

POWERONE® TEMPLATES



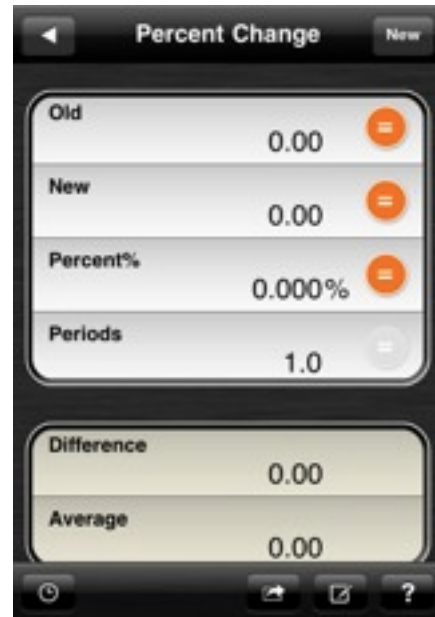
A DOCUMENT DESCRIBING HOW TO CREATE TEMPLATES

Templates

What is a template?

powerOne uses templates as its primary medium for showing calculations to its customers. Think of templates as simple forms that make what-if scenario calculation easy. They are a cross between a spreadsheet and a calculator, borrowing the simple two-column approach of spreadsheets and the ability to calculate any variable from calculators.

Using a template is simple: enter the known data into the rows you know then select equals to calculate the unknown. Colored equals buttons are unknown data; grayed equals buttons are known.



Template design

A template is made up of a series of interconnected rows. Each row can be designated by a combination of visible and hidden elements. For instance, the first row in the Percent Change image pictured above has the following visible elements:

- A label that indicates to the user what data is required for the row. (“Old” for the first row.)
- A value for the row. In this case, it shows 0.
- An equals button, which also implies it can be calculated. Not all rows can be calculated.

This template is made up of seven (7) rows, although you only see six: Old, New, Percent%, Periods, Difference and Average. The seventh row indicates to powerOne that there should be a new section starting with Difference.

Rows are interconnected via their formulas.

Additions

Besides the template itself, help is also available. Help includes both a template overview and optional examples for using the template. These are described in more detail later.

Defining a Row

As mentioned a template is defined by a combination of visible and hidden elements. Each element is defined below.

Labels

A label is a row description visible to the template's user. While it should be a short description it can be multiple words, use any kind of characters, punctuation or math symbols. Examples include:

- Net Present Value
- Interest / Year%
- Tablespoons [tbs]
- Food+Drink
- #Advances
- Net Revenue \$

Each row does not require a unique label name. A single template could have multiple rows that use the same label name.

Abbreviated Name (Variable)

While the Label is the way customers reference a row, variables are the method by which powerOne references a row internally. Variables are nothing more than abbreviated names for the row. Examples include:

- npv
- iy
- tbs
- f_d
- advances
- sep1

Variables can consist of US and International letters (capital or lower case 'a' through 'z'), numbers (0 through 9) or underscore (_). No spaces are allowed. The variable cannot start with a number or underscore. Variables can be as short as a single character or as long as desired. The variable names pi, e and i are not allowed.

Each row must have a unique variable name and case does matter. Note that the following variable names are all different:

- ABC
- AbC
- abC

While using the same name with different cases is possible it is not recommended. Instead you should append numbers to the end, such as the following:

- abc1
- abc2
- abc3

Data Types

The data type defines the expected input/output for a row. There are five available types: numbers, dates, lists, tables and separators.

- **Numbers:** The most common type of row. Number rows can display as real (decimal) numbers, fractions or feet-inch.
- **Dates:** Consist of days, months and years.
- **Lists:** List of options available in a selection box. Only one list option can be selected at a time.
- **Tables:** Group of related numbers.
- **Separators:** Designate the start of a new section within the template. Separators are used for visual or descriptive effect only. Separators can either be a blank space or have an optional Label.

Each type is entered using a different editor. Each is pictured below (ignore color differences):



number editor



date editor



table editor



list editor

Each Data Type has its own attributes. Each attribute is described below:

- **Default:** A default value. In many cases there is a common value used, for example 30 years for a US Mortgage or 1 for the number of Doors in the average room. In these cases you can designate a default value to minimize the amount of data to be entered. Numbers can have optional default values. Dates assume the default is today if one is not designated. Lists assume the first element is a default if one is not designated. Tables have no default.
- **Precision:** Numbers can be displayed in various formats and precisions. Format options include real (decimal) numbers, scientific notation, engineering notation,

fractions or feet-inches. Real numbers, scientific and engineering notation can be set to float (show as many decimal places as available), or 0 through 10 decimal places fixed. Fractions can be displayed based on the denominator. Fractions are first rounded to the desired precision and then reduced. Fraction denominator options include 1 through 4 places (x, xx, xxx, xxxx), 2, 4, 8, 16, 32, 64, 10, 100 or 1000. Feet-inches work the same way, where the fractions of inches are first rounded to 2, 4, 8, 16, 32 or 64 and then reduced. Real, scientific and engineering notation defaults to float; fractions to xxxx and feet-inch to 64. Tables also take this attribute but it is restricted to real numbers, float/0-10.

