# Command Reference: ReadTableFromExcel()

## Read a cell range from a Microsoft Excel file and create a new Table

Version 10.18.00, 2012-02-25

The `ReadTableFromExcelFile()` command reads a table from a Microsoft Excel file, more specifically from a worksheet in an Excel workbook file. A contiguous block of cells (rectangle) must be specified in one of the following ways:

- Specify a range of cells using Excel address notation (e.g., `A1:D10`) (TODO – figure out if worksheet can be specified in this address, in which case the `Worksheet` parameter is not required).
- Specify the name of an Excel named range.
- Specify a table name (essentially a named range).

Table column types (number, text, etc.) are determined from the cells in the first data row being read (NOT the column name row) – data types must be consistent for all cells in a column, although blanks are allowed. Table column names are determined according to the `ExcelColumnNames` command parameter.

TSTool uses the Apache POI software, version 3.9 (http://poi.apache.org) to read the Excel file and consequently functionality is constrained by the features of that software package. The software reads and writes Excel files. It does not communicate with a running Excel program, as does other software tools (for example IronPython using Excel interoperability libraries). POI does not fully implement Excel functionality and consequently some formula capabilities are not available, which will generate errors getting values for some cells. One solution, for example to create test data in Excel, is to copy cells with "paste special" and then paste the values. It is expected that updates to POI will continue to add more formula support.

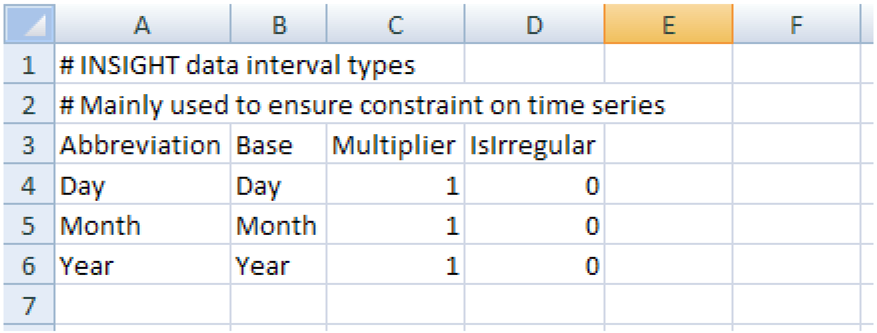
Table columns must contain consistent data types (all strings, all numeric, etc.). The following table describes how column types are determined and data values are transferred to the table. Column type determination uses the first data row in the specified address range. If a column is determined to be a type and then cell values in the column are different, conversions are made to maintain the intent of the values if possible. For example, a Boolean value stored in a cell will get converted to 1.0 if the table column has been determined to be for double precision numbers. Errors in processing cells may result in empty cell values in the output table.

**Excel Data Type Conversion to Table**

| Excel Cell Format ("Number Category" | Conversion from Excel to TSTool Table |
|---|---|
| Number:<br>• General<br>• Number<br>• Currency<br>• Accounting<br>• Percentage<br>• Fraction<br>• Scientific<br>• Special<br>• Custom | • If Excel cell is internally a "numeric", convert to a double-precision number, where the format "Decimal places" is used in the TSTool table for formatting. The number of decimal places in Excel is fixed for some of the number categories shown on the left (e.g., Special=Zip Code). Excel internally stores integers as numbers with zero decimals. Need to figure out how to get the Excel cell formatting number of decimals to similarly set in the output table – but DO NOT assume zero decimals should convert to an integer.<br>• See the `ExcelIntegerColumns` parameter, which specifies the output table to use integers.<br>• If Excel cell is internally a "Boolean", convert to an integer having |

| Excel Cell Format (“Number Category” | Conversion from Excel to TSTool Table |
|---|---|
| | values 0 or 1. Need to evaluate having a parameter ExcelBooleanColumns to transfer to a Boolean column in the output table. Excel seems to handle Booleans as text with values True or False. |
| Date:<br>• Date<br>• Time | TSTool does not generally deal with only time and therefore implementation is limited. The POI library does not seem to have all date/time functions implemented. |
| Text:<br>• Text | Converts to a string. |
| Blank | • Treated as Text (may in the future scan down the column to determine data type from first non-blank cell).<br>• Blank cells found once the column type is determined are set to empty strings in text columns, and null in number and date columns. |
| Error | • Treated as Text (may in the future scan down the column to determine data type from first non-error cell).<br>• Blank cells found once the column type is determined are set to empty strings in text columns, and null in number and date columns. |
| Formula | Expanded internally and the resulting cell value is set in the output table. POI does not support all formulas and errors may be generated, which result in empty output table cells. |

