

**Move Up / Move Down**: Change the placement of elements or groups. Click to select the element then click the appropriate **Move** button.
**Function:** Apply date, numeric or alpha-numeric functions to any element of the expression. Click the element, click **Function** and then select the appropriate function. Date, Math or String functions are displayed depending on the data type of the element.
PowerFields – Math Functions

Math functions are available for data elements defined as integers, decimals, percentages or currency data type. They return numerical values.

None: Selecting this option will remove any function previously placed on the data element.

Square Root: Applies a square root to data in this field.

Example  The square root of 36 returns the value 6

Squared: This option will square the number, or take it to the second power.

Example 2 squared returns the value 4

To Power X: Applies an exponential value to the value in this field.

Example To Power 4 takes the value 2 to the fourth power returning the value 16
PowerFields – Math Functions

**Round:** This option will round the data in this field to the nearest whole number.

  **Example** 2.75 rounds up to 3 and 2.2 rounds down to 2

**Round Up:** This option will round the data in the field UP to the next whole number.

  **Example** Both 2.75 and 2.2 round up to 3

**Round Down:** This option will round the data in this field DOWN to the next whole number.

  **Example** Both 2.75 and 2.2 round down to 2

**Absolute Value:** Measures how far numerical data entries are from zero, thereby eliminating negative numbers.

  **Example** Both 3 and -3 return an absolute value of 3

**Sine/Tangent/Arc Sine/Arc Cosine/Arc Tangent/Natural Log:** These options return the specified value of the data in this field.
PowerFields – String Functions

String functions are available for data elements defined as **String** or text data type. This may be non-numerical data or a number with no numerical significance, for instance the number on a basketball player's jersey or a student id number.

None: Selecting this option will remove any function previously placed on the data element.

Upper Case (ABC): This option will convert any text in the field to all capital letters.

  **Example** John Smith returns JOHN SMITH

Lower Case (abc): This option will convert any text in the field to all lower case letters.

  **Example** John Smith becomes john smith

Proper Case (Abc): This option will convert the first letter of any word into a capital letter and make the remaining letters lower case.

  **Example** johN SMITH becomes John Smith
PowerFields – String Functions

Trim Spaces: This option removes all spaces between words or characters.

Example  John Smith becomes JohnSmith

First (x) Characters: This option displays only the first "x" number of characters in that entry.

Example  The First 3 Characters of a phone number takes 843-556-5565 and returns 843

Last (x) Characters: This option displays only the last "x" characters in that entry.

Example  The Last 7 Characters of Customer ID information takes 0001374829 and returns 1374829

Middle (y) Characters Start at Position (x): This option starts "x" characters into the entry and returns "y" number of characters.

Example  The Middle 3 Characters Start at Position 5 of a phone number takes 843-556-5565 and returns 556
PowerFields – String Functions

**Convert to Number:** This takes numerical data previously categorized with a Freeform or String data type and converts it to a number. This allows it to be used for numerical calculations.

**Convert to Date:** This takes a date previously categorized with a Freeform or String data type and converts it to a date. This allows it to be used for date calculations.
Date functions are available for data elements defined with a Date data type.

None: Selecting this option will remove any function previously placed on the date element.

Get Day (Number): Pulls the day out of a full date entry.
   Example 6/28/2010 or June 28, 2010 returns 28

Get Month: Pulls the month out of a full date entry.
   Example 6/28/2010 or June 28, 2010 returns 6

Get Year: 2 Digit (Number): Pulls the year out of a full date entry as a two digit number.
   Example 6/28/2010 or June 28, 2010 returns 10

Get Year: 4 Digit (Number): Pulls the year out of a full date entry as a four digit number.
   Example 6/28/2010 or June 28, 2010 returns 2010
Convert to String (‘01/01/09’): Converts the date into a mm/dd/yy format and removes the “date” functionality so that the data will be treated as a string.

**Example** June 28, 2010 or 6/28/2010 returns “06/28/10”

Convert to String (‘01/01/2009’): Converts the date into a mm/dd/yyyy format and removes the “date” functionality so that the data will be treated as a string.