- **Table Columns:** Tables can either be one or two columns, both with an optional occurrences column. Table columns designates whether it is a one-column or two-column table. Most tables are one-column.
- **List Options:** Each list item is really two parts: a label and a value in real number format. The label is displayed to the user while the value is used internally to designate which list option was selected and can also be used as a value in a formula. For instance, a yes/no list may be designated as 1/0 values, respectively. Another example is rough/smooth surfaces with designated values of 400/250, respectively, which is then used in a formula to calculate the number of gallons needed to paint a surface.

More About Numbers

It is important to understand that numbers are always stored and used in formulas as a real number. powerOne has no concept of units. Feet-Inch, fraction, and real number are just display formats. It is displayed as feet and inches, or displayed as a fraction or displayed as a real number. For fraction, real, scientific and engineering, it is displayed as is but the real number equivalent. For feet-inch, it is displayed as a real number of inches.

For example, take the number 40.5. If you ask to display it as a real number with two decimal places it will display as 40.50. A real number with 5 decimal places will display as 40.50000. A fraction will display as $40\frac{1}{2}$ and a feet-inch number will display as $3'4\frac{1}{2}$.

Equation

The equation is used to calculate a row. A minimum of one row must have an equation. Equations combine row variables, constants (a value that doesn't change), and math functions. Over 200 math functions are available. A complete list can be found at: <http://www.infinitysw.com/help/functions>

For instance the following formula can be used to calculate area of a triangle:

$$\text{area} = \text{base} * \text{height} / 2$$

In this example, area, base and height are variables, 2 is a constant and * (multiplication) and / (division) are math functions.

Note that the same exact formula can also be used to calculate base and height, if so desired. In other words, given any two of area, base and height, powerOne can calculate the third. There is no need for you to algebraically manipulate the formula to isolate the unknown variable.

Sometimes there are equations that you wish to use in other equations. powerOne can handle this in a separate section called macros. Macros are variables that, instead of being assigned a value, are assigned an equation. When powerOne calculates, it physically substitutes that equation into the equation. For example:

$$\begin{aligned} \text{ab} &: \text{a+b} \\ \text{c} &: \text{ab*d} \end{aligned}$$

When powerOne calculates 'c' it will first substitute '(a+b)' for variable 'ab' and then will calculate so the equation becomes:

$$\text{c} = (\text{a+b}) * \text{d}$$

Note that the parentheses are added automatically to ensure order of operations are maintained.

Template Structure

Lesson 1: Getting Started

As discussed above, a template is made up of a series of interconnected rows. Creating a template is as simple as defining each row. Also mentioned above, there are five primary row types: numbers, dates, tables, lists and separators. The basic structure for a number row is as follows:

“Label” “default” variable : equation ::

Each of these elements were discussed in the section above. Note that it is recommended that labels and default values be in quotations. Also note that the variable and equation are separated with a single colon and that the entire row ends with the double-colon.

Let’s work with a simple example: percent change. The simplest equation for percent change is as follows:

$$\text{change} = (\text{old}-\text{new})/\text{old} * 100$$

A template to calculate “change” would look like this:

“Old Value” “” old ::

“New Value” “” new ::

“Percent Change%” “” change : change = (old-new)/old * 100 ::

Notice a few things:

1. old and new have no equation because we are only calculating change. When there is no equation we don’t need the leading single colon but still need the double-colon.
2. Since there is no default values we still include the empty quotes (“”) where the default would go but don’t include a value.
3. change does have an equation so includes the colon after the variable. Also note that the label for change, “Percent Change%” has a percentage symbol on the end. powerOne will display a % after the number in these cases.

Lesson 2: More About Equations

What if you wanted to also be able to calculate old and new? Easy! Just add the same equation we used for change to the other rows:

“Old Value” “” old : change = (old-new)/old * 100 ::

“New Value” “” new : change = (old-new)/old * 100 ::

“Percent Change%” “” change : change = (old-new)/old * 100 ::

Because each row uses the same variables, these rows are interconnected and can automatically calculate when any two of three variables are entered. In this way, you can pick and choose which rows have formulas and which don't.