Consider the following Excel worksheet example, which is equivalent to a comma-separated-value (CSV) file that has comments at the top and four columns:



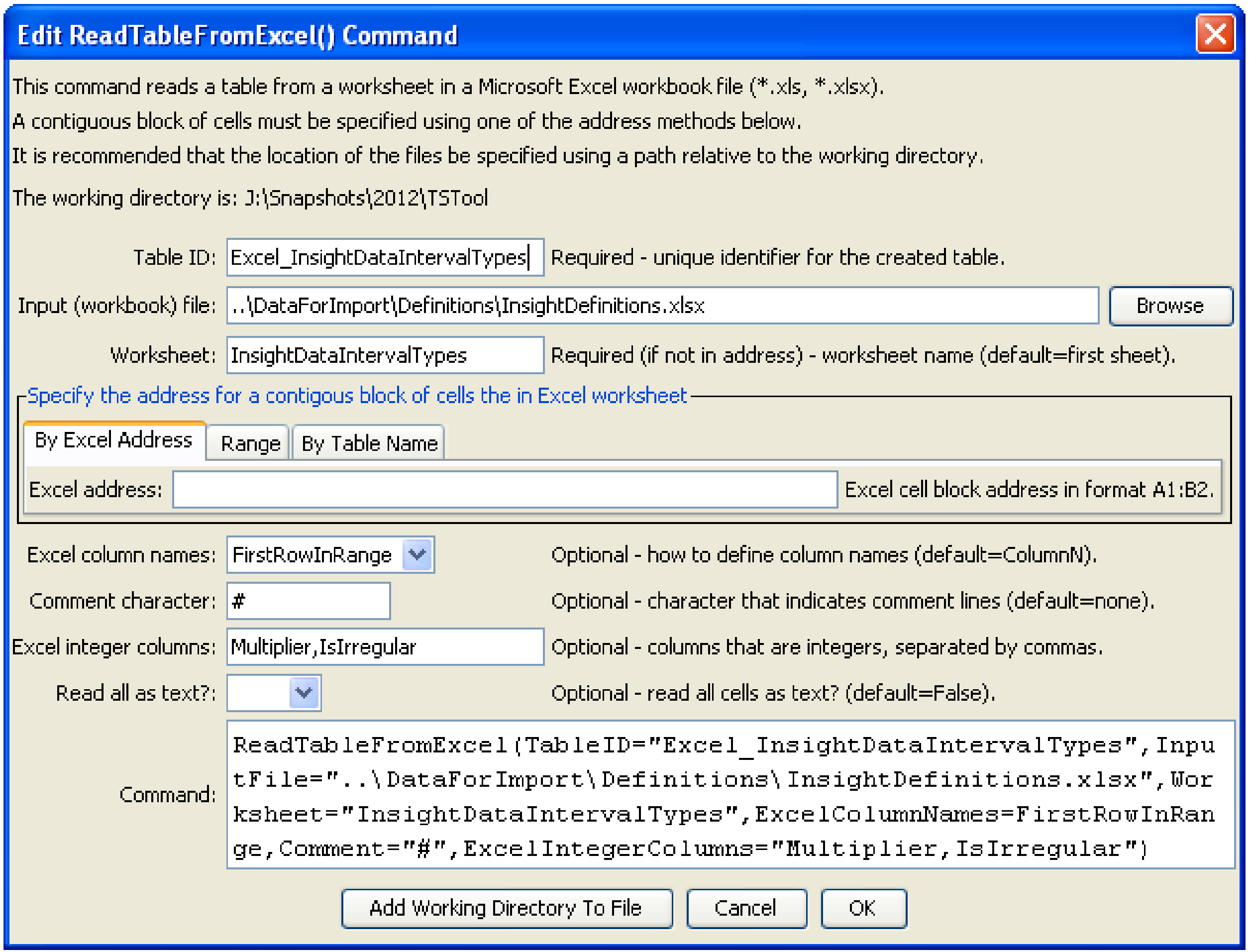
| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | # INSIGHT data interval types | | | | | |
| 2 | # Mainly used to ensure constraint on time series | | | | | |
| 3 | Abbreviation | Base | Multiplier | IsIrregular | | |
| 4 | Day | Day | 1 | 0 | | |
| 5 | Month | Month | 1 | 0 | | |
| 6 | Year | Year | 1 | 0 | | |
| 7 | | | | | | |