**Example** June 28, 2010 or 6/28/2010 returns “06/28/2010”

Convert to String (‘Jan 1, 2009’): Converts the date into a month date, year format and removes the “date” functionality so that the data will be treated as a string.

**Example** June 28, 2010 or 6/28/2010 returns “June 28, 2010”
**PowerFields – Date Functions**

**Adjust (x) Days:** Enter the number of days to add or subtract from the date.


**Adjust (x) Months:** Enter the number of months to add or subtract from the date.


**Adjust (x) Years:** Enter the number of years to add or subtract from the date.


**Last Day of Month:** Returns the last day of the month of the date.

PowerFields – Date Functions

**Last Day of Quarter:** Returns the last day of the quarter of the date.

**Example**  10/28/2010 returns 12/31/2010

**Last Day of Year:** Returns the last day of the year of the date.

**Example**  10/28/2010 returns 12/31/2010

**Day (x) of Month:** Returns the requested day of the month of the date.

**Example**  Day 15 of Month for 6/28/2010 returns 6/15/2010

**Day (x) of Quarter:** Returns the requested day of quarter of the date.

**Example**  Day 1 of Quarter for 11/28/2010 returns 10/01/2010

**Day (x) of Year:** Returns the requested day of the year of the date.

**Example**  Day 15 of Year for 10/28/2010 returns 1/15/2010
PowerFields - eXpression

Right click an element to apply Measure functionality to the expression. This allows for calculations using group and total amounts to be applied at row level.

**Sum-Group:** Returns the sub-total amount of the column for each group.

**Average-Group:** Returns the average amount of the column for each group.

**Maximum-Group:** Returns largest numerical entry or the last alphabetically in the column for each group.

**Minimum-Group:** Returns the lowest numerical entry or the lowest alphabetically in the column for each group.

**Count-Group:** Returns the total row count in the group.

**Sum-Total:** Returns the grand total amount of that column in the DataBook.

**Average-Total:** Returns the average amount of the column in the DataBook.

**Maximum-Total:** Returns largest numerical entry or the last alphabetically in the column in the DataBook.

**Minimum-Total:** Returns the lowest numerical entry or the lowest alphabetically in the column in the DataBook.
Placing **Sum-Group** on the Actual Amount element returns the total amount of the group for each row. **Sum-Total** returns the grand total on each row.
In this example, we are using **Sum Total** to calculate the percent of total spending in department 00720 for each project and account.

The eXpression is Actual Amount / Actual Amount (**Sum-Total**).

Our first row, Project 5649, determines “Percent of Spending” for Books expense by using the calculation: 99 / 4,424.23 = 0.02238.

The **Data Type** is set to **Percent** so that our result is displayed as 2.238%.

By placing a **Sum** measure on the Percent of Spending column, we see that the percent of spending for Books in the department is 42.920%.
PowerFields – eXpression

When the expression is complete, click the OK button to return to the Add PowerField dialog.

To place a condition under which the expression should be applied to each row, click the Edit button under For these Records.

The Condition Builder is displayed.
PowerFields - eXpression

Drag and drop or double click fields to create a conditional filter.
Multiple expressions and conditions may be added using **Expression** button in the lower left corner. Click on an expression and then click the **Trashcan** button to delete.

In this example:

Actual Amount is being multiplied by .10 for all rows where the Account Class is either an Expense or a Travel Expense.

For the rows where the Account Class is Payroll Expense the Actual Amount is multiplied by .05.

Leave the For These Records blank to apply the expression to all rows.

When the expression is complete, click the **Test** button. Then click Save. Click **Add** to add another expression or **OK** to return to the DataBook.
PowerFields - eXpression

To view the results of your expression, click **Refresh** on the DataBook. The results are displayed in a new column at the far right.

The new column may be acted upon in the same manner as any other data column. It may now be sorted, filtered or grouped. You may apply Column Header functions or use it in other expressions.

Expression columns are shaded yellow to distinguish them from data columns originating in the source system.

