

Users sometimes have two external lists of values. And there is often a need to approximate a value from one list, given a value from the other list. To address this need, we created a library type called a "Lookup Table".

## Create A Lookup Table Library

To create a "Lookup Table", click the System Setup button at the top of the screen, then click the Library tab. Use the Add button to create a new library item of type "Lookup Table". As you add new XY pairs of values, a corresponding data point is added to the graph. The graph allows you to visualize your data – and spot trends and outliers.

If you don't check "Use best fit line" then the values for X and Y must all be either increasing or decreasing. But the lists can go in opposite directions.

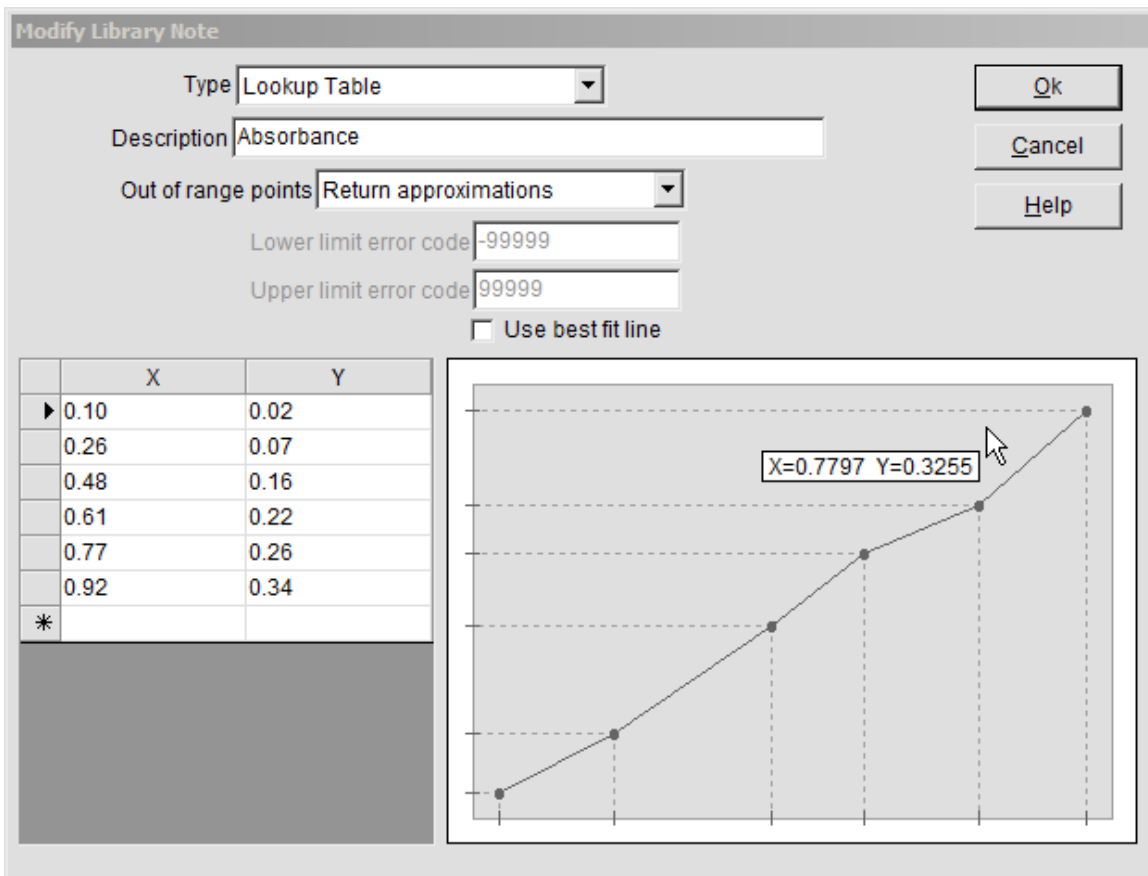


Figure 1: A "Lookup Table" Library

## lib.TableLookupX and lib.TableLookupY

Two variables return approximations from a "Lookup Table" library. The format for them is: lib.TableLookupY (library name; givenX) and lib.TableLookupX(library name; givenY). Note that semicolons are used as

parameter separators. These variables require that the table contains at least two data points. In the above example, if you want to approximate X when Y is 0.29, you'd use: `[lib.TableLookupX(Absorbance; 0.29)]`

Notice the square brackets around the entire variable. TrueChem variables must be enclosed in square brackets. So if you are using, say, `lib.TableLookupX` in a Result script, your script may look similar to this:

```
Result = [lib.TableLookupX(Absorbance; [INP1])]
```

### Limit Error Codes

The Lookup variables make use of *limit error codes* as defined on the library creation screen. See Figure 1. If your given value is either smaller or larger than all the values in its list, we deem it "out of range". There are two ways for the Lookup variables to generate the return value when the given value is "out of range":

1. the return value can be approximated, or
2. the return value can be a pre-defined *limit error code*. See Figure 1.

### Best Fit Line

Checking the "Use best fit line" displays a least square best fit line through the data points. And `[lib.TableLookupX]` and `[lib.TableLookupY]` will return approximations based on that line.

### Temporarily replace a table value

When using `[lib.TableLookupX]` or `[lib.TableLookupY]`, you can temporarily replace table values by using parameters like `X2=33` (which temporarily replaces the 2<sup>nd</sup> X-value with 33) or `Y4=17.4` (which replaces the 4<sup>th</sup> Y-value with 17.4).

Example:

```
[lib.TableLookupY(Absorbance; 0.29; X2=0.28; Y6=0.36)]
```

### Temporarily create a table

When using `[lib.TableLookupX]` or `[lib.TableLookupY]`, you can leave off the table name to temporarily create your own table. The return value will be based on the best fit line.

Example,

```
[lib.TableLookupY( ; 3.4; X1=1.6; Y1=10; X2=2.9; Y2=12)]
```

This returns an approximation for Y when X=3.4, based on the best fit line through the two points specified in the parameter list.