ReadTableFromExcel_SheetComments

**Example Excel Workshet With Comments, Column Names, and Text and Integer Columns**

Although it is possible to use comments in Excel (annotation on cells), these comments cannot be saved in simple text files like CSV files. Consequently, for transparency and automation of a full process, embedding comments in the worksheet may make sense. Note also that the numeric cells are formatted as type “Number” with 0 decimals in Excel. Internally, Excel does not have an integer data type and consequently it is difficult for the ReadTableFromExcel() command to know when to convert a zero-decimal number in Excel to a floating point or integer number in the output table (it therefore defaults to a floating point number in output). To make this conversion more explicit, use the ExcelIntegerColumns command parameter. The comment lines in the above example will be ignored in determining the headings, and any data rows that have a first cell value starting with the comment character will be ignored.

The following dialog is used to edit the command and illustrates the syntax for the command when reading the above Excel worksheet.



Edit ReadTableFromExcel() Command

This command reads a table from a worksheet in a Microsoft Excel workbook file (*.xls, *.xlsx).
A contiguous block of cells must be specified using one of the address methods below.
It is recommended that the location of the files be specified using a path relative to the working directory.

The working directory is: J:\Snapshots\2012\TSTool

Table ID: Excel_InsightDataIntervalTypes Required - unique identifier for the created table.

Input (workbook) file: ..\DataForImport\Definitions\InsightDefinitions.xlsx

Worksheet: InsightDataIntervalTypes Required (if not in address) - worksheet name (default=first sheet).

Specify the address for a contigous block of cells the in Excel worksheet

Excel address: Excel cell block address in format A1:B2.

Excel column names: FirstRowInRange Optional - how to define column names (default=ColumnN).

Comment character: # Optional - character that indicates comment lines (default=none).

Excel integer columns: Multiplier,IsIrregular Optional - columns that are integers, separated by commas.

Read all as text?: Optional - read all cells as text? (default=False).

Command:

```
ReadTableFromExcel(TableID="Excel_InsightDataIntervalTypes",InputFile="..\DataForImport\Definitions\InsightDefinitions.xlsx",Worksheet="InsightDataIntervalTypes",ExcelColumnNames=FirstRowInRange,Comment="#",ExcelIntegerColumns="Multiplier,IsIrregular")
```

ReadTableFromExcel

**ReadTableFromExcel() Command Editor**

The command syntax is as follows:

```
ReadTableFromExcelFile(Parameter=Value,…)
```

**Command Parameters**

| Parameter | Description | Default |
|---|---|---|
| `TableID` | Identifier to assign to the table that is read, which allows the table data to be used with other commands. | None – must be specified. |
| `InputFile` | The name of the Excel workbook file (*.xls* or *.xlsx*) to read, as an absolute path or relative to the command file location. | None – must be specified. |
| `Worksheet` | The name of the worksheet in the workbook to read. Currently this is required if a specific sheet is read but in the future it may be made optional because | Read the first worksheet. If no address parameter is specified, read the entire worksheet. |

| Parameter | Description | Default |
|---|---|---|
| | the sheet can be determined from named range and table names (global resources in the workbook) and absolute Excel addresses that include the sheet name. | |
| ExcelAddress | Indicates the block of cells to read into the table, using Excel address notation (e.g., A1:D10). | Must specify address using one of available address parameters. |
| ExcelNamedRange | Indicates the block of cells to read into the table, using an Excel named range. | Must specify address using one of available address parameters. |
| ExcelTableName | Indicates the block of cells to read into the table, using an Excel named range. | Must specify address using one of available address parameters. |
| ExcelColumnNames | Indicate how to determine the column names for the table, one of:<br>• ColumnN – column name will be Column1, Column2, etc.<br>• FirstRowInRange – column names are taken from the first non-comment row in the address range<br>• RowBeforeRange – column names are taken from the first non-comment row before the address range | ColumnN, or FirstRowInRange when ExcelTableName is specified? |
| Comment | Specify the character that if found at the start of the first column in a row (not just the specified address range) indicates that the row is a comment and can be ignored in transferring data to the output table. Comments are particularly useful when processing entire data sheets. | No comments are used. |
| ExcelIntegerColumns | Indicate the names of columns (separated by commas) that should be treated as integer columns in the output table. | Numeric columns are treated as double-precision values in the output table. |
| ReadAllAsText | Indicate with True or False whether all columns in the Excel address block should be treated as text columns. | False – set table column types using the first data row |