<table>
<thead>
<tr>
<th>Center Name</th>
<th>Account Class</th>
<th>Actual Amount</th>
<th>Projected Spend Incr</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Department: 00411</strong></td>
<td>Payroll Expenses</td>
<td>$251.57</td>
<td>$264.15</td>
</tr>
<tr>
<td>Project Number 6595</td>
<td>Payroll Expenses</td>
<td>$65.13</td>
<td>$63.14</td>
</tr>
<tr>
<td><strong>Total for Department: 00411</strong></td>
<td></td>
<td>$311.70</td>
<td>$327.29</td>
</tr>
<tr>
<td><strong>Department: 00404</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project Number 7412</td>
<td>Expenses</td>
<td>$189.53</td>
<td>$196.58</td>
</tr>
<tr>
<td>Project Number 5325</td>
<td>Expenses</td>
<td>$130.91</td>
<td>$144.00</td>
</tr>
<tr>
<td>Project Number 7412</td>
<td>Expenses</td>
<td>$20.00</td>
<td>$22.00</td>
</tr>
<tr>
<td>Project Number 5325</td>
<td>Expenses</td>
<td>$32.49</td>
<td>$35.74</td>
</tr>
<tr>
<td>Project Number 7368</td>
<td>Expenses</td>
<td>$348.81</td>
<td>$383.69</td>
</tr>
<tr>
<td>Project Number 5958</td>
<td>Expenses</td>
<td>$14.99</td>
<td>$16.49</td>
</tr>
<tr>
<td>Project Number 7308</td>
<td>Expenses</td>
<td>$206.00</td>
<td>$226.60</td>
</tr>
<tr>
<td>Project Number 5958</td>
<td>Expenses</td>
<td>$47.76</td>
<td>$52.54</td>
</tr>
<tr>
<td><strong>Total for Department: 00404</strong></td>
<td></td>
<td>$981.49</td>
<td>$1,079.64</td>
</tr>
</tbody>
</table>

Note in this example that the calculation against the rows with an Account Class of Payroll Expenses is \((\text{Actual Amount} \times 0.05) + \text{Actual Amount}\). The calculation against rows with an Account Class of Expense is \((\text{Actual Amount} \times 0.10) + \text{Actual Amount}\).

This is in keeping with the conditions placed on the expression.
The Substitution PowerField is used to create a new column, in which the value in the original column is replaced with a specified new value.

For example, a column in this DataBook contains the full or part time status of an employee. In the source system this field is described as F – Full Time and P – Part Time. Use the Substitution PowerField to simplify the data to F or P.
PowerFields - Substitution

Field Name: This will be the header name for the new column that will contain the results of the expression.

Data Type: Determines how the data will be displayed. See PowerFields – Data Types for definitions of the different data types.

Source Column: Select the original column of data to be substituted.

Collect Unique Values: Each unique value from the source column will be displayed.

In the Substitution Value column, enter the corresponding value to be displayed.
The Margin PowerField contains several pre-defined expressions - Margin, Markup, Difference and Ratio.

**Field Name:** This will be the header name for the new column that will contain the results of the expression.

**Type:** Select a pre-defined expression.

**Column 1:** Select from all columns that are defined as numeric within the current DataBook.

**Column 2:** Select from all columns that are defined as numeric within the current DataBook.

**Decimal Result:** Select to view result in decimal format.

**% Result:** Select to view result in percent format.
PowerFields - Margin

**Margin:** Subtracts column 2 from column 1 and divides that total by column 1.

**Example** Calculate profit margin where Column 1 = Sales Price  Column 2 = Cost  
(Sales Price – Cost) / Sales Price  or  \((374.38 – 359.38) / 374.38 = 0.040\) or 4.007%

*All expressions in the Margin PowerField are weighted.* The group totals are calculated using the Margin formula. In the Profit Margin column the group total is the Margin of Total Cost and Total Sales Price rather then a sum of all numbers in the Profit Margin column.
**PowerFields - Margin**