Note that powerOne is pretty smart about doing math. If the equals sign is missing, powerOne assumes that it is that row's variable equals the equation. For change above, the “change =” portion of the equation is redundant. It is completely up to you as to whether you wish to include it or not. In other words, this is exactly the same as the equation above:

“Old Value” “” old : change = (old-new)/old * 100 ::

“New Value” “” new : change = (old-new)/old * 100 ::

“Percent Change%” “” change : (old-new)/old * 100 ::

Lesson 3: Multiple Equations

Let's say we would also like to know the difference between the Old Value and New Value. We need another row in the template!

“Old Value” “” old : change = (old-new)/old * 100 ::

“New Value” “” new : change = (old-new)/old * 100 ::

“Percent Change%” “” change : (old-new)/old * 100 ::

“Difference” “” difference : new - old ::

Now when we enter old and new, difference will calculate also.

Lesson 4: Display Formats

As discussed in the Precision section above, numbers can be displayed in a variety of formats. Each of old, new, change and difference in our example above are real numbers. The way that template is designed right now, all results will be displayed as real numbers with as many decimal places as can be shown by powerOne (called float number format). To change the format, we need to add precision to each of our rows:

```
"Label" "default" variable, precision : equation ::
```

Precision follows the variable name and the two are separated with a comma. Let's say instead we would like to see old, new and difference with two places and change with three:

```
"Old Value" "" old, 2 : change = (old-new)/old * 100 ::
```

```
"New Value" "" new, 2 : change = (old-new)/old * 100 ::
```

```
"Percent Change%" "" change, 3 : (old-new)/old * 100 ::
```

```
"Difference" "" difference, 2 : new - old ::
```

There are five number formats, as described in the Precision section above. Real decimal numbers are designated with a number 0 - 10. Scientific notation is set with an "S" followed by 0 - 10 (S0, S1, S2, etc.). Engineering notation is set with an "E" followed by 0 - 10 (E0, E1, E2, etc.). Fractions are set with an "FR" followed by the denominator options: FRx, FRxx, FRxxx, FRxxxx means 1-4 number places; FR2, FR4, FR8, FR16, FR32, FR64, FR10, FR100 and FR1000 means the number is rounded to that number of places (2-64, 10-1000). After rounding the fraction is reduced to its lowest common denominator. Feet-Inch is also available with the designation "FT". Options include FT2, FT4, FT8, FT16, FT32, and FT64. Remember, feet-inch numbers are stored internally as a real number of inches and displayed as feet and inches when requested.

Lesson 5: Separators

Sometimes it is nice to add a little space between rows. It makes the template easier to read and groups data in logical ways. In the Percent Change example, old, new and change are all calculated together while difference is just an extra piece of data.

Separators can either be blank or include a label that indicates what the section is about. The options are as follows:

```
"Label" ::  
::
```

The first option includes a label that defines the section. Template like Mortgage include separator labels. The second option includes no label. We will use the second in our Percent Change example:

```
"Old Value" "" old : change = (old-new)/old * 100 ::  
"New Value" "" new : change = (old-new)/old * 100 ::  
"Percent Change%" "" change : (old-new)/old * 100 ::  
::  
"Difference" "" difference : new - old ::
```

More!

More Help

This document is meant as a getting started tutorial on how to create advanced templates. Infinity Softworks provides additional, detailed information at its web site:

<http://www.infinitysw.com/help/create>

Other Data Types

These examples used all number and separator rows. Most templates consist of these two rows. In addition, a row can be a date, a table, a list or a text string. All of these types are detailed on the Infinity Softworks' web site:

<http://www.infinitysw.com/help/create>

More Examples

Almost every template Infinity Softworks offers within its apps or in the Library has the source code available to review and copy. Please click the edit button (square box with pencil) to see the source for any templates.

Creating Help

Overview and Examples

Help is in two sections: 1) overview of the template, its rows and any additional information; and 2) step-by-step examples.

The description text should be in the following format:

Description

Rows:

- Row1: description

- Row2: description

...

Other Information:

description

The examples should follow this format:

Example Name: (if more than one)

Word problem

- Row1: what should be entered

- Row2: what should be entered

...

Select '=' on ____ row. Describe the answer the customer should see if they follow your example.

Feel free to add as many examples and be as descriptive as you feel is needed for a customer to learn how to use a template, assuming they generally know how to enter data and calculate. We highly recommend reviewing existing template help to see how we write them.

Help Extras

Help content can link to external web pages. Just provide the text and accompanying web site URL. In addition, images can be embedded in the help text. While you may want to embed the image in the help text in your document so we know where it should go (or indicate it in some other way) please provide the images outside of your document as well so we can upload them as appropriate. For details, see this web page:

http://www.infinitysw.com/help/create_helptext