**Markup:** Divides column 1 by column 2 and subtracts 1.

**Example** % of price change where Column 1 = New Price and Column 2 = Original Price

\[(\text{New Price} / \text{Original Price}) - 1\] or \((53.00 / 38.00) - 1 = 0.395\) or 39.474%

All expressions in the Margin PowerField are weighted. The group totals are calculated using the Markup formula.
**PowerFields - Margin**

**Difference:** Subtracts column 2 from column 1.

**Example** Price difference where Column 1 = New Price and Column 2 = Original Price

\[ \text{New Price} - \text{Original Price} \text{ or } 515.00 - 500.00 = 15.00 \]
**PowerFields - Margin**

**Ratio:** Divides column 1 by column 2.

**Example** In a DataBook containing population information including the number of females and males in each county, find the female to male ratio for each county.

Column 1 = the number of females and Column 2 = number of males

Number of Females / Number of Males or 12,552 / 11,310 = 1.109

There are 1.109 females to each male in Abbeville County.

<table>
<thead>
<tr>
<th>County</th>
<th>Population</th>
<th>Males</th>
<th>Females</th>
<th>Female to Male Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abbeville</td>
<td>23,862</td>
<td>11,310</td>
<td>12,552</td>
<td>1.109</td>
</tr>
<tr>
<td>Aiken</td>
<td>120,940</td>
<td>58,596</td>
<td>62,344</td>
<td>1.064</td>
</tr>
<tr>
<td>Allendale</td>
<td>11,722</td>
<td>5,937</td>
<td>5,785</td>
<td>0.974</td>
</tr>
<tr>
<td>Anderson</td>
<td>145,196</td>
<td>69,436</td>
<td>75,760</td>
<td>1.091</td>
</tr>
<tr>
<td>Bamberg</td>
<td>16,902</td>
<td>7,921</td>
<td>8,981</td>
<td>1.134</td>
</tr>
<tr>
<td>Barnwell</td>
<td>20,293</td>
<td>9,739</td>
<td>10,554</td>
<td>1.084</td>
</tr>
<tr>
<td>Beaufort</td>
<td>86,425</td>
<td>43,803</td>
<td>42,622</td>
<td>0.973</td>
</tr>
<tr>
<td>Berkeley</td>
<td>128,776</td>
<td>64,972</td>
<td>63,804</td>
<td>0.982</td>
</tr>
<tr>
<td>Calhoun</td>
<td>12,753</td>
<td>6,042</td>
<td>6,711</td>
<td>1.111</td>
</tr>
<tr>
<td>Charleston</td>
<td>295,039</td>
<td>146,565</td>
<td>148,474</td>
<td>1.013</td>
</tr>
<tr>
<td>Cherokee</td>
<td>44,506</td>
<td>21,328</td>
<td>23,178</td>
<td>1.087</td>
</tr>
</tbody>
</table>
PowerFields - Tabulator

The Tabulator PowerField calculates either a running total or the amount of change from row to row.

**Field Name:** This will be the header name for the new column that will contain the results of the expression.

**Type:** Select to see a cumulative row total or the amount of change between the rows.

**Column:** Select the column of data to be tabulated.
In this example, the Actual Amount column is being tabulated for a running total in the Amount Run Total column.

In the Amount Change column, the amount of change is calculated and an up arrow indicates an increase from the previous row or group and a down arrow a decrease.
PowerFields - Ranking

Field Name: This will be the header name for the new column that will contain the results of the expression.

Column 1: Select the column of data to be ranked from all columns that are defined as numeric within the current DataBook.

Order: Select to rank in ascending or descending order.

<table>
<thead>
<tr>
<th>County</th>
<th>Population</th>
<th>Males</th>
<th>Females</th>
<th>Females Ranking</th>
<th>Males Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spartanburg</td>
<td>225,800</td>
<td>109,230</td>
<td>117,570</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lexington</td>
<td>167,611</td>
<td>81,668</td>
<td>85,943</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Anderson</td>
<td>145,196</td>
<td>69,436</td>
<td>75,760</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>Horry</td>
<td>144,053</td>
<td>70,516</td>
<td>73,537</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>York</td>
<td>131,497</td>
<td>63,181</td>
<td>68,316</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>Berkeley</td>
<td>128,776</td>
<td>64,972</td>
<td>63,804</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>Aiken</td>
<td>120,940</td>
<td>58,596</td>
<td>62,344</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

In this example, one column contains the rank order of males and one that of females so that they may be compared per county. While Anderson ranks 6 in the number of females, they rank 7 in the number of males.
PowerFields - % of Total

Field Name: This will be the header name for the new column that will contain the results of the expression.

Column: Select the column of data to view as percent of total.

Calculating against the Amount column, the Amount Pct of Total shows the percentage for each Quarter (group) of the grand total.

| Date: 3rd Quarter, 2008 | $166,979.31 | 19.114 % |
| Date: 4th Quarter, 2008 | $122,630.87 | 14.037 % |
| Date: 1st Quarter, 2009 | $155,350.57 | 17.783 % |
| Date: 2nd Quarter, 2009 | $174,878.94 | 20.018 % |
| Date: 3rd Quarter, 2009 | $90,765.06 | 10.390 % |
| Date: 4th Quarter, 2009 | $75,810.66 | 8.678 % |
| Date: 1st Quarter, 2010 | $87,051.07 | 9.964 % |
| Date: 2nd Quarter, 2010 | $21.00 | 0.002 % |
| Date: 3rd Quarter, 2010 | $125.00 | 0.014 % |

Grand Total (all pages) $873,612.48 100.000 %
In the Available Fields grid on the left are all numeric values that can be set as a benchmark. Highlight the desired field and click **Add Benchmark**.

The field is added to the Active Benchmark grid on the right. In the **Benchmark Value** field enter the benchmark amount.

Click **OK** to save the benchmark and return to the DataBook.
PowerFields - Benchmark

Benchmark values may now be used in filter options, in eXpressions or viewed as eValuate comparisons.
Control Values allow you to create end-user defined parameter value(s) for a DataBook filter or eXpression. Click Add to add a parameter.

Control Values give the creator of the DataBook the ability to build complex filters or expressions that may be locked down and the end-user prompted for only specific values. The end-user is prompted for all the control values on opening the DataBook.
PowerFields – Control Values

**Control Value:** Type the prompt to be presented to the end-user on entering the DataBook.

**Data Type:** Click the field to open the dropdown. Select the data type appropriate to the end-user response.

**Column:** If data type is DataBook Column, select the appropriate DataBook column. The values from the column selected here will be presented to the end-user for selection. If any other data type is selected this field will remain blank.

**Value:** Enter a default control value. The end-user may edit this value on entry to the DataBook. If this value is a DataBook Column data type, the field will contain an ellipses button. Click the button to select a value from a list of the column values. Otherwise enter a value directly into the field.

Click **OK** to save the Control Values and return to the Databook. Click **Refresh**.
PowerFields – Control Values

Using Control Values in Filters:
Drag the column used in the Control Value to the Filter panel.
Click the Operator dropdown and select the appropriate operator for the filter.
Select the Data Field value option.
From the Field dropdown, scroll to the bottom to find all Control Value options in bold red text.

When the DataBook is saved, select Ask for Control Values to display the Control Values prompt each time the DataBook is opened.
PowerFields – Control Values

Using Control Values in Filters:
When the DataBook is opened the Control Values prompt will be displayed. You may enter a new value by picking from a list or typing directly into the field.

When the DataBook opens it will be pre-filtered based on the value selected.
Using Control Values in Expressions:

When creating an expression, the Control Values are displayed as a value in the Column Name options and may be added to the expression.

The value of the field is determined by the end-user on opening the DataBook.

(see PowerFields – Expressions for more information on expressions)

When the DataBook is saved, select **Ask for Control Values** to display the Control Values prompt each time the DataBook is opened.

The end-user may change the value of the Control Value by clicking on the PowerFields menu option and selecting **Enter Control Values**. This option is only available when there are Control Values